

Eryone TMC2209 with Thinker V2 Board Guide-UART+Sensorless

@Thanks for [Marco Martin's](#) help

Part:

Hardware

Firmware

Software (Calibrate)

Hardware:

-Prepare

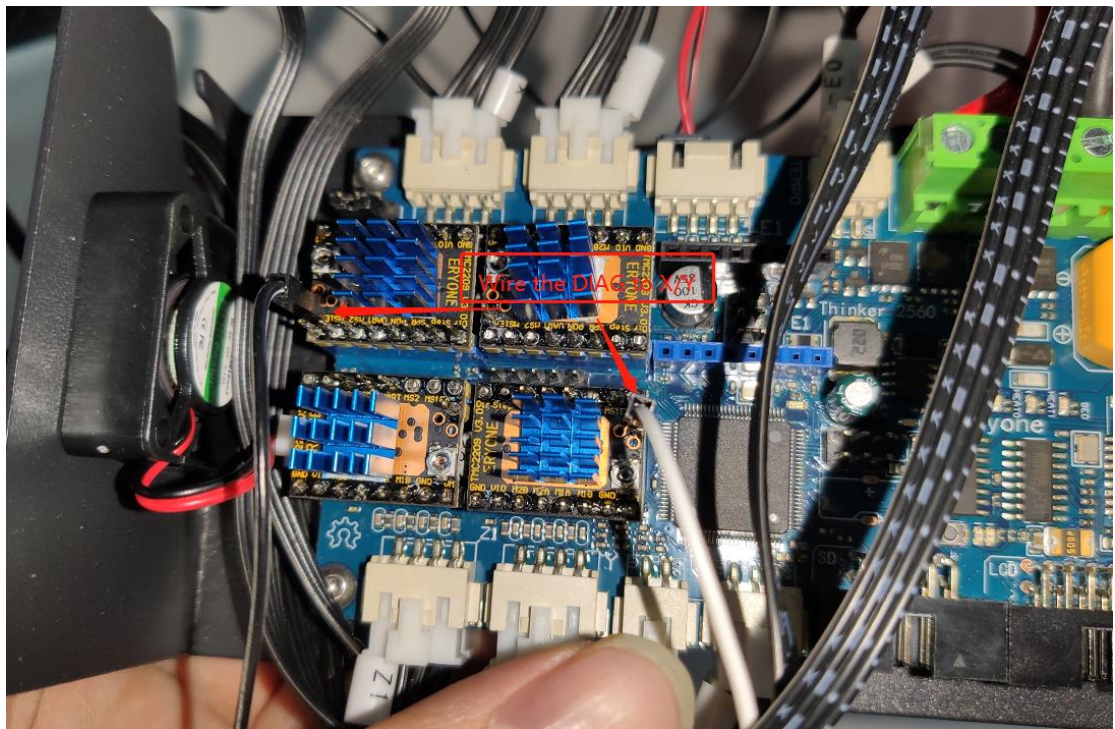
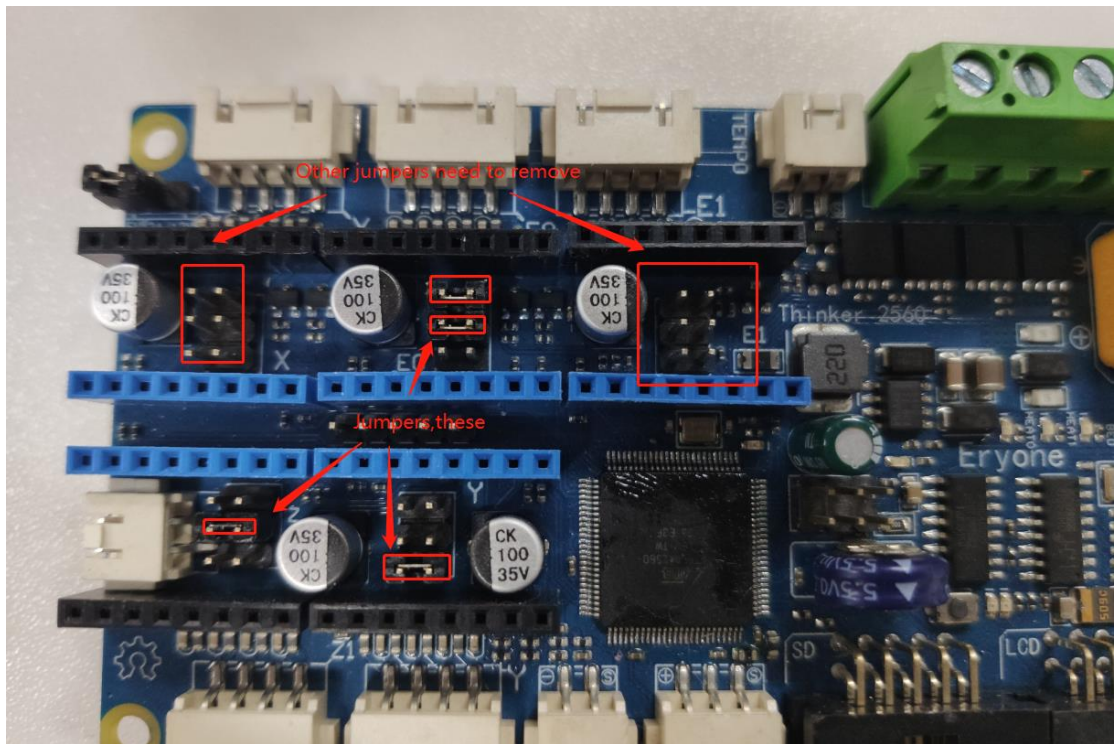
1. Eryone TMC2209 4 pcs
2. 2x 10cm extent cable
3. A box of candy

