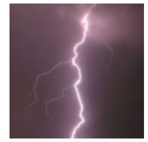
Stepper Driver Configurations for SKR PRO V1.1 Board


By
@GadgetAngel

Based on Work by  @rflulling

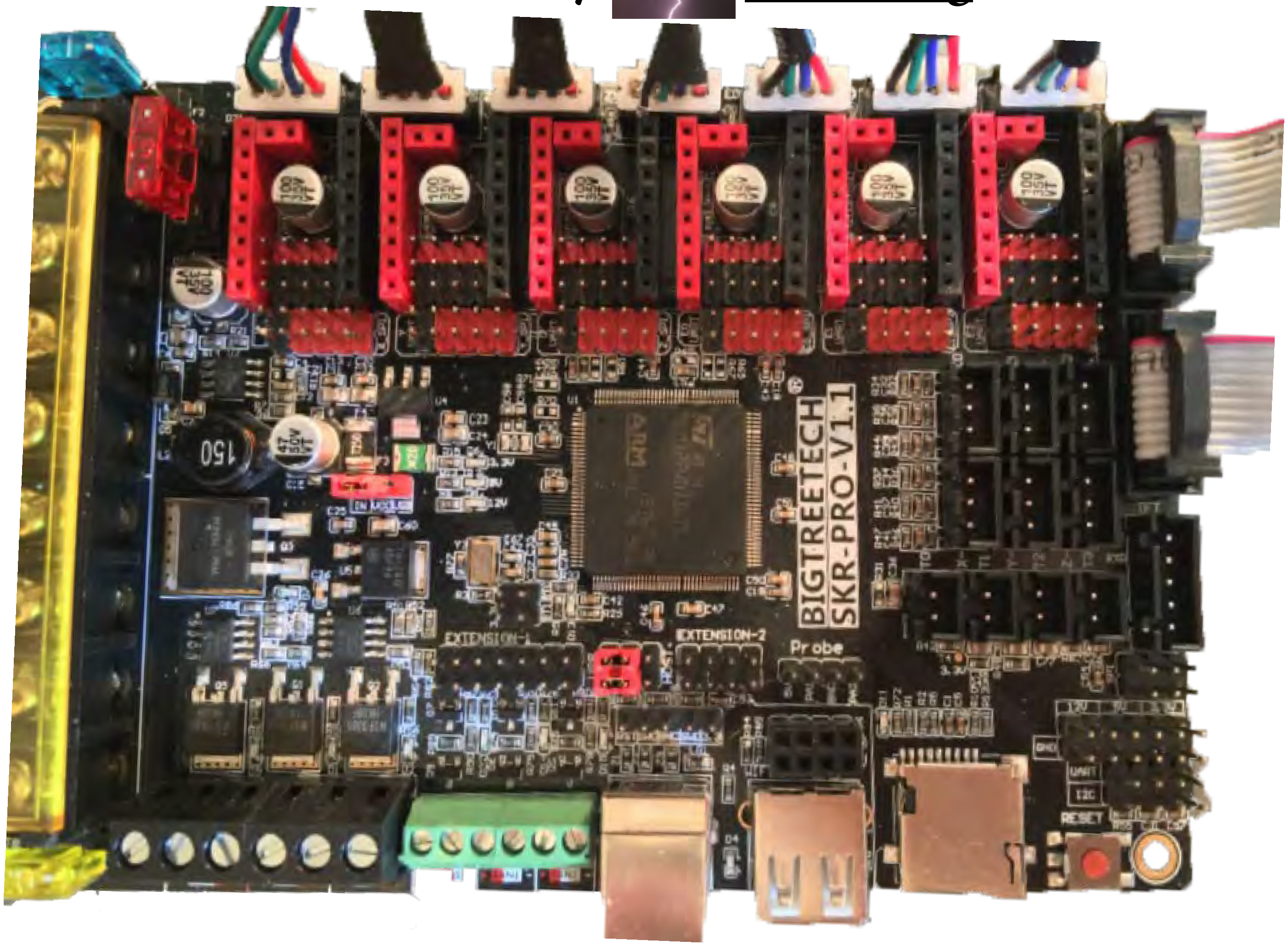


Table of Contents

POLOLU A4988 1

The (latest release of) Marlin Setup for POLOLU A4988 Drivers 5

BIQU A4988 9

The (latest release of) Marlin Setup for BIQU A4988 Drivers 13

DRV8825 17

The (latest release of) Marlin Setup for DRV8825 Drivers 22

BIQU LV8729 27

The (latest release of) Marlin Setup for BIQU LV8729 Drivers 32

FYSETC LV8729 38

The (latest release of) Marlin Setup for FYSETC LV8729 Drivers 43

LERDGE LV8729 49

The (latest release of) Marlin Setup for LERDGE LV8729 Drivers 54

MKS LV8729 60

The (latest release of) Marlin Setup for MKS LV8729 Drivers 65

FYSETC S6128 V1.1 71

The (latest release of) Marlin Setup for FYSETC S6128 V1.1 Drivers 76

FYSETC ST820 82

The (latest release of) Marlin Setup for FYSETC ST820 Drivers 87

BIQU ST820 94

The (latest release of) Marlin Setup for BIQU ST820 Drivers 99

POLOLU ST820 (STSPIN820) 106

The (latest release of) Marlin Setup for POLOLU ST820 (STSPIN820) Drivers 111

POLOLU MP6500 118

The (latest release of) Marlin Setup for POLOLU MP6500 Drivers 122

Table of Contents

POLOLU TB67S249FTG 128

The (latest release of) Marlin Setup for POLOLU TB67S249FTG Drivers 133

BIQU S109 139

The (latest release of) Marlin Setup for BIQU S109 Drivers 144

FYSETC S109 150

The (latest release of) Marlin Setup for FYSETC S109 Drivers 155

BIQU TMC2100 Stand-alone Mode 161

The (latest release of) Marlin Setup for BIQU TMC2100 Drivers in Stand-alone Mode 166

MKS TMC2100 Stand-alone Mode 172

The (latest release of) Marlin Setup for MKS TMC2100 Drivers in Stand-alone Mode 177

BIQU TMC2130 Stand-alone Mode 183

The (latest release of) Marlin Setup for BIQU TMC2130 Drivers in Stand-alone Mode 188

BIQU TMC2130 SPI Mode 194

The (latest release of) Marlin Setup for BIQU TMC2130 Drivers in SPI Mode 197

BIQU TMC2208 V3.0 Stand-alone Mode 211

The (latest release of) Marlin Setup for BIQU TMC2208 V3.0 Drivers in Stand-alone Mode 215

BIQU TMC2208 V3.0 One Time Programming (OTP) Mode 221

The (latest release of) Marlin Setup for BIQU TMC2208 V3.0 Drivers in One Time Programming (OTP) Mode ... 225

BIQU TMC2208 V3.0 UART Mode 231

The (latest release of) Marlin Setup for BIQU TMC2208 V3.0 Drivers in UART Mode 234

FYSETC TMC2208 V1.2 Stand-alone Mode 245

The (latest release of) Marlin Setup for FYSETC TMC2208 V1.2 Drivers in Stand-alone Mode 249

FYSETC TMC2208 V1.2 One Time Programming (OTP) Mode 255

The (latest release of) Marlin Setup for FYSETC TMC2208 V1.2 Drivers in One Time Programming (OTP) Mode ... 259

Table of Contents

FYSETC TMC2208 V1.2 UART Mode 265

The (latest release of) Marlin Setup for FYSETC TMC2208 V1.2 Drivers in UART Mode 268

BIQU TMC2225 V1.0 Stand-alone Mode 279

The (latest release of) Marlin Setup for BIQU TMC2225 V1.0 Drivers in Stand-alone Mode 283

BIQU TMC2225 V1.0 One Time Programming (OTP) Mode 289

The (latest release of) Marlin Setup for BIQU TMC2225 V1.0 Drivers in One Time Programming (OTP) Mode ... 293

BIQU TMC2225 V1.0 UART Mode 299

The (latest release of) Marlin Setup for BIQU TMC2225 V1.0 Drivers in UART Mode 302

BIQU TMC2209 V1.2 Stand-alone Mode for StealthChop 313

The (latest release of) Marlin Setup for BIQU TMC2209 V1.2 Drivers in Stand-alone Mode for StealthChop 317

BIQU TMC2209 V1.2 Stand-alone Mode for SpreadCycle 323

The (latest release of) Marlin Setup for BIQU TMC2209 V1.2 Drivers in Stand-alone Mode for SpreadCycle 327

BIQU TMC2209 V1.2 UART Mode 333

The (latest release of) Marlin Setup for BIQU TMC2209 V1.2 Drivers in UART Mode 336

BIQU TMC5160 V1.2 SPI Mode 350

The (latest release of) Marlin Setup for BIQU TMC5160 V1.2 Drivers in SPI Mode 353

BIQU TMC5161 V1.0 SPI Mode 367

The (latest release of) Marlin Setup for BIQU TMC5161 V1.0 Drivers in SPI Mode 370

APPENDIX A 384

How to adjust the Vref on a Stepper Motor Driver board using the Potentiometer 384

APPENDIX B 386

For the TMC drivers what's the difference between stand-alone mode and ("UART" or "SPI ") modes? 386

How to Calculate Vref for Non-TMC Stepper Motor Divers 386

How to Calculate Vref for TMC Stepper Motor Drivers 387

Table of Contents

Driving Current Calculation Formulas for TMC Stepper Motor Drivers 388

#1 TMC2100 with R_s=0.110Ω (110mΩ) 388

#2 TMC2130 with R_s=0.110Ω (110mΩ) 388

#3 TMC2208 with R_s=0.110Ω (110mΩ) for Stand-alone Mode 3 88

#4 TMC2208 with R_s=0.110Ω (110mΩ) for UART Mode 389

#5 TMC2209 with R_s=0.110Ω (110mΩ) for Stand-alone Mode 389

#6 TMC2209 with R_s=0.110Ω (110mΩ) for UART Mode 390

#7 TMC2225 with R_s=0.150Ω (150mΩ) for UART Mode 390

#8 TMC5160 with R_s=0.075Ω (75mΩ) for SPI Mode 390

#9 TMC5161 with R_s=0.062Ω (62mΩ) for SPI Mode 391

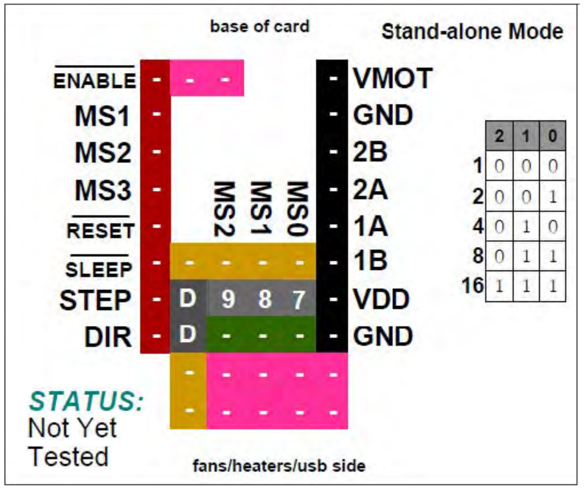
#10 TMC2225 with R_s=0.150Ω (150mΩ) for Stand-alone Mode 391

APPENDIX C 392

The (Latest Release of) Marlin Setup That Is Common To ALL Stepper Motor Drivers 392

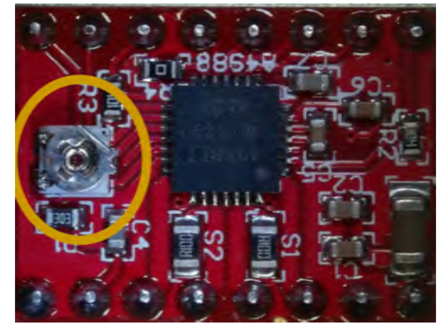
SKR PRO V1.1 Color PINS Diagram 405

SKR PRO V1.1 Color Wiring Diagram 406

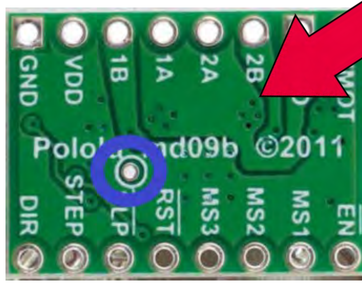


POLOLU A4988

NOTE: Use the potentiometer (POT) on the top of the board (or use the board's "V_{ref} Test point") to adjust your V_{ref}.



Note: "V_{ref} Test point" location is on the bottom of the driver board.



Note: **Use 90% of the calculated V_{ref} when tuning the stepper driver board**

See **Appendix A** for instructions on how to set the V_{ref} on a driver board.

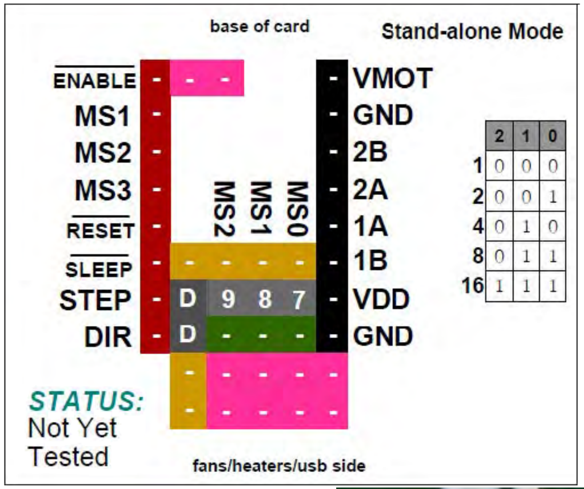
Note: See this video about current sense resistors (R_s) and their possible locations: **https://youtu.be/8wk1elugv5A**

Driver Chip	MS2	MS1	MS0	Microstep Resolution	Excitation Mode
Pololu A4988 Maximum 16 Subdivision 35V DC 2A (peak)	Low	Low	Low	Full step	2 Phase
	Low	Low	High	Half step	1-2 Phase
	Low	High	Low	Quarter step	W1-2 Phase
	Low	High	High	Eighth step	2W1-2 Phase
	High	High	High	Sixteenth step	4W1-2 Phase

Driving Current Calculation Formula		
R_S(Typical Sense Resistor)$= 0.1\Omega$	$I_{MAX}=V_{ref}/(8*R_S)$	$V_{ref}=8*I_{MAX}*R_S$

NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

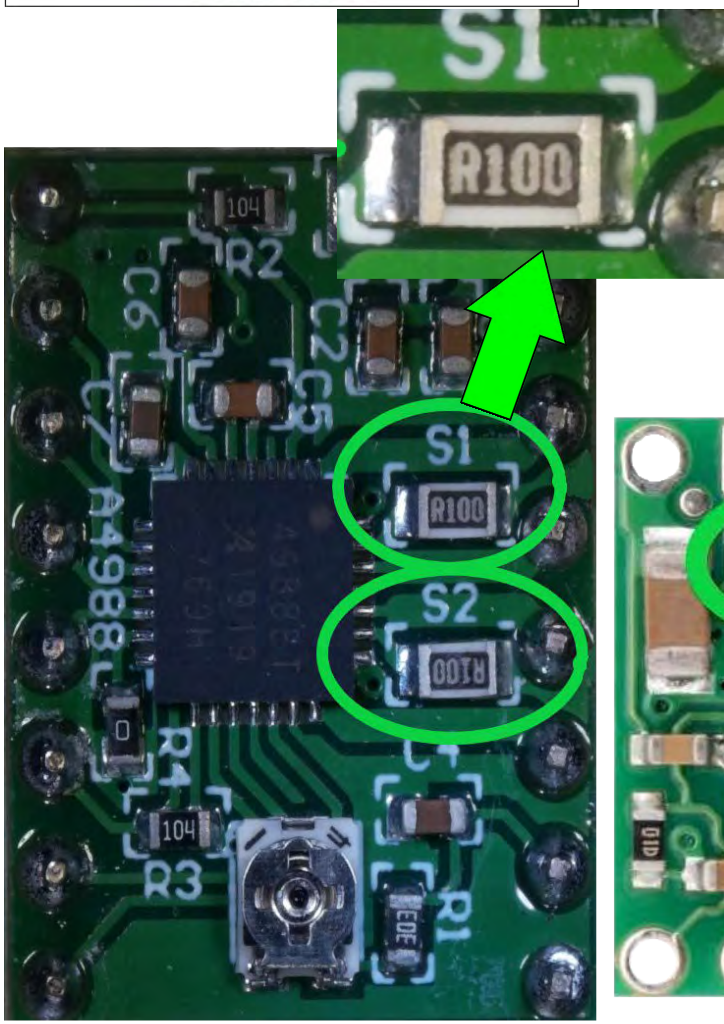
POLOLU A4988

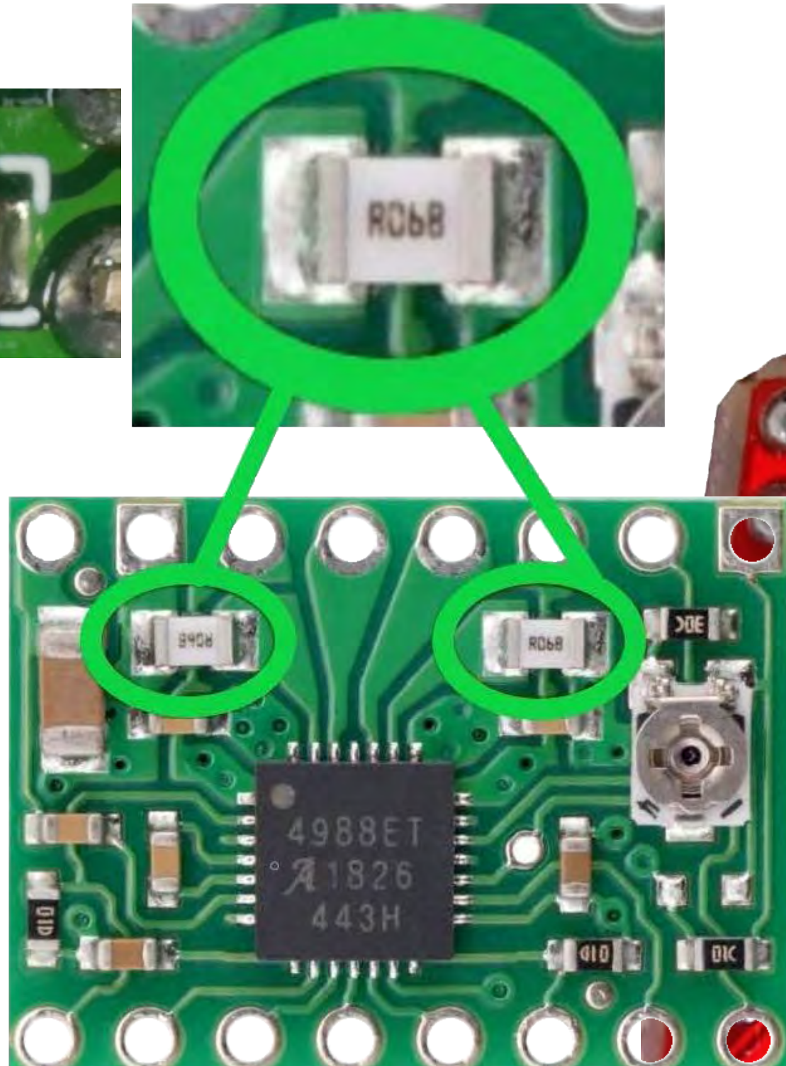


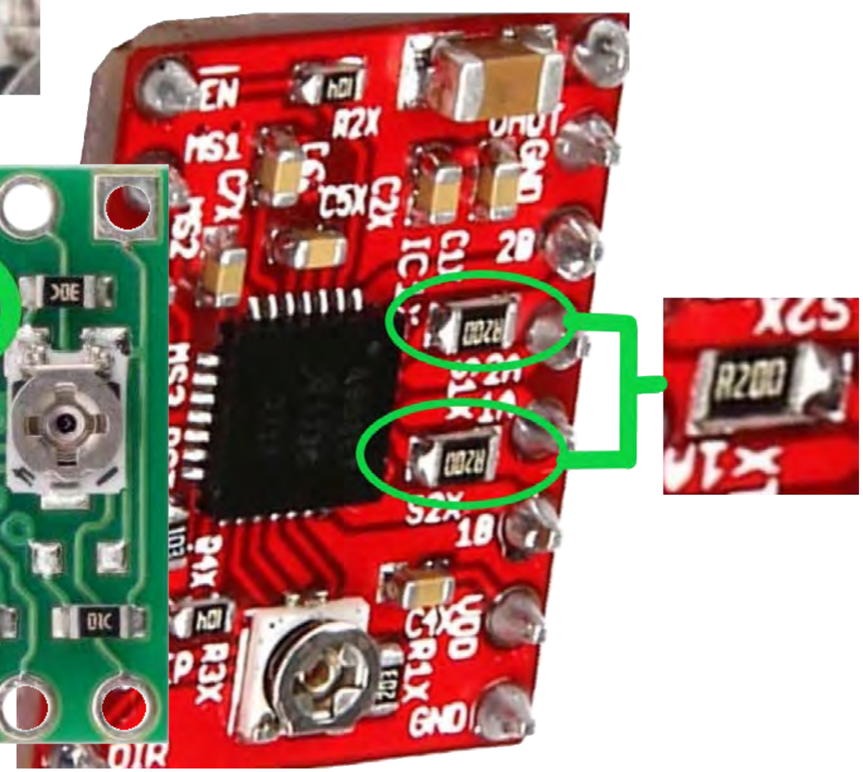
	2	1	0
1	0	0	0
2	0	0	1
4	0	1	0
8	0	1	1
16	1	1	1

Note: Not all driver boards for the A4988 use the same current sense resistors (R_S); check your driver board for the value of the (R_S) resistors by examining the board, as shown in **GREEN** below. The **GREEN PCB** shows a 0.1 Ohm (R100) sense resistor value. The **RED PCB** shows a 0.2 Ohms (R200) sense resistor value. Sense resistors (R_S) can appear in the following values, (these are just a few values): R050=0.05 Ohms; R068=0.068 Ohms; R100=0.1 Ohms; R200 = 0.2 Ohms.

R_S = R050 is 0.05 Ohms
R_S = R068 is 0.068 Ohms
R_S = R100 is 0.1 Ohms
R_S = R150 is 0.15 Ohms
R_S = R200 is 0.2 Ohms
R_S = R220 is 0.22 Ohms





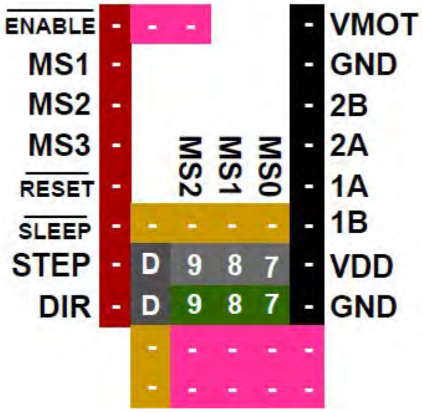


NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

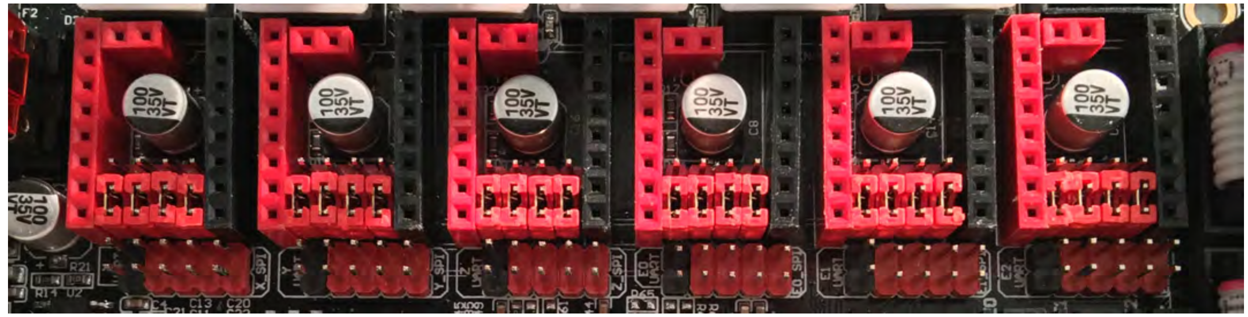
POLOLU A4988

Stand-alone Mode

STEP

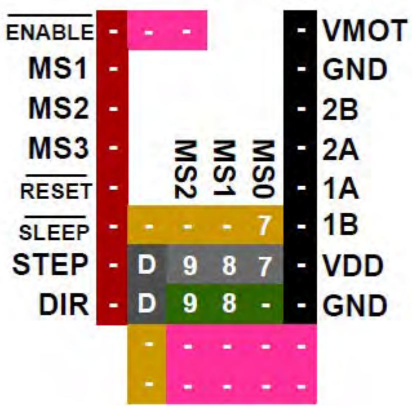


Note: The "D" jumper MUST be SET!

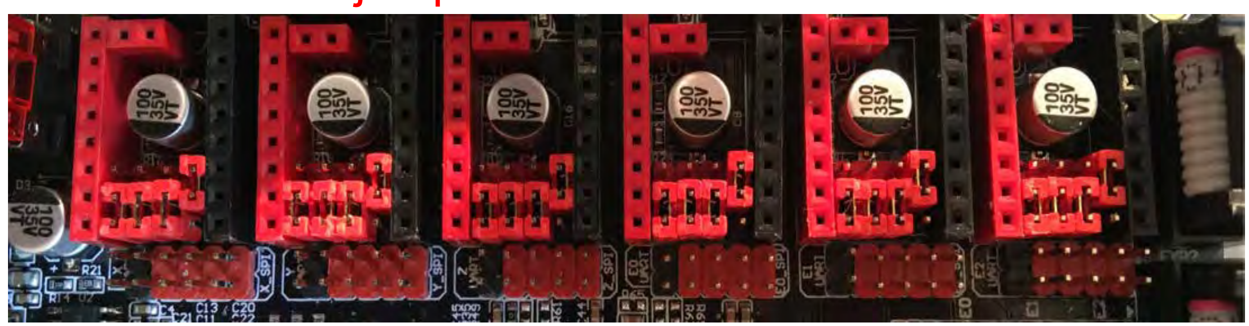


---

1 / 2



Note: The "D" jumper MUST be SET!

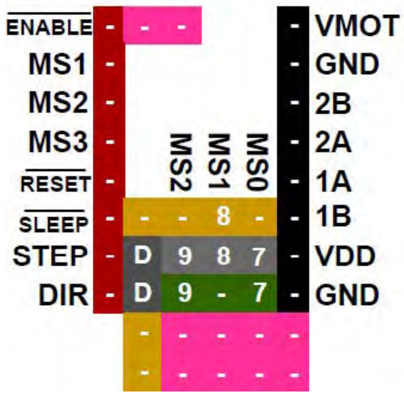


NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

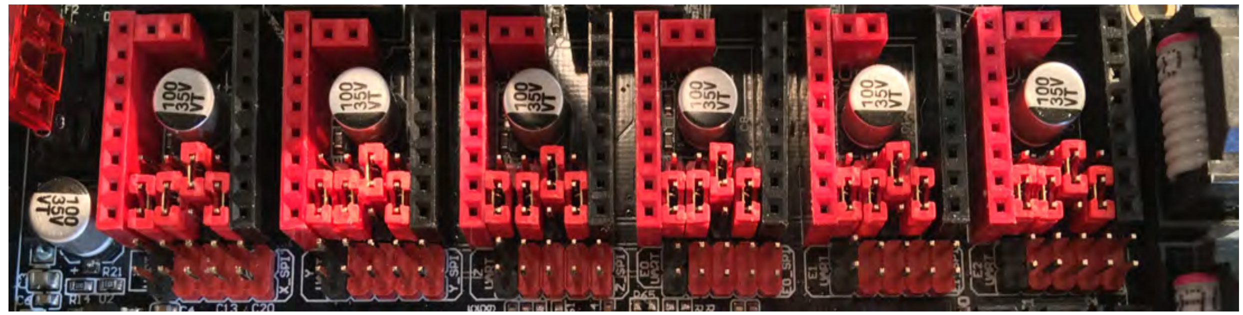
POLOLU A4988

Stand-alone Mode

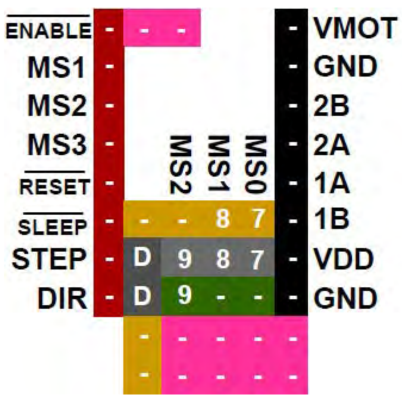
1 / 4



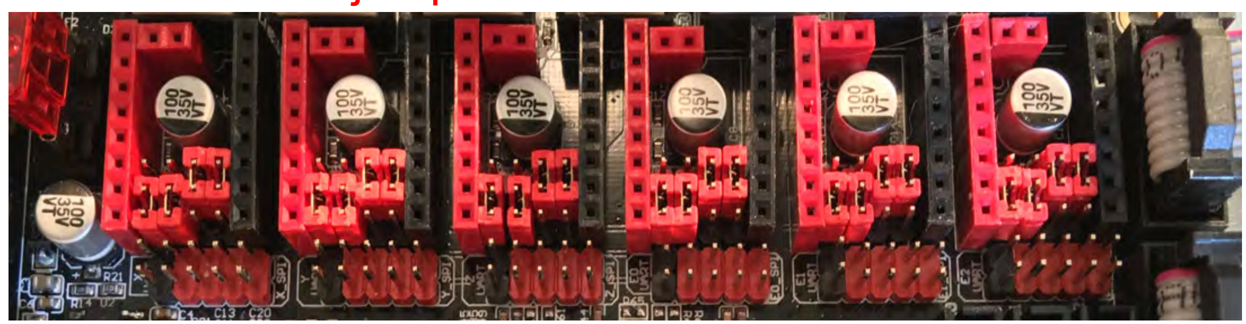
Note: The "D" jumper MUST be SET!



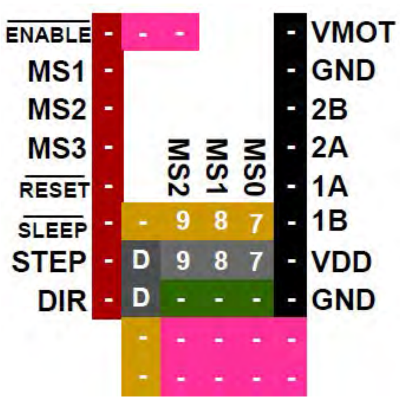
1 / 8



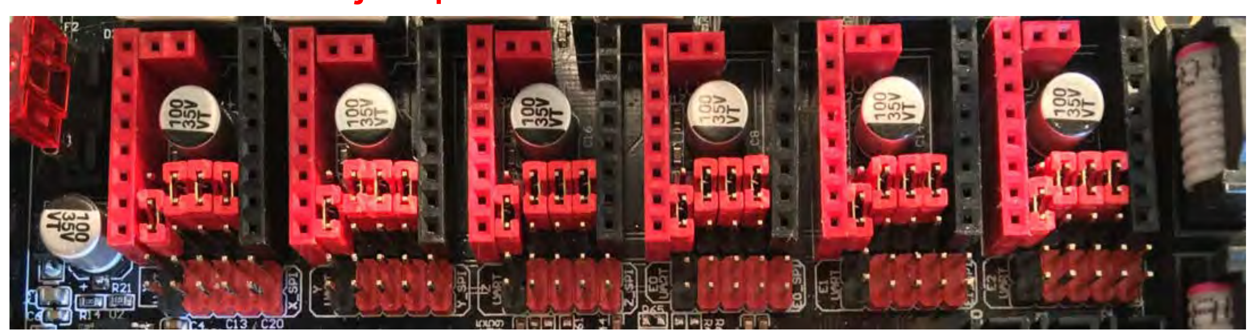
Note: The "D" jumper MUST be SET!



1 / 16



Note: The "D" jumper MUST be SET!

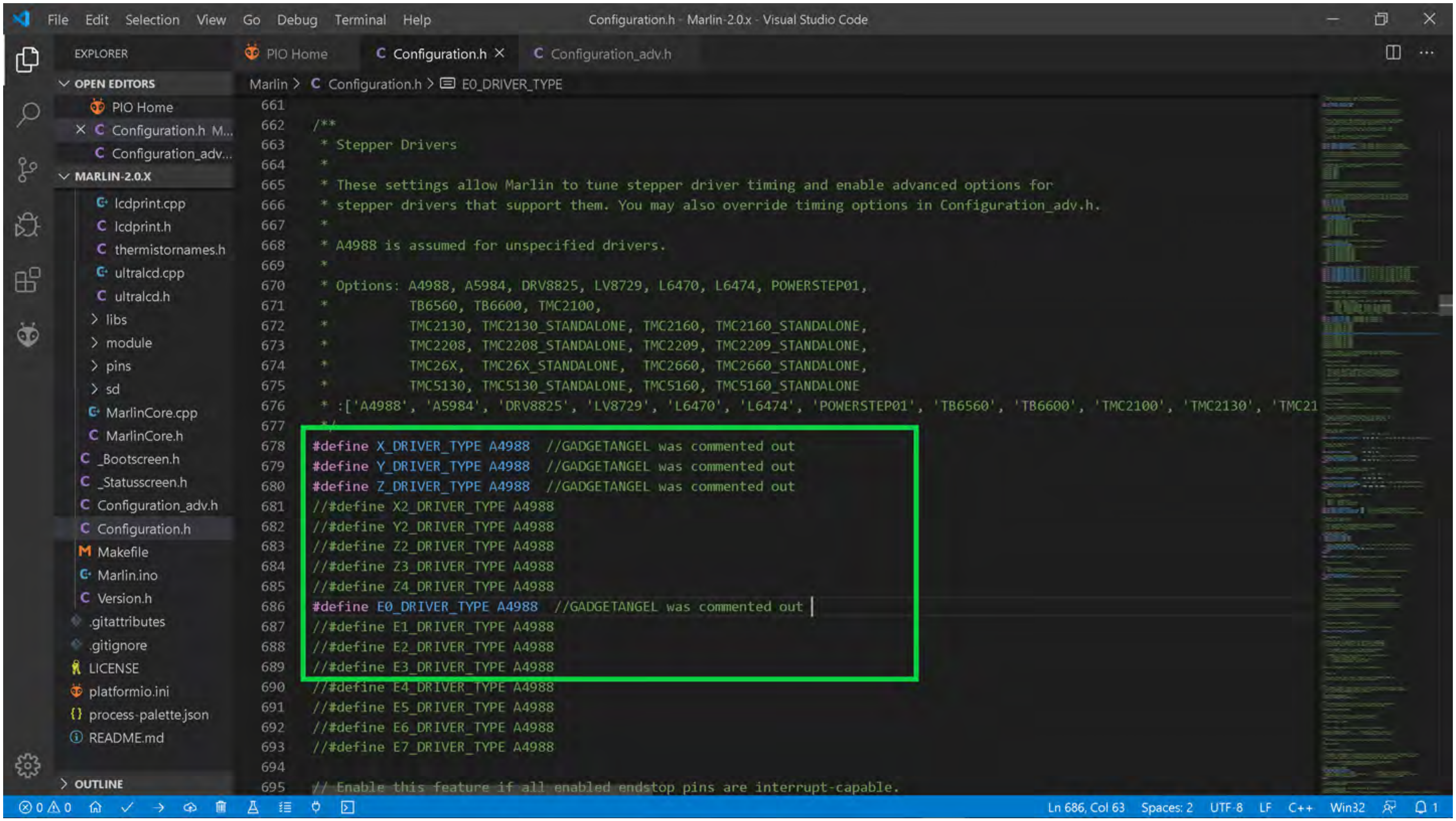


NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

The (latest release of) Marlin Setup for POLOLU A4988 Drivers

NOTE: **Go to Appendix C**, and then come back here for the changes to Marlin for POLOLU A4988 stepper motor drivers.

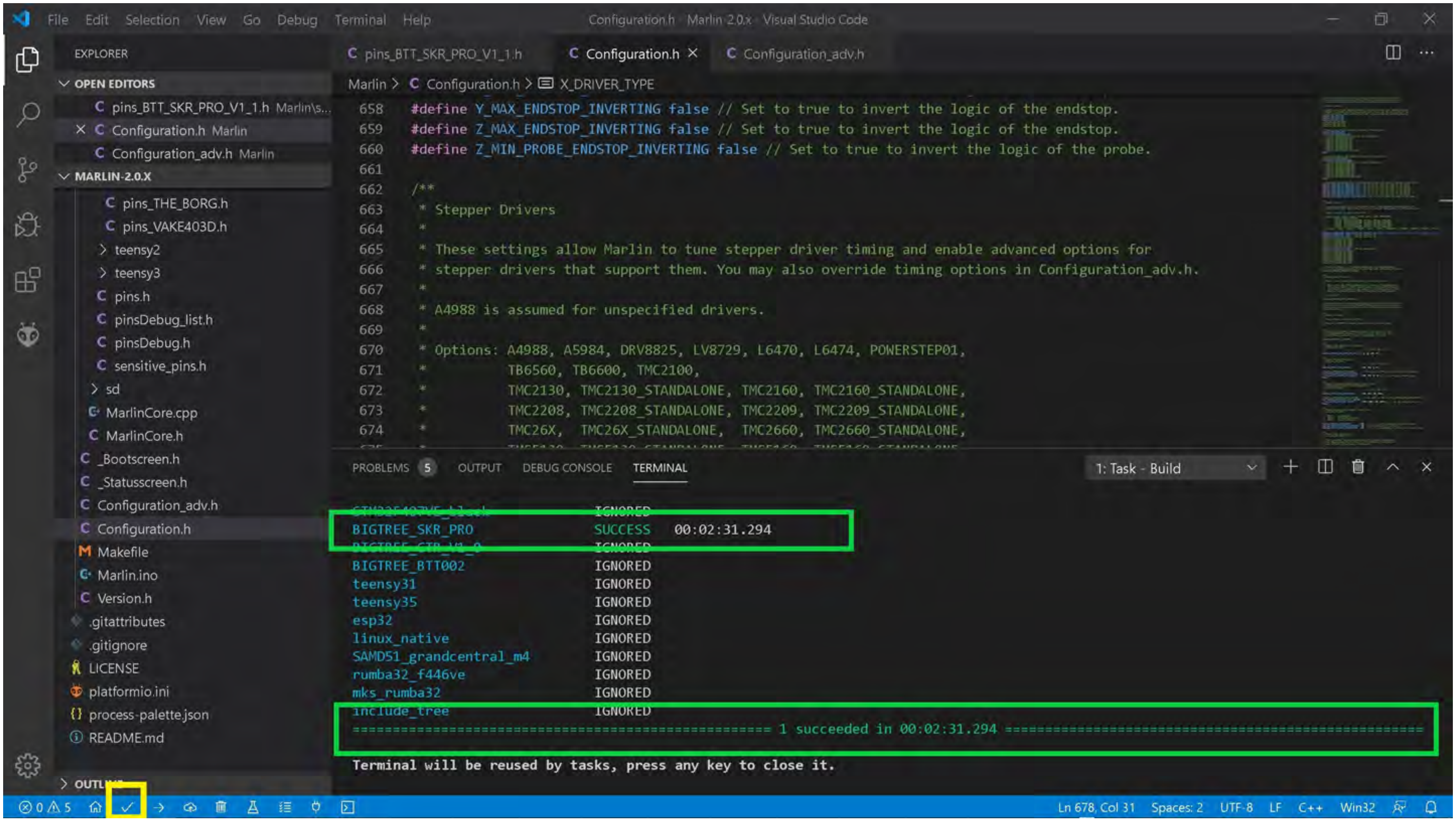
- Change the stepper motor drivers so that Marlin knows you are using POLOLU A4988 drivers. Change one line for each axis and one for each extruder you will be using. See the picture below for an example of how to use POLOLU A4988 drivers. When two "//" appear at the beginning of a line that means that line is commented out. To un-comment a line just remove the leading two forward slashes ("//").



- Go to the next page.

The (latest release of) Marlin Setup for POLOLU A4988 Drivers

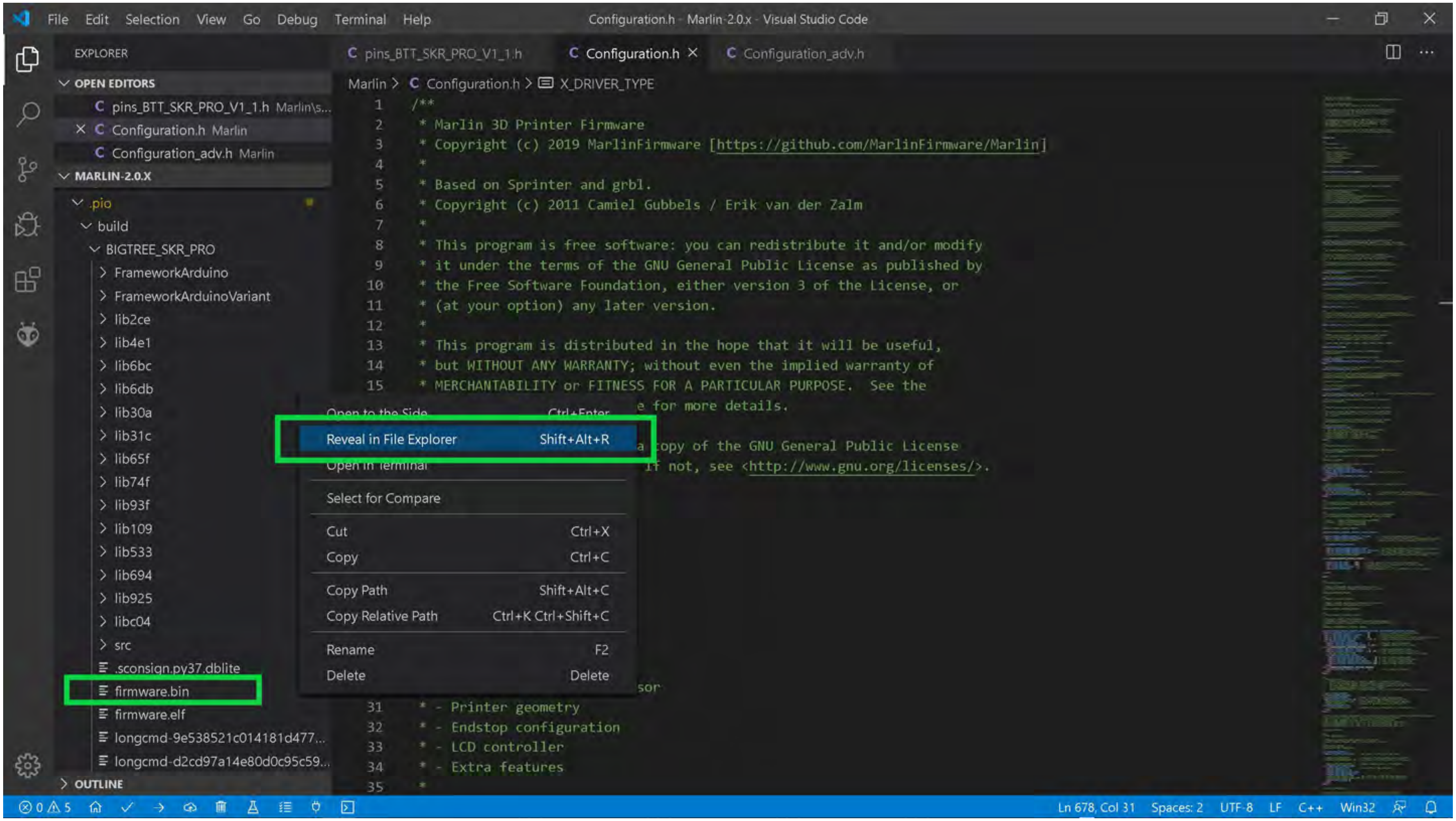
- The end of Marlin setup for POLOLU A4988 drivers. Now, compile your code by clicking on the check mark, as seen in the **YELLOW** box below. Once the firmware has compiled successfully you will see the following messages, as shown in the **GREEN** boxes below.



- For the SKR PRO V1.1 board you must save the "firmware.bin" file to the micro SD card then place the micro SD card into the micro SD card reader of the SKR PRO V1.1 board and turn on the power to your printer for the NEW firmware to be automatically applied to the SKR PRO MCU. To locate the "firmware.bin" file, and learn how to transfer it to your micro SD card go to the next page.

The (latest release of) Marlin Setup for POLOLU A4988 Drivers

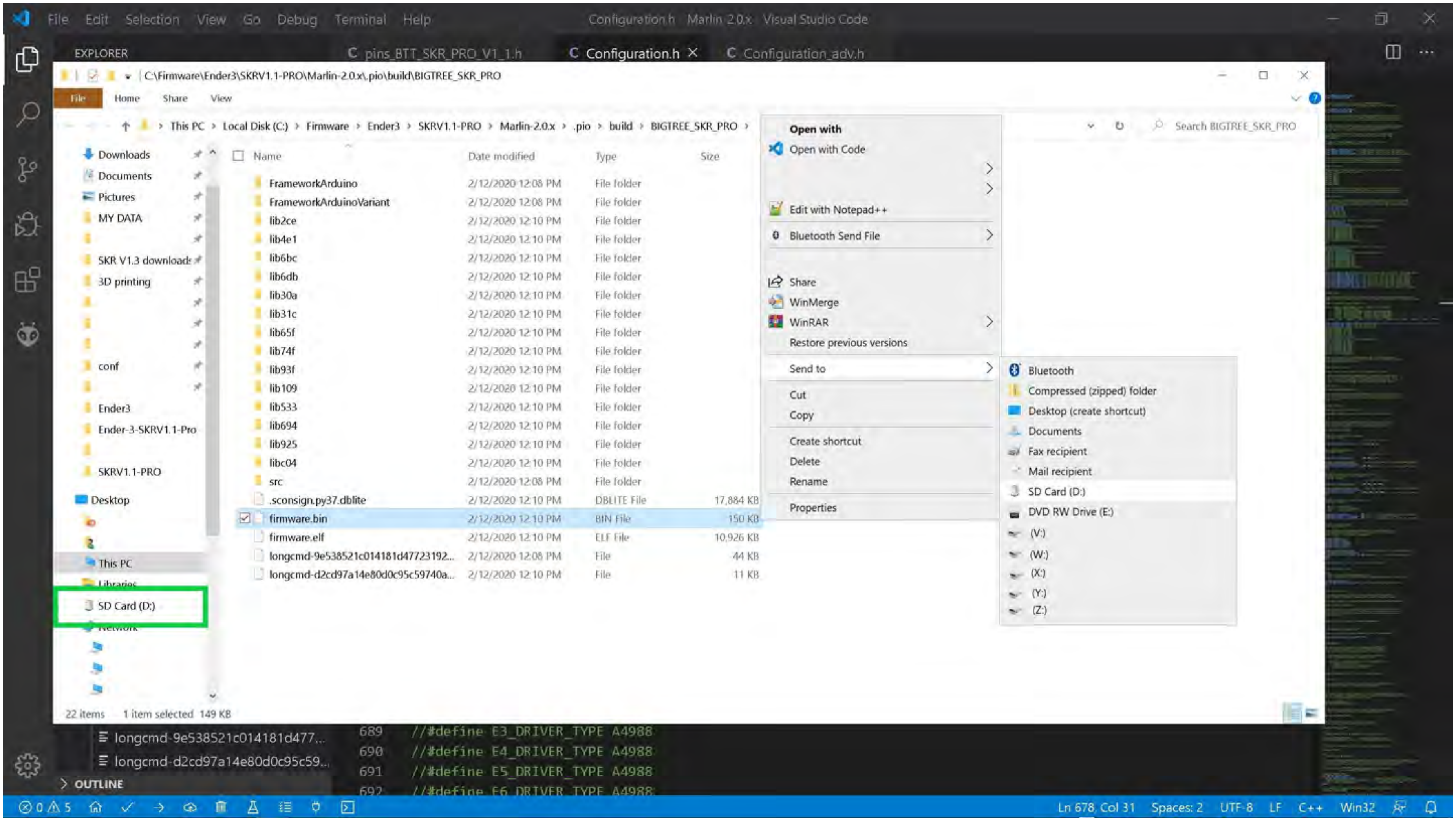
- The "firmware.bin" file can be found by using the left side of the VS code window and going to folder Marlin-2.0.X/.pio/build/ BIGTREE_SKR_PRO/ and right clicking on the "firmware.bin" file name to bring up the content menu. Select "Reveal in File Explorer" to have your Window's machine open a file explorer window.



- Go to the next page.

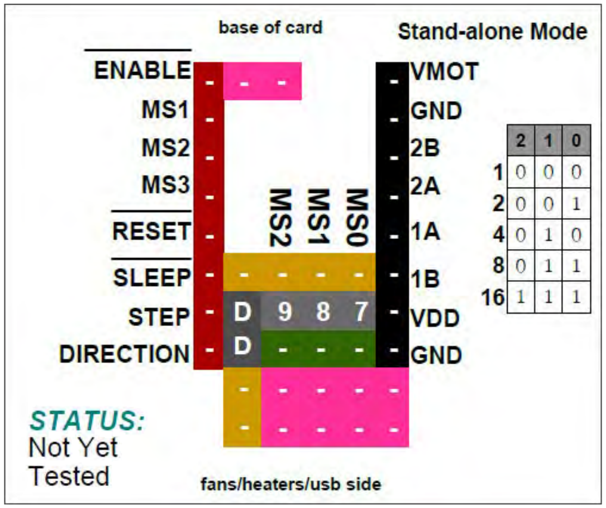
The (latest release of) Marlin Setup for POLOLU A4988 Drivers

- Ensure your micro SD card, which came with your SKR PRO V1.1 board, is in a micro-to-SD-card adapter and plugged into your Window's SD card reader.
    - From the file explorer window right click on the "firmware.bin" file name and select "Send to" from the content menu. For my computer "D:" is my SD card reader. For your computer it may be a different drive letter. The "firmware.bin" file will be written to the SD card and renamed to "FIRMWARE.CUR".

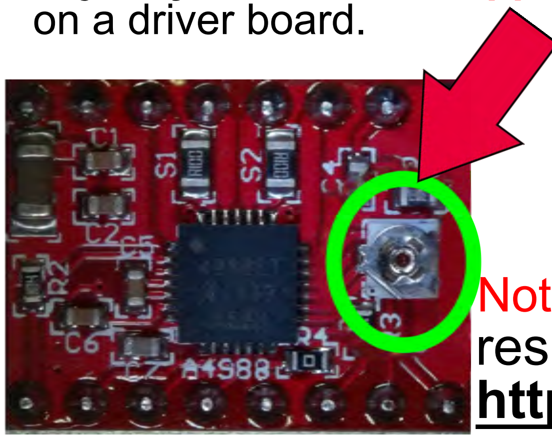


- Right click on the SD card reader, as seen in the **GREEN** box above, and select "Eject" from the context menu. Take the micro SD card out of the adapter and place it into the micro SD card reader of the SKR PRO V1.1 board. Turn on the power to the printer (i.e. the SKR PRO V1.1 board) and the "FIRMWARE.CUR" file will be uploaded to the board. After the upload the firmware file will be renamed to "firmware.bin" and stored on the micro SD card.

BIQU A4988



NOTE: **Use the potentiometer (POT) on the top of the board to adjust your V_{ref}.** See **Appendix A** for instructions on how to set the V_{ref} on a driver board.



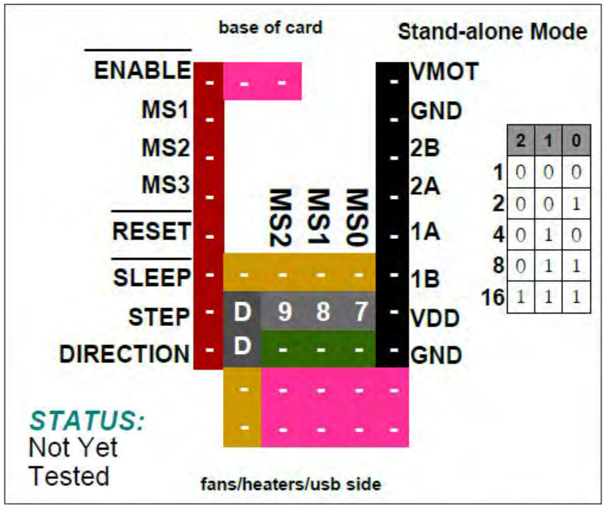
Note: **Use 90% of the calculated V_{ref} when tuning the stepper driver board**

Note: See this video about current sense resistors (R_s) and their possible locations: **https://youtu.be/8wk1elugv5A**

Driver Chip	MS2	MS1	MS0	Microstep Resolution	Excitation Mode
BIQU® A4988 Maximum 16 Subdivision 35V DC 2A (peak)	Low	Low	Low	Full step	2 Phase
	Low	Low	High	Half step	1-2 Phase
	Low	High	Low	Quarter step	W1-2 Phase
	Low	High	High	Eighth step	2W1-2 Phase
	High	High	High	Sixteenth step	4W1-2 Phase

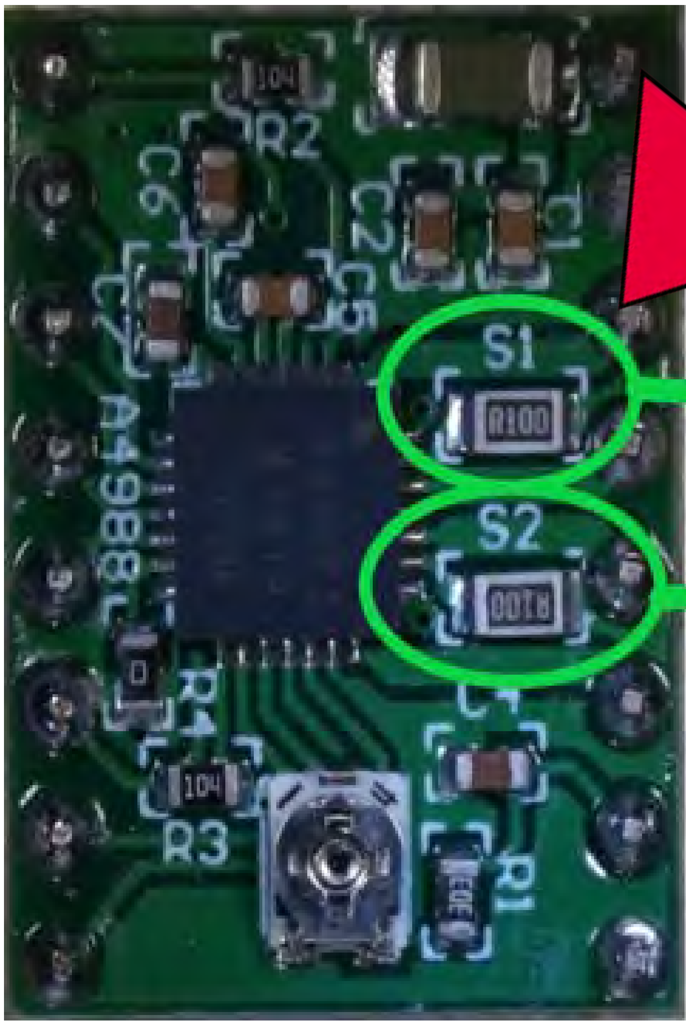
Driving Current Calculation Formula R_S(Typical Sense Resistor)$=0.1\Omega$	$I_{MAX}=V_{ref}/(8*R_S)$	$V_{ref}=8*I_{MAX}*R_S$

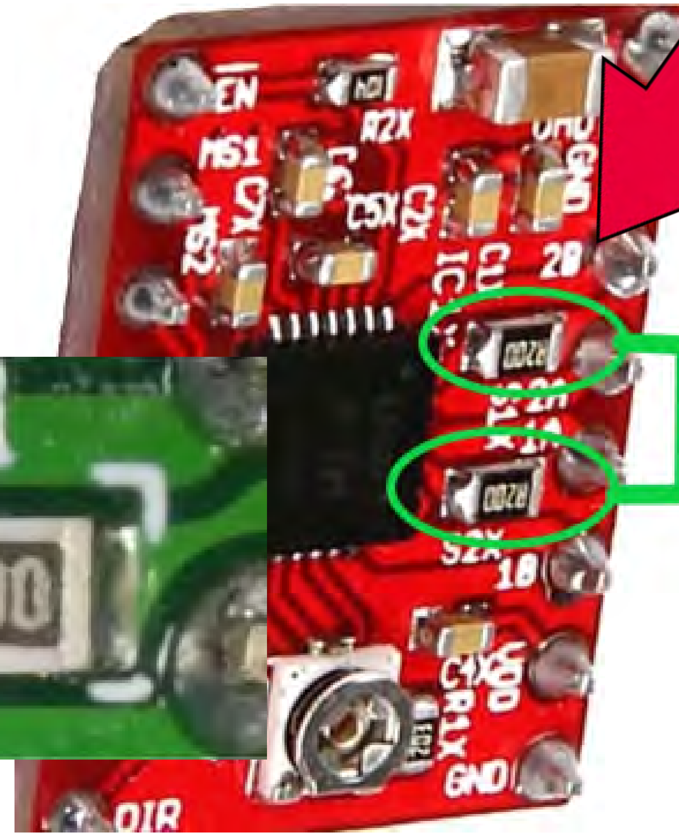
BIQU A4988

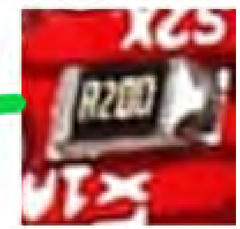


	2	1	0
1	0	0	0
2	0	0	1
4	0	1	0
8	0	1	1
16	1	1	1

Note: Not all driver boards for the A4988 use the same current sense resistors (R_S); check your driver board for the value of the (R_S) resistors by examining the board, as shown in **GREEN** below. The **GREEN PCB** shows a 0.1 Ohm (R100) sense resistor value. The **RED PCB** shows a 0.2 Ohms (R200) sense resistor value. Sense resistors (R_S) can appear in the following values, (these are just a few values): R050=0.05 Ohms; R068=0.068 Ohms; R100=0.1 Ohms; R200 = 0.2 Ohms.







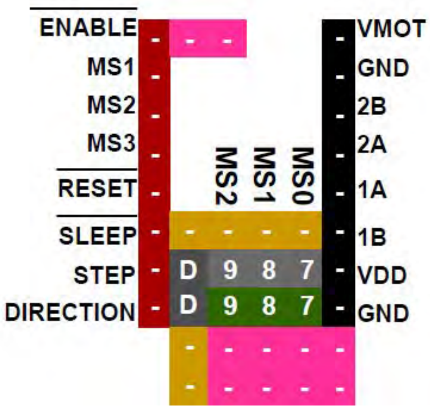
R_s = R050 is 0.05 Ohms; R_s = R068 is 0.068 Ohms
R_s = R100 is 0.1 Ohms; R_s = R150 is 0.15 Ohms
R_s = R200 is 0.2 Ohms; R_s = R220 is 0.22 Ohms

NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

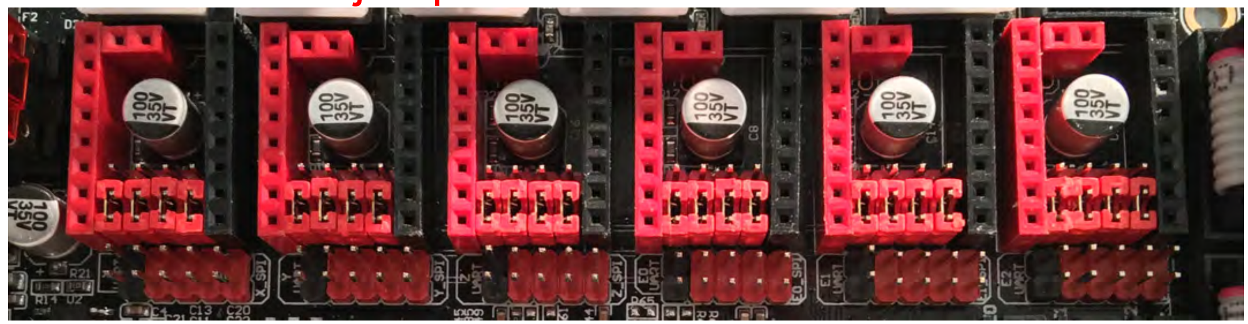
BIQU A4988

Stand-alone Mode

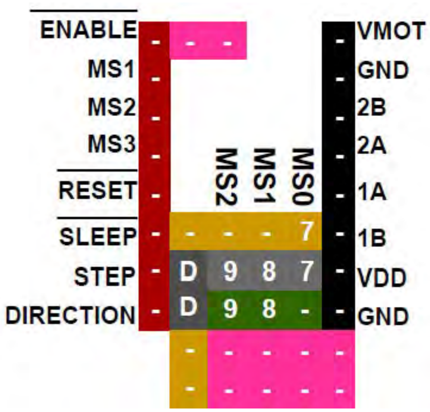
STEP



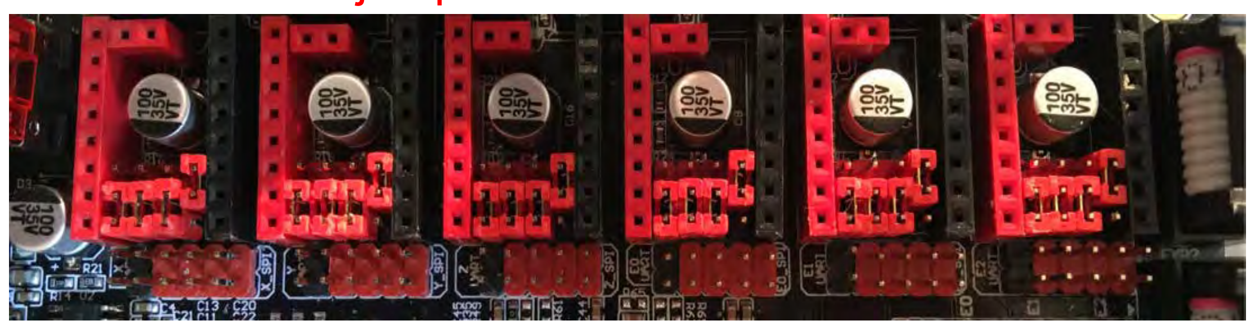
Note: The "D" jumper MUST be SET!



1 / 2



Note: The "D" jumper MUST be SET!

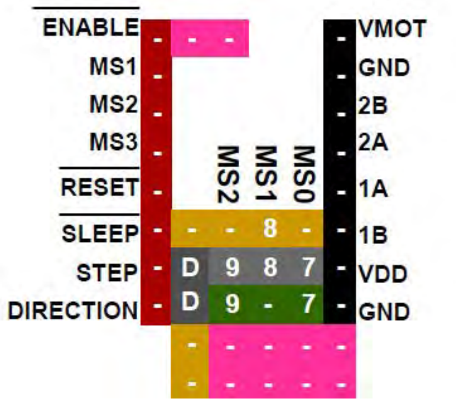


NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

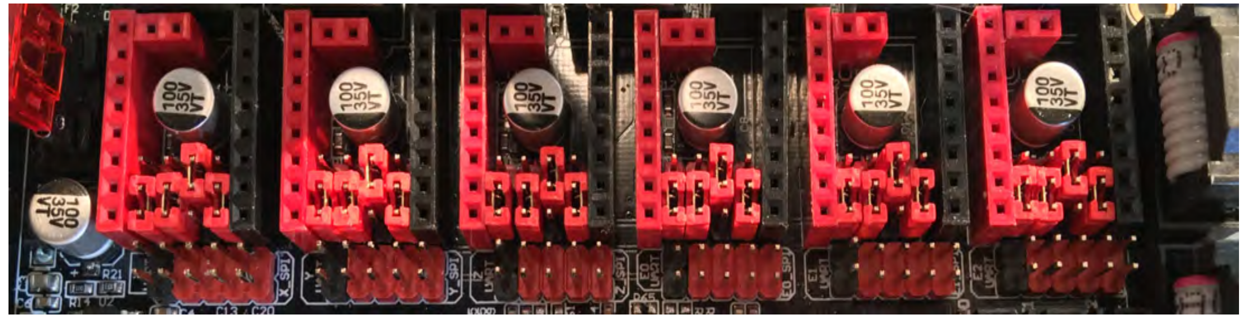
BIQU A4988

Stand-alone Mode

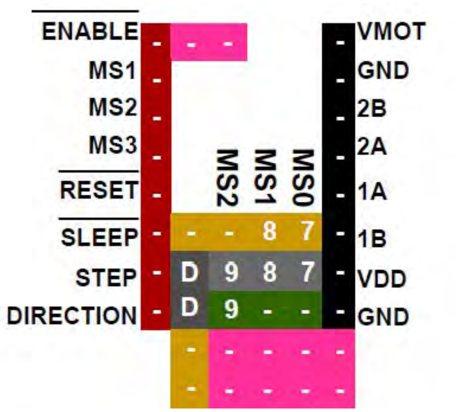
1 / 4



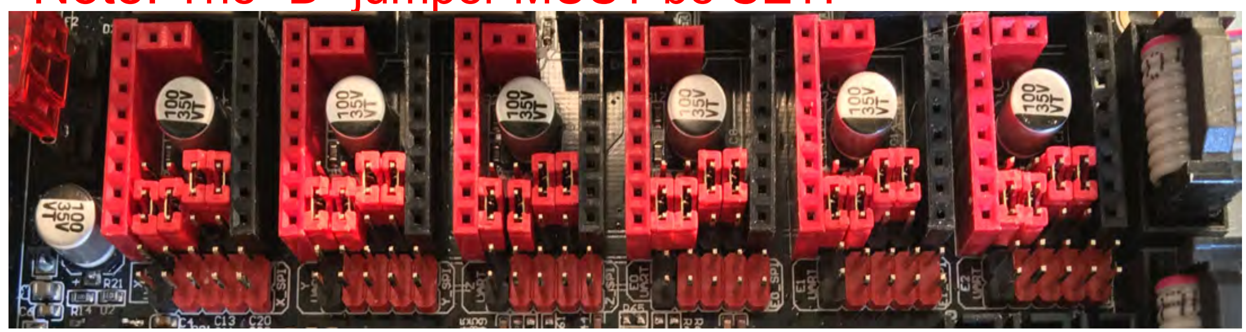
Note: The "D" jumper MUST be SET!



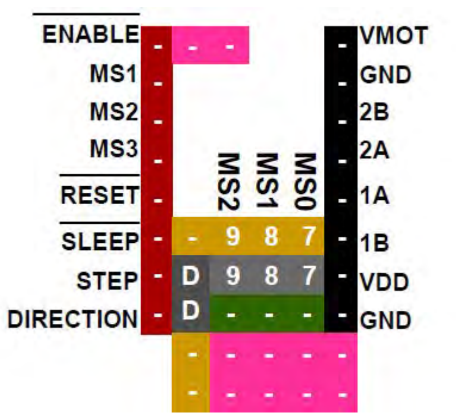
1 / 8



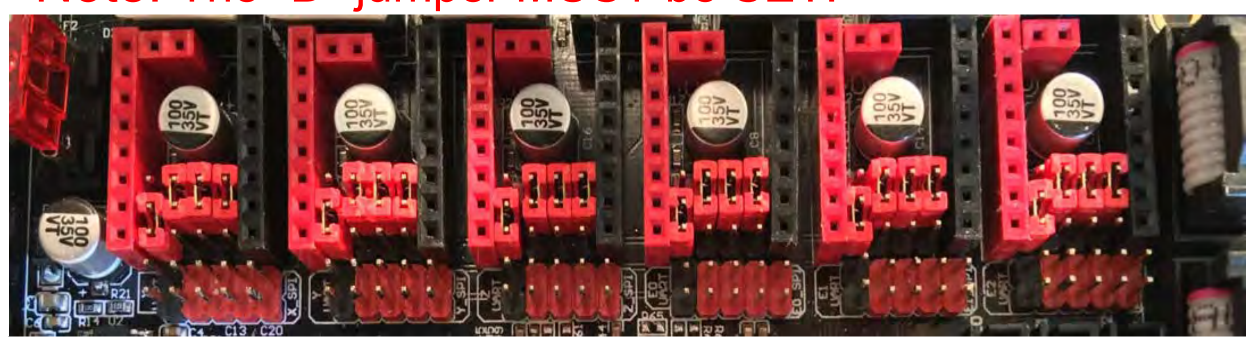
Note: The "D" jumper MUST be SET!



1 / 16



Note: The "D" jumper MUST be SET!

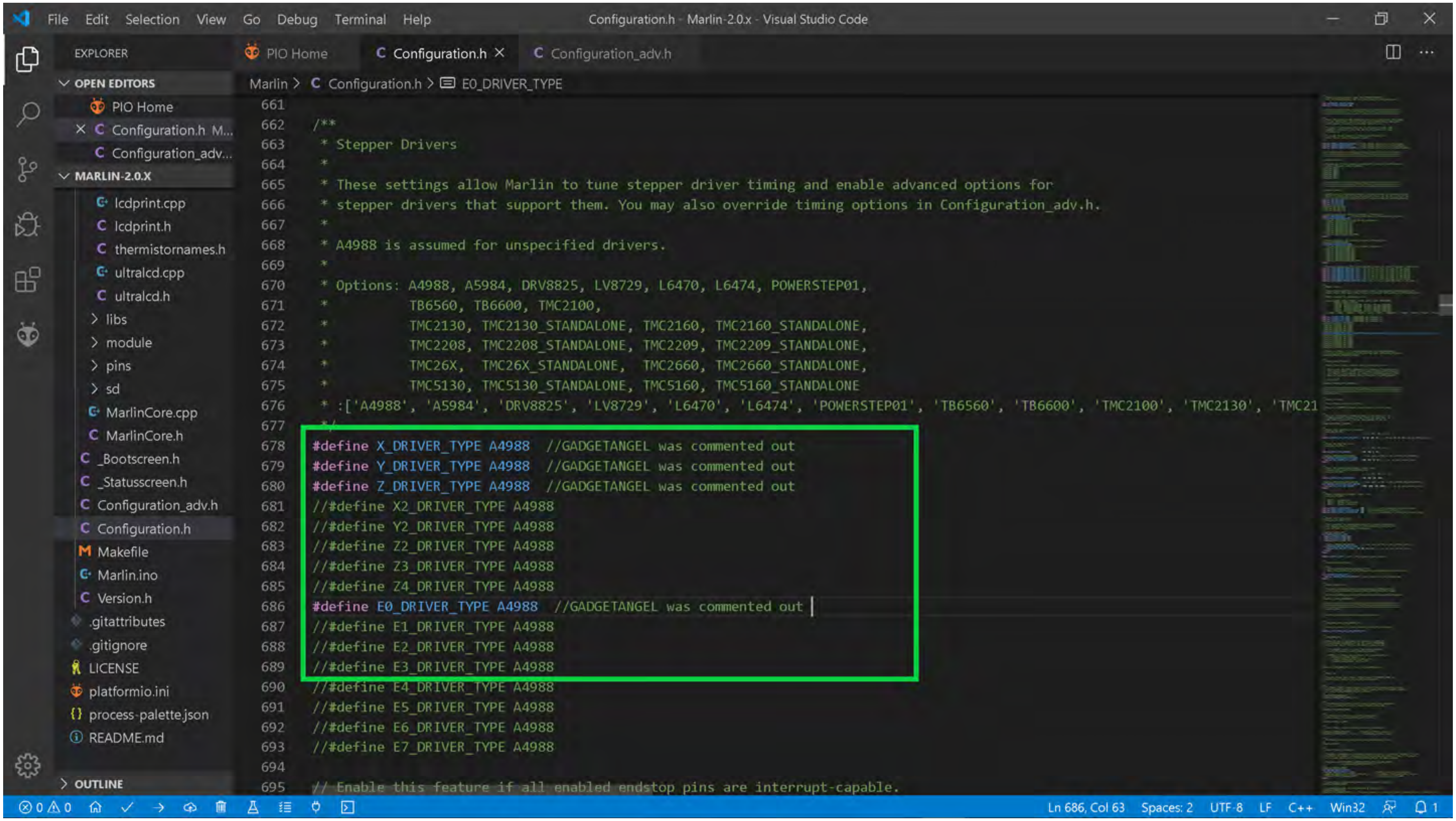


NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

The (latest release of) Marlin Setup for BIQU A4988 Drivers

NOTE: **Go to Appendix C,** and then come back here for the changes to Marlin for BIQU A4988 stepper motor drivers.

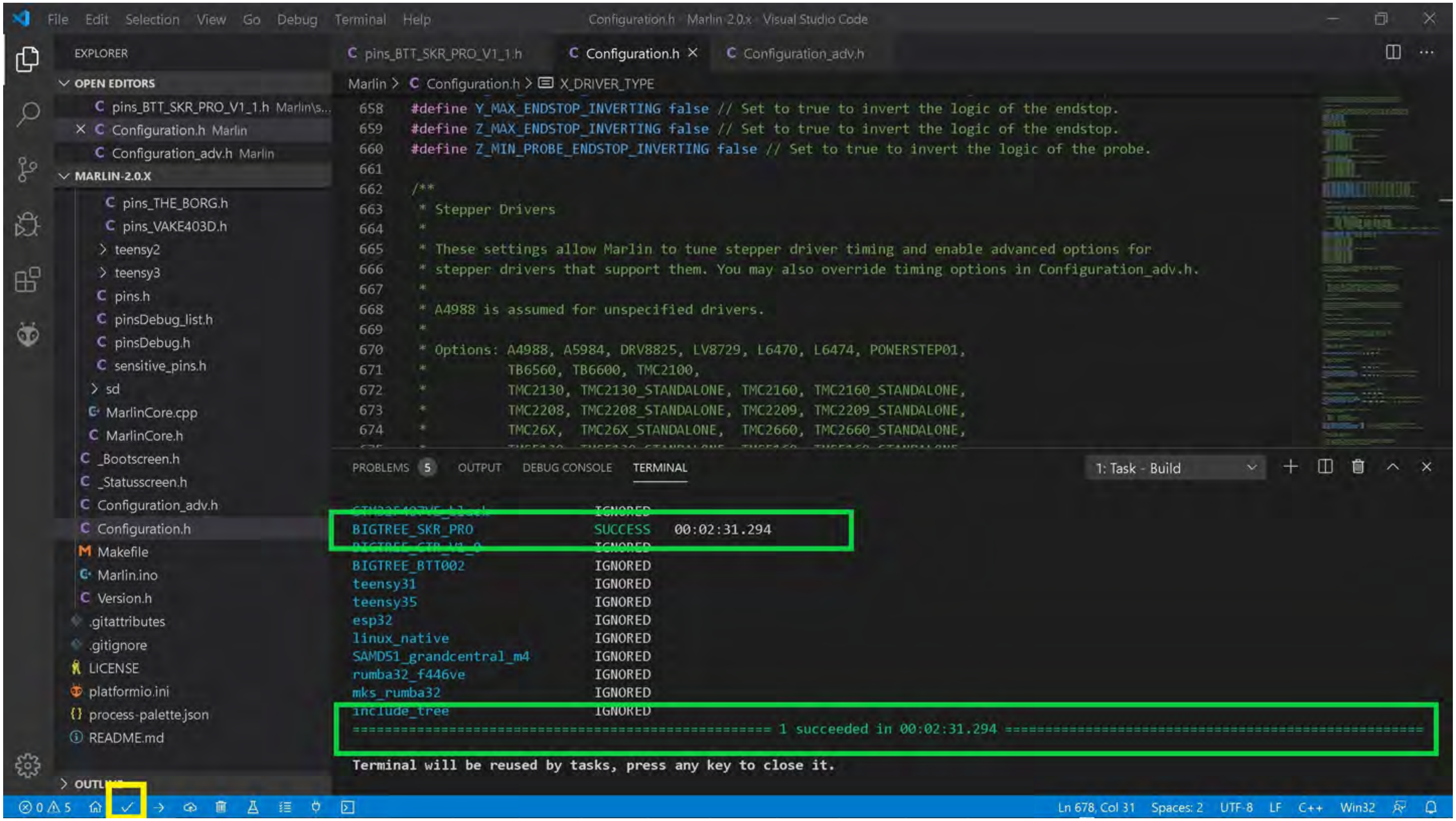
- Change the stepper motor drivers so that Marlin knows you are using BIQU A4988 drivers. Change one line for each axis and one for each extruder you will be using. See the picture below for an example of how to use BIQU A4988 drivers. When two "//" appear at the beginning of a line that means that line is commented out. To un-comment a line just remove the leading two forward slashes ("//").



- Go to the next page.

The (latest release of) Marlin Setup for BIQU A4988 Drivers

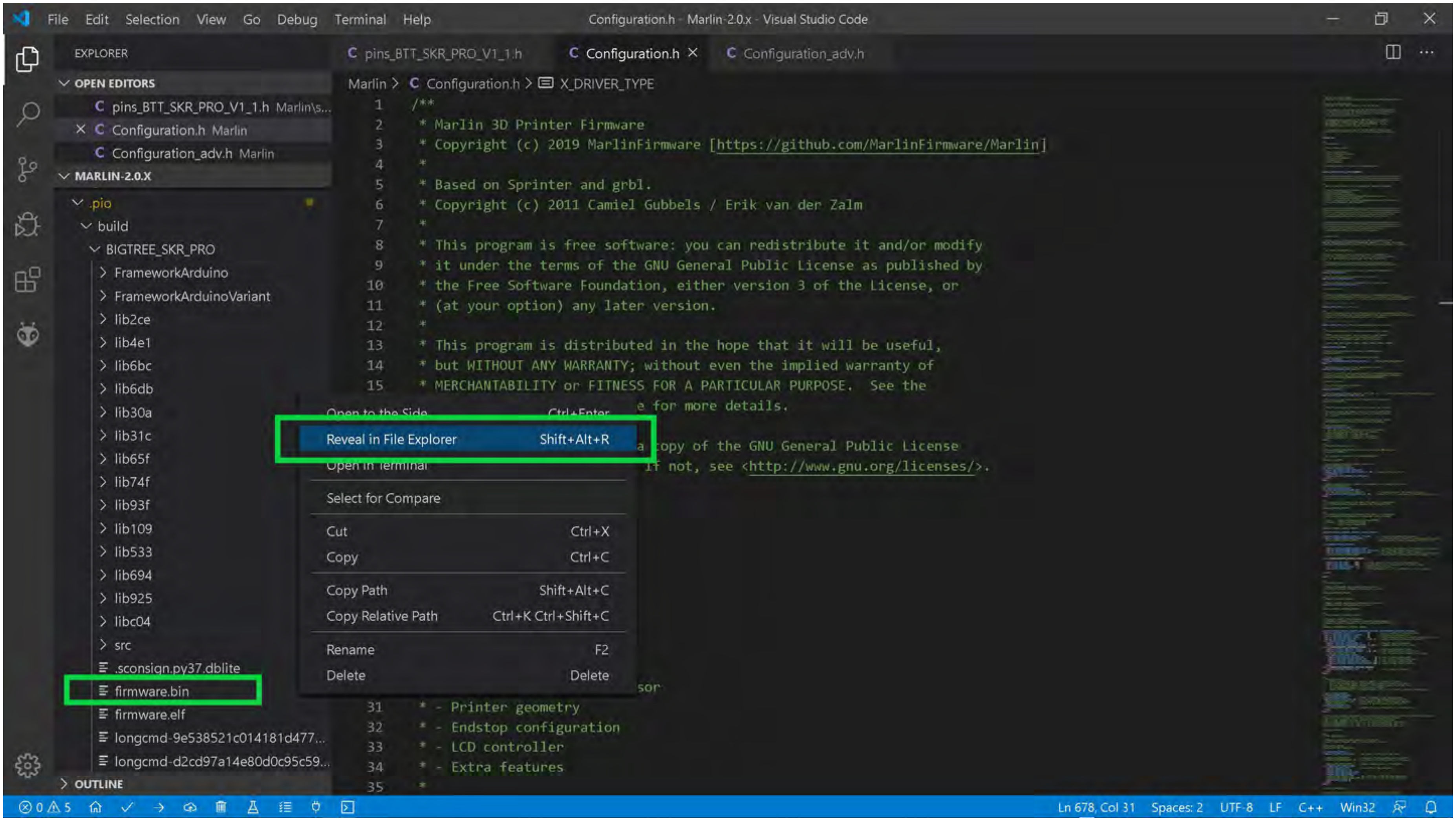
- The end of Marlin setup for BIQU A4988 drivers. Now, compile your code by clicking on the check mark, as seen in the **YELLOW** box below. Once the firmware has compiled successfully you will see the following messages, as shown in the **GREEN** boxes below.



- For the SKR PRO V1.1 board you must save the "firmware.bin" file to the micro SD card then place the micro SD card into the micro SD card reader of the SKR PRO V1.1 board and turn on the power to your printer for the NEW firmware to be automatically applied to the SKR PRO MCU. To locate the "firmware.bin" file, and learn how to transfer it to your micro SD card go to the next page.

The (latest release of) Marlin Setup for BIQU A4988 Drivers

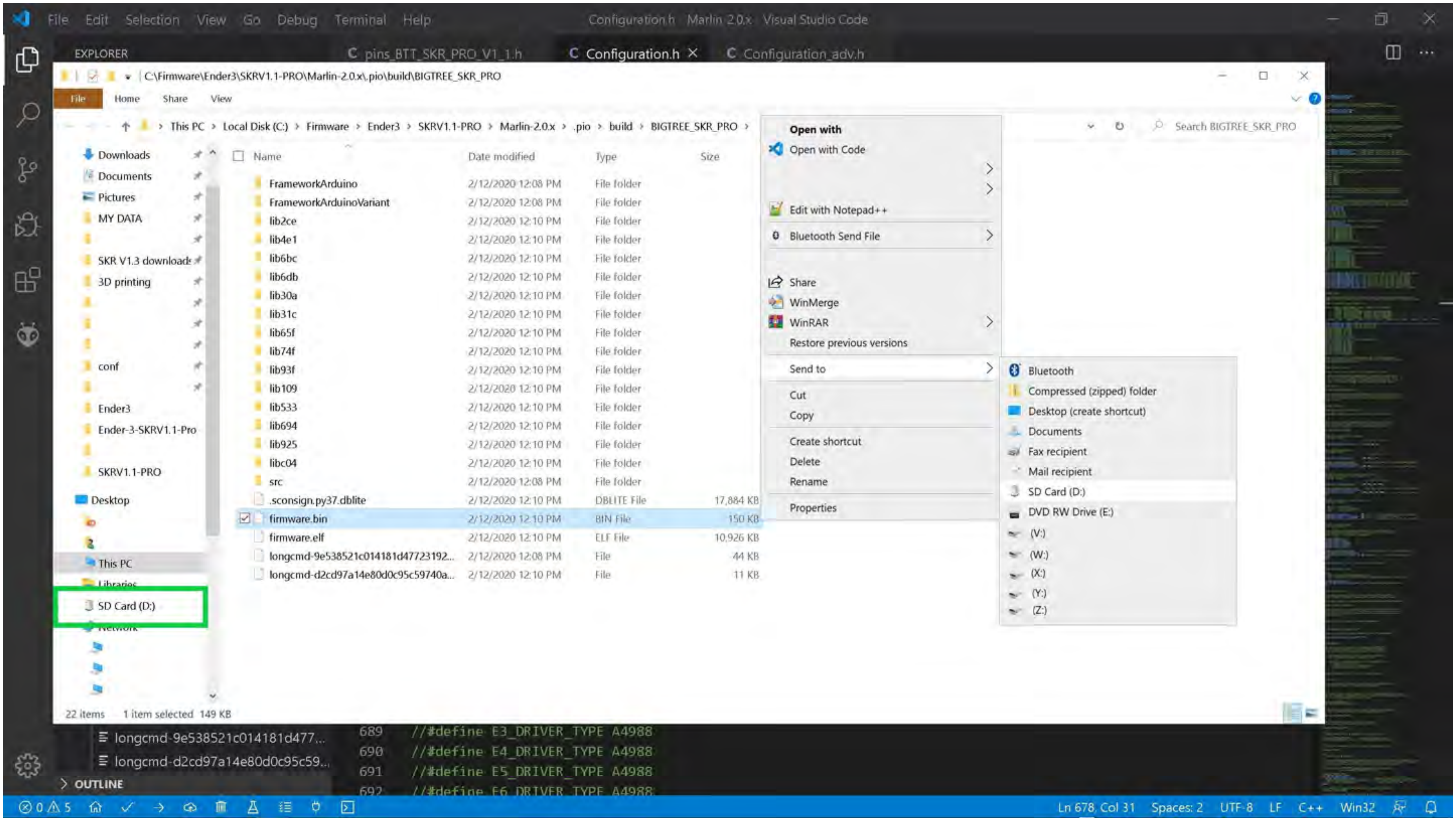
- The "firmware.bin" file can be found by using the left side of the VS code window and going to folder Marlin-2.0.X/.pio/build/ BIGTREE_SKR_PRO/ and right clicking on the "firmware.bin" file name to bring up the content menu. Select "Reveal in File Explorer" to have your Window's machine open a file explorer window.



- Go to the next page.

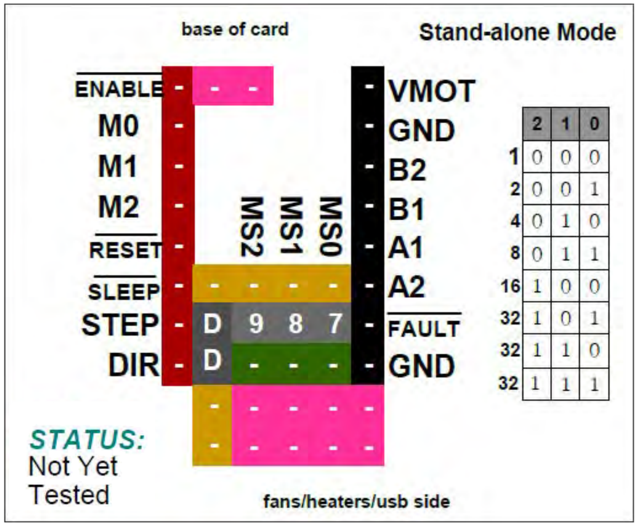
The (latest release of) Marlin Setup for BIQU A4988 Drivers

- Ensure your micro SD card, which came with your SKR PRO V1.1 board, is in a micro-to-SD-card adapter and plugged into your Window's SD card reader.
    - From the file explorer window right click on the "firmware.bin" file name and select "Send to" from the content menu. For my computer "D:" is my SD card reader. For your computer it may be a different drive letter. The "firmware.bin" file will be written to the SD card and renamed to "FIRMWARE.CUR".



- Right click on the SD card reader, as seen in the **GREEN** box above, and select "Eject" from the context menu. Take the micro SD card out of the adapter and place it into the micro SD card reader of the SKR PRO V1.1 board. Turn on the power to the printer (i.e. the SKR PRO V1.1 board) and the "FIRMWARE.CUR" file will be uploaded to the board. After the upload the firmware file will be renamed to "firmware.bin" and stored on the micro SD card.

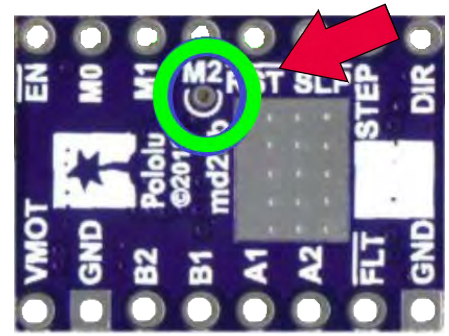
DRV8825



NOTE: **Use the potentiometer (POT) on the top of the board (or the board's "V_{ref} Test point") to adjust your V_{ref}.** See Appendix A for instructions on how to set the V_{ref} on a driver board.

Note: **Use 90% of the calculated V_{ref} when tuning the stepper driver board**

Note: "V_{ref} Test point" location is on the bottom of the driver board, as shown in GREEN



Driver Chip	MS2	MS1	MS0	Microstep Resolution	Excitation Mode
Pololu DRV8825 Maximum 32 Subdivision 45V DC 2.2A (peak)	Low	Low	Low	Full step	2 Phase
	Low	Low	High	Half step	1-2 Phase
	Low	High	Low	1/4 step	W1-2 Phase
	Low	High	High	1/8 step	2W1-2 Phase
	High	Low	Low	1/16 step	4W1-2 Phase
	High	Low	High	1/32 step	8W1-2 Phase
	High	High	Low	1/32 step	8W1-2 Phase
	High	High	High	1/32 step	8W1-2 Phase

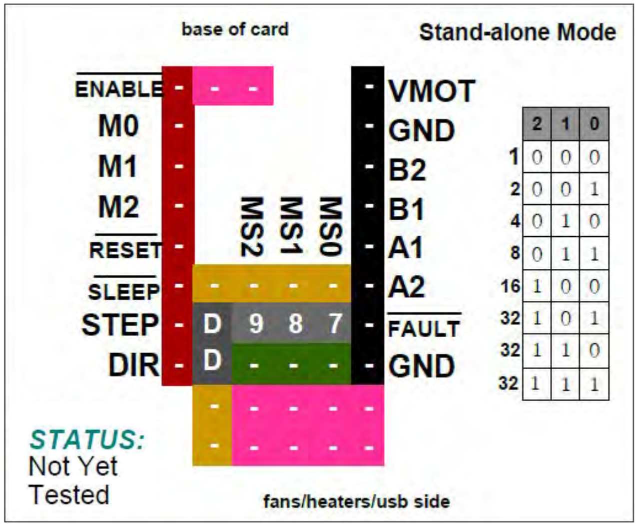
Driving Current Calculation Formula

R_S(Typical Sense Resistor)$=0.1\Omega$

$$I_{MAX} = \frac{V_{ref}}{5 * R_S}$$

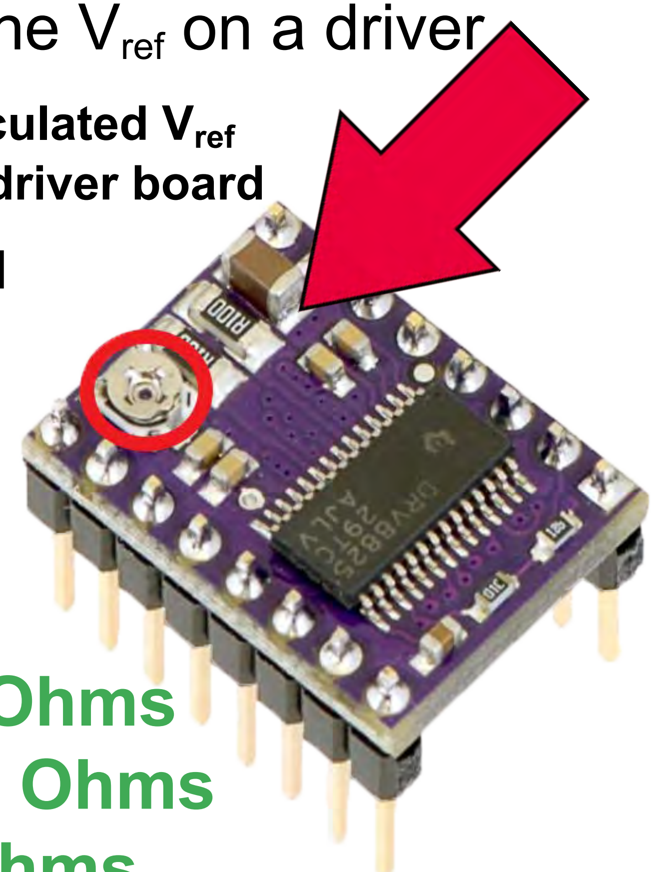
$$V_{ref} = 5 * I_{MAX} * R_S$$

DRV8825



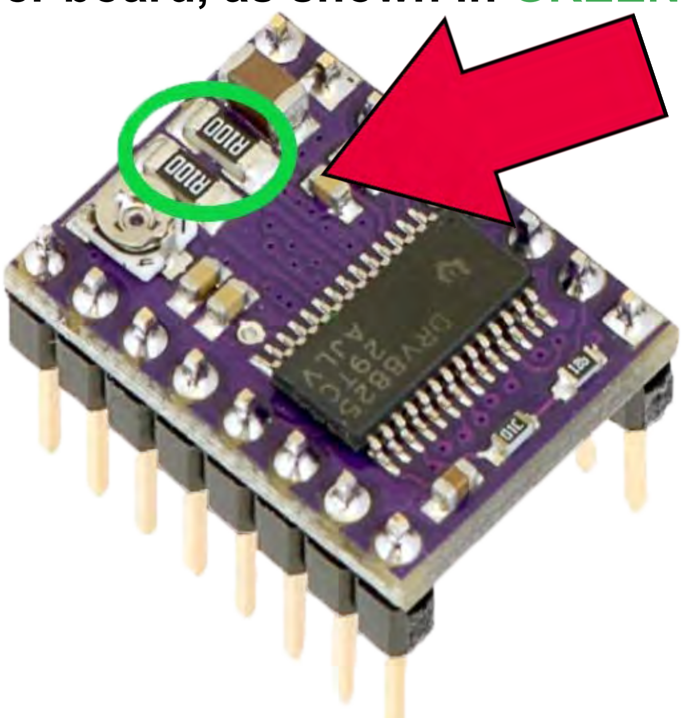
NOTE: **Use the potentiometer (POT) on the top of the board (or the board's "V_{ref} Test point") to adjust your V_{ref}.** See **Appendix A** for instructions on how to set the V_{ref} on a driver board.

