

Manuel BIGTREETECH TMC2208 V3.0 I3D Service

avec l'aimable autorisation de Shenzhen BIGTREE technology co., LTD

BIGTREETECH TMC2208 V3.0

Driver pour moteur pas à pas



Manuel BIGTREETECH TMC2208 V3.0

I3D Service

avec l'aimable autorisation de Shenzhen BIGTREE technology co., LTD

Caractéristiques des TMC2208 V3.0 :

Courant nominal de 1.2A avec un courant maximum de 2A

Jusqu'à 256 micro-pas par interpolation

Mode StealthChop2 pour des impressions à basse vitesse et avec des accélérations modérées

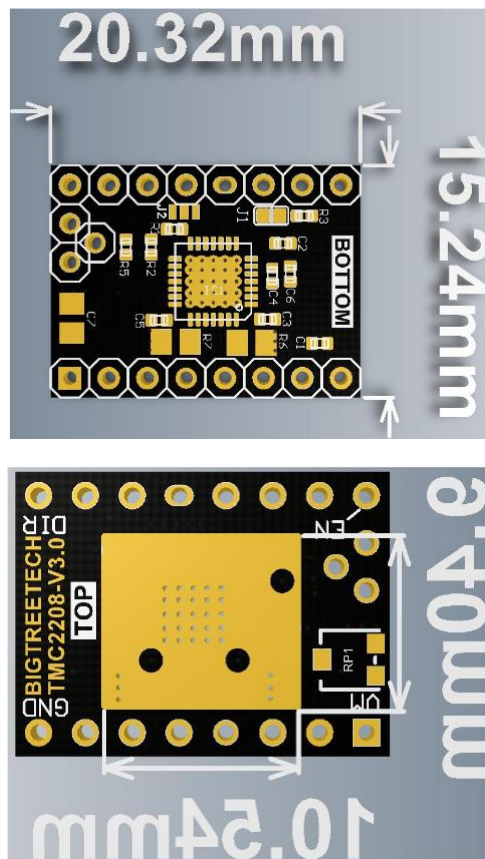
Mode spreadCycle pour des impressions à haute vitesse et avec des accélérations importantes

Faible résistance RDS(on) : LS 280mΩ et HS 290mΩ (à 25°C)

Tension d'alimentation comprise entre 12V et 24V DC

Communication UART par une seule entrée pour les options de configuration avancées

Générateur d'impulsions intégré



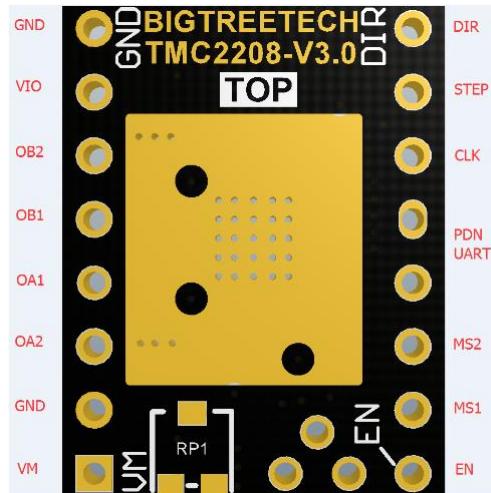
Manuel BIGTREETECH TMC2208 V3.0

I3D Service

avec l'aimable autorisation de Shenzhen BIGTREE technology co., LTD

Mode de travail et réglage par potentiomètre

1. Mode STEP/DIR (réglage du courant par potentiomètre)



Choix des micro-pas :

MS1	MS2	Pas	Interpolation	Mode
GND	GND	8	Oui jusqu'à 256	Stealthchop2
VIO	GND	2	Oui jusqu'à 256	Stealthchop2
GND	VIO	4	Oui jusqu'à 256	Stealthchop2
VIO	VIO	16	Oui jusqu'à 256	Stealthchop2

Référence des courants de travail :

```

VRef 0...2.5V (0.11 Ohm sense resistor)
>=2.50V 100% - 1.77A RMS
 1.25V  50% - 0.88A RMS
 0.50V  20% - 0.35A RMS
    
```

```

EN (with pull-up)
GND driver enabled
VCC driver disabled
    
```

```

PDN/UART (with pull-down)
GND automatic standstill current reduction
VCC automatic standstill power down disable
optional UART interface
    
```

```

CLK (with pull-down)
GND internal clock
optional supply external clock
    
```

Manuel BIGTREETECH TMC2208 V3.0

I3D Service

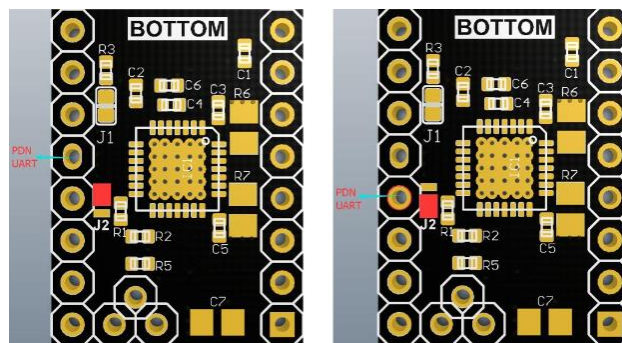
avec l'aimable autorisation de Shenzhen BIGTREE technology co., LTD

2. Mode UART :

A Noter : Les TMC2208 V3.0 de BIGTREETECH sont vendus soit en mode DIY (STEP/DIR) ou UART. Ceci facilite le montage car vous n'aurez rien à souder. Les TMC seront déjà prêt à fonction de votre commande donc en STEP/DIR ou UART.