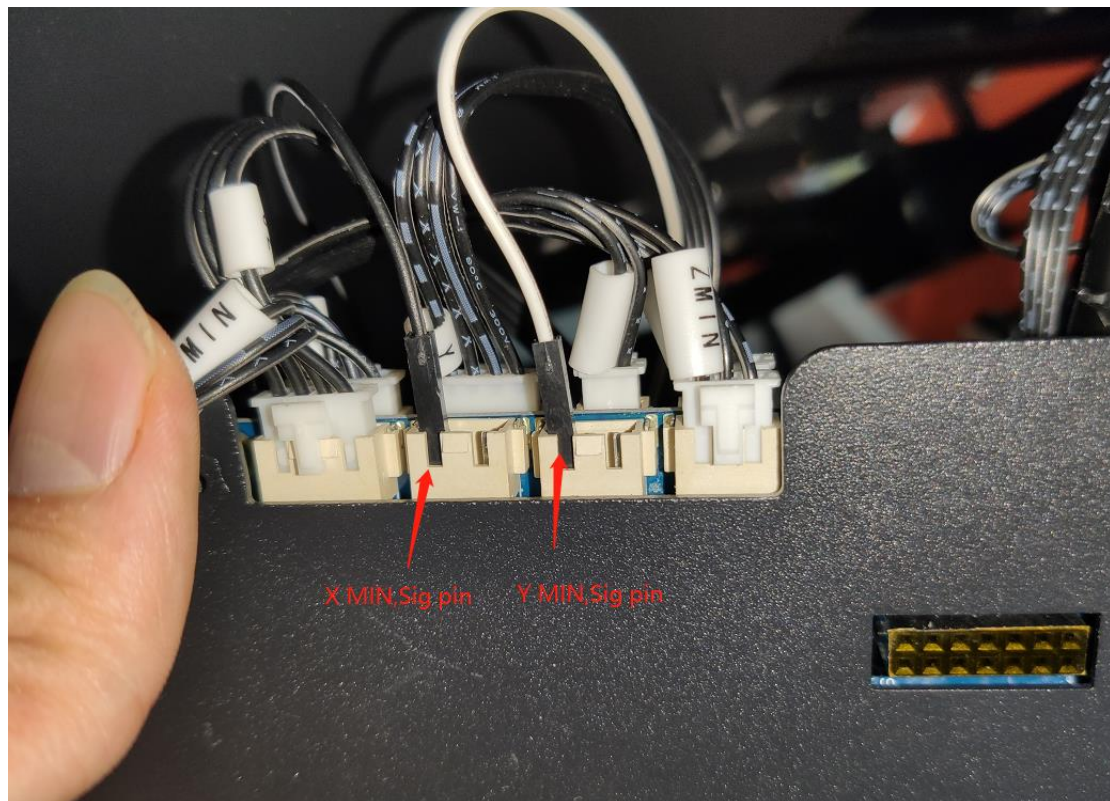
-Setup the TMC2209 to thinker V2 board

- 1) Open the board case

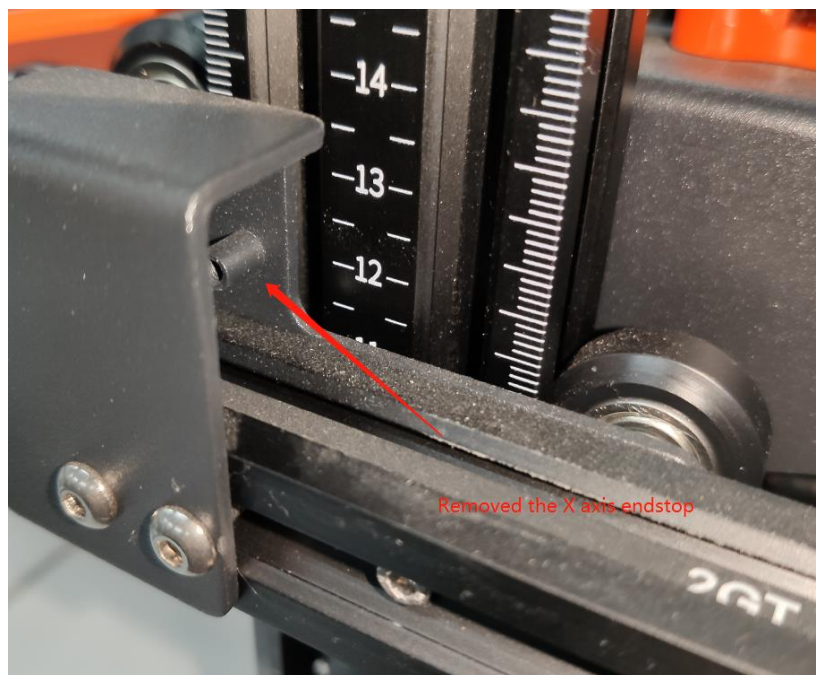