Note: **Use 90% of the calculated V_{ref} when tuning the stepper driver board**



Note: See this video about current sense resistors (R_s) and their possible locations: **https://youtu.be/8wk1elugv5A**

Note: Check your current sense resistors (R_s) values on the driver board, as shown in GREEN below.



R_s = R050 is 0.05 Ohms

R_s = R068 is 0.068 Ohms

R_s = R100 is 0.1 Ohms

R_s = R150 is 0.15 Ohms

R_s = R200 is 0.2 Ohms

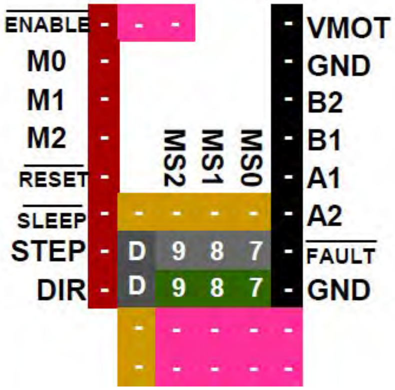
R_s = R220 is 0.22 Ohms

NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

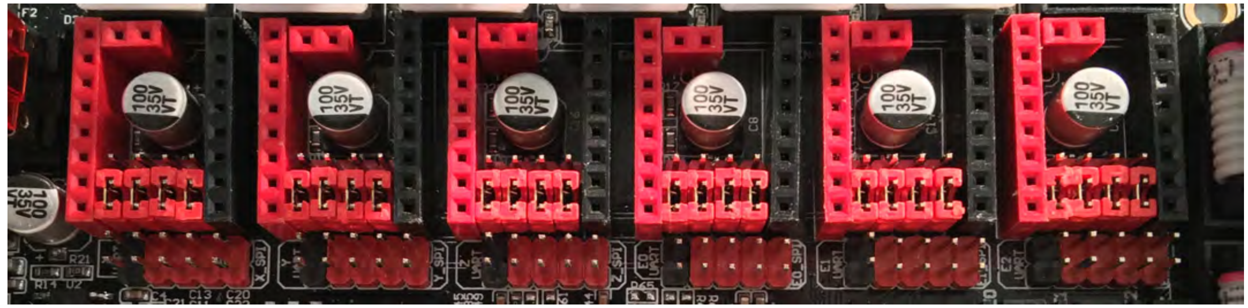
DRV8825

Stand-alone Mode

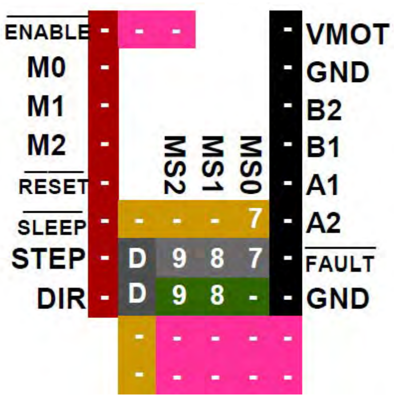
STEP



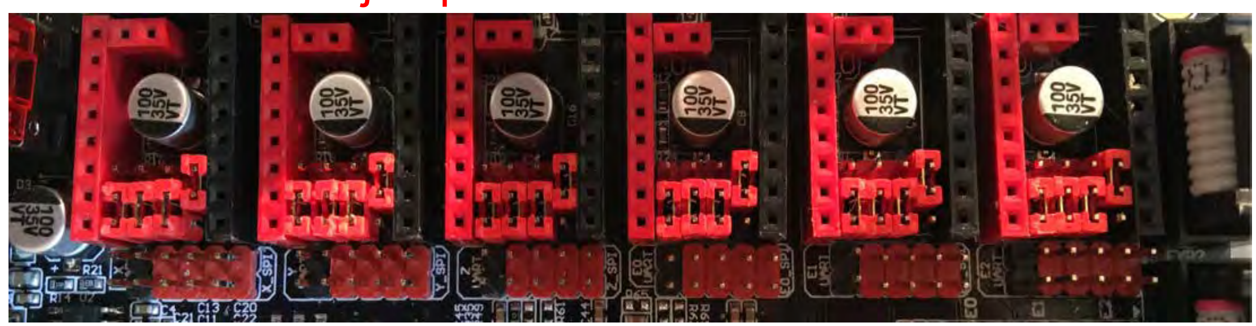
Note: The "D" jumper MUST be SET!



1 / 2



Note: The "D" jumper MUST be SET!

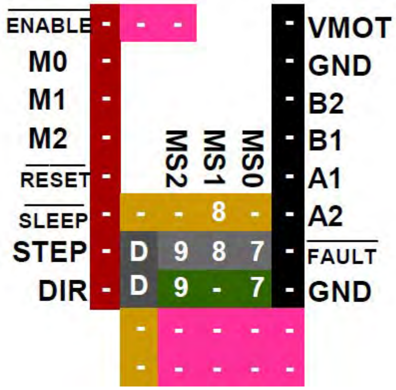


NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

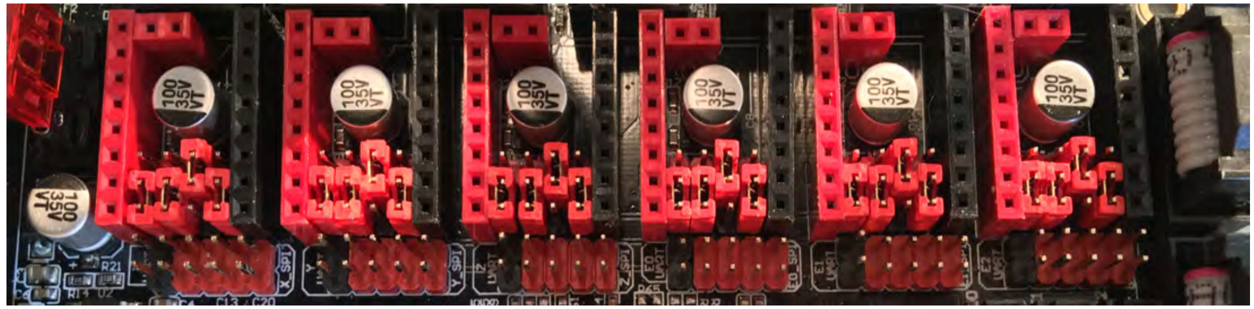
DRV8825

Stand-alone Mode

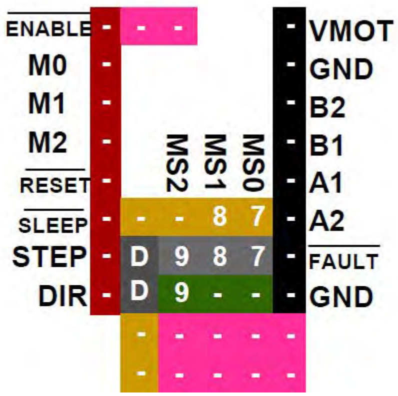
1 / 4



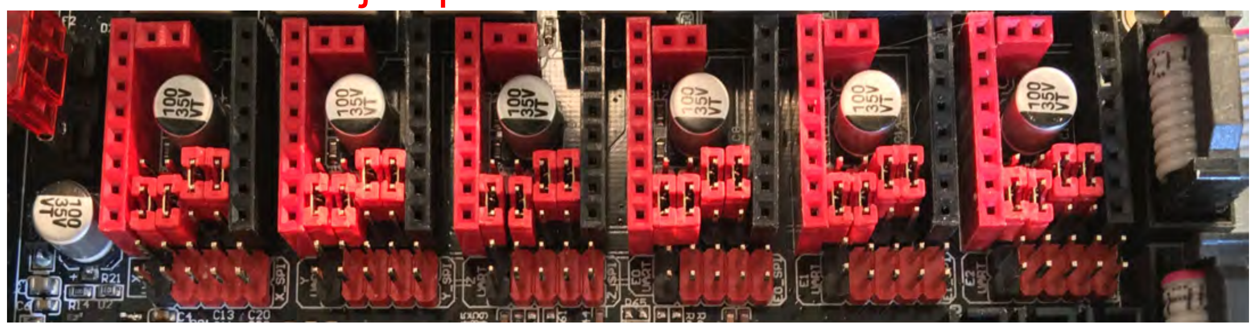
Note: The "D" jumper MUST be SET!



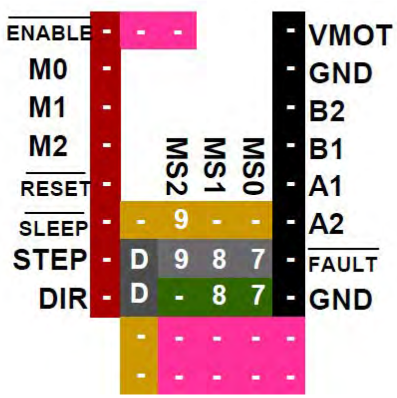
1 / 8



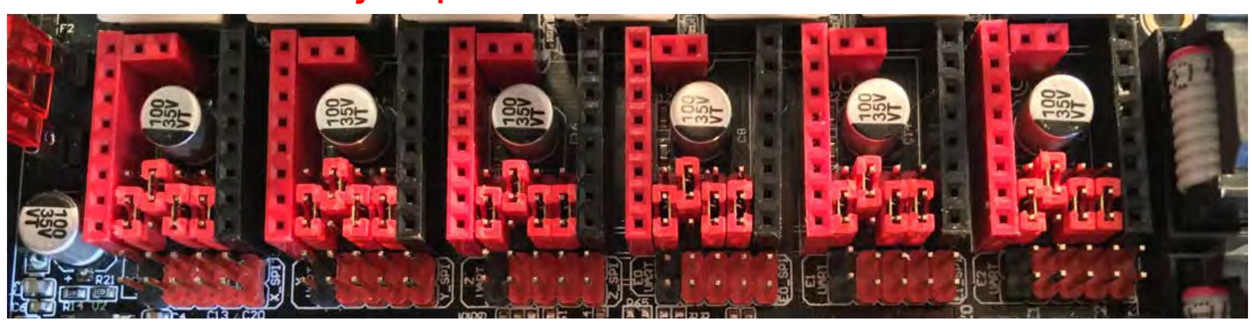
Note: The "D" jumper MUST be SET!



1 / 16



Note: The "D" jumper MUST be SET!



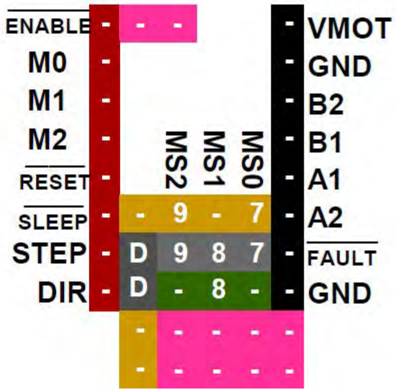
NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

DRV8825

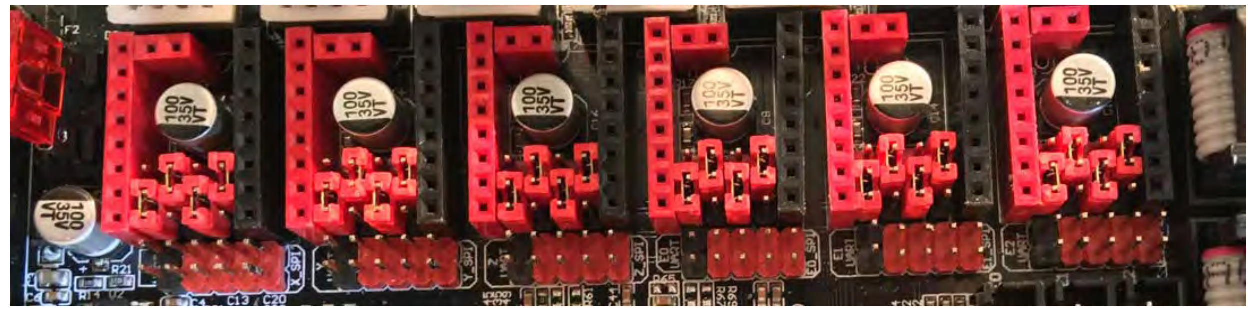
Stand-alone Mode

Note: All THREE of these settings will work for 1/32, choose your preference!!

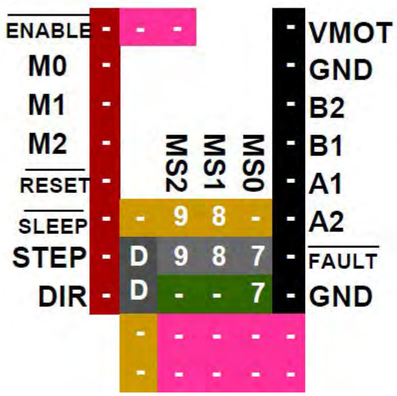
1 / 32



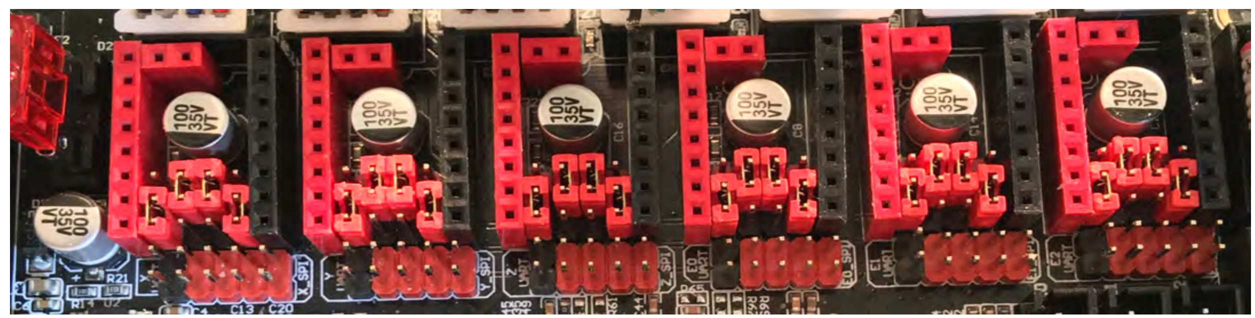
Note: The "D" jumper MUST be SET!



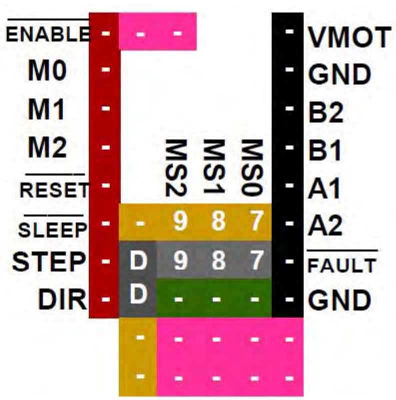
1 / 32



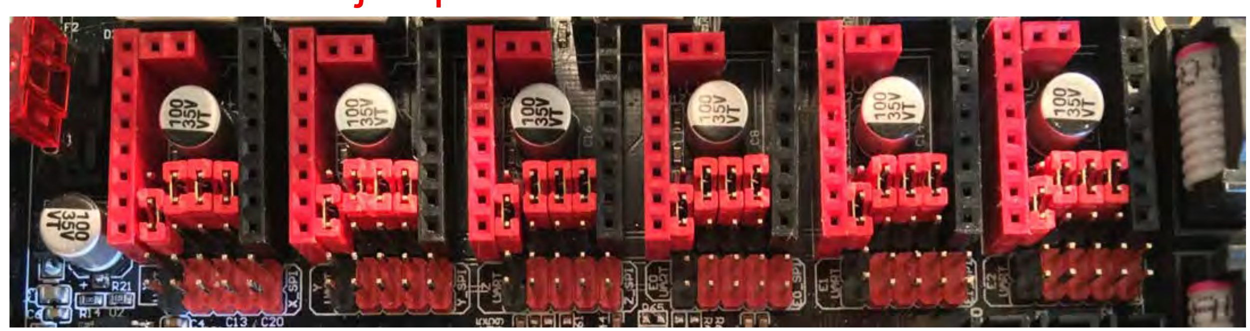
Note: The "D" jumper MUST be SET!



1 / 32



Note: The "D" jumper MUST be SET!

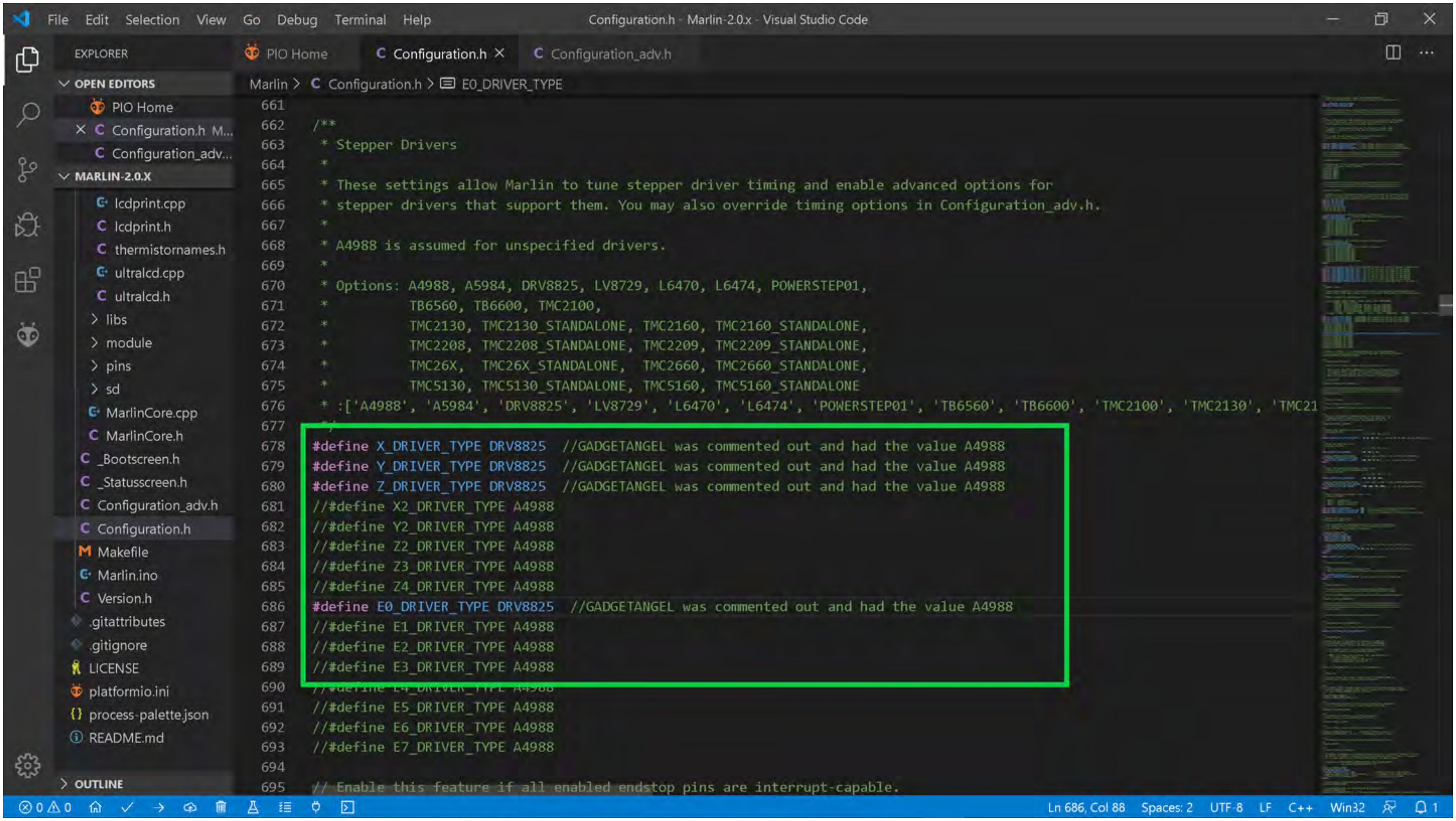


NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

The (latest release of) Marlin Setup for DRV8825 Drivers

NOTE: **Go to Appendix C**, and then come back here for the changes to Marlin for DRV8825 stepper motor drivers.

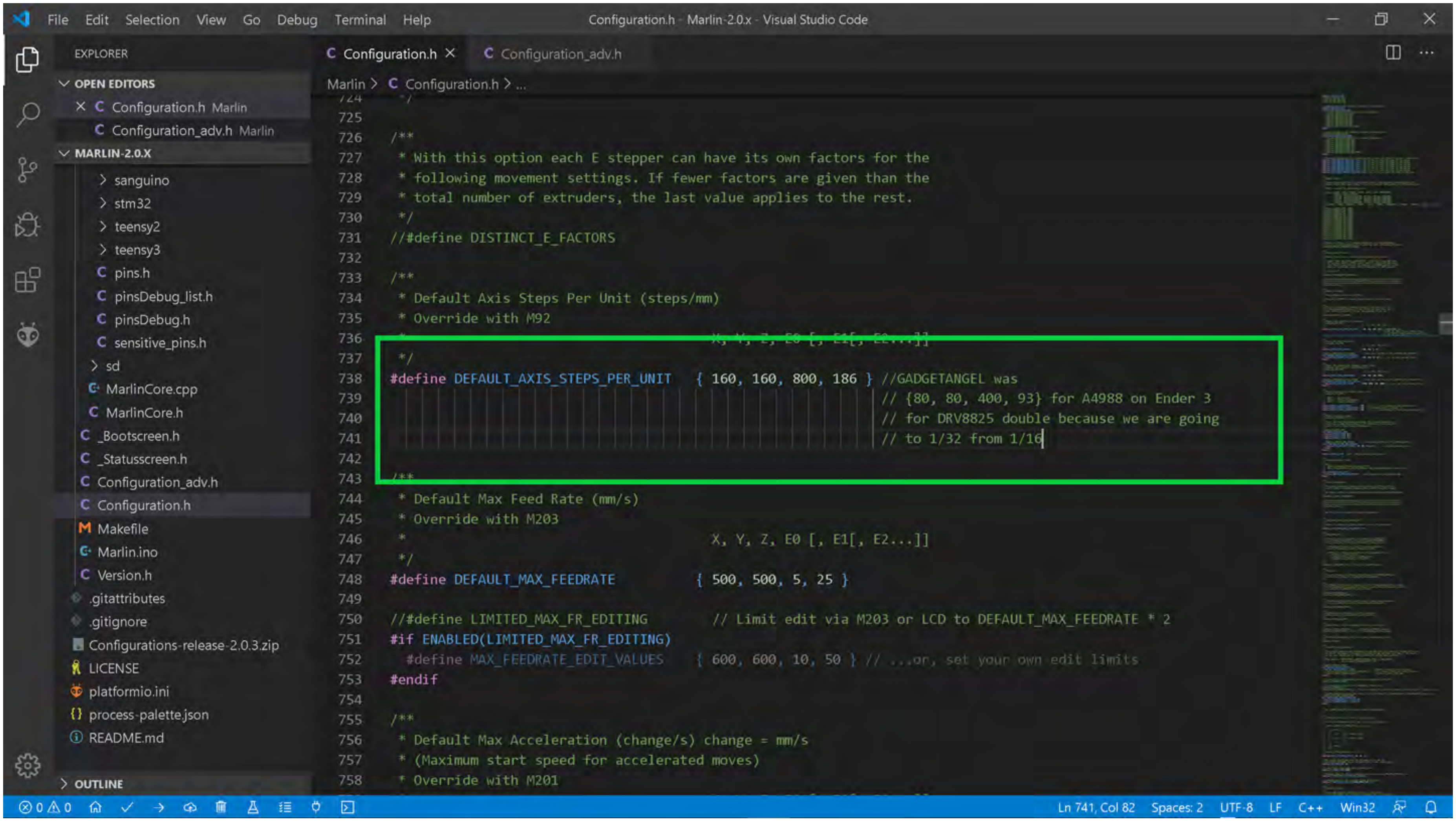
- Change the stepper motor drivers so that Marlin knows you are using DRV8825 drivers. Change one line for each axis and one for each extruder you will be using. See the picture below for an example of how to use DRV8825 drivers. When two "//" appear at the beginning of a line that means that line is commented out. To un-comment a line just remove the leading two forward slashes ("//").



- Go to the next page.

The (latest release of) Marlin Setup for DRV8825 Drivers

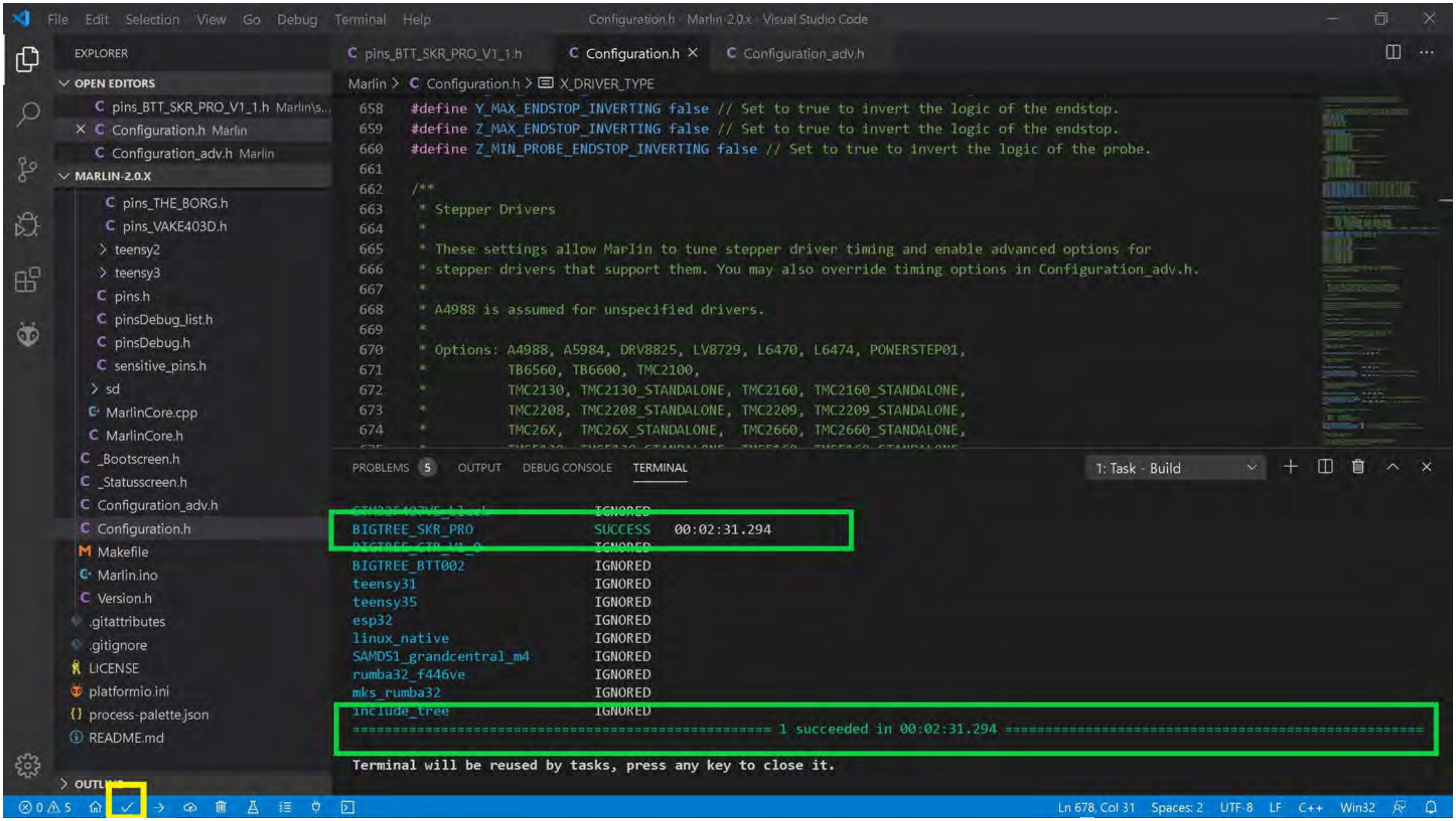
- We are changing from A4988 stepper motor drivers on the Ender 3 to DRV8825 stepper motor drivers for each axis and the extruder stepper motor driver. We will be going from 1/16 stepping to 1/32 stepping. So we are doubling our STEPS. Therefore, we must adjust our "DEFAULT_AXIS_STEPS_PER_UNIT" anytime our STEPS are NOT 1/16. So change "DEFAULT_AXIS_STEPS_PER_UNIT" to {160, 160, 800, 186}, as seen in the **GREEN** box below.



- Go to the next page.

The (latest release of) Marlin Setup for DRV8825 Drivers

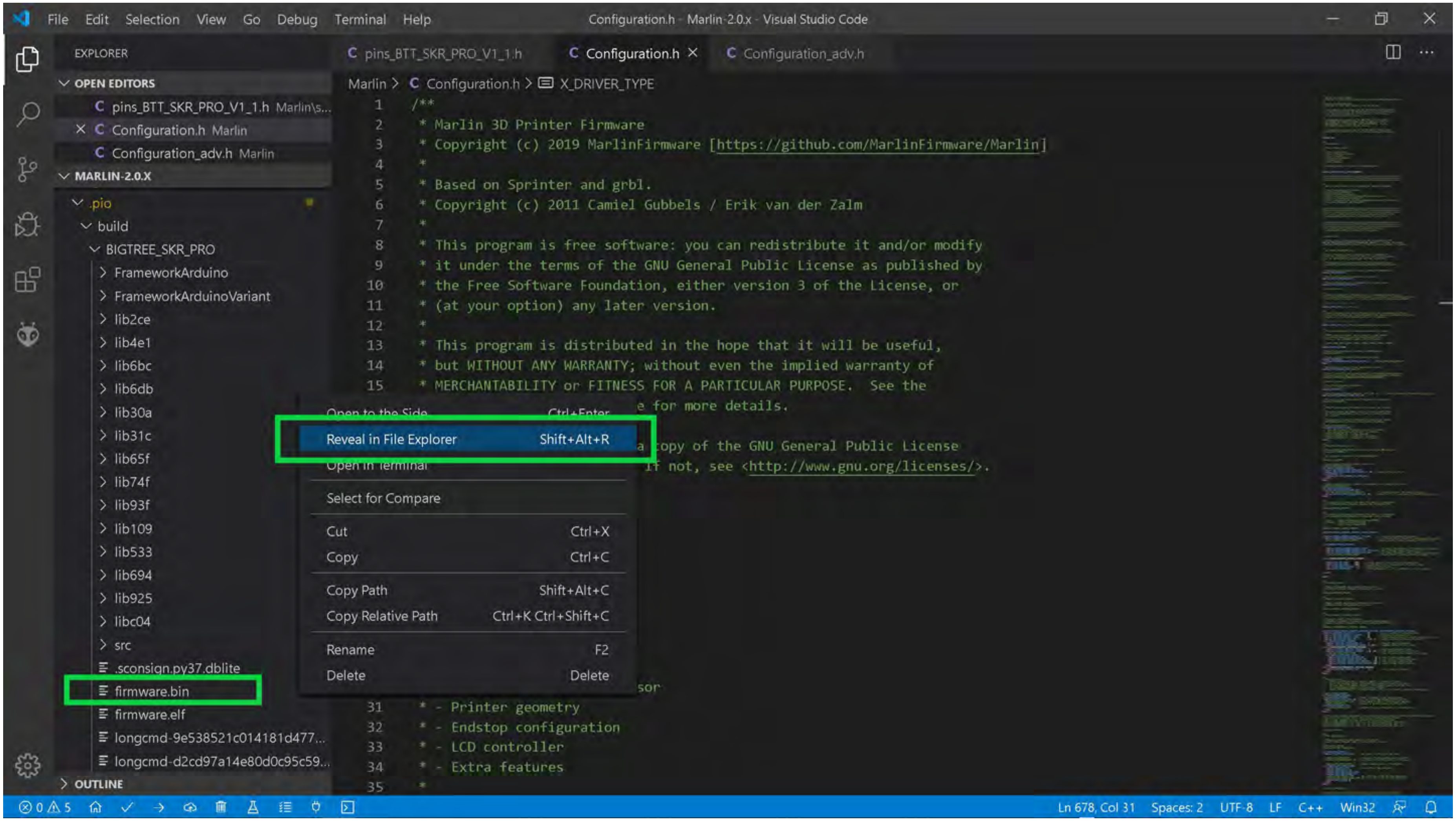
- The end of Marlin setup for DRV8825 drivers. Now, compile your code by clicking on the check mark, as seen in the **YELLOW** box below. Once the firmware has compiled successfully you will see the following messages, as shown in the **GREEN** boxes below.



- For the SKR PRO V1.1 board you must save the "firmware.bin" file to the micro SD card then place the micro SD card into the micro SD card reader of the SKR PRO V1.1 board and turn on the power to your printer for the NEW firmware to be automatically applied to the SKR PRO MCU. To locate the "firmware.bin" file, and learn how to transfer it to your micro SD card go to the next page.

The (latest release of) Marlin Setup for DRV8825 Drivers

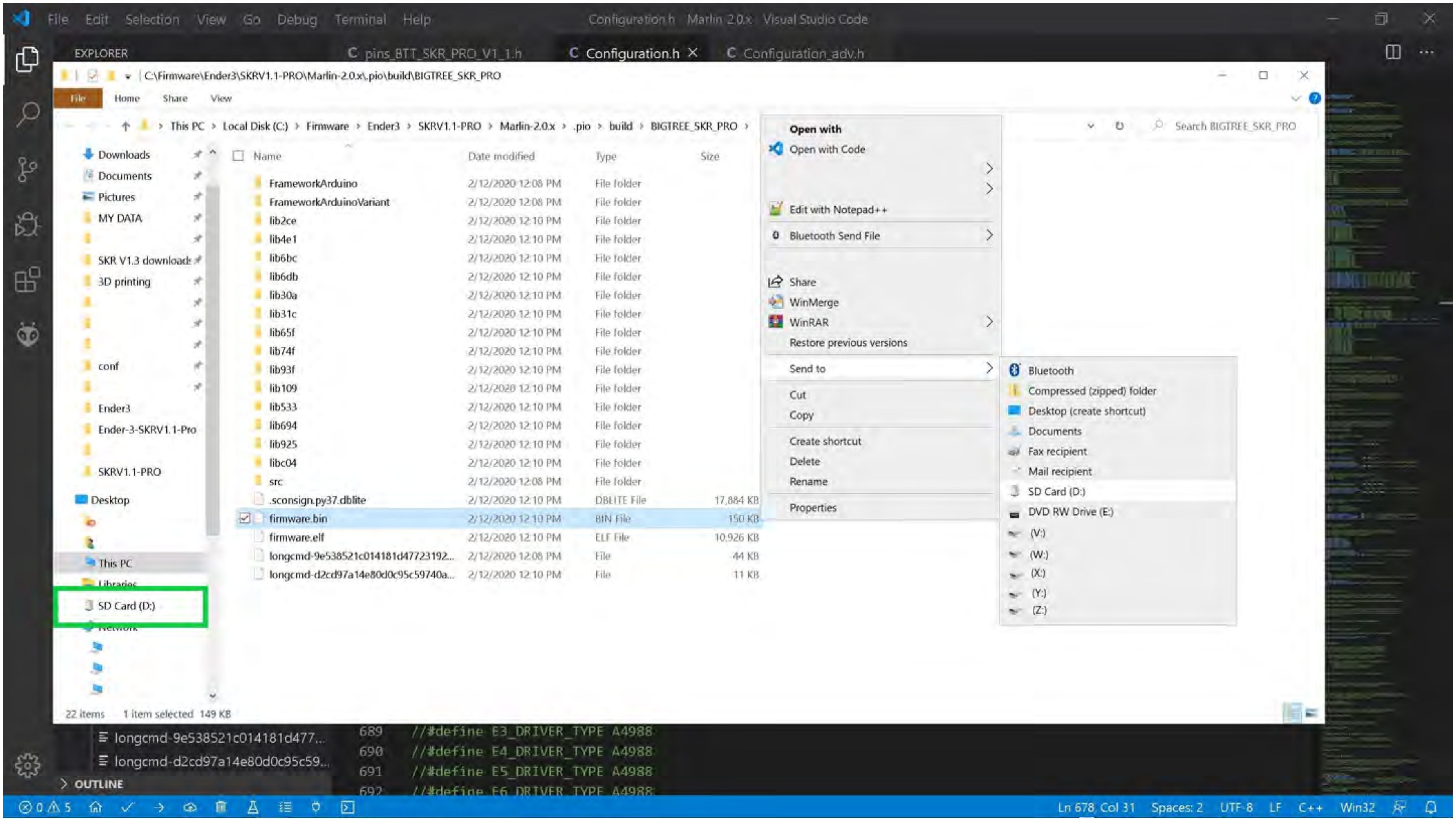
- The "firmware.bin" file can be found by using the left side of the VS code window and going to folder Marlin-2.0.X/.pio/build/ BIGTREE_SKR_PRO/ and right clicking on the "firmware.bin" file name to bring up the content menu. Select "Reveal in File Explorer" to have your Window's machine open a file explorer window.



- Go to the next page.

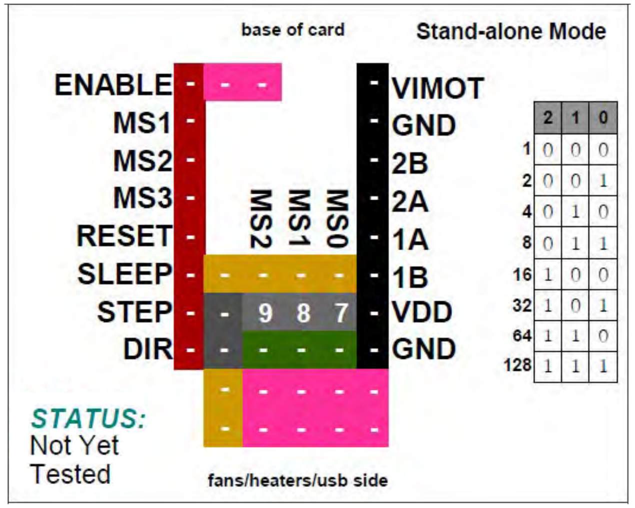
The (latest release of) Marlin Setup for DRV8825 Drivers

- Ensure your micro SD card, which came with your SKR PRO V1.1 board, is in a micro-to-SD-card adapter and plugged into your Window's SD card reader.
    - From the file explorer window right click on the "firmware.bin" file name and select "Send to" from the content menu. For my computer "D:" is my SD card reader. For your computer it may be a different drive letter. The "firmware.bin" file will be written to the SD card and renamed to "FIRMWARE.CUR".



- Right click on the SD card reader, as seen in the **GREEN** box above, and select "Eject" from the context menu. Take the micro SD card out of the adapter and place it into the micro SD card reader of the SKR PRO V1.1 board. Turn on the power to the printer (i.e. the SKR PRO V1.1 board) and the "FIRMWARE.CUR" file will be uploaded to the board. After the upload the firmware file will be renamed to "firmware.bin" and stored on the micro SD card.

BIQU LV8729



Note: **See the next page for information about location of the current sense resistors and how to set V_{ref} on the stepper motor driver board.**

Driver Chip	MS2	MS1	MS0	Microstep Resolution	Excitation Mode
BIQU® LV8729 Maximum 128 Subdivision 36V DC 1.5A (peak)	Low	Low	Low	Full Step	2 Phase
	Low	Low	High	1/2 Step	1-2 Phase
	Low	High	Low	1/4 Step	W1-2 Phase
	Low	High	High	1/8 Step	2W1-2 Phase
	High	Low	Low	1/16 Step	4W1-2 Phase
	High	Low	High	1/32 Step	8W1-2 Phase
	High	High	Low	1/64 Step	16W1-2 Phase
	High	High	High	1/128 Step	32W1-2 Phase

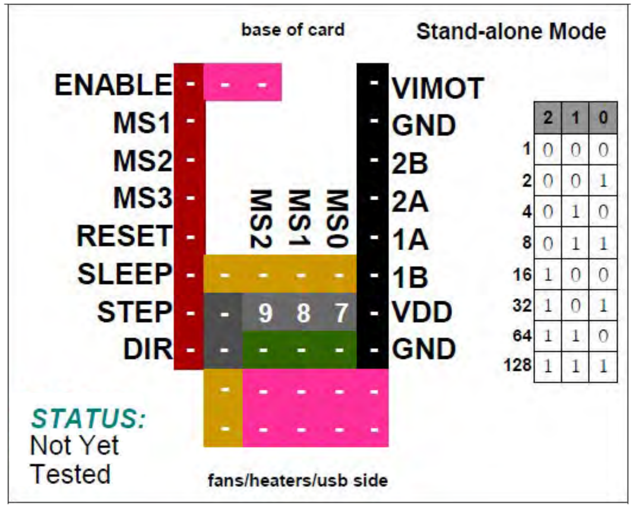
Driving Current Calculation Formula

R_S(Typical Sense Resistor)=0.22Ω

$$I_{MAX} = \frac{V_{ref}}{5 * R_S}$$

$$V_{ref} = 5 * I_{MAX} * R_S$$

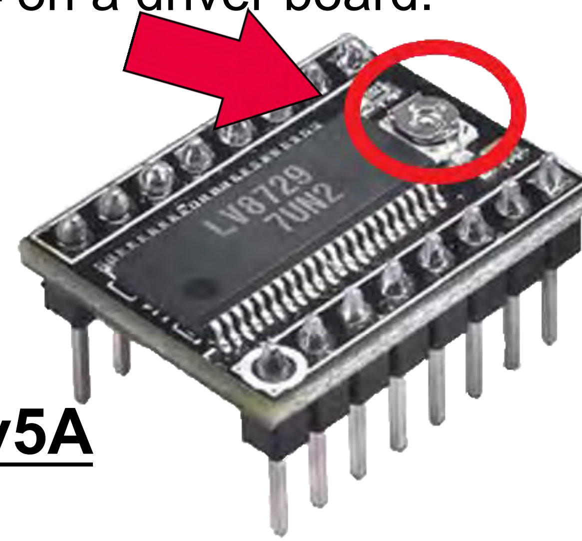
NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.



	2	1	0
1	0	0	0
2	0	0	1
4	0	1	0
8	0	1	1
16	1	0	0
32	1	0	1
64	1	1	0
128	1	1	1

BIQU LV8729

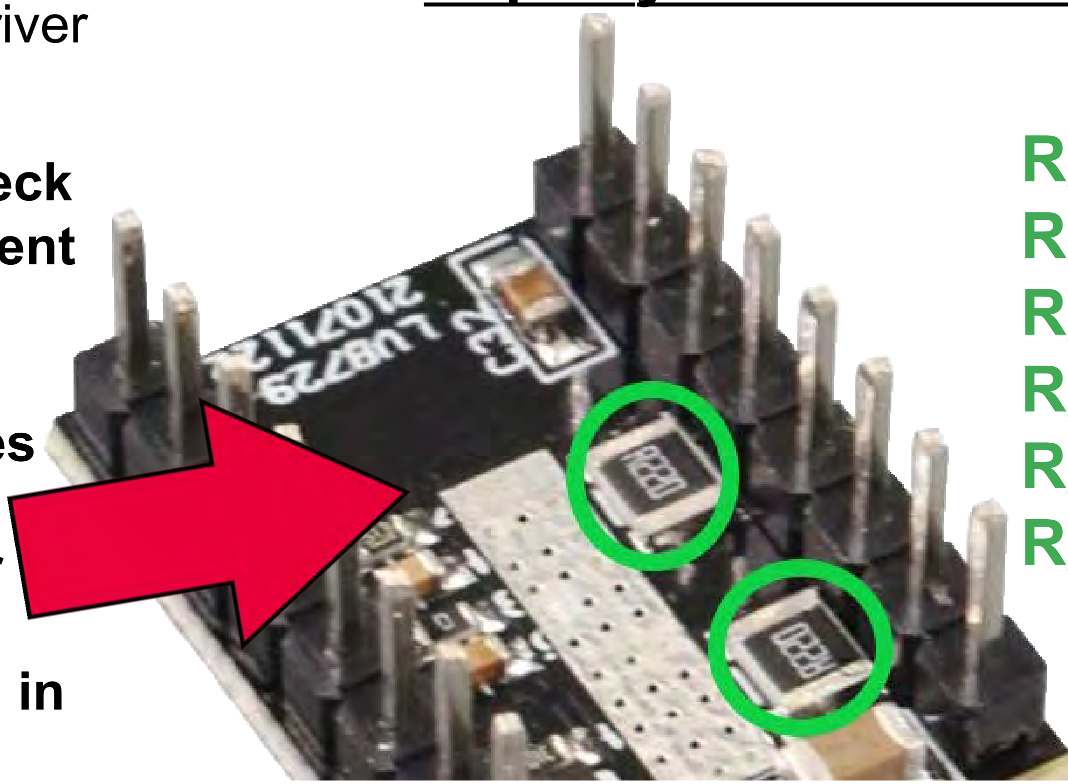
NOTE: Use the potentiometer (POT) on the top of the board to adjust your V_{ref}. See **Appendix A** for instructions on how to set the V_{ref} on a driver board.



Note: See this video about current sense resistors (R_s) and their possible locations: **https://youtu.be/8wk1elugv5A**

Note: Use 90% of the calculated V_{ref} when tuning the stepper driver board.

Note: Check your current sense resistors (R_s) values on the driver board, as shown in GREEN

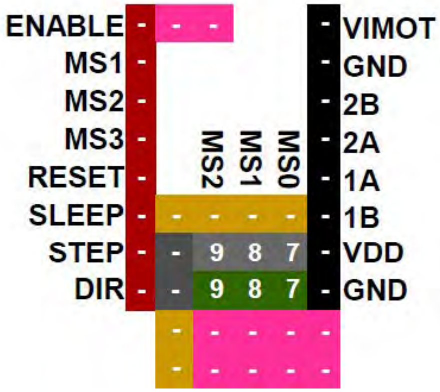


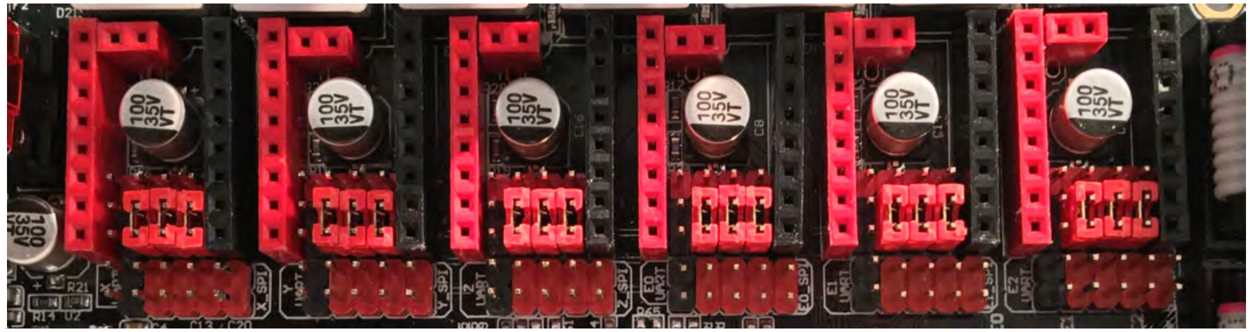
R_s = R050 is 0.05 Ohms
R_s = R068 is 0.068 Ohms
R_s = R100 is 0.1 Ohms
R_s = R150 is 0.15 Ohms
R_s = R200 is 0.2 Ohms
R_s = R220 is 0.22 Ohms

BIQU LV8729

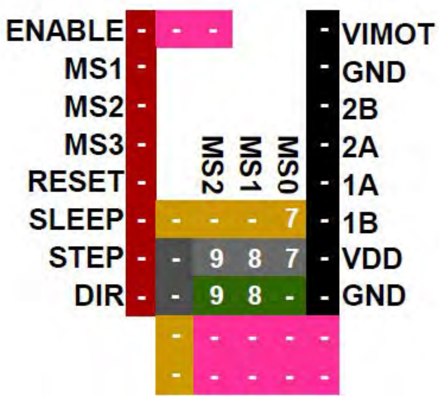
Stand-alone Mode

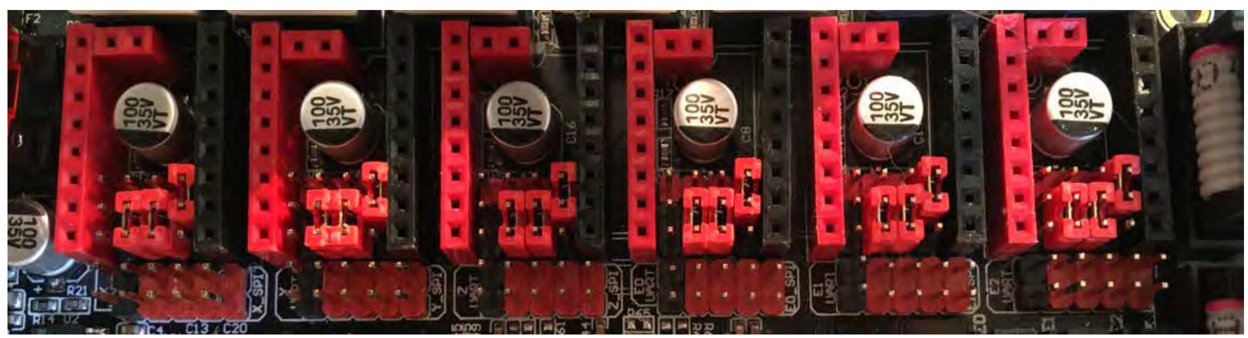
STEP





1 / 2

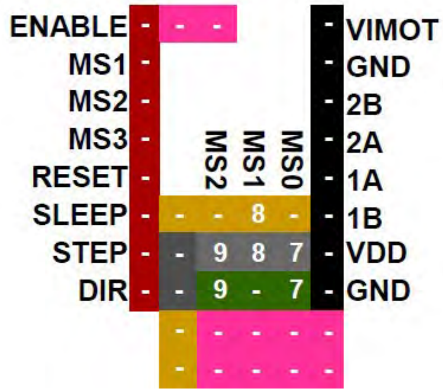


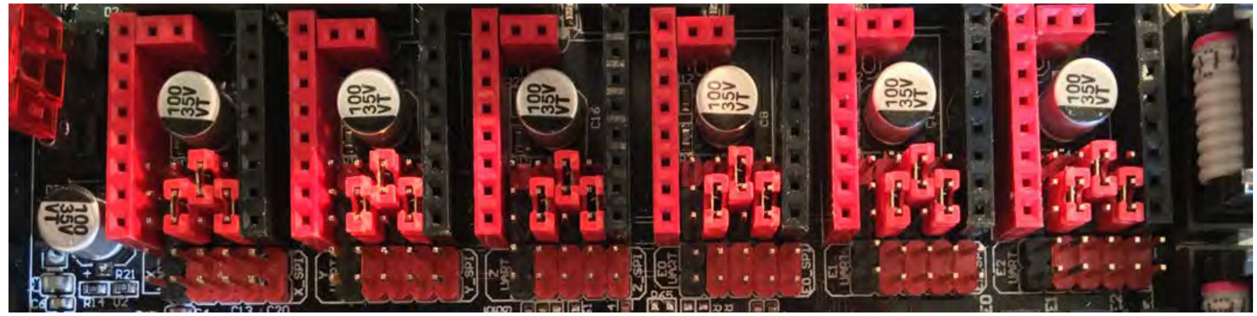


BIQU LV8729

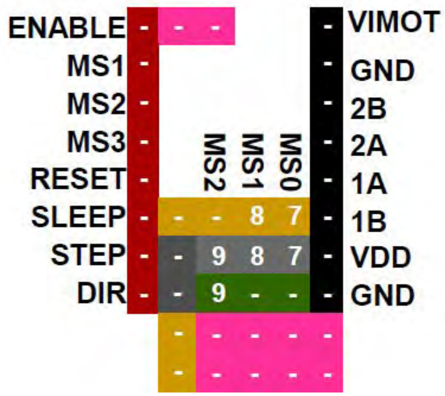
Stand-alone Mode

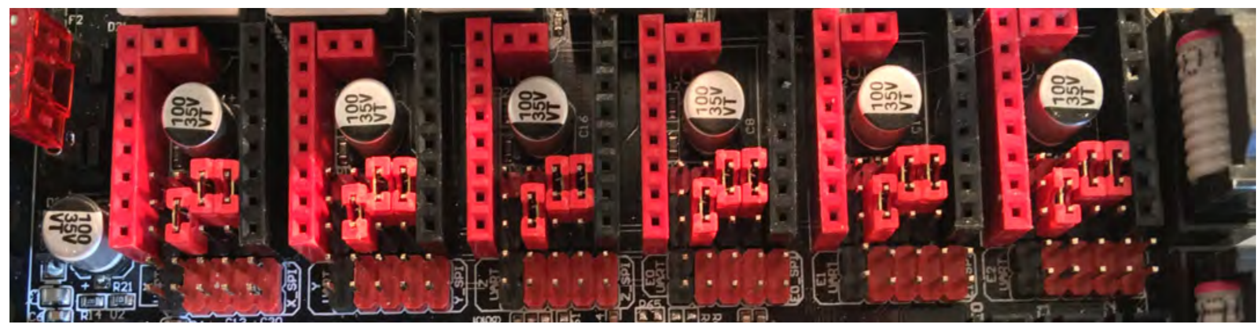
1 / 4



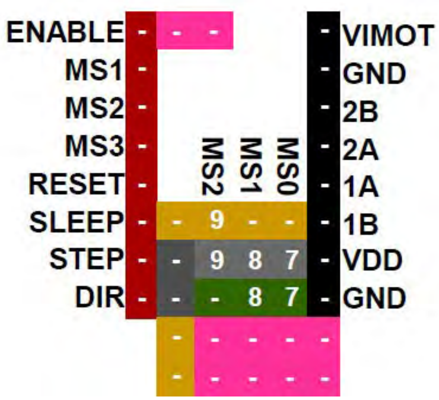


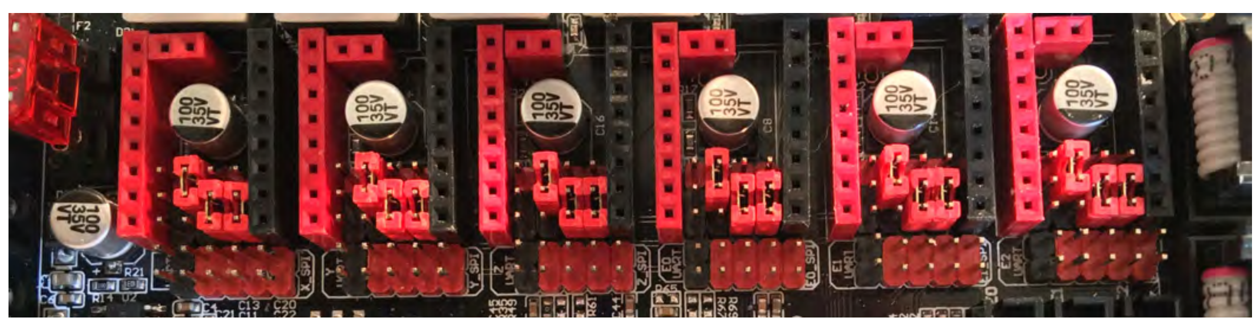
1 / 8





1 / 16



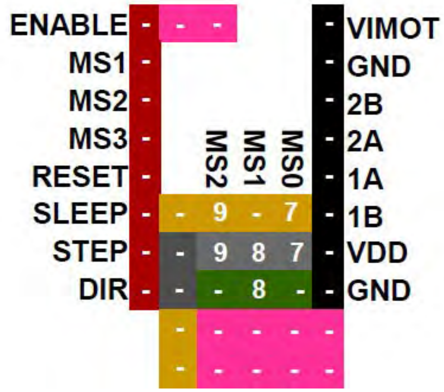


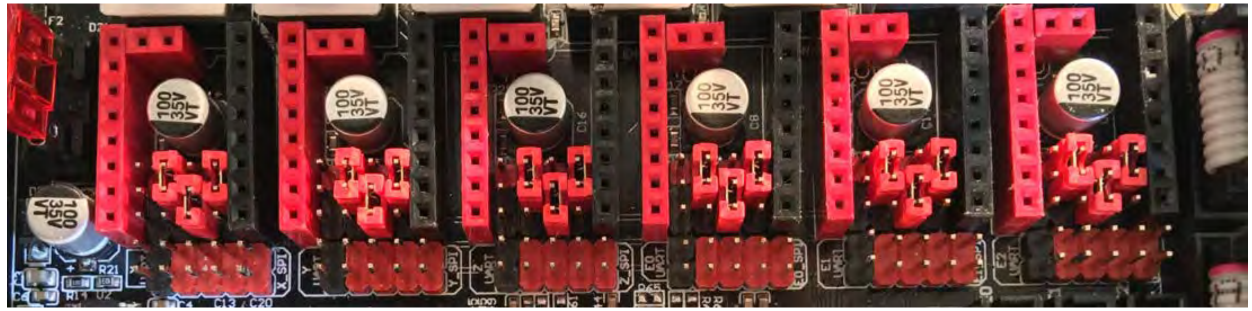
NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

BIQU LV8729

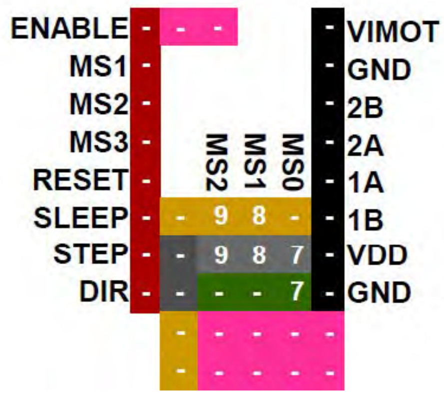
Stand-alone Mode

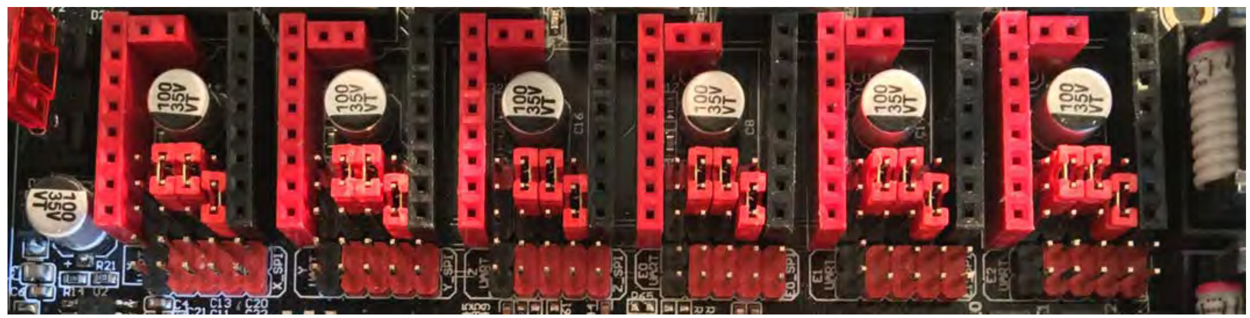
1 / 32



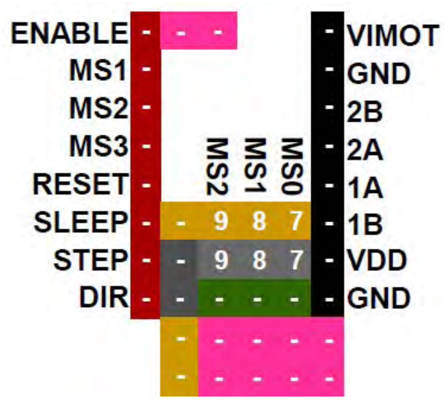


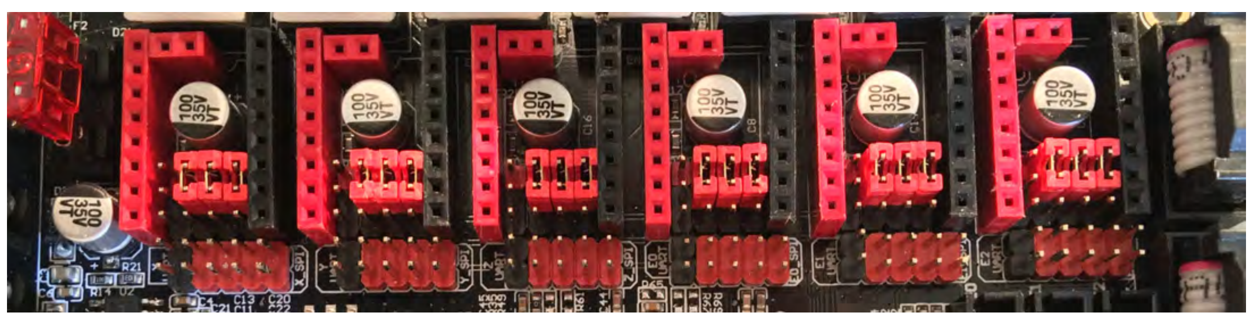
1 / 64





1 / 128



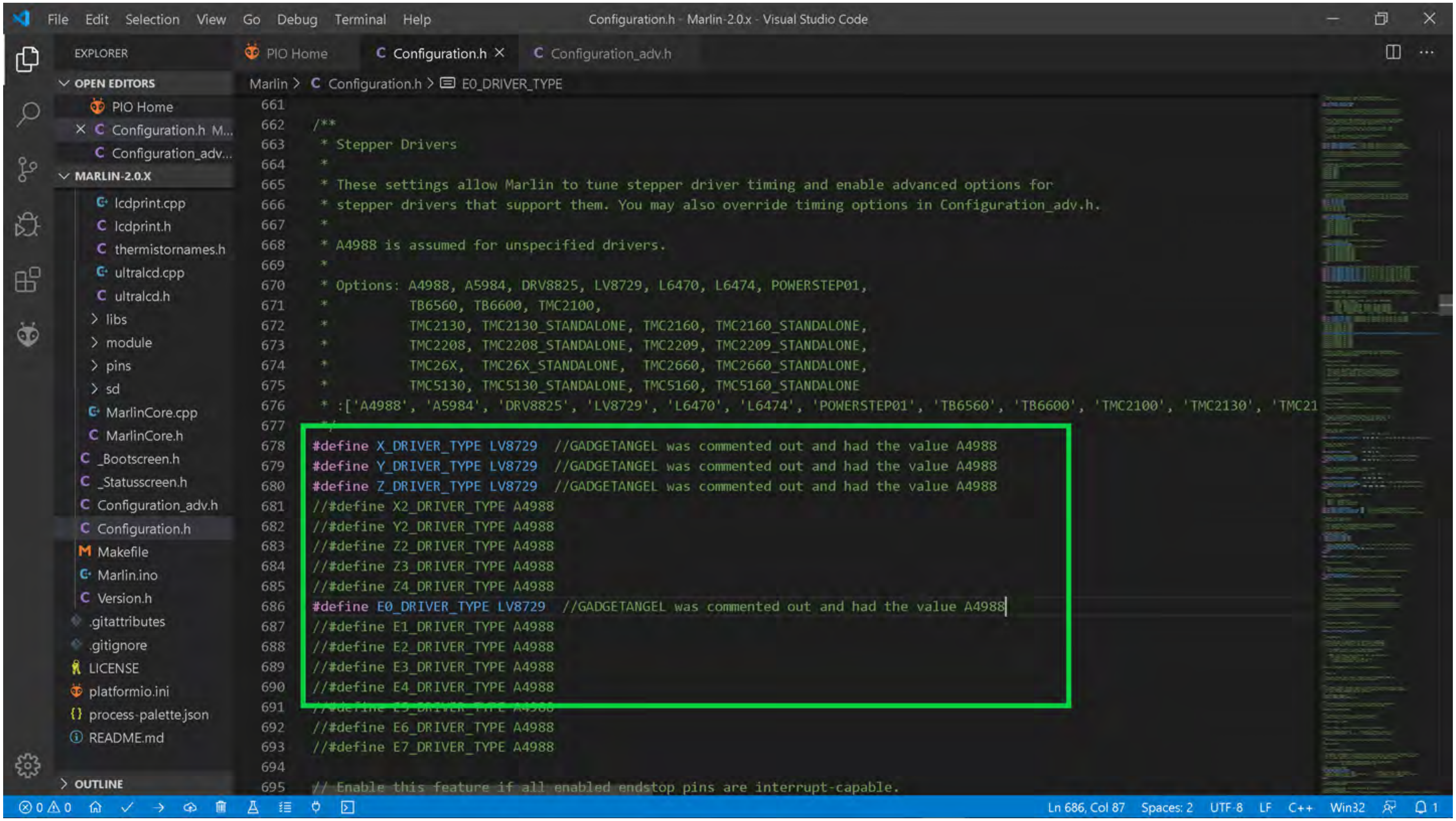


NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

The (latest release of) Marlin Setup for BIQU LV8729 Drivers

NOTE: **Go to Appendix C**, and then come back here for the changes to Marlin for BIQU LV8729 stepper motor drivers.

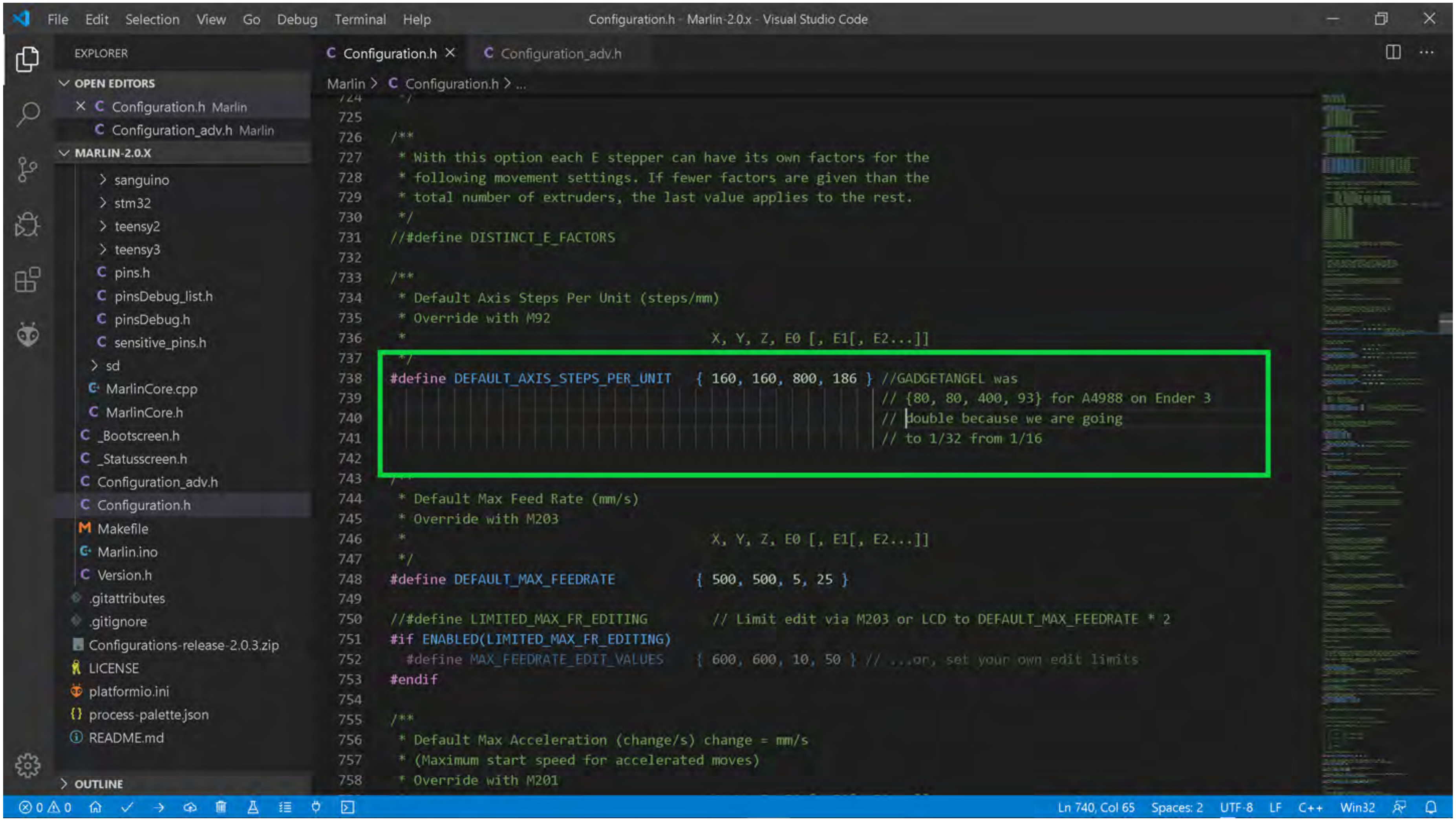
- Change the stepper motor drivers so that Marlin knows you are using BIQU LV8729 drivers. Change one line for each axis and one for each extruder you will be using. See the picture below for an example of how to use BIQU LV8729 drivers. When two "//" appear at the beginning of a line that means that line is commented out. To un-comment a line just remove the leading two forward slashes ("//").



- Go to the next page.

The (latest release of) Marlin Setup for BIQU LV8729 Drivers

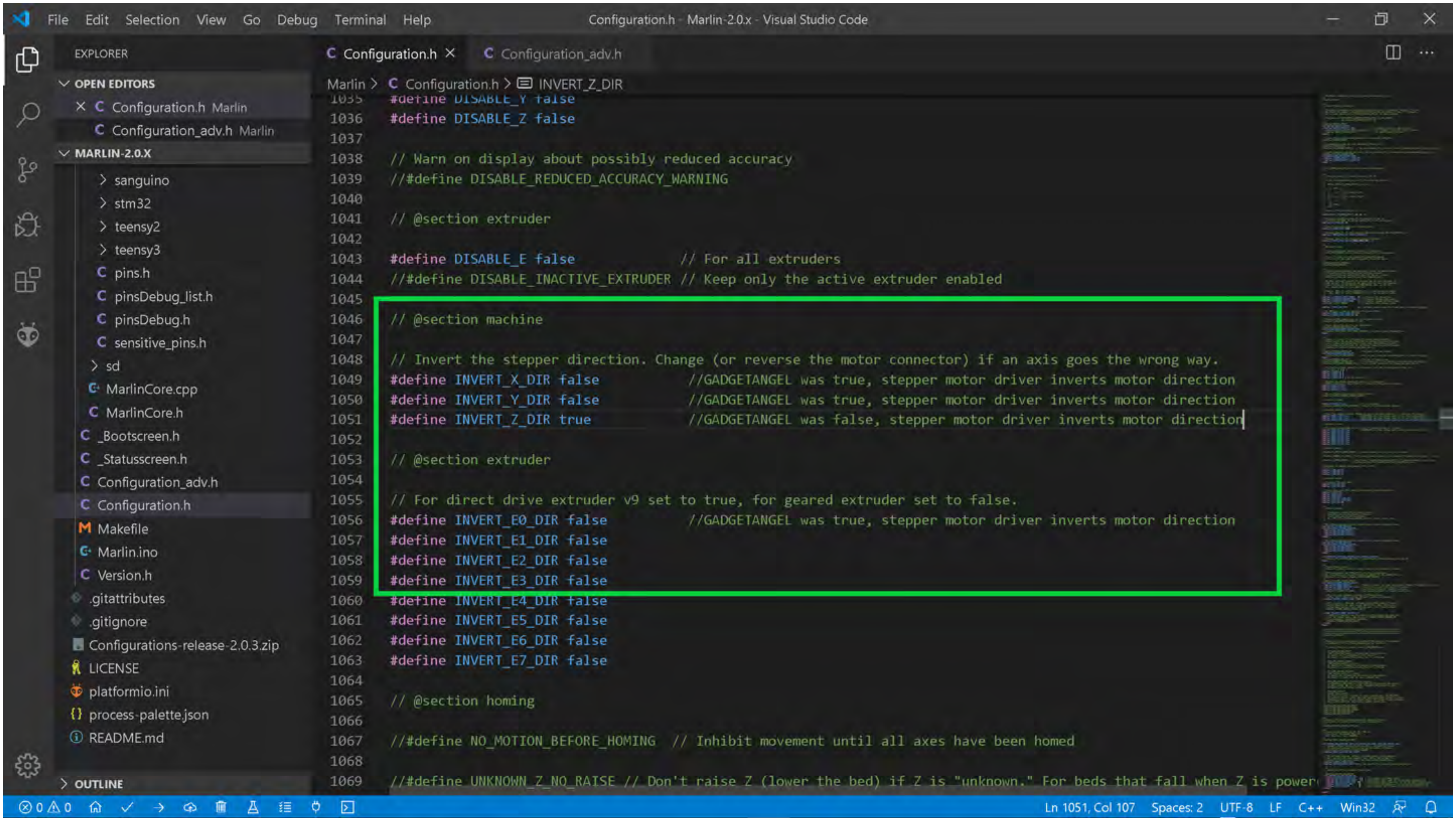
- We are changing from A4988 stepper motor drivers on the Ender 3 to LV8729 stepper motor drivers for each axis and the extruder stepper motor driver, we will be going from 1/16 stepping to 1/32 stepping. So we are doubling our STEPS. Therefore, we must adjust our "DEFAULT_AXIS_STEPS_PER_UNIT" anytime our STEPS are NOT 1/16. So change "DEFAULT_AXIS_STEPS_PER_UNIT" to {160, 160, 800, 186}, as seen in the **GREEN** box below.



- Go to the next page.

The (latest release of) Marlin Setup for BIQU LV8729 Drivers

- Since the A4988 driver is what my Ender 3 used, but, now I want to use LV8729 drivers, I must invert the stepper motor direction because the LV8729 driver will turn the motors in the opposite direction than the A4988 driver's motor direction. So if the axis' setting you will be using the LV8729 driver on was "true" change it to "false", as shown in the **GREEN** box below. If the setting was "false", now set it to "true", as show in the **GREEN** box below

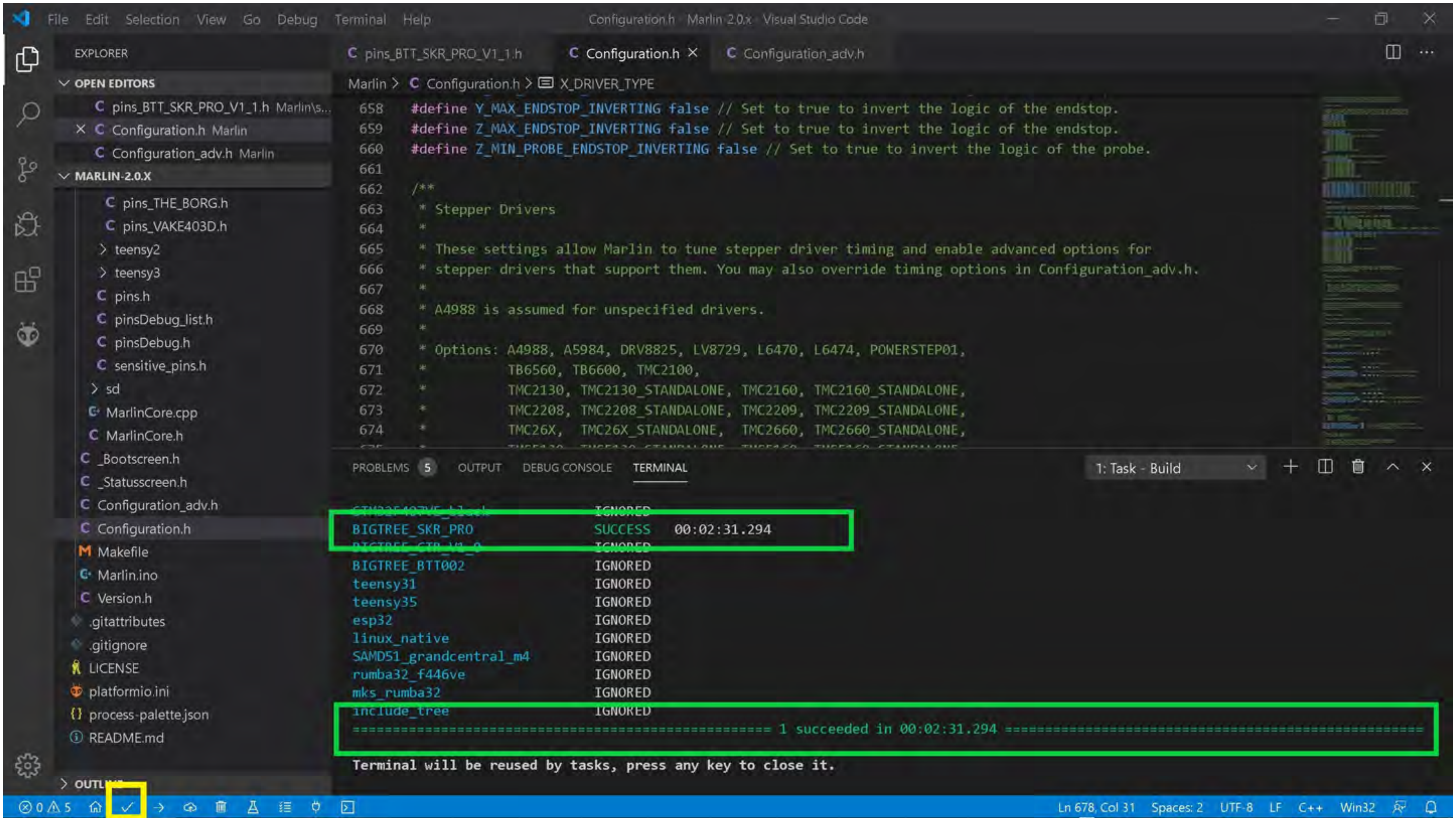


- Go to the next page.

NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

The (latest release of) Marlin Setup for BIQU LV8729 Drivers

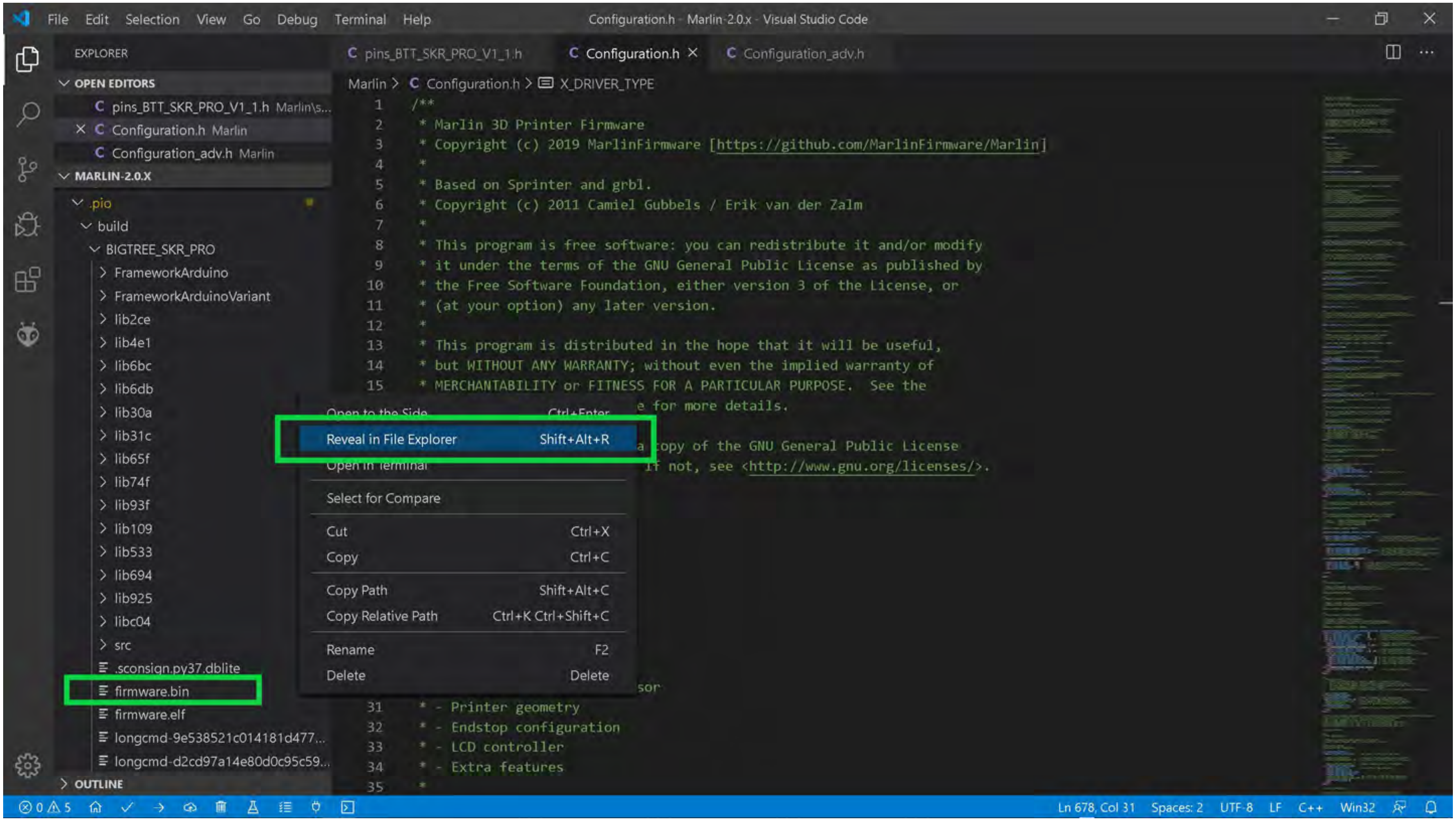
- The end of Marlin setup for BIQU LV8729 drivers. Now, compile your code by clicking on the check mark, as seen in the **YELLOW** box below. Once the firmware has compiled successfully you will see the following messages, as shown in the **GREEN** boxes below.



- For the SKR PRO V1.1 board you must save the "firmware.bin" file to the micro SD card then place the micro SD card into the micro SD card reader of the SKR PRO V1.1 board and turn on the power to your printer for the NEW firmware to be automatically applied to the SKR PRO MCU. To locate the "firmware.bin" file, and learn how to transfer it to your micro SD card go to the next page.

The (latest release of) Marlin Setup for BIQU LV8729 Drivers

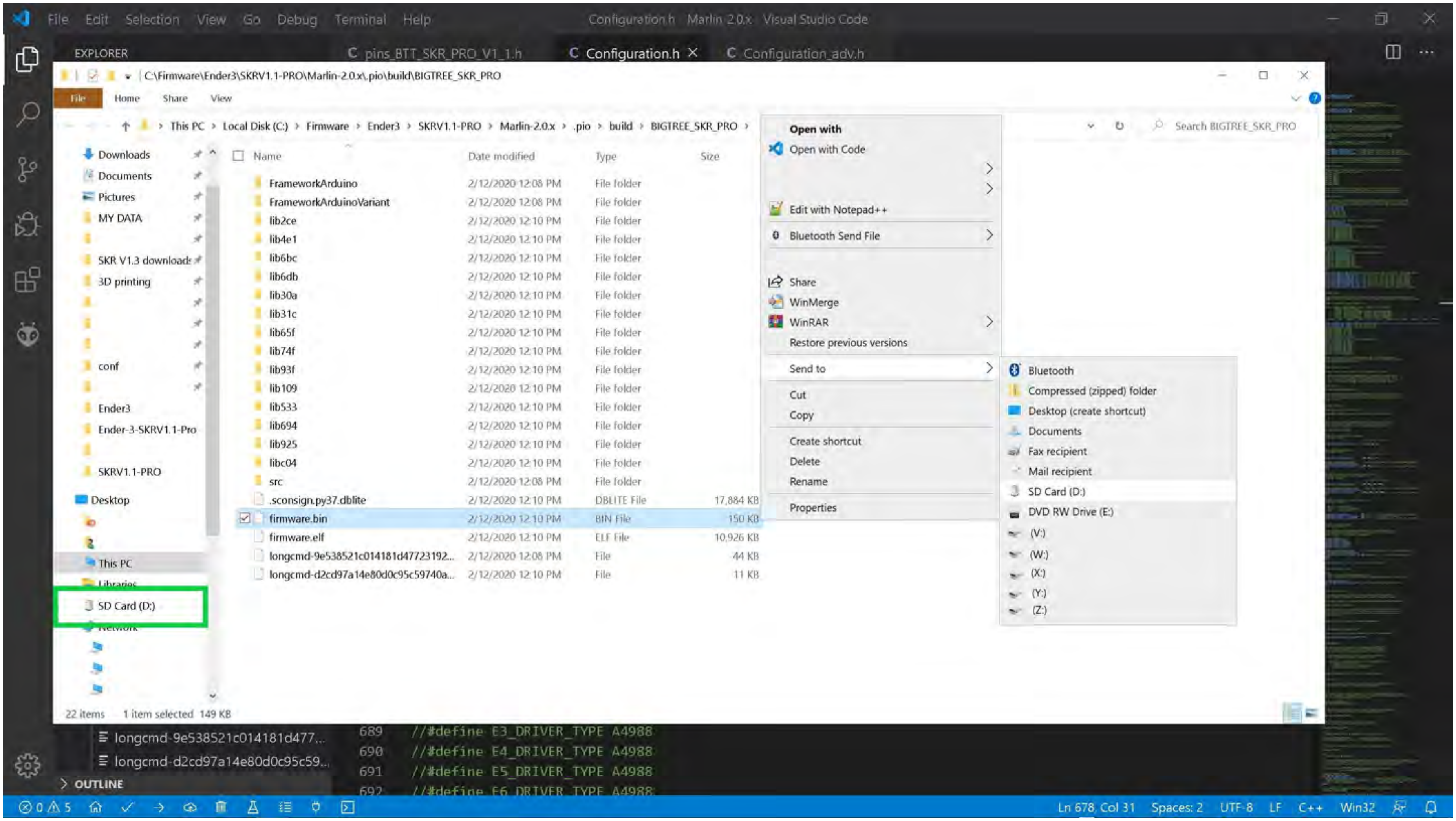
- The "firmware.bin" file can be found by using the left side of the VS code window and going to folder Marlin-2.0.X/.pio/build/ BIGTREE_SKR_PRO/ and right clicking on the "firmware.bin" file name to bring up the content menu. Select "Reveal in File Explorer" to have your Window's machine open a file explorer window.



- Go to the next page.

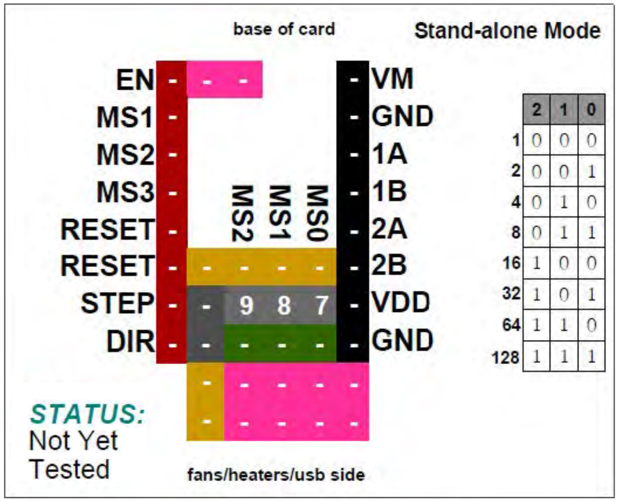
The (latest release of) Marlin Setup for BIQU LV8729 Drivers

- Ensure your micro SD card, which came with your SKR PRO V1.1 board, with your SKR PRO V1.1 board is in a micro-to-SD-card adapter and plugged into your Window's SD card reader.
    - From the file explorer window right click on the "firmware.bin" file name and select "Send to" from the content menu. For my computer "D:" is my SD card reader. For your computer it may be a different drive letter. The "firmware.bin" file will be written to the SD card and renamed to "FIRMWARE.CUR".



- Right click on the SD card reader, as seen in the **GREEN** box above, and select "Eject" from the context menu. Take the micro SD card out of the adapter and place it into the micro SD card reader of the SKR PRO V1.1 board. Turn on the power to the printer (i.e. the SKR PRO V1.1 board) and the "FIRMWARE.CUR" file will be uploaded to the board. After the upload the firmware file will be renamed to "firmware.bin" and stored on the micro SD card.

FYSETC LV8729



Note: See the next page for information about location of the current sense resistors and how to set V_{ref} on the stepper motor driver board.

Driver Chip	MS2	MS1	MS0	Microstep Resolution	Excitation Mode
FYSETC LV8729	Low	Low	Low	Full Step	2 Phase
Maximum 128 Subdivision	Low	Low	High	1/2 Step	1-2 Phase
36V DC	Low	High	Low	1/4 Step	W1-2 Phase
1.5A (peak)	Low	High	High	1/8 Step	2W1-2 Phase
	High	Low	Low	1/16 Step	4W1-2 Phase
	High	Low	High	1/32 Step	8W1-2 Phase
	High	High	Low	1/64 Step	16W1-2 Phase
	High	High	High	1/128 Step	32W1-2 Phase

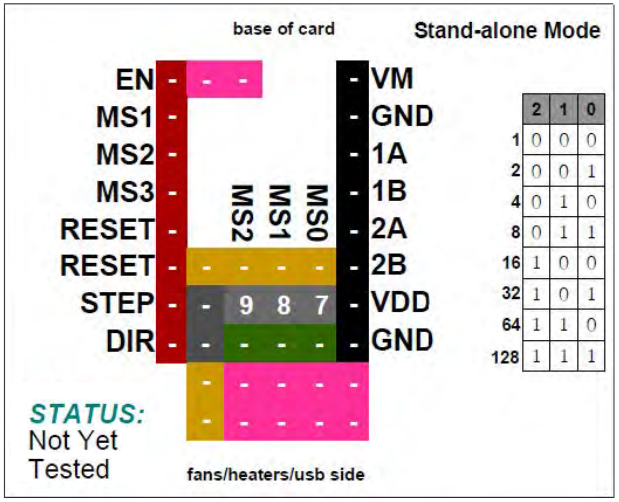
Driving Current Calculation Formula

R_S(Typical Sense Resistor) $= 0.22\Omega$

$$I_{MAX} = \frac{V_{ref}}{5 * R_S}$$

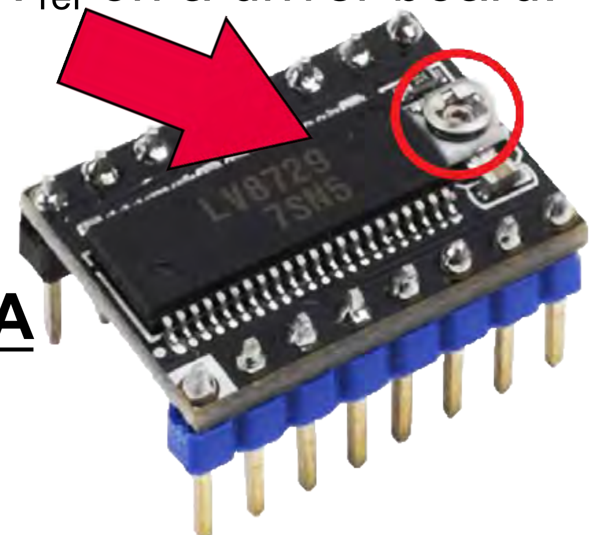
$$V_{ref} = 5 * I_{MAX} * R_S$$

FYSETC LV8729



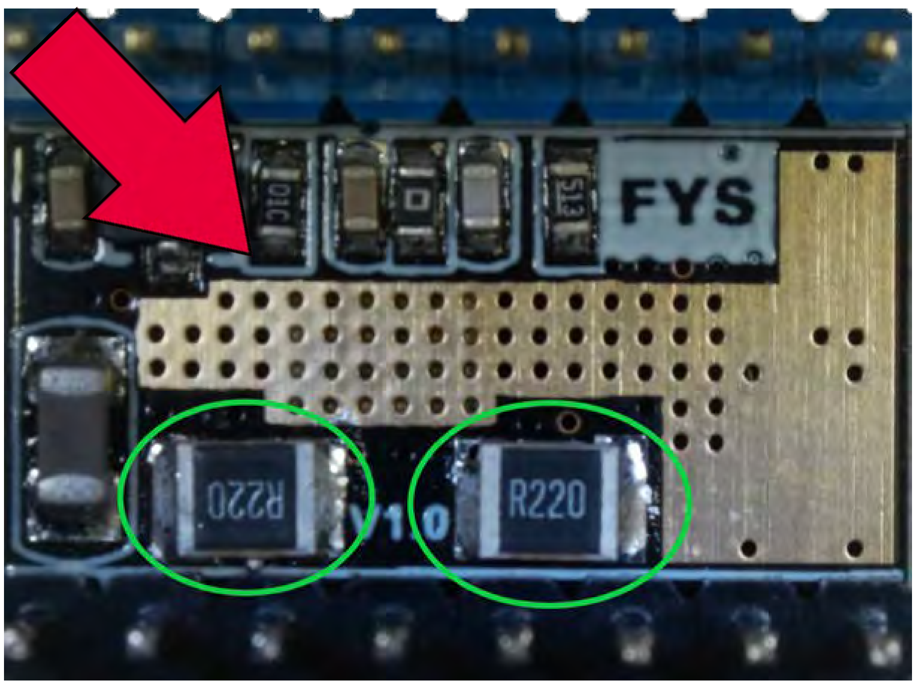
Note: Use 90% of the calculated V_{ref} when tuning the stepper driver board.

NOTE: Use the potentiometer (POT) on the top of the board to adjust your V_{ref}. See **Appendix A** for instructions on how to set the V_{ref} on a driver board.



Note: Check your current sense resistors (R_s) values on the driver board, as shown in GREEN

Note: See this video about current sense resistors (R_s) and their possible locations: **https://youtu.be/8wk1elugv5A**



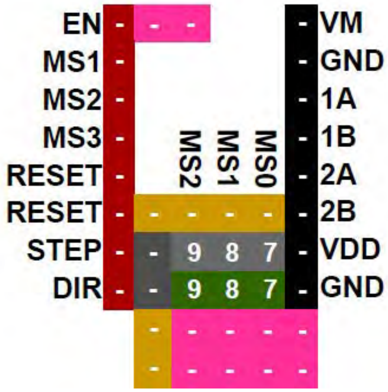
R_s = R050 is 0.05 Ohms;
R_s = R068 is 0.068 Ohms
R_s = R100 is 0.1 Ohms;
R_s = R150 is 0.15 Ohms
R_s = R200 is 0.2 Ohms;
R_s = R220 is 0.22 Ohms

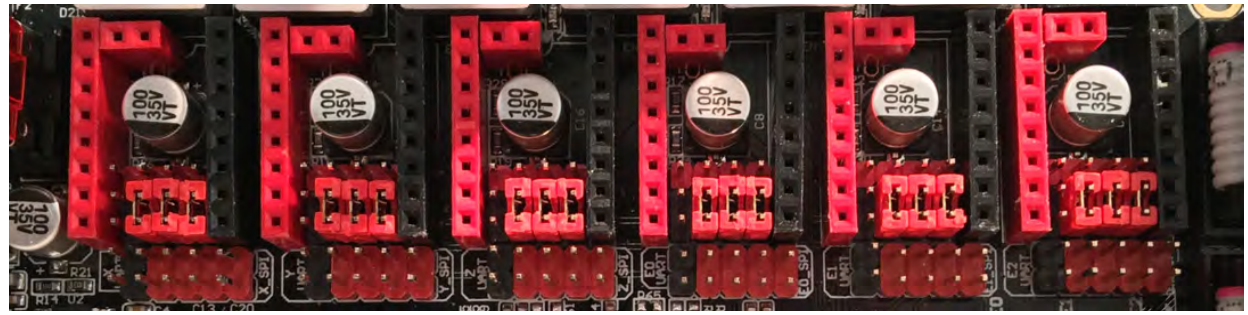
NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

FYSETC LV8729

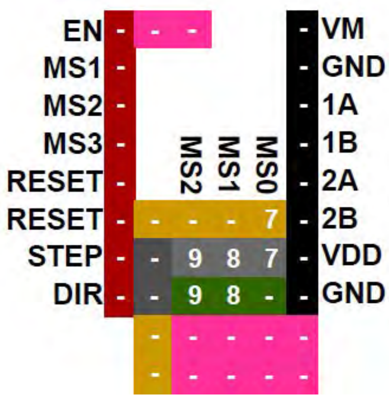
Stand-alone Mode

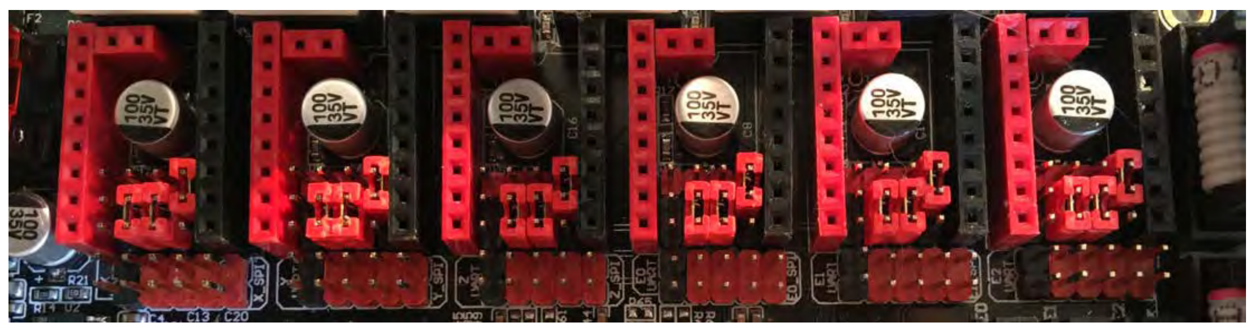
STEP





1 / 2



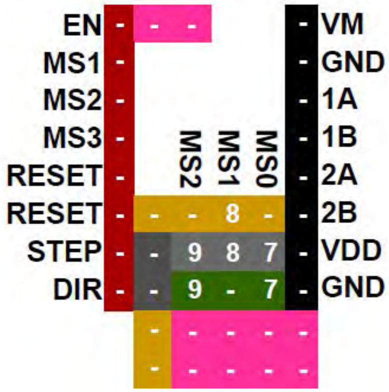


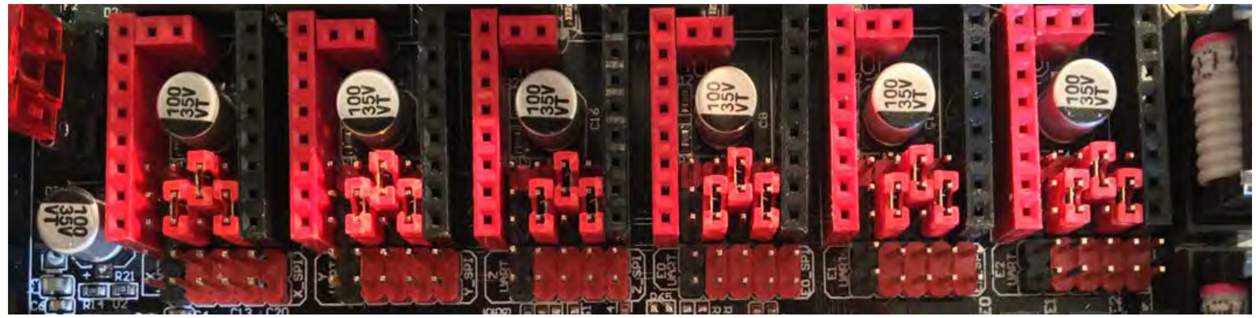
NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

FYSETC LV8729

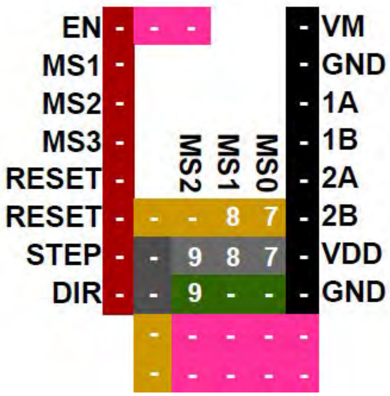
Stand-alone Mode

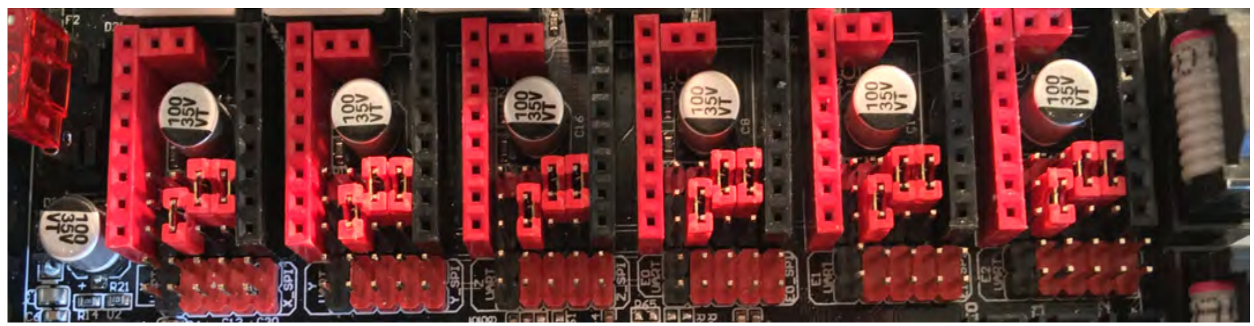
1 / 4



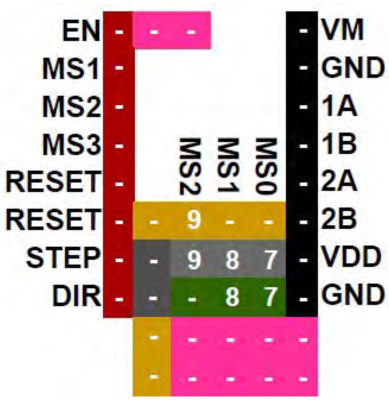


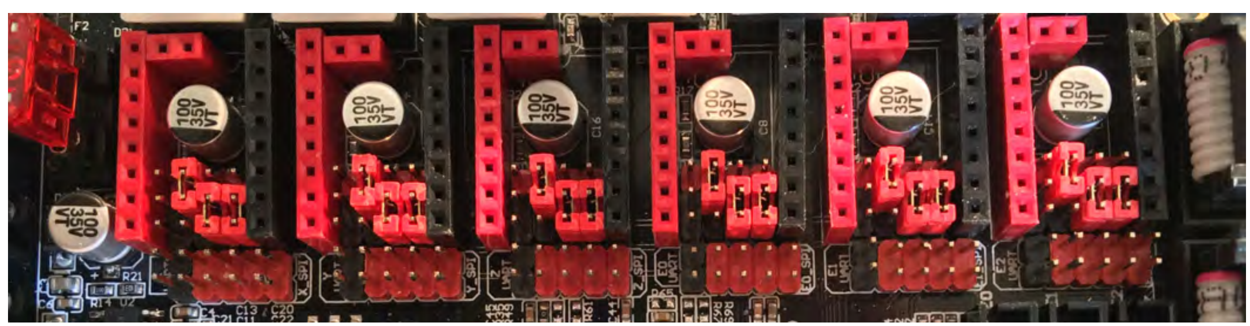
1 / 8





1 / 16



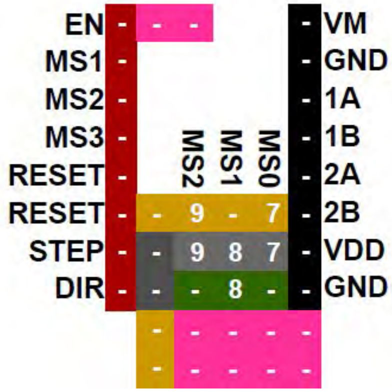


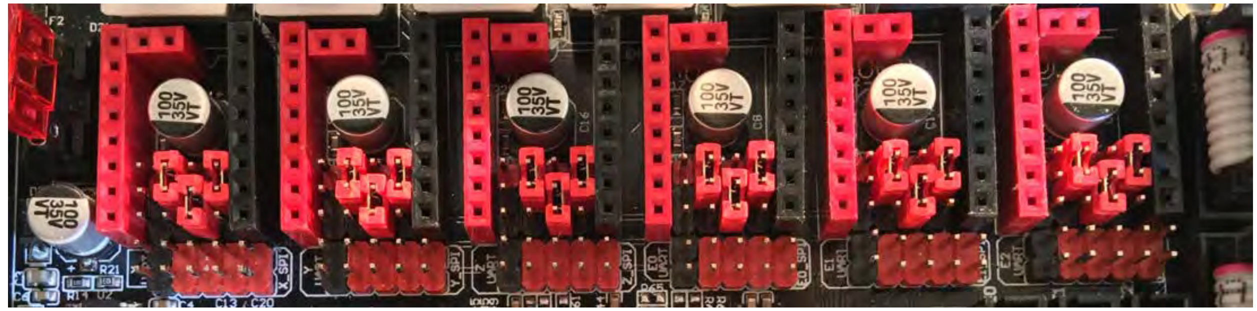
NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

FYSETC LV8729

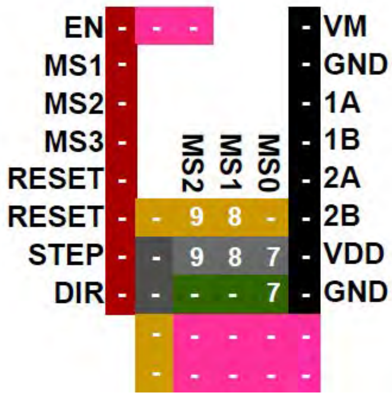
Stand-alone Mode

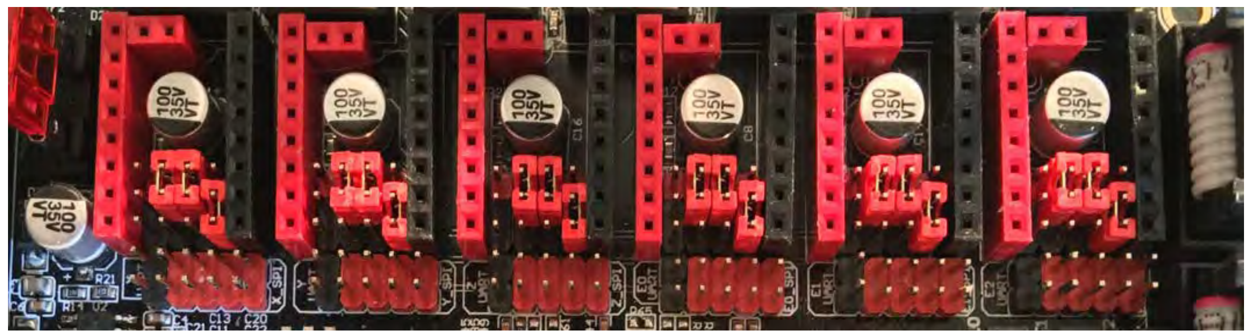
1 / 32



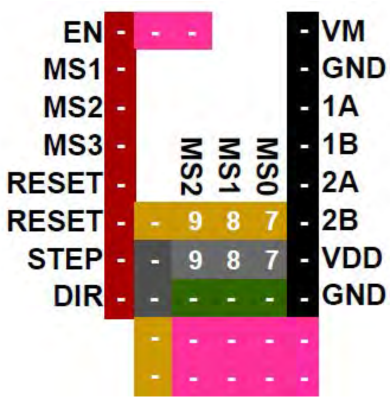


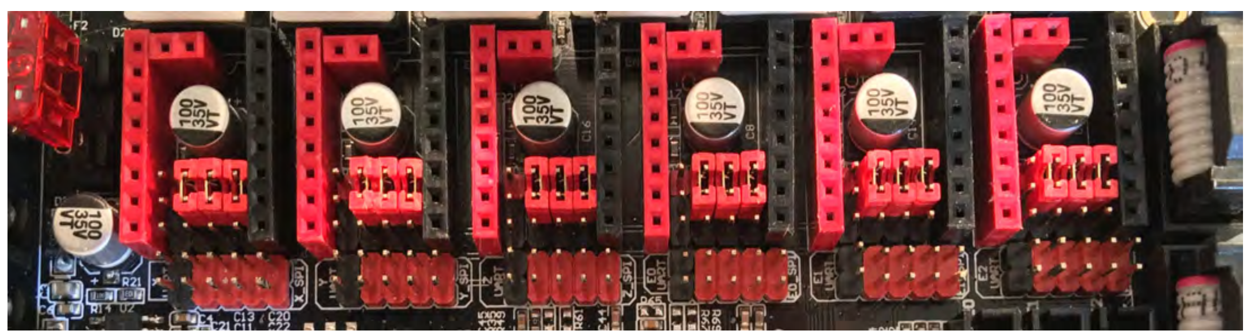
1 / 64





1 / 128



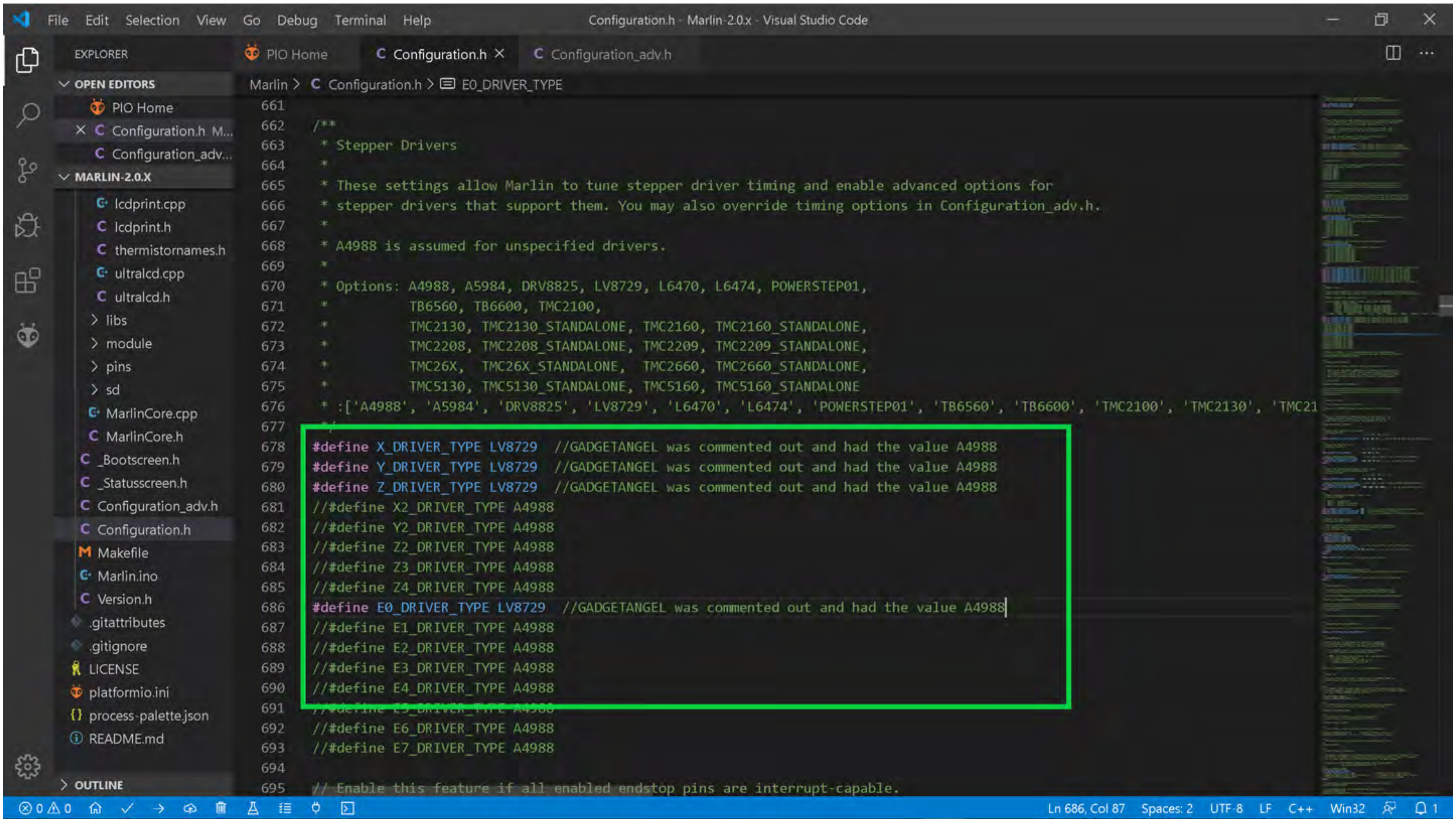


NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

The (latest release of) Marlin Setup for FYSETC LV8729 Drivers

NOTE: **Go to Appendix C**, and then come back here for the changes to Marlin for FYSETC LV8729 stepper motor drivers.

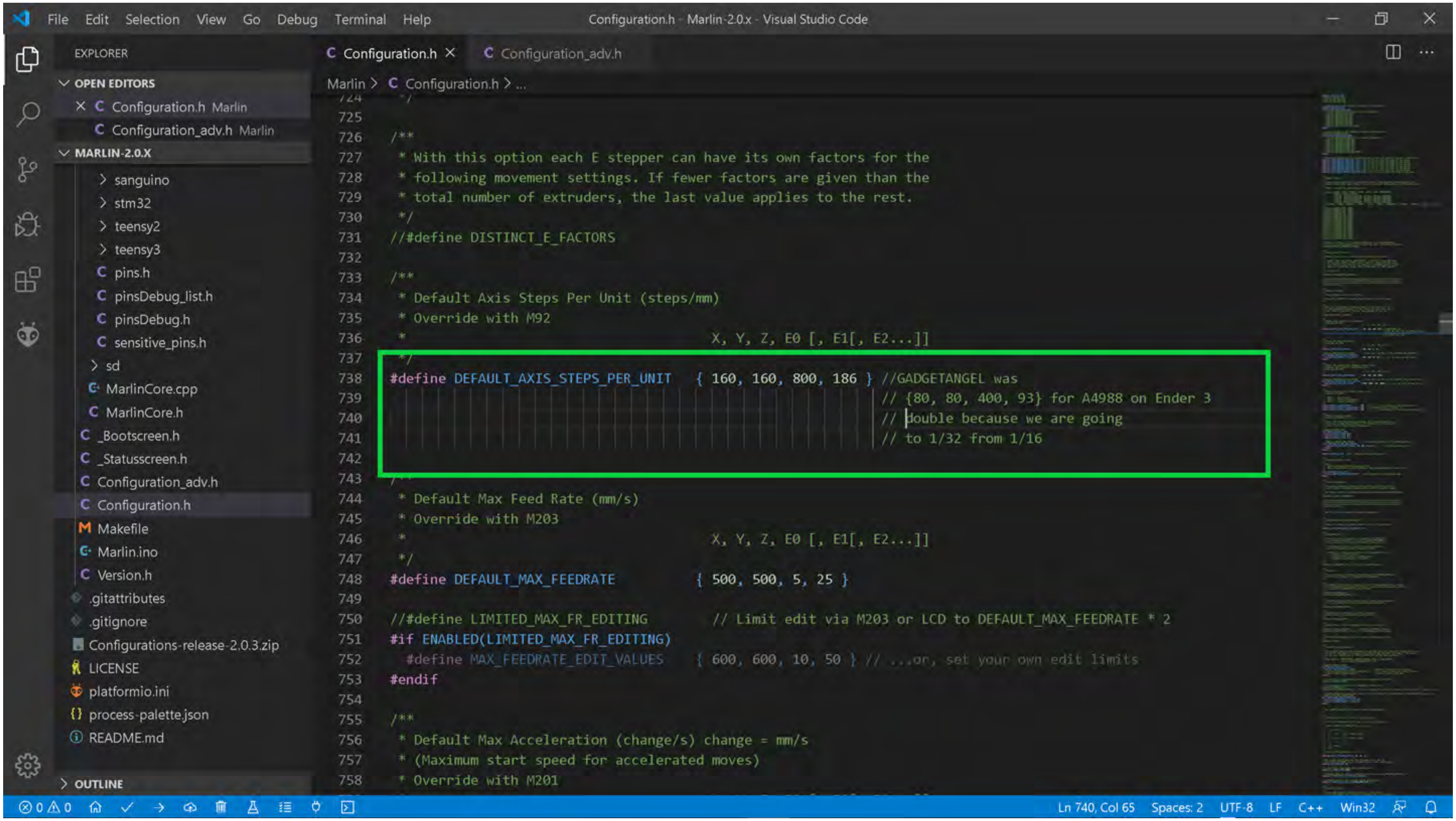
- Change the stepper motor drivers so that Marlin knows you are using FYSETC LV8729 drivers. Change one line for each axis and one for each extruder you will be using. See the picture below for an example of how to use FYSETC LV8729 drivers. When two "//" appear at the beginning of a line that means that line is commented out. To un-comment a line just remove the leading two forward slashes ("//").



- Go to the next page.

The (latest release of) Marlin Setup for FYSETC LV8729 Drivers

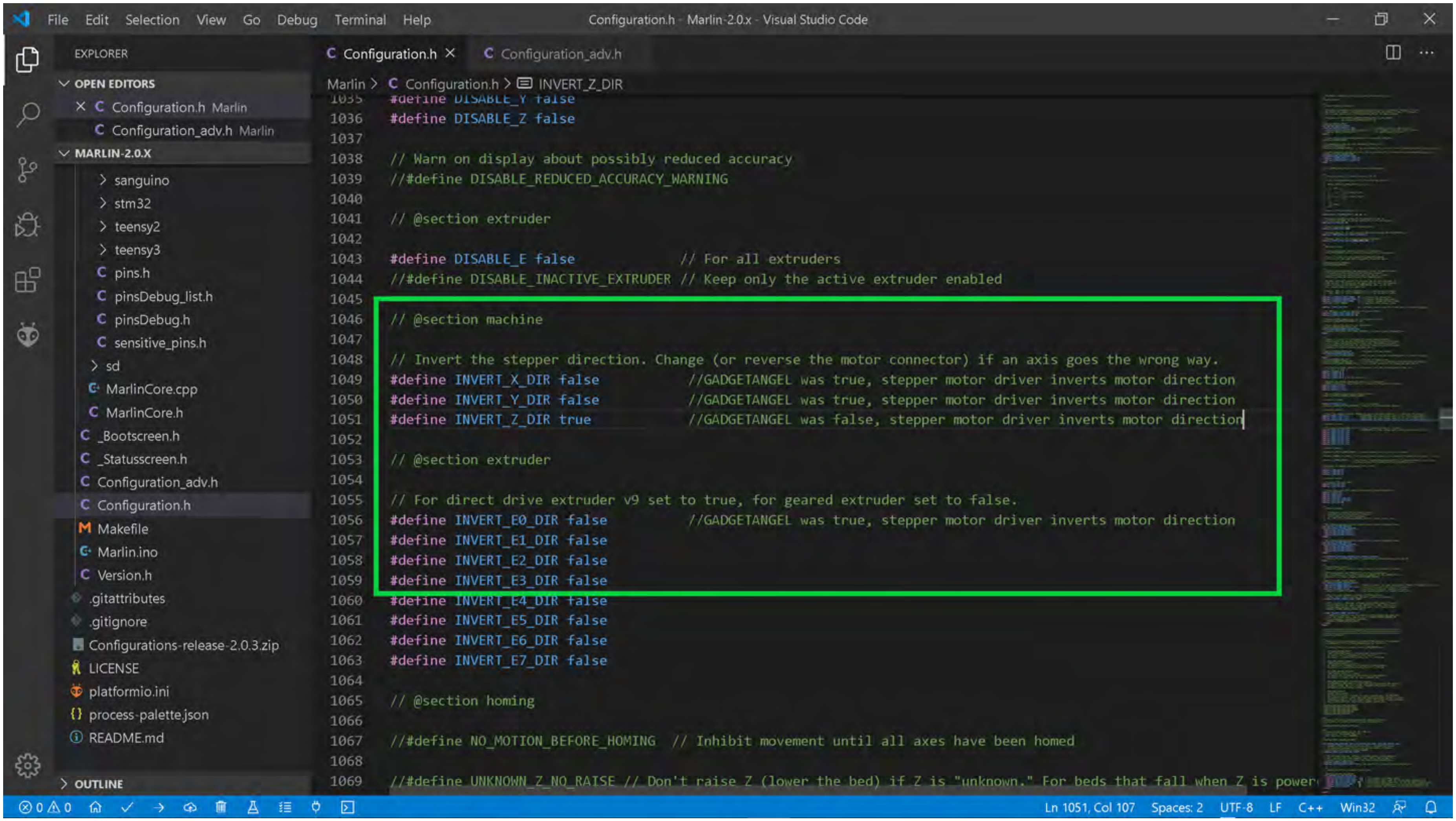
- We are changing from A4988 stepper motor drivers on the Ender 3 to LV8729 stepper motor drivers for each axis and the extruder stepper motor driver, we will be going from 1/16 stepping to 1/32 stepping. So we are doubling our STEPS. Therefore, we must adjust our "DEFAULT_AXIS_STEPS_PER_UNIT" anytime our STEPS are NOT 1/16. So change "DEFAULT_AXIS_STEPS_PER_UNIT" to {160, 160, 800, 186}, as seen in the **GREEN** box below.



- Go to the next page.

The (latest release of) Marlin Setup for FYSETC LV8729 Drivers

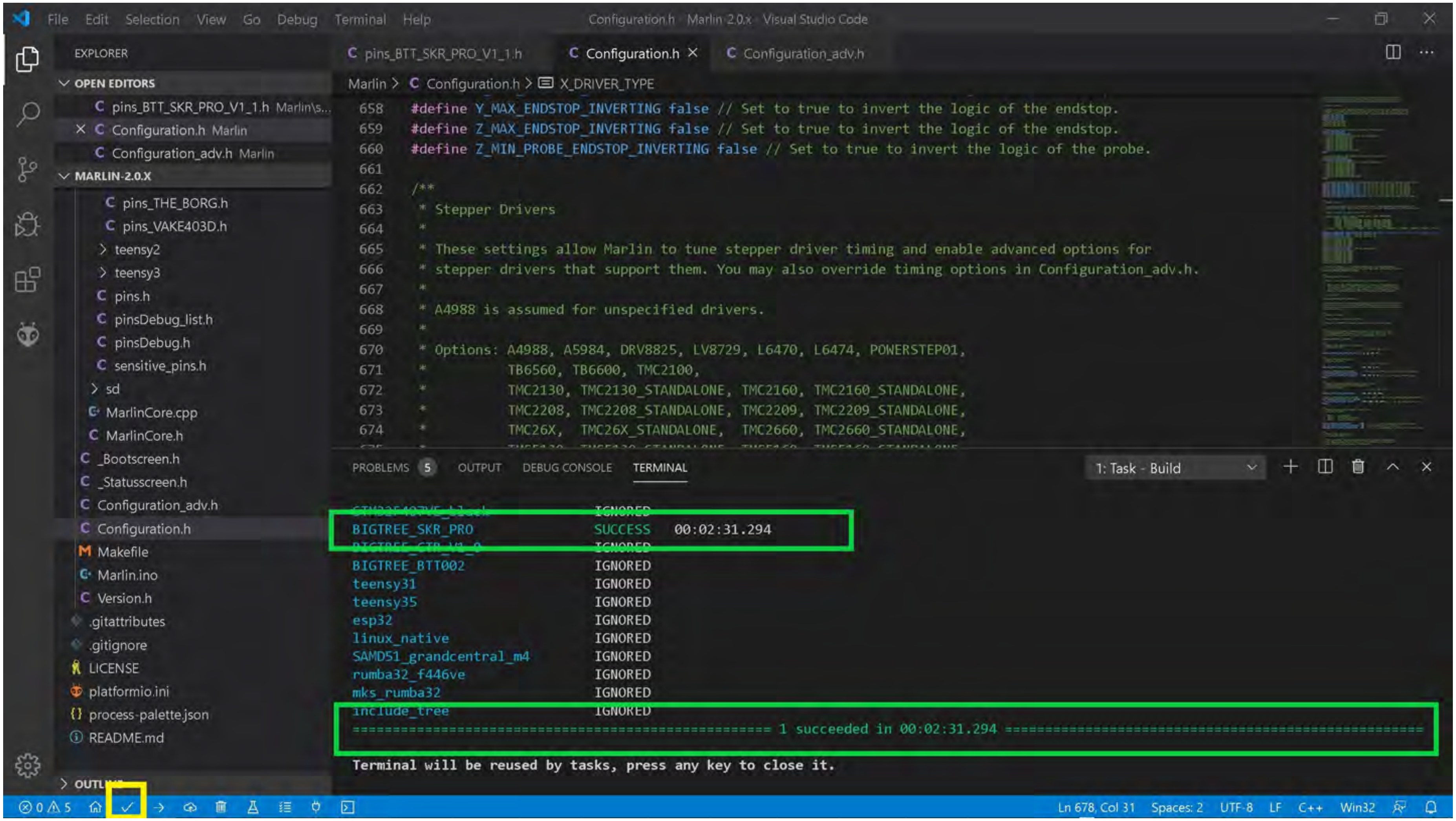
- Since the A4988 driver is what my Ender 3 used, but, now I want to use LV8729 drivers, I must invert the stepper motor direction because the LV8729 driver will turn the motors in the opposite direction than the A4988 driver's motor direction. So if the axis' setting you will be using the LV8729 driver on was "true" change it to "false", as shown in the **GREEN** box below. If the setting was "false", now set it to "true", as show in the **GREEN** box below



- Go to the next page.

The (latest release of) Marlin Setup for FYSETC LV8729 Drivers

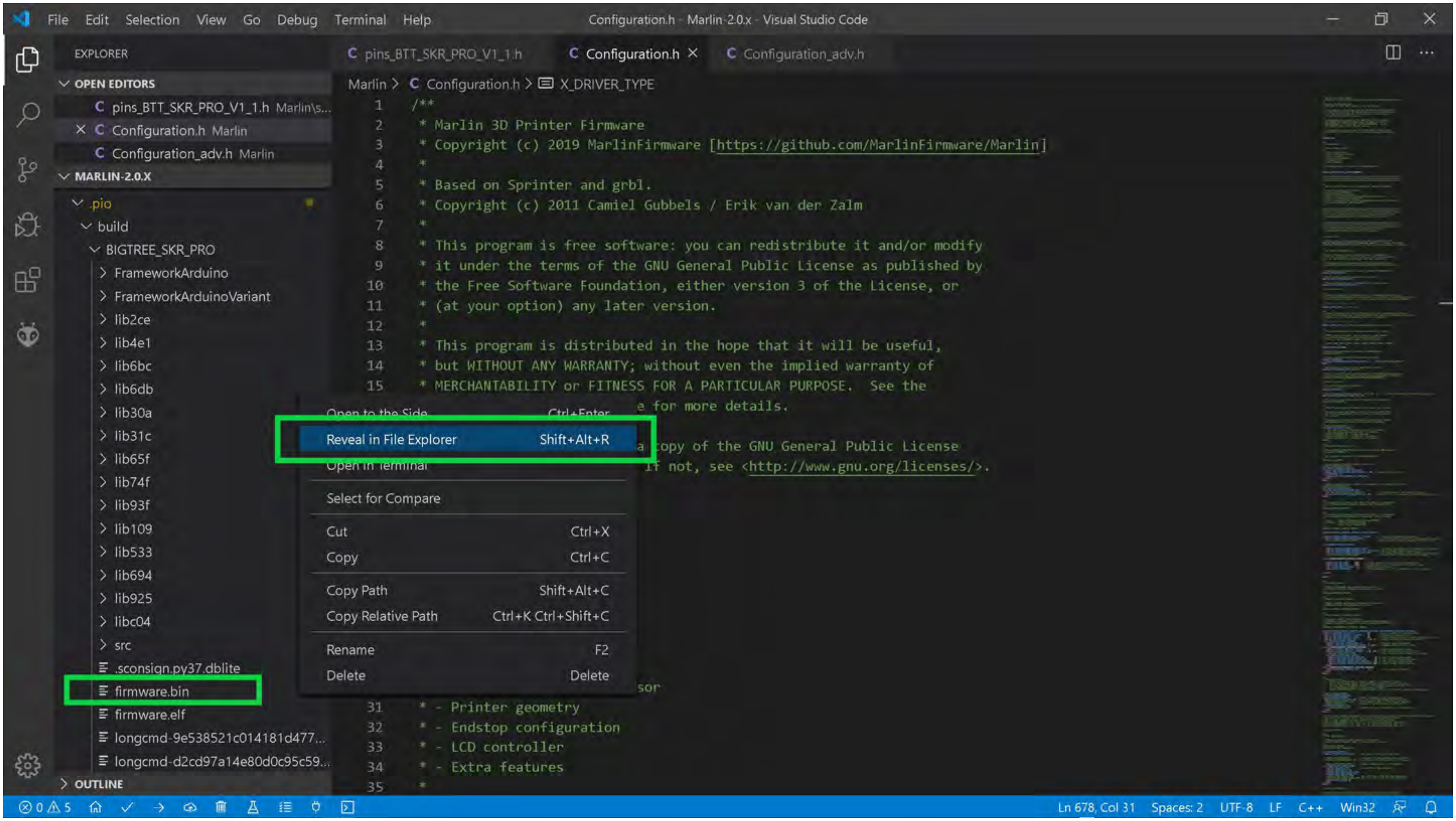
- The end of Marlin setup for FYSETC LV8729 drivers. Now, compile your code by clicking on the check mark, as seen in the **YELLOW** box below. Once the firmware has compiled successfully you will see the following messages, as shown in the **GREEN** boxes below.



- For the SKR PRO V1.1 board you must save the "firmware.bin" file to the micro SD card then place the micro SD card into the micro SD card reader of the SKR PRO V1.1 board and turn on the power to your printer for the NEW firmware to be automatically applied to the SKR PRO MCU. To locate the "firmware.bin" file, and learn how to transfer it to your micro SD card go to the next page.

The (latest release of) Marlin Setup for FYSETC LV8729 Drivers

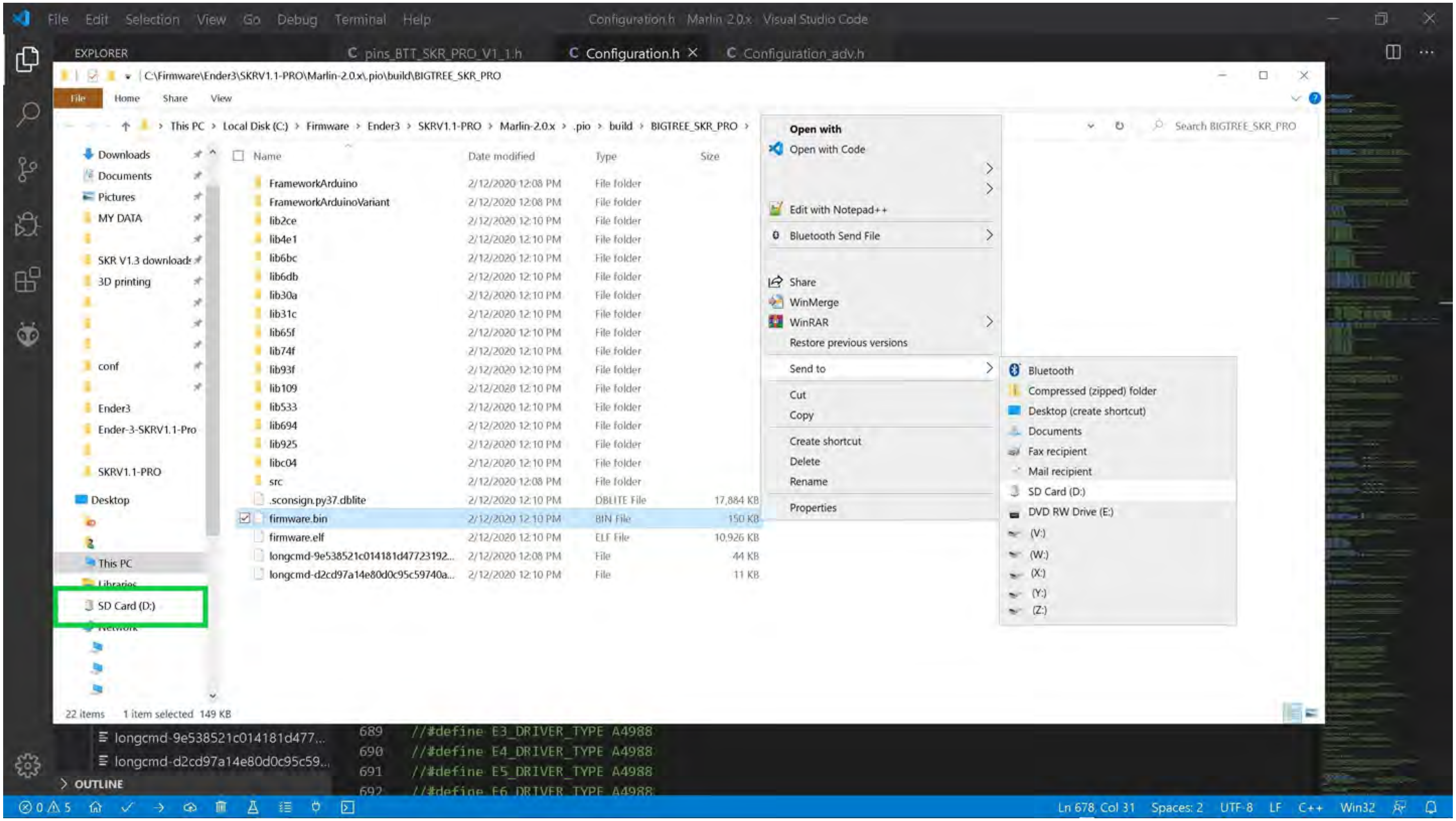
- The "firmware.bin" file can be found by using the left side of the VS code window and going to folder Marlin-2.0.X/.pio/build/ BIGTREE_SKR_PRO/ and right clicking on the "firmware.bin" file name to bring up the content menu. Select "Reveal in File Explorer" to have your Window's machine open a file explorer window.



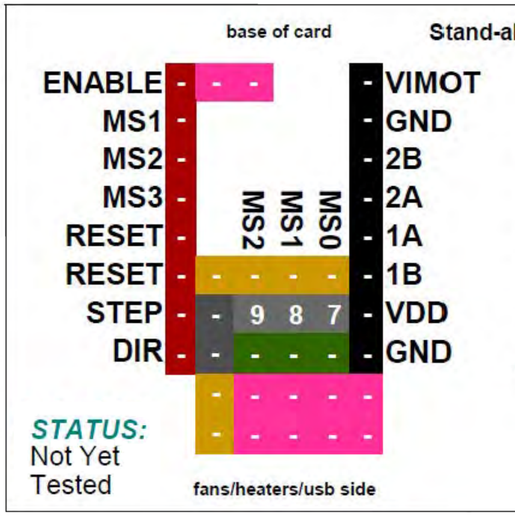
- Go to the next page.

The (latest release of) Marlin Setup for FYSETC LV8729 Drivers

- Ensure your micro SD card, which came with your SKR PRO V1.1 board, with your SKR PRO V1.1 board is in a micro-to-SD-card adapter and plugged into your Window's SD card reader.
    - From the file explorer window right click on the "firmware.bin" file name and select "Send to" from the content menu. For my computer "D:" is my SD card reader. For your computer it may be a different drive letter. The "firmware.bin" file will be written to the SD card and renamed to "FIRMWARE.CUR".



- Right click on the SD card reader, as seen in the **GREEN** box above, and select "Eject" from the context menu. Take the micro SD card out of the adapter and place it into the micro SD card reader of the SKR PRO V1.1 board. Turn on the power to the printer (i.e. the SKR PRO V1.1 board) and the "FIRMWARE.CUR" file will be uploaded to the board. After the upload the firmware file will be renamed to "firmware.bin" and stored on the micro SD card.



	2	1	0
1	0	0	0
2	0	0	1
4	0	1	0
8	0	1	1
16	1	0	0
32	1	0	1
64	1	1	0
128	1	1	1

LERDGE LV8729

Note: See the next page for information about location of the current sense resistors and how to set V_{ref} on the stepper motor driver board.

Driver Chip	MS2	MS1	MS0	Microstep Resolution	Excitation Mode
LV8729 **Maximum 128 Subdivision** **36V DC** **1.5A (peak)**	Low	Low	Low	Full Step	2 Phase
	Low	Low	High	1/2 Step	1-2 Phase
	Low	High	Low	1/4 Step	W1-2 Phase
	Low	High	High	1/8 Step	2W1-2 Phase
	High	Low	Low	1/16 Step	4W1-2 Phase
	High	Low	High	1/32 Step	8W1-2 Phase
	High	High	Low	1/64 Step	16W1-2 Phase
	High	High	High	1/128 Step	32W1-2 Phase

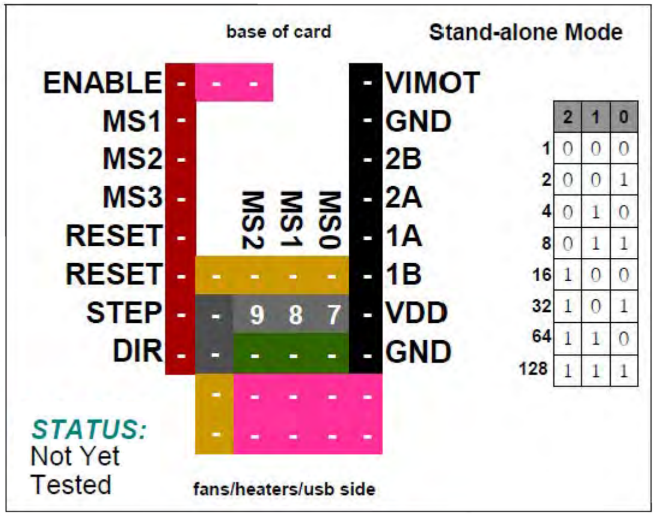
Driving Current Calculation Formula

R_S(Typical Sense Resistor)$=0.22\Omega$

$$I_{MAX} = \frac{V_{ref}}{5 * R_S}$$

$$V_{ref} = 5 * I_{MAX} * R_S$$

NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

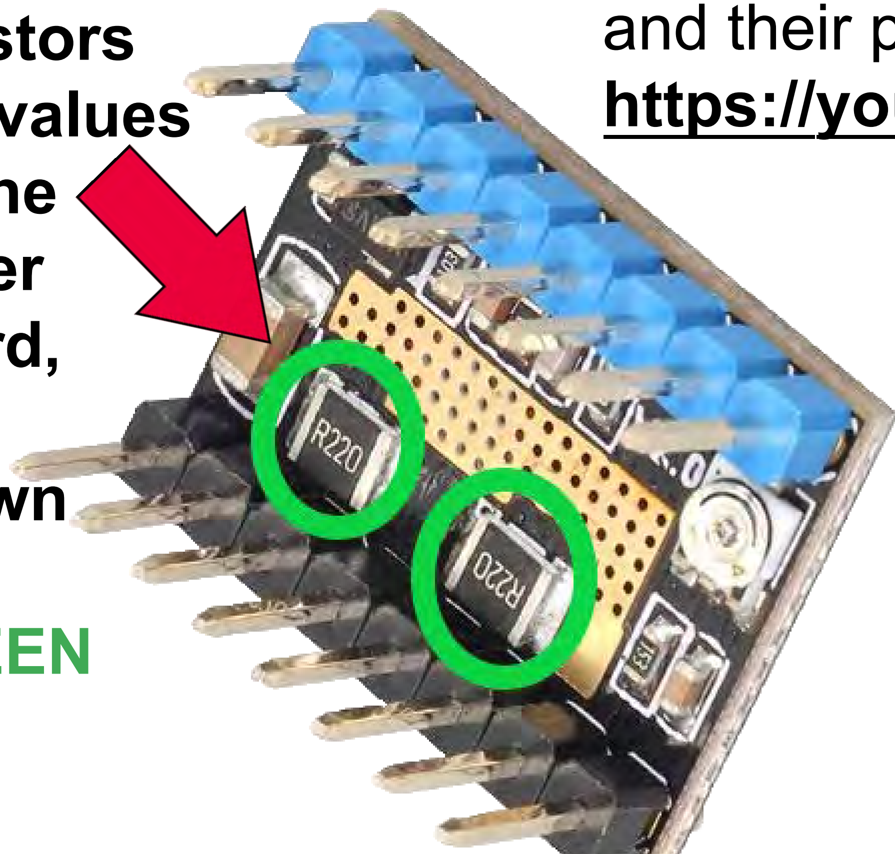


LERDGE LV8729

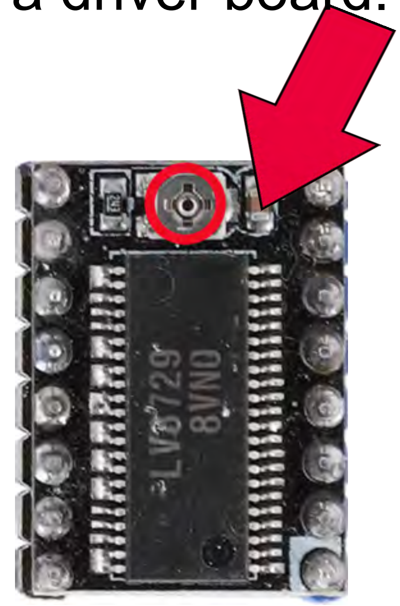
Note: Use 90% of the calculated V_{ref} when tuning the stepper driver board.

NOTE: Use the potentiometer (POT) on the top of the board to adjust your V_{ref}. See **Appendix A** for instructions on how to set the V_{ref} on a driver board.

Note: Check your current sense resistors (R_s) values on the driver board, as shown in GREEN



Note: See this video about current sense resistors (R_s) and their possible locations: **https://youtu.be/8wk1elugv5A**

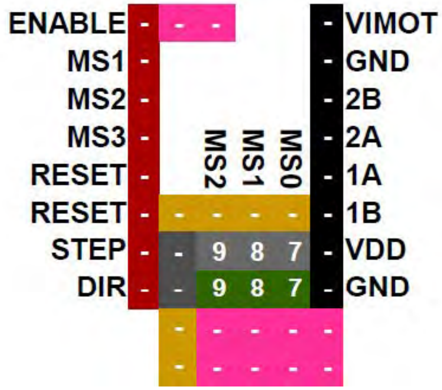


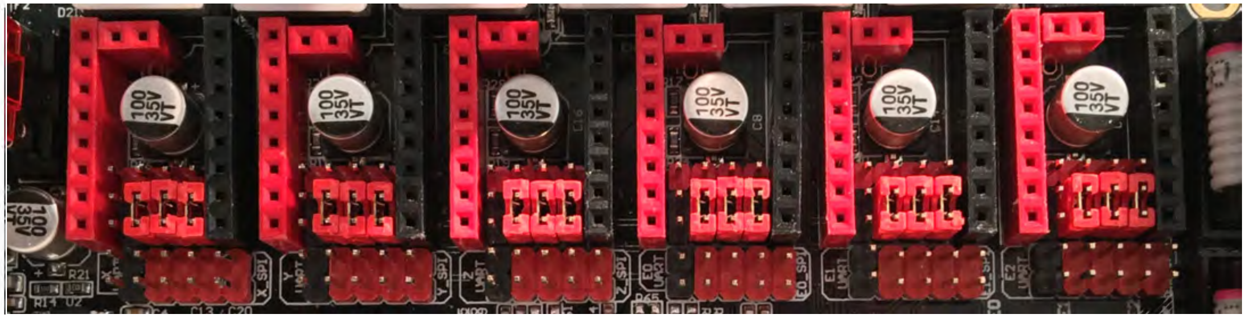
R_s = R050 is 0.05 Ohms
R_s = R068 is 0.068 Ohms
R_s = R100 is 0.1 Ohms
R_s = R150 is 0.15 Ohms
R_s = R200 is 0.2 Ohms
R_s = R220 is 0.22 Ohms

LERDGE LV8729

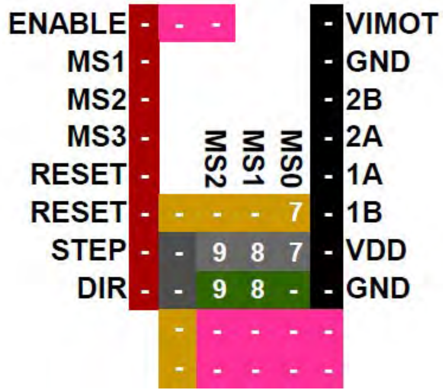
Stand-alone Mode

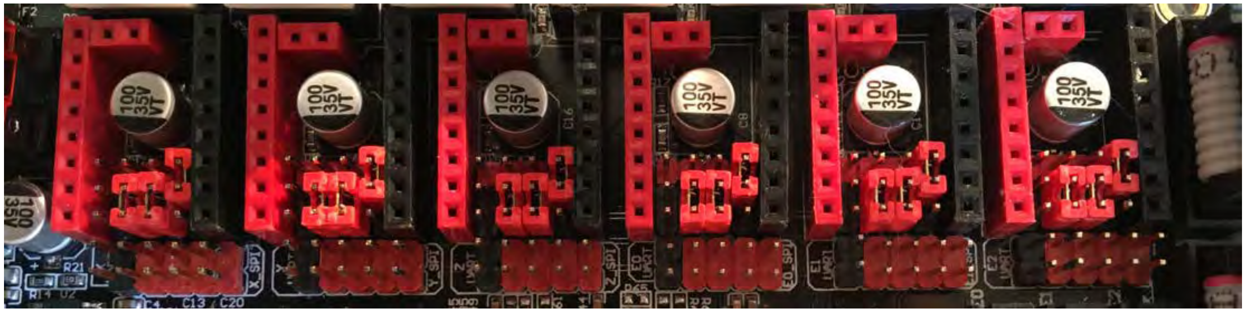
STEP





1 / 2



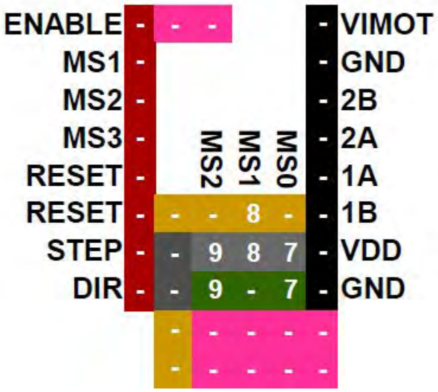


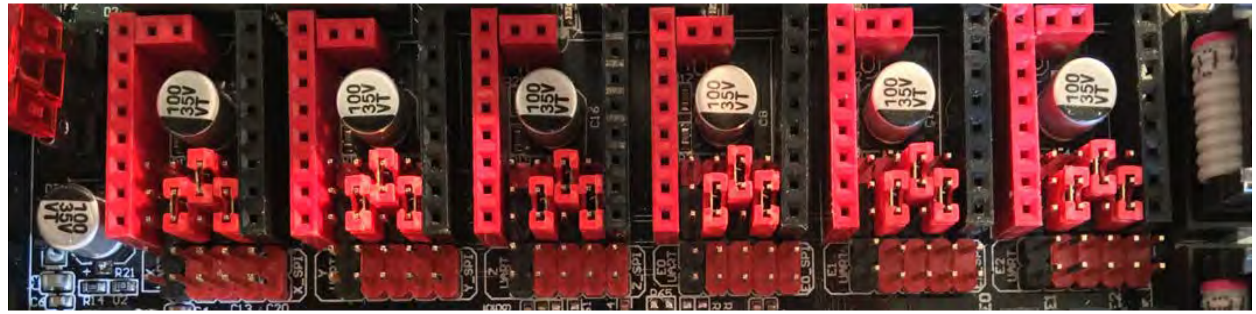
NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

LERDGE LV8729

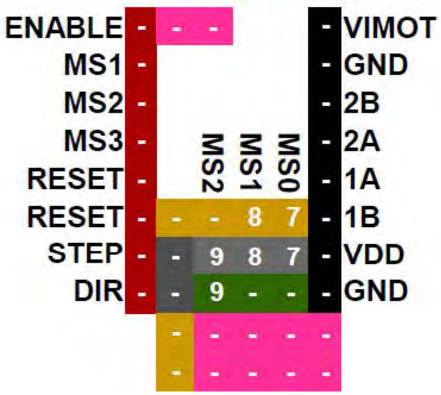
Stand-alone Mode

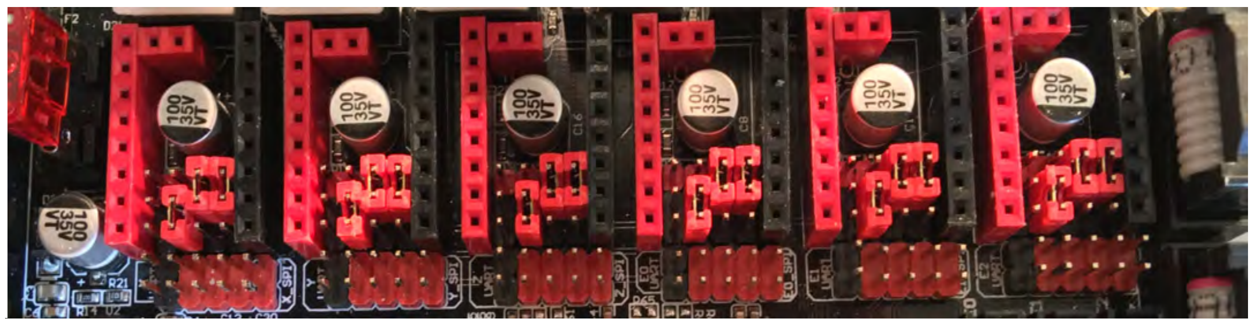
1 / 4



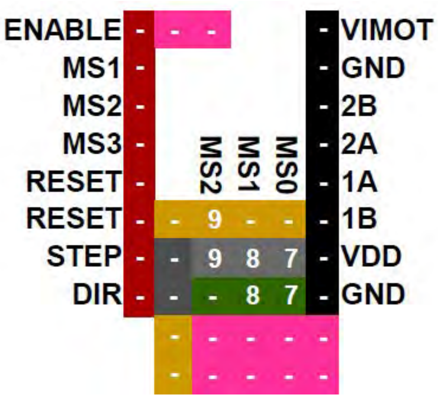


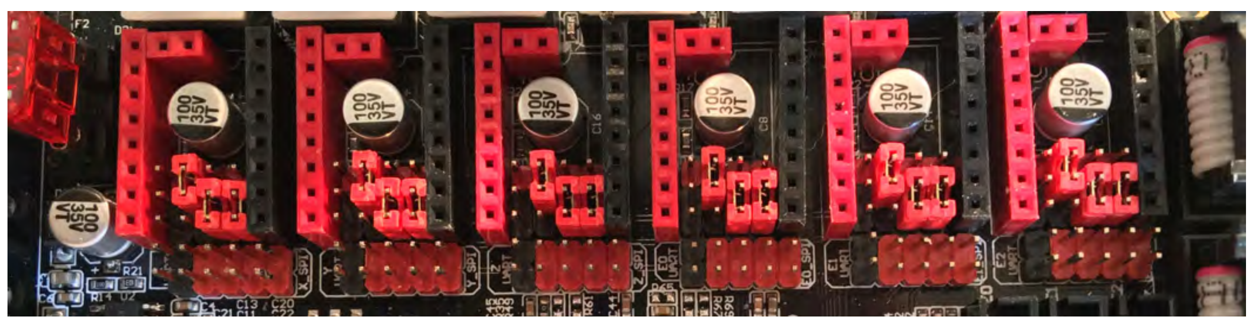
1 / 8





1 / 16



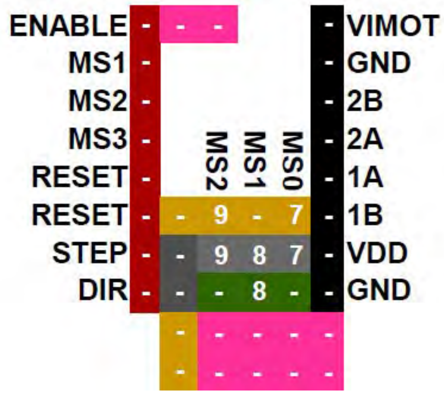


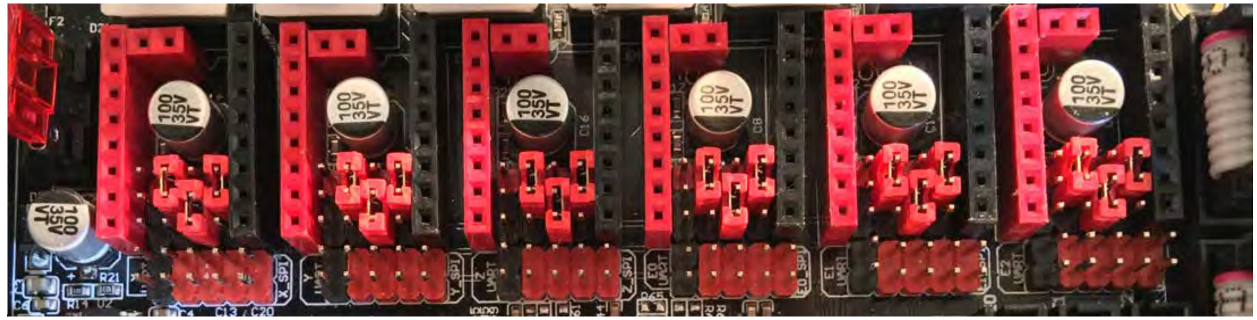
NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

LERDGE LV8729

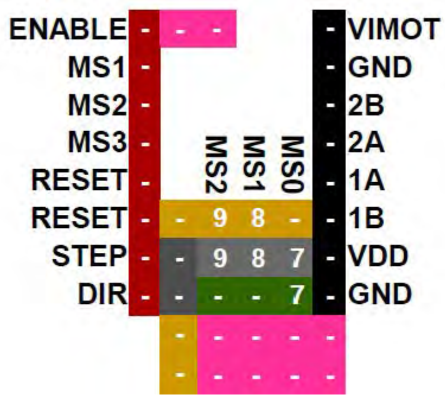
Stand-alone Mode

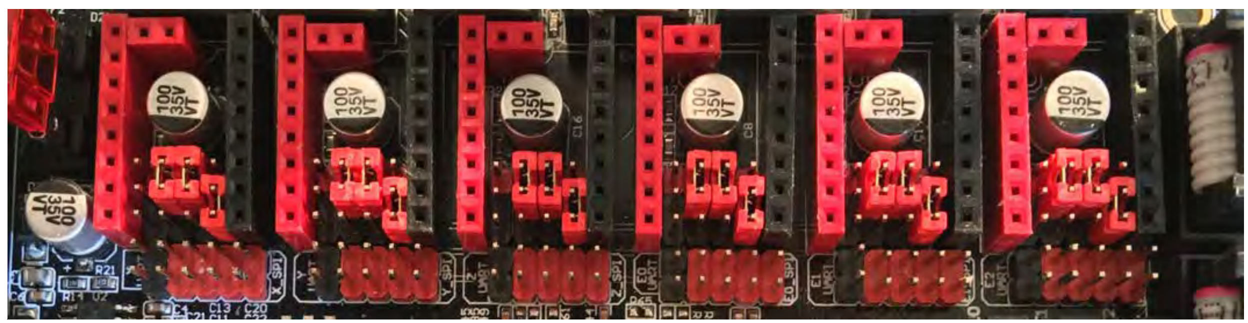
1 / 32



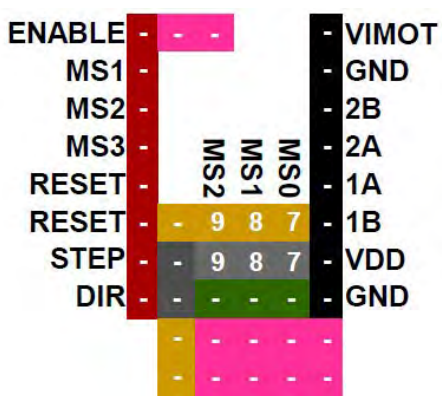


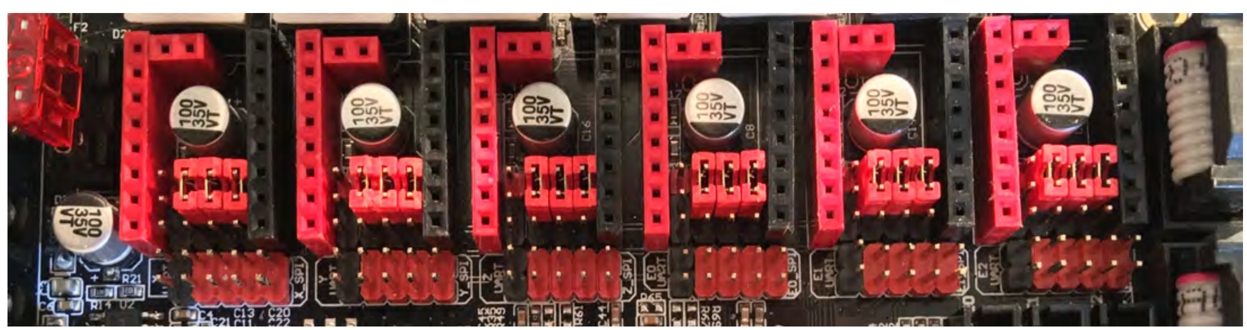
1 / 64





1 / 128



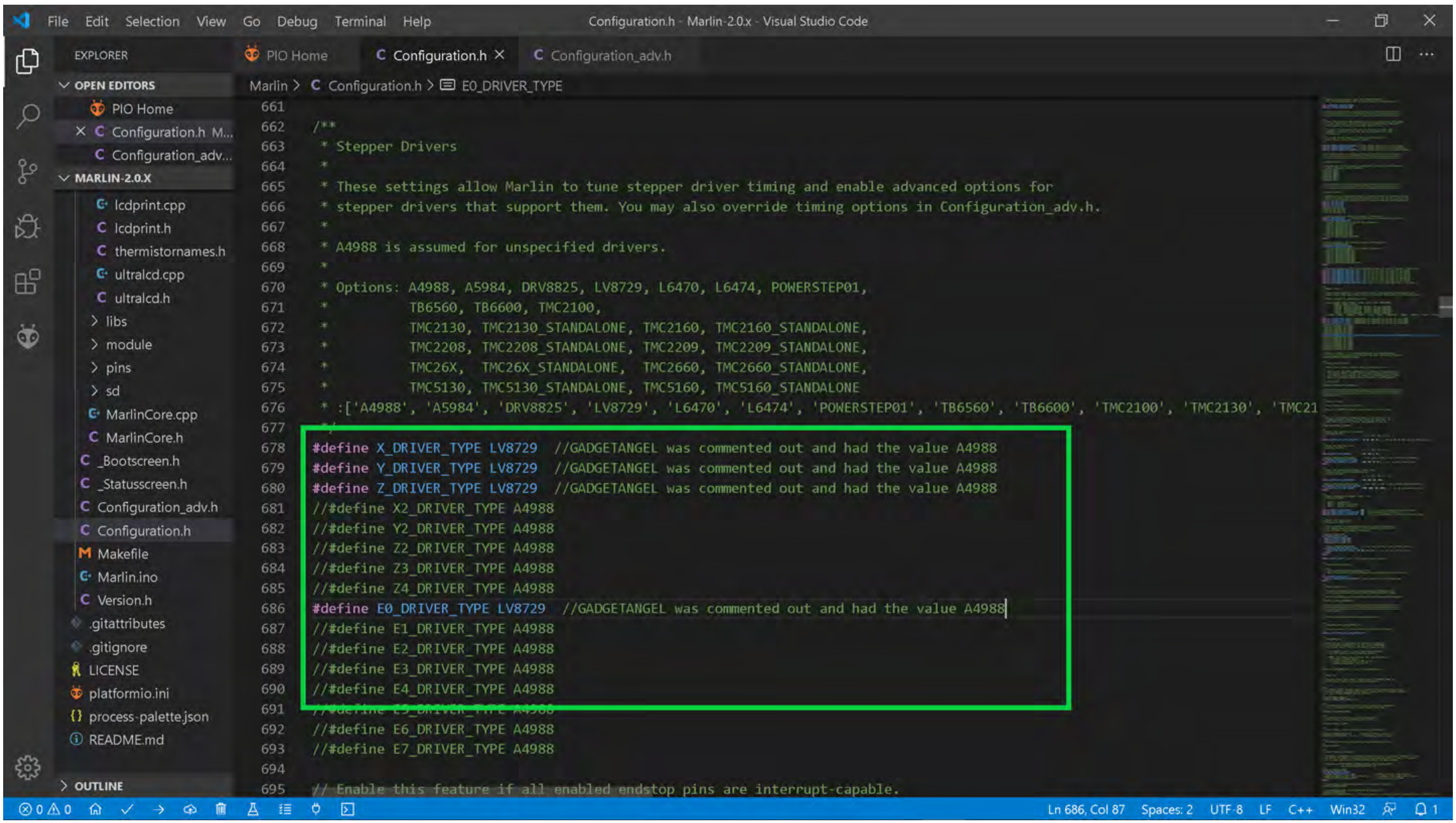


NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

The (latest release of) Marlin Setup for LERDGE LV8729 Drivers

NOTE: **Go to Appendix C**, and then come back here for the changes to Marlin for Lerdge LV8729 stepper motor drivers.

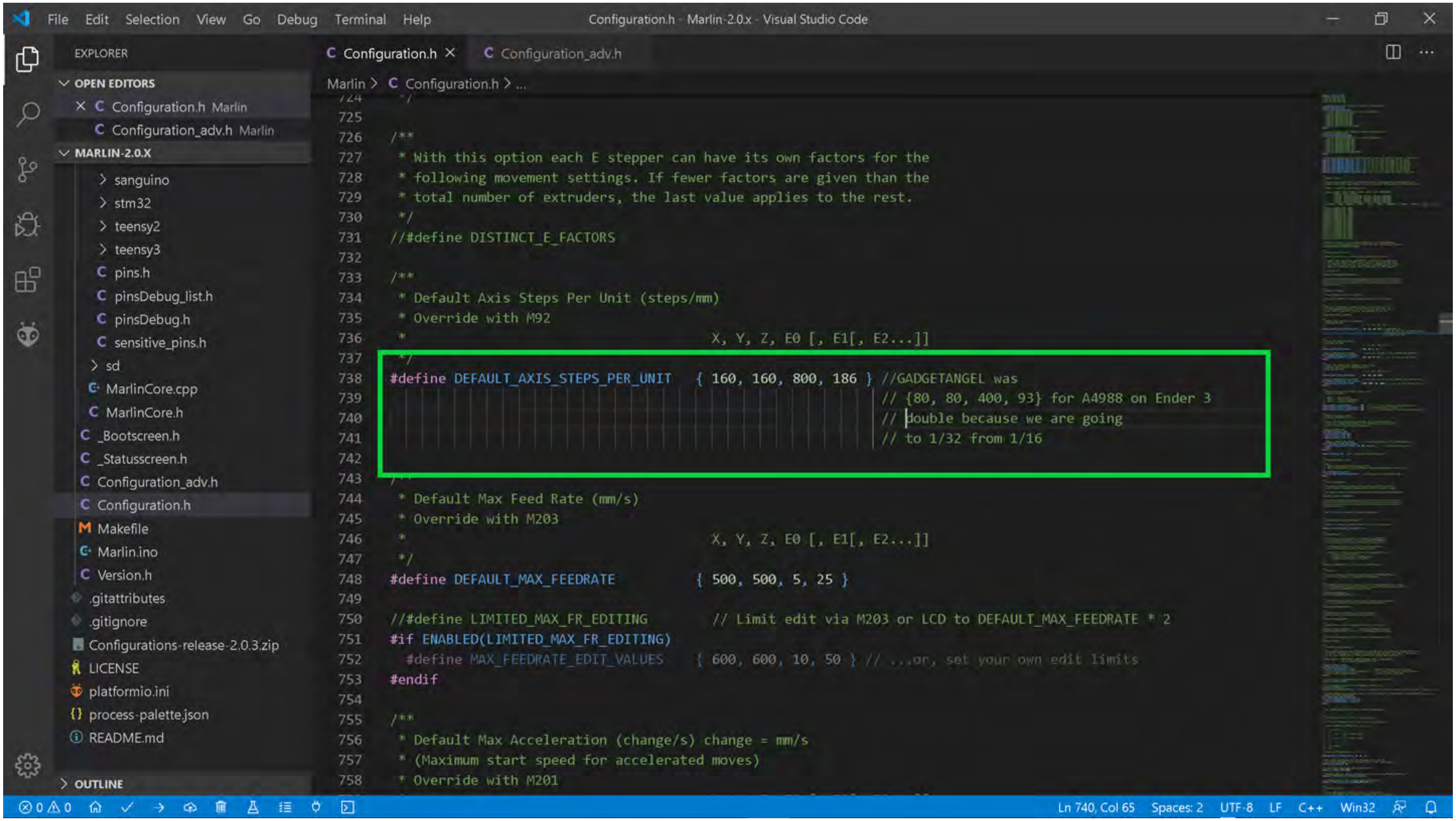
- Change the stepper motor drivers so that Marlin knows you are using Lerdge LV8729 drivers. Change one line for each axis and one for each extruder you will be using. See the picture below for an example of how to use Lerdge LV8729 drivers. When two "//" appear at the beginning of a line that means that line is commented out. To un-comment a line just remove the leading two forward slashes ("//").



- Go to the next page.

The (latest release of) Marlin Setup for LERDGE LV8729 Drivers

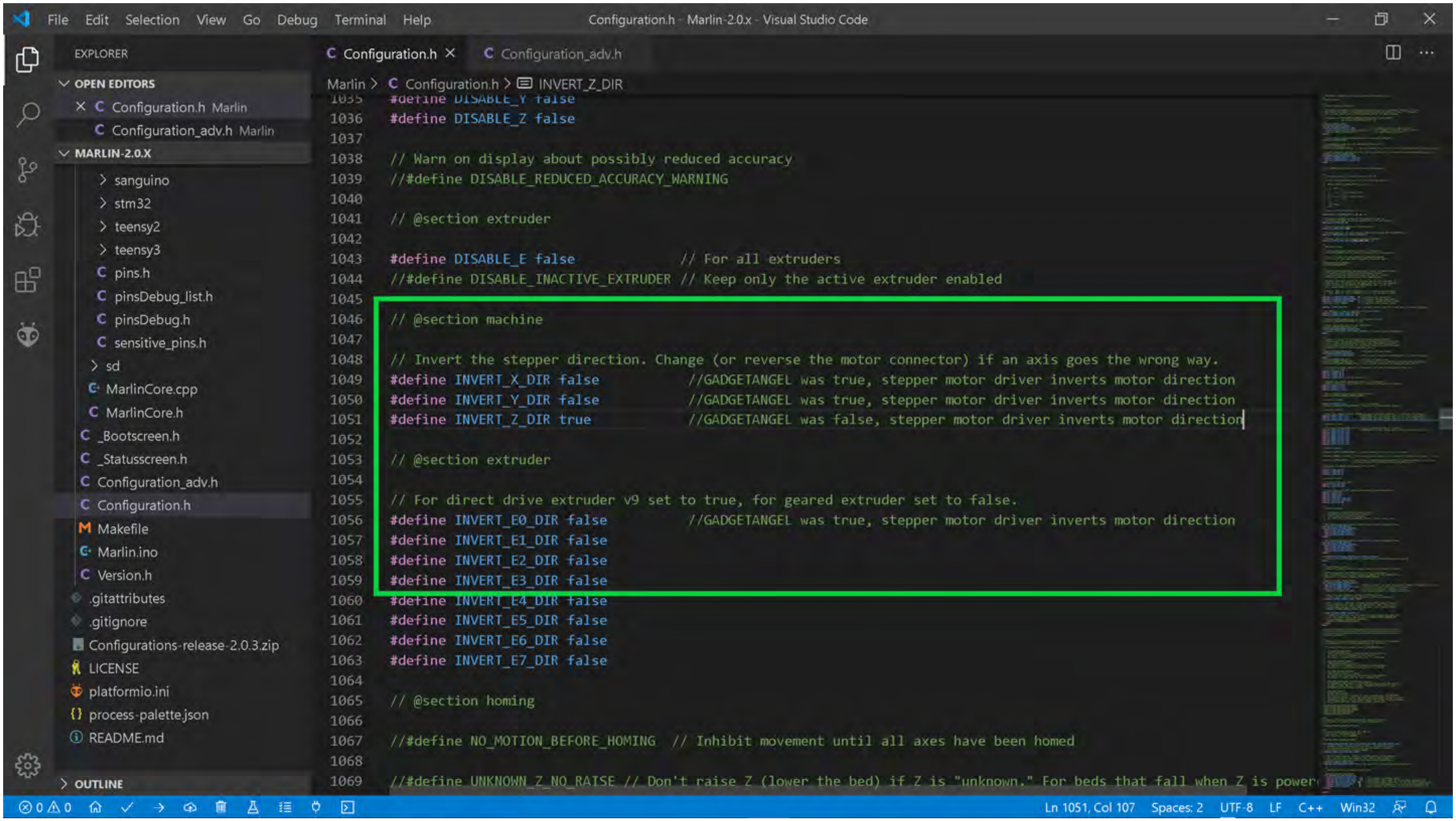
- We are changing from A4988 stepper motor drivers on the Ender 3 to LV8729 stepper motor drivers for each axis and the extruder stepper motor driver, we will be going from 1/16 stepping to 1/32 stepping. So we are doubling our STEPS. Therefore, we must adjust our "DEFAULT_AXIS_STEPS_PER_UNIT" anytime our STEPS are NOT 1/16. So change "DEFAULT_AXIS_STEPS_PER_UNIT" to {160, 160, 800, 186}, as seen in the **GREEN** box below.



- Go to the next page.

The (latest release of) Marlin Setup for LERDGE LV8729 Drivers

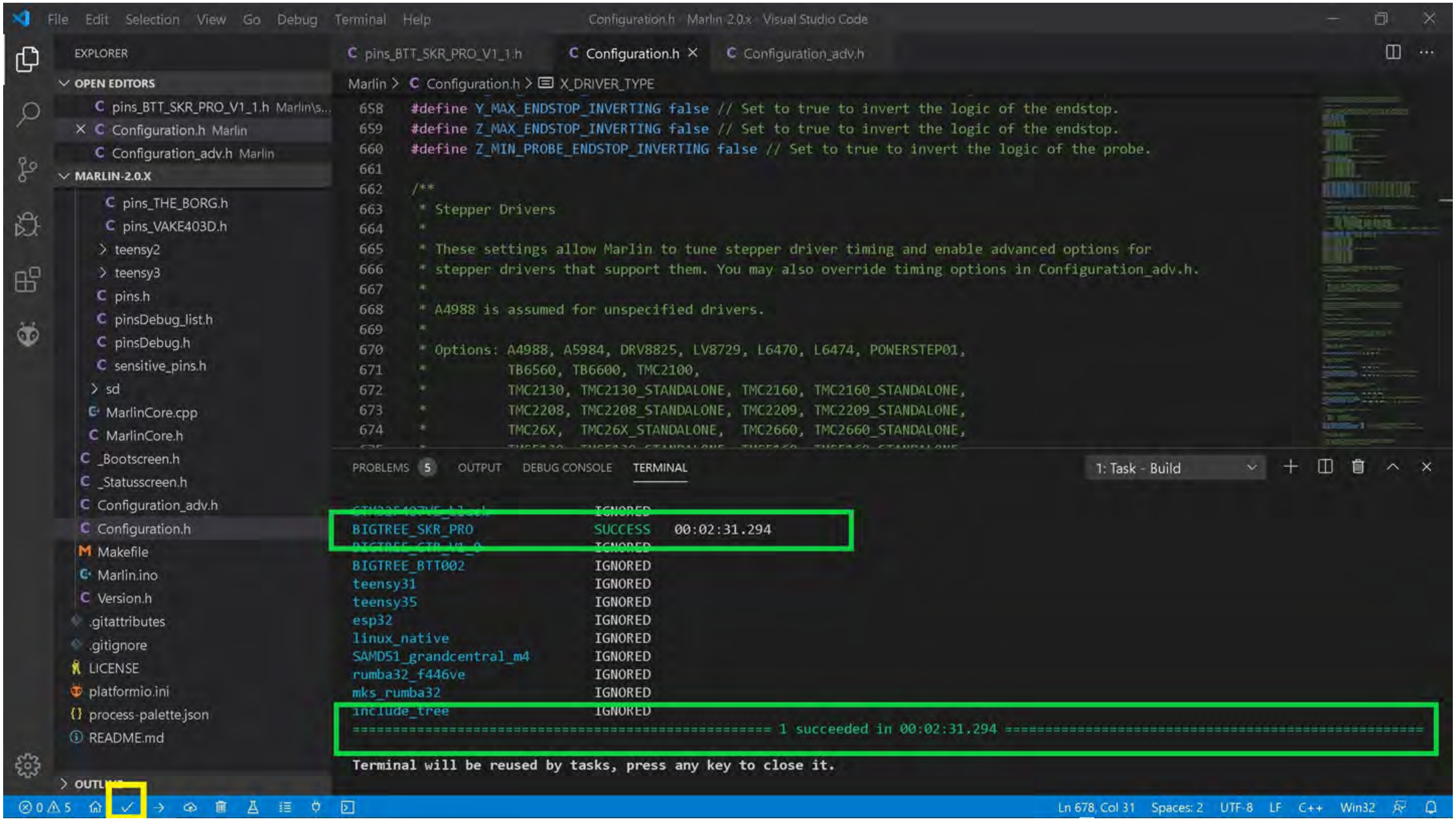
- Since the A4988 driver is what my Ender 3 used, but, now I want to use LV8729 drivers, I must invert the stepper motor direction because the LV8729 driver will turn the motors in the opposite direction than the A4988 driver's motor direction. So if the axis' setting you will be using the LV8729 driver on was "true" change it to "false", as shown in the **GREEN** box below. If the setting was "false", now set it to "true", as show in the **GREEN** box below



- Go to the next page.

The (latest release of) Marlin Setup for LERDGE LV8729 Drivers

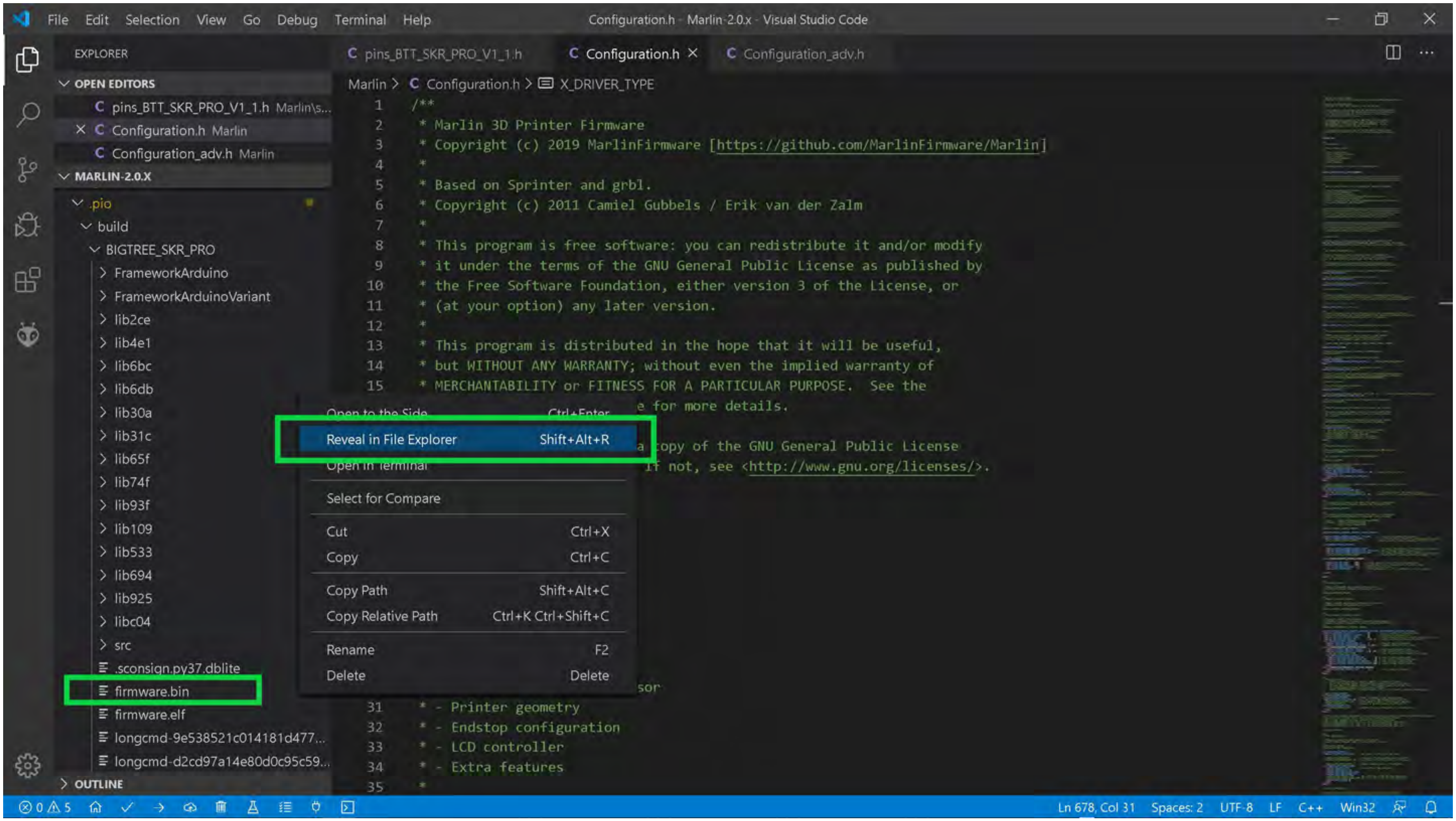
- The end of Marlin setup for LERDGE LV8729 drivers. Now, compile your code by clicking on the check mark, as seen in the **YELLOW** box below. Once the firmware has compiled successfully you will see the following messages, as shown in the **GREEN** boxes below.



- For the SKR PRO V1.1 board you must save the "firmware.bin" file to the micro SD card then place the micro SD card into the micro SD card reader of the SKR PRO V1.1 board and turn on the power to your printer for the NEW firmware to be automatically applied to the SKR PRO MCU. To locate the "firmware.bin" file, and learn how to transfer it to your micro SD card go to the next page.

The (latest release of) Marlin Setup for LERDGE LV8729 Drivers

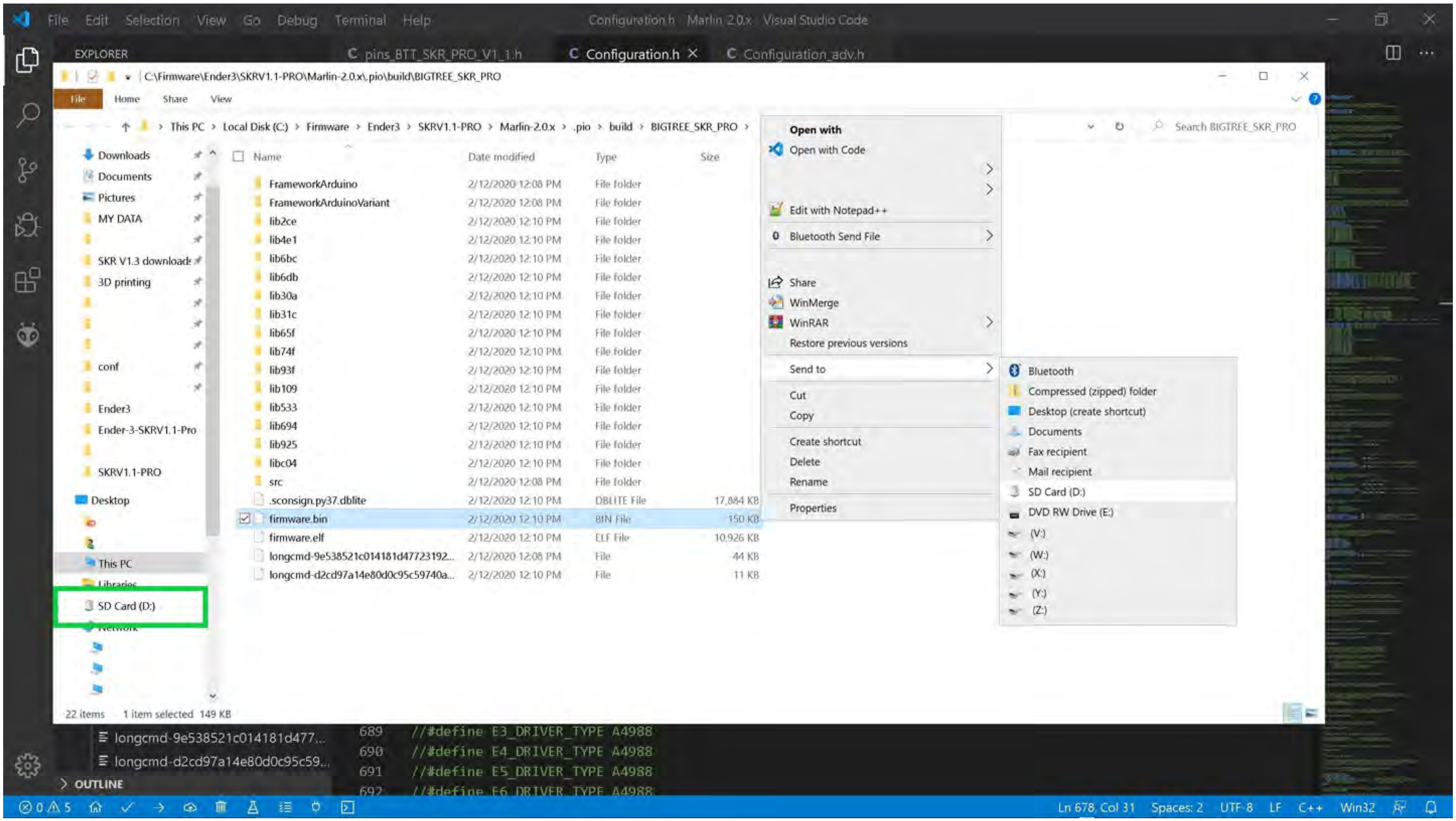
- The "firmware.bin" file can be found by using the left side of the VS code window and going to folder Marlin-2.0.X/.pio/build/ BIGTREE_SKR_PRO/ and right clicking on the "firmware.bin" file name to bring up the content menu. Select "Reveal in File Explorer" to have your Window's machine open a file explorer window.



- Go to the next page.

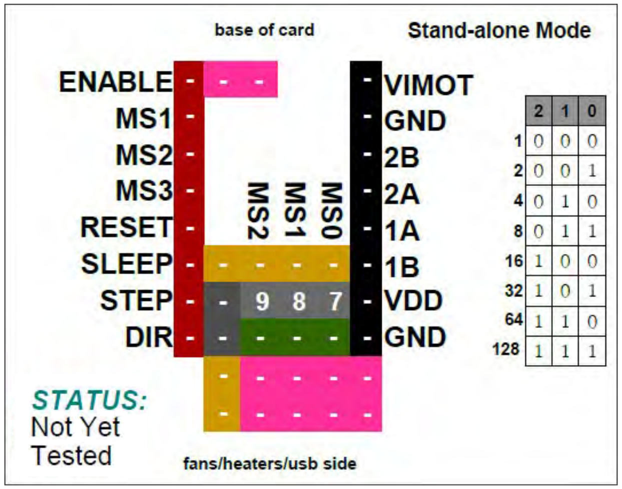
The (latest release of) Marlin Setup for LERDGE LV8729 Drivers

- Ensure your micro SD card, which came with your SKR PRO V1.1 board, with your SKR PRO V1.1 board is in a micro-to-SD-card adapter and plugged into your Window's SD card reader.
    - From the file explorer window right click on the "firmware.bin" file name and select "Send to" from the content menu. For my computer "D:" is my SD card reader. For your computer it may be a different drive letter. The "firmware.bin" file will be written to the SD card and renamed to "FIRMWARE.CUR".



- Right click on the SD card reader, as seen in the **GREEN** box above, and select "Eject" from the context menu. Take the micro SD card out of the adapter and place it into the micro SD card reader of the SKR PRO V1.1 board. Turn on the power to the printer (i.e. the SKR PRO V1.1 board) and the "FIRMWARE.CUR" file will be uploaded to the board. After the upload the firmware file will be renamed to "firmware.bin" and stored on the micro SD card.

MKS LV8729



Note: See the next page for information about location of the current sense resistors and how to set V_{ref} on the stepper motor driver board.

Driver Chip	MS2	MS1	MS0	Microstep Resolution	Excitation Mode
Makerbase LV8729 Maximum 128 Subdivision 36V DC 1.5A (peak)	Low	Low	Low	Full Step	2 Phase
	Low	Low	High	1/2 Step	1-2 Phase
	Low	High	Low	1/4 Step	W1-2 Phase
	Low	High	High	1/8 Step	2W1-2 Phase
	High	Low	Low	1/16 Step	4W1-2 Phase
	High	Low	High	1/32 Step	8W1-2 Phase
	High	High	Low	1/64 Step	16W1-2 Phase
	High	High	High	1/128 Step	32W1-2 Phase

Driving Current Calculation Formula

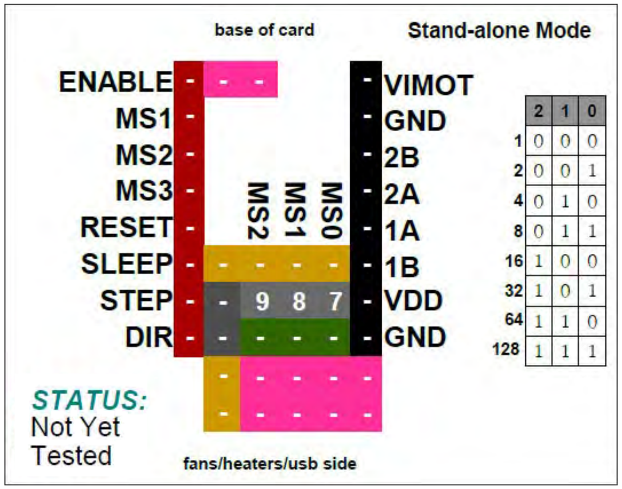
R_S (Typical Sense Resistor) $= 0.22\Omega$

$$I_{MAX} = \frac{V_{ref}}{5 * R_S}$$

$$V_{ref} = 5 * I_{MAX} * R_S$$

NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

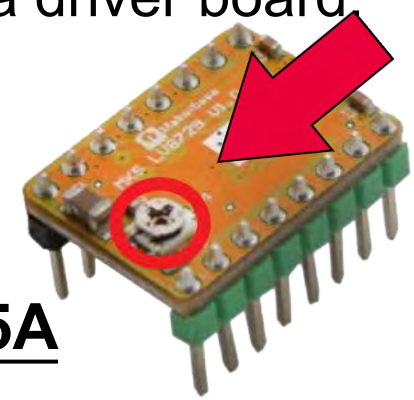
MKS LV8729



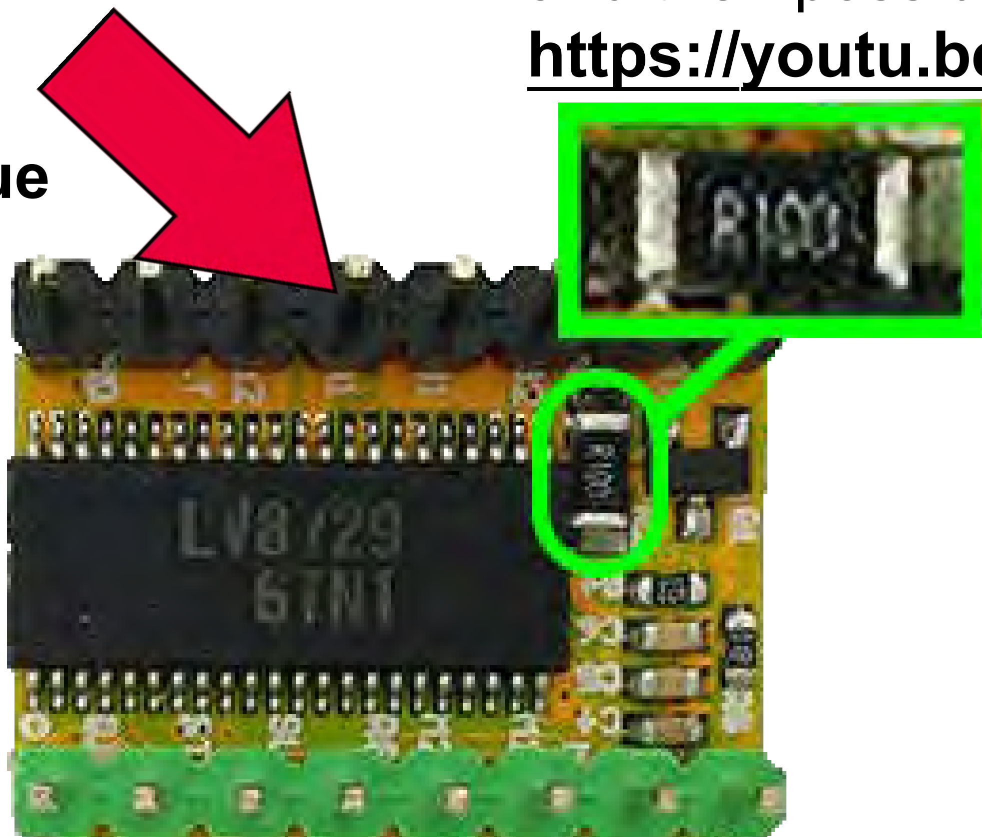
Note: Use 90% of the calculated V_{ref} when tuning the stepper driver board.

NOTE: Use the potentiometer (POT) on the top of the board to adjust your V_{ref}. See **Appendix A** for instructions on how to set the V_{ref} on a driver board.

Note: See this video about current sense resistors (R_s) and their possible locations: **https://youtu.be/8wk1elugv5A**



Note: Check your current sense resistor (R_s) value on the driver board, as shown in GREEN



R_s = R050 is 0.05 Ohms

R_s = R068 is 0.068 Ohms

R_s = R100 is 0.1 Ohms

R_s = R150 is 0.15 Ohms

R_s = R200 is 0.2 Ohms

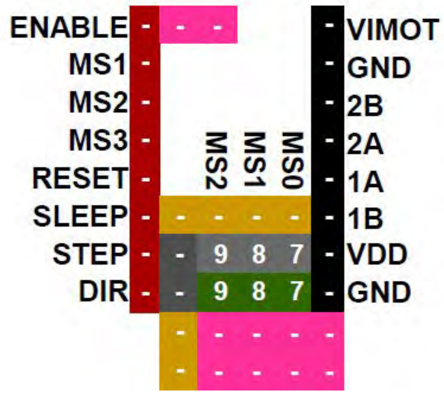
R_s = R220 is 0.22 Ohms

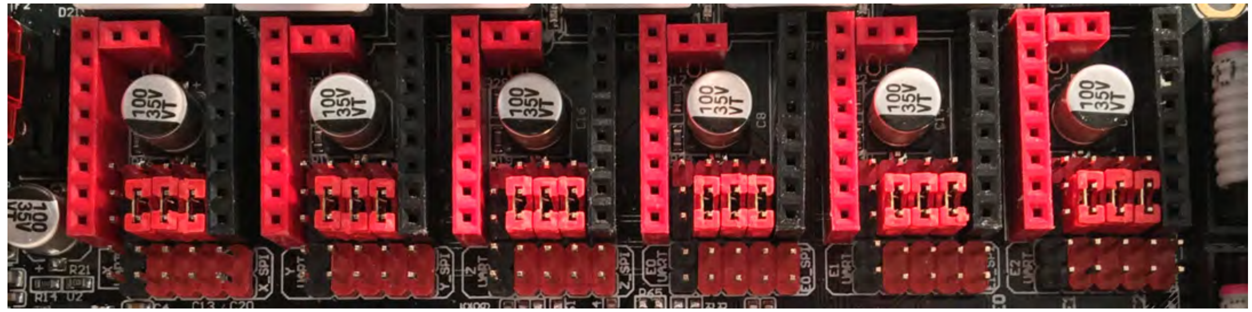
NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

MKS LV8729

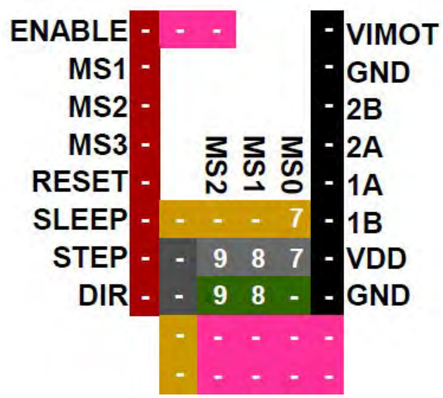
Stand-alone Mode

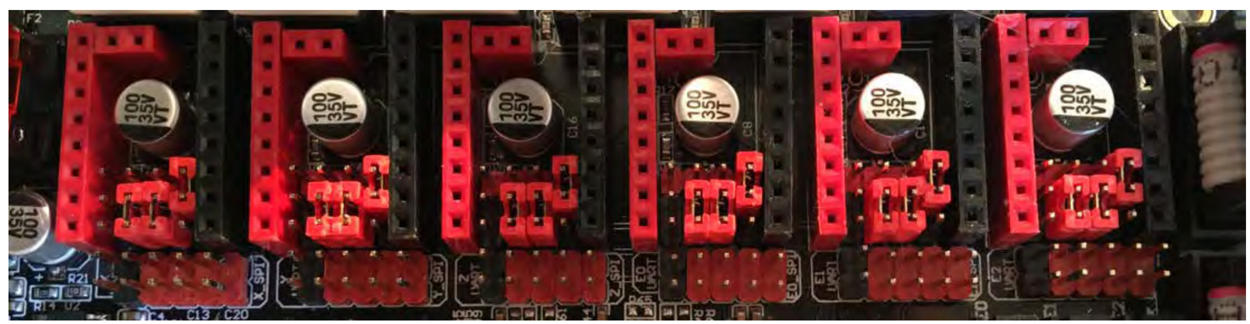
STEP





1 / 2



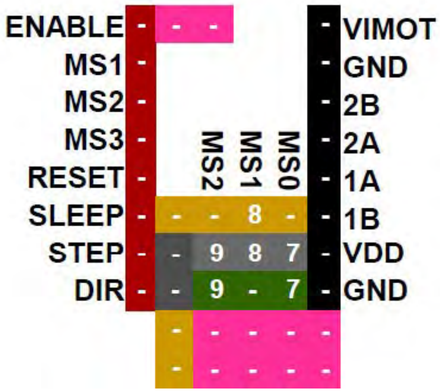


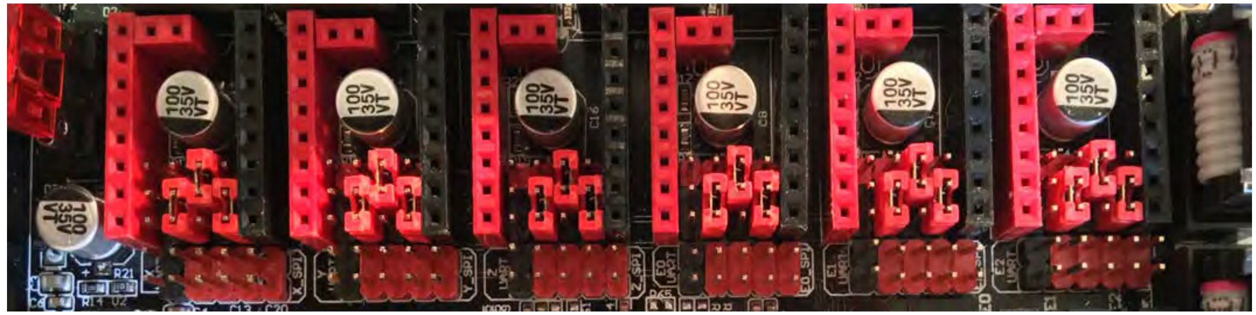
NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

MKS LV8729

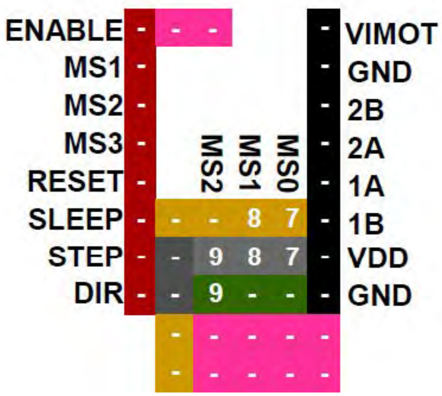
Stand-alone Mode

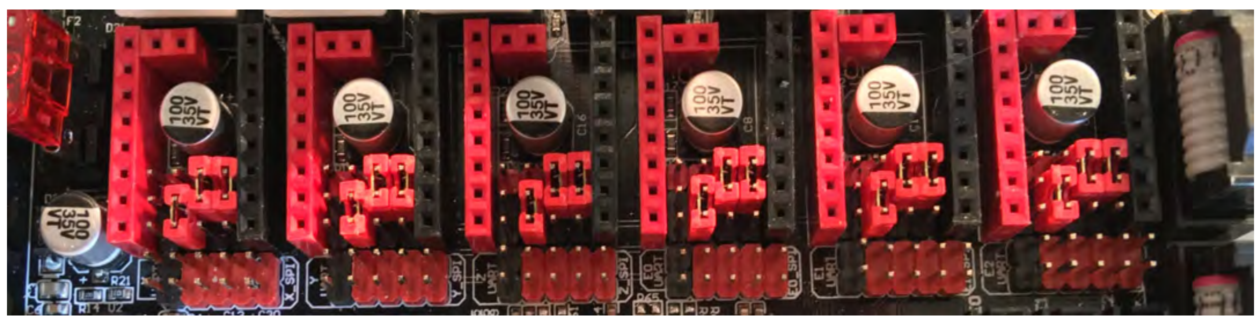
1 / 4



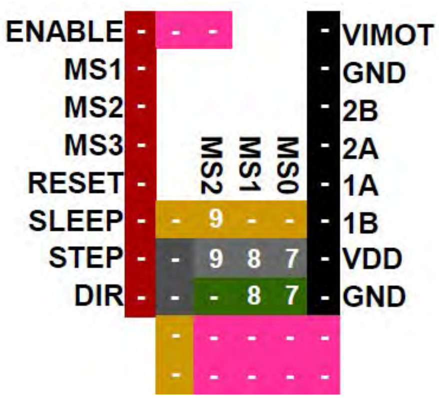


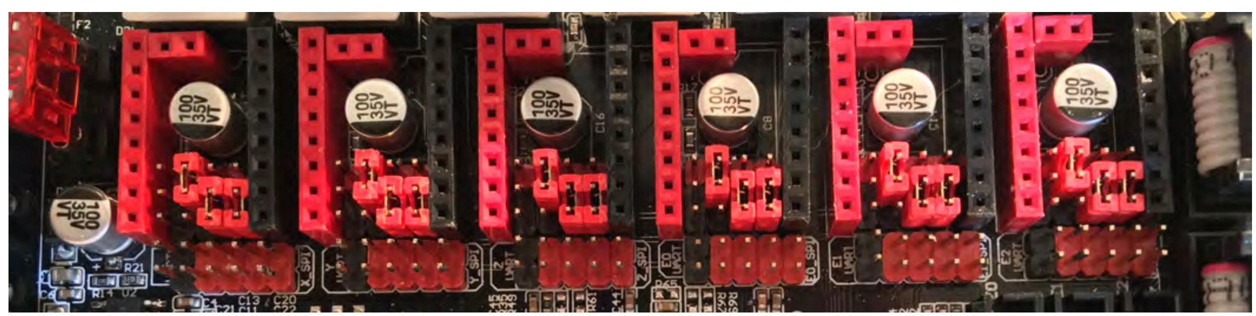
1 / 8





1 / 16



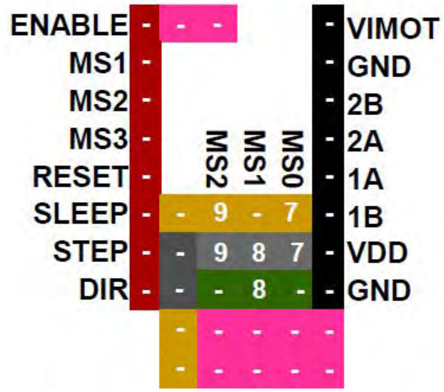


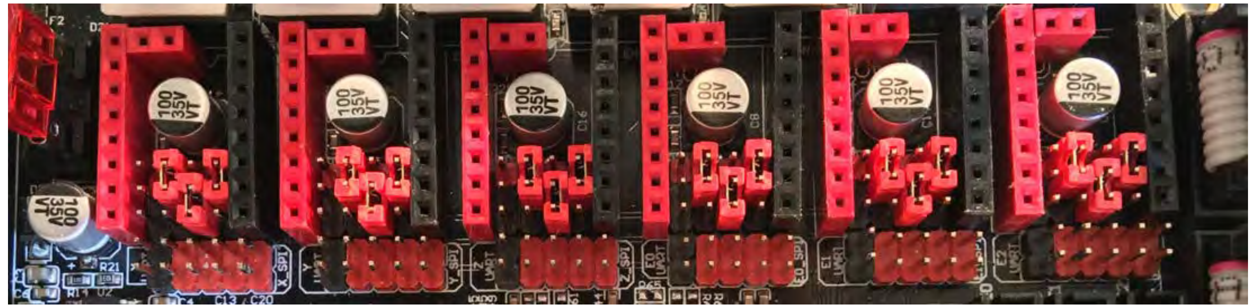
NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

MKS LV8729

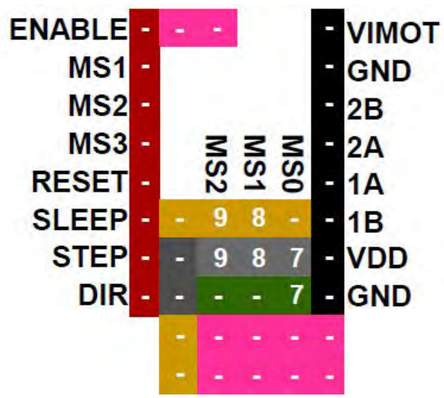
Stand-alone Mode

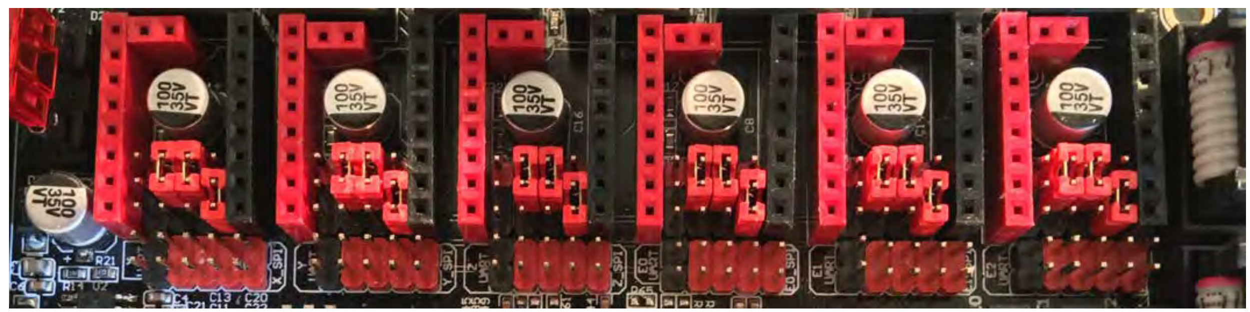
1 / 32



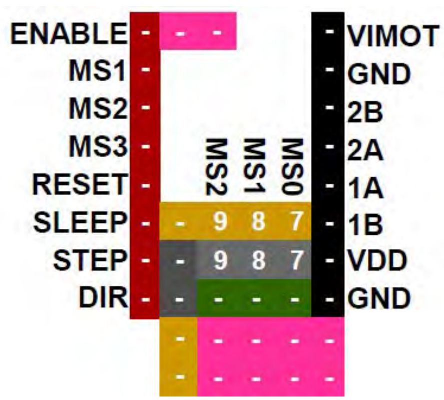


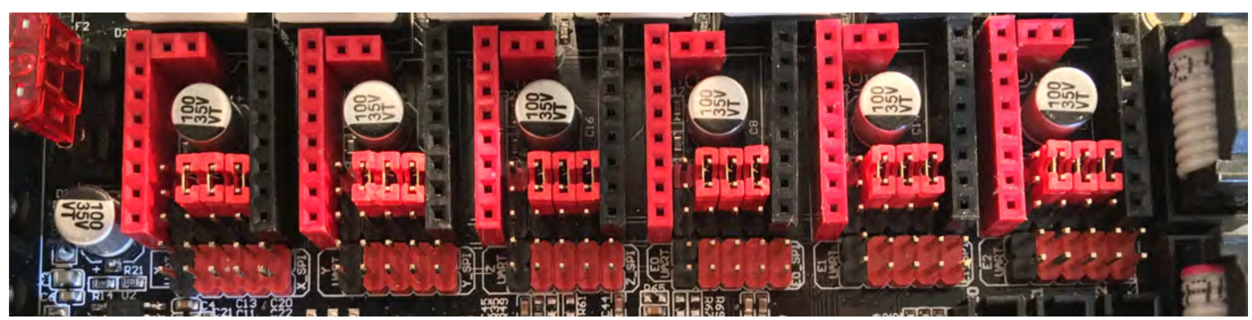
1 / 64





1 / 128



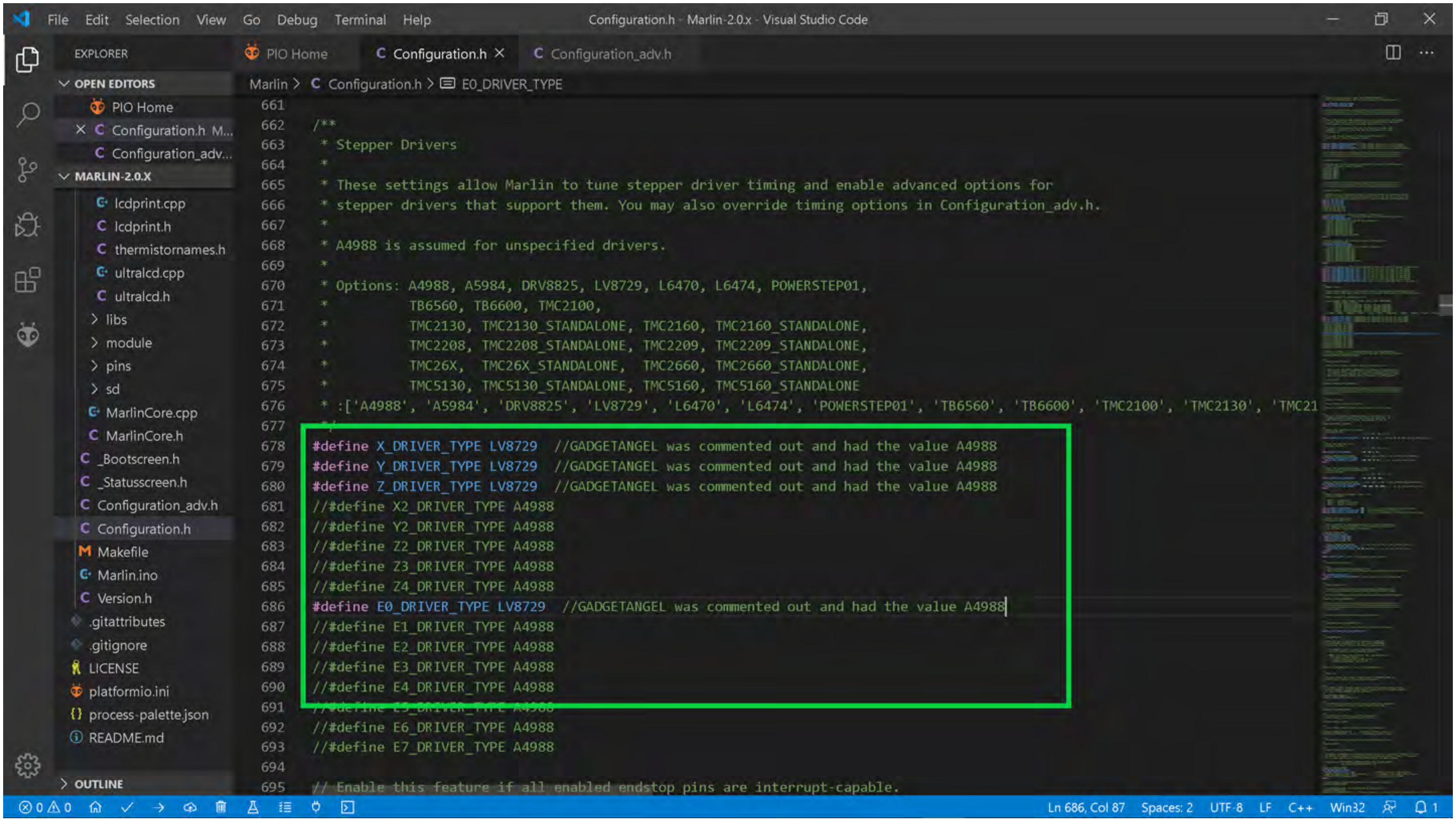


NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

The (latest release of) Marlin Setup for MKS LV8729 Drivers

NOTE: **Go to Appendix C,** and then come back here for the changes to Marlin for MKS LV8729 stepper motor drivers.

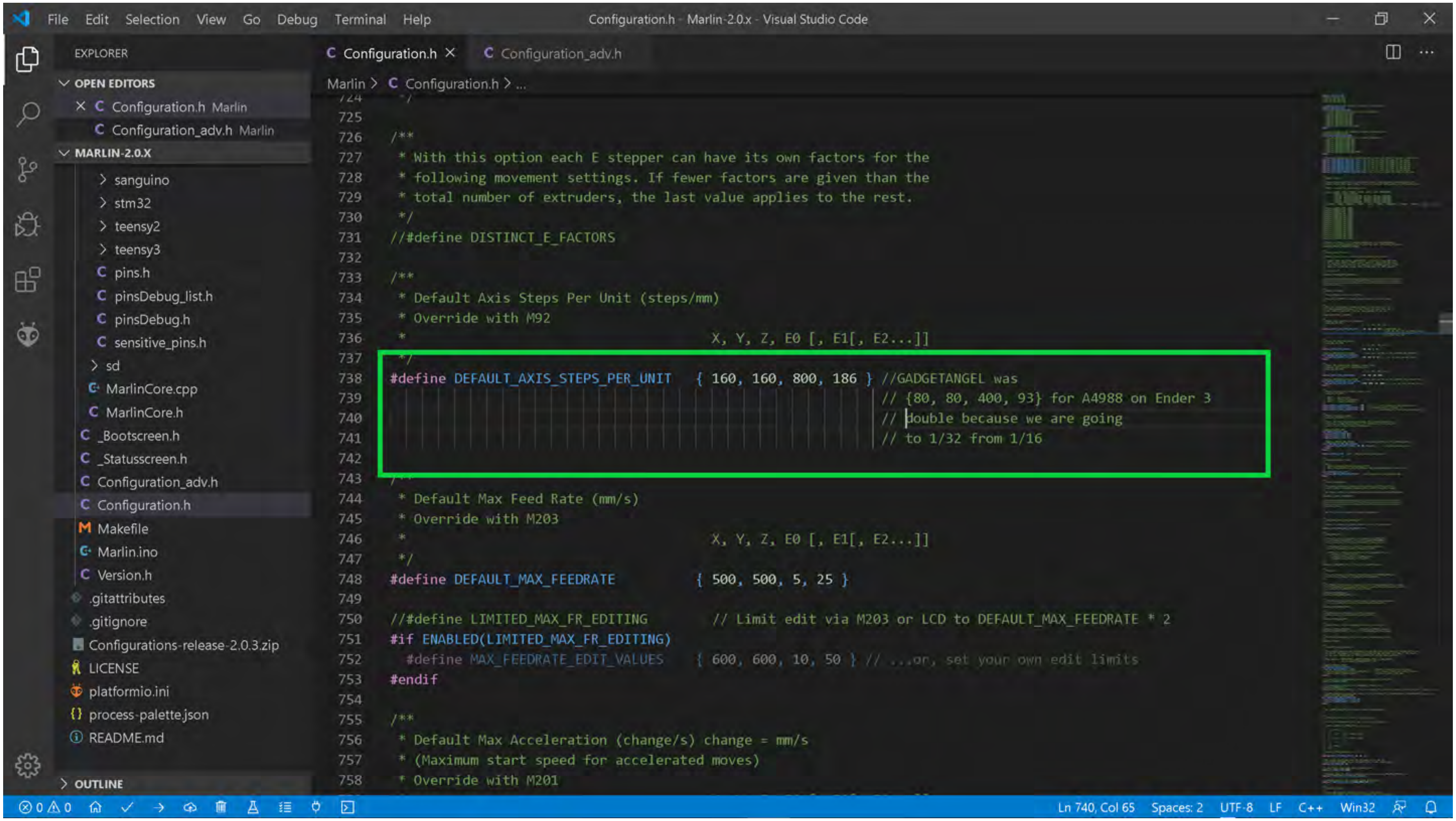
- Change the stepper motor drivers so that Marlin knows you are using MKS LV8729 drivers. Change one line for each axis and one for each extruder you will be using. See the picture below for an example of how to use MKS LV8729 drivers. When two "//" appear at the beginning of a line that means that line is commented out. To un-comment a line just remove the leading two forward slashes ("//").



- Go to the next page.

The (latest release of) Marlin Setup for MKS LV8729 Drivers

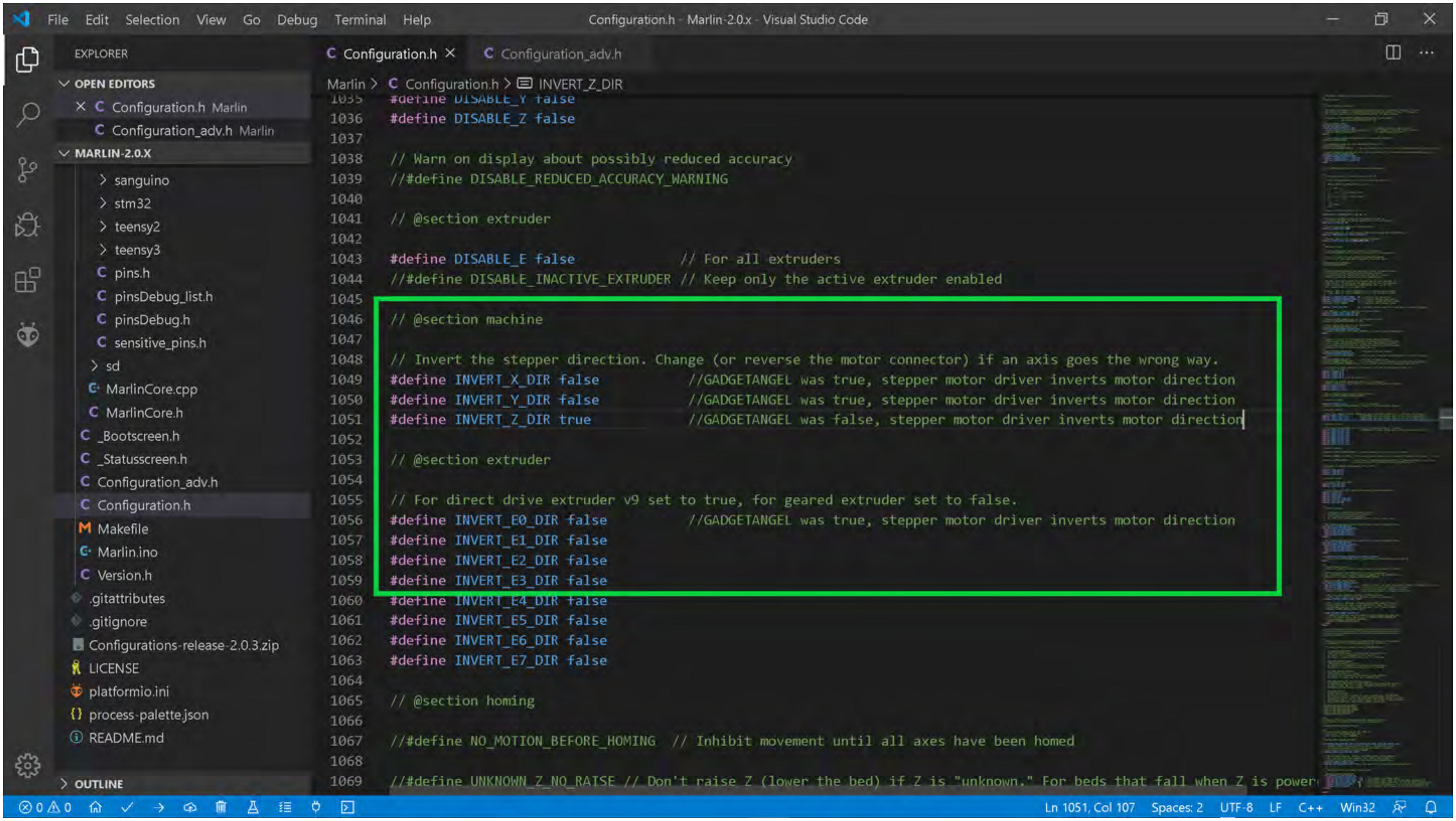
- We are changing from A4988 stepper motor drivers on the Ender 3 to LV8729 stepper motor drivers for each axis and the extruder stepper motor driver, we will be going from 1/16 stepping to 1/32 stepping. So we are doubling our STEPS. Therefore, we must adjust our "DEFAULT_AXIS_STEPS_PER_UNIT" anytime our STEPS are NOT 1/16. So change "DEFAULT_AXIS_STEPS_PER_UNIT" to {160, 160, 800, 186}, as seen in the **GREEN** box below.



- Go to the next page.

The (latest release of) Marlin Setup for MKS LV8729 Drivers

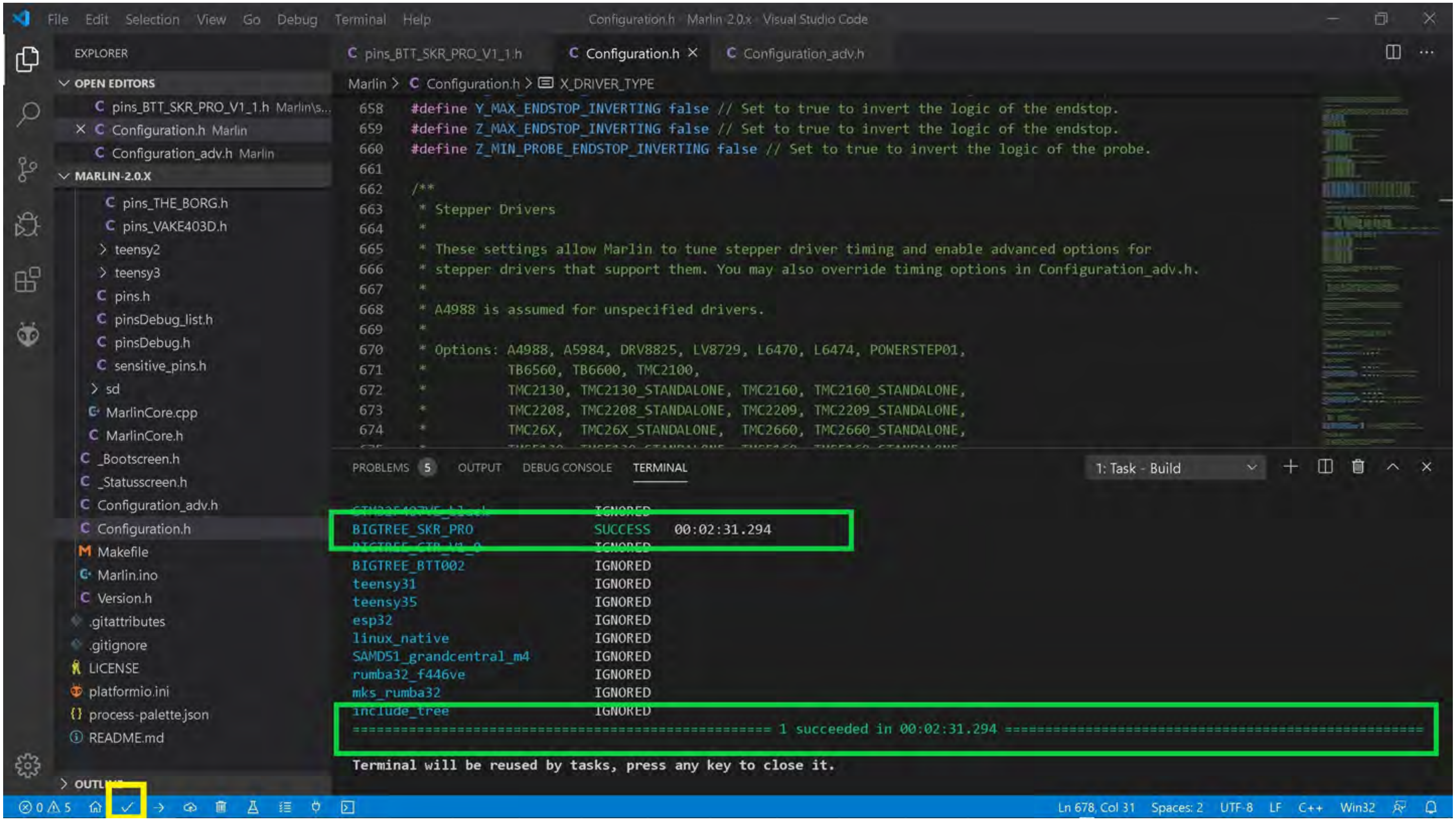
- Since the A4988 driver is what my Ender 3 used, but, now I want to use LV8729 drivers, I must invert the stepper motor direction because the LV8729 driver will turn the motors in the opposite direction than the A4988 driver's motor direction. So if the axis' setting you will be using the LV8729 driver on was "true" change it to "false", as shown in the **GREEN** box below. If the setting was "false", now set it to "true", as show in the **GREEN** box below



- Go to the next page.

The (latest release of) Marlin Setup for MKS LV8729 Drivers

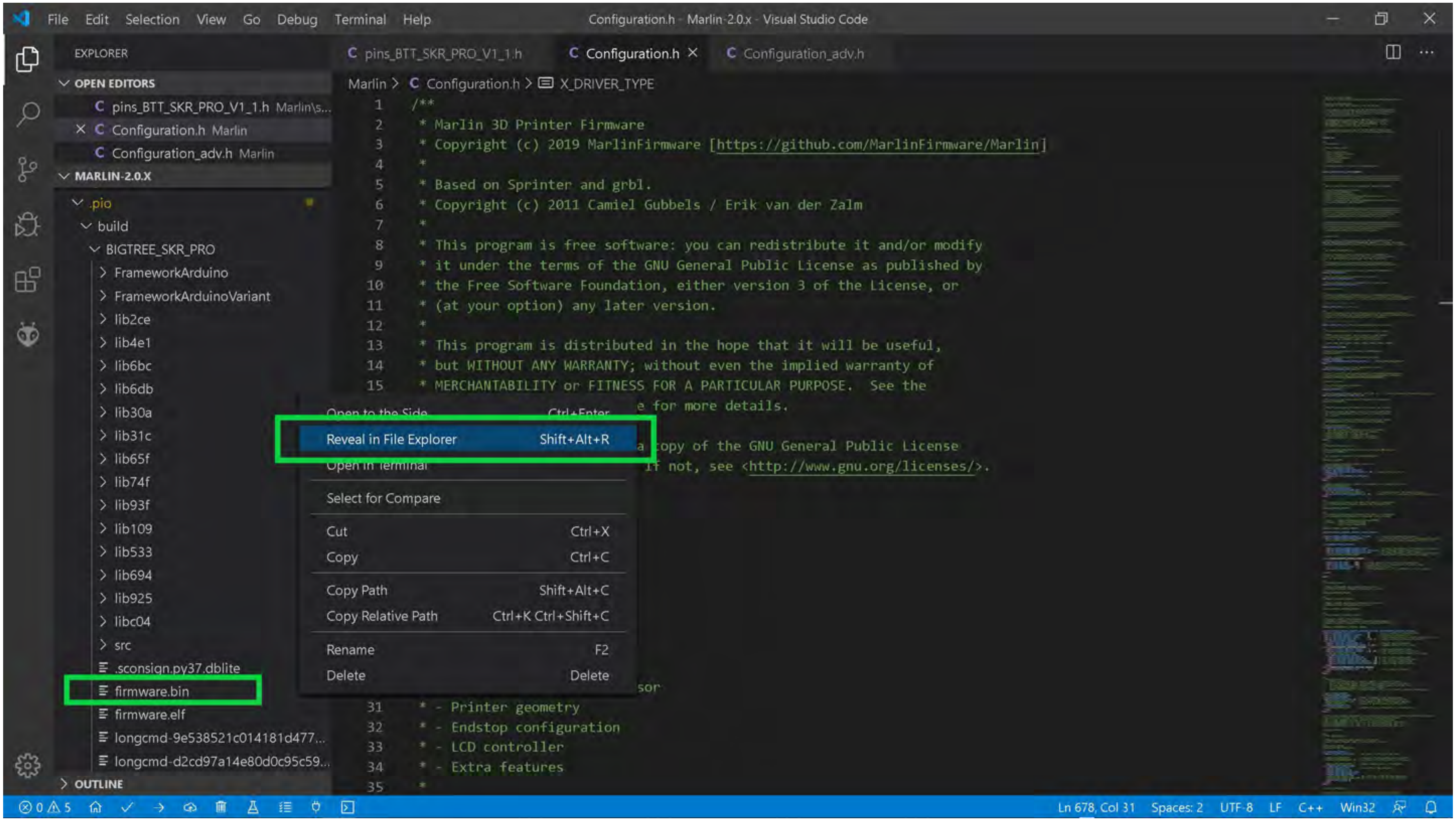
- The end of Marlin setup for MKS LV8729 drivers. Now, compile your code by clicking on the check mark, as seen in the **YELLOW** box below. Once the firmware has compiled successfully you will see the following messages, as shown in the **GREEN** boxes below.



- For the SKR PRO V1.1 board you must save the "firmware.bin" file to the micro SD card then place the micro SD card into the micro SD card reader of the SKR PRO V1.1 board and turn on the power to your printer for the NEW firmware to be automatically applied to the SKR PRO MCU. To locate the "firmware.bin" file, and learn how to transfer it to your micro SD card go to the next page.

The (latest release of) Marlin Setup for MKS LV8729 Drivers

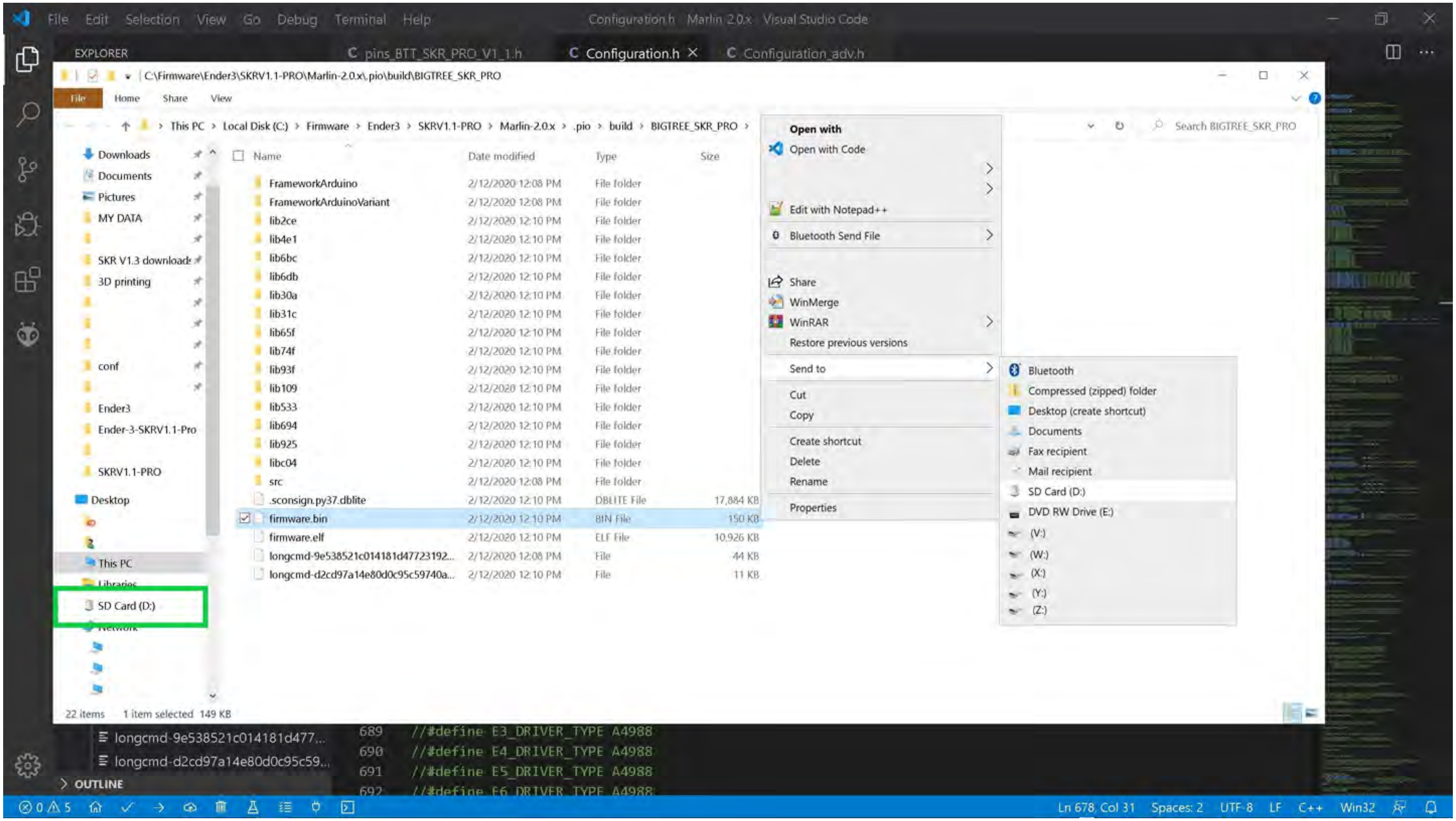
- The "firmware.bin" file can be found by using the left side of the VS code window and going to folder Marlin-2.0.X/.pio/build/BIGTREE_SKR_PRO/ and right clicking on the "firmware.bin" file name to bring up the content menu. Select "Reveal in File Explorer" to have your Window's machine open a file explorer window.



- Go to the next page.

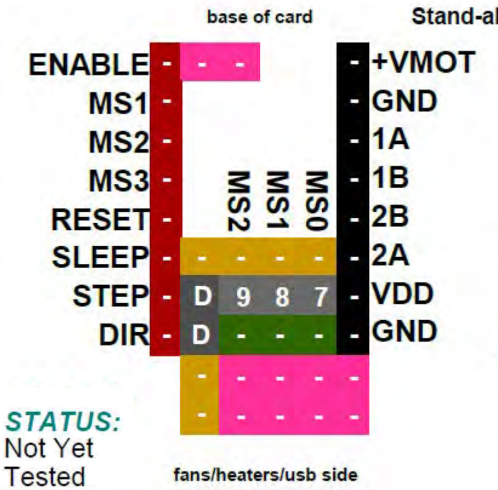
The (latest release of) Marlin Setup for MKS LV8729 Drivers

- Ensure your micro SD card, which came with your SKR PRO V1.1 board, with your SKR PRO V1.1 board is in a micro-to-SD-card adapter and plugged into your Window's SD card reader.
  - From the file explorer window right click on the "firmware.bin" file name and select "Send to" from the content menu. For my computer "D:" is my SD card reader. For your computer it may be a different drive letter. The "firmware.bin" file will be written to the SD card and renamed to "FIRMWARE.CUR".



- Right click on the SD card reader, as seen in the **GREEN** box above, and select "Eject" from the context menu. Take the micro SD card out of the adapter and place it into the micro SD card reader of the SKR PRO V1.1 board. Turn on the power to the printer (i.e. the SKR PRO V1.1 board) and the "FIRMWARE.CUR" file will be uploaded to the board. After the upload the firmware file will be renamed to "firmware.bin" and stored on the micro SD card.

FYSETC S6128 V1.1



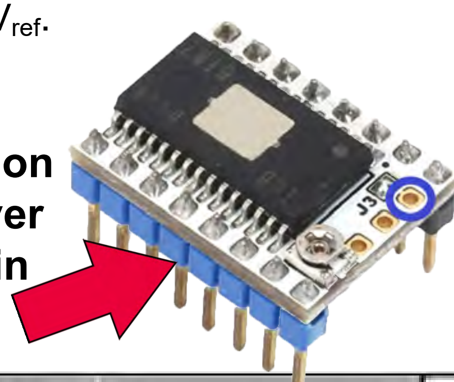
	2	1	0
1	0	0	0
2	0	0	1
4	0	1	0
8	0	1	1
16	1	0	0
32	1	0	1
64	1	1	0
128	1	1	1

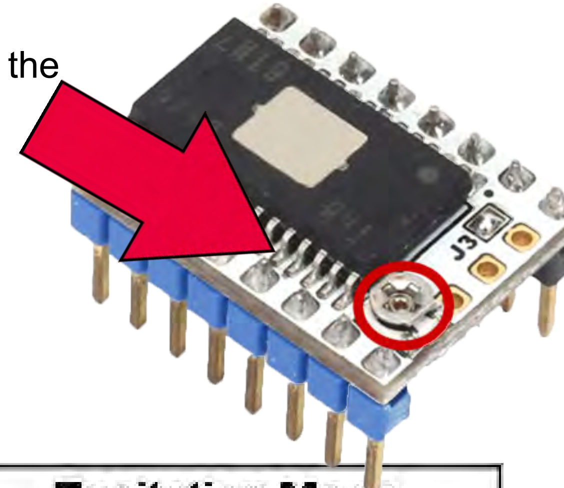
STATUS: Not Yet Tested

FYSETC S6128 V1.1 states: "SureStepr SD6128 is a stepper driver board based on the THB6128 chip"; my understanding is that the THB6128 and the LV8729 are the same chip in different packages.

NOTE: Use the potentiometer (POT) on the top of the board to adjust your V_{ref}.

Note: "V_{ref} Test point" location is on the top of the driver board, as shown in BLUE





Driver Chip	MS2	MS1	MS0	Microstep Resolution	Excitation Mode
FYSETC SureStepr SD6128 Maximum 128 Subdivision 35V DC 2.2A (peak)	Low	Low	Low	Full Step	2 Phase
	Low	Low	High	1/2 Step	1-2 Phase
	Low	High	Low	1/4 Step	W1-2 Phase
	Low	High	High	1/8 Step	2W1-2 Phase
	High	Low	Low	1/16 Step	4W1-2 Phase
	High	Low	High	1/32 Step	8W1-2 Phase
	High	High	Low	1/64 Step	16W1-2 Phase
	High	High	High	1/128 Step	32W1-2 Phase

Driving Current Calculation Formula

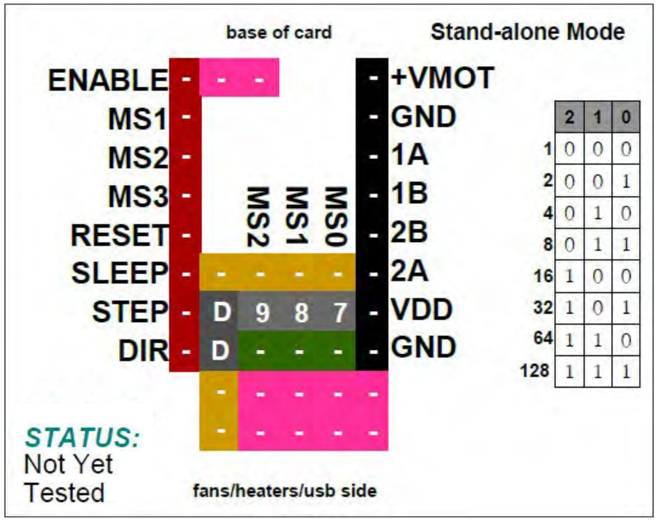
R_S(Typical Sense Resistor)=0.1Ω

$$I_{MAX} = \frac{V_{ref}}{5 * R_S}$$

$$V_{ref} = 5 * I_{MAX} * R_S$$

NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

FYSETC S6128 V1.1



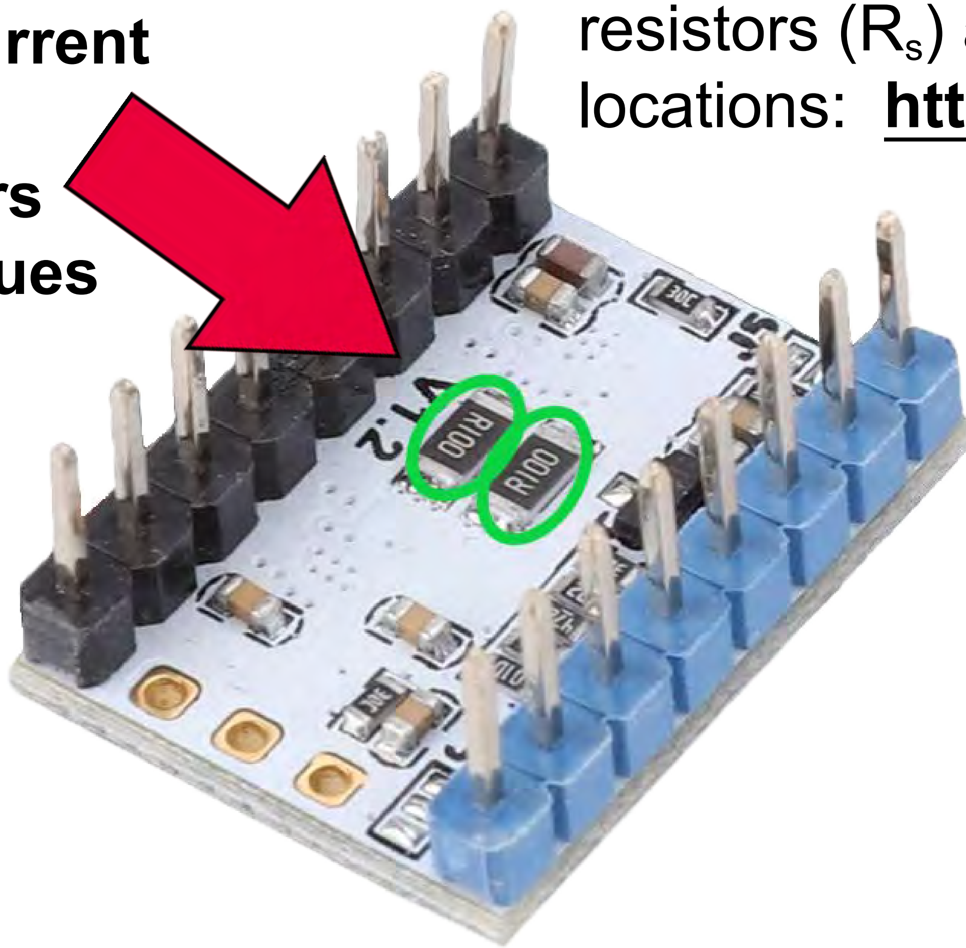
	2	1	0
1	0	0	0
2	0	0	1
4	0	1	0
8	0	1	1
16	1	0	0
32	1	0	1
64	1	1	0
128	1	1	1

Note: Use 90% of the calculated V_{ref} when tuning the stepper driver board.

NOTE: Use the potentiometer (POT) on the top of the board to adjust your V_{ref}. See **Appendix A** for instructions on how to set the V_{ref} on a driver board.

Note: See this video about current sense resistors (R_s) and their possible locations: **https://youtu.be/8wk1elugv5A**

Note: Check your current sense resistors (R_s) values on the driver board, as shown in GREEN



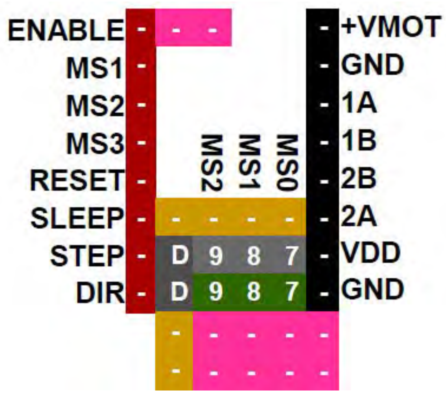
R_s = R050 is 0.05 Ohms
R_s = R068 is 0.068 Ohms
R_s = R100 is 0.1 Ohms
R_s = R150 is 0.15 Ohms
R_s = R200 is 0.2 Ohms
R_s = R220 is 0.22 Ohms

FYSETC S6128 V1.1

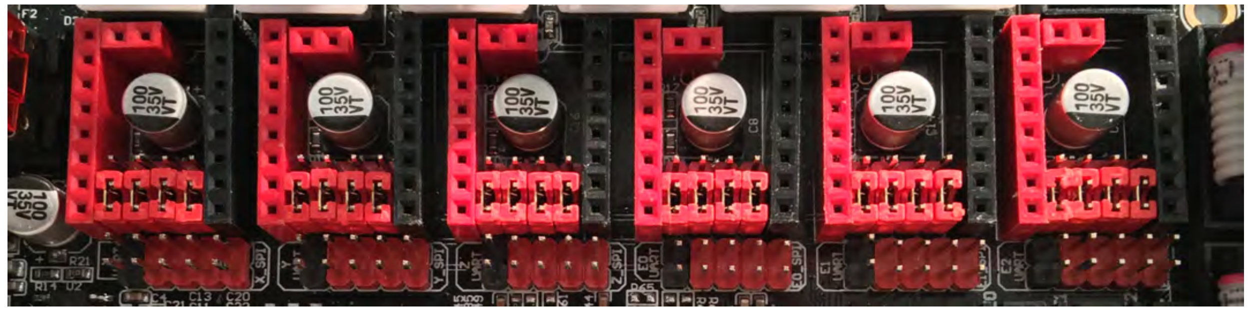
Stand-alone Mode

Important: This driver has special requirements in the Configuration and Configuration_adv.h. Also, this driver requires constant cooling the moment any motor is used, or it will switch on and off.

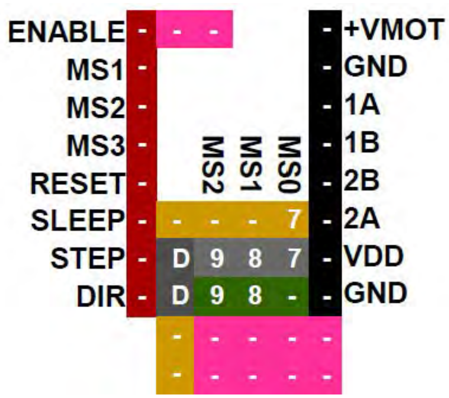
STEP



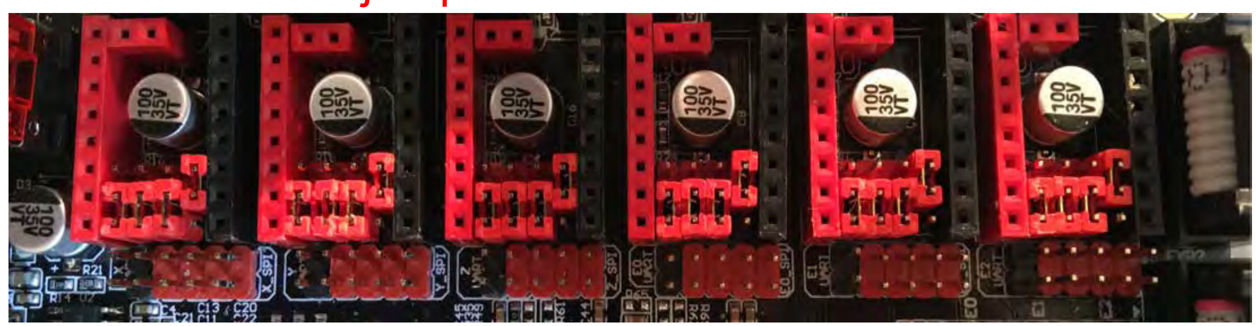
Note: The "D" jumper MUST be SET!



1 / 2



Note: The "D" jumper MUST be SET!



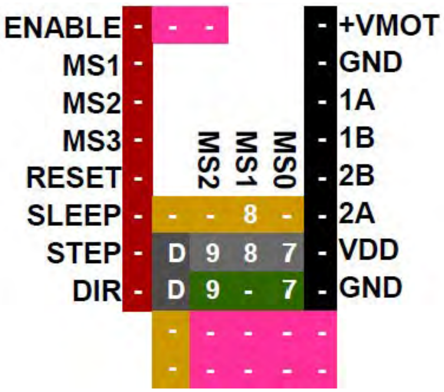
NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

FYSETC S6128 V1.1

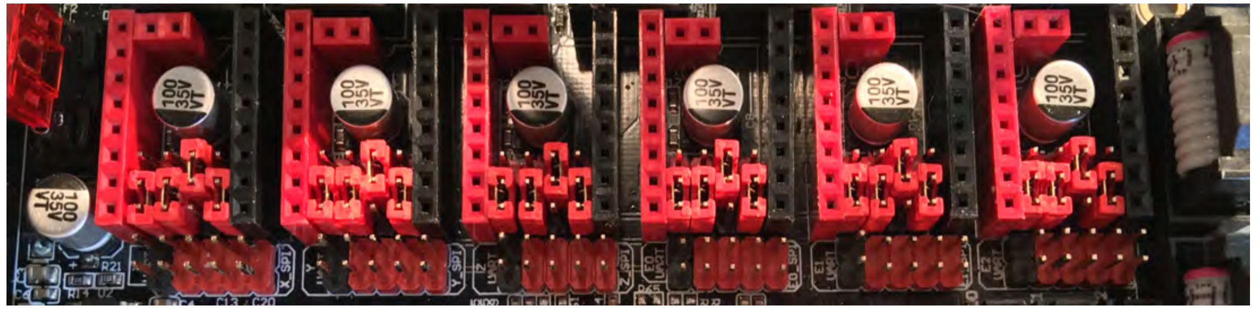
Stand-alone Mode

Important: This driver has special requirements in the Configuration and Configuration_adv.h. Also, this driver requires constant cooling the moment any motor is used, or it will switch on and off.

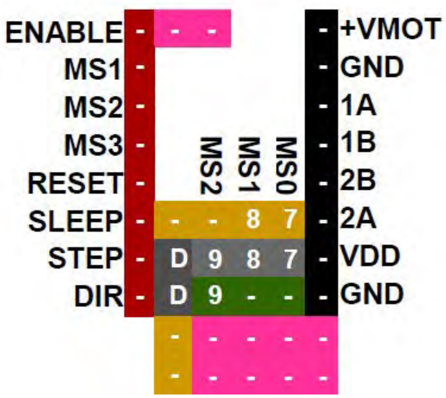
1 / 4



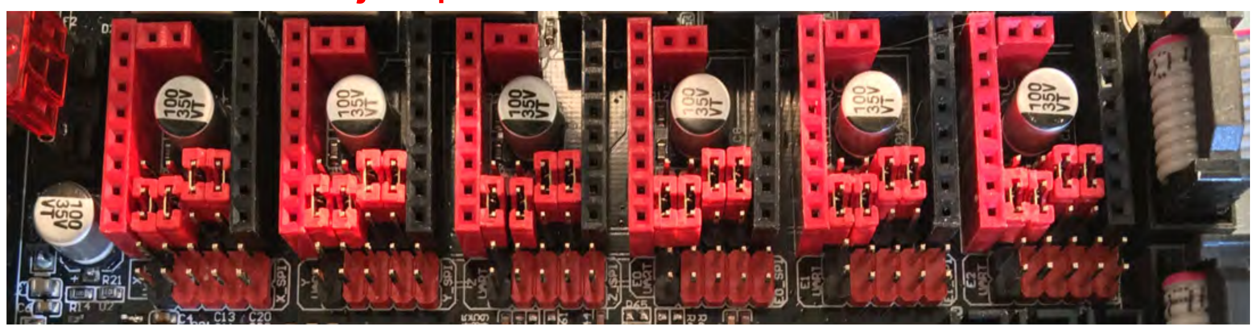
Note: The "D" jumper MUST be SET!



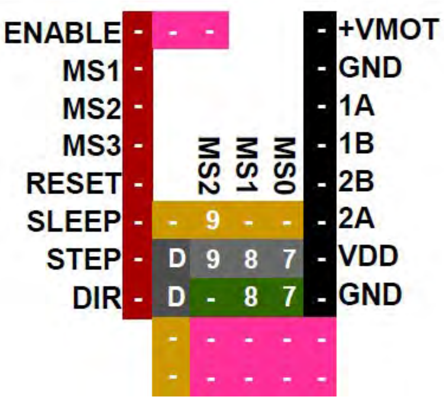
1 / 8



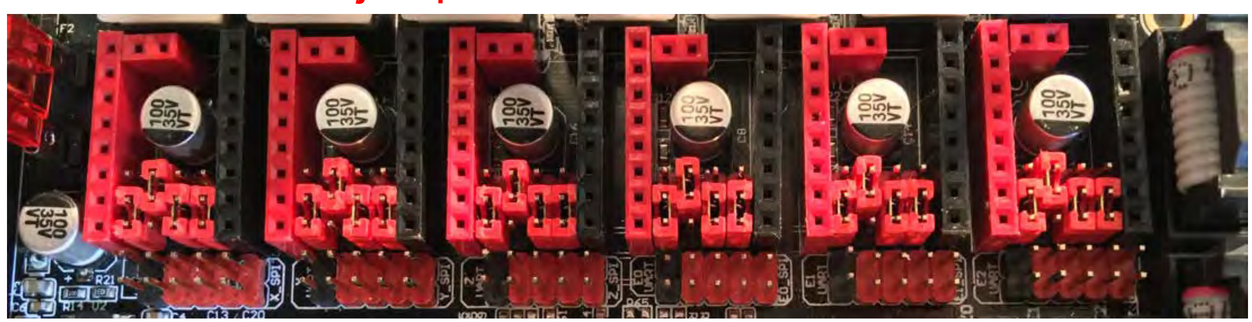
Note: The "D" jumper MUST be SET!



1 / 16



Note: The "D" jumper MUST be SET!



NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

FYSETC S6128 V1.1

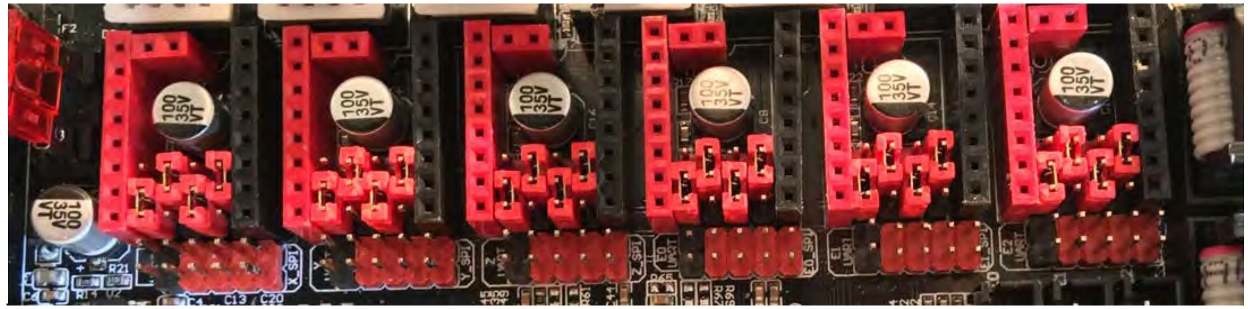
Stand-alone Mode

Important: This driver has special requirements in the Configuration and Configuration_adv.h. Also, this driver requires constant cooling the moment any motor is used, or it will switch on and off.

1 / 32

			MS2	MS1	MS0		
ENABLE	-	-	-			-	+VMOT
MS1	-					-	GND
MS2	-					-	1A
MS3	-					-	1B
RESET	-					-	2B
SLEEP	-	-	9	-	7	-	2A
STEP	-	D	9	8	7	-	VDD
DIR	-	D	-	8	-	-	GND
		-	-	-	-	-	
		-	-	-	-	-	

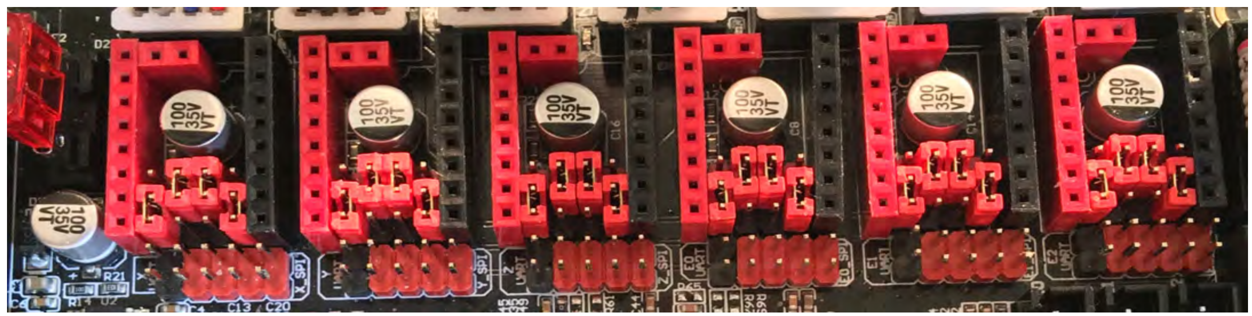
Note: The "D" jumper MUST be SET!



1 / 64

			MS2	MS1	MS0		
ENABLE	-	-	-			-	+VMOT
MS1	-					-	GND
MS2	-					-	1A
MS3	-					-	1B
RESET	-					-	2B
SLEEP	-	-	9	8	-	-	2A
STEP	-	D	9	8	7	-	VDD
DIR	-	D	-	-	7	-	GND
		-	-	-	-	-	
		-	-	-	-	-	

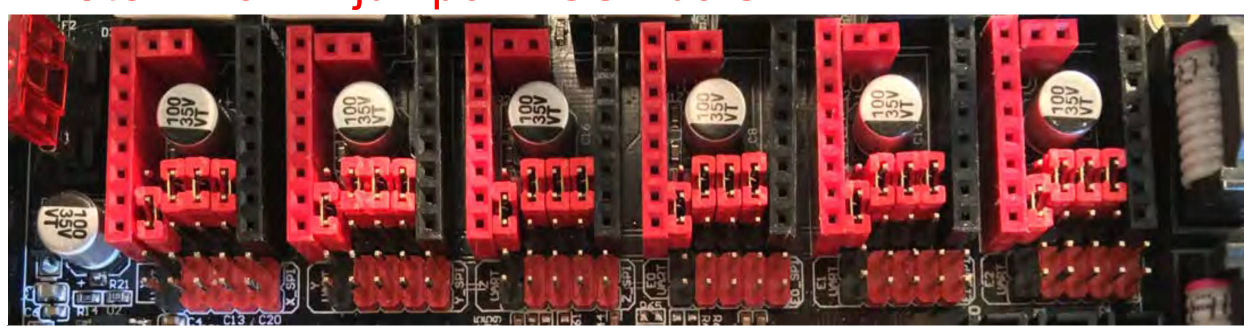
Note: The "D" jumper MUST be SET!



1 / 128

			MS2	MS1	MS0		
ENABLE	-	-	-			-	+VMOT
MS1	-					-	GND
MS2	-					-	1A
MS3	-					-	1B
RESET	-					-	2B
SLEEP	-	-	9	8	7	-	2A
STEP	-	D	9	8	7	-	VDD
DIR	-	D	-	-	-	-	GND
		-	-	-	-	-	
		-	-	-	-	-	

Note: The "D" jumper MUST be SET!

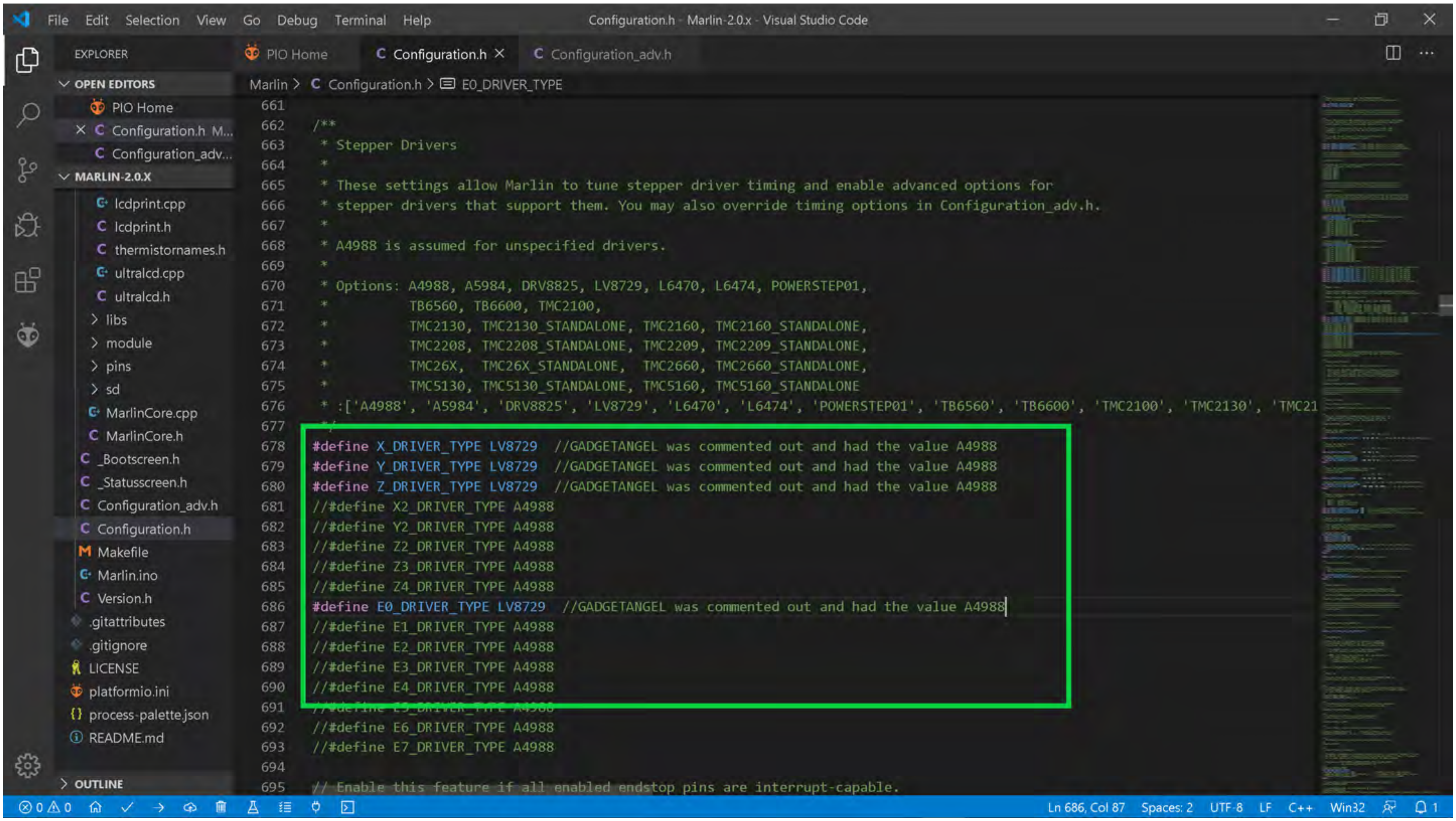


NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

The (latest release of) Marlin Setup for FYSETC S6128 V1.1 Drivers

NOTE: **Go to Appendix C,** and then come back here for the changes to Marlin for FYSETC S6128 stepper motor drivers.

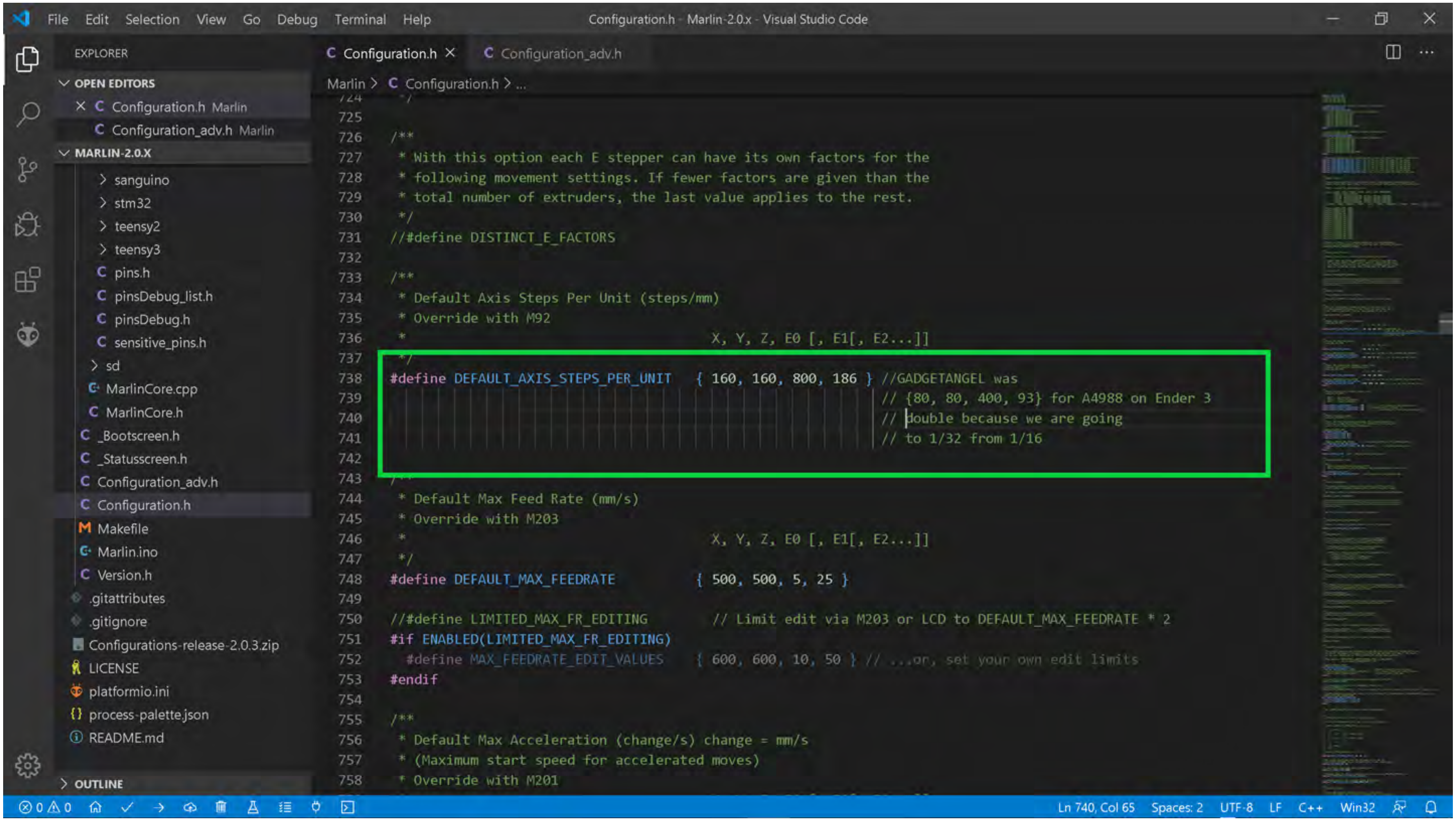
- Change the stepper motor drivers so that Marlin knows you are using FYSETC S6128 drivers. Change one line for each axis and one for each extruder you will be using. See the picture below for an example of how to use FYSETC S6128 drivers. When two "//" appear at the beginning of a line that means that line is commented out. To un-comment a line just remove the leading two forward slashes ("//").
- The S6128 is a drop in replacement for the LV8729. Since Marlin does not have an option for S6128 we will use the LV8729 option.



- Go to the next page.

The (latest release of) Marlin Setup for FYSETC S6128 V1.1 Drivers

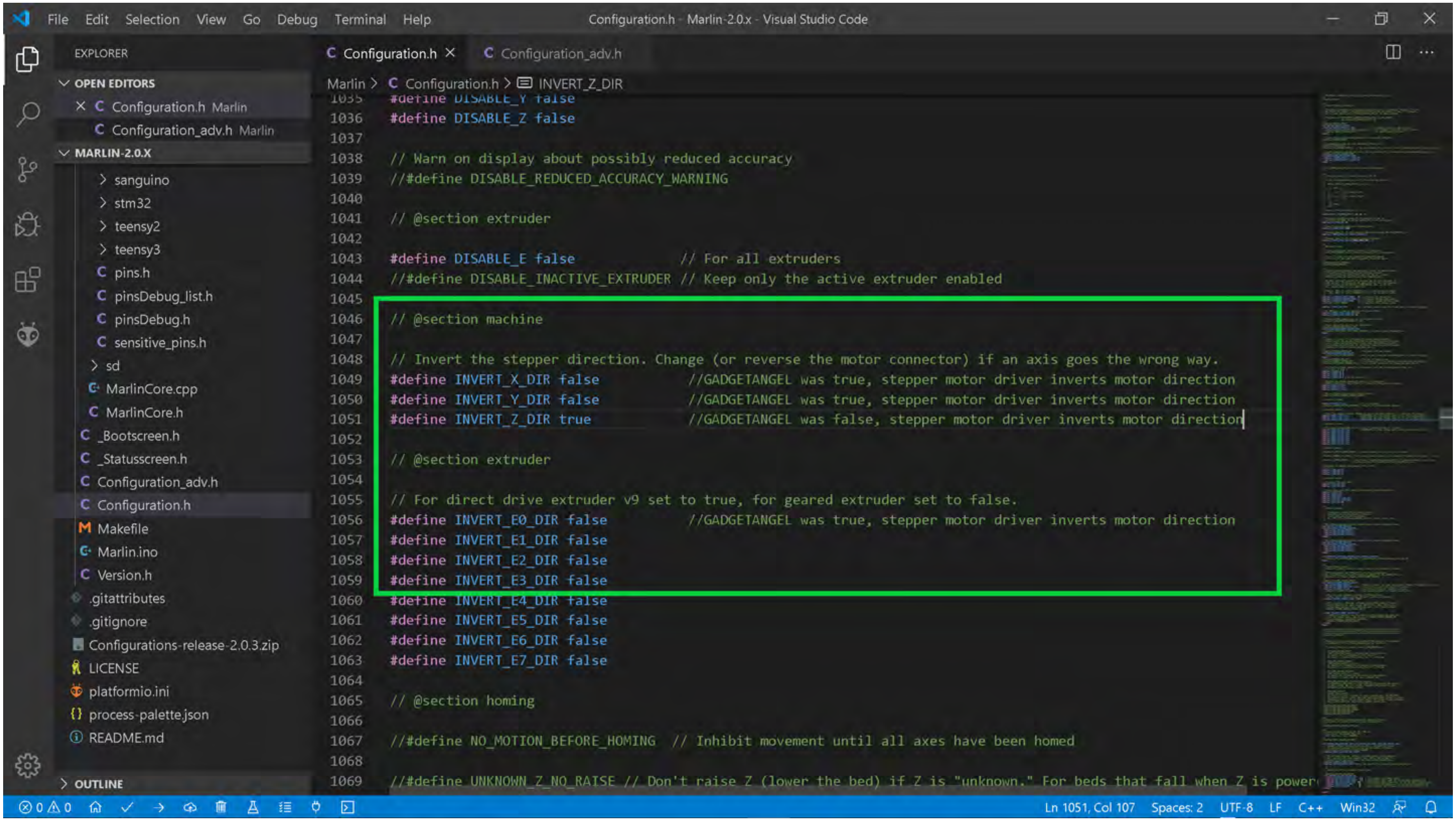
- We are changing from A4988 stepper motor drivers on the Ender 3 to S6128 stepper motor drivers for each axis and the extruder stepper motor driver, we will be going from 1/16 stepping to 1/32 stepping. So we are doubling our STEPS. Therefore, we must adjust our "DEFAULT_AXIS_STEPS_PER_UNIT" anytime our STEPS are NOT 1/16. So change "DEFAULT_AXIS_STEPS_PER_UNIT" to {160, 160, 800, 186}, as seen in the **GREEN** box below.



- Go to the next page.

The (latest release of) Marlin Setup for FYSETC S6128 V1.1 Drivers

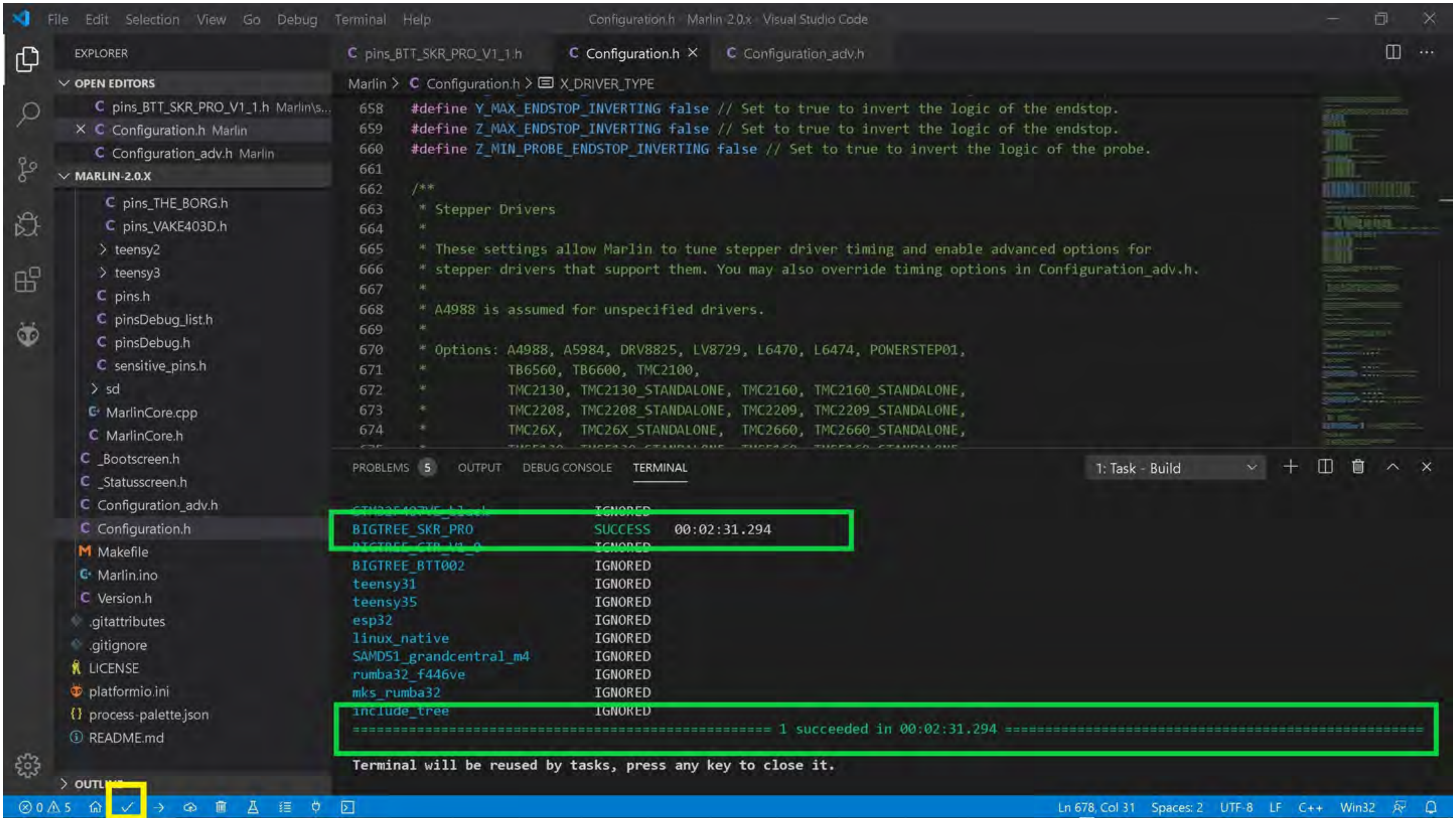
- Since the A4988 driver is what my Ender 3 used, but, now I want to use S6128 drivers, I must invert the stepper motor direction because the S6128 or LV8729 driver will turn the motors in the opposite direction than the A4988 driver's motor direction. So if the axis' setting you will be using the S6128 driver on was "true" change it to "false", as shown in the **GREEN** box below. If the setting was "false", now set it to "true", as show in the **GREEN** box below



- Go to the next page.

The (latest release of) Marlin Setup for FYSETC S6128 V1.1 Drivers

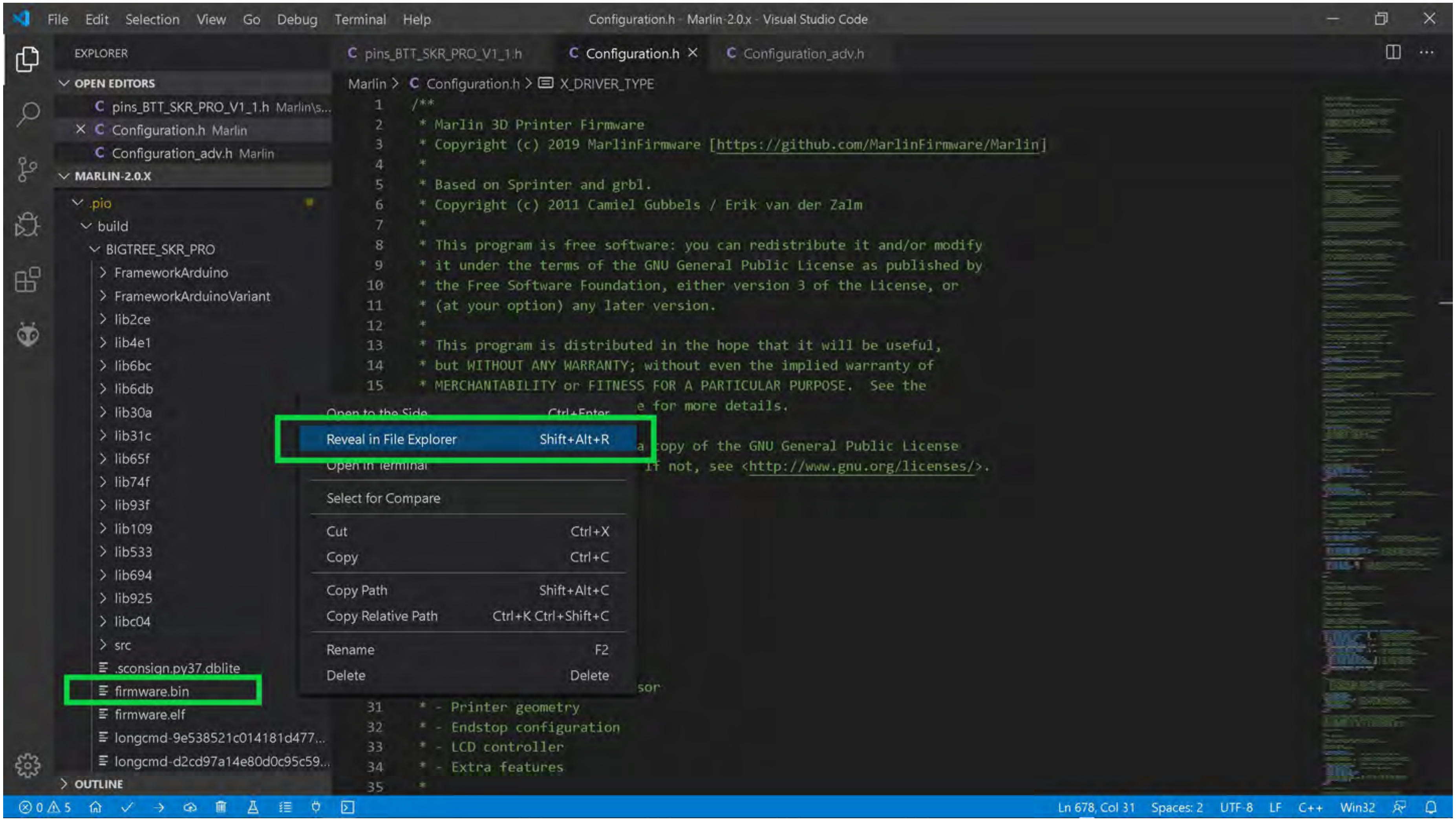
- The end of Marlin setup for FYSETC S6128 drivers. Now, compile your code by clicking on the check mark, as seen in the **YELLOW** box below. Once the firmware has compiled successfully you will see the following messages, as shown in the **GREEN** boxes below.



- For the SKR PRO V1.1 board you must save the "firmware.bin" file to the micro SD card then place the micro SD card into the micro SD card reader of the SKR PRO V1.1 board and turn on the power to your printer for the NEW firmware to be automatically applied to the SKR PRO MCU. To locate the "firmware.bin" file, and learn how to transfer it to your micro SD card go to the next page.

The (latest release of) Marlin Setup for FYSETC S6128 V1.1 Drivers

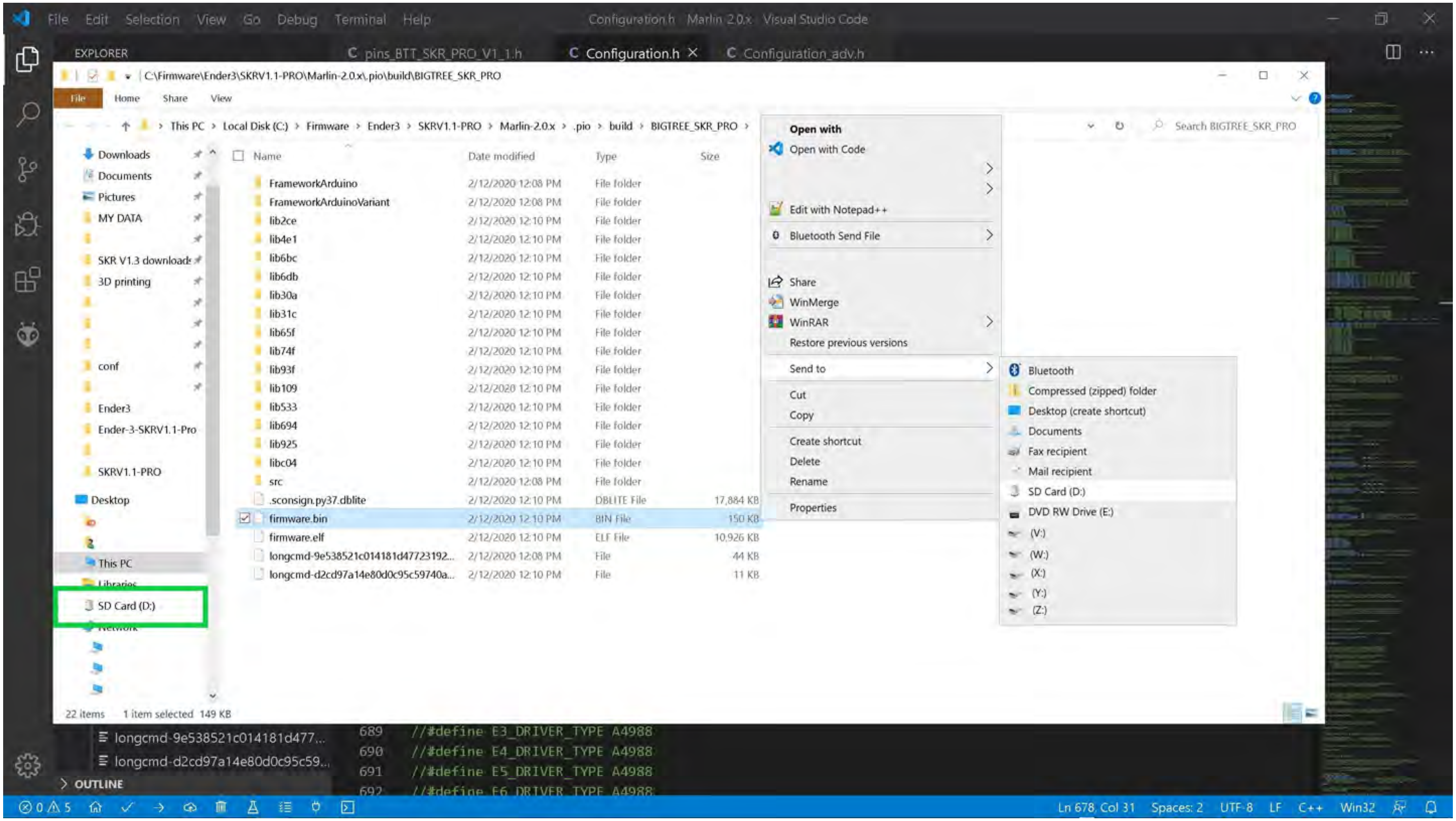
- The "firmware.bin" file can be found by using the left side of the VS code window and going to folder Marlin-2.0.X/.pio/build/BIGTREE_SKR_PRO/ and right clicking on the "firmware.bin" file name to bring up the content menu. Select "Reveal in File Explorer" to have your Window's machine open a file explorer window.



- Go to the next page.

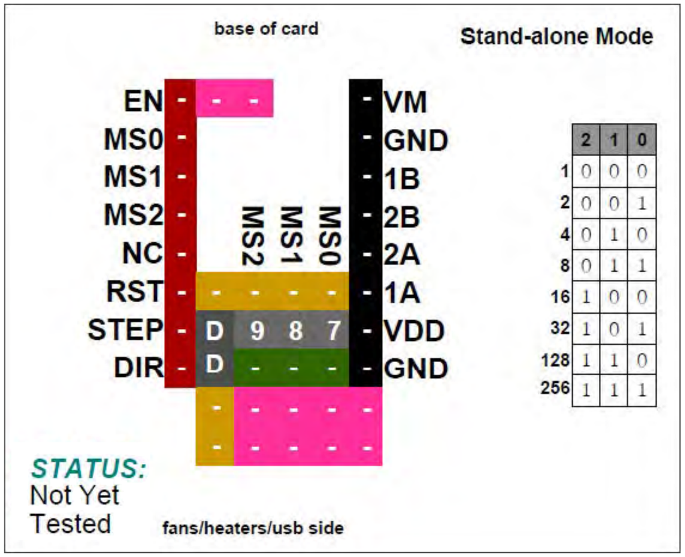
The (latest release of) Marlin Setup for FYSETC S6128 V1.1 Drivers

- Ensure your micro SD card, which came with your SKR PRO V1.1 board, with your SKR PRO V1.1 board is in a micro-to-SD-card adapter and plugged into your Window's SD card reader.
    - From the file explorer window right click on the "firmware.bin" file name and select "Send to" from the content menu. For my computer "D:" is my SD card reader. For your computer it may be a different drive letter. The "firmware.bin" file will be written to the SD card and renamed to "FIRMWARE.CUR".



- Right click on the SD card reader, as seen in the **GREEN** box above, and select "Eject" from the context menu. Take the micro SD card out of the adapter and place it into the micro SD card reader of the SKR PRO V1.1 board. Turn on the power to the printer (i.e. the SKR PRO V1.1 board) and the "FIRMWARE.CUR" file will be uploaded to the board. After the upload the firmware file will be renamed to "firmware.bin" and stored on the micro SD card.

FYSETC ST820



Note: **See the next page for information about location of the current sense resistors and how to set V_{ref} on the stepper motor driver board.**

Driver Chip	MS2	MS1	MS0	Microstep Resolution	Excitation Mode
FYSETC ST820 Maximum 256 Subdivision 45V DC 1.5A (peak)	Low	Low	Low	Full step	2 Phase
	Low	Low	High	Half step	1-2 Phase
	Low	High	Low	1/4 step	W1-2 Phase
	Low	High	High	1/8 step	2W1-2 Phase
	High	Low	Low	1/16 step	4W1-2 Phase
	High	Low	High	1/32 step	8W1-2 Phase
	High	High	Low	1/128 step	16W1-2 Phase
	High	High	High	1/256 step	32W1-2 Phase

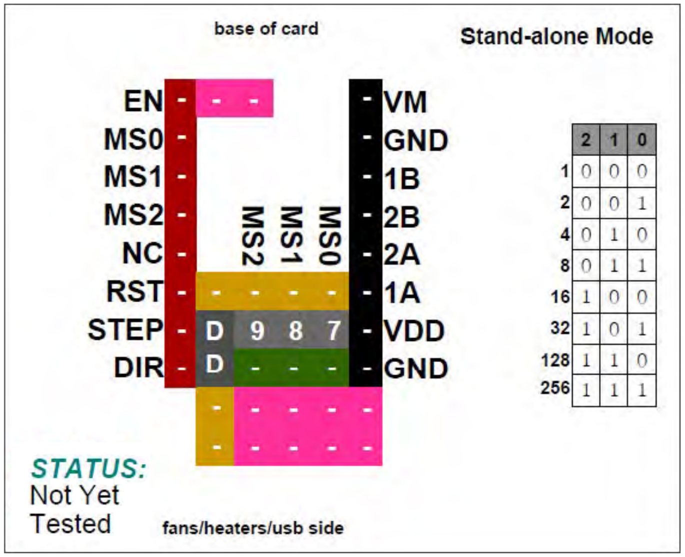
Driving Current Calculation Formula

V_{DD} = 3.3 V or 5 V DC

R_S (Typical Sense Resistor) = 0.15Ω

$$I_{MAX}=V_{ref}*\left(\frac{V_{DD}}{5}\right)*\frac{1}{R_S}$$

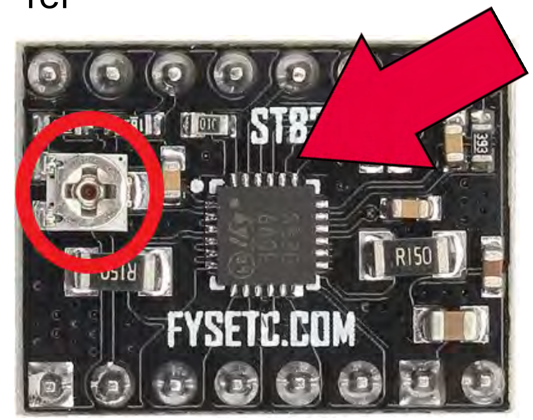
$$V_{ref}=I_{MAX}*\left(\frac{5}{V_{DD}}\right)*R_S$$



FYSETC ST820

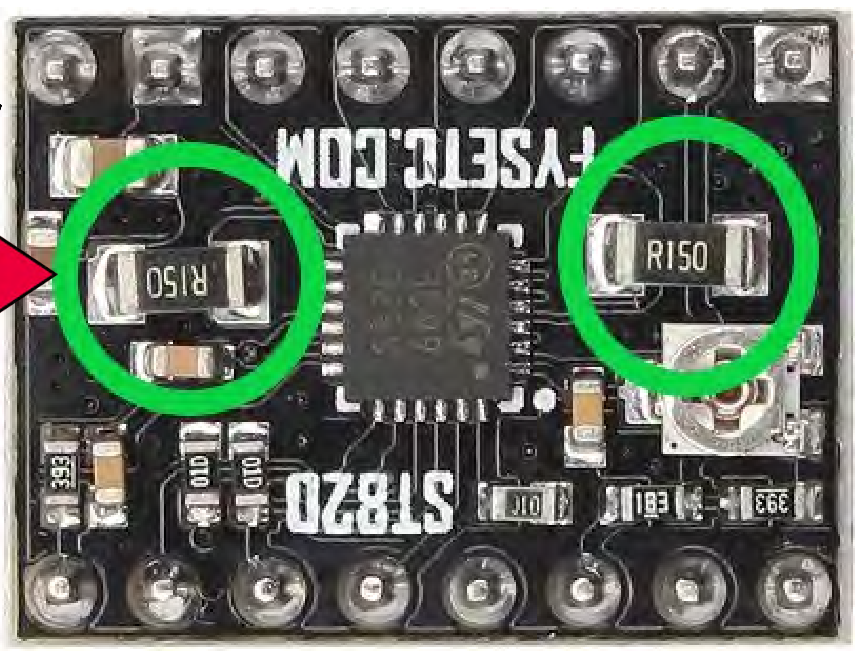
Note: Use 90% of the calculated V_{ref} when tuning the stepper driver board.

NOTE: Use the potentiometer (POT) on the top of the board to adjust your V_{ref}. See **Appendix A** for instructions on how to set the V_{ref} on a driver board.



Note: See this video about current sense resistors (R_s) and their possible locations: **https://youtu.be/8wk1elugv5A**

Note: Check your current sense resistors (R_s) values on the driver board, as shown in GREEN



R_s = R050 is 0.05 Ohms
R_s = R068 is 0.068 Ohms
R_s = R100 is 0.1 Ohms
R_s = R150 is 0.15 Ohms
R_s = R200 is 0.2 Ohms
R_s = R220 is 0.22 Ohms

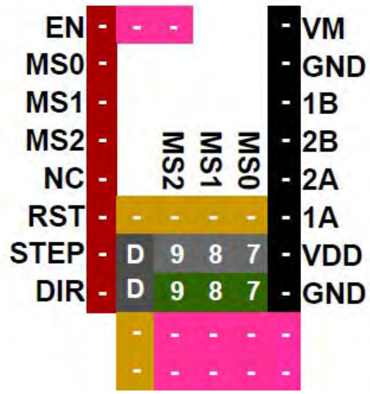
NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

FYSETC ST820

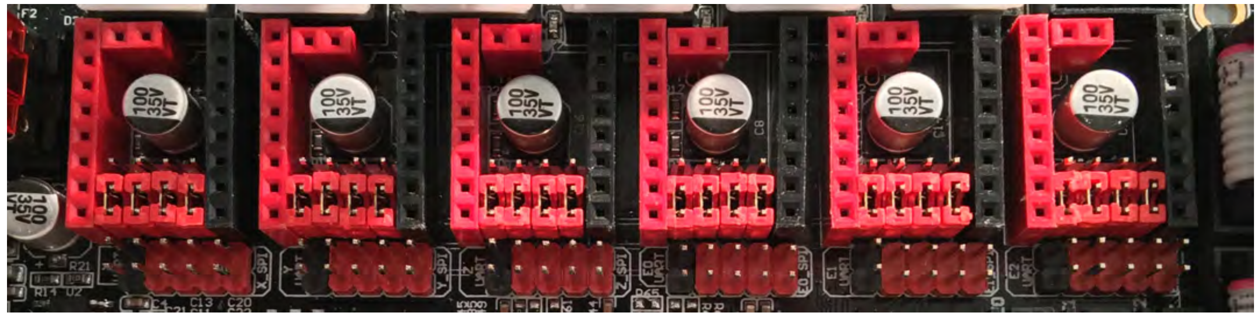
Stand-alone Mode

Important: This driver has special requirements in the Configuration and Configuration_adv.h. Also, this driver requires constant cooling the moment any motor is used, or it will switch on and off.

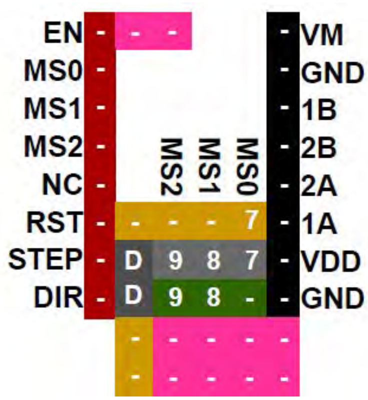
STEP



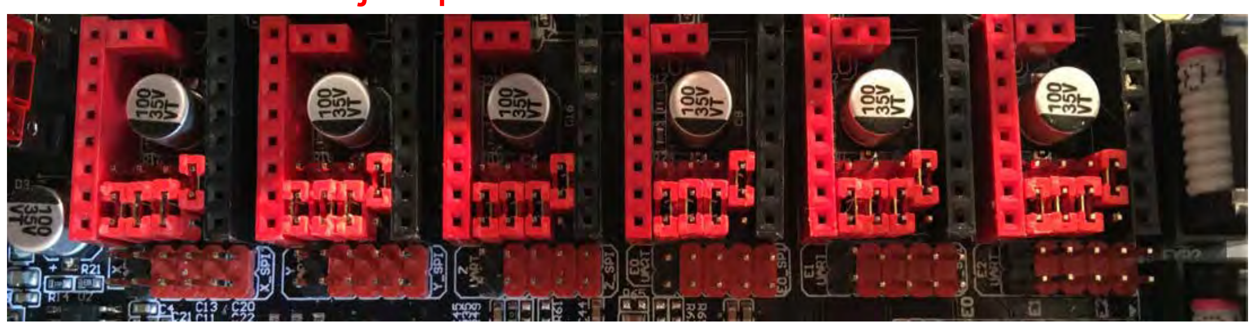
Note: The "D" jumper MUST be SET!



1 / 2



Note: The "D" jumper MUST be SET!



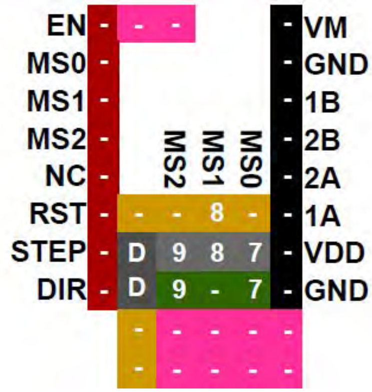
NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

FYSETC ST820

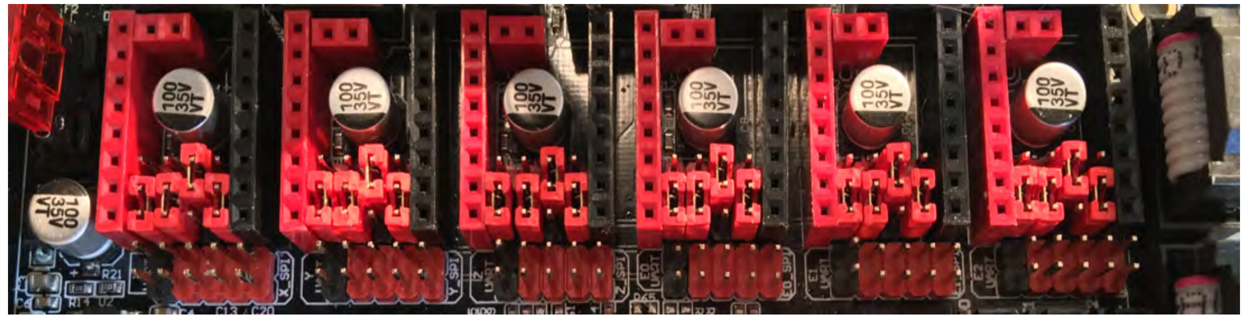
Stand-alone Mode

Important: This driver has special requirements in the Configuration and Configuration_adv.h. Also, this driver requires constant cooling the moment any motor is used, or it will switch on and off.

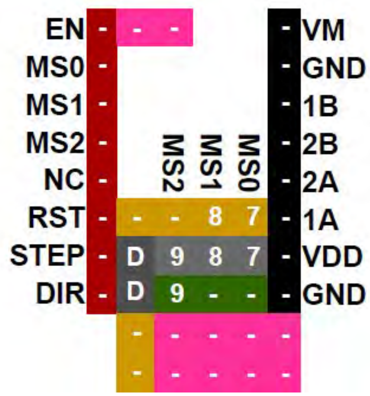
1 / 4



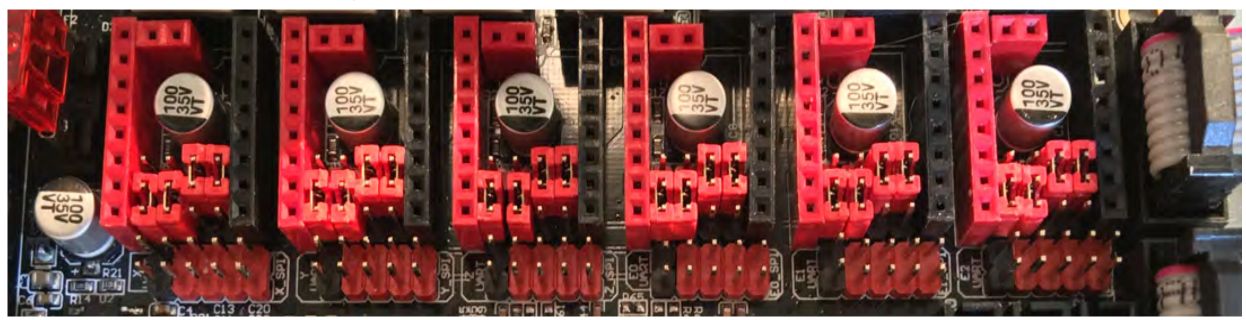
Note: The "D" jumper MUST be SET!



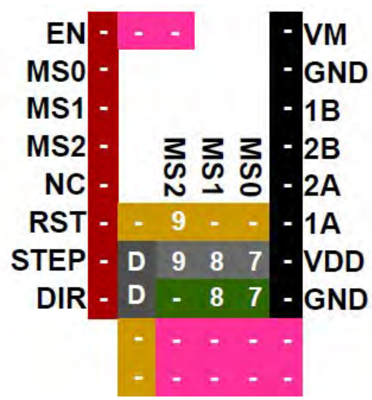
1 / 8



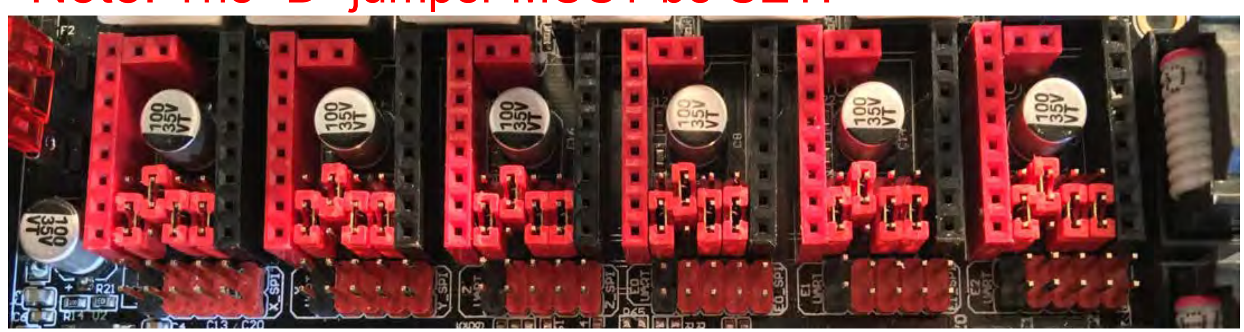
Note: The "D" jumper MUST be SET!



1 / 16



Note: The "D" jumper MUST be SET!



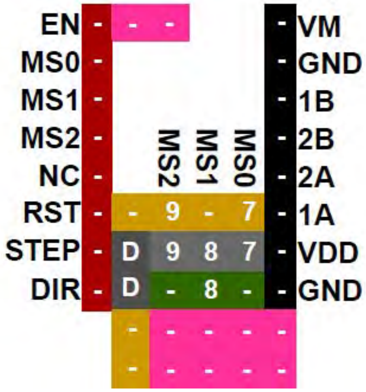
NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

FYSETC ST820

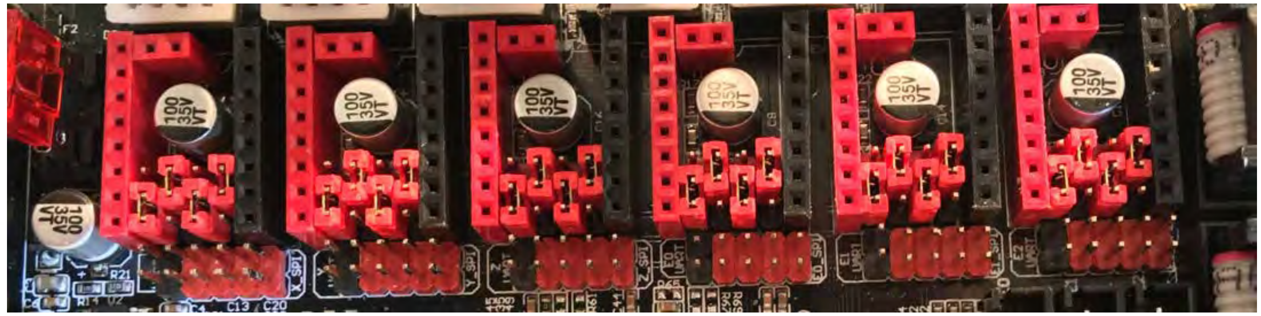
Stand-alone Mode

Important: This driver has special requirements in the Configuration and Configuration_adv.h. Also, this driver requires constant cooling the moment any motor is used, or it will switch on and off.

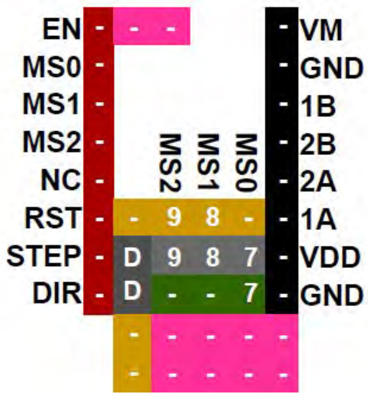
1 / 32



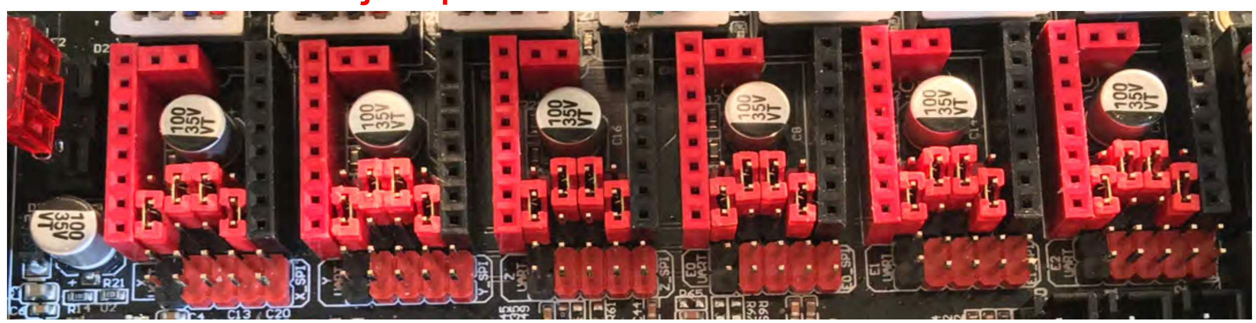
Note: The "D" jumper MUST be SET!



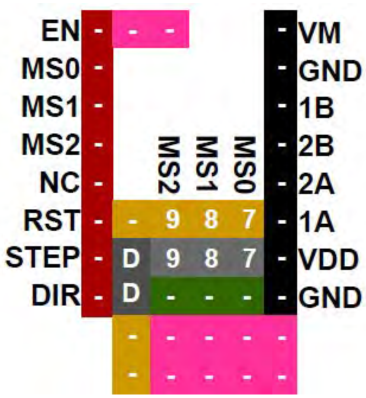
1 / 128



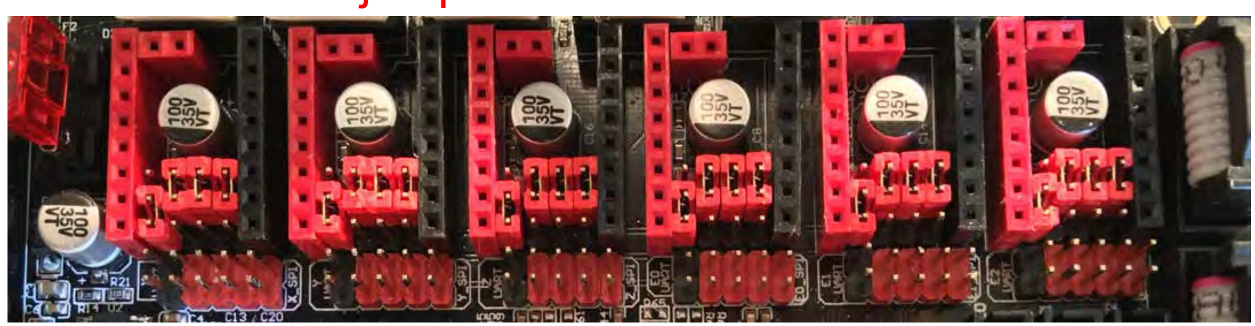
Note: The "D" jumper MUST be SET!



1 / 256



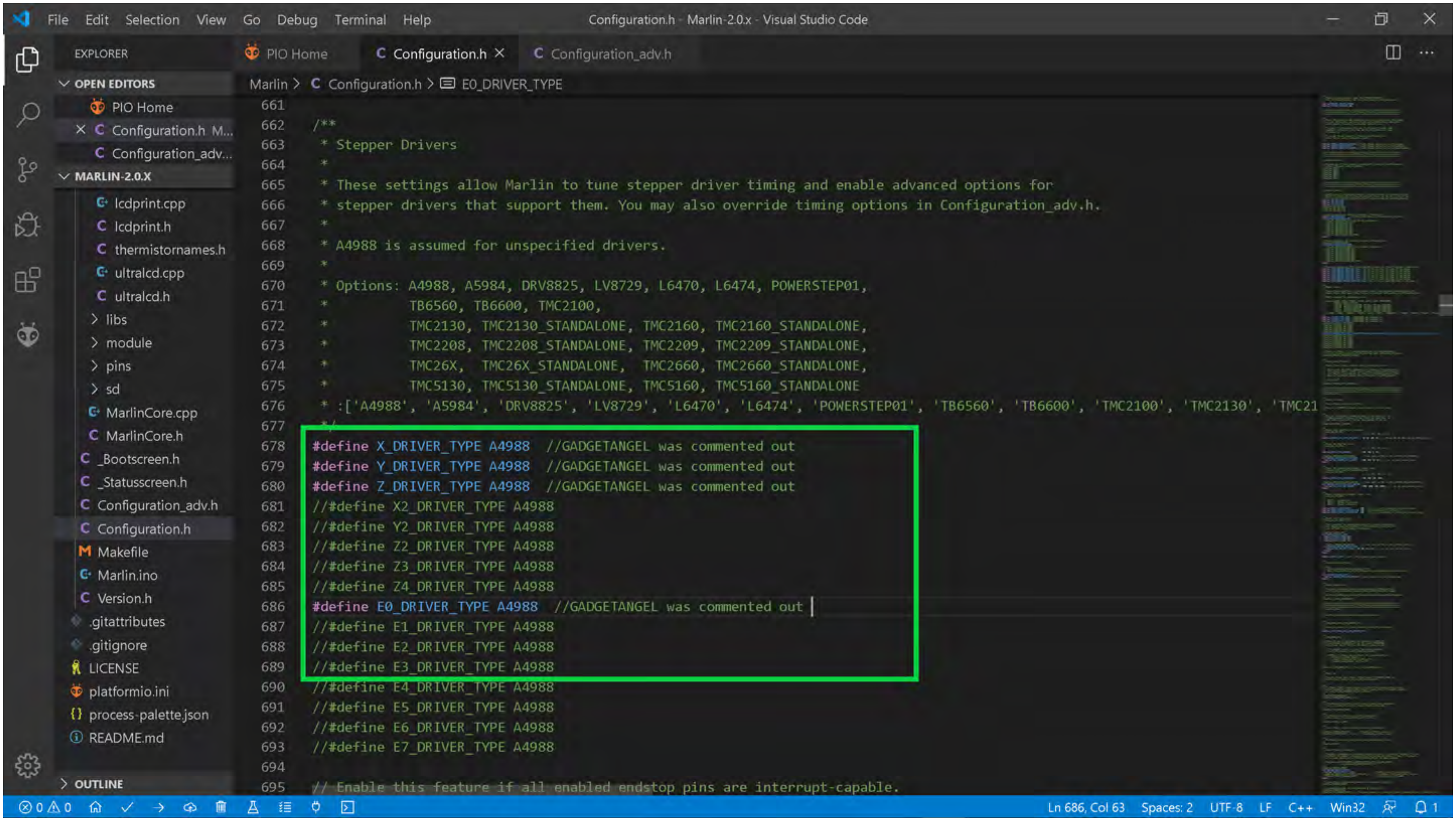
Note: The "D" jumper MUST be SET!



The (latest release of) Marlin Setup for FYSETC ST820 Drivers

NOTE: **Go to Appendix C,** and then come back here for the changes to Marlin for FYSETC ST820 stepper motor drivers.

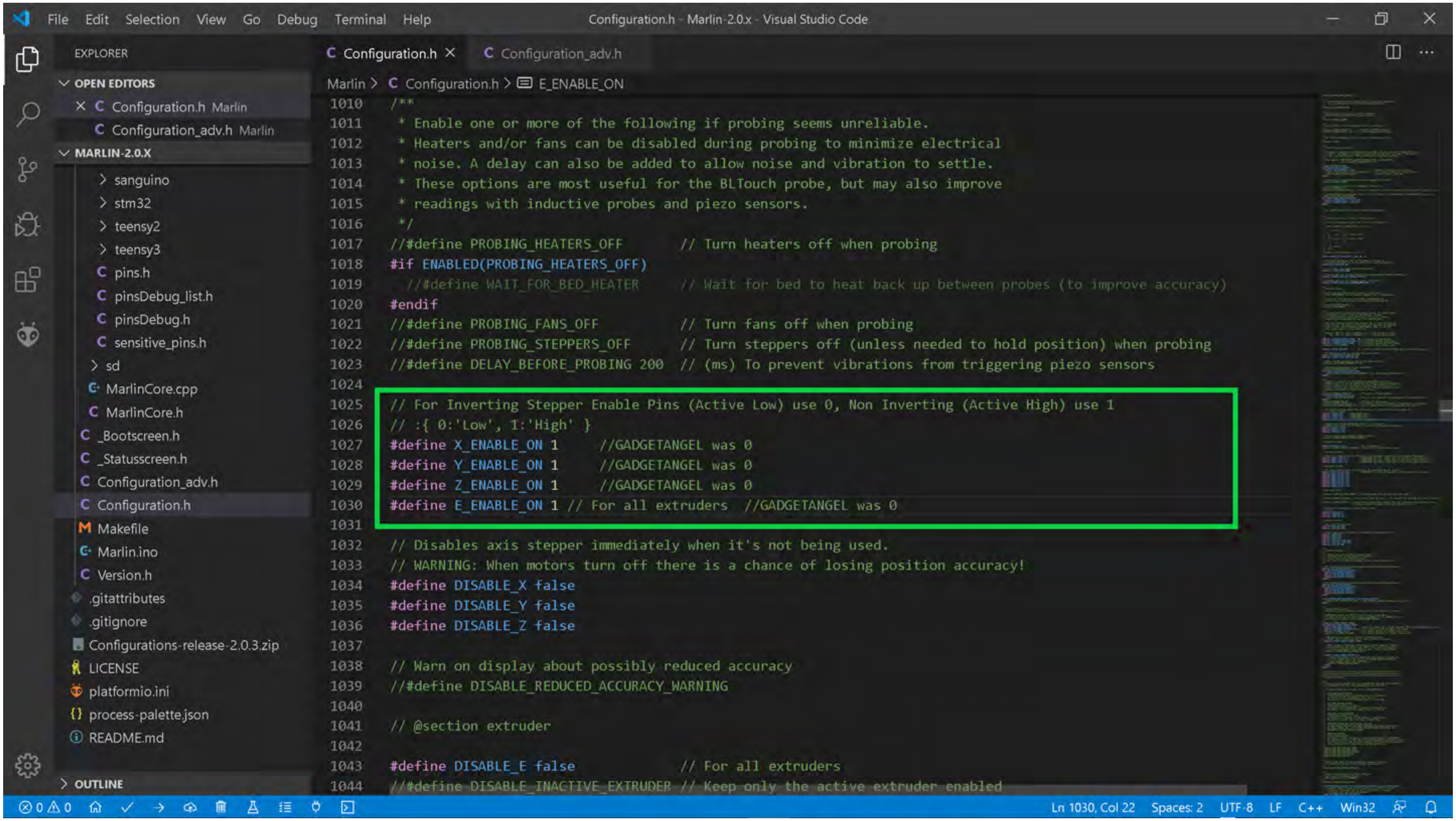
- Change the stepper motor drivers so that Marlin knows you are using FYSETC ST820 drivers. Change one line for each axis and one for each extruder you will be using. See the picture below for an example of how to use FYSETC ST820 drivers. When two "//" appear at the beginning of a line that means that line is commented out. To un-comment a line just remove the leading two forward slashes ("//").
- The ST820 is a drop in replacement for the A4988. Since Marlin does not have an option for ST820 we will use the A4988 option.



- Go to the next page.

The (latest release of) Marlin Setup for FYSETC ST820 Drivers

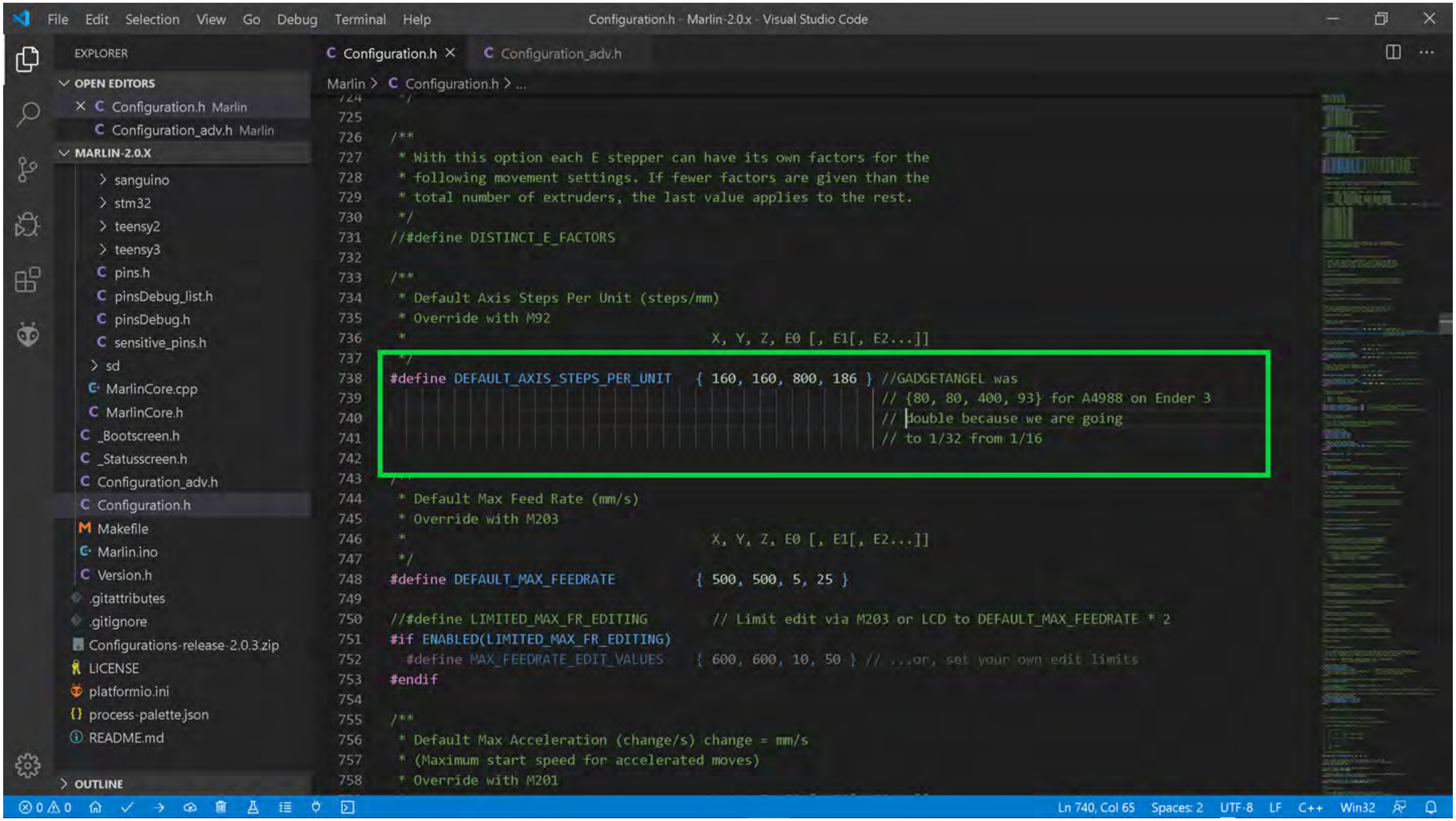
- In the Marlin firmware, the ST820 drivers needs an ACTIVE HIGH for the stepper motor driver's enable pin, so set "X_ENABLE_ON" to 1, "Y_ENABLE_ON" to 1, "Z_ENABLE_ON" to 1 and "E_ENABLE_ON" to 1, as seen in the **GREEN** box below.



- Go to the next page.

The (latest release of) Marlin Setup for FYSETC ST820 Drivers

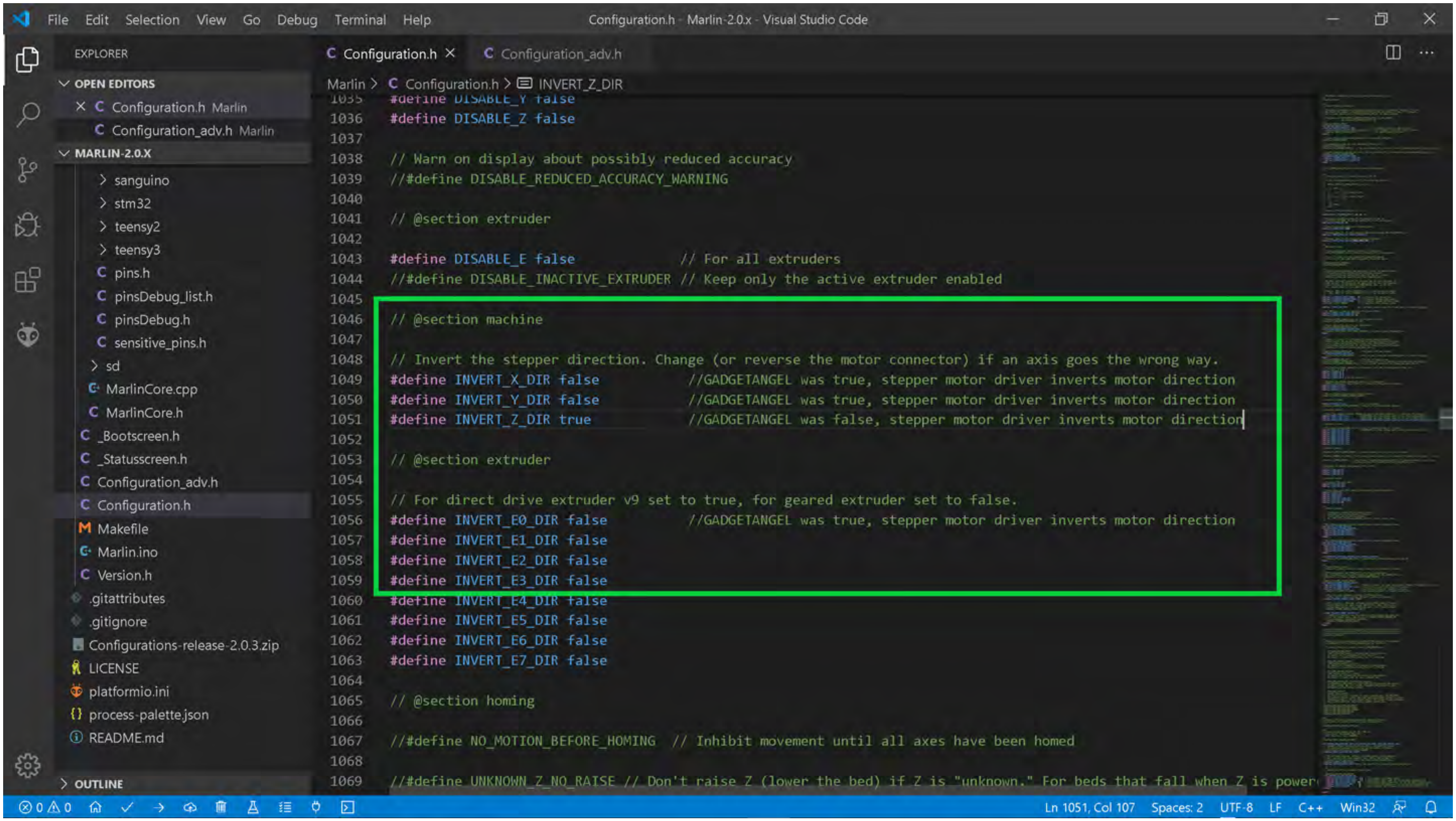
- We are changing from A4988 stepper motor drivers on the Ender 3 to FYSETC ST820 stepper motor drivers for each axis and the extruder stepper motor driver, we will be going from 1/16 stepping to 1/32 stepping. So we are doubling our STEPS. Therefore, we must adjust our "DEFAULT_AXIS_STEPS_PER_UNIT" anytime our STEPS are NOT 1/16. So change "DEFAULT_AXIS_STEPS_PER_UNIT" to {160, 160, 800, 186}, as seen in the **GREEN** box below.



- Go to the next page.

The (latest release of) Marlin Setup for FYSETC ST820 Drivers

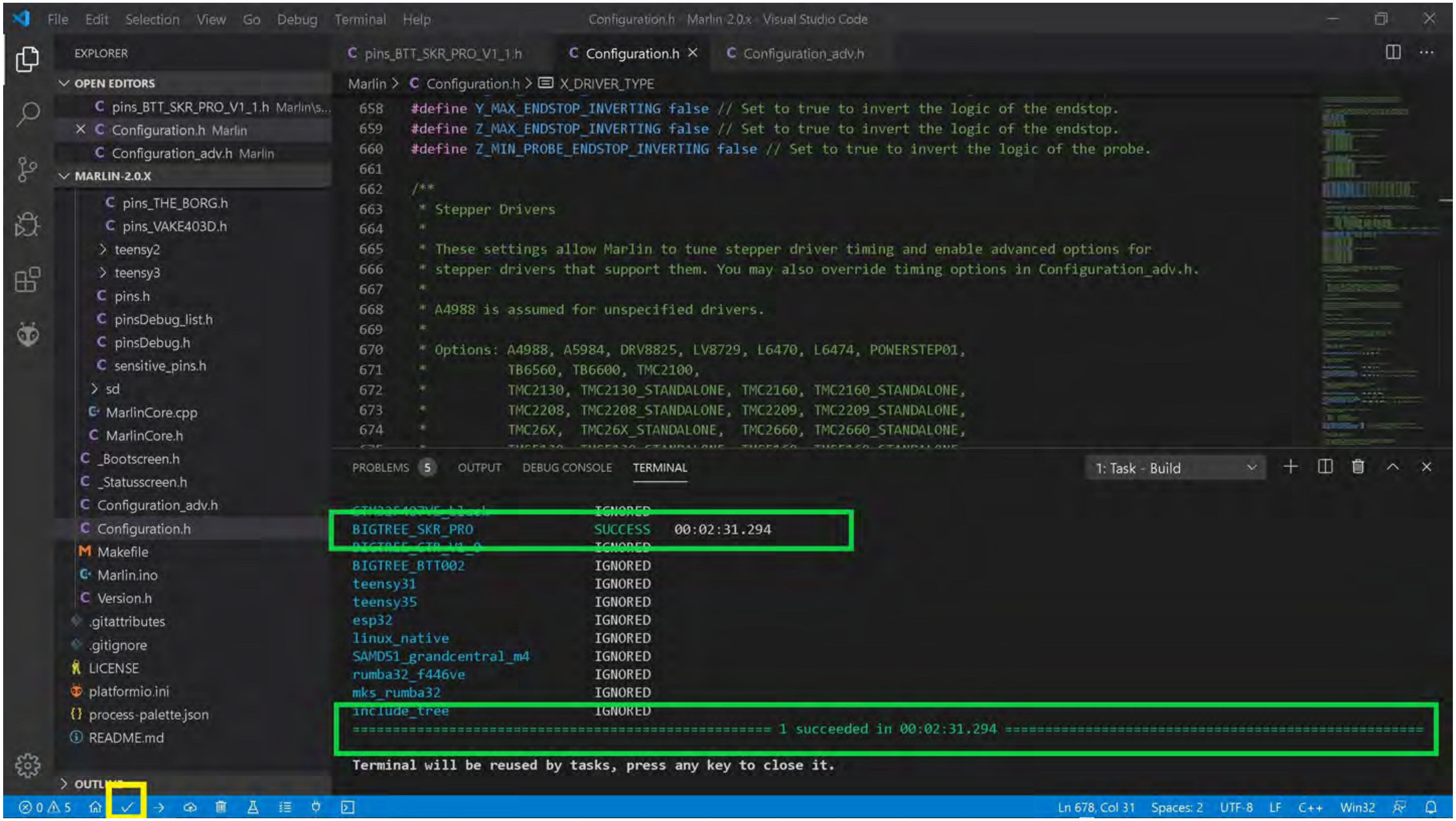
- Optional Step: I found conflicting information on the ST820 driver. Some sources say you will need to change the motor direction others say you may not. So I provide the below information in case you do need to change the stepper motor direction. If you prefer to change the motor direction with wiring instead of the Marlin firmware, here is a link on how to change the motor direction via the wiring (look for section labeled "Motor moving the wrong direction") **https://reprap.org/wiki/Stepper_wiring**. Other people prefer to change the motor direction in the Marlin firmware. So if you want or need to change the motor direction in Marlin, then if the axis' setting you will be using the ST820 driver on was "true" change it to "false", as shown in the **GREEN** box below. If the setting was "false", now set it to "true", as shown in the **GREEN** box below



- Go to the next page.

The (latest release of) Marlin Setup for FYSETC ST820 Drivers

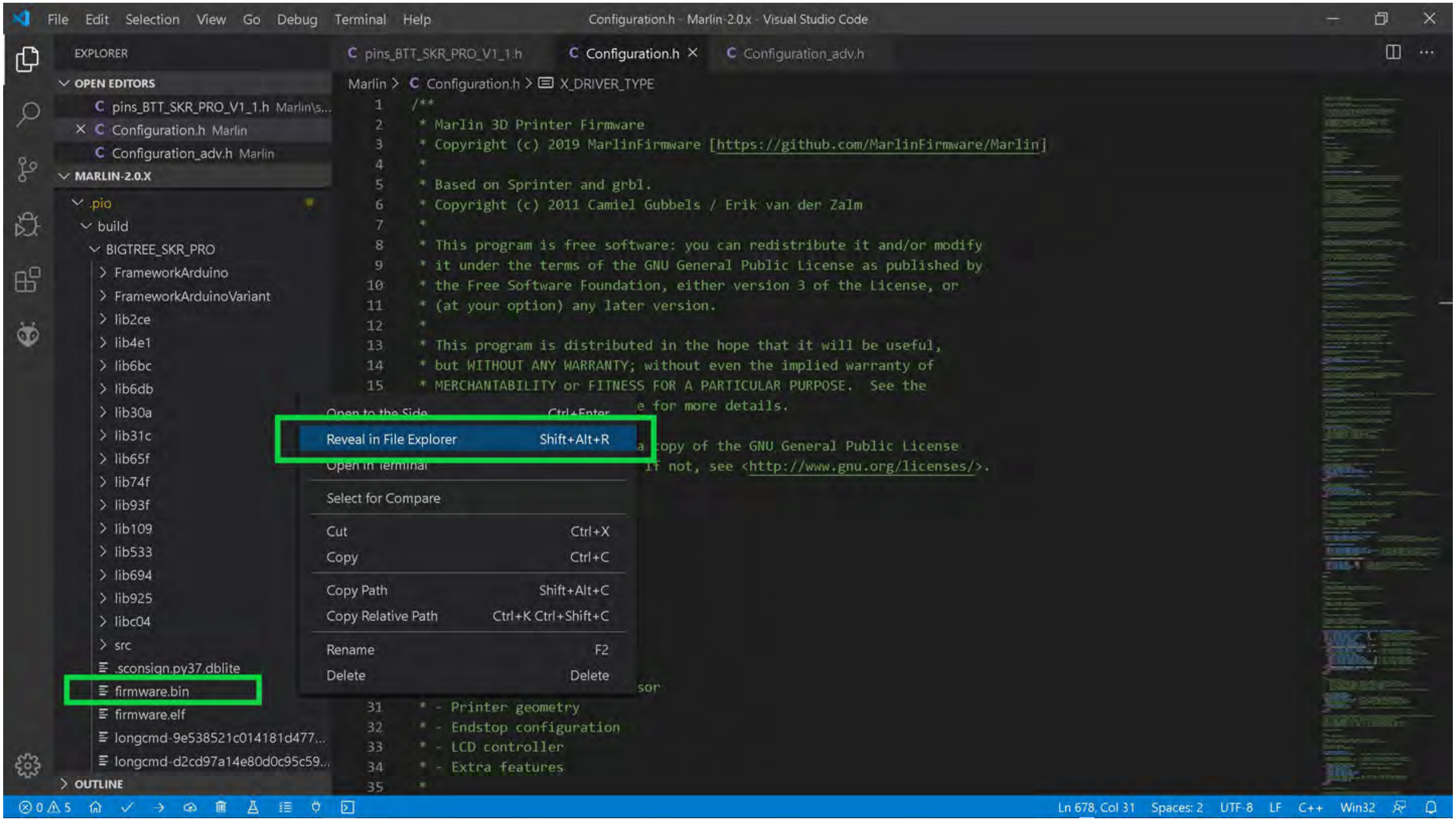
- The end of Marlin setup for FYSETC ST820 drivers. Now, compile your code by clicking on the check mark, as seen in the **YELLOW** box below. Once the firmware has compiled successfully you will see the following messages, as shown in the **GREEN** boxes below.



- For the SKR PRO V1.1 board you must save the "firmware.bin" file to the micro SD card then place the micro SD card into the micro SD card reader of the SKR PRO V1.1 board and turn on the power to your printer for the NEW firmware to be automatically applied to the SKR PRO MCU. To locate the "firmware.bin" file, and learn how to transfer it to your micro SD card go to the next page.

The (latest release of) Marlin Setup for FYSETC ST820 Drivers

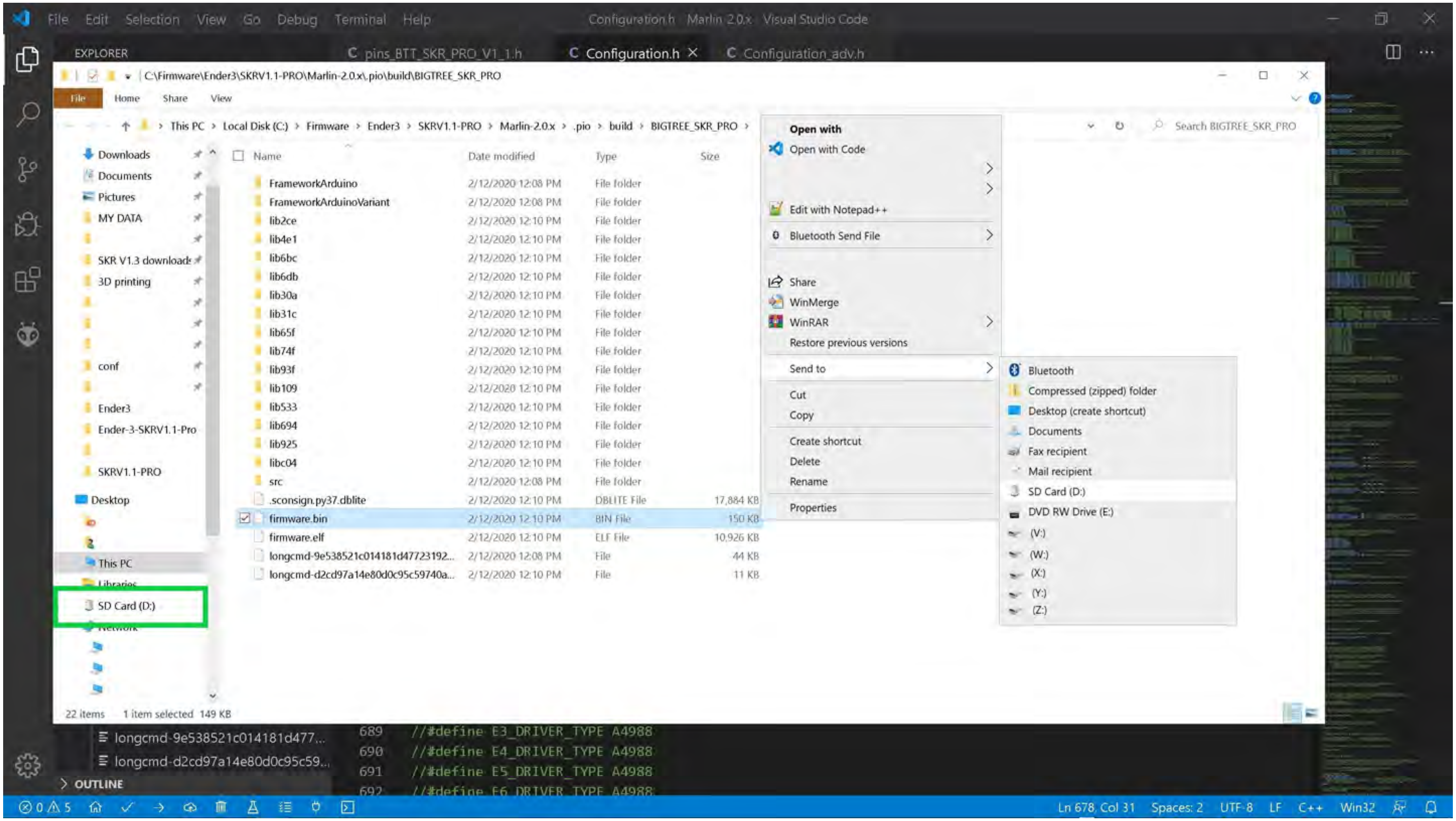
- The "firmware.bin" file can be found by using the left side of the VS code window and going to folder Marlin-2.0.X/.pio/build/ BIGTREE_SKR_PRO/ and right clicking on the "firmware.bin" file name to bring up the content menu. Select "Reveal in File Explorer" to have your Window's machine open a file explorer window.



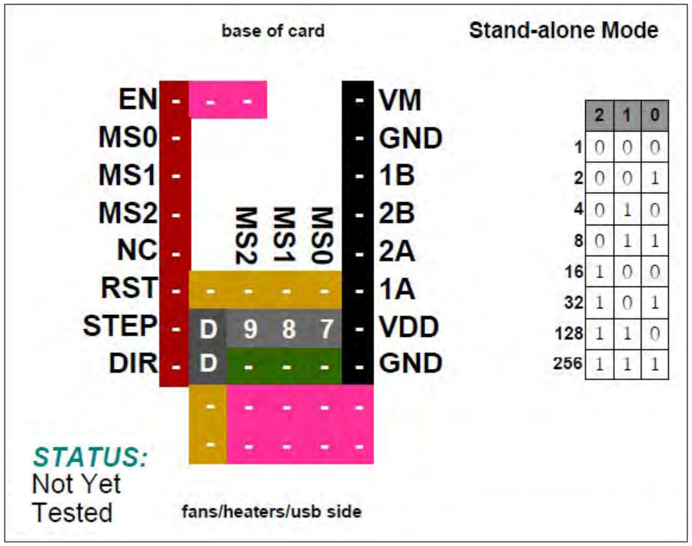
- Go to the next page.

The (latest release of) Marlin Setup for FYSETC ST820 Drivers

- Ensure your micro SD card, which came with your SKR PRO V1.1 board, is in a micro-to-SD-card adapter and plugged into your Window's SD card reader.
    - From the file explorer window right click on the "firmware.bin" file name and select "Send to" from the content menu. For my computer "D:" is my SD card reader. For your computer it may be a different drive letter. The "firmware.bin" file will be written to the SD card and renamed to "FIRMWARE.CUR".



- Right click on the SD card reader, as seen in the **GREEN** box above, and select "Eject" from the context menu. Take the micro SD card out of the adapter and place it into the micro SD card reader of the SKR PRO V1.1 board. Turn on the power to the printer (i.e. the SKR PRO V1.1 board) and the "FIRMWARE.CUR" file will be uploaded to the board. After the upload the firmware file will be renamed to "firmware.bin" and stored on the micro SD card.



BIQU ST820

Note: See the next page for information about location of the current sense resistors and how to set V_{ref} on the stepper motor driver board.

Driver Chip	MS2	MS1	MS0	Microstep Resolution	Excitation Mode
BIQU® ST820	Low	Low	Low	Full step	2 Phase
Maximum 256 Subdivision	Low	Low	High	Half step	1-2 Phase
45V DC	Low	High	Low	1/4 step	W1-2 Phase
1.5A (peak)	Low	High	High	1/8 step	2W1-2 Phase
	High	Low	Low	1/16 step	4W1-2 Phase
	High	Low	High	1/32 step	8W1-2 Phase
	High	High	Low	1/128 step	16W1-2 Phase
	High	High	High	1/256 step	32W1-2 Phase

Driving Current Calculation Formula

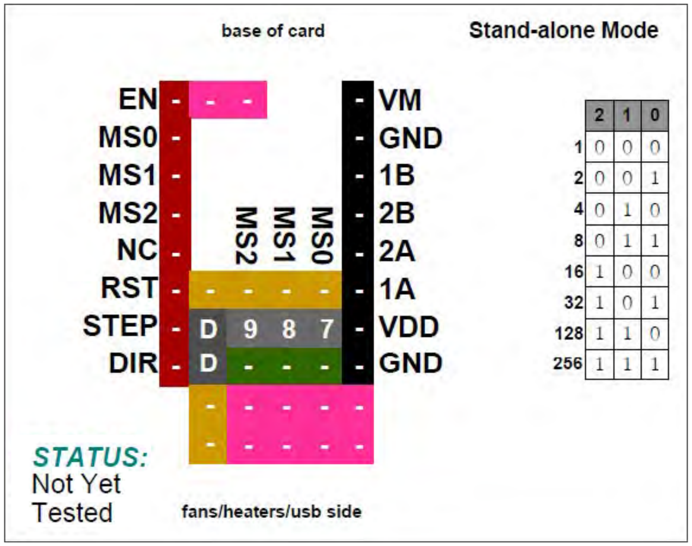
V_{DD} = 3.3 V or 5 V DC

R_S(Typical Sense Resistor)= 0.15Ω

$$I_{MAX}=V_{ref}*\left(\frac{V_{DD}}{5}\right)*\frac{1}{R_S}$$

$$V_{ref}=I_{MAX}*\left(\frac{5}{V_{DD}}\right)*R_S$$

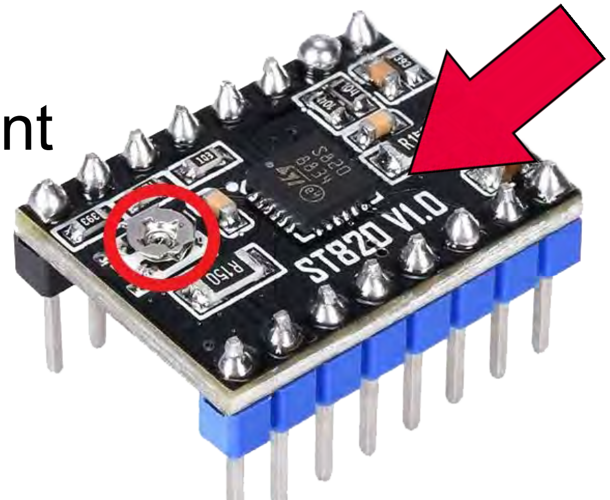
NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.



BIQU ST820

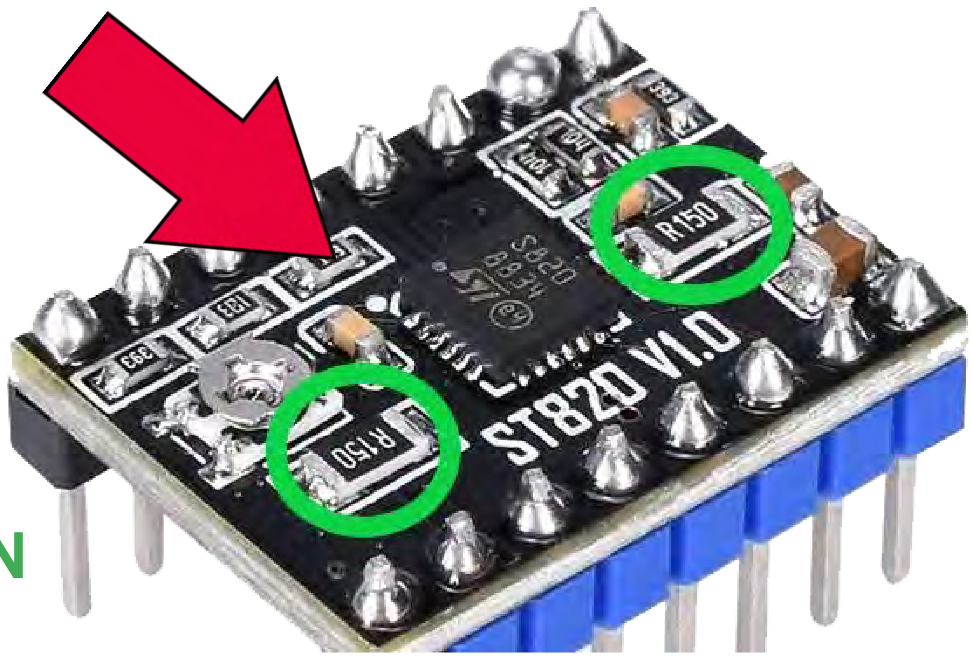
Note: Use 90% of the calculated V_{ref} when tuning the stepper driver board.

NOTE: Use the potentiometer (POT) on the top of the board to adjust your V_{ref}. See **Appendix A** for instructions on how to set the V_{ref} on a driver board.



Note: Check your current sense resistors (R_s) values on the driver board, as shown in GREEN

Note: See this video about current sense resistors (R_s) and their possible locations:
https://youtu.be/8wk1elugv5A



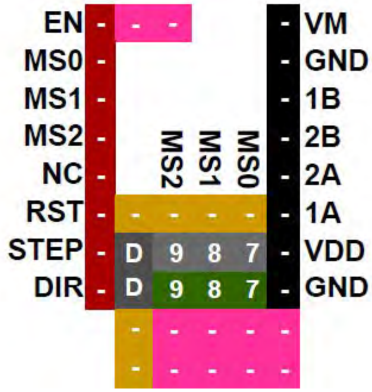
R_s = R050 is 0.05 Ohms
R_s = R068 is 0.068 Ohms
R_s = R100 is 0.1 Ohms
R_s = R150 is 0.15 Ohms
R_s = R200 is 0.2 Ohms
R_s = R220 is 0.22 Ohms

BIQU ST820

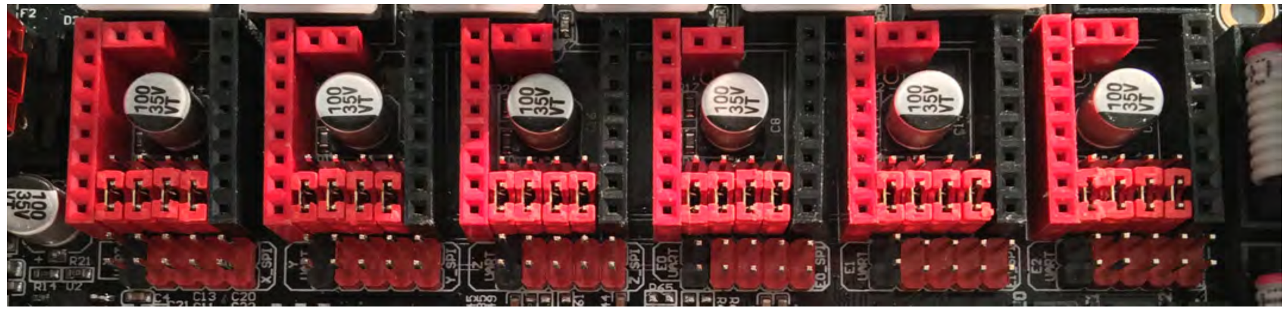
Stand-alone Mode

Important: This driver has special requirements in the Configuration and Configuration_adv.h. Also, this driver requires constant cooling the moment any motor is used, or it will switch on and off.

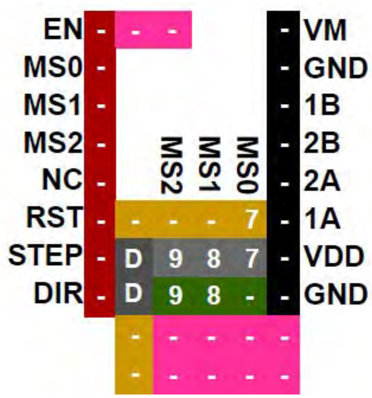
STEP



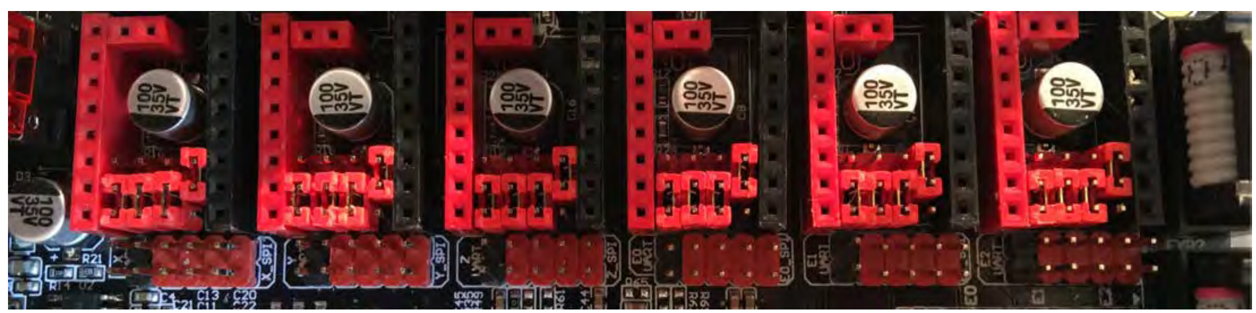
Note: The "D" jumper MUST be SET!



1 / 2



Note: The "D" jumper MUST be SET!



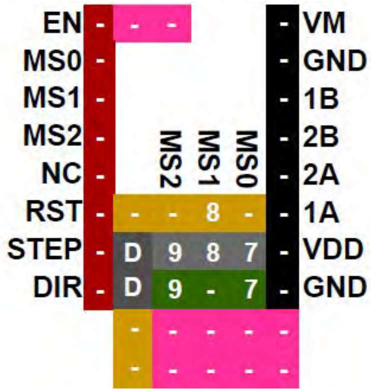
NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

BIQU ST820

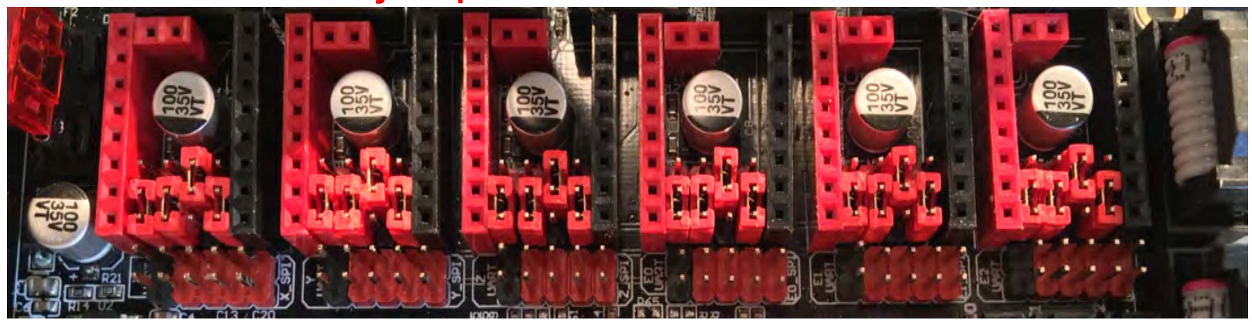
Stand-alone Mode

Important: This driver has special requirements in the Configuration and Configuration_adv.h. Also, this driver requires constant cooling the moment any motor is used, or it will switch on and off.

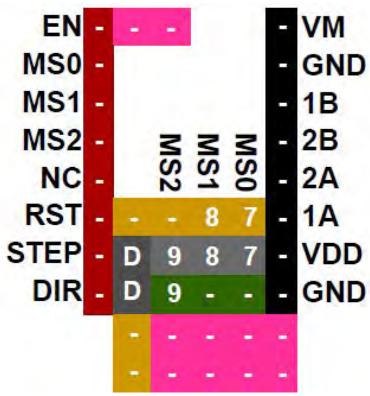
1 / 4



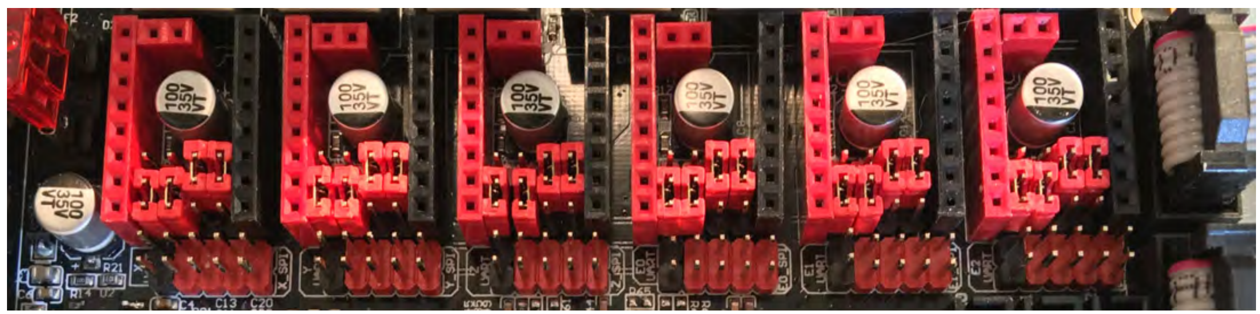
Note: The "D" jumper MUST be SET!



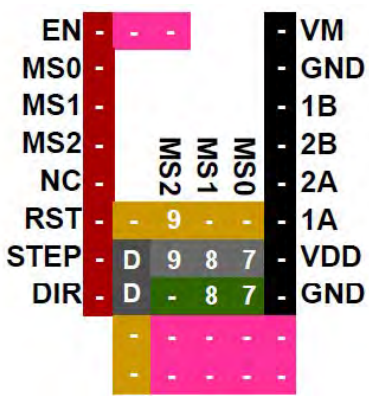
1 / 8



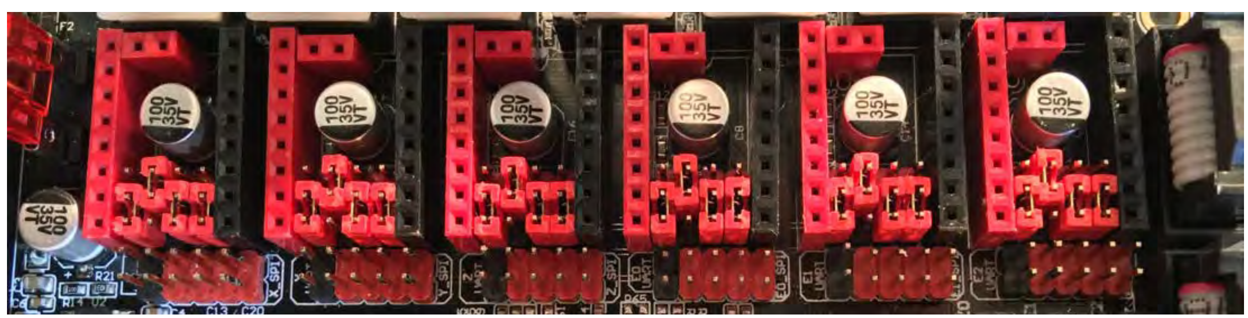
Note: The "D" jumper MUST be SET!



1 / 16



Note: The "D" jumper MUST be SET!



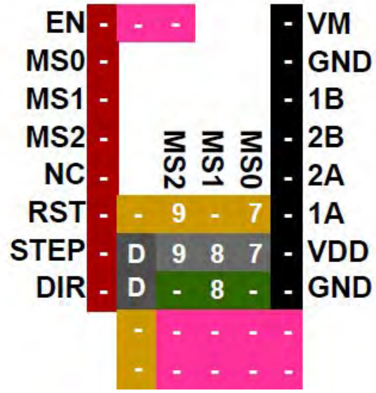
NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

BIQU ST820

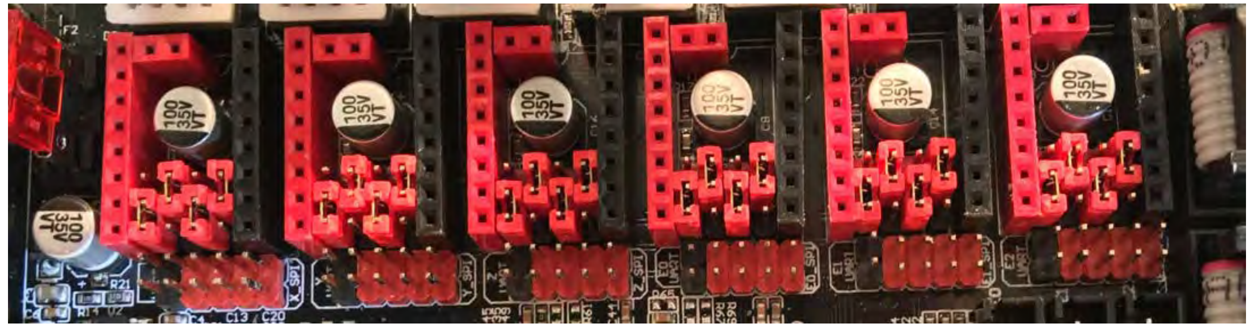
Stand-alone Mode

Important: This driver has special requirements in the Configuration and Configuration_adv.h. Also, this driver requires constant cooling the moment any motor is used, or it will switch on and off.

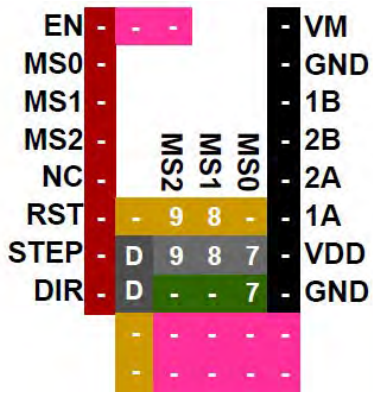
1 / 32



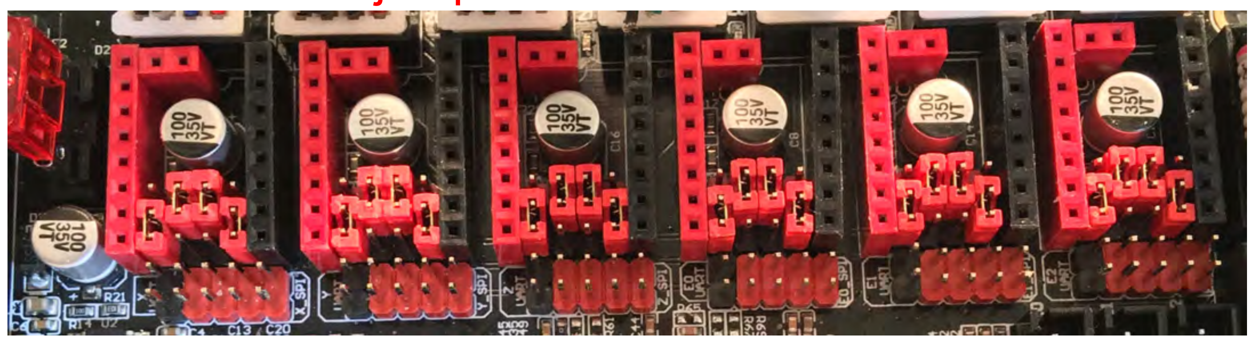
Note: The "D" jumper MUST be SET!



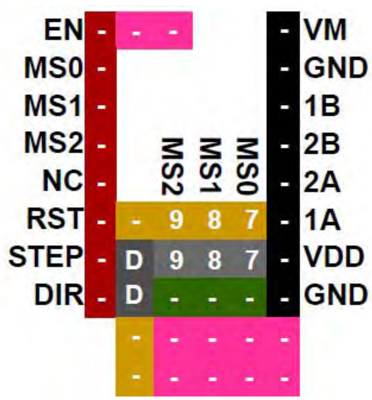
1 / 128



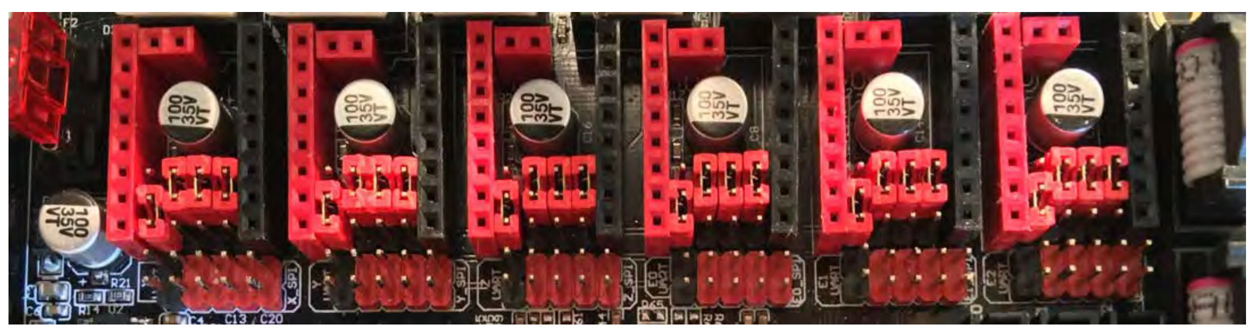
Note: The "D" jumper MUST be SET!



1 / 256



Note: The "D" jumper MUST be SET!

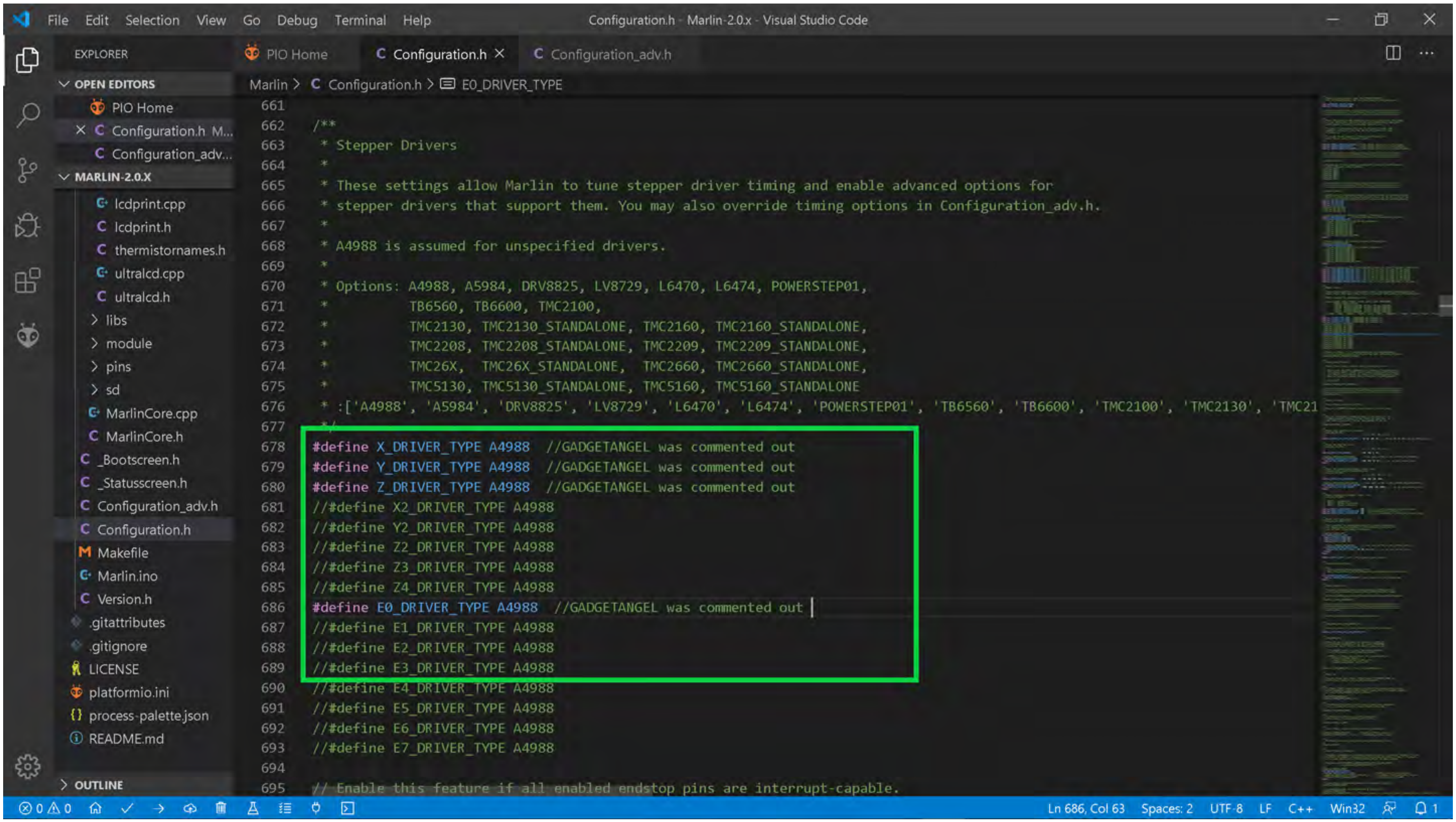


NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

The (latest release of) Marlin Setup for BIQU ST820 Drivers

NOTE: **Go to Appendix C,** and then come back here for the changes to Marlin for BIQU ST820 stepper motor drivers.

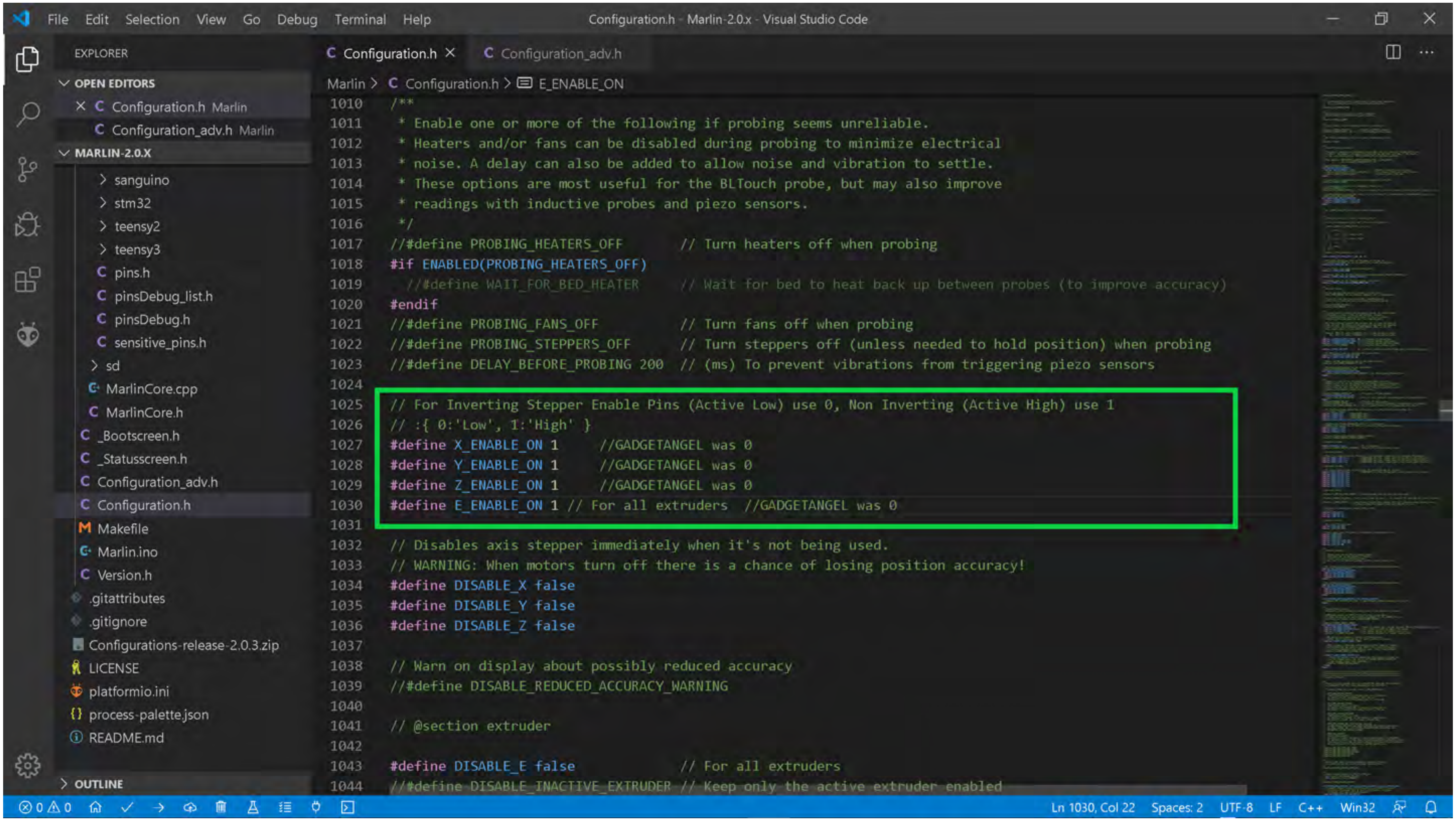
- Change the stepper motor drivers so that Marlin knows you are using BIQU ST820 drivers. Change one line for each axis and one for each extruder you will be using. See the picture below for an example of how to use BIQU ST820 drivers. When two "//" appear at the beginning of a line that means that line is commented out. To un-comment a line just remove the leading two forward slashes ("//").
- The ST820 is a drop in replacement for the A4988. Since Marlin does not have an option for ST820 we will use the A4988 option.



- Go to the next page.

The (latest release of) Marlin Setup for BIQU ST820 Drivers

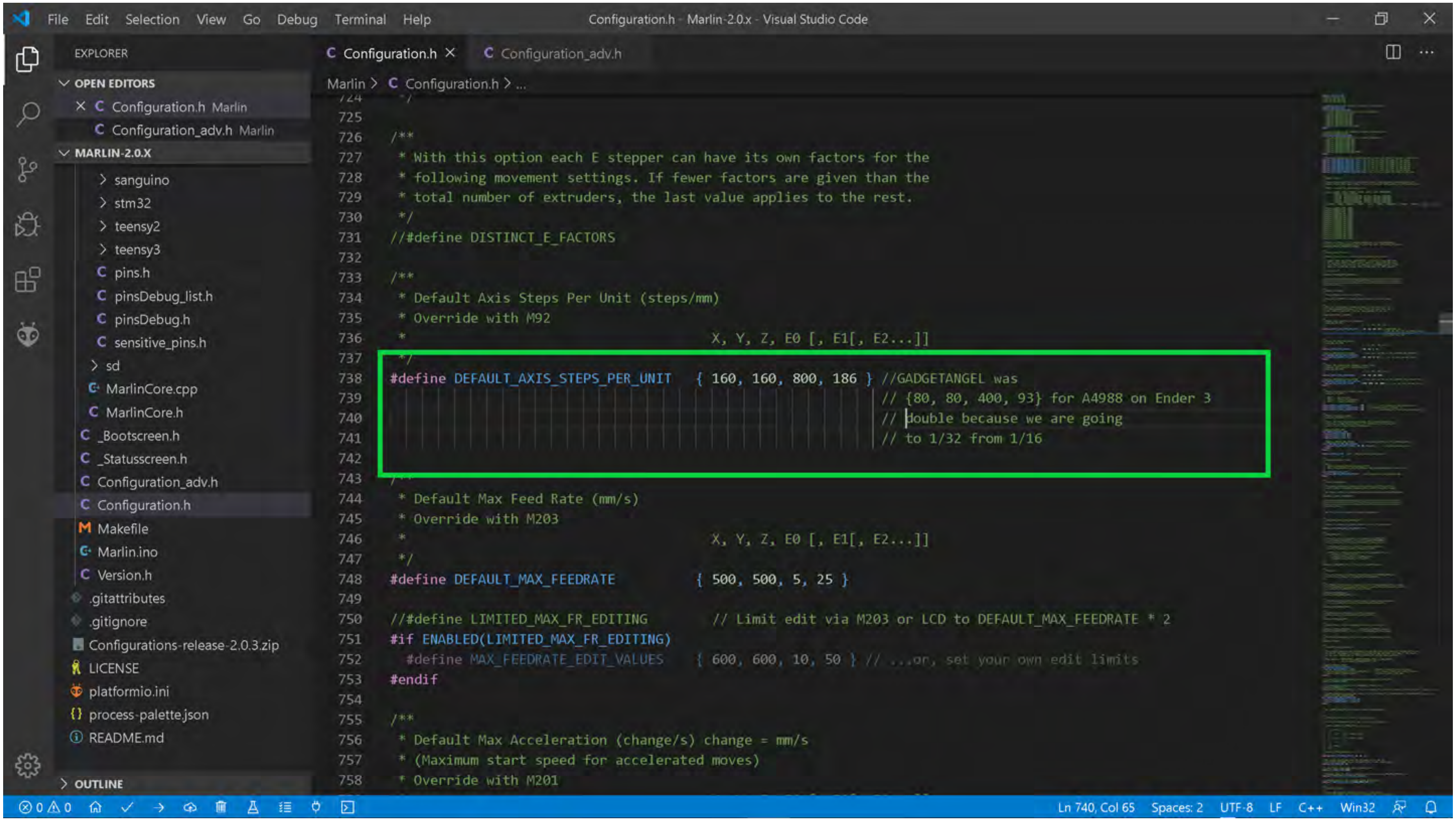
- In the Marlin firmware, the ST820 drivers needs an ACTIVE HIGH for the stepper motor driver's enable pin, so set "X_ENABLE_ON" to 1, "Y_ENABLE_ON" to 1, "Z_ENABLE_ON" to 1 and "E_ENABLE_ON" to 1, as seen in the **GREEN** box below.



- Go to the next page.

The (latest release of) Marlin Setup for BIQU ST820 Drivers

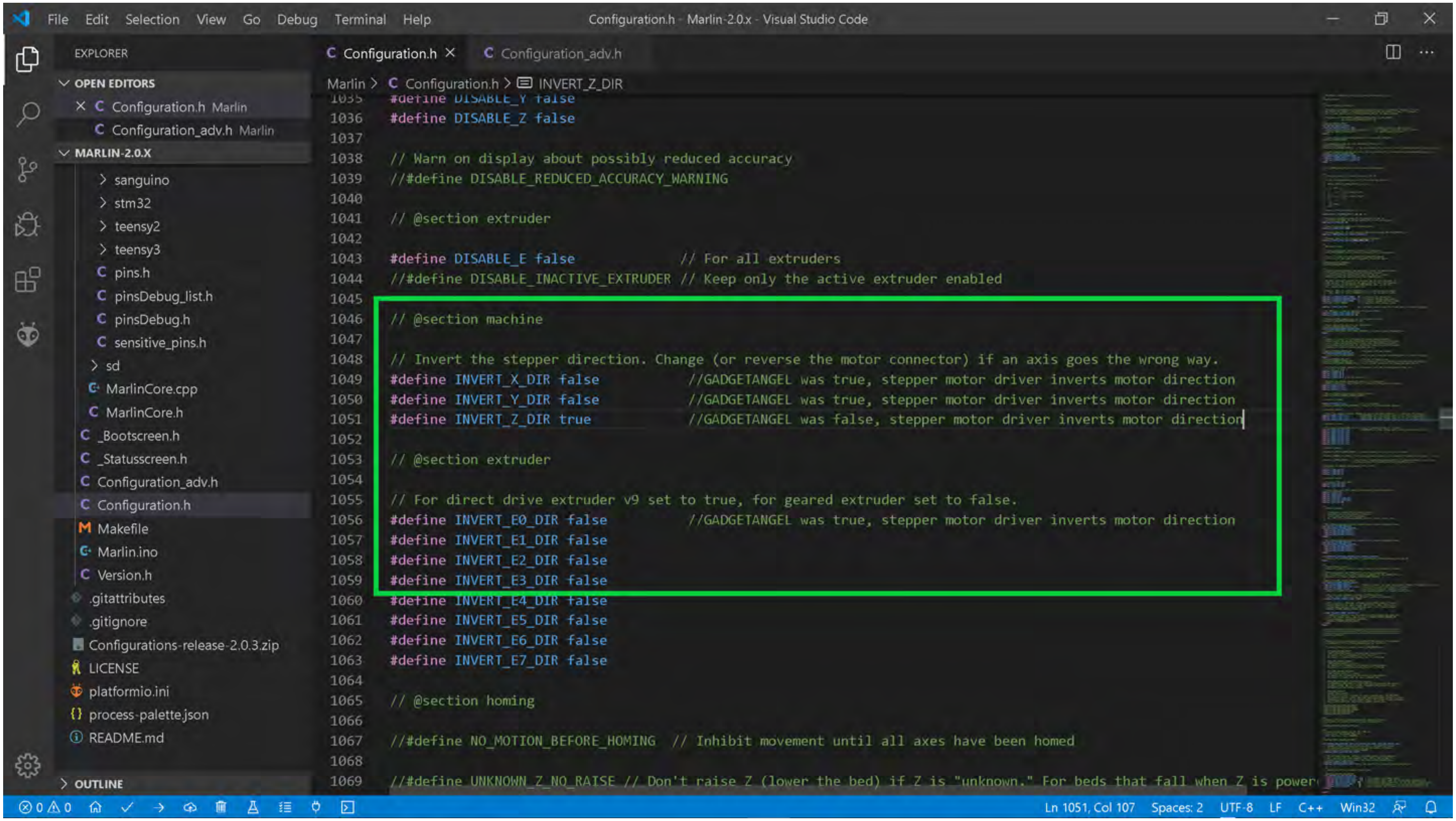
- Since we are changing from A4988 stepper motor drivers on the Ender 3 to BIQU ST820 stepper motor drivers for each axis and the extruder stepper motor driver, we will be going from 1/16 stepping to 1/32 stepping. So we are doubling our STEPS. Therefore, we must adjust our "DEFAULT_AXIS_STEPS_PER_UNIT" anytime our STEPS are NOT 1/16. So change "DEFAULT_AXIS_STEPS_PER_UNIT" to {160, 160, 800, 186}, as seen in the **GREEN** box below.



- Go to the next page.

The (latest release of) Marlin Setup for BIQU ST820 Drivers

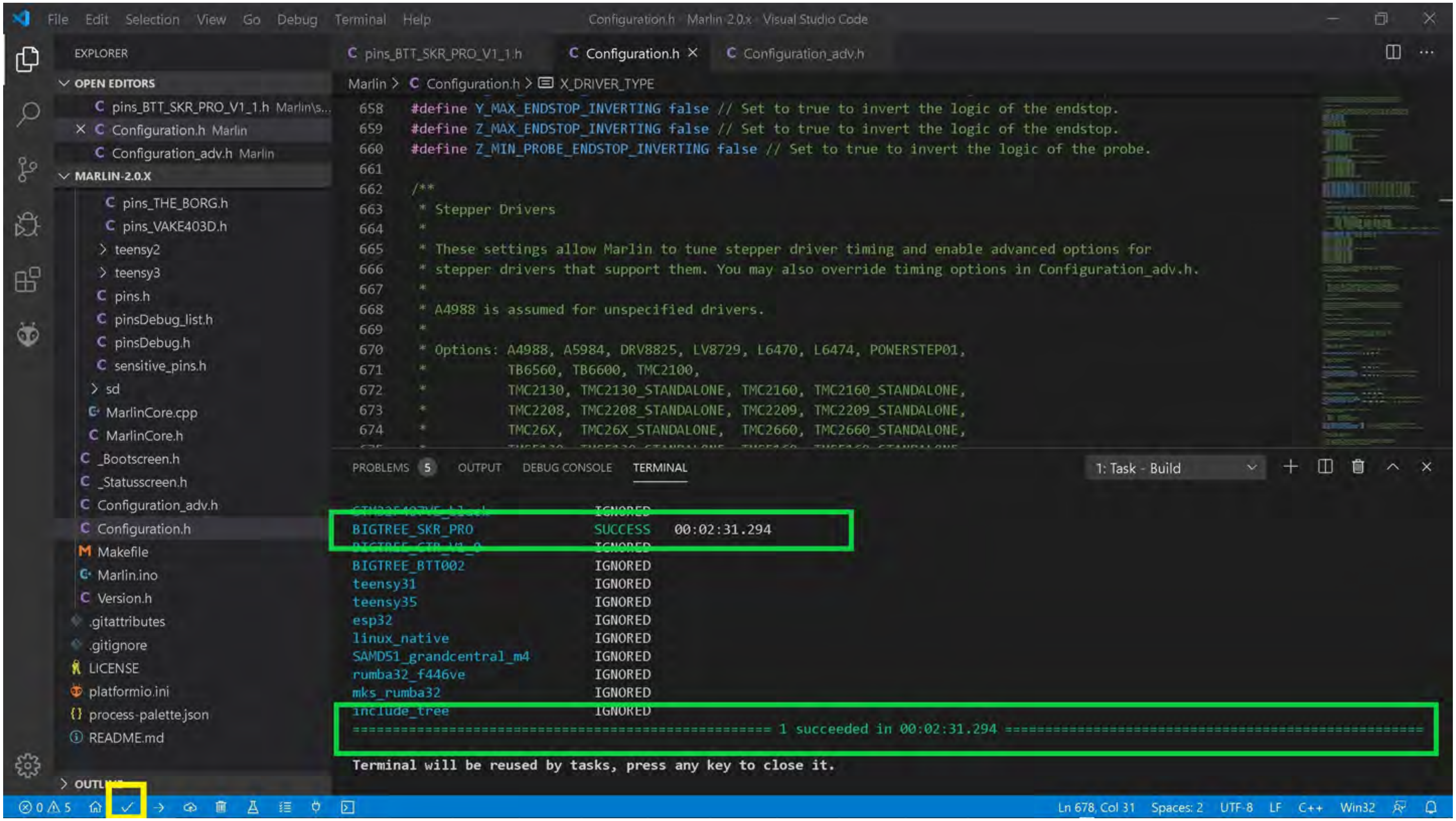
- Optional Step: I found conflicting information on the ST820 driver. Some sources say you will need to change the motor direction others say you may not. So I provide, the below information, in case you do need to change the stepper motor direction. If you prefer to change the motor direction with wiring instead of the Marlin firmware, here is a link on how to change the motor direction via the wiring (look for section labeled "Motor moving the wrong direction") **https://reprap.org/wiki/Stepper_wiring**. Other people prefer to change the motor direction in the Marlin firmware. So if you want or need to change the motor direction in Marlin, then if the axis' setting you will be using the ST820 driver on was "true" change it to "false", as shown in the **GREEN** box below. If the setting was "false", now set it to "true", as shown in the **GREEN** box below



- Go to the next page.

The (latest release of) Marlin Setup for BIQU ST820 Drivers

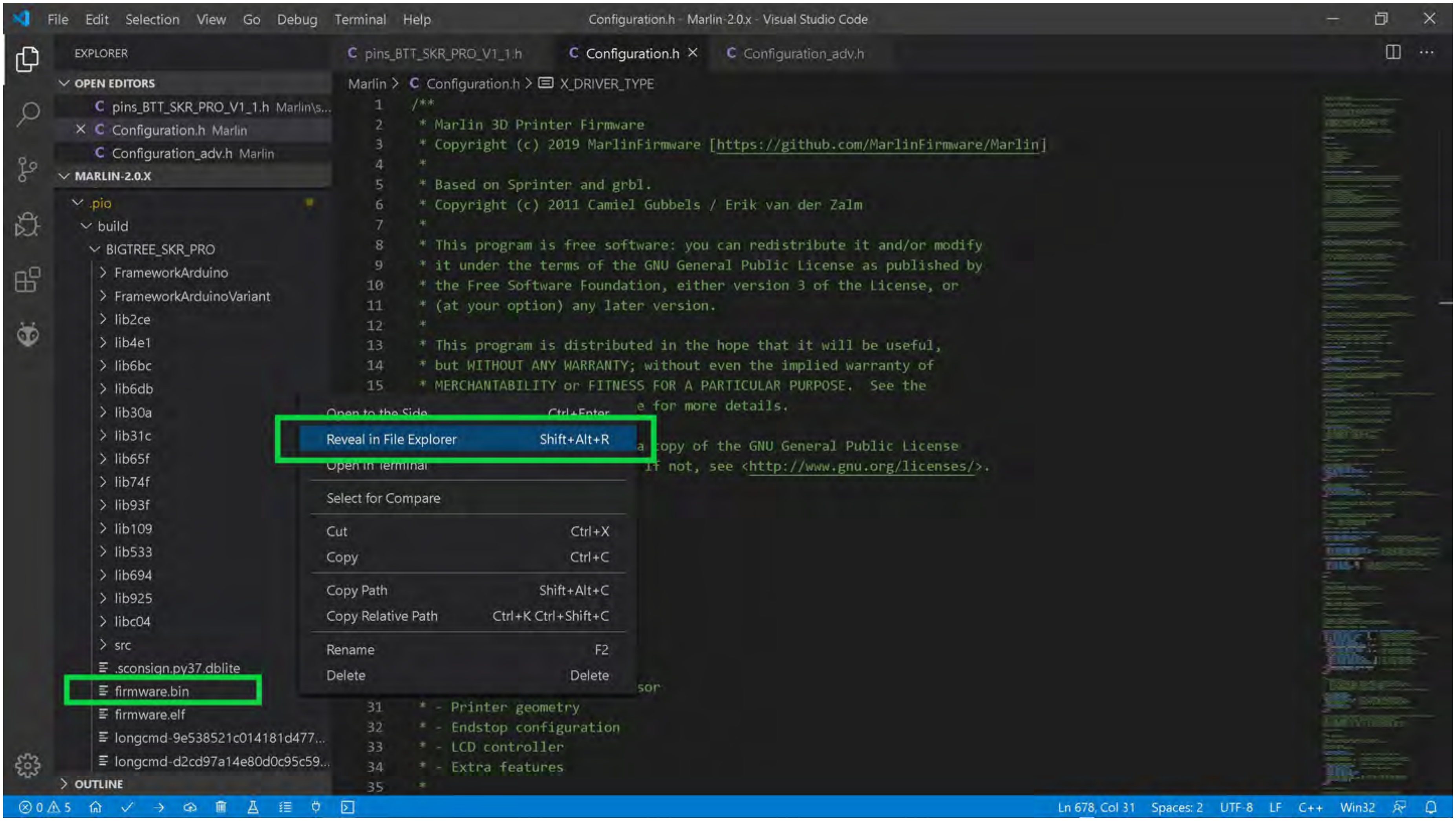
- The end of Marlin setup for BIQU ST820 drivers. Now, compile your code by clicking on the check mark, as seen in the **YELLOW** box below. Once the firmware has compiled successfully you will see the following messages, as shown in the **GREEN** boxes below.



- For the SKR PRO V1.1 board you must save the "firmware.bin" file to the micro SD card then place the micro SD card into the micro SD card reader of the SKR PRO V1.1 board and turn on the power to your printer for the NEW firmware to be automatically applied to the SKR PRO MCU. To locate the "firmware.bin" file, and learn how to transfer it to your micro SD card go to the next page.

The (latest release of) Marlin Setup for BIQU ST820 Drivers

- The "firmware.bin" file can be found by using the left side of the VS code window and going to folder Marlin-2.0.X/.pio/build/BIGTREE_SKR_PRO/ and right clicking on the "firmware.bin" file name to bring up the content menu. Select "Reveal in File Explorer" to have your Window's machine open a file explorer window.

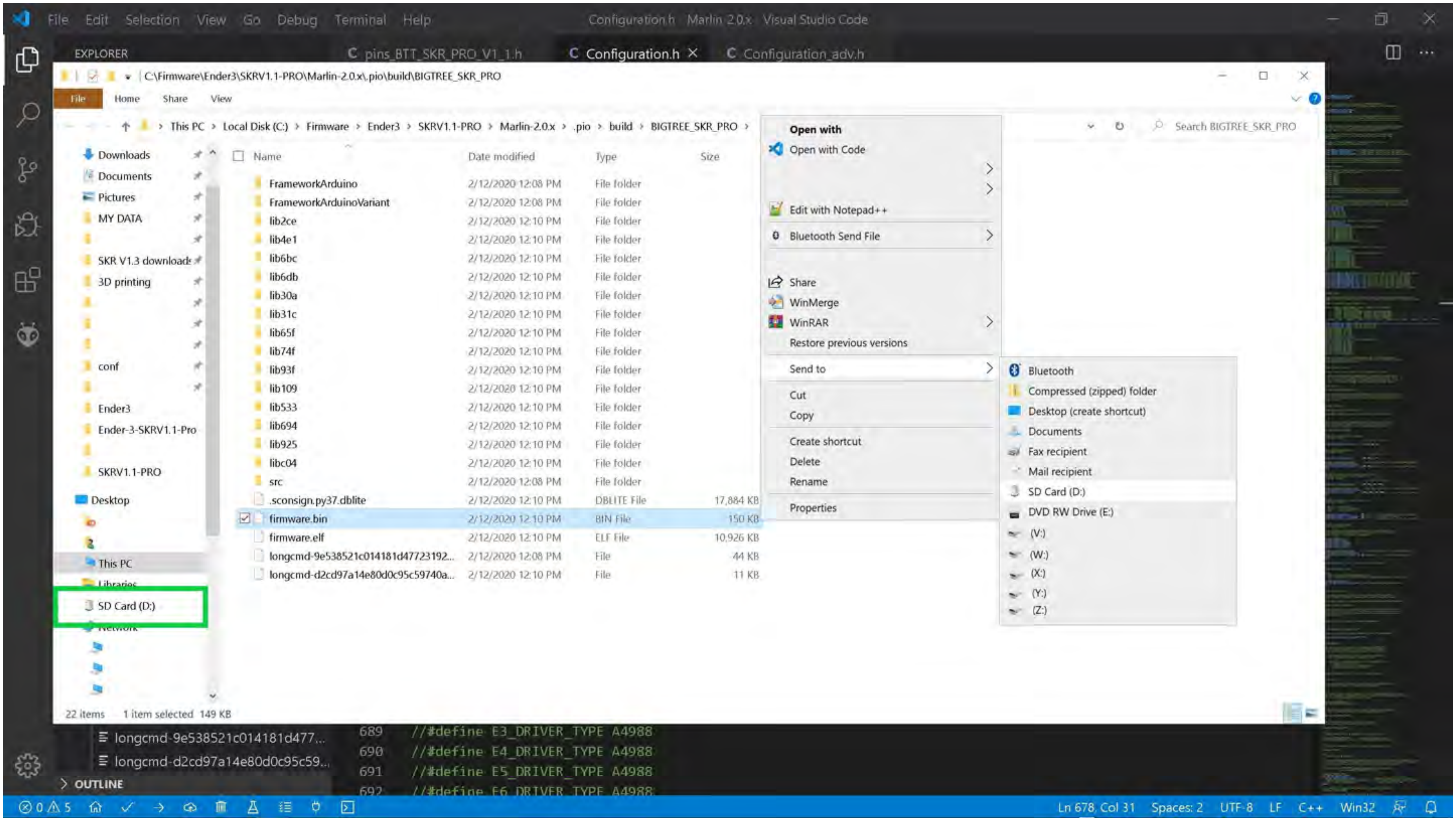


- Go to the next page.

The (latest release of) Marlin Setup for BIQU ST820 Drivers

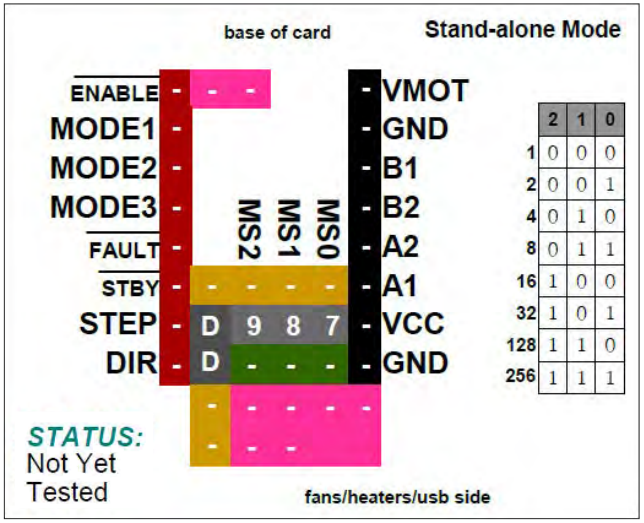
- Ensure your micro SD card, which came with your SKR PRO V1.1 board, is in a micro-to-SD-card adapter and plugged into your Window's SD card reader.

    - From the file explorer window right click on the "firmware.bin" file name and select "Send to" from the content menu. For my computer "D:" is my SD card reader. For your computer it may be a different drive letter. The "firmware.bin" file will be written to the SD card and renamed to "FIRMWARE.CUR".



- Right click on the SD card reader, as seen in the **GREEN** box above, and select "Eject" from the context menu. Take the micro SD card out of the adapter and place it into the micro SD card reader of the SKR PRO V1.1 board. Turn on the power to the printer (i.e. the SKR PRO V1.1 board) and the "FIRMWARE.CUR" file will be uploaded to the board. After the upload the firmware file will be renamed to "firmware.bin" and stored on the micro SD card.

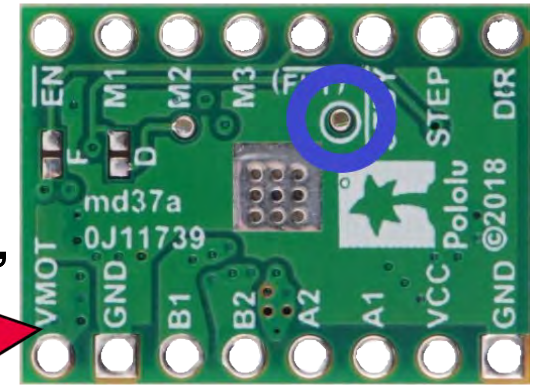
POLOLU ST820 (STSPIN820)



	2	1	0
1	0	0	0
2	0	0	1
4	0	1	0
8	0	1	1
16	1	0	0
32	1	0	1
128	1	1	0
256	1	1	1

NOTE: Use the potentiometer (POT) on the top of the board to adjust your V_{ref}.

Note: "V_{ref} Test point" location is on the bottom of the driver board, as shown in BLUE



Driver Chip

Pololu

ST820

Maximum 256 Subdivision

45V DC
1.5A (peak)

MS2	MS1	MS0	Microstep Resolution	Excitation Mode
Low	Low	Low	Full step	2 Phase
Low	Low	High	Half step	1-2 Phase
Low	High	Low	1/4 step	W1-2 Phase
Low	High	High	1/8 step	2W1-2 Phase
High	Low	Low	1/16 step	4W1-2 Phase
High	Low	High	1/32 step	8W1-2 Phase
High	High	Low	1/128 step	16W1-2 Phase
High	High	High	1/256 step	32W1-2 Phase

Driving Current Calculation Formula

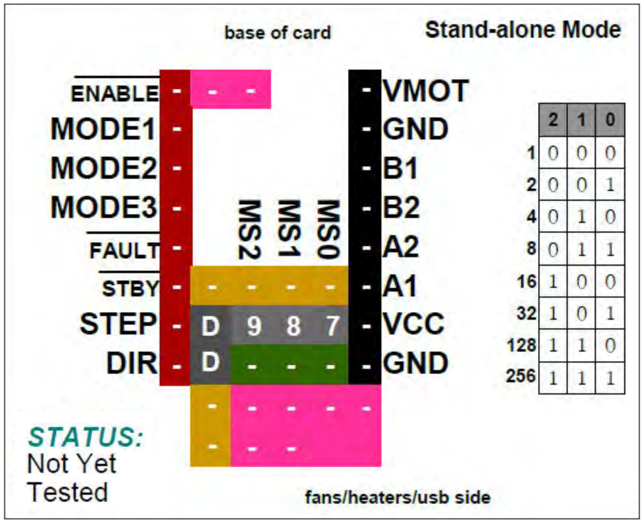
R_S(Typical Sense Resistor)= 0.2Ω

$$I_{MAX}=V_{ref} * \frac{1}{R_S}$$

$$V_{ref}=I_{MAX} * R_S$$

NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

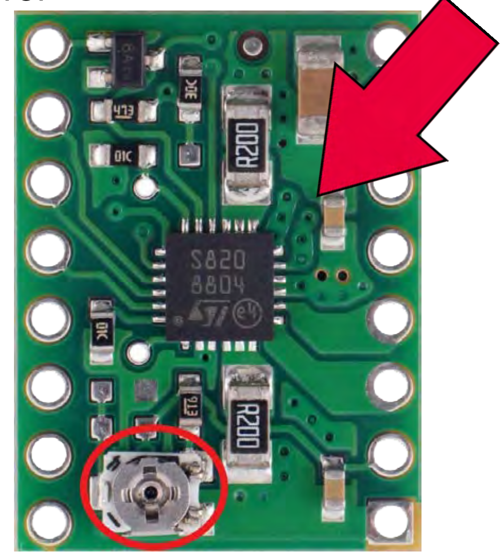
POLOLU ST820 (STSPIN820)



	2	1	0
1	0	0	0
2	0	0	1
4	0	1	0
8	0	1	1
16	1	0	0
32	1	0	1
128	1	1	0
256	1	1	1

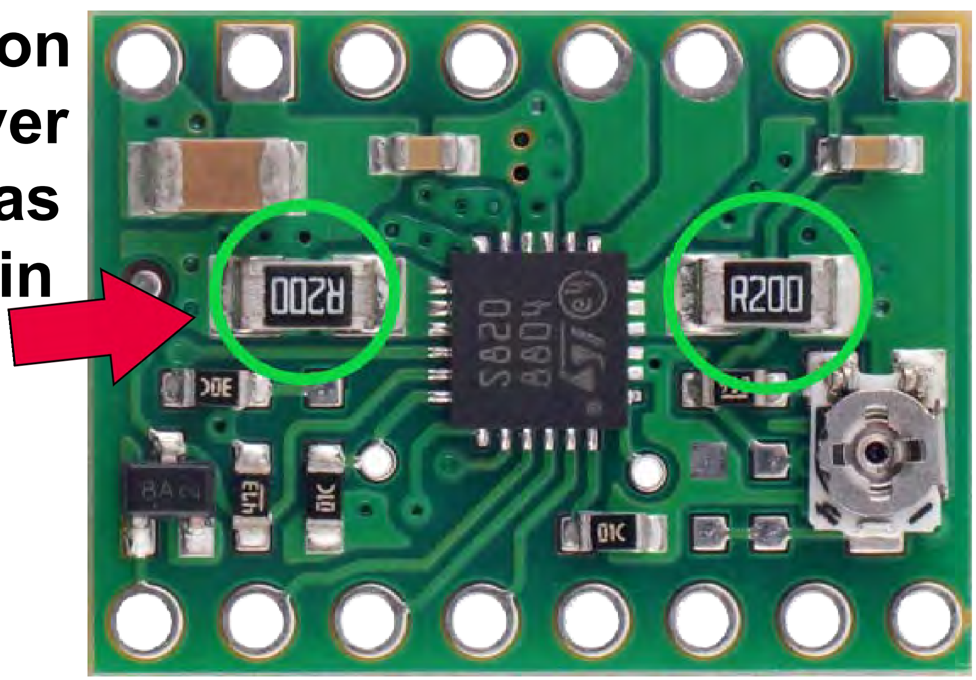
Note: Use 90% of the calculated V_{ref} when tuning the stepper driver board.

NOTE: Use the potentiometer (POT) on the top of the board to adjust your V_{ref}. See **Appendix A** for instructions on how to set the V_{ref} on a driver board.



Note: Check your current sense resistors (R_s) values on the driver board, as shown in GREEN

Note: See this video about current sense resistors (R_s) and their possible locations: **https://youtu.be/8wk1elugv5A**



R_s = R050 is 0.05 Ohms

R_s = R068 is 0.068 Ohms

R_s = R100 is 0.1 Ohms

R_s = R150 is 0.15 Ohms

R_s = R200 is 0.2 Ohms

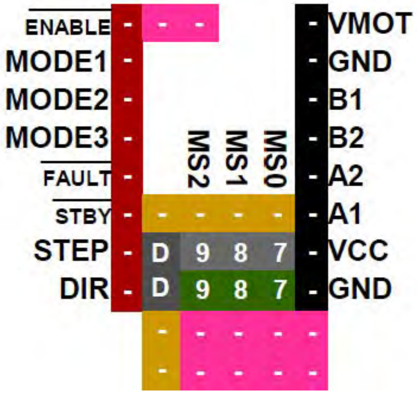
R_s = R220 is 0.22 Ohms

POLOLU ST820 (STSPIN820)

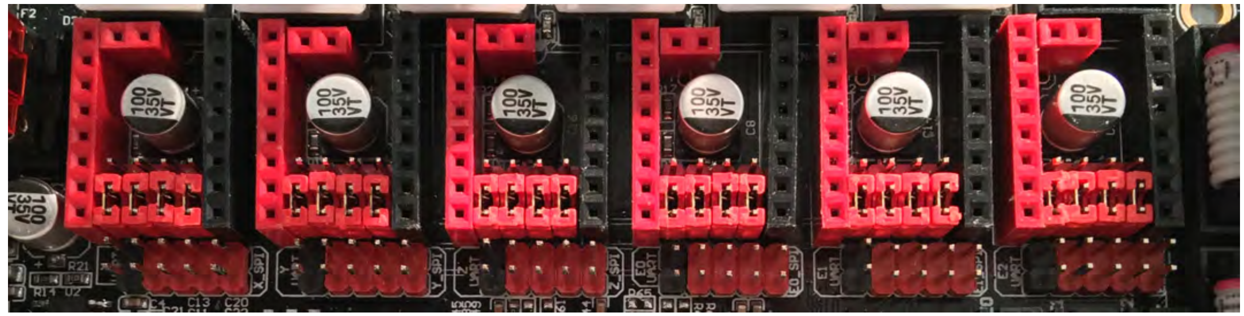
Stand-alone Mode

Important: This driver has special requirements in the Configuration and Configuration_adv.h. Also, this driver requires constant cooling the moment any motor is used, or it will switch on and off.

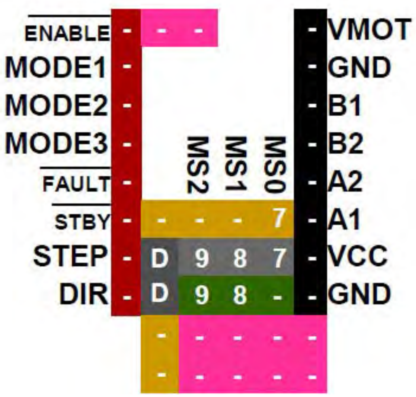
STEP



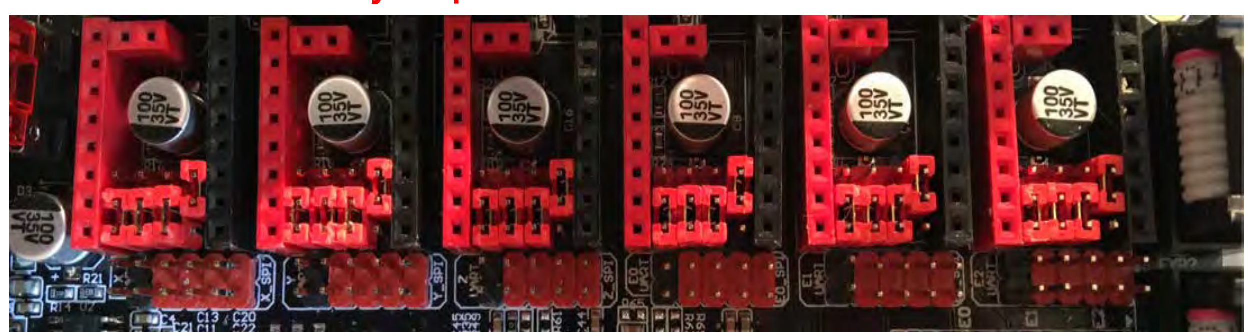
Note: The "D" jumper MUST be SET!



1 / 2



Note: The "D" jumper MUST be SET!



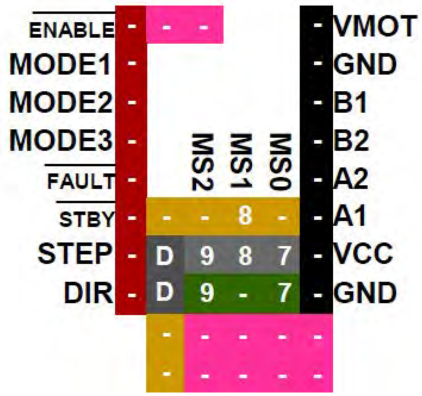
NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

POLOLU ST820 (STSPIN820)

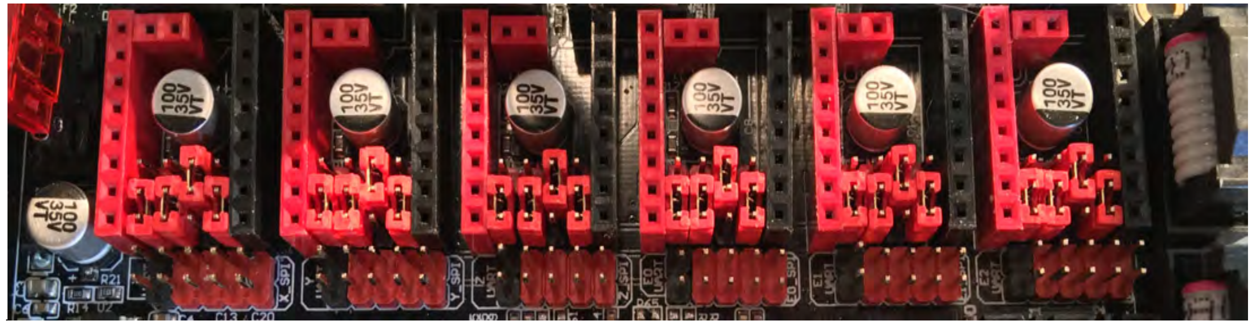
Stand-alone Mode

Important: This driver has special requirements in the Configuration and Configuration_adv.h. Also, this driver requires constant cooling the moment any motor is used, or it will switch on and off.

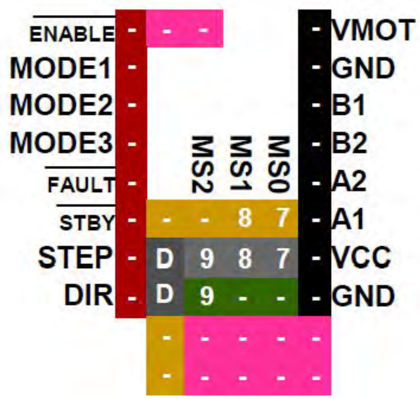
1 / 4



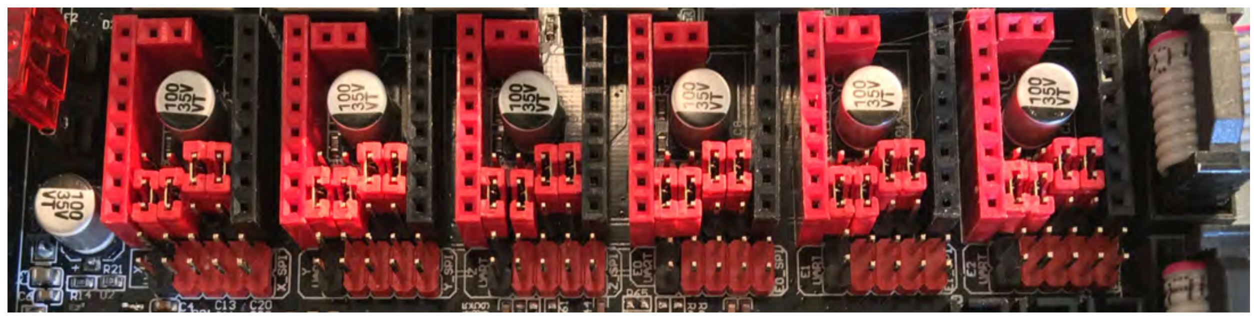
Note: The "D" jumper MUST be SET!



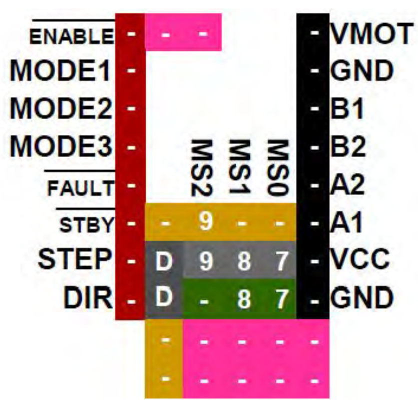
1 / 8



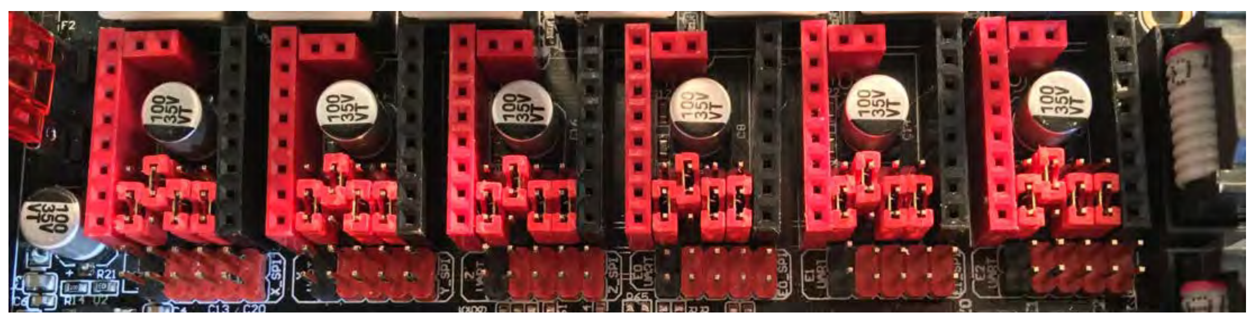
Note: The "D" jumper MUST be SET!



1 / 16



Note: The "D" jumper MUST be SET!



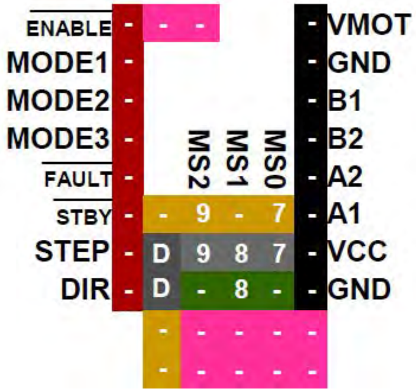
NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

POLOLU ST820 (STSPIN820)

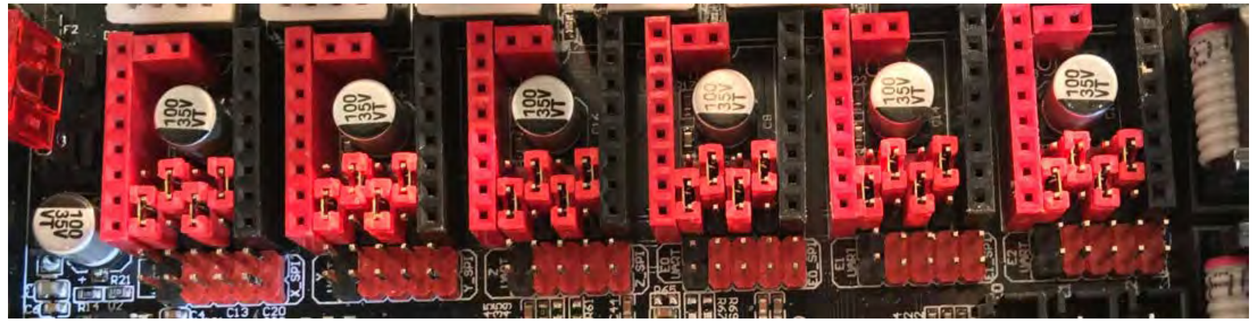
Stand-alone Mode

Important: This driver has special requirements in the Configuration and Configuration_adv.h. Also, this driver requires constant cooling the moment any motor is used, or it will switch on and off.

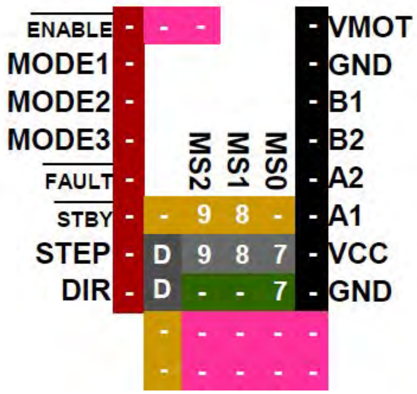
1 / 32



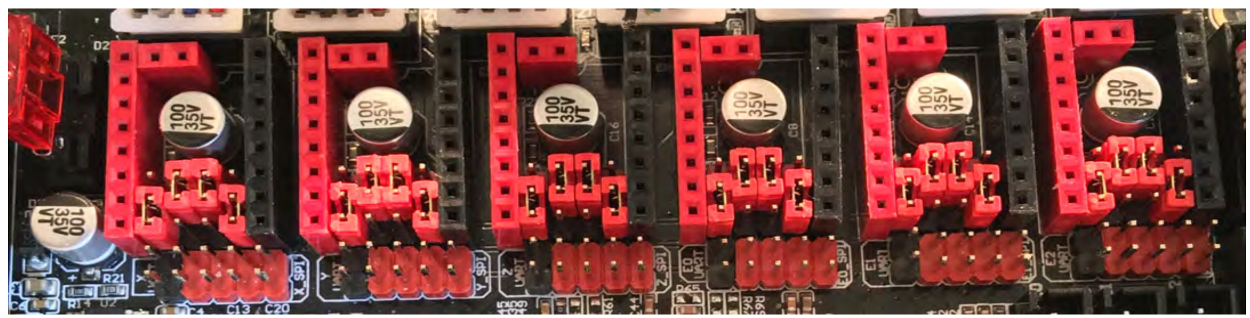
Note: The "D" jumper MUST be SET!



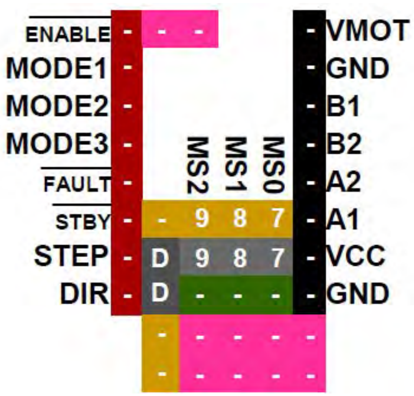
1 / 128



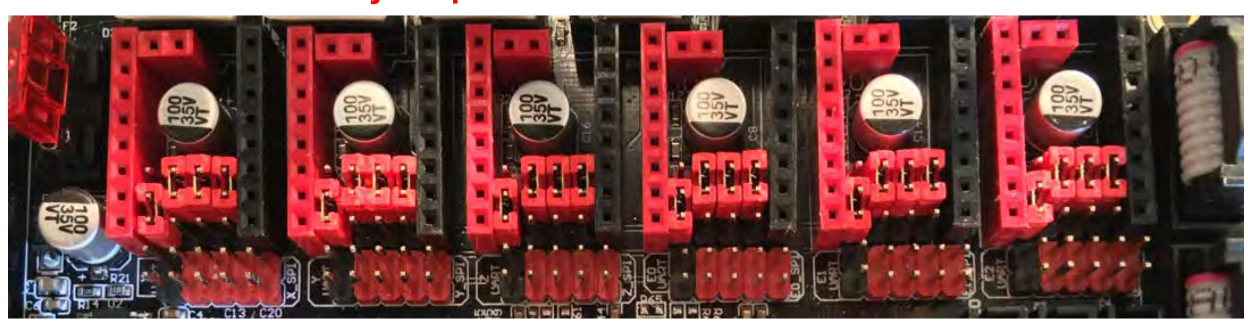
Note: The "D" jumper MUST be SET!



1 / 256



Note: The "D" jumper MUST be SET!

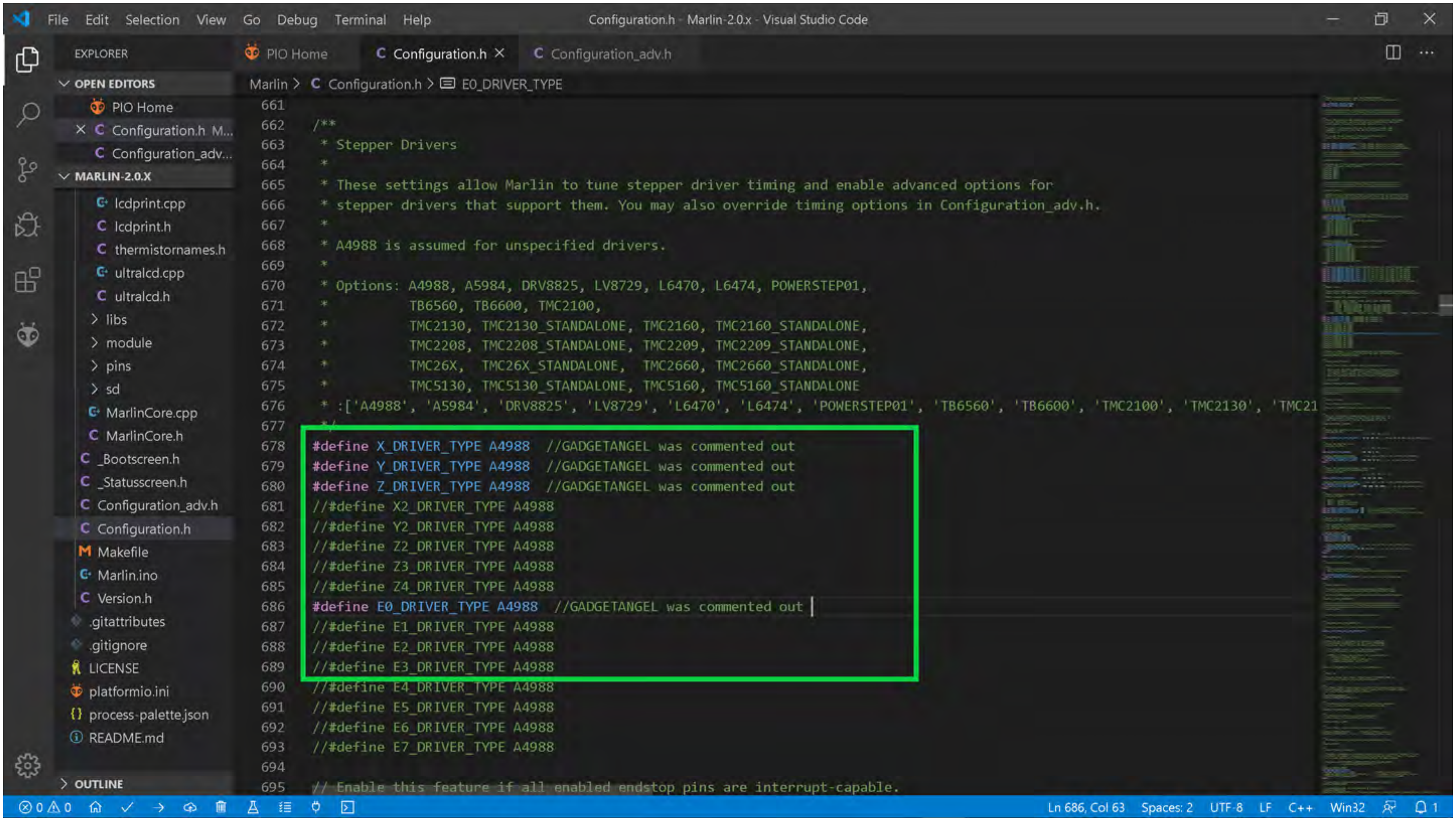


NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

The (latest release of) Marlin Setup for POLOLU ST820 (STSPIN820) Drivers

NOTE: **Go to Appendix C,** and then come back here for the changes to Marlin for POLOLU ST820 (STSPIN820) stepper motor drivers.

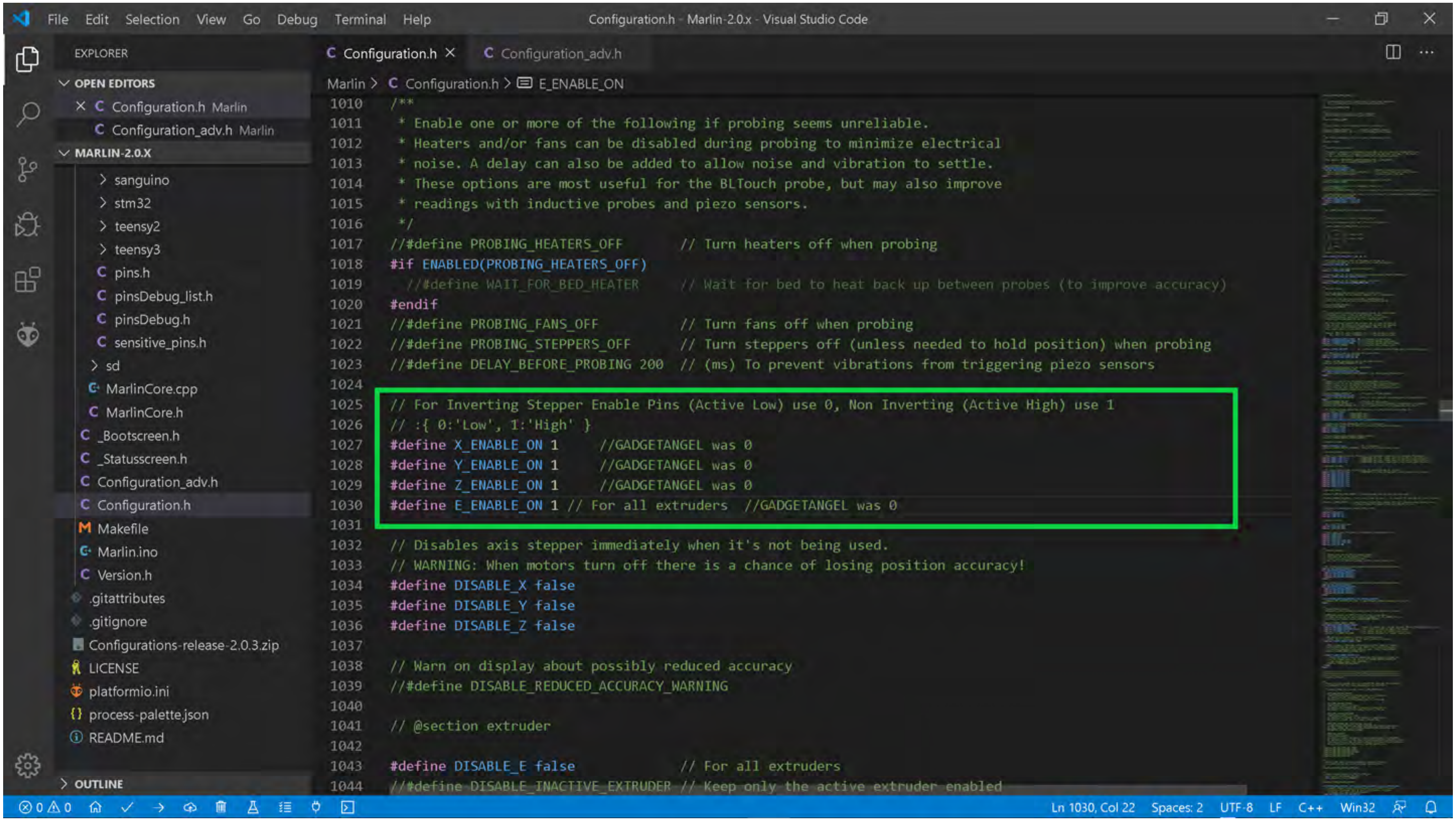
- Change the stepper motor drivers so that Marlin knows you are using POLOLU ST820 (STSPIN820) drivers. Change one line for each axis and one for each extruder you will be using. See the picture below for an example of how to use POLOLU ST820 (STSPIN820) drivers. When two "//" appear at the beginning of a line that means that line is commented out. To un-comment a line just remove the leading two forward slashes ("//").
- The ST820 is a drop in replacement for the A4988. Since Marlin does not have an option for ST820 we will use the A4988 option.



- Go to the next page.

The (latest release of) Marlin Setup for POLOLU ST820 (STSPIN820) Drivers

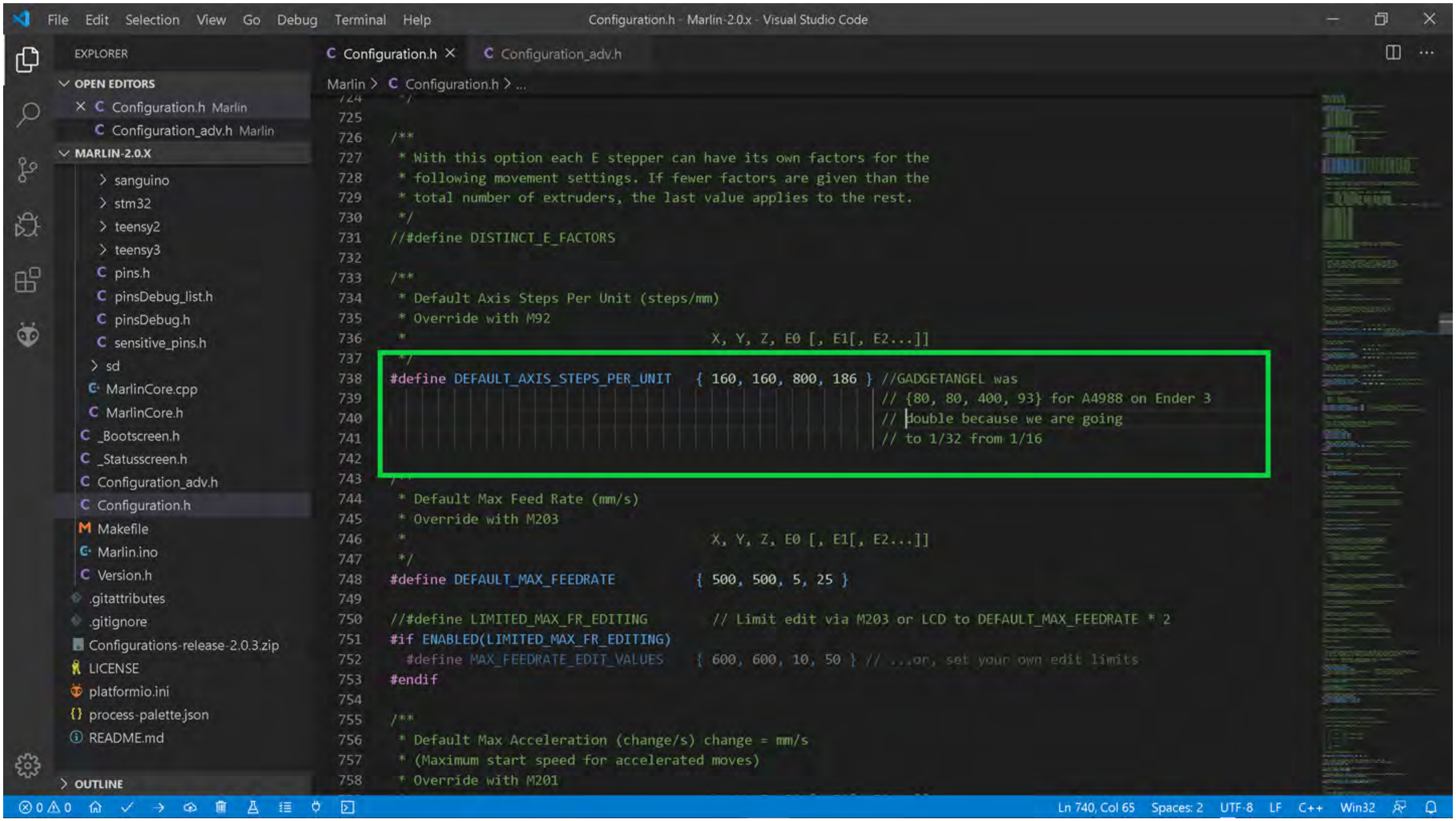
- In the Marlin firmware, the ST820 drivers needs an ACTIVE HIGH for the stepper motor driver's enable pin, so set "X_ENABLE_ON" to 1, "Y_ENABLE_ON" to 1, "Z_ENABLE_ON" to 1 and "E_ENABLE_ON" to 1, as seen in the **GREEN** box below.



- Go to the next page.

The (latest release of) Marlin Setup for POLOLU ST820 (STSPIN820) Drivers

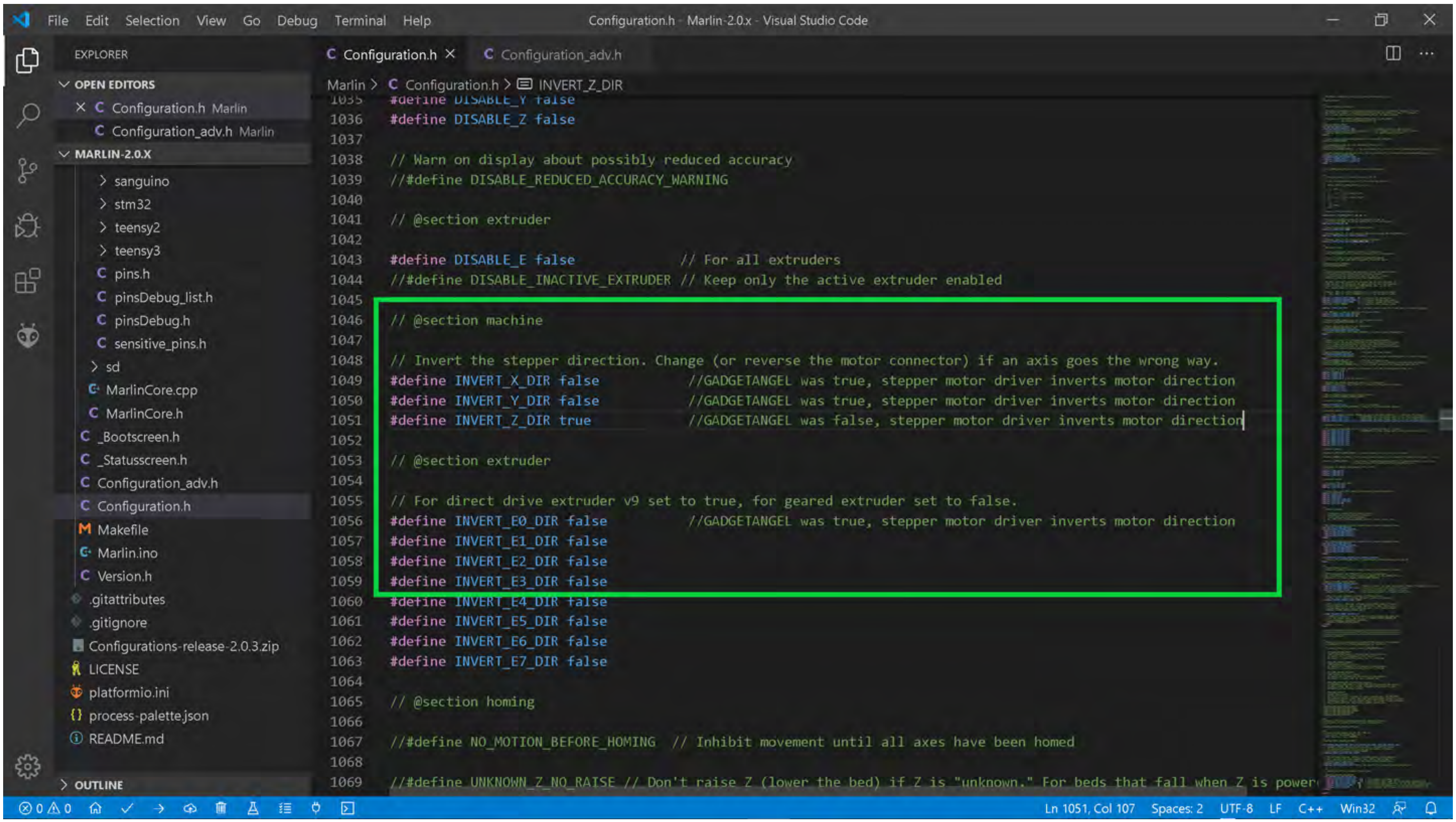
- Since we are changing from A4988 stepper motor drivers on the Ender 3 to POLOLU ST820 (STSPIN820) stepper motor drivers for each axis and the extruder stepper motor driver, we will be going from 1/16 stepping to 1/32 stepping. So we are doubling our STEPS. Therefore, we must adjust our "DEFAULT_AXIS_STEPS_PER_UNIT" anytime our STEPS are NOT 1/16. So change "DEFAULT_AXIS_STEPS_PER_UNIT" to {160, 160, 800, 186}, as seen in the **GREEN** box below.



- Go to the next page.

The (latest release of) Marlin Setup for POLOLU ST820 (STSPIN820) Drivers

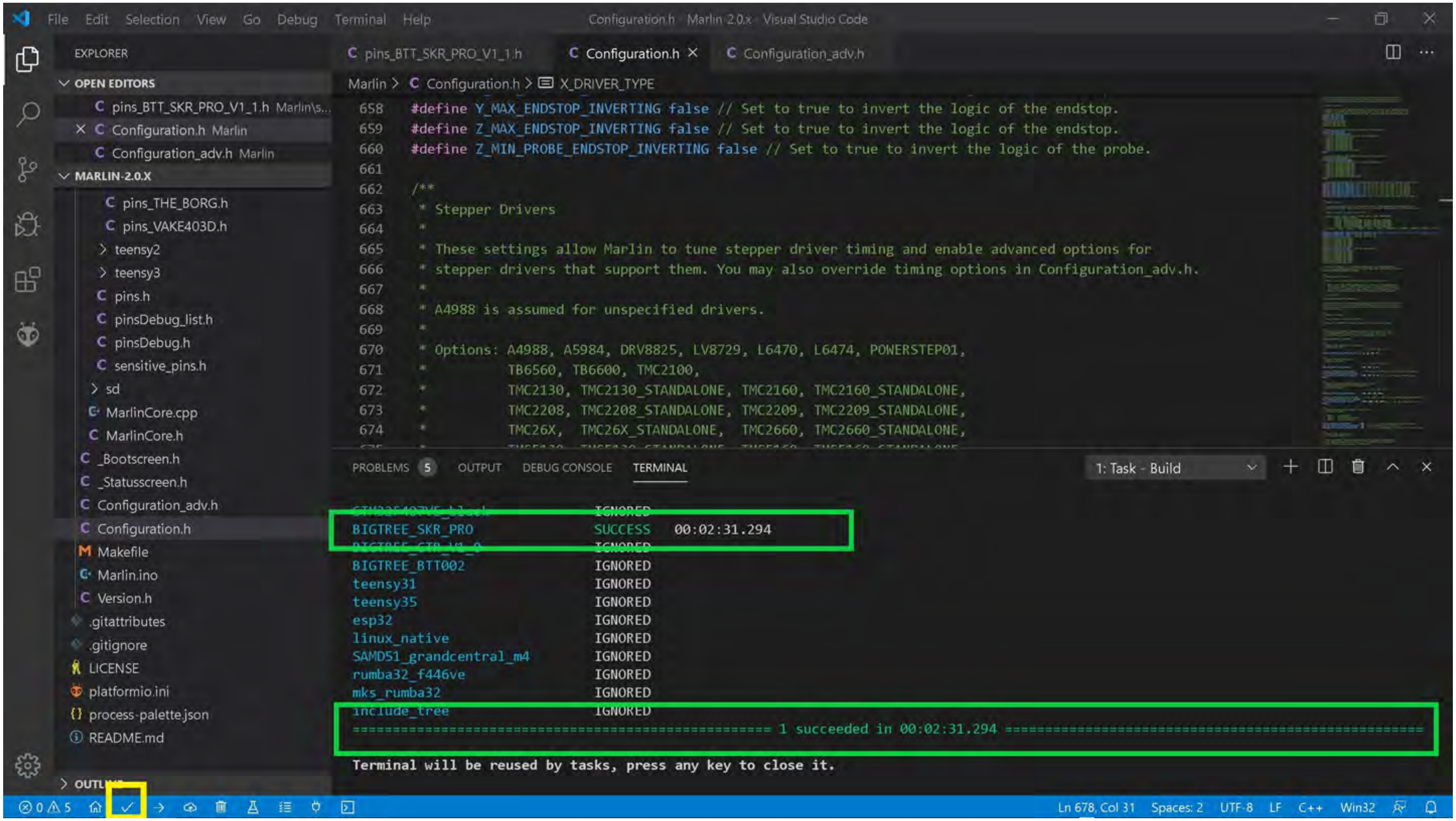
- Optional Step: I found conflicting information on the ST820 driver. Some sources say you will need to change the motor direction others say you may not. So I provide the below information in case you do need to change the stepper motor direction. If you prefer to change the motor direction with wiring instead of the Marlin firmware, here is a link on how to change the motor direction via the wiring (look for section labeled "Motor moving the wrong direction") **https://reprap.org/wiki/Stepper_wiring**. Other people prefer to change the motor direction in the Marlin firmware. So if you want or need to change the motor direction in Marlin, then if the axis' setting you will be using the ST820 driver on was "true" change it to "false", as shown in the **GREEN** box below. If the setting was "false", now set it to "true", as shown in the **GREEN** box below



- Go to the next page.

The (latest release of) Marlin Setup for POLOLU ST820 (STSPIN820) Drivers

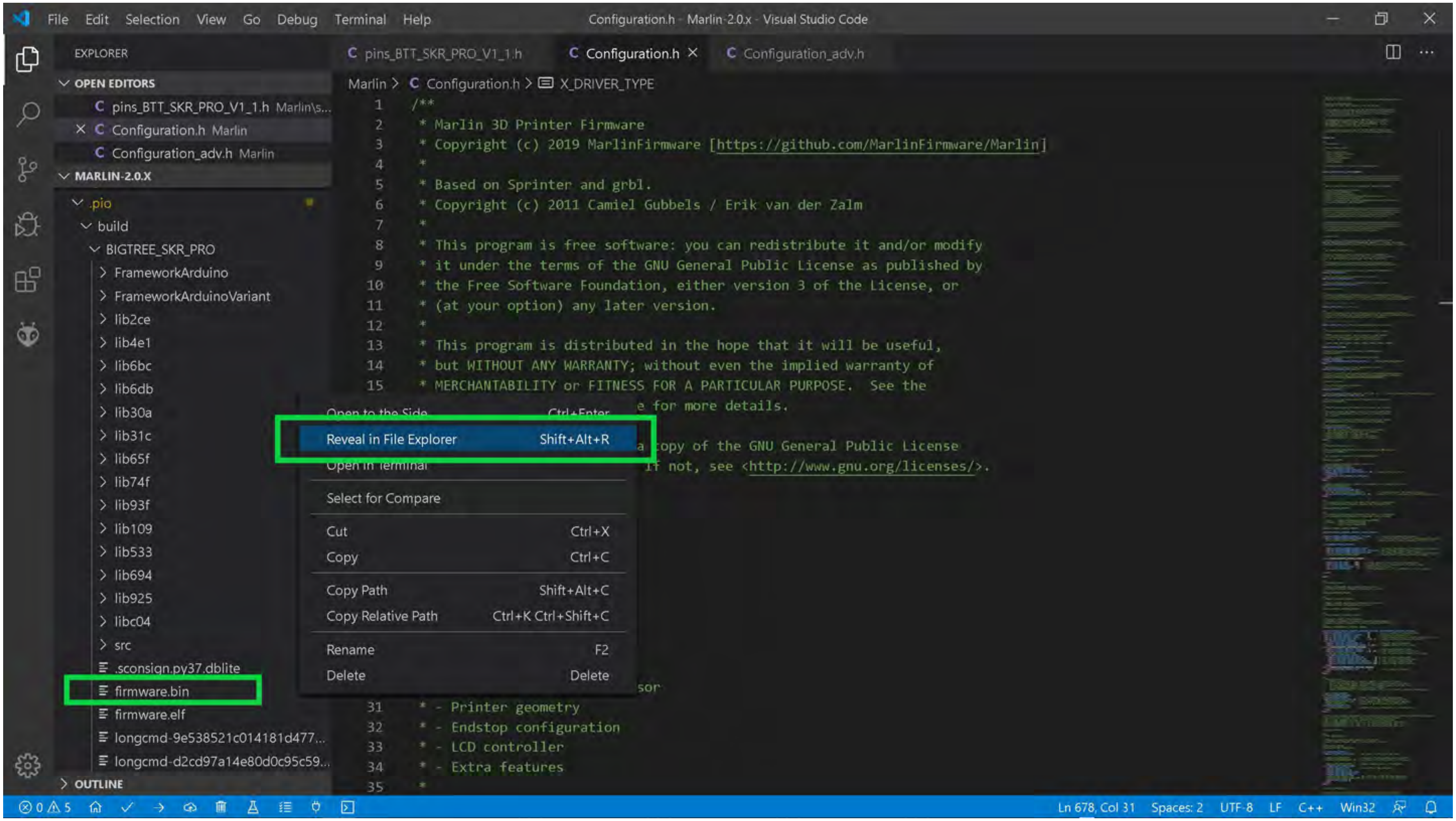
- The end of Marlin setup for POLOLU ST820 (STSPIN820) drivers. Now, compile your code by clicking on the check mark, as seen in the **YELLOW** box below. Once the firmware has compiled successfully you will see the following messages, as shown in the **GREEN** boxes below.



- For the SKR PRO V1.1 board you must save the "firmware.bin" file to the micro SD card then place the micro SD card into the micro SD card reader of the SKR PRO V1.1 board and turn on the power to your printer for the NEW firmware to be automatically applied to the SKR PRO MCU. To locate the "firmware.bin" file, and learn how to transfer it to your micro SD card go to the next page.

The (latest release of) Marlin Setup for POLOLU ST820 (STSPIN820) Drivers

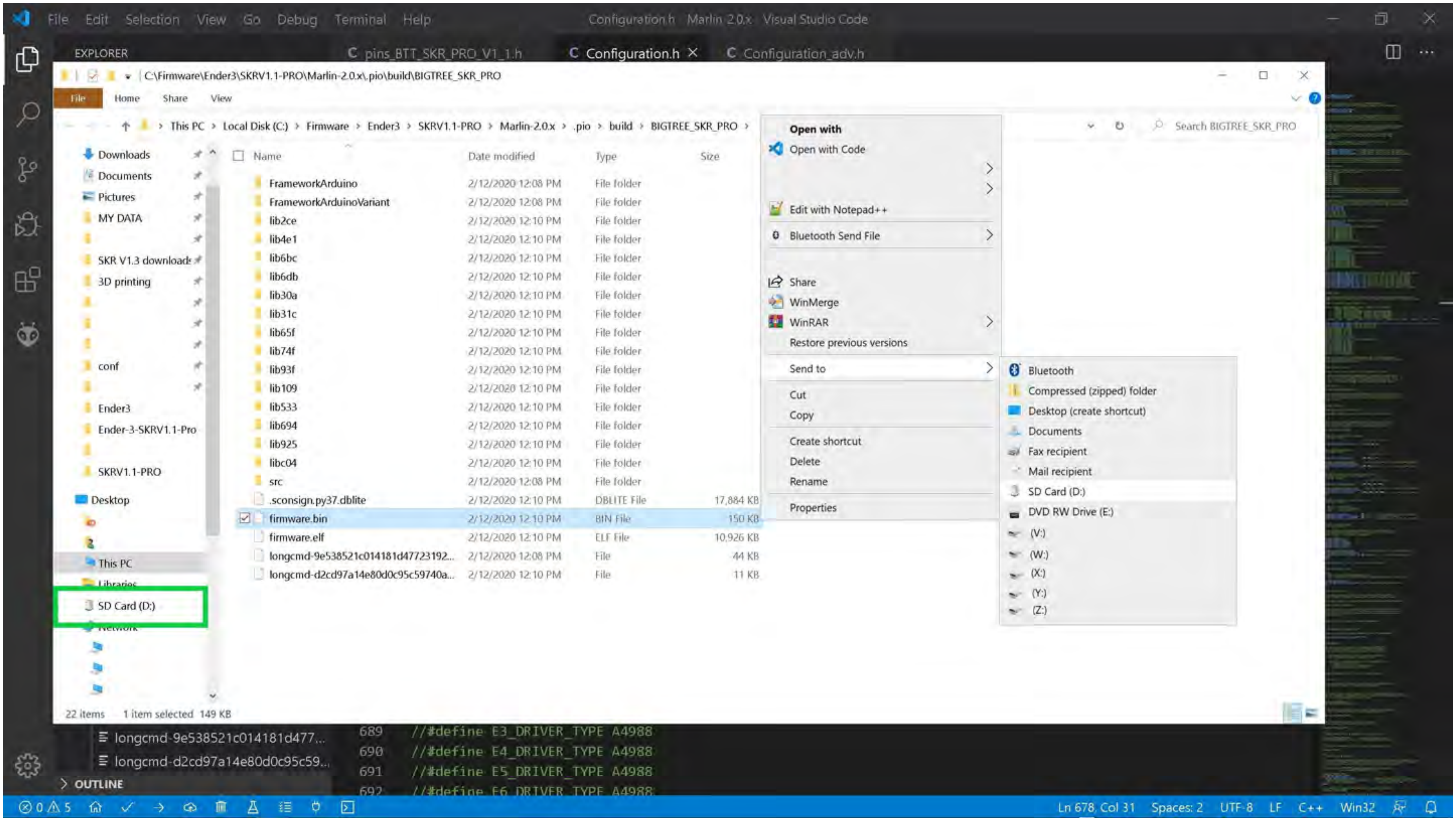
- The "firmware.bin" file can be found by using the left side of the VS code window and going to folder Marlin-2.0.X/.pio/build/ BIGTREE_SKR_PRO/ and right clicking on the "firmware.bin" file name to bring up the content menu. Select "Reveal in File Explorer" to have your Window's machine open a file explorer window.



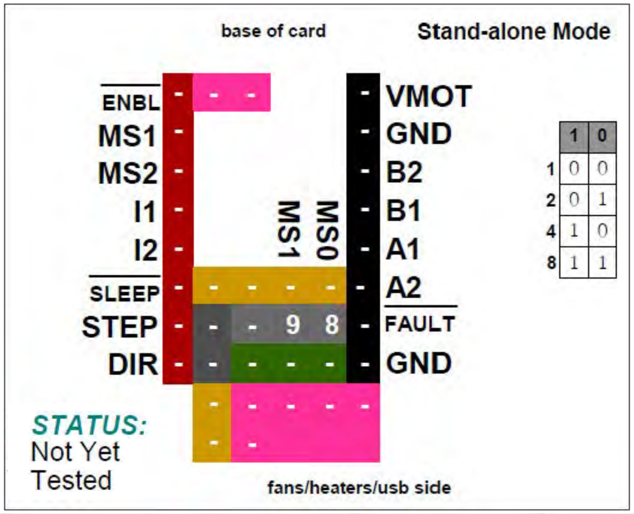
- Go to the next page.

The (latest release of) Marlin Setup for POLOLU ST820 (STSPIN820) Drivers

- Ensure your micro SD card, which came with your SKR PRO V1.1 board, with your SKR PRO V1.1 board is in a micro-to-SD-card adapter and plugged into your Window's SD card reader.
    - From the file explorer window right click on the "firmware.bin" file name and select "Send to" from the content menu. For my computer "D:" is my SD card reader. For your computer it may be a different drive letter. The "firmware.bin" file will be written to the SD card and renamed to "FIRMWARE.CUR".



- Right click on the SD card reader, as seen in the **GREEN** box above, and select "Eject" from the context menu. Take the micro SD card out of the adapter and place it into the micro SD card reader of the SKR PRO V1.1 board. Turn on the power to the printer (i.e. the SKR PRO V1.1 board) and the "FIRMWARE.CUR" file will be uploaded to the board. After the upload the firmware file will be renamed to "firmware.bin" and stored on the micro SD card.

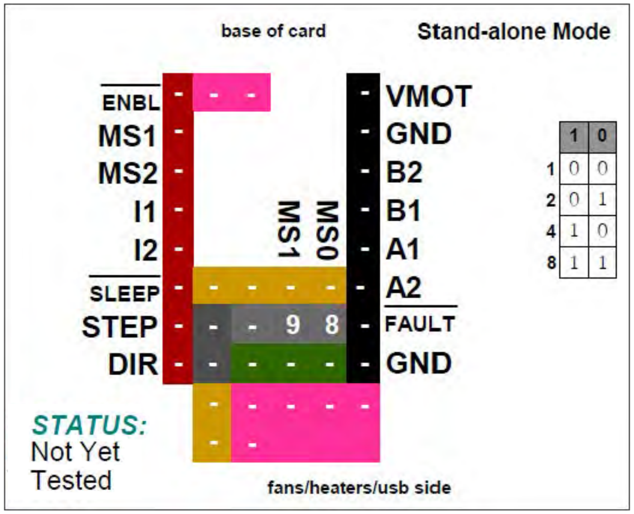


POLOLU MP6500

Note: See the next page for information about location of the current sense resistors and how to set V_{ref} on the stepper motor driver board.

Driver Chip	MS1	MS0	Microstep Resolution	Excitation Mode
Pololu MP6500 Maximum 8 Subdivision 35V DC 2.5A (peak)	Low	Low	Full step	2 Phase
	Low	High	Half (1/2) step	1-2 Phase
	High	Low	Quarter (1/4) step	W1-2 Phase
	High	High	Eighth (1/8) step	2W1-2 Phase
Driving Current Calculation Formula	$I_{MAX} = V_{ref} * 3.5$			$V_{ref} = \frac{I_{MAX}}{3.5}$

NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

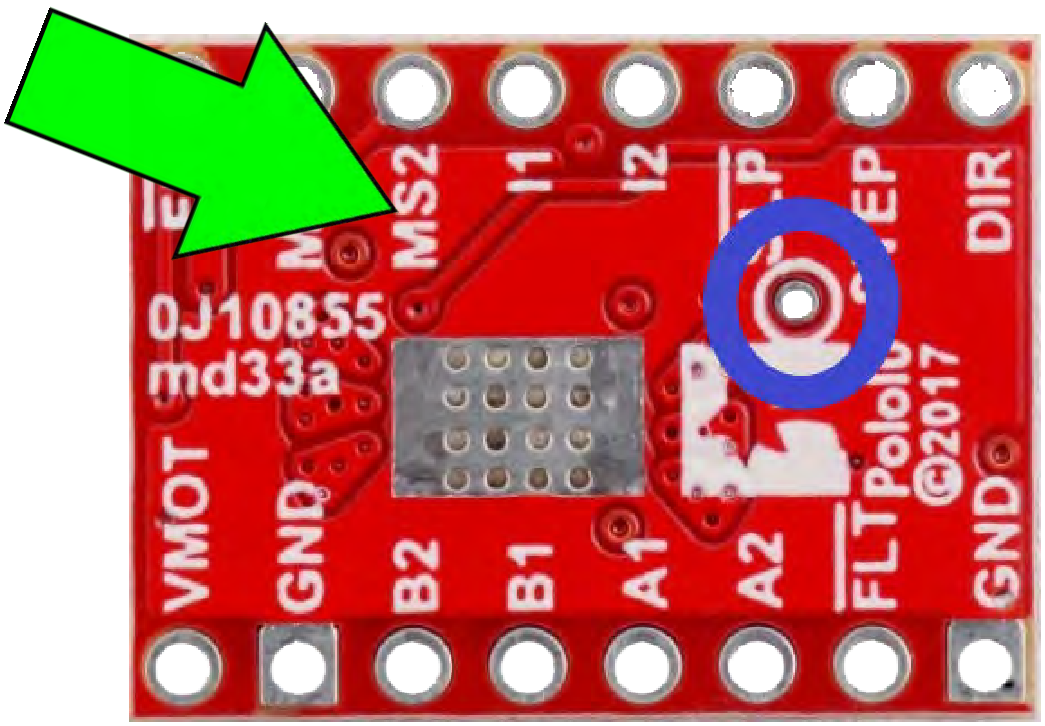


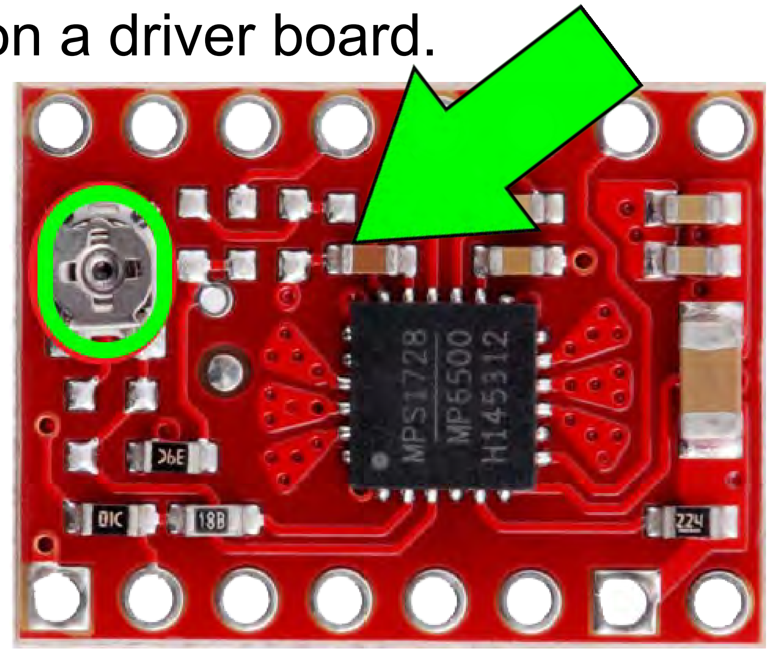
POLOLU MP6500

Note: Use 90% of the calculated V_{ref} when tuning the stepper driver board.

NOTE: Use the potentiometer (POT) on the top of the board (or use the board's "V_{ref} Test point") to adjust your V_{ref}. See **Appendix A** for instructions on how to set the V_{ref} on a driver board.

Note: "V_{ref} Test point" location is on the bottom of the driver board, as shown in BLUE



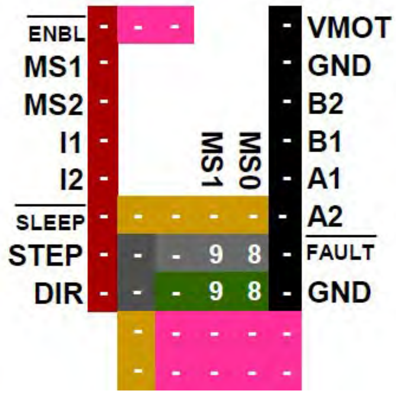


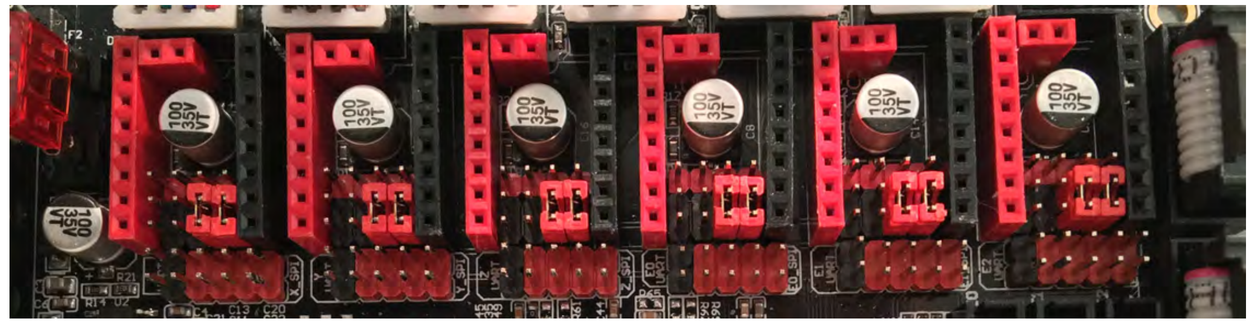
Note: MP6500 driver board does not use external current sense resistors (R_s).

POLOLU MP6500

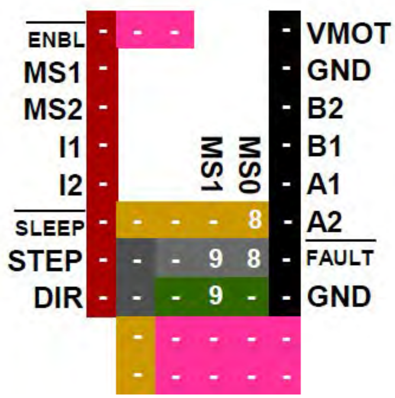
Stand-alone Mode

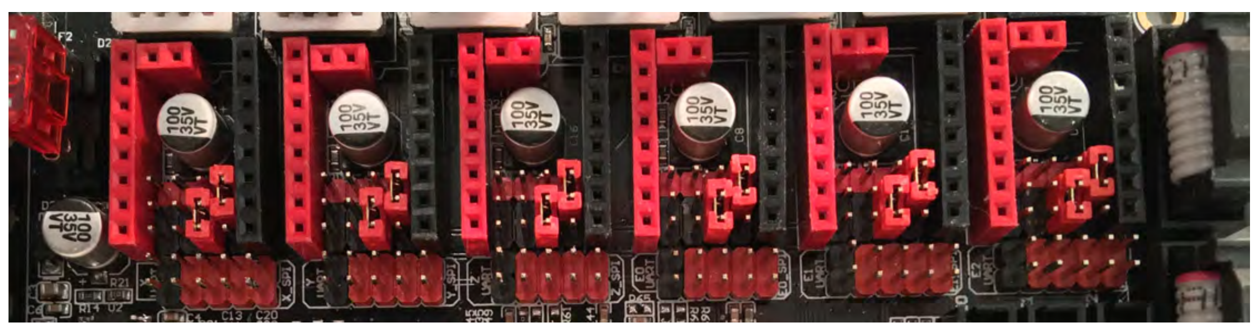
STEP





1 / 2



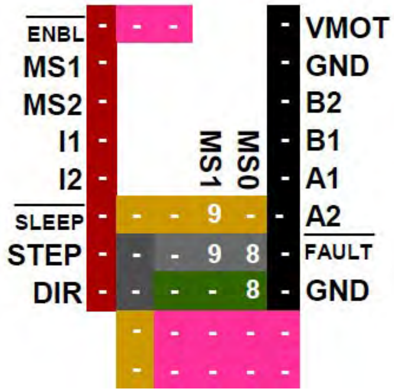


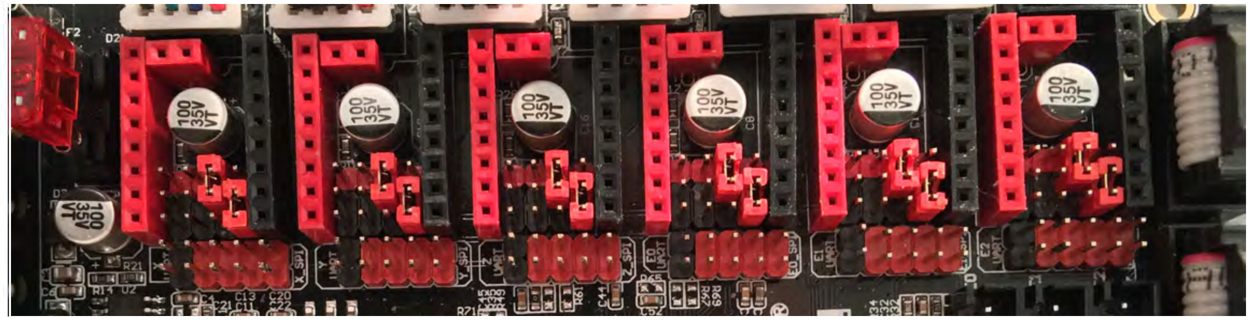
NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

POLOLU MP6500

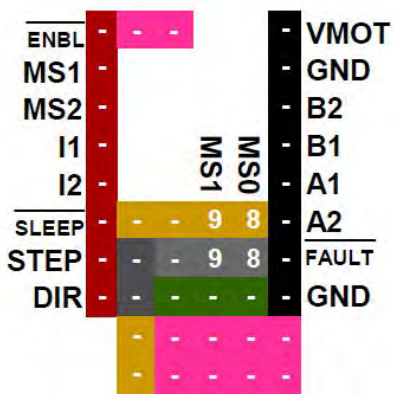
Stand-alone Mode

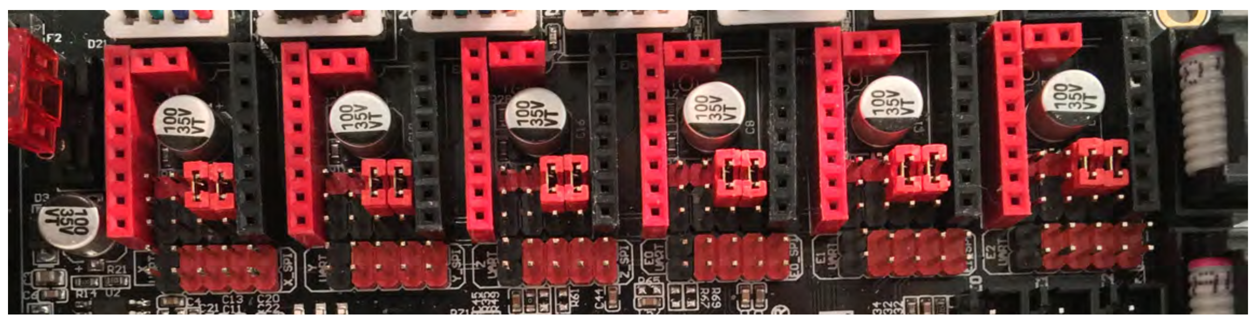
1 / 4





1 / 8



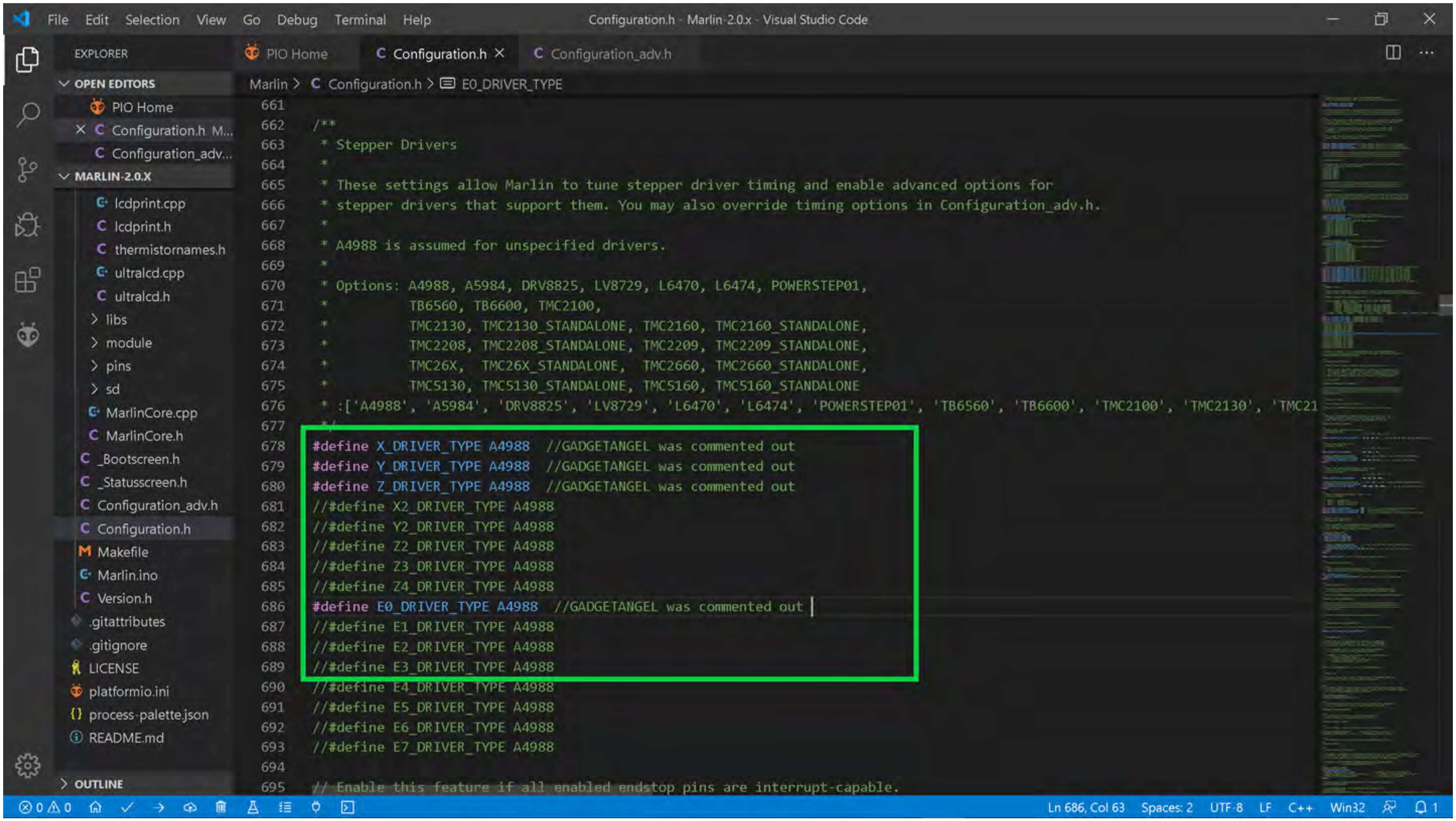


NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

The (latest release of) Marlin Setup for POLOLU MP6500 Drivers

NOTE: **Go to Appendix C**, and then come back here for the changes to Marlin for POLOLU MP6500 stepper motor drivers.

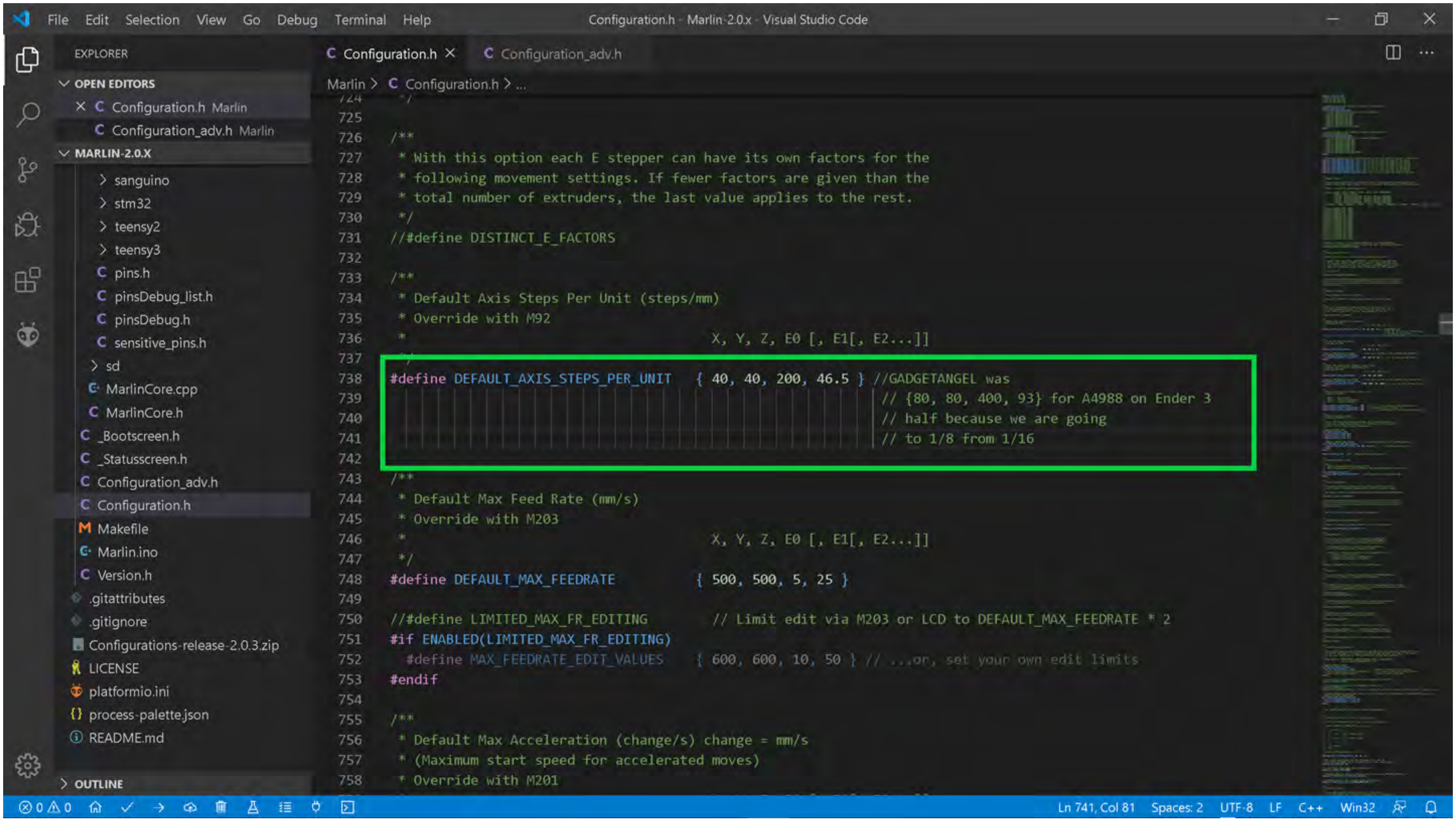
- Change the stepper motor drivers so that Marlin knows you are using POLOLU MP6500 drivers. Change one line for each axis and one for each extruder you will be using. See the picture below for an example of how to use POLOLU MP6500 drivers. When two "//" appear at the beginning of a line that means that line is commented out. To un-comment a line just remove the leading two forward slashes ("//").
- The POLOLU MP6500 is a drop in replacement for the A4988. Since Marlin does not have an option for POLOLU MP6500 we will use the A4988 as the driver type.



- Go to the next page.

The (latest release of) Marlin Setup for POLOLU MP6500 Drivers

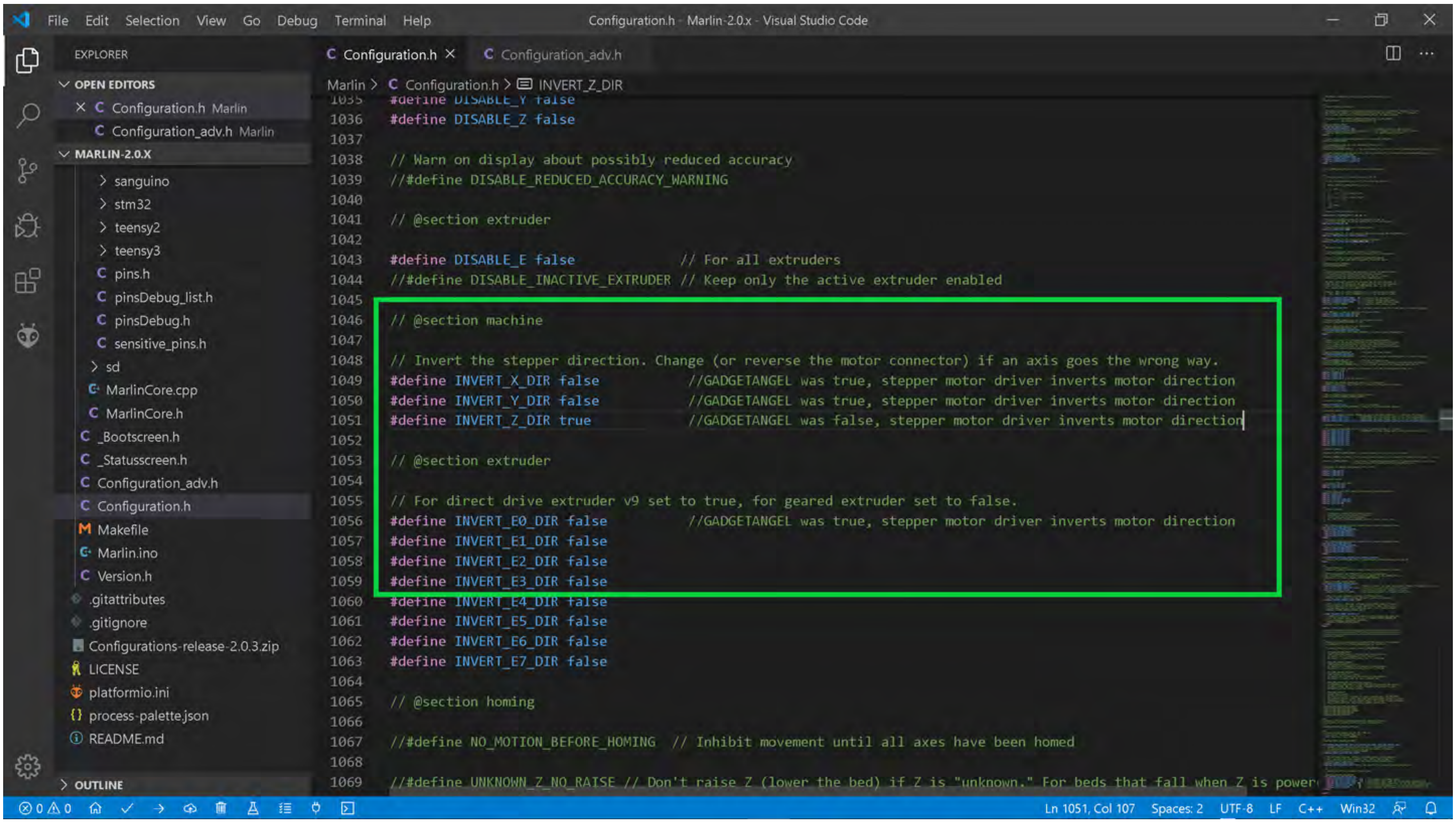
- Since we are changing from A4988 stepper motor drivers on the Ender 3 to for POLOLU MP6500 stepper motor drivers for each axis and the extruder stepper motor driver, we will be going from 1/16 stepping to 1/8 stepping. So we are cutting our STEPS in half. Therefore, we must adjust our "DEFAULT_AXIS_STEPS_PER_UNIT" anytime our STEPS are NOT 1/16. So change "DEFAULT_AXIS_STEPS_PER_UNIT" to {40, 40, 200, 46.5}, as seen in the **GREEN** box below.



- Go to the next page.

The (latest release of) Marlin Setup for POLOLU MP6500 Drivers

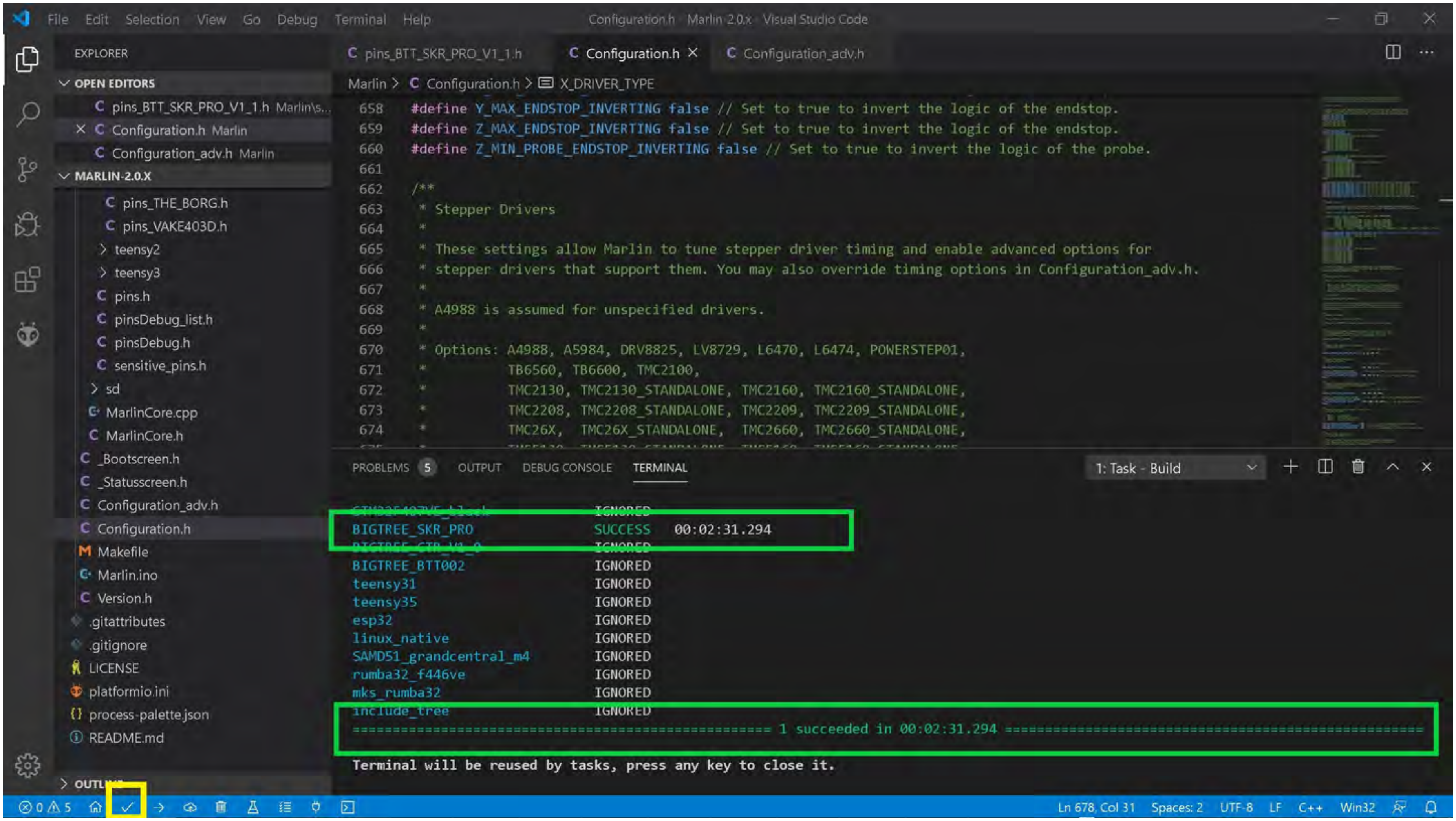
- Optional Step: I cannot find information on the POLOLU MP6500 driver's impact on motor direction. So I provide the below information in case you do need to change the stepper motor direction. If you prefer to change the motor direction with wiring instead of the Marlin firmware, here is a link on how to change the motor direction via the wiring (look for section labeled "Motor moving the wrong direction") **https://reprap.org/wiki/Stepper_wiring**. Other people prefer to change the motor direction in the Marlin firmware. So if you want or need to change the motor direction in Marlin, then if the axis' setting you will be using the MP6500 driver on was "true" change it to "false", as shown in the **GREEN** box below. If the setting was "false", now set it to "true", as shown in the **GREEN** box below



- Go to the next page.

The (latest release of) Marlin Setup for POLOLU MP6500 Drivers

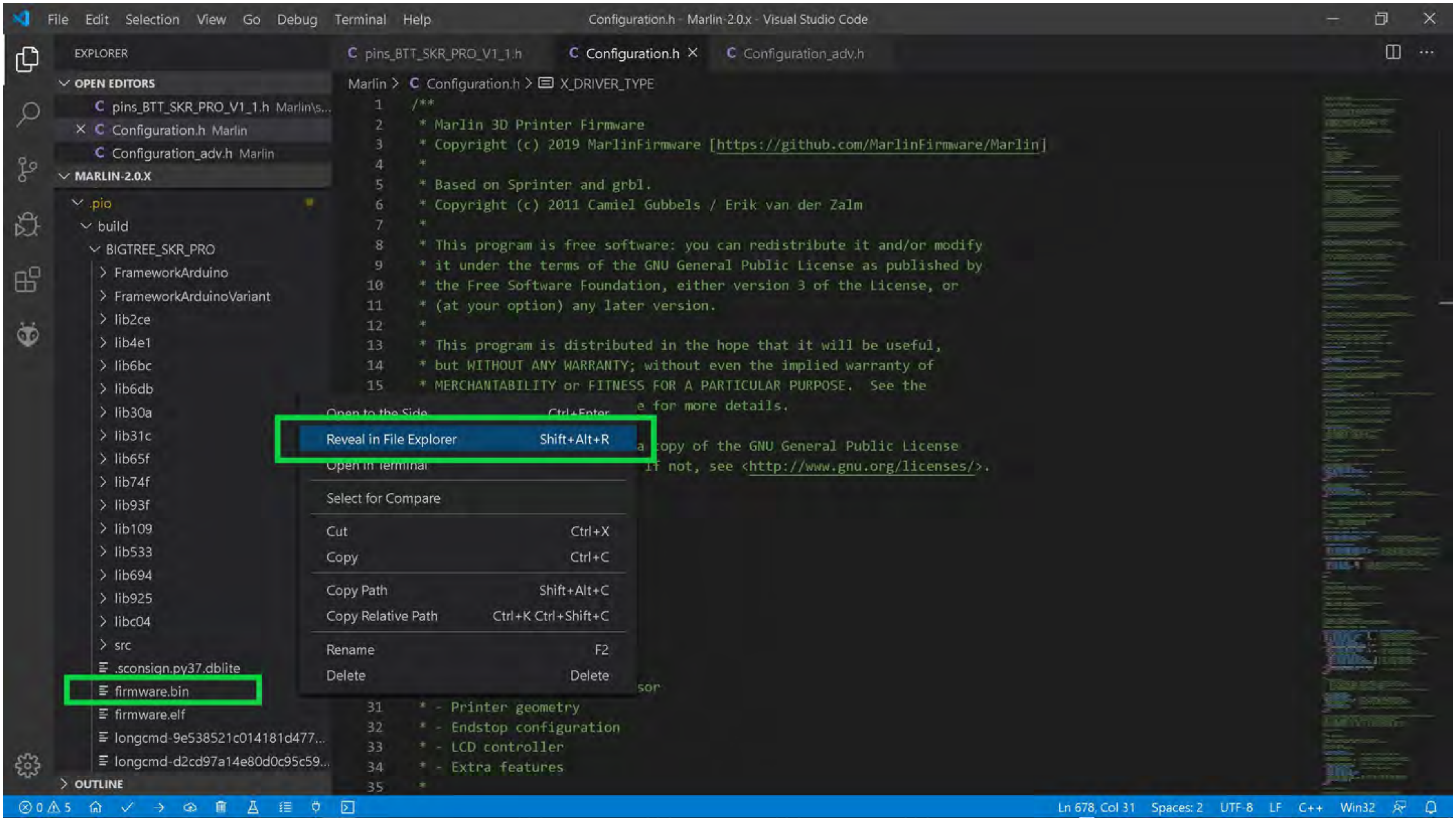
- The end of Marlin setup for POLOLU MP6500 drivers. Now, compile your code by clicking on the check mark, as seen in the **YELLOW** box below. Once the firmware has compiled successfully you will see the following messages, as shown in the **GREEN** boxes below.



- For the SKR PRO V1.1 board you must save the "firmware.bin" file to the micro SD card then place the micro SD card into the micro SD card reader of the SKR PRO V1.1 board and turn on the power to your printer for the NEW firmware to be automatically applied to the SKR PRO MCU. To locate the "firmware.bin" file, and learn how to transfer it to your micro SD card go to the next page.

The (latest release of) Marlin Setup for POLOLU MP6500 Drivers

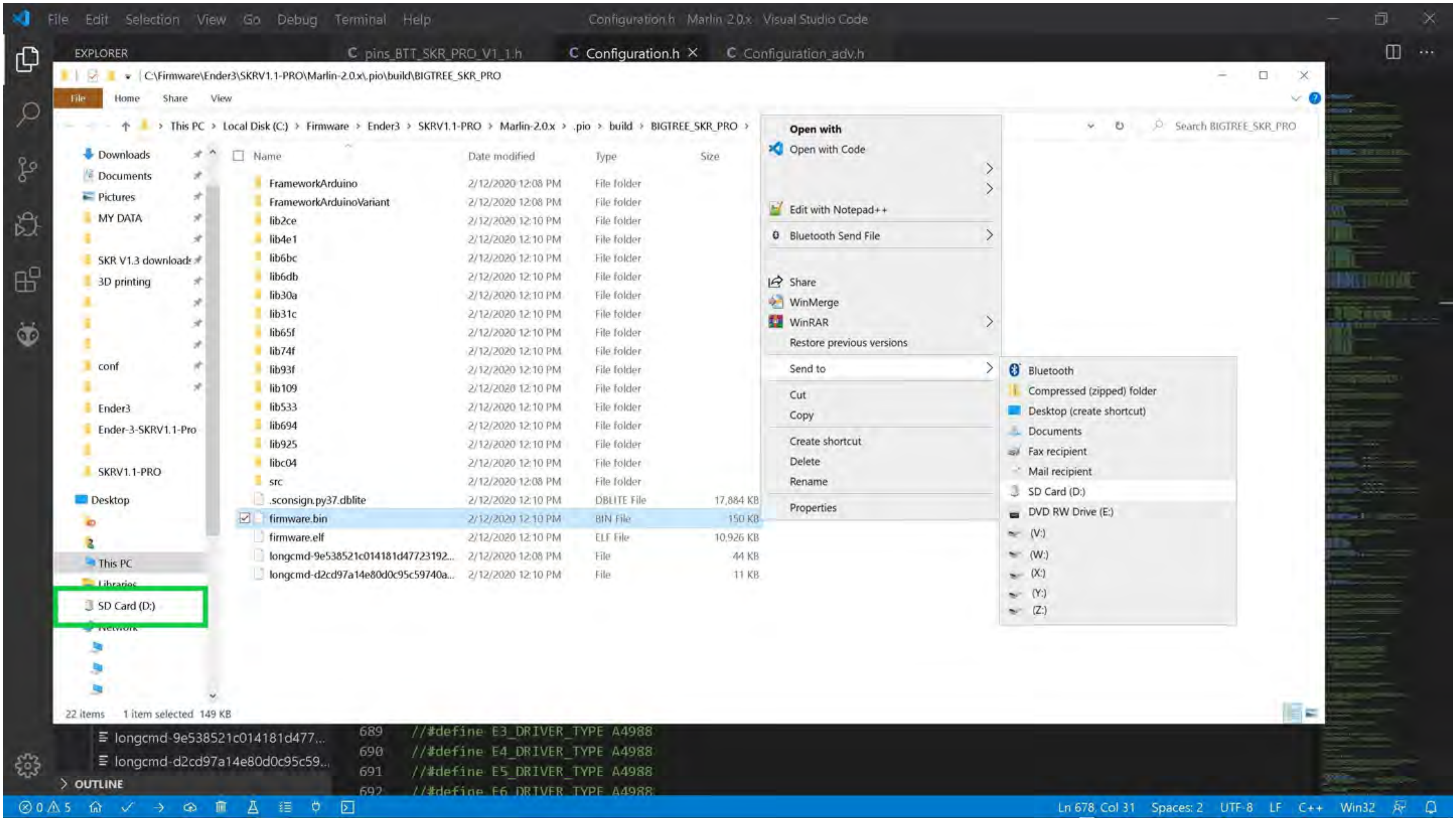
- The "firmware.bin" file can be found by using the left side of the VS code window and going to folder Marlin-2.0.X/.pio/build/ BIGTREE_SKR_PRO/ and right clicking on the "firmware.bin" file name to bring up the content menu. Select "Reveal in File Explorer" to have your Window's machine open a file explorer window.



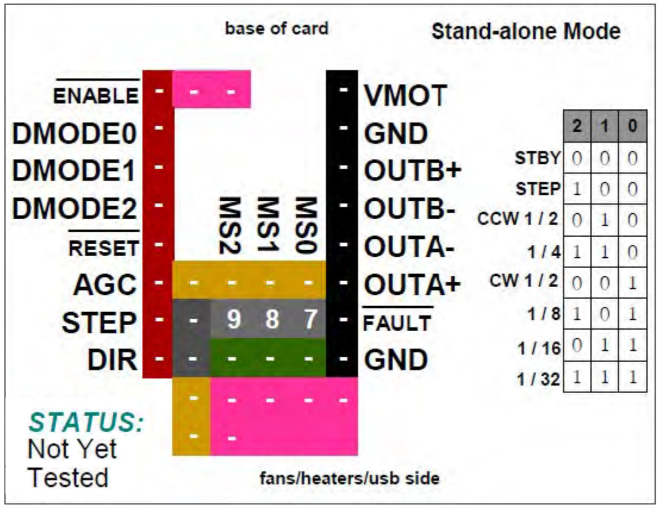
- Go to the next page.

The (latest release of) Marlin Setup for POLOLU MP6500 Drivers

- Ensure your micro SD card, which came with your SKR PRO V1.1 board, with your SKR PRO V1.1 board is in a micro-to-SD-card adapter and plugged into your Window's SD card reader.
    - From the file explorer window right click on the "firmware.bin" file name and select "Send to" from the content menu. For my computer "D:" is my SD card reader. For your computer it may be a different drive letter. The "firmware.bin" file will be written to the SD card and renamed to "FIRMWARE.CUR".



- Right click on the SD card reader, as seen in the **GREEN** box above, and select "Eject" from the context menu. Take the micro SD card out of the adapter and place it into the micro SD card reader of the SKR PRO V1.1 board. Turn on the power to the printer (i.e. the SKR PRO V1.1 board) and the "FIRMWARE.CUR" file will be uploaded to the board. After the upload the firmware file will be renamed to "firmware.bin" and stored on the micro SD card.



POLOLU TB67S249FTG

Note: See the next page for information about location of the current sense resistors and how to set V_{ref} on the stepper motor driver board.

Driver Chip

Pololu

TB67S249FTG

Maximum 32 Subdivision

47V DC
4.5A (peak)

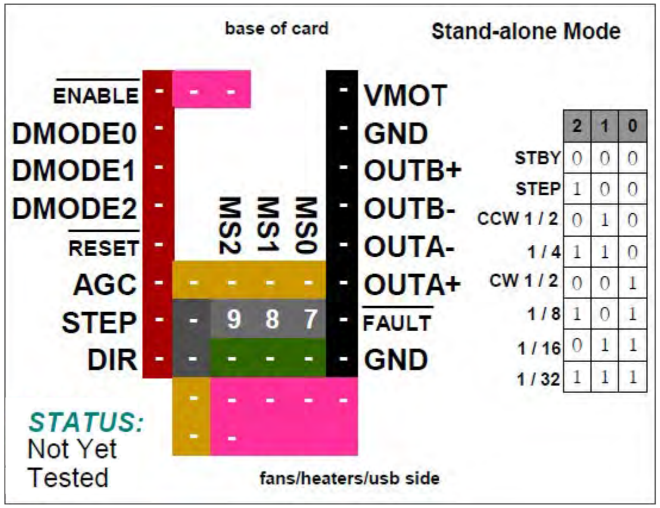
MS2	MS1	MS0	Microstep Resolution
Low	Low	Low	Standby mode (outputs disabled)
High	Low	Low	Full step
Low	High	Low	Non-circular half step ("a")
High	High	Low	1/4 step
Low	Low	High	Circular half step ("b")
High	Low	High	1/8 step
Low	High	High	1/16 step
High	High	High	1/32 step

Driving Current Calculation Formula

$$I_{MAX}=V_{ref}*1.25$$

$$V_{ref}=\frac{I_{MAX}}{1.25}$$

NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

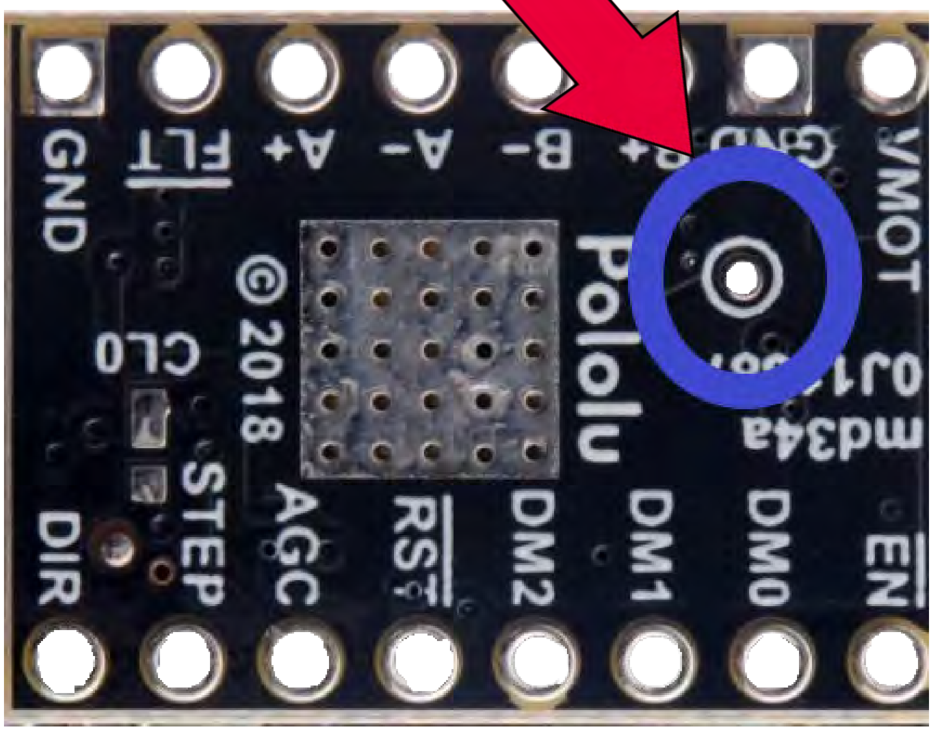


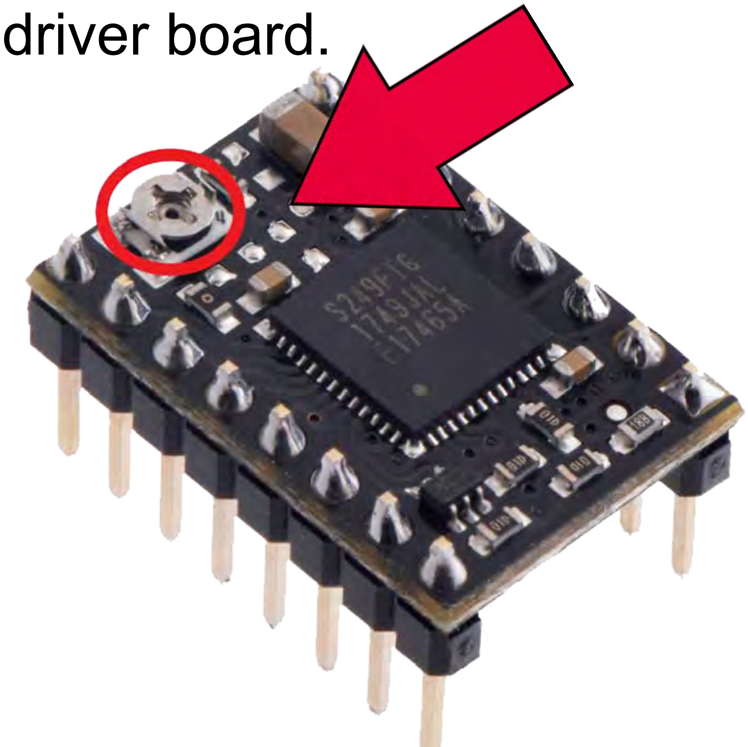
POLOLU TB67S249FTG

Note: Use 90% of the calculated V_{ref} when tuning the stepper driver board.

NOTE: Use the potentiometer (POT) on the top of the board (or use the board's "V_{ref} Test point") to adjust your V_{ref}. See **Appendix A** for instructions on how to set the V_{ref} on a driver board.

Note: "V_{ref} Test point" location is on the Bottom of the driver board, as shown in BLUE





Note: **TB67S249FTG driver board does not use external current sense resistors (R_s).**

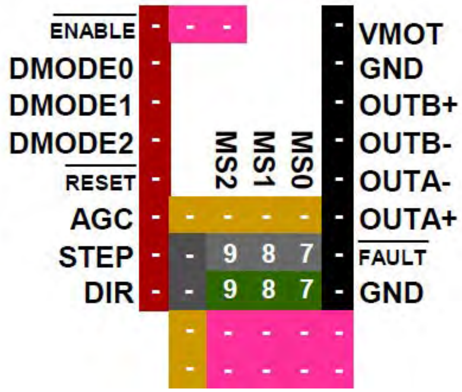
NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

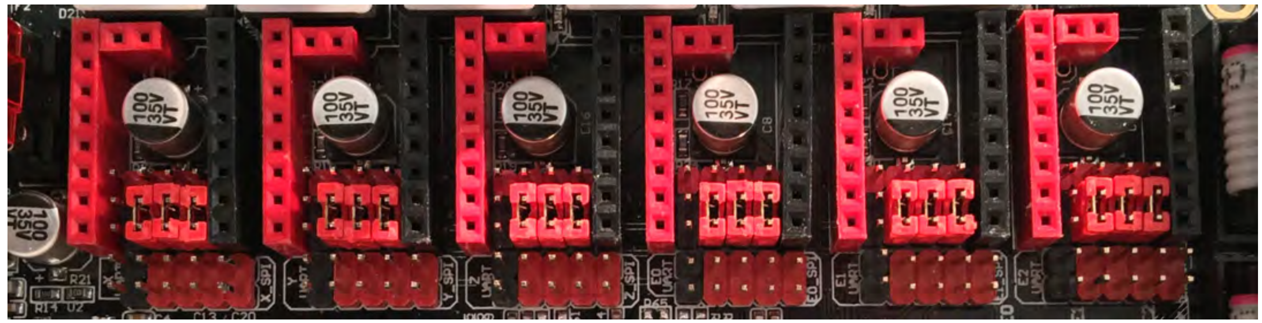
POLOLU TB67S249FTG

Stand-alone Mode

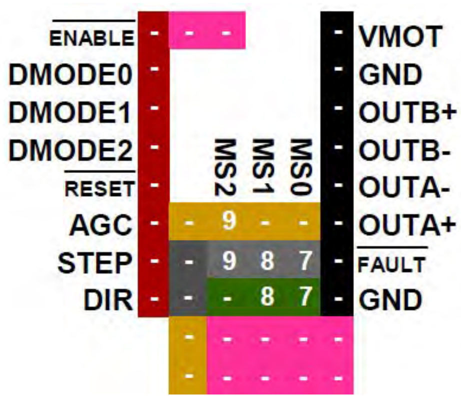
Note: 'Circular half step ("b")' means "clockwise (CW) motor direction with half step resolution". 'Non-circular half step ("a")' means "counterclockwise (CCW) motor direction with half step resolution".

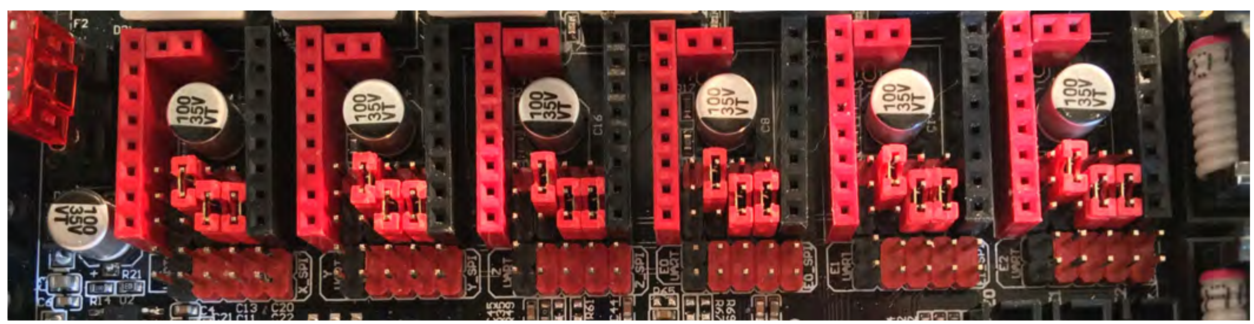
Stand By





STEP





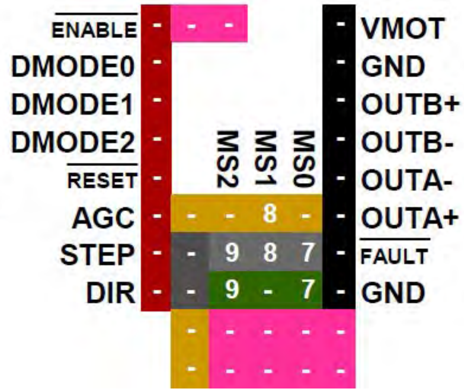
NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

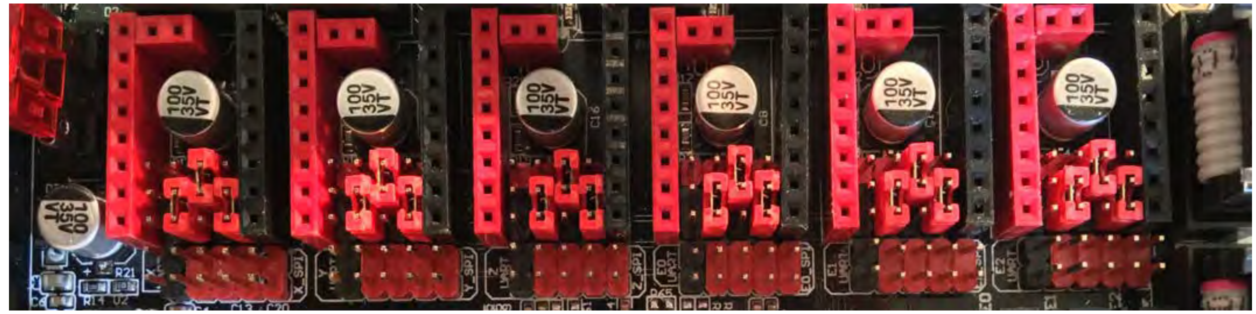
POLOLU TB67S249FTG

Stand-alone Mode

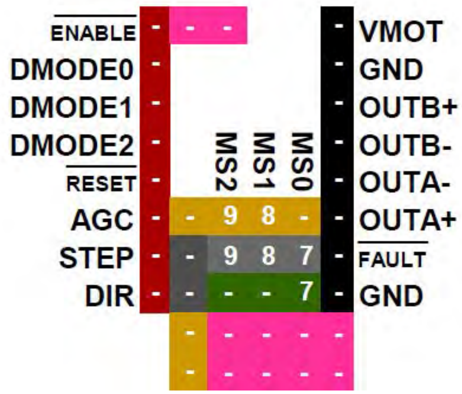
Note: **'Circular half step ("b")' means "clockwise (CW) motor direction with half step resolution". 'Non-circular half step ("a")' means "counterclockwise (CCW) motor direction with half step resolution".**

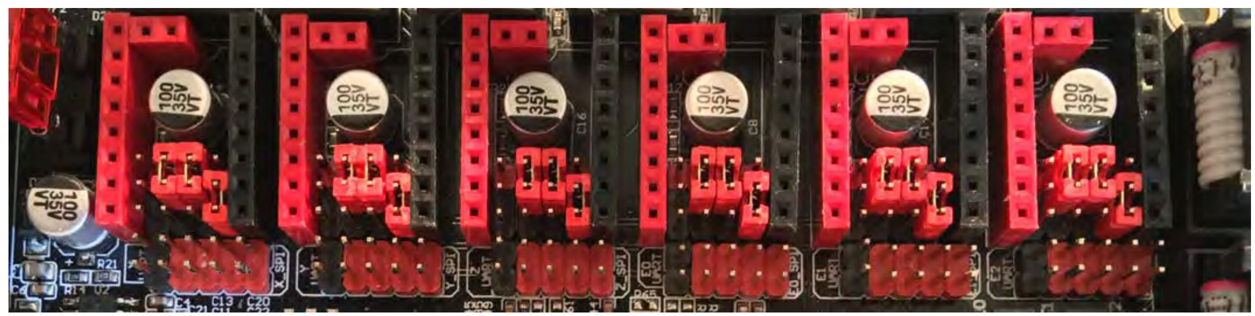
CCW 1 / 2



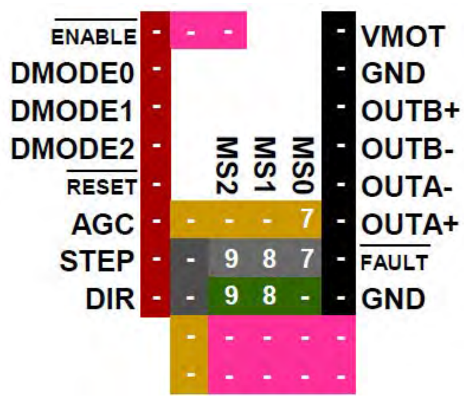


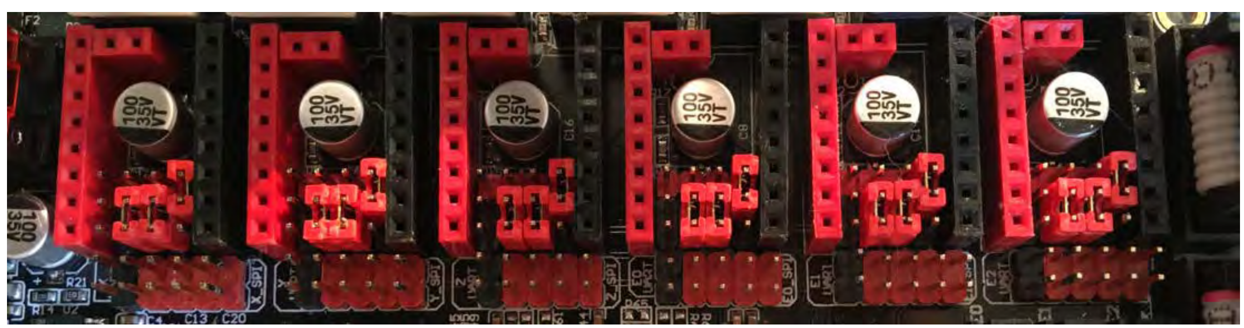
1 / 4





CW 1 / 2





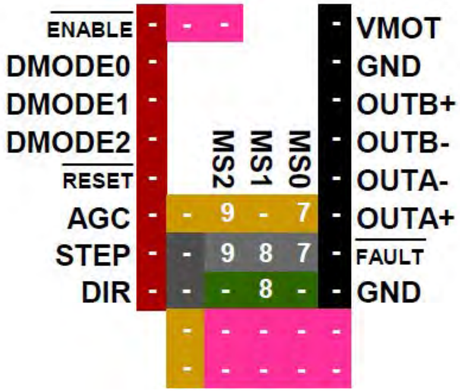
NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

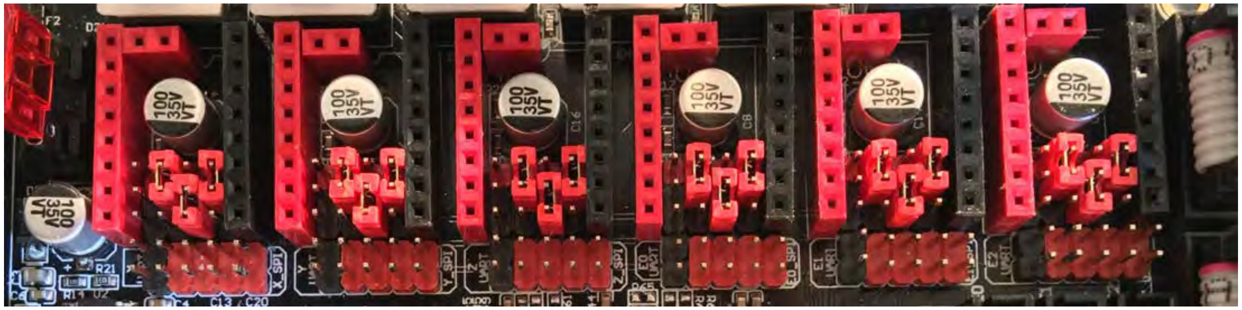
POLOLU TB67S249FTG

Stand-alone Mode

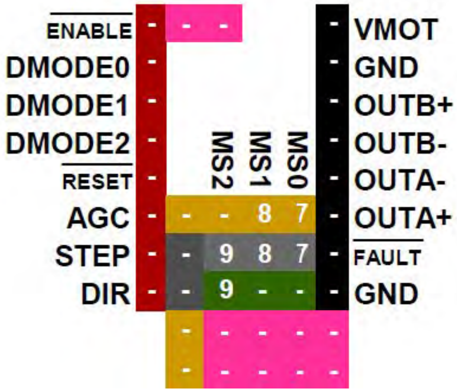
Note: **'Circular half step ("b")' means "clockwise (CW) motor direction with half step resolution". 'Non-circular half step ("a")' means "counterclockwise (CCW) motor direction with half step resolution".**

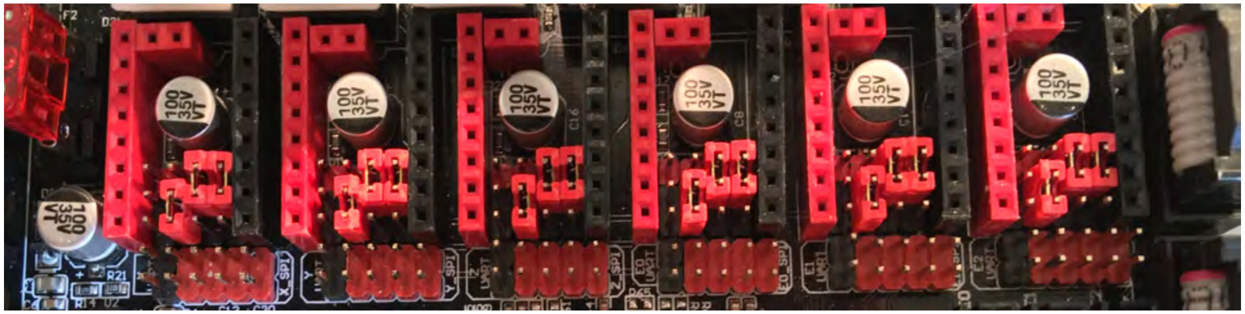
1 / 8



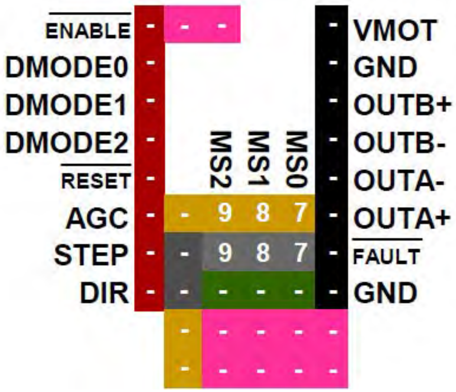


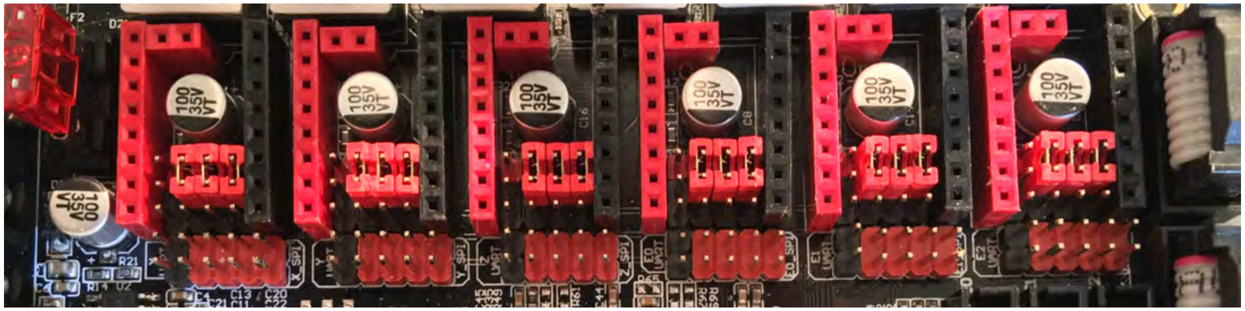
1 / 16





1 / 32



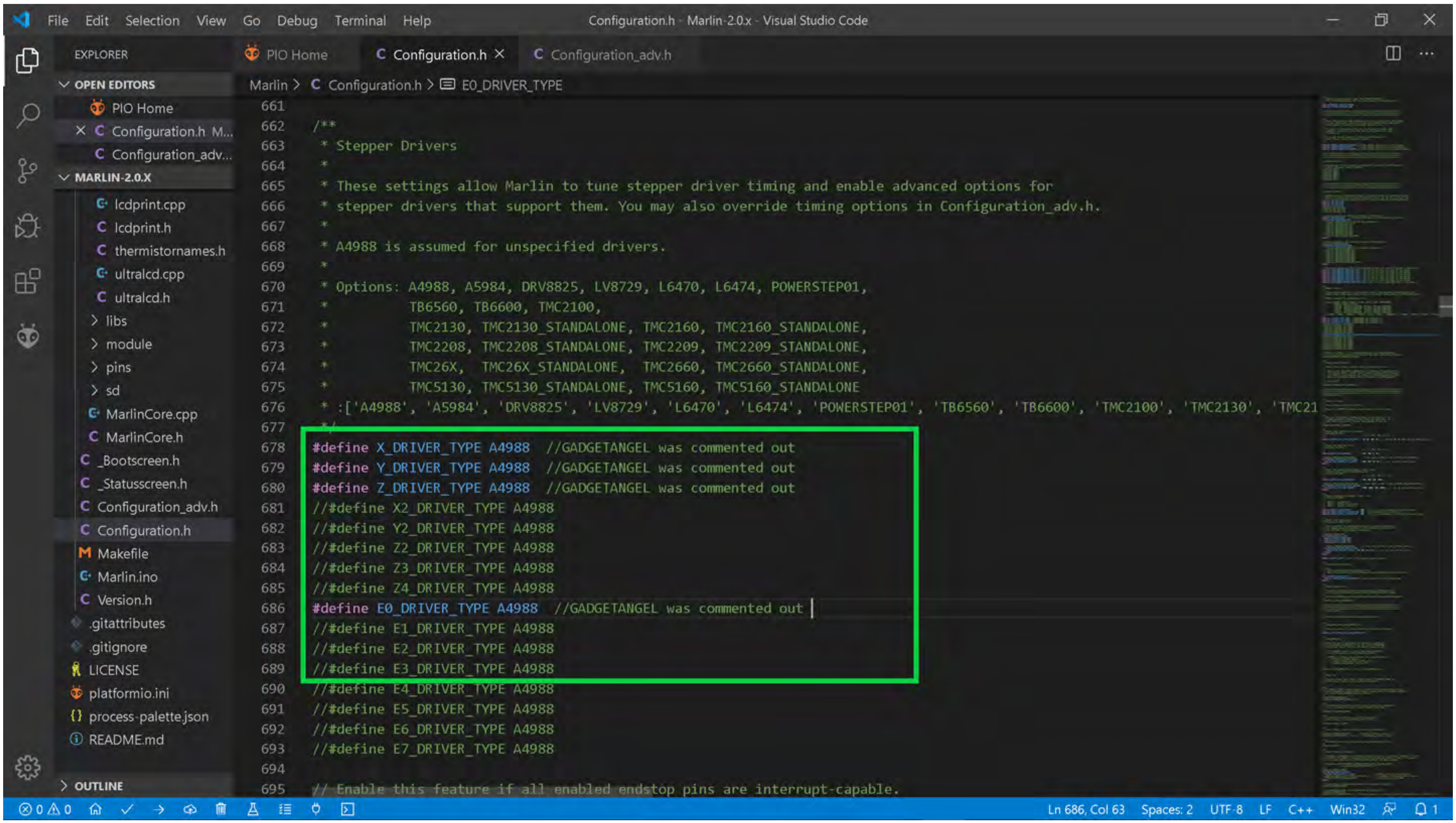


NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

The (latest release of) Marlin Setup for POLOLU TB67S249FTG Drivers

NOTE: **Go to Appendix C,** and then come back here for the changes to Marlin for POLOLU TB67S249FTG stepper motor drivers.

- Change the stepper motor drivers so that Marlin knows you are using POLOLU TB67S249FTG drivers. Change one line for each axis and one for each extruder you will be using. See the picture below for an example of how to use POLOLU TB67S249FTG drivers. When two "//" appear at the beginning of a line that means that line is commented out. To un-comment a line just remove the leading two forward slashes ("//").
- The POLOLU TB67S249FTG is a drop in replacement for the A4988. Since Marlin does not have an option for POLOLU TB67S249FTG we will use the A4988 as the driver type.

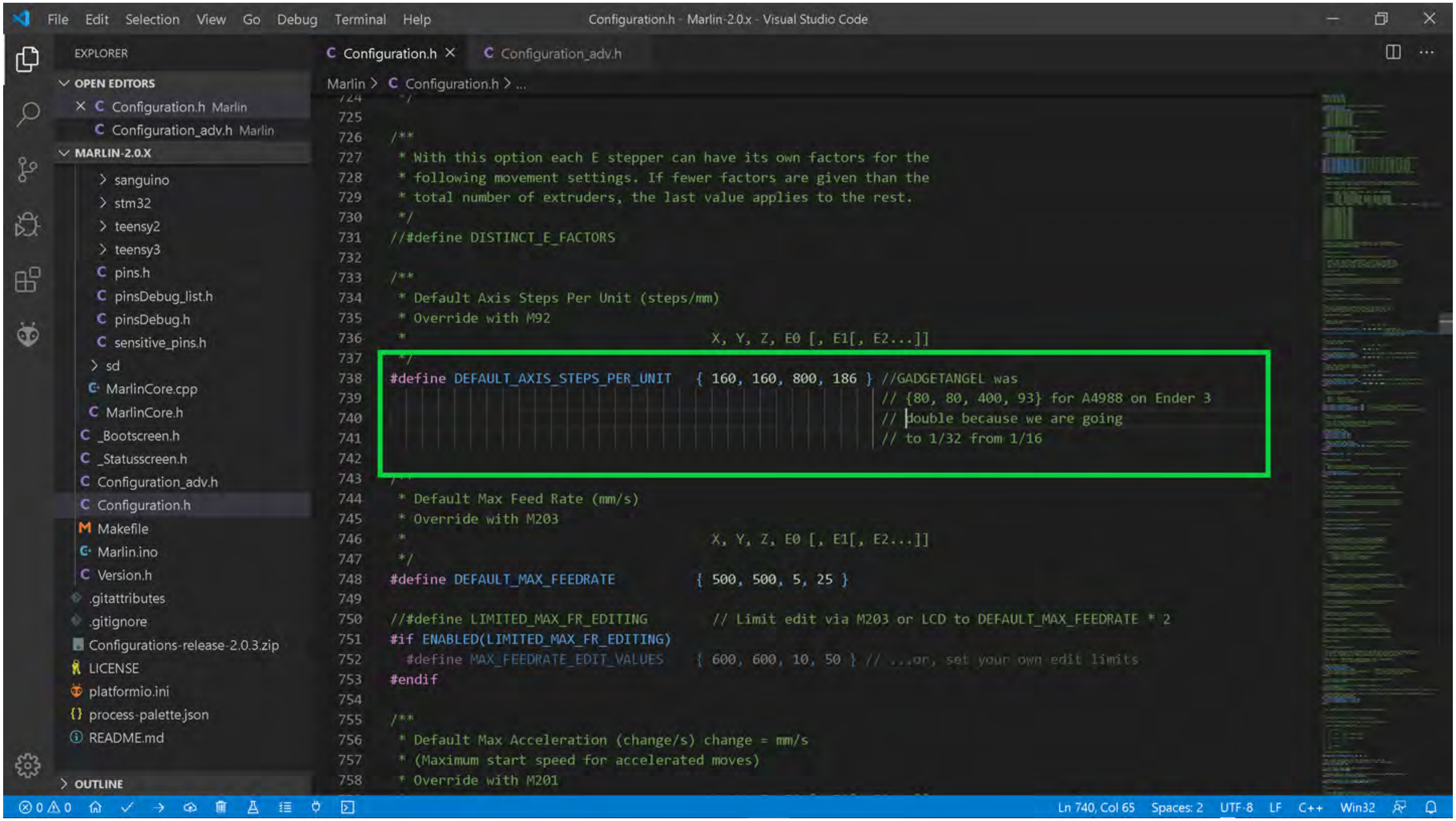


- Go to the next page.

NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

The (latest release of) Marlin Setup for POLOLU TB67S249FTG Drivers

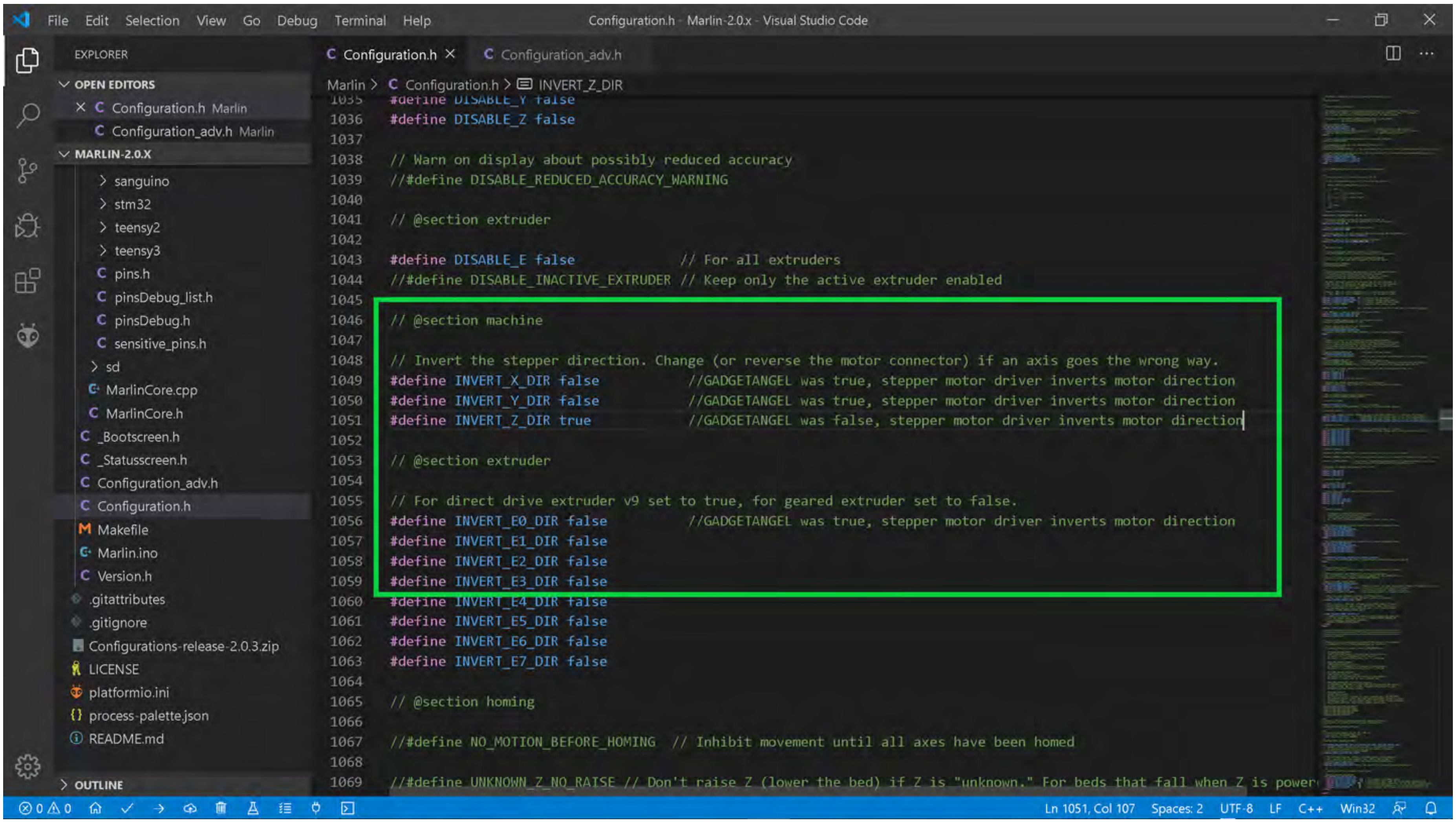
- Since we are changing from A4988 stepper motor drivers on the Ender 3 to for POLOLU TB67S249FTG stepper motor drivers for each axis and the extruder stepper motor driver, we will be going from 1/16 stepping to 1/32 stepping. So we are doubling our STEPS. Therefore, we must adjust our "DEFAULT_AXIS_STEPS_PER_UNIT" anytime our STEPS are NOT 1/16. So change "DEFAULT_AXIS_STEPS_PER_UNIT" to {160, 160, 800, 186}, as seen in the **GREEN** box below.



- Go to the next page.

The (latest release of) Marlin Setup for POLOLU TB67S249FTG Drivers

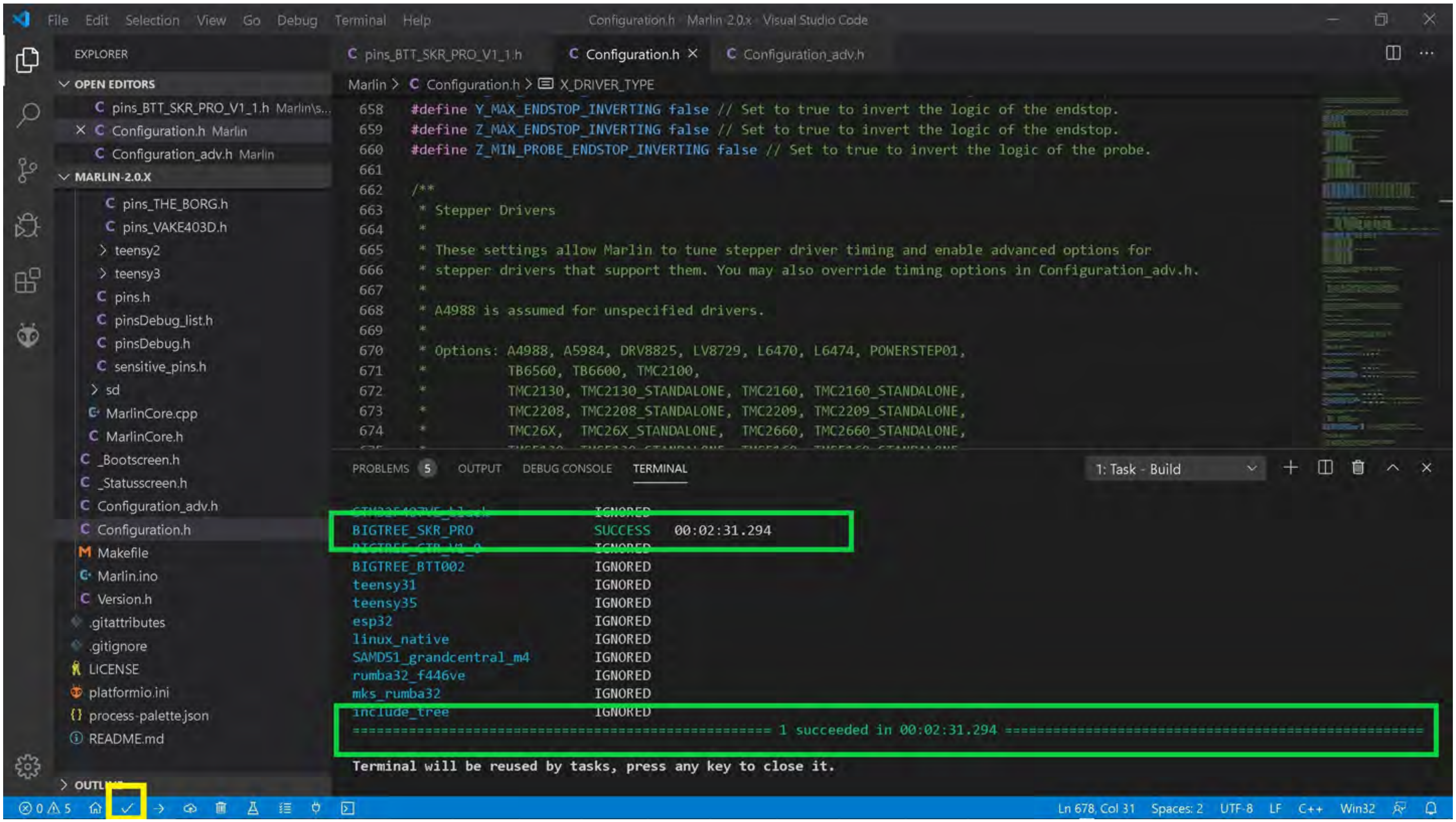
- Optional Step: I cannot find information on the POLOLU TB67S249FTG driver's impact on motor direction. So I provide the below information in case you do need to change the stepper motor direction. If you prefer to change the motor direction with wiring instead of the Marlin firmware, here is a link on how to change the motor direction via the wiring (look for section labeled "Motor moving the wrong direction") **https://reprap.org/wiki/Stepper_wiring**. Other people prefer to change the motor direction in the Marlin firmware. So if you want or need to change the motor direction in Marlin, then if the axis' setting you will be using the TB67S249FTG driver on was "true" change it to "false", as shown in the **GREEN** box below. If the setting was "false", now set it to "true", as shown in the **GREEN** box below



- Go to the next page.

The (latest release of) Marlin Setup for POLOLU TB67S249FTG Drivers

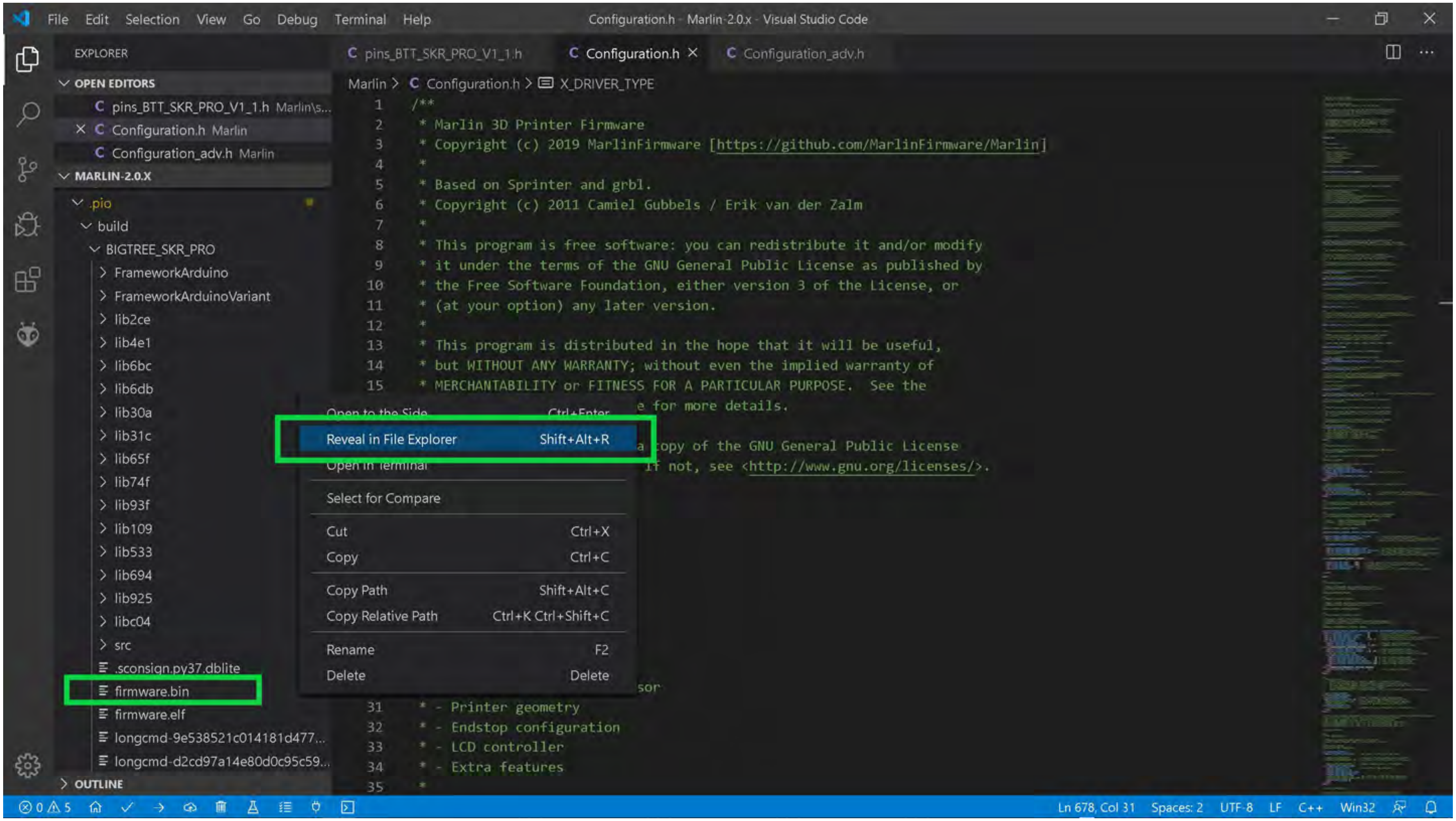
- The end of Marlin setup for POLOLU TB67S249FTG drivers. Now, compile your code by clicking on the check mark, as seen in the **YELLOW** box below. Once the firmware has compiled successfully you will see the following messages, as shown in the **GREEN** boxes below.



- For the SKR PRO V1.1 board you must save the "firmware.bin" file to the micro SD card then place the micro SD card into the micro SD card reader of the SKR PRO V1.1 board and turn on the power to your printer for the NEW firmware to be automatically applied to the SKR PRO MCU. To locate the "firmware.bin" file, and learn how to transfer it to your micro SD card go to the next page.

The (latest release of) Marlin Setup for POLOLU TB67S249FTG Drivers

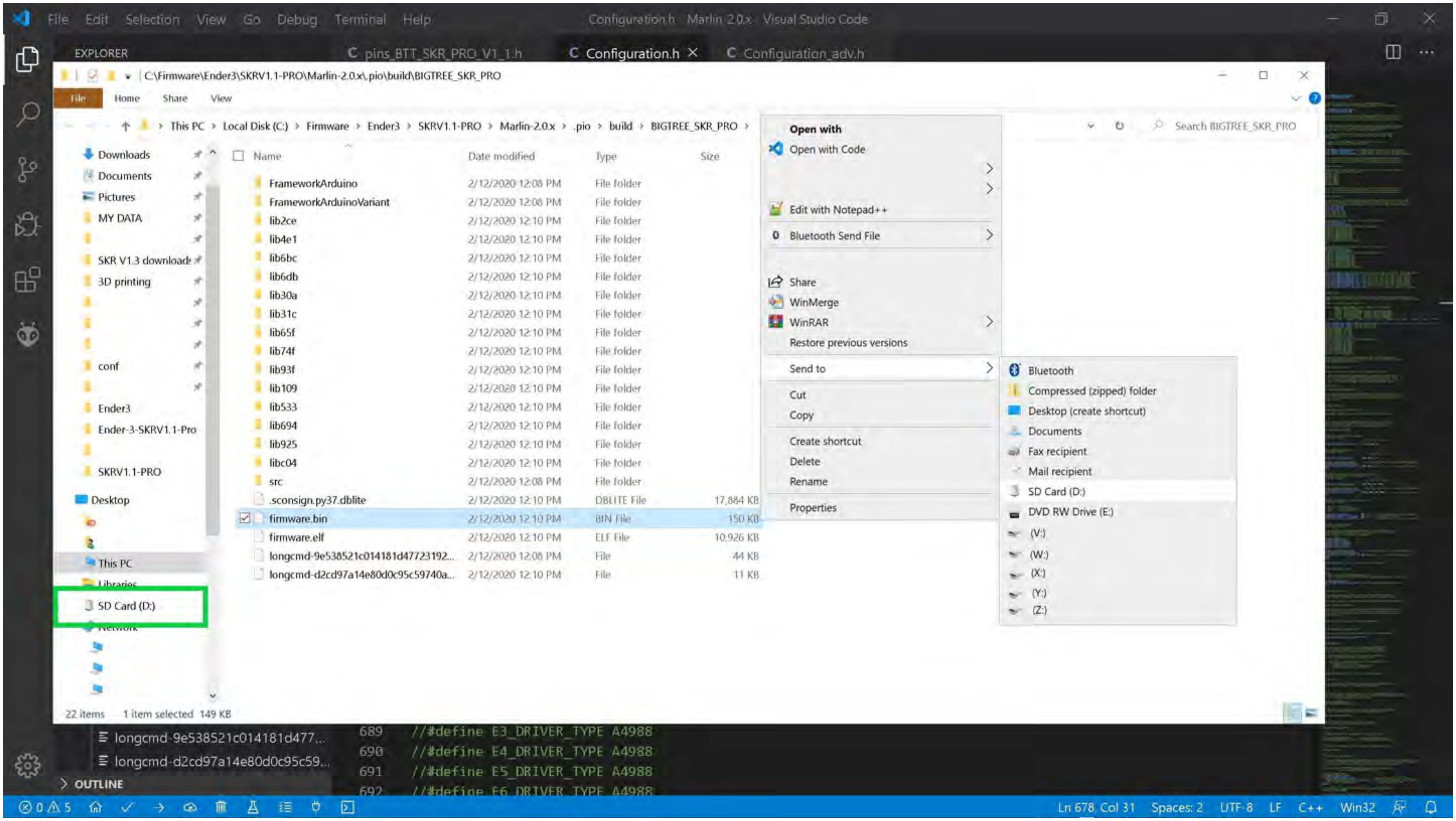
- The "firmware.bin" file can be found by using the left side of the VS code window and going to folder Marlin-2.0.X/.pio/build/ BIGTREE_SKR_PRO/ and right clicking on the "firmware.bin" file name to bring up the content menu. Select "Reveal in File Explorer" to have your Window's machine open a file explorer window.



- Go to the next page.

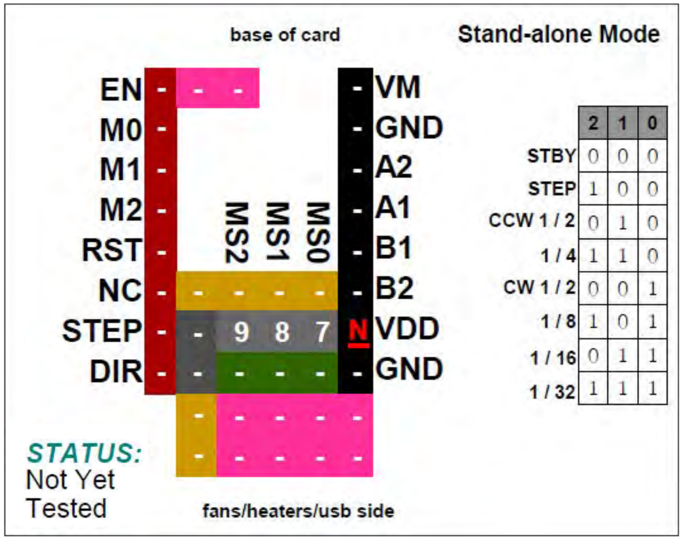
The (latest release of) Marlin Setup for POLOLU TB67S249FTG Drivers

- Ensure your micro SD card, which came with your SKR PRO V1.1 board, with your SKR PRO V1.1 board is in a micro-to-SD-card adapter and plugged into your Window's SD card reader.
  - From the file explorer window right click on the "firmware.bin" file name and select "Send to" from the content menu. For my computer "D:" is my SD card reader. For your computer it may be a different drive letter. The "firmware.bin" file will be written to the SD card and renamed to "FIRMWARE.CUR".



- Right click on the SD card reader, as seen in the **GREEN** box above, and select "Eject" from the context menu. Take the micro SD card out of the adapter and place it into the micro SD card reader of the SKR PRO V1.1 board. Turn on the power to the printer (i.e. the SKR PRO V1.1 board) and the "FIRMWARE.CUR" file will be uploaded to the board. After the upload the firmware file will be renamed to "firmware.bin" and stored on the micro SD card.

BIQU S109

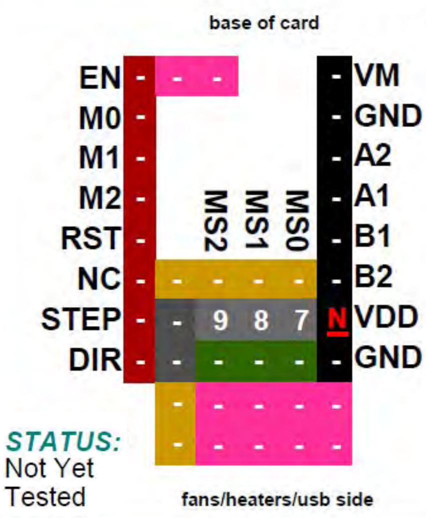


Note: V_{DD} is an Output (**N**), ***on a 3.3 V MCU like SKR PRO V1.1 the V_{DD} must be disconnected!!***

Note: See the next page for information about location of the current sense resistors and how to set V_{ref} on the stepper motor driver board.

Driver Chip	MS2	MS1	MS0	Microstep Resolution
BIQU® S109	Low	Low	Low	Standby mode (outputs disabled)
Maximum 32 Subdivision	High	Low	Low	Full step
25V DC	Low	High	Low	Non-circular half step ("a")
4A (peak)	High	High	Low	1/4 step
	Low	Low	High	Circular half step ("b")
	High	Low	High	1/8 step
	Low	High	High	1/16 step
	High	High	High	1/32 step
Driving Current Calculation Formula $V_{DD} = 5\ V\ DC$ R_S(Typical Sense Resistor)$=0.1\Omega$	$I_{MAX} = V_{ref} * \left(\frac{V_{DD}}{5}\right) * \frac{1}{(5*R_S)}$			$V_{ref} = I_{MAX} * \left(\frac{5}{V_{DD}}\right) * (5*R_S)$

BIQU S109



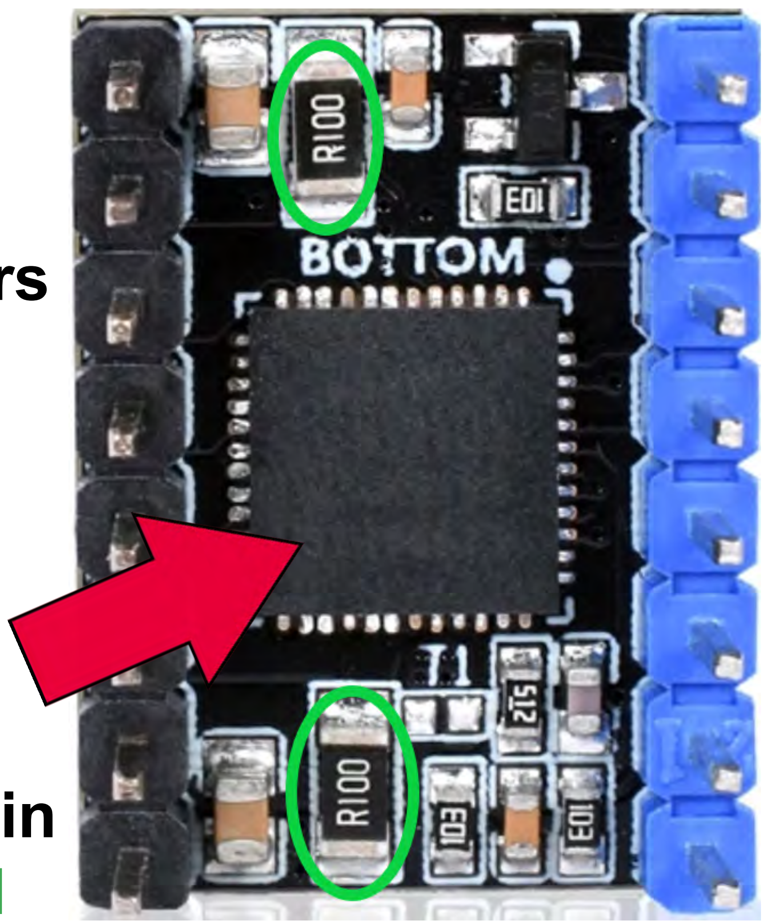
Stand-alone Mode

	2	1	0
STBY	0	0	0
STEP	1	0	0
CCW 1 / 2	0	1	0
1 / 4	1	1	0
CW 1 / 2	0	0	1
1 / 8	1	0	1
1 / 16	0	1	1
1 / 32	1	1	1

Note: Use 90% of the calculated V_{ref} when tuning the stepper driver board. V_{DD} is an Output (**N**), **on a 3.3 V MCU like SKR PRO V1.1 the V_{DD} must be disconnected!**

NOTE: Use the potentiometer (POT) on the top of the board to adjust your V_{ref}. See **Appendix A** for instructions on how to set the V_{ref} on a driver board.

Note: Check your current sense resistors (R_s) values on the driver board, as shown in GREEN



R_s = R050 is 0.05 Ohms
R_s = R068 is 0.068 Ohms
R_s = R100 is 0.1 Ohms
R_s = R150 is 0.15 Ohms
R_s = R200 is 0.2 Ohms
R_s = R220 is 0.22 Ohms

Note: See this video about current sense resistors (R_s) and their possible locations: **https://youtu.be/8wk1elugv5A**

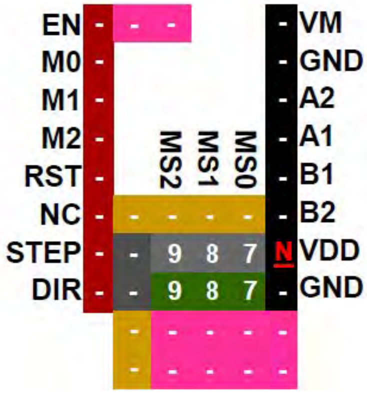
NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

BIQU S109

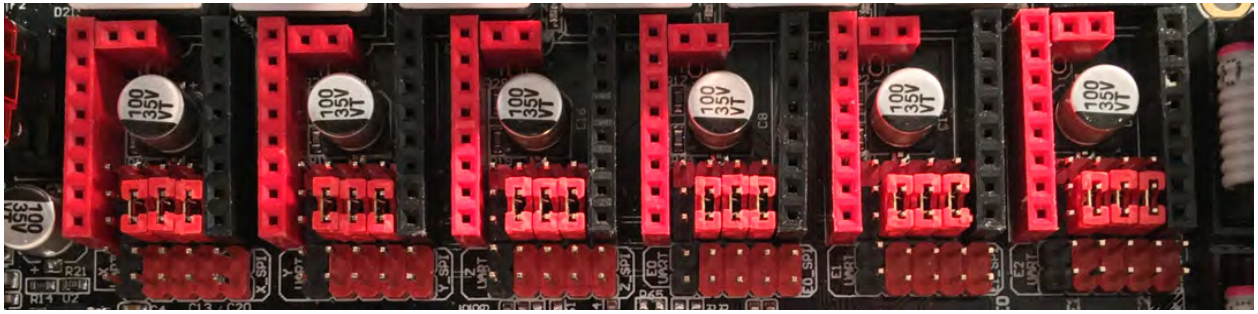
Stand-alone Mode

Note: **'Circular half step ("b")' means "clockwise (CW) motor direction with half step resolution". 'Non-circular half step ("a")' means "counterclockwise (CCW) motor direction with half step resolution".**

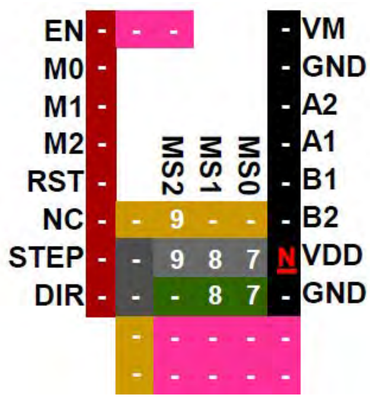
Stand By



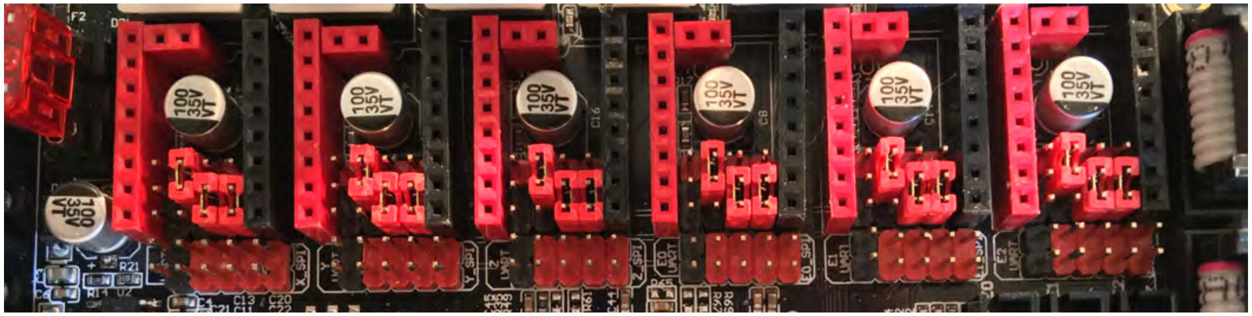
Note: **N**, on a 3.3 V MCU, like SKR PRO V1.1, **MUST** be disconnected!



STEP



Note: **N**, on a 3.3 V MCU, like SKR PRO V1.1, **MUST** be disconnected!



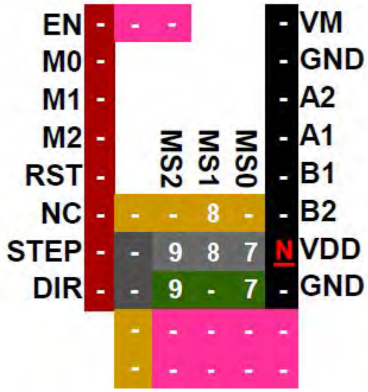
NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

BIQU S109

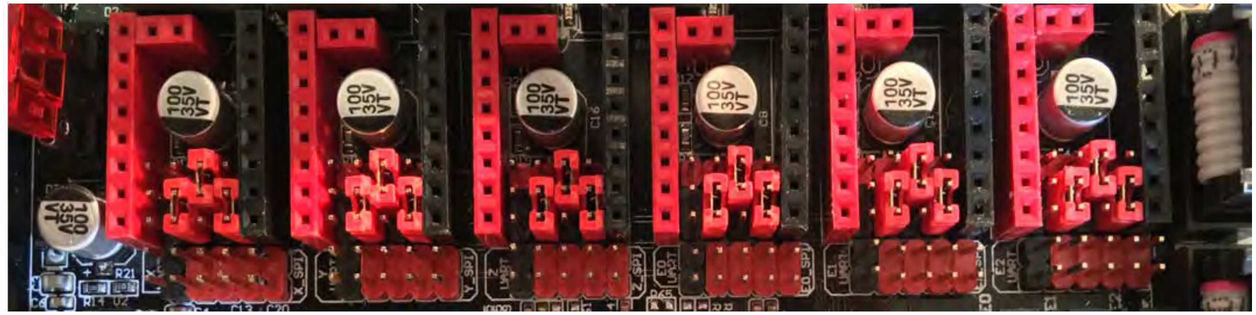
Stand-alone Mode

Note: **'Circular half step ("b")' means "clockwise (CW) motor direction with half step resolution". 'Non-circular half step ("a")' means "counterclockwise (CCW) motor direction with half step resolution".**

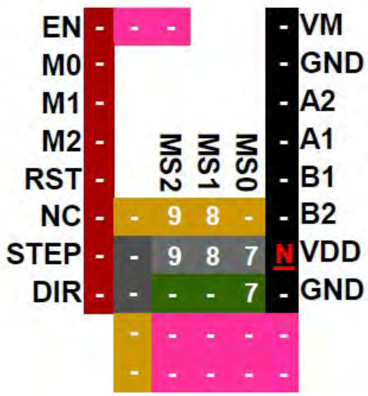
CCW 1 / 2



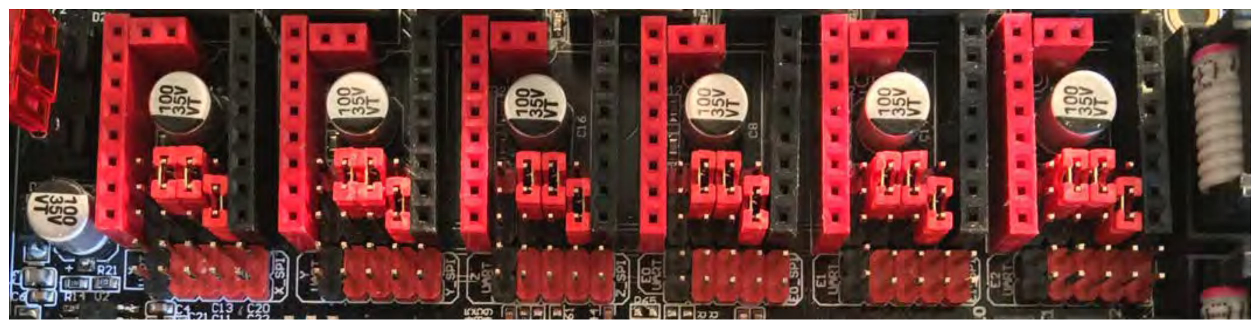
Note: **N**, on a 3.3 V MCU, like SKR PRO V1.1, **MUST** be disconnected!



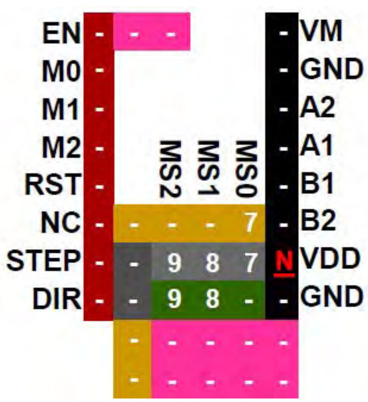
1 / 4



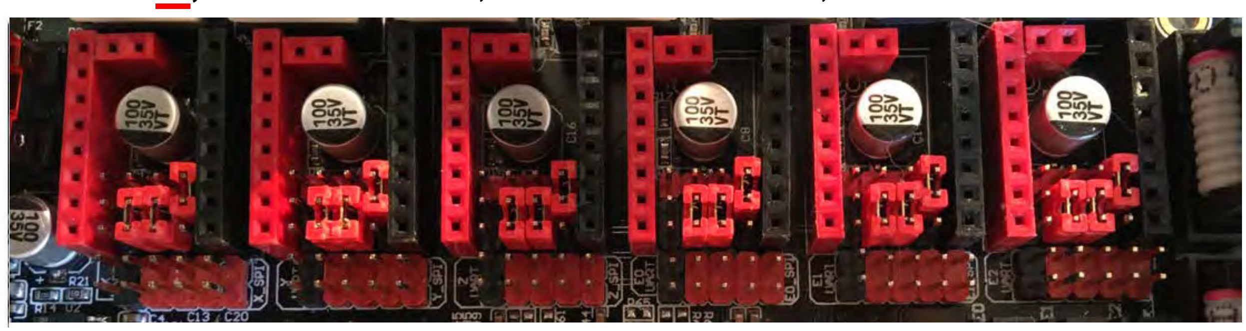
Note: **N**, on a 3.3 V MCU, like SKR PRO V1.1, **MUST** be disconnected!



CW 1 / 2



Note: **N**, on a 3.3 V MCU, like SKR PRO V1.1, **MUST** be disconnected!



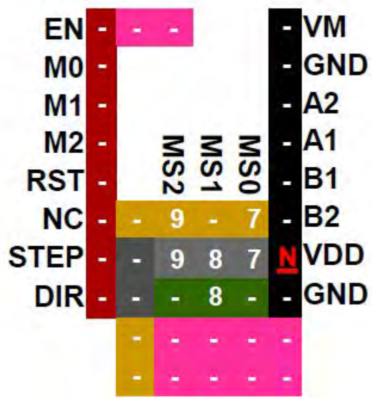
NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

BIQU S109

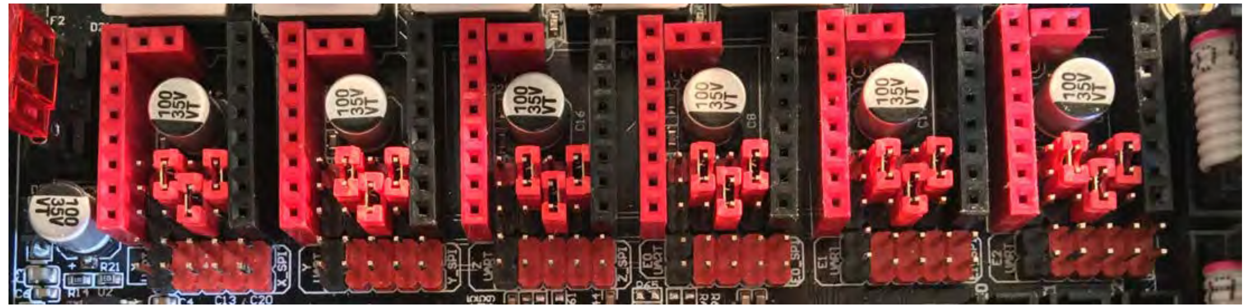
Stand-alone Mode

Note: **'Circular half step ("b")' means "clockwise (CW) motor direction with half step resolution". 'Non-circular half step ("a")' means "counterclockwise (CCW) motor direction with half step resolution".**

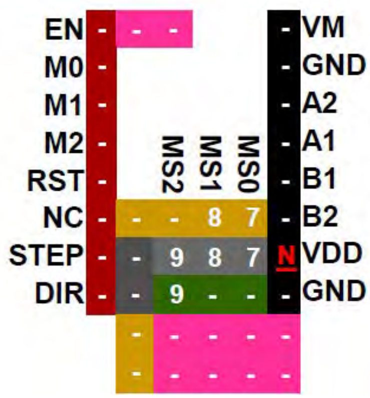
1 / 8



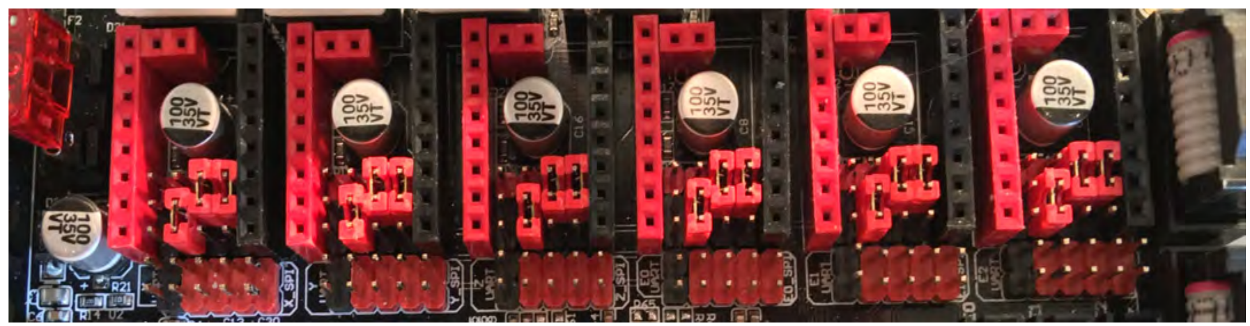
Note: **N**, on a 3.3 V MCU, like SKR PRO V1.1, **MUST** be disconnected!



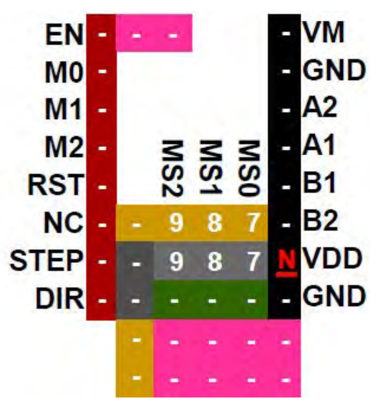
1 / 16



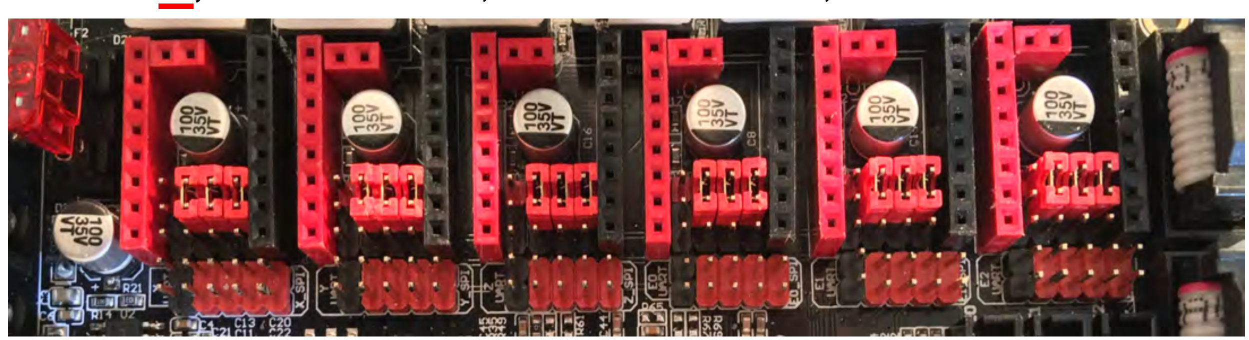
Note: **N**, on a 3.3 V MCU, like SKR PRO V1.1, **MUST** be disconnected!



1 / 32



Note: **N**, on a 3.3 V MCU, like SKR PRO V1.1, **MUST** be disconnected!

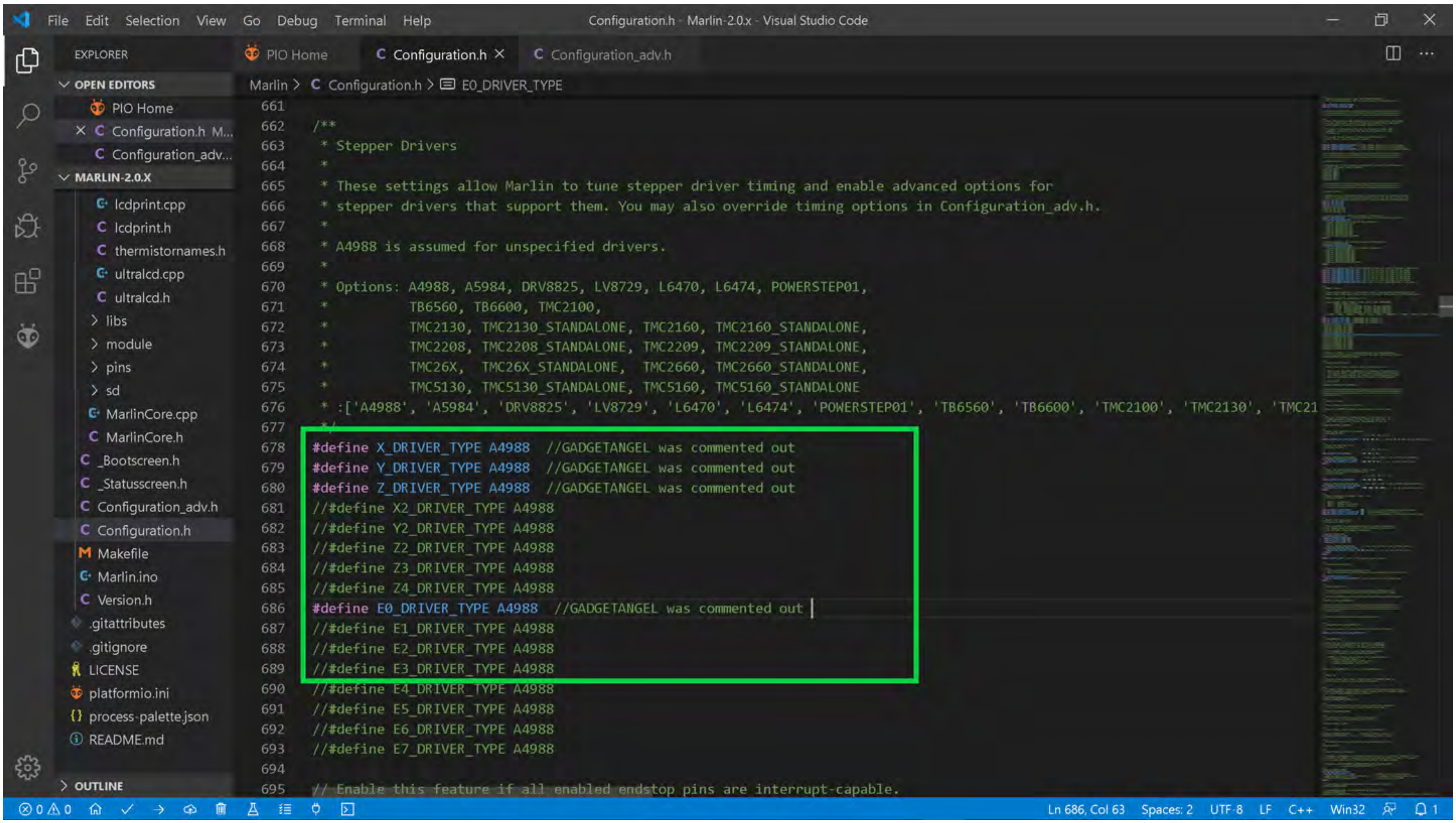


NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

The (latest release of) Marlin Setup for BIQU S109 Drivers

NOTE: **Go to Appendix C,** and then come back here for the changes to Marlin for BIQU S109 stepper motor drivers.

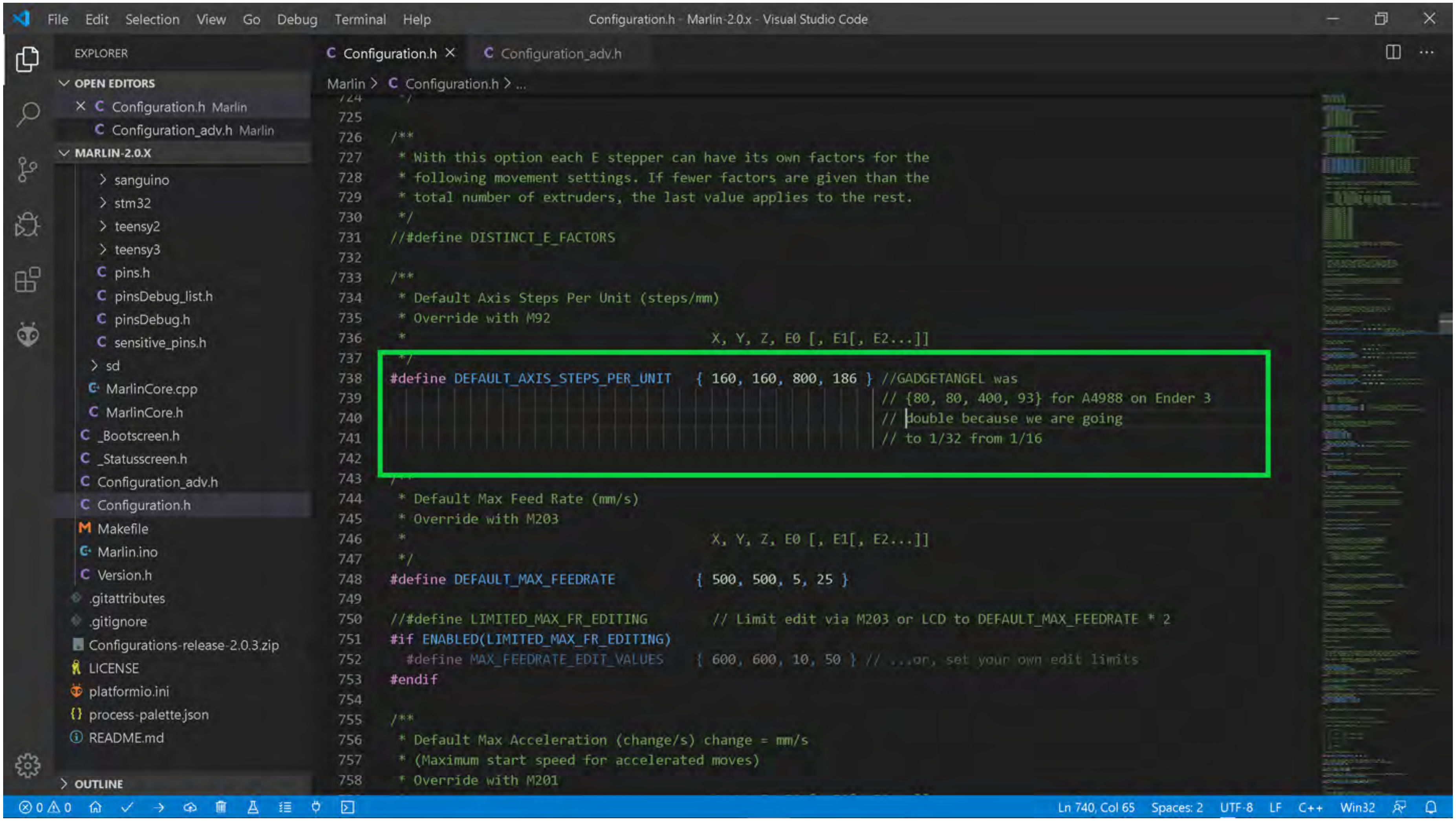
- Change the stepper motor drivers so that Marlin knows you are using BIQU S109 drivers. Change one line for each axis and one for each extruder you will be using. See the picture below for an example of how to use BIQU S109 drivers. When two "//" appear at the beginning of a line that means that line is commented out. To un-comment a line just remove the leading two forward slashes ("//").
- The BIQU S109 is a drop in replacement for the A4988. Since Marlin does not have an option for BIQU S109 we will use the A4988 as the driver type.



- Go to the next page.

The (latest release of) Marlin Setup for BIQU S109 Drivers

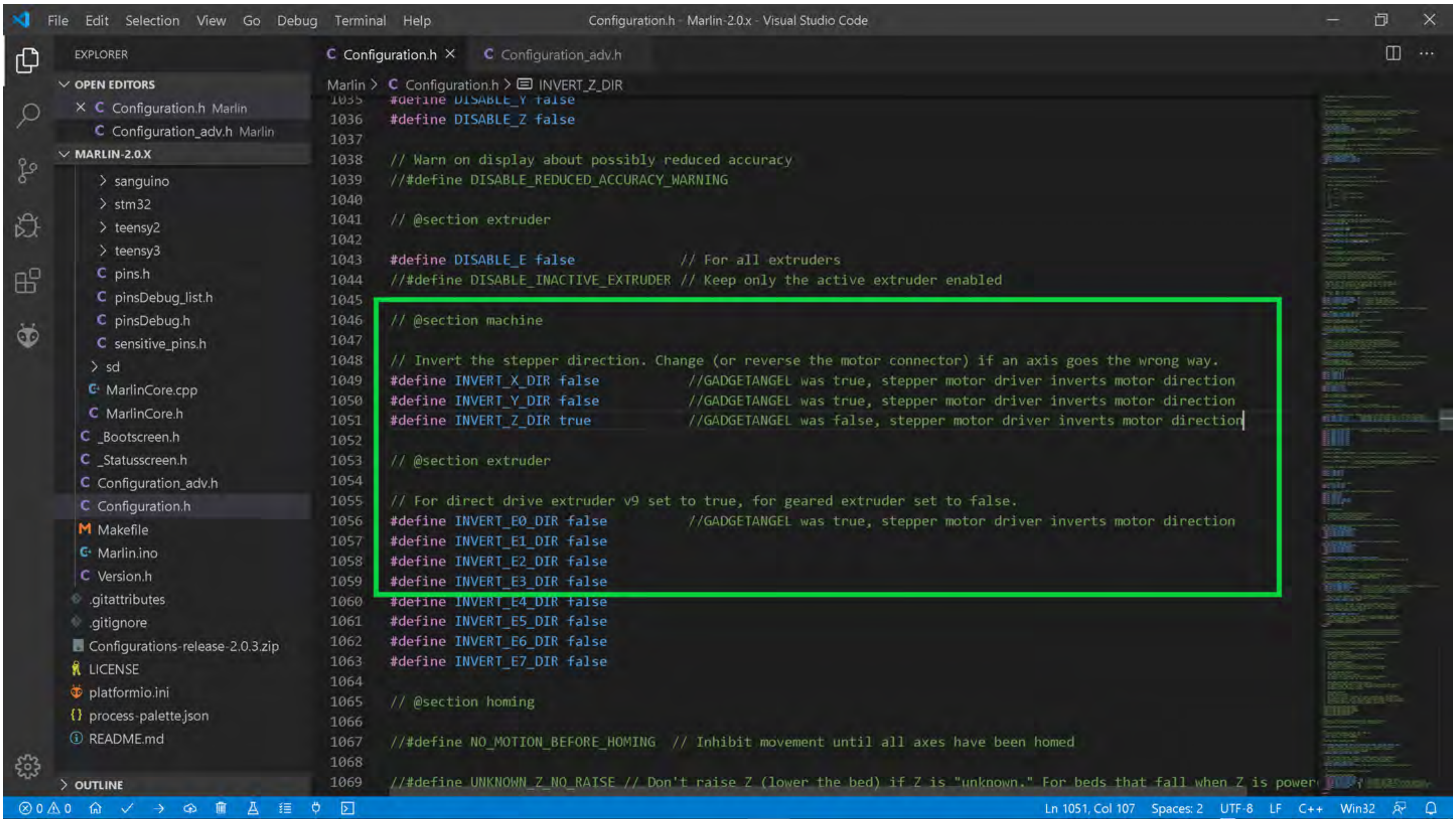
- Since we are changing from A4988 stepper motor drivers on the Ender 3 to for BIQU S109 stepper motor drivers for each axis and the extruder stepper motor driver, we will be going from 1/16 stepping to 1/32 stepping. So we are doubling our STEPS. Therefore, we must adjust our "DEFAULT_AXIS_STEPS_PER_UNIT" anytime our STEPS are NOT 1/16. So change "DEFAULT_AXIS_STEPS_PER_UNIT" to {160, 160, 800, 186}, as seen in the **GREEN** box below.



- Go to the next page.

The (latest release of) Marlin Setup for BIQU S109 Drivers

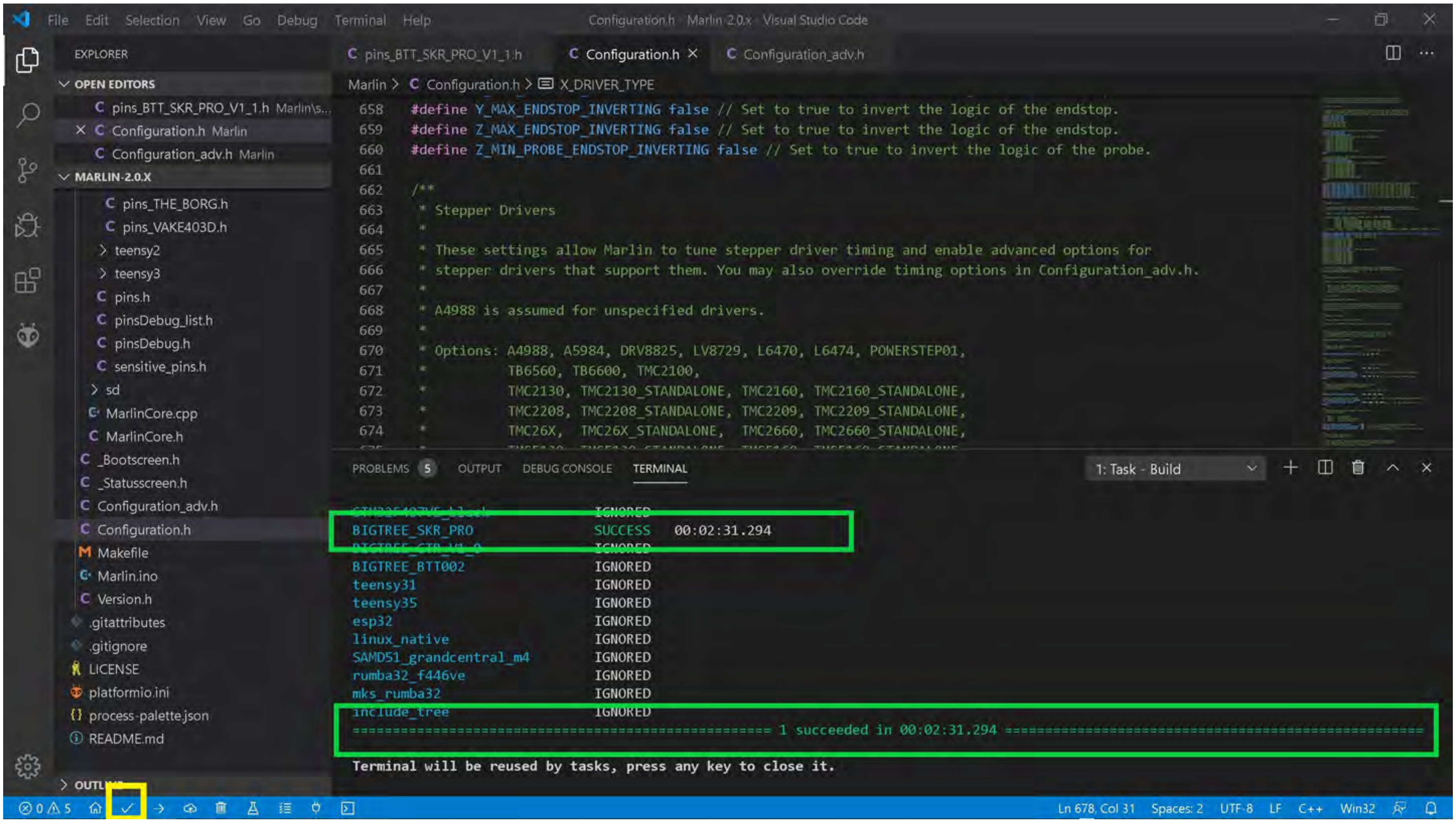
- Optional Step: I cannot find information on the BIQU S109 driver's impact on motor direction. So I provide the below information in case you do need to change the stepper motor direction. If you prefer to change the motor direction with wiring instead of the Marlin firmware, here is a link on how to change the motor direction via the wiring (look for section labeled "Motor moving the wrong direction") **https://reprap.org/wiki/Stepper_wiring**. Other people prefer to change the motor direction in the Marlin firmware. So if you want or need to change the motor direction in Marlin, then if the axis' setting you will be using the S109 driver on was "true" change it to "false", as shown in the **GREEN** box below. If the setting was "false", now set it to "true", as shown in the **GREEN** box below



- Go to the next page.

The (latest release of) Marlin Setup for BIQU S109 Drivers

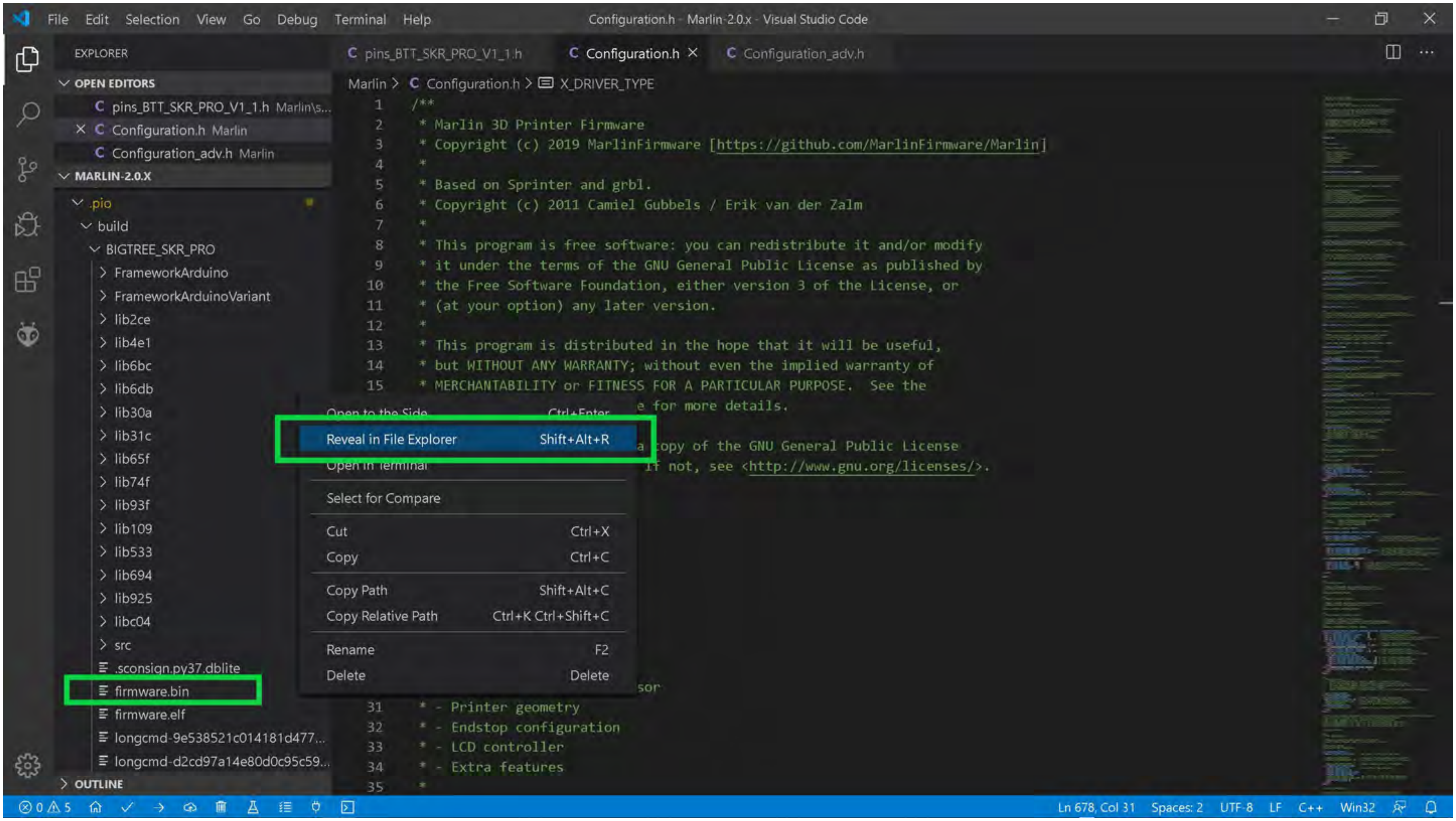
- The end of Marlin setup for BIQU S109 drivers. Now, compile your code by clicking on the check mark, as seen in the **YELLOW** box below. Once the firmware has compiled successfully you will see the following messages, as shown in the **GREEN** boxes below.



- For the SKR PRO V1.1 board you must save the "firmware.bin" file to the micro SD card then place the micro SD card into the micro SD card reader of the SKR PRO V1.1 board and turn on the power to your printer for the NEW firmware to be automatically applied to the SKR PRO MCU. To locate the "firmware.bin" file, and learn how to transfer it to your micro SD card go to the next page.

The (latest release of) Marlin Setup for BIQU S109 Drivers

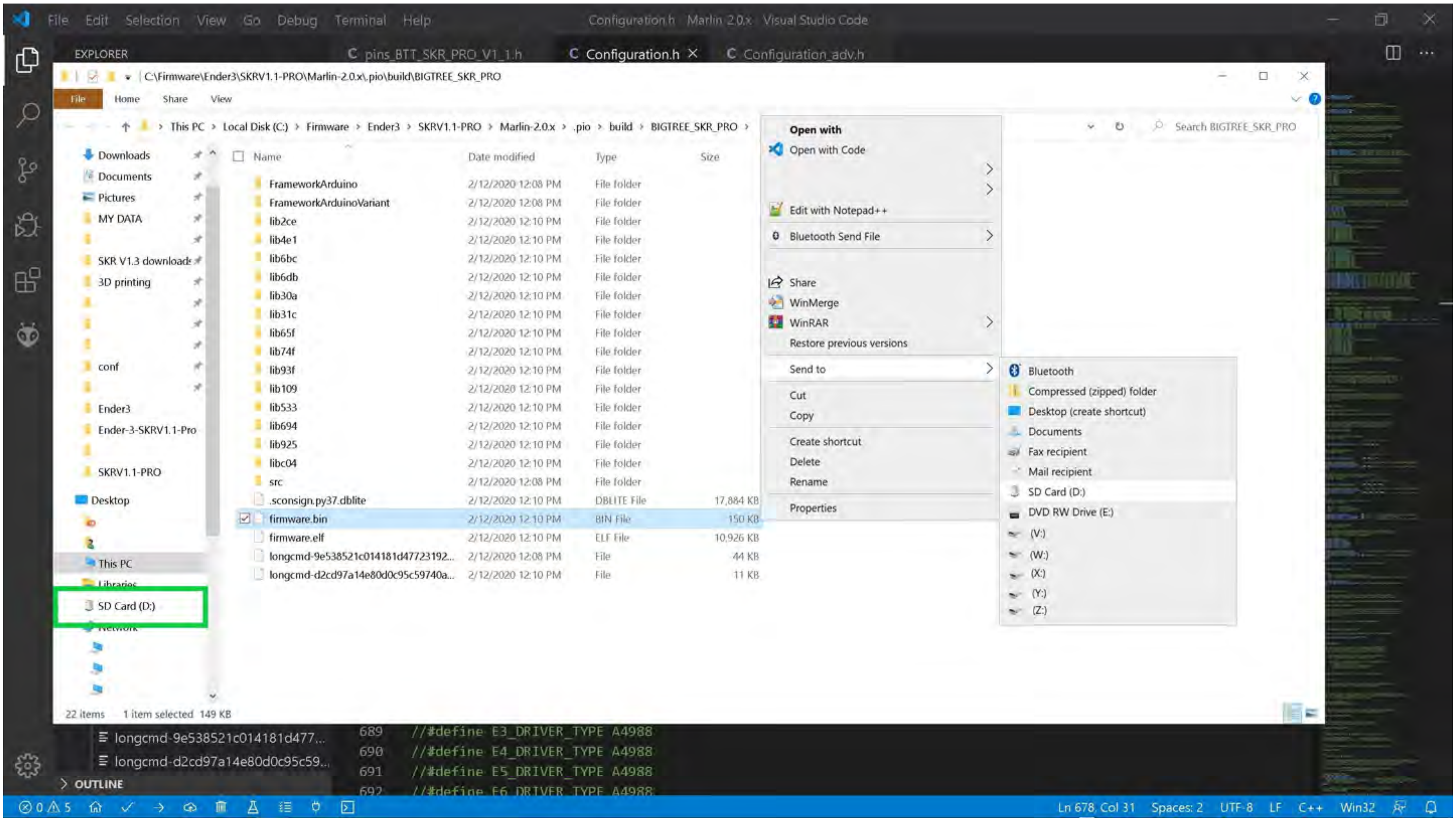
- The "firmware.bin" file can be found by using the left side of the VS code window and going to folder Marlin-2.0.X/.pio/build/BIGTREE_SKR_PRO/ and right clicking on the "firmware.bin" file name to bring up the content menu. Select "Reveal in File Explorer" to have your Window's machine open a file explorer window.



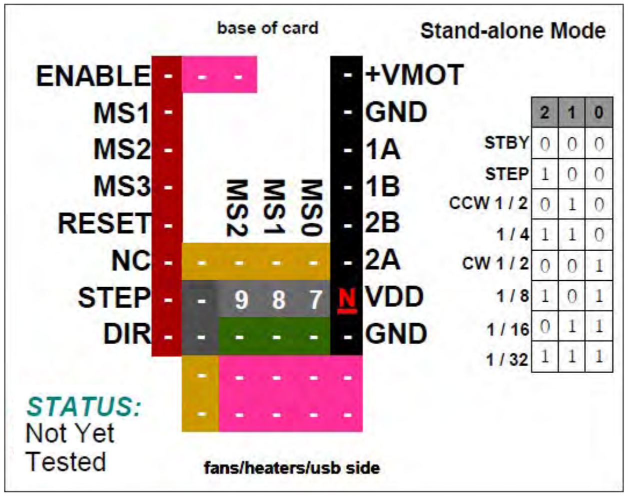
- Go to the next page.

The (latest release of) Marlin Setup for BIQU S109 Drivers

- Ensure your micro SD card, which came with your SKR PRO V1.1 board, with your SKR PRO V1.1 board is in a micro-to-SD-card adapter and plugged into your Window's SD card reader.
    - From the file explorer window right click on the "firmware.bin" file name and select "Send to" from the content menu. For my computer "D:" is my SD card reader. For your computer it may be a different drive letter. The "firmware.bin" file will be written to the SD card and renamed to "FIRMWARE.CUR".



- Right click on the SD card reader, as seen in the **GREEN** box above, and select "Eject" from the context menu. Take the micro SD card out of the adapter and place it into the micro SD card reader of the SKR PRO V1.1 board. Turn on the power to the printer (i.e. the SKR PRO V1.1 board) and the "FIRMWARE.CUR" file will be uploaded to the board. After the upload the firmware file will be renamed to "firmware.bin" and stored on the micro SD card.



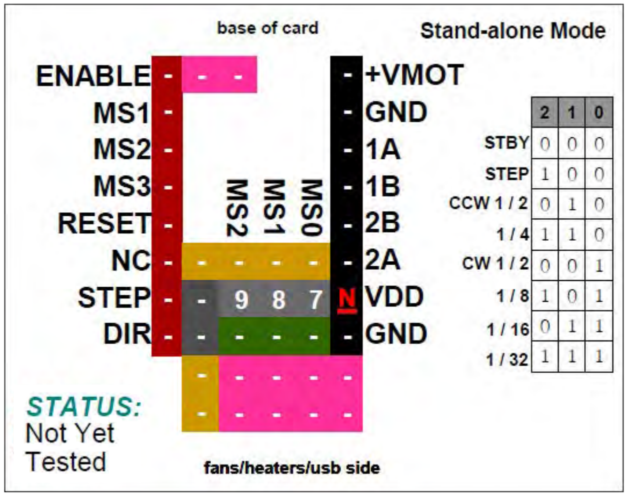
FYSETC S109

Note: V_{DD} is an Output (**N**), ***on a 3.3 V MCU like SKR PRO V1.1 the V_{DD} must be disconnected!!***

Note: See the next page for information about location of the current sense resistors and how to set V_{ref} on the stepper motor driver board.

Driver Chip	MS2	MS1	MS0	Microstep Resolution
FYSETC S109 Maximum 32 Subdivision 50V DC 4A (peak)	Low	Low	Low	Standby mode (outputs disabled)
	High	Low	Low	Full step
	Low	High	Low	Non-circular half step ("a")
	High	High	Low	1/4 step
	Low	Low	High	Circular half step ("b")
	High	Low	High	1/8 step
	Low	High	High	1/16 step
	High	High	High	1/32 step
Driving Current Calculation Formula V_{DD} = 5 V DC R_S(Typical Sense Resistor)=0.1Ω	$I_{MAX}=V_{ref}*(\frac{V_{DD}}{5})*\frac{1}{(5*R_S)}$			$V_{ref}=I_{MAX}*(\frac{5}{V_{DD}})*(5*R_S)$

NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.



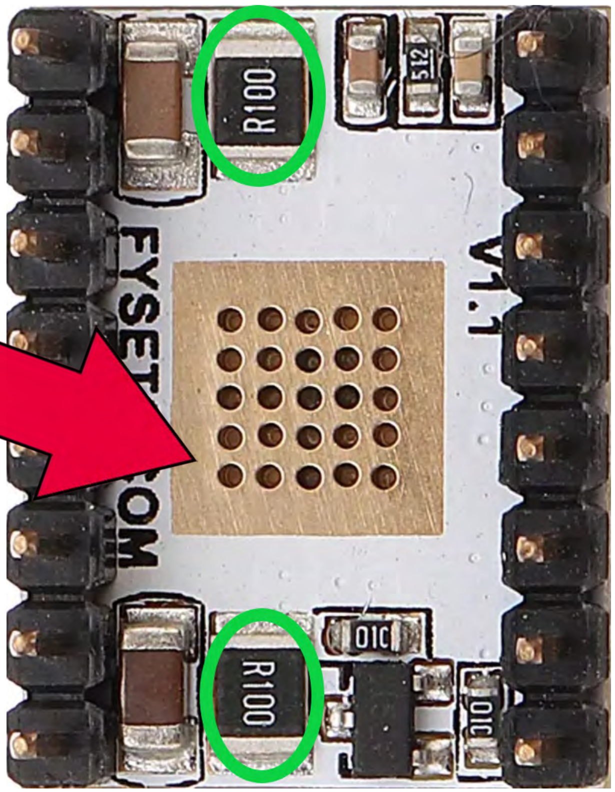
	2	1	0
STBY	0	0	0
STEP	1	0	0
CCW 1 / 2	0	1	0
1 / 4	1	1	0
CW 1 / 2	0	0	1
1 / 8	1	0	1
1 / 16	0	1	1
1 / 32	1	1	1

FYSETC S109

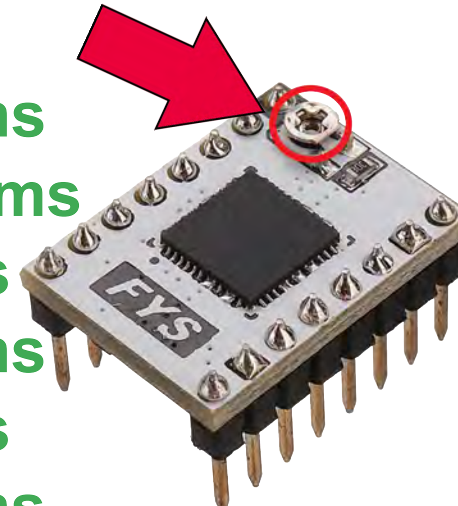
Note: Use 90% of the calculated V_{ref} when tuning the stepper driver board. V_{DD} is an Output (**N**), **on a 3.3 V MCU like SKR PRO V1.1 the V_{DD} must be disconnected!**

NOTE: Use the potentiometer (POT) on the top of the board to adjust your V_{ref}. See **Appendix A** for instructions on how to set the V_{ref} on a driver board.

Note: Check your current sense resistors (R_s) values on the driver board, as shown in GREEN



R_s = R050 is 0.05 Ohms
R_s = R068 is 0.068 Ohms
R_s = R100 is 0.1 Ohms
R_s = R150 is 0.15 Ohms
R_s = R200 is 0.2 Ohms
R_s = R220 is 0.22 Ohms



Note: See this video about current sense resistors (R_s) and their possible locations:
https://youtu.be/8wk1elugv5A

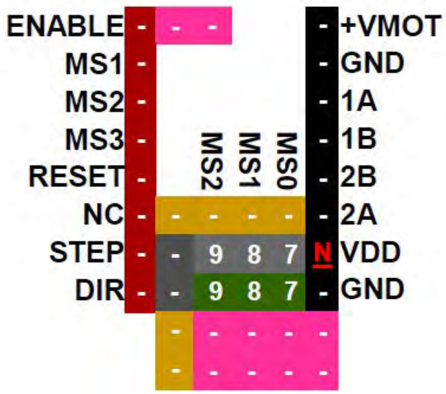
NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

FYSETC S109

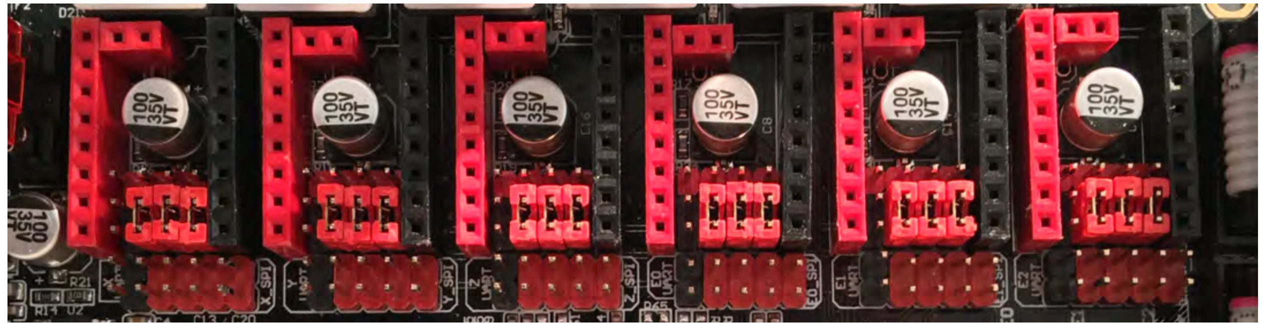
Stand-alone Mode

Note: 'Circular half step ("b")' means "clockwise (CW) motor direction with half step resolution". 'Non-circular half step ("a")' means "counterclockwise (CCW) motor direction with half step resolution".

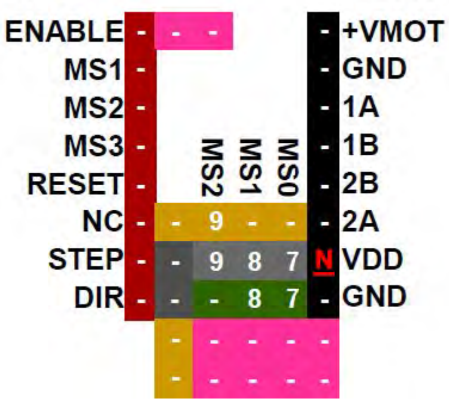
Stand By



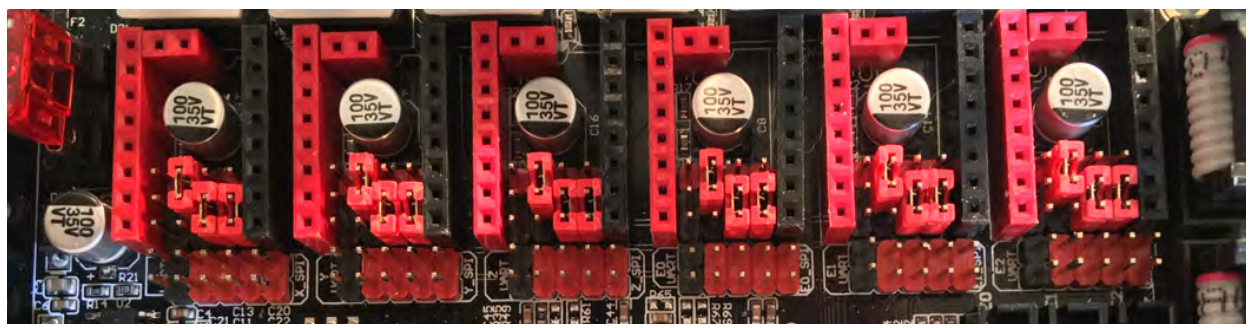
Note: **N**, on a 3.3 V MCU, like SKR PRO V1.1, **MUST** be disconnected!



STEP



Note: **N**, on a 3.3 V MCU, like SKR PRO V1.1, **MUST** be disconnected!



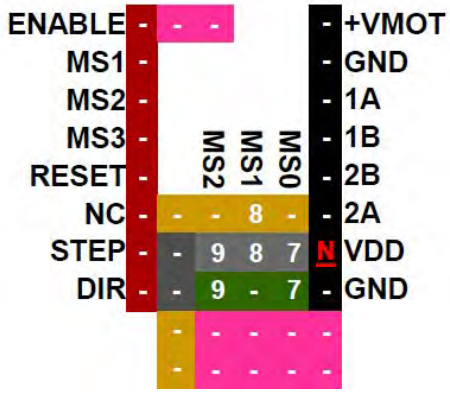
NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

FYSETC S109

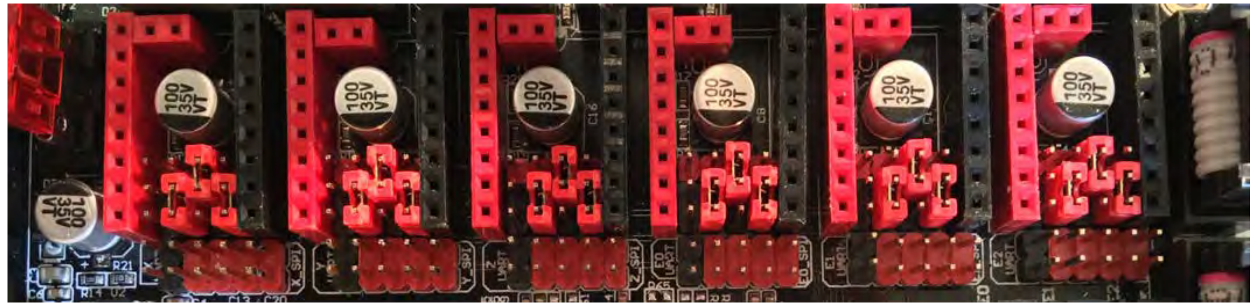
Stand-alone Mode

Note: 'Circular half step ("b")' means "clockwise (CW) motor direction with half step resolution". 'Non-circular half step ("a")' means "counterclockwise (CCW) motor direction with half step resolution".

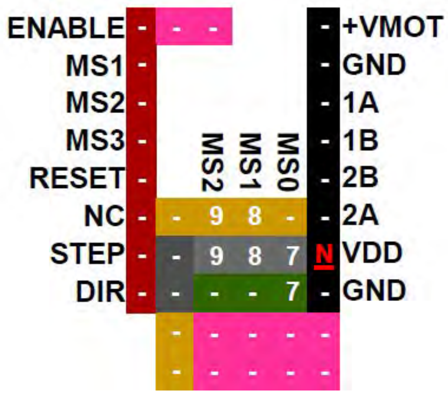
CCW 1 / 2



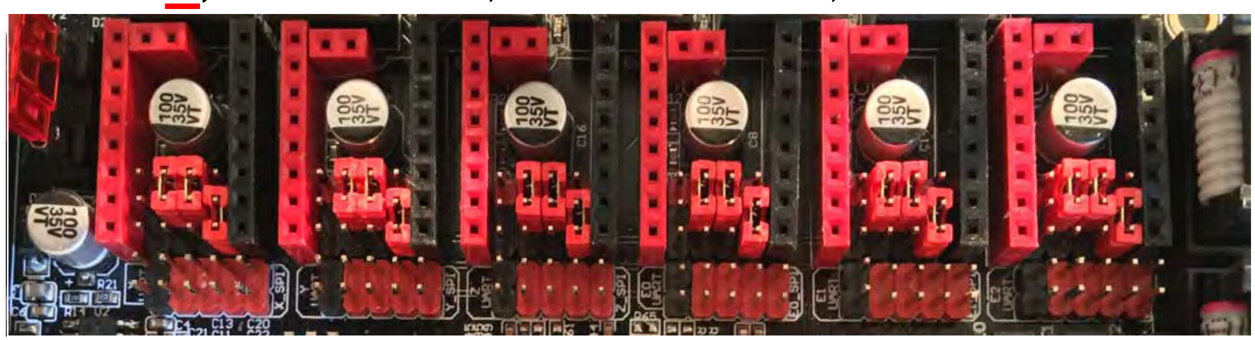
Note: **N**, on a 3.3 V MCU, like SKR PRO V1.1, **MUST** be disconnected!



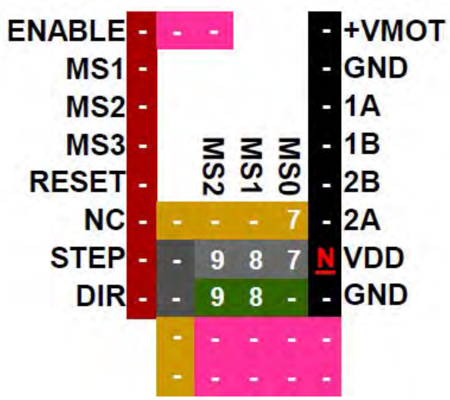
1 / 4



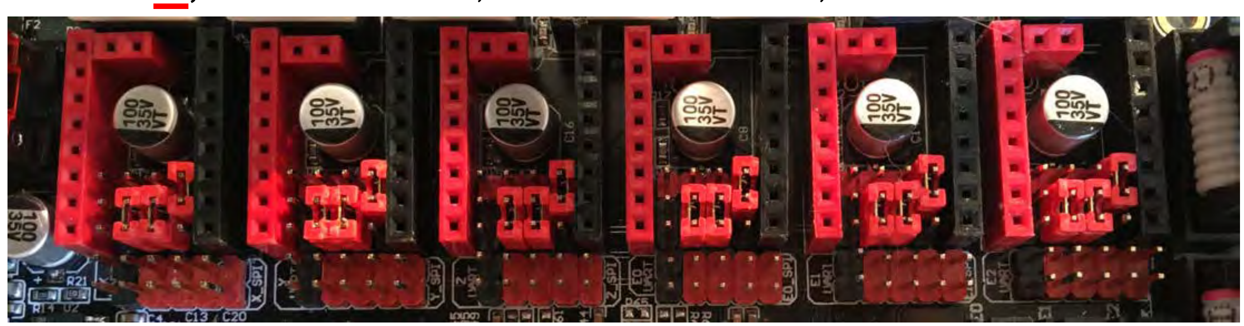
Note: **N**, on a 3.3 V MCU, like SKR PRO V1.1, **MUST** be disconnected!



CW 1 / 2



Note: **N**, on a 3.3 V MCU, like SKR PRO V1.1, **MUST** be disconnected!



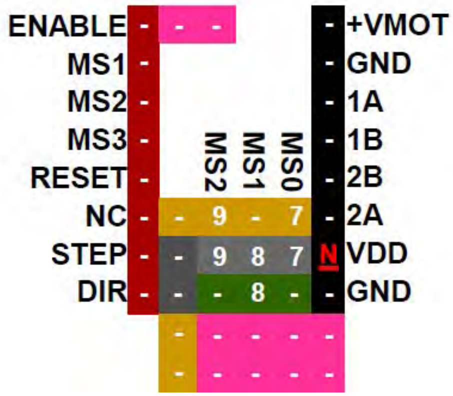
NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

FYSETC S109

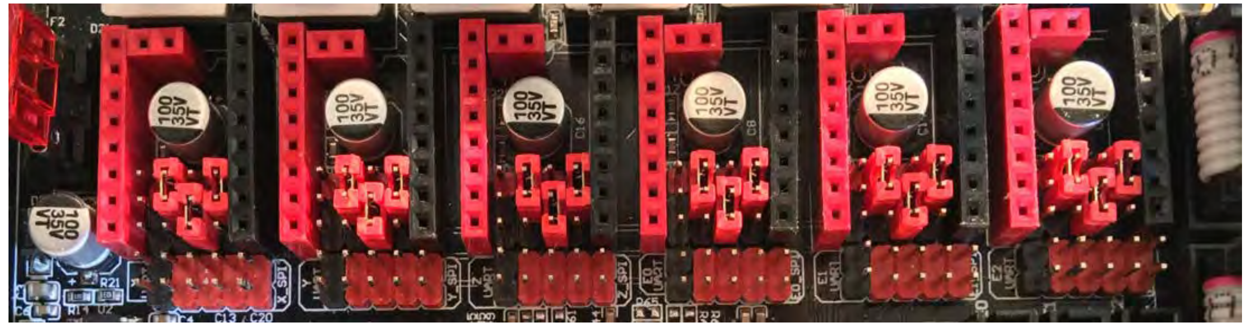
Stand-alone Mode

Note: 'Circular half step ("b")' means "clockwise (CW) motor direction with half step resolution". 'Non-circular half step ("a")' means "counterclockwise (CCW) motor direction with half step resolution".

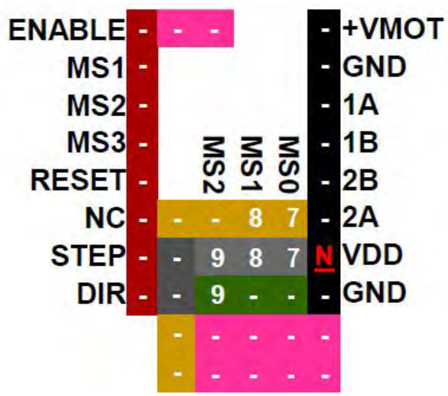
1 / 8



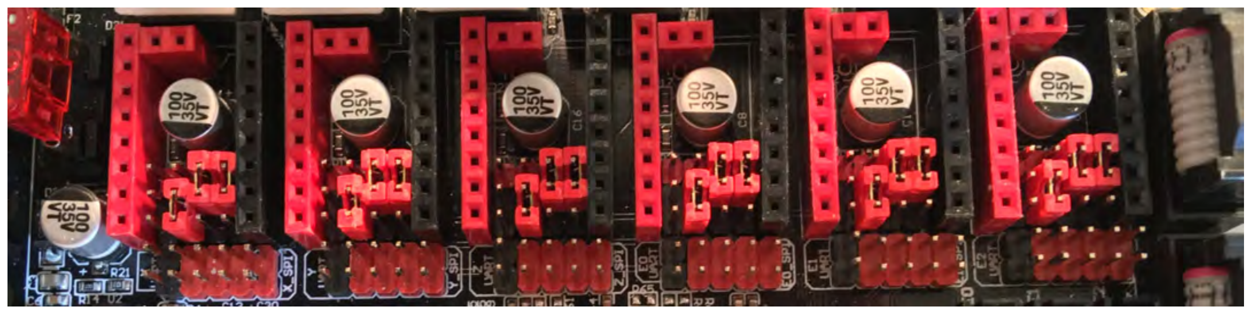
Note: **N**, on a 3.3 V MCU, like SKR PRO V1.1, **MUST** be disconnected!



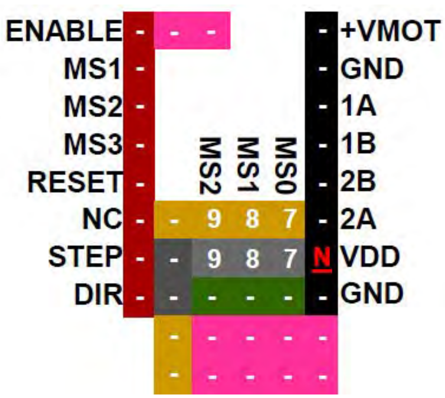
1 / 16



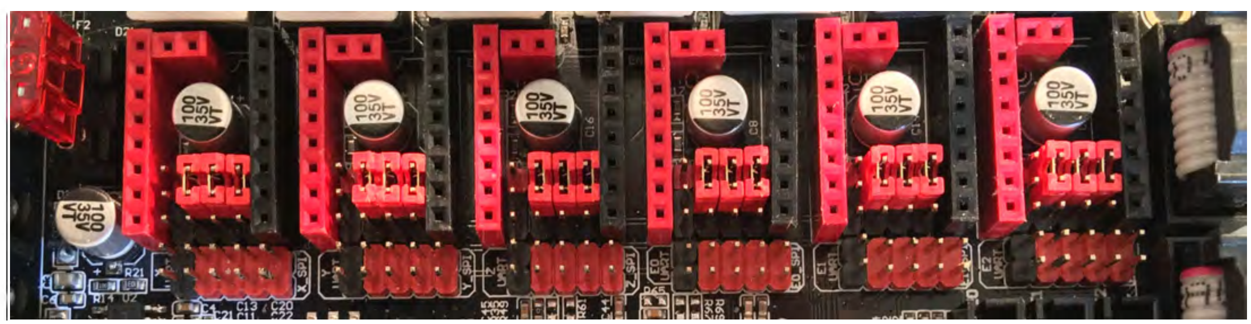
Note: **N**, on a 3.3 V MCU, like SKR PRO V1.1, **MUST** be disconnected!



1 / 32



Note: **N**, on a 3.3 V MCU, like SKR PRO V1.1, **MUST** be disconnected!

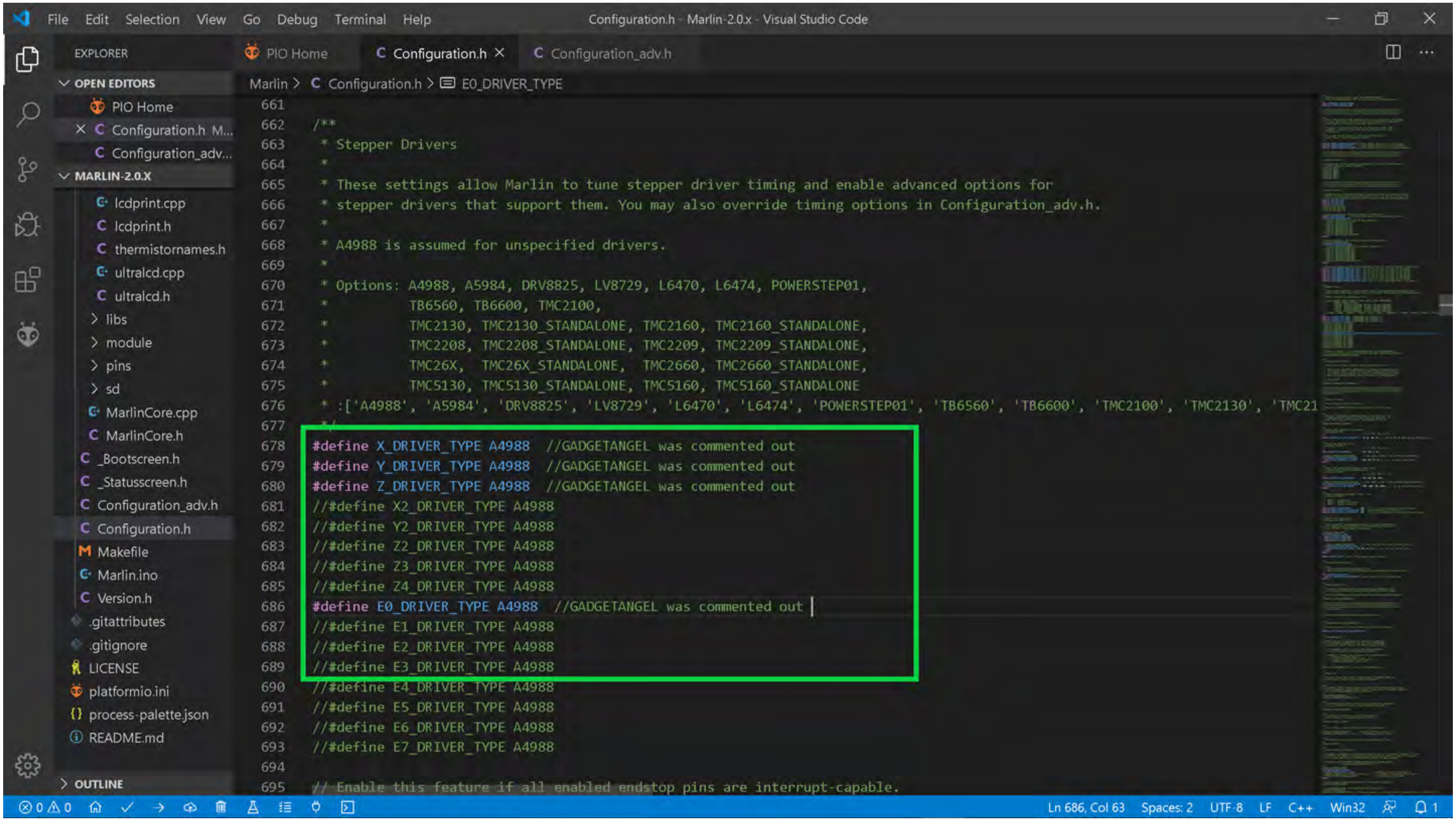


NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

The (latest release of) Marlin Setup for FYSETC S109 Drivers

NOTE: **Go to Appendix C,** and then come back here for the changes to Marlin for FYSETC S109 stepper motor drivers.

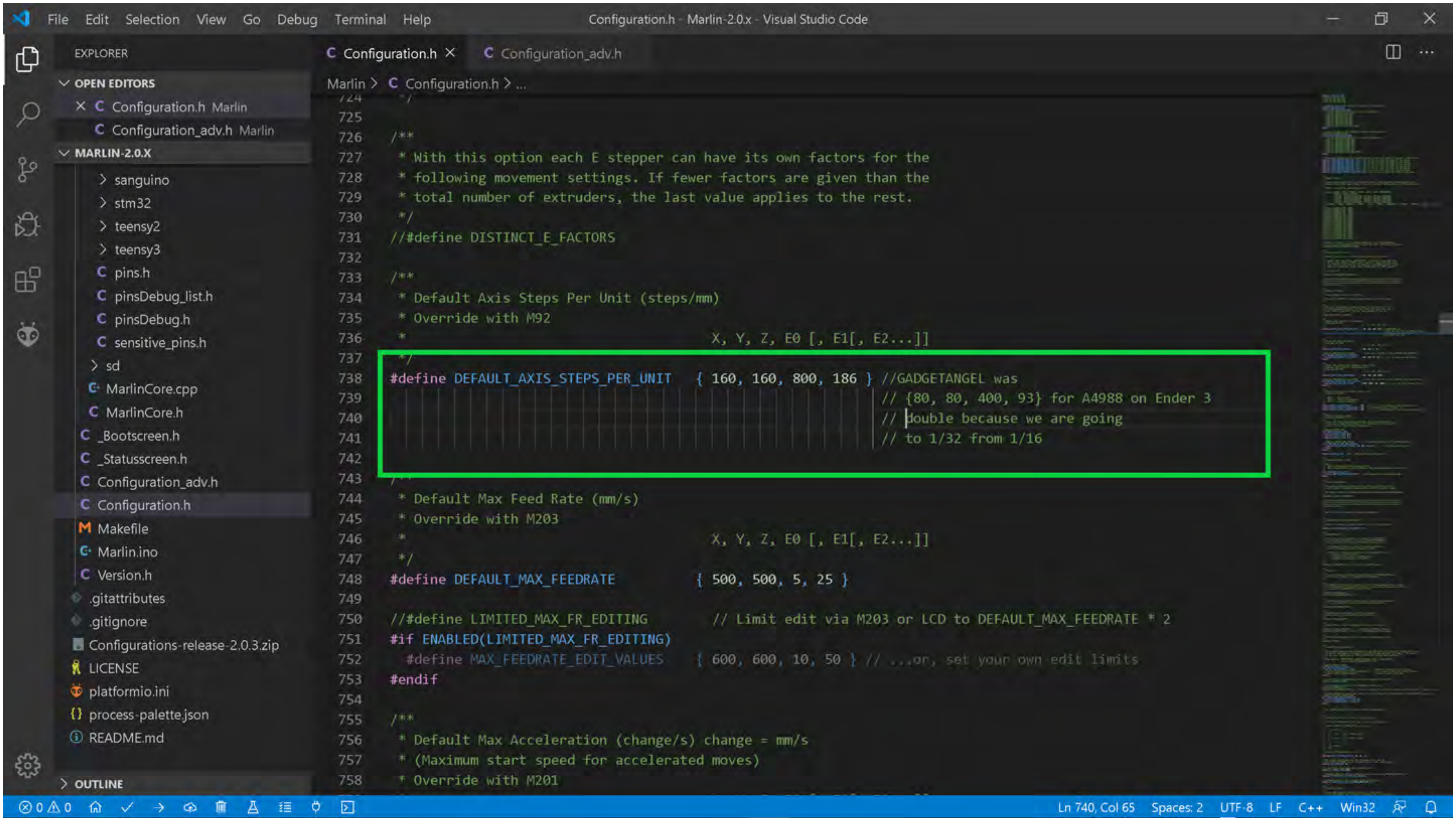
- Change the stepper motor drivers so that Marlin knows you are using FYSETC S109 drivers. Change one line for each axis and one for each extruder you will be using. See the picture below for an example of how to use FYSETC S109 drivers. When two "//" appear at the beginning of a line that means that line is commented out. To un-comment a line just remove the leading two forward slashes ("//").
- The FYSETC S109 is a drop in replacement for the A4988. Since Marlin does not have an option for FYSETC S109 we will use the A4988 as the driver type.



- Go to the next page.

The (latest release of) Marlin Setup for FYSETC S109 Drivers

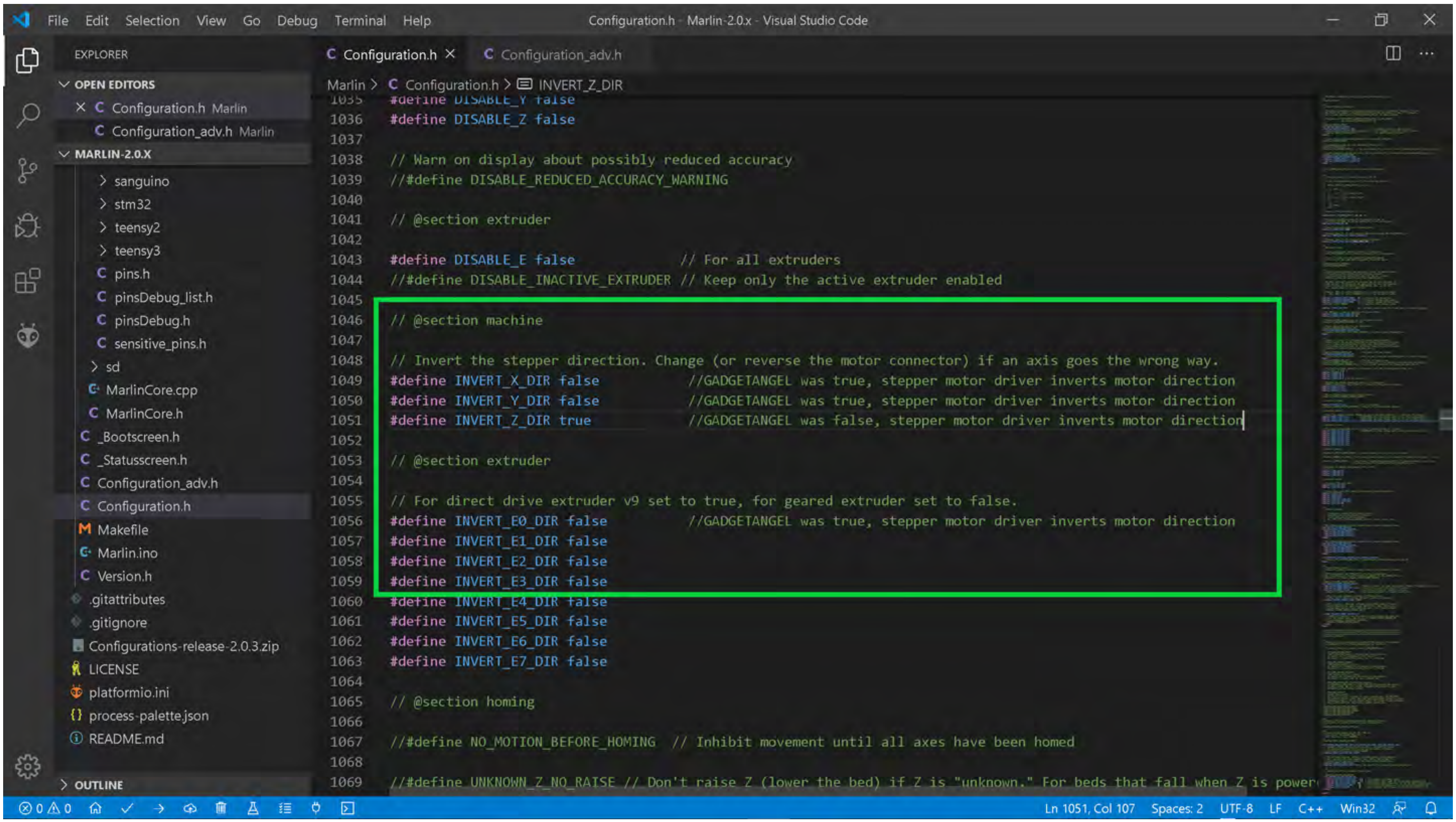
- Since we are changing from A4988 stepper motor drivers on the Ender 3 to for FYSETC S109 stepper motor drivers for each axis and the extruder stepper motor driver, we will be going from 1/16 stepping to 1/32 stepping. So we are doubling our STEPS. Therefore, we must adjust our "DEFAULT_AXIS_STEPS_PER_UNIT" anytime our STEPS are NOT 1/16. So change "DEFAULT_AXIS_STEPS_PER_UNIT" to {160, 160, 800, 186}, as seen in the **GREEN** box below.



- Go to the next page.

The (latest release of) Marlin Setup for FYSETC S109 Drivers

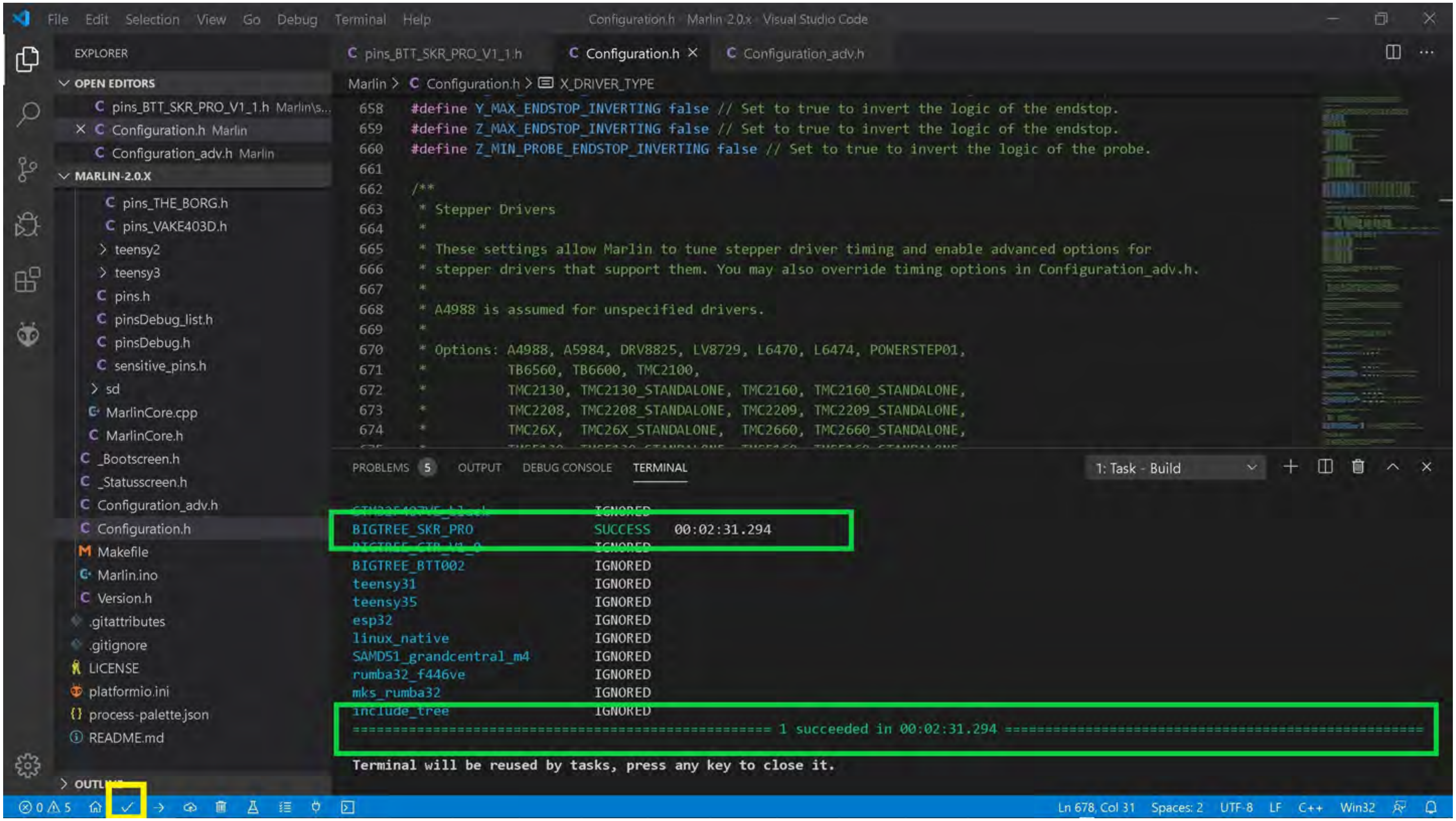
- Optional Step: I cannot find information on the FYSETC S109 driver's impact on motor direction. So I provide the below information in case you do need to change the stepper motor direction. If you prefer to change the motor direction with wiring instead of the Marlin firmware, here is a link on how to change the motor direction via the wiring (look for section labeled "Motor moving the wrong direction") **https://reprap.org/wiki/Stepper_wiring**. Other people prefer to change the motor direction in the Marlin firmware. So if you want or need to change the motor direction in Marlin, then if the axis' setting you will be using the S109 driver on was "true" change it to "false", as shown in the **GREEN** box below. If the setting was "false", now set it to "true", as shown in the **GREEN** box below



- Go to the next page.

The (latest release of) Marlin Setup for FYSETC S109 Drivers

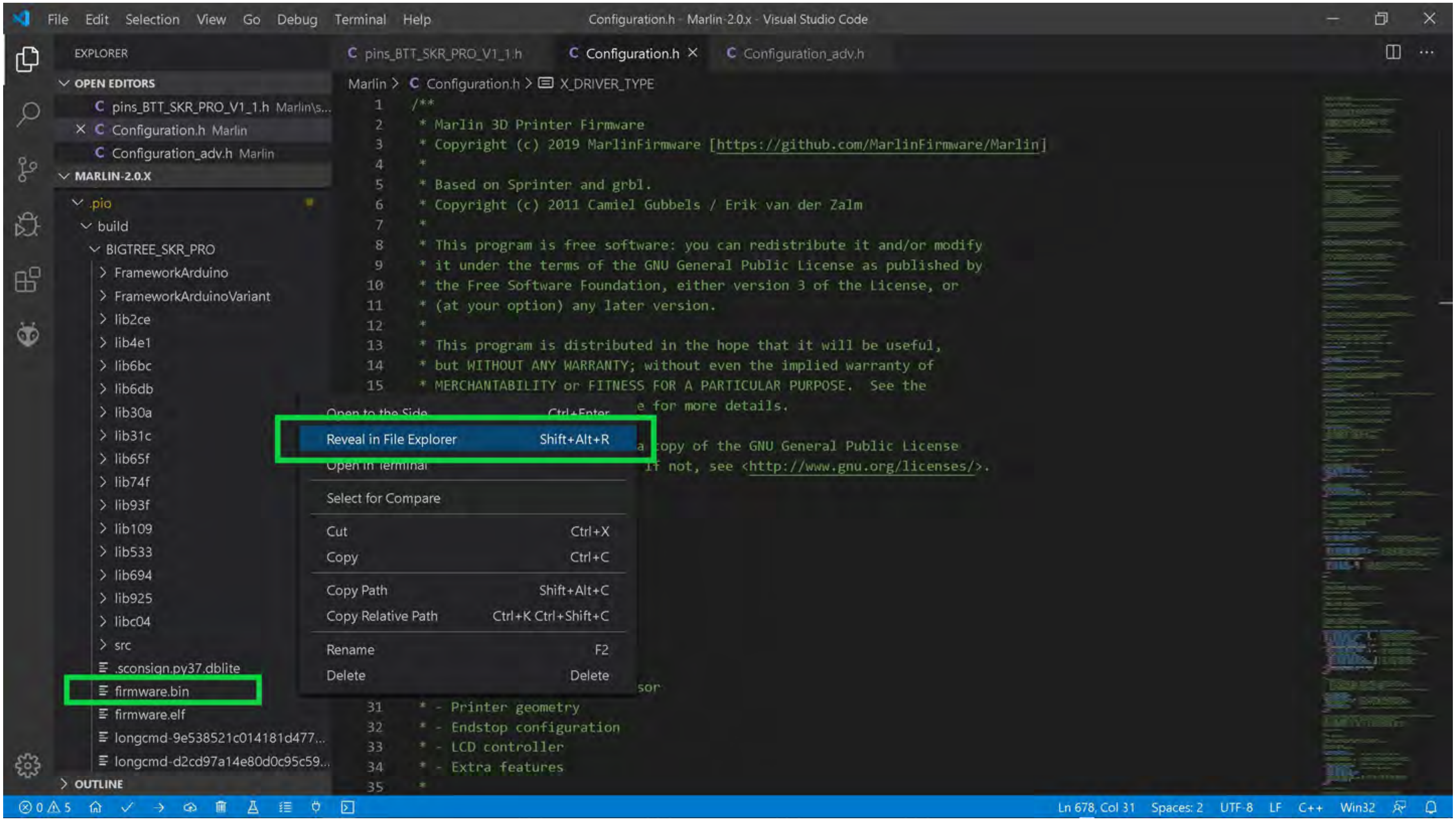
- The end of Marlin setup for FYSETC S109 drivers. Now, compile your code by clicking on the check mark, as seen in the **YELLOW** box below. Once the firmware has compiled successfully you will see the following messages, as shown in the **GREEN** boxes below.



- For the SKR PRO V1.1 board you must save the "firmware.bin" file to the micro SD card then place the micro SD card into the micro SD card reader of the SKR PRO V1.1 board and turn on the power to your printer for the NEW firmware to be automatically applied to the SKR PRO MCU. To locate the "firmware.bin" file, and learn how to transfer it to your micro SD card go to the next page.

The (latest release of) Marlin Setup for FYSETC S109 Drivers

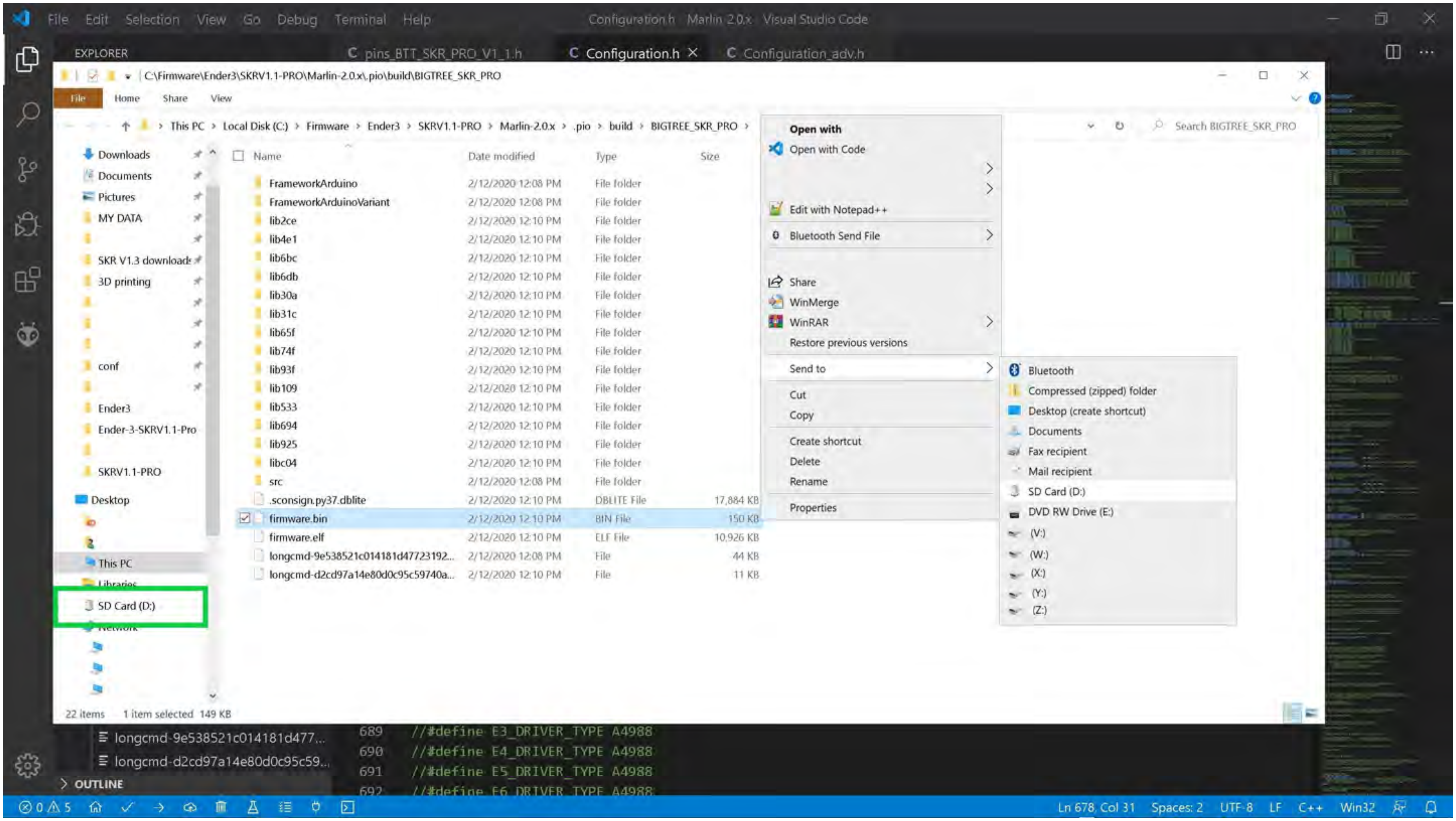
- The "firmware.bin" file can be found by using the left side of the VS code window and going to folder Marlin-2.0.X/.pio/build/BIGTREE_SKR_PRO/ and right clicking on the "firmware.bin" file name to bring up the content menu. Select "Reveal in File Explorer" to have your Window's machine open a file explorer window.



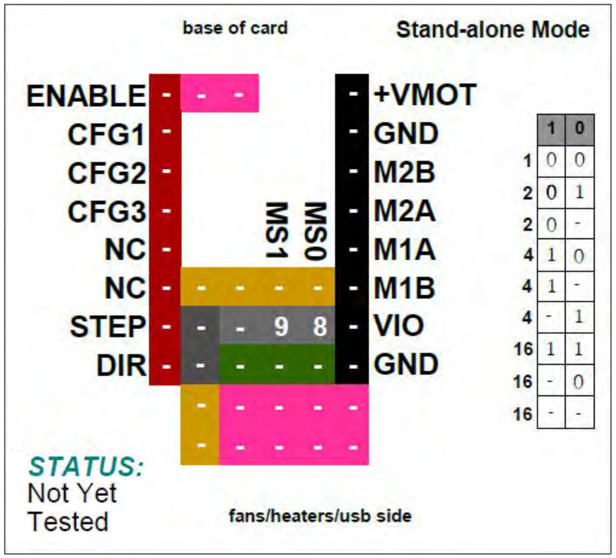
- Go to the next page.

The (latest release of) Marlin Setup for FYSETC S109 Drivers

- Ensure your micro SD card, which came with your SKR PRO V1.1 board, with your SKR PRO V1.1 board is in a micro-to-SD-card adapter and plugged into your Window's SD card reader.
    - From the file explorer window right click on the "firmware.bin" file name and select "Send to" from the content menu. For my computer "D:" is my SD card reader. For your computer it may be a different drive letter. The "firmware.bin" file will be written to the SD card and renamed to "FIRMWARE.CUR".



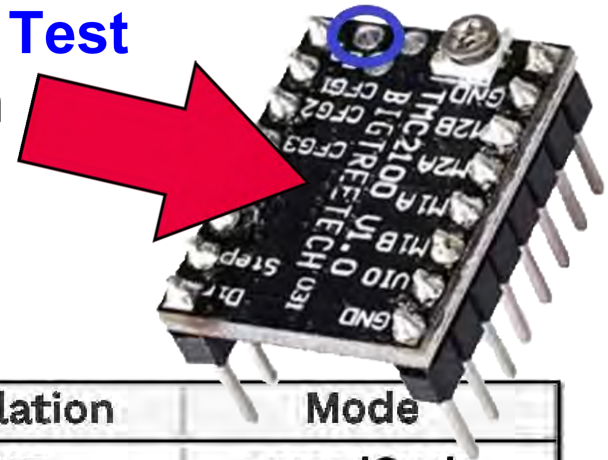
- Right click on the SD card reader, as seen in the **GREEN** box above, and select "Eject" from the context menu. Take the micro SD card out of the adapter and place it into the micro SD card reader of the SKR PRO V1.1 board. Turn on the power to the printer (i.e. the SKR PRO V1.1 board) and the "FIRMWARE.CUR" file will be uploaded to the board. After the upload the firmware file will be renamed to "firmware.bin" and stored on the micro SD card.



BIQU TMC2100

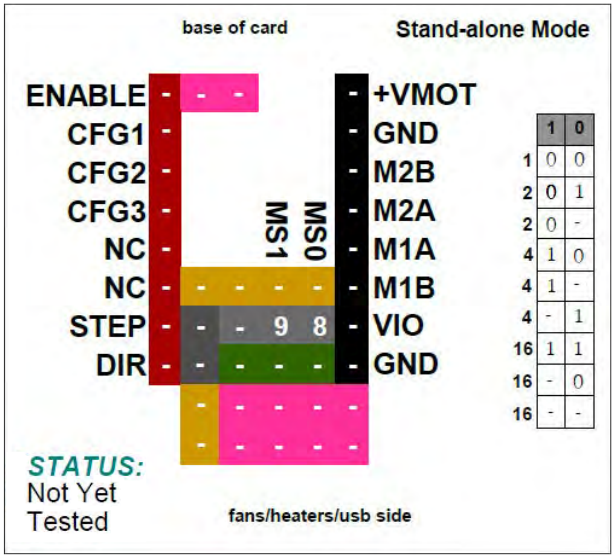
Stand-alone Mode

NOTE: **Use the potentiometer (POT) on the top of the board or use the "V_{ref} Test point" location, as shown in BLUE, to adjust your driver board's V_{ref}**



Driver Chip	MS1	MS0	Steps	Interpolation	Mode
BIQU® TMC2100 Stand Alone Mode Maximum 16 Subdivision 46V DC 2.5A (peak)	GND	GND	1	NONE	spreadCycle
	GND	VIO	1 / 2	NONE	spreadCycle
	GND	OPEN	1 / 2	1 / 256	spreadCycle
	VIO	GND	1 / 4	NONE	spreadCycle
	VIO	OPEN	1 / 4	1 / 256	spreadCycle
	OPEN	VIO	1 / 4	1 / 256	stealthChop
	VIO	VIO	1 / 16	NONE	spreadCycle
	OPEN	GND	1 / 16	1 / 256	spreadCycle
	OPEN	OPEN	1 / 16	1 / 256	stealthChop

Driving Current Calculation Formula		
R_S(Typical Sense Resistor)=0.11Ω	$I_{MAX}=V_{ref}$ See Appendix B #1. Use 50% to 90% as shown below: $I_{MAX}=I_{MAX} * 0.90$	$V_{ref}=I_{MAX}$ See Appendix B #1. Use 50% to 90% as shown below: $V_{ref}=V_{ref} * 0.90$



	1	0
1	0	0
2	0	1
2	0	-
4	1	0
4	1	-
4	-	1
16	1	1
16	-	0
16	-	-

BIQU TMC2100

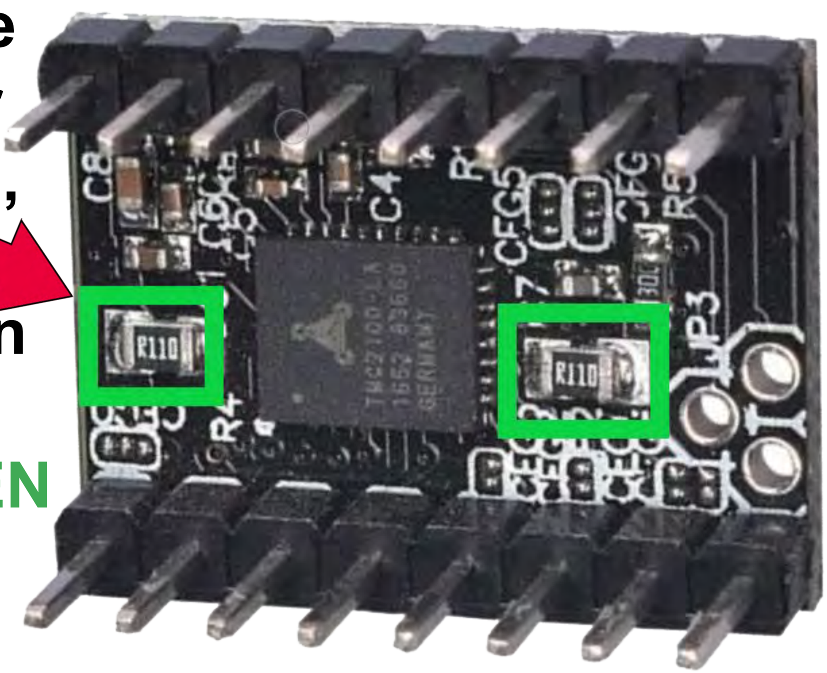
Stand-alone Mode

Note: Use 90% of the calculated V_{ref} when tuning the stepper driver board.

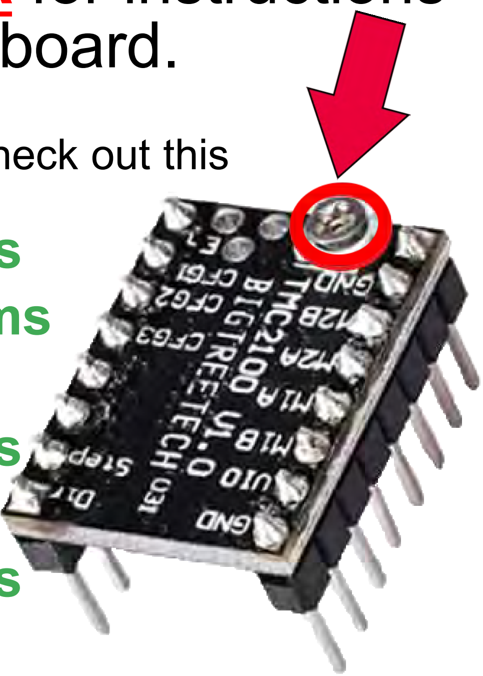
NOTE: Use the potentiometer (POT) on the top of the board (or use the board's "V_{ref} Test point") to set your V_{ref}. See **Appendix A** for instructions on how to set the V_{ref} on a driver board.

To learn more, please watch this **YouTube video done by Teaching Tech** and check out this link on the **TMC2100 Driver**

Note: Check your current sense resistors (R_s) values on the driver board, as shown in GREEN



R_s = R050 is 0.05 Ohms
R_s = R068 is 0.068 Ohms
R_s = R100 is 0.1 Ohms
R_s = R150 is 0.15 Ohms
R_s = R200 is 0.2 Ohms
R_s = R220 is 0.22 Ohms



Note: See this video about current sense resistors (R_s) and their possible locations: **https://youtu.be/8wk1elugv5A**

BIQU TMC2100

Stand-alone Mode

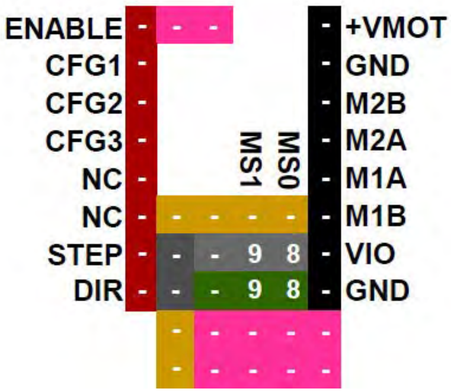
Stand-alone Mode

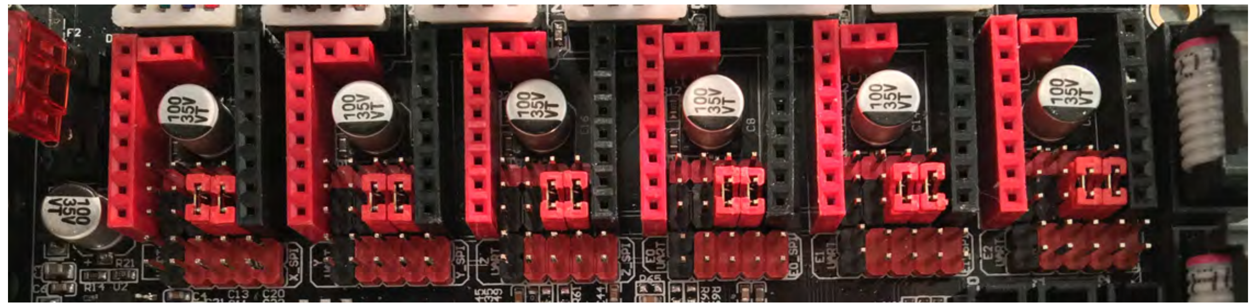
Stand-alone Mode

STEP

Interpolation: **none**

SpreadCycle



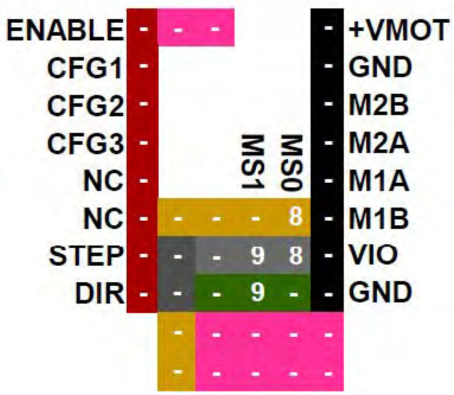


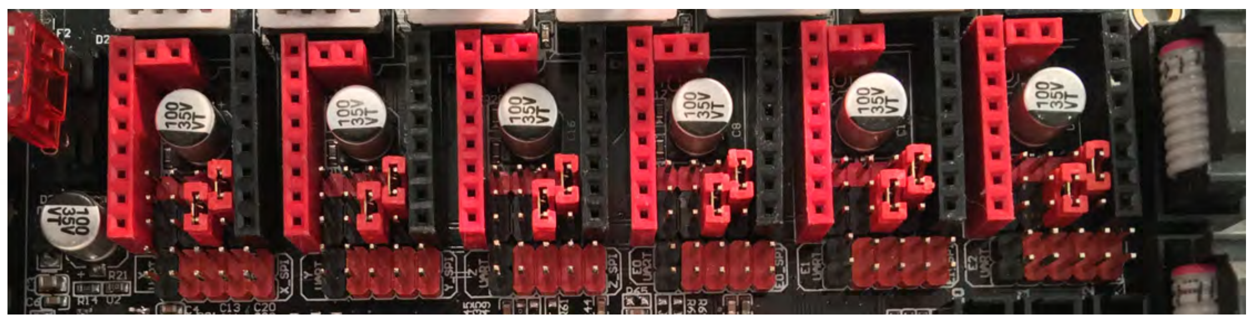
Stand-alone Mode

1 / 2

Interpolation: **none**

SpreadCycle



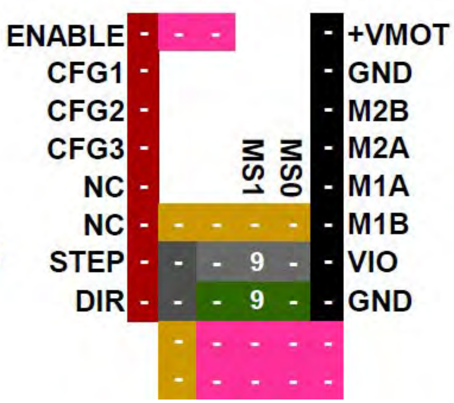


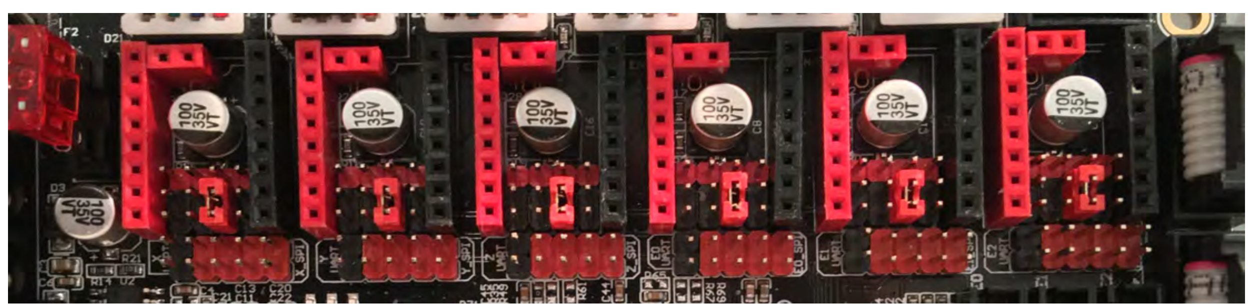
Stand-alone Mode

1 / 2

Interpolation: **1/256**

SpreadCycle





NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

BIQU TMC2100

Stand-alone Mode

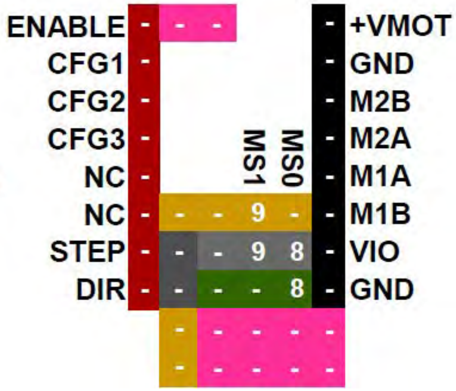
Stand-alone Mode

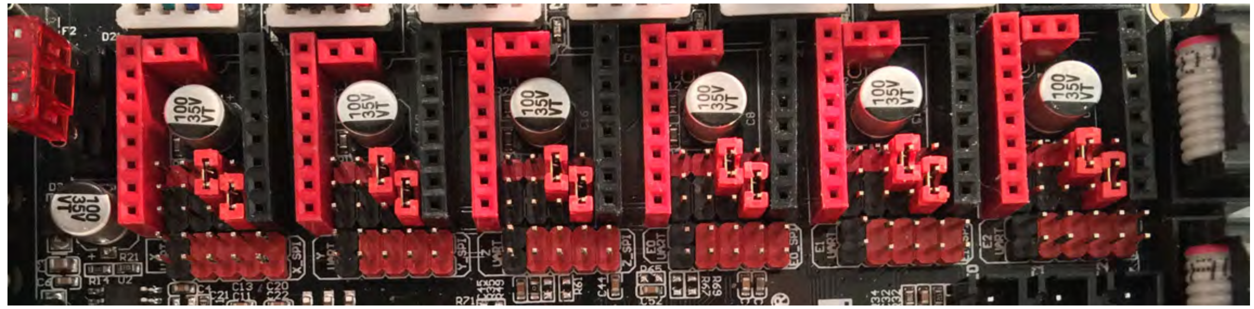
Stand-alone Mode

1 / 4

Interpolation: **none**

SpreadCycle



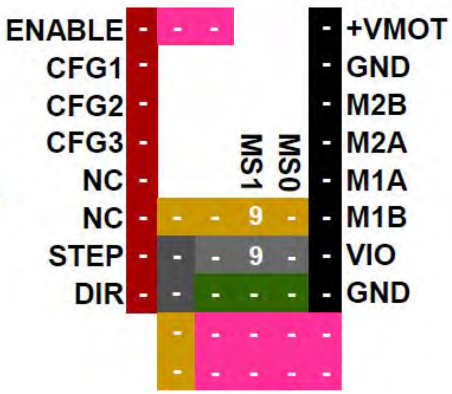


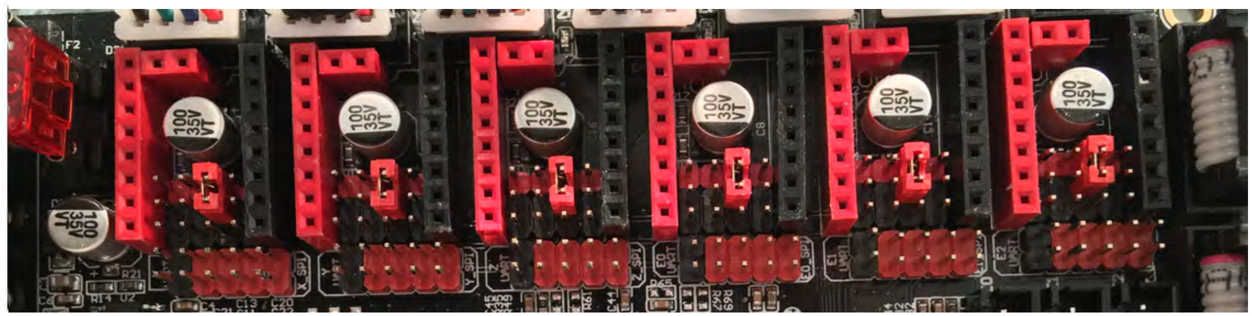
Stand-alone Mode

1 / 4

Interpolation: **1/256**

SpreadCycle



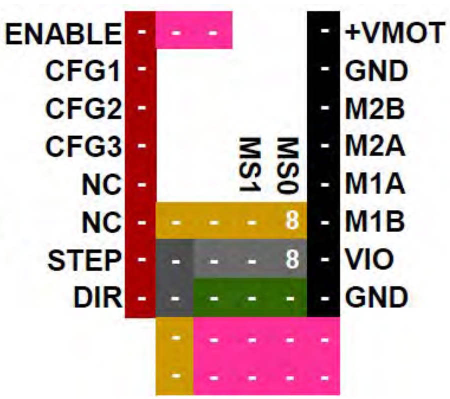