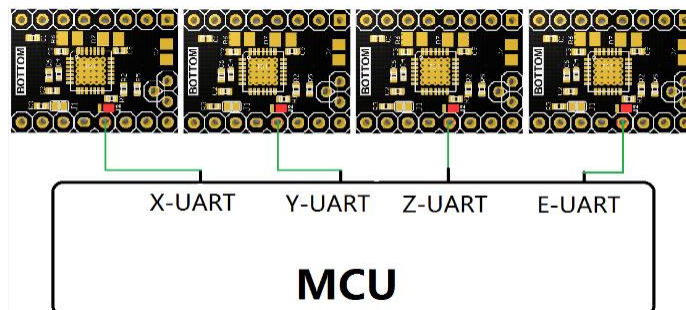
Les utilisateurs ayant acheté des TMC2208 V3.0 en DIY (STEP/DIR) et voulant les passer en mode UART doivent faire l'opération suivante :

Il faudra souder la résistance J2 comme sur l'image suivante (mode STEP/DIR à gauche et mode UART à droite)



Les avantages du mode UART :

- Le courant moteur peut être réglé directement dans le firmware
- Les micro-pas peuvent être réglés dans le firmware (jusqu'à 256 micro-pas)
- Combinaison des micro-pas réels et interpolés pour atteindre un couple maximal
- Le firmware peut changer dynamiquement les modes moteur stealthChop2 et spreadCycle via le mode UART
- Lorsqu'il n'y a pas de mouvement du moteur, le courant de veille peut être réduit dynamiquement



Manuel BIGTREETECH TMC2208 V3.0

I3D Service

avec l'aimable autorisation de Shenzhen BIGTREE technology co., LTD

Instruction pour réglage du courant par potentiomètre (mode STEP/DIR) :

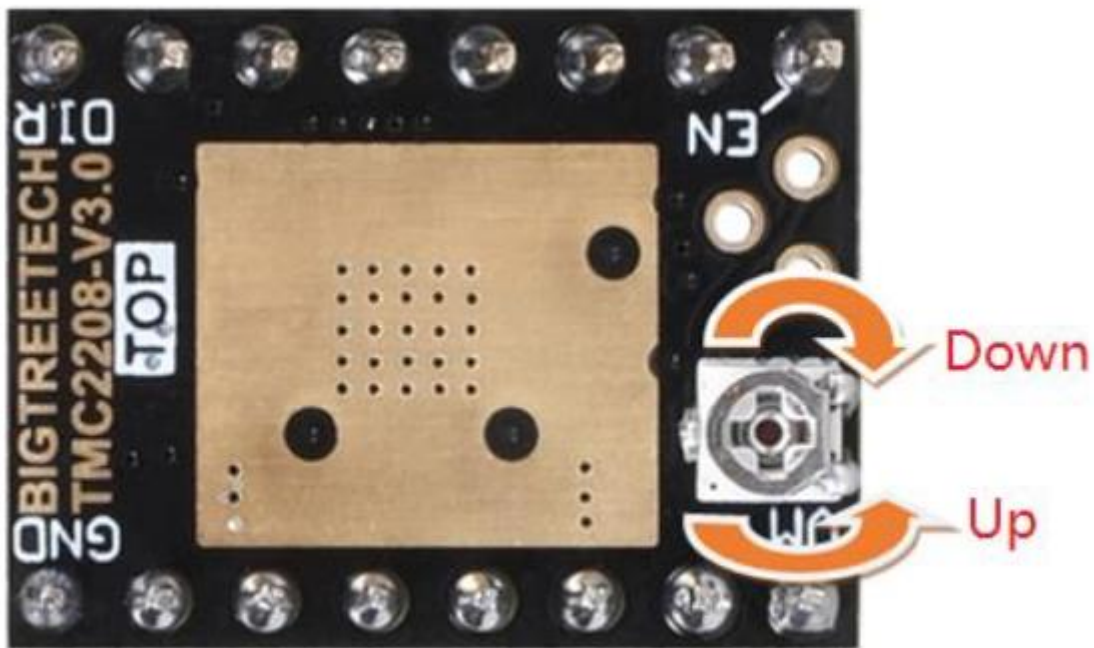
Rotation du potentiomètre dans le sens des aiguilles d'une montre : réduction de la Vref, réduction du courant dans le moteur

Rotation du potentiomètre dans le sens inverse des aiguilles d'une montre : augmentation de la Vref, augmentation du courant dans le moteur

La tension de la Vref ne peut être mesurée que lorsque votre carte mère est alimentée en 12V ou 24V.