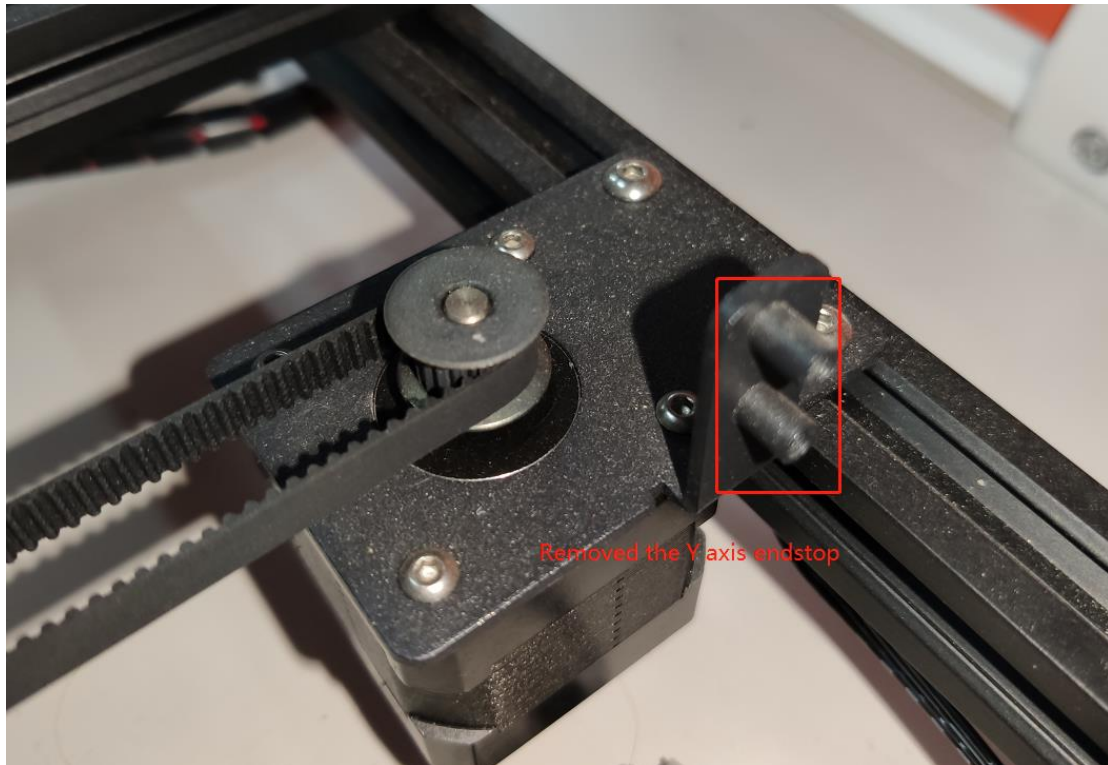
- 1) Remove all the tmc2208 off
- 2) Set the jumpers for TMC2209
- 3) Use the cable connect the DIAG pin of X/Y sensorless homing





4) Remove the endstop of X/Y





Then hardware setup finished. Congratulations, you did it! Not difficult, yes?

Firmware

The firmware based on the Marlin2.0.x for thinker V2" [Thinker V2_Marlin2.0_20200317.zip](#)"

<https://www.eryone.com/forum/viewtopic.php?f=6&t=5>

1) pins_rambo.h

Folder:

Marlin>src>pins>Rambo>

Add these lines at last

```
#if HAS_TRINAMIC
/**
 * TMC2208/TMC2209 stepper drivers
 *
 * Hardware serial communication ports.
 * If undefined software serial is used according to the pins below
 */
#define X_HARDWARE_SERIAL Serial1
#define Y_HARDWARE_SERIAL Serial1
#define Z_HARDWARE_SERIAL Serial1
#define E0_HARDWARE_SERIAL Serial1
// #define E1_HARDWARE_SERIAL Serial1

// Reduce baud rate to improve software serial reliability
#define TMC_BAUD_RATE 19200
#endif
```

2) configuration.h

Folder: Marlin

Invert the endstop logic of X/Y/Z

eg: Before (If you find the Z move at the opposite direction, please try to invert the Z min logic)

```
// Mechanical endstop with COM to ground and NC to Signal uses "false" here (most common setup).
//If your sensor is NPN sensor,try to set Z_MIN_ENDSTOP_INVERTING true,Z_MIN_PROBE_ENDSTOP_INVERTING true
//If your sensor is PNP sensor,try to set Z_MIN_ENDSTOP_INVERTING false,Z_MIN_PROBE_ENDSTOP_INVERTING false
#define X_MIN_ENDSTOP_INVERTING true//false // set to true to invert the logic of the endstop.
#define Y_MIN_ENDSTOP_INVERTING true//false // set to true to invert the logic of the endstop.
#define Z_MIN_ENDSTOP_INVERTING false//true // set to true to invert the logic of the endstop.
#define X_MAX_ENDSTOP_INVERTING false // set to true to invert the logic of the endstop.
#define Y_MAX_ENDSTOP_INVERTING false // set to true to invert the logic of the endstop.
#define Z_MAX_ENDSTOP_INVERTING false // set to true to invert the logic of the endstop.
#define Z_MIN_PROBE_ENDSTOP_INVERTING false // set to true to invert the logic of the probe.
```

After:

```
// Mechanical endstop with COM to ground and NC to Signal uses "false" here (most common setup).
//If your sensor is NPN sensor,try to set Z_MIN_ENDSTOP_INVERTING true,Z_MIN_PROBE_ENDSTOP_INVERTING true
//If your sensor is PNP sensor,try to set Z_MIN_ENDSTOP_INVERTING false,Z_MIN_PROBE_ENDSTOP_INVERTING false
#define X_MIN_ENDSTOP_INVERTING false // set to true to invert the logic of the endstop.
#define Y_MIN_ENDSTOP_INVERTING false // set to true to invert the logic of the endstop.
#define Z_MIN_ENDSTOP_INVERTING true//false // set to true to invert the logic of the endstop.
#define X_MAX_ENDSTOP_INVERTING false // set to true to invert the logic of the endstop.
#define Y_MAX_ENDSTOP_INVERTING false // set to true to invert the logic of the endstop.
#define Z_MAX_ENDSTOP_INVERTING false // set to true to invert the logic of the endstop.
#define Z_MIN_PROBE_ENDSTOP_INVERTING true//false // set to true to invert the logic of the probe.
```

Set the driver type to 'TMC2209'

```
#define X_DRIVER_TYPE  TMC2209//A4988
#define Y_DRIVER_TYPE  TMC2209//A4988
#define Z_DRIVER_TYPE  TMC2209//A4988
//#define X2_DRIVER_TYPE A4988
//#define Y2_DRIVER_TYPE A4988
//#define Z2_DRIVER_TYPE A4988
//#define Z3_DRIVER_TYPE A4988
#define E0_DRIVER_TYPE TMC2209//A4988
//#define E1_DRIVER_TYPE A4988
//#define E2_DRIVER_TYPE A4988
//#define E3_DRIVER_TYPE A4988
//#define E4_DRIVER_TYPE A4988
//#define E5_DRIVER_TYPE A4988
```

3) Configuration_adv.h

Folder: Marlin

Modify these lines:

Before

```
2020 #define X_SLAVE_ADDRESS 0
2021 #define Y_SLAVE_ADDRESS 0
2022 #define Z_SLAVE_ADDRESS 0
2023 #define X2_SLAVE_ADDRESS 0
2024 #define Y2_SLAVE_ADDRESS 0
2025 #define Z2_SLAVE_ADDRESS 0
2026 #define Z3_SLAVE_ADDRESS 0
2027 #define E0_SLAVE_ADDRESS 0
2028 #define E1_SLAVE_ADDRESS 0
2029 #define E2_SLAVE_ADDRESS 0
2030 #define E3_SLAVE_ADDRESS 0
2031 #define E4_SLAVE_ADDRESS 0
2032 #define E5_SLAVE_ADDRESS 0
```

After:

```
#define X_SLAVE_ADDRESS 0
#define Y_SLAVE_ADDRESS 1
#define Z_SLAVE_ADDRESS 2
// #define X2_SLAVE_ADDRESS -1//0
// #define Y2_SLAVE_ADDRESS -1//0
// #define Z2_SLAVE_ADDRESS -1//0
// #define Z3_SLAVE_ADDRESS -1//0
#define E0_SLAVE_ADDRESS 3
// #define E1_SLAVE_ADDRESS -1//0
// #define E2_SLAVE_ADDRESS -1//0
// #define E3_SLAVE_ADDRESS -1//0
// #define E4_SLAVE_ADDRESS -1//0
// #define E5_SLAVE_ADDRESS -1//0
```

Set voltage(This step can skip):

```
#define CHOPPER_TIMING CHOPPER_DEFAULT_24V
/**
```

Enable the sensor homing:

You can set the sensitivity then, so start with 60~70 is good.

```
2134 #define SENSORLESS_HOMING // StallGuard capable drivers only
2135
2136 /**
2137  * Use StallGuard2 to probe the bed with the nozzle.
2138  *
2139  * CAUTION: This could cause damage to machines that use a lead screw or
2140  *           to move the Z axis. Take extreme care when attempting to enable
2141  */
2142 // #define SENSORLESS_PROBING // StallGuard capable drivers only
2143
2144 #if EITHER(SENSORLESS_HOMING, SENSORLESS_PROBING)
2145   // TMC2209: 0...255. TMC2130: -64...63
2146   #define X_STALL_SENSITIVITY 60//8
2147   #define X2_STALL_SENSITIVITY X_STALL_SENSITIVITY
2148   #define Y_STALL_SENSITIVITY 70//8
2149   // #define Z_STALL_SENSITIVITY 8
2150   // #define SPI_ENDSTOPS // TMC2130 only
2151   // #define HOME_USING_SPREADCYCLE
2152   // #define IMPROVE_HOMING_RELIABILITY
2153 #endif
2154
```

Enable the TMC-DEBUG

```
2161 /**
2162  * Enable M122 debugging command for TMC stepper drivers.
2163  * M122 S0/1 will enable continous reporting.
2164  */
2165 #define TMC_DEBUG
2166
```

Done, then you should upload the firmware to your printer

How to upload the firmware, the Link:

<https://www.youtube.com/watch?v=b2D4I9Yxejw>

by Tom