Plage de tension de la Vref : par défaut la tension Vref est réglée à 1V. Dans tous les cas la tension Vref sera comprise entre 0V et 2V.

Il ne faut pas trop forcer sur le potentiomètre pour éviter des dommages irréversibles. Lorsque la rotation dans le sens inverse des aiguilles d'une montre atteint le maximum et si vous continuez à forcer alors elle deviendra le minimum. Il en est de même dans le sens des aiguilles d'une montre ou si vous continuez à forcer le minimum deviendra le maximum.



Manuel BIGTREETECH TMC2208 V3.0

I3D Service

avec l'aimable autorisation de Shenzhen BIGTREE technology co., LTD

Instruction sur les modifications à faire dans le firmware

Exemple avec le firmware Marlin (marlin bugfix 2.0) et une carte mère BIGTREETECH SKR V1.3

Fichier configuration.h :

```
* Stepper Drivers
*
* These settings allow Marlin to tune stepper driver timing and enable advanced options for
* stepper drivers that support them. You may also override timing options in Configuration_adv.h.
*
* A4988 is assumed for unspecified drivers.
*
* Options: A4988, A5984, DRV8825, LV8729, L6470, TB6560, TB6600, TMC2100,
*          TMC2130, TMC2130_STANDALONE, TMC2160, TMC2160_STANDALONE,
*          TMC2208, TMC2208_STANDALONE, TMC2209, TMC2209_STANDALONE,
*          TMC26X, TMC26X_STANDALONE, TMC2660, TMC2660_STANDALONE,
*          TMC5130, TMC5130_STANDALONE, TMC5160, TMC5160_STANDALONE
* :['A4988', 'A5984', 'DRV8825', 'LV8729', 'L6470', 'TB6560', 'TB6600', 'TMC2100', 'TMC2130', 'TMC2130_STANDALONE', 'TMC2160', 'TMC2160_ST
*/
#define X_DRIVER_TYPE  TMC2208
#define Y_DRIVER_TYPE  TMC2208
#define Z_DRIVER_TYPE  TMC2208
//#define X2_DRIVER_TYPE  A4988
//#define Y2_DRIVER_TYPE  A4988
//#define Z2_DRIVER_TYPE  A4988
//#define Z3_DRIVER_TYPE  A4988
#define E0_DRIVER_TYPE  TMC2208
//#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
```

TMC2208 : pour utiliser les TMC2208 V3.0 en mode UART

TMC2208_STANDALONE : pour utiliser les TMC2208 V3.0 en mode STEP/DIR

Manuel BIGTREETECH TMC2208 V3.0

I3D Service

avec l'aimable autorisation de Shenzhen BIGTREE technology co., LTD

Fichier configuration_adv.h :

```
/**
 * TMC2130, TMC2160, TMC2208, TMC2209, TMC5130 and TMC5160 only
 * Use Trinamic's ultra quiet stepping mode.
 * When disabled, Marlin will use spreadCycle stepping mode.
 */
#define STEALTHCHOP_XY
#define STEALTHCHOP_Z
#define STEALTHCHOP_E

/**
 * Optimize spreadCycle chopper parameters by using predefined parameter sets
 * or with the help of an example included in the library.
 * Provided parameter sets are
 * CHOPPER_DEFAULT_12V
 * CHOPPER_DEFAULT_19V
 * CHOPPER_DEFAULT_24V
 * CHOPPER_DEFAULT_36V
 * CHOPPER_PRUSAMK3_24V // Imported parameters from the official Prusa firmware for MK3 (24V)
 * CHOPPER_MARLIN_119 // Old defaults from Marlin v1.1.9
 *
 * Define your own with
 * { <off_time[1..15]>, <hysteresis_end[-3..12]>, hysteresis_start[1..8] }
 */
#define CHOPPER_TIMING CHOPPER_DEFAULT_24V
```

Fichier Pins_BTT_SKR_V1.3.h

```
// Software serial
//
#define X_SERIAL_TX_PIN P4_29
#define X_SERIAL_RX_PIN P1_17

#define Y_SERIAL_TX_PIN P1_16
#define Y_SERIAL_RX_PIN P1_15

#define Z_SERIAL_TX_PIN P1_14
#define Z_SERIAL_RX_PIN P1_10

#define E0_SERIAL_TX_PIN P1_09
#define E0_SERIAL_RX_PIN P1_08

#define E1_SERIAL_TX_PIN P1_04
#define E1_SERIAL_RX_PIN P1_01

#define Z2_SERIAL_TX_PIN P1_04
#define Z2_SERIAL_RX_PIN P1_01
```

Après les modifications dans le firmware, vous devez le compiler puis le transférer sur votre carte mère. Vérifier si les drivers sont bien installés et détectés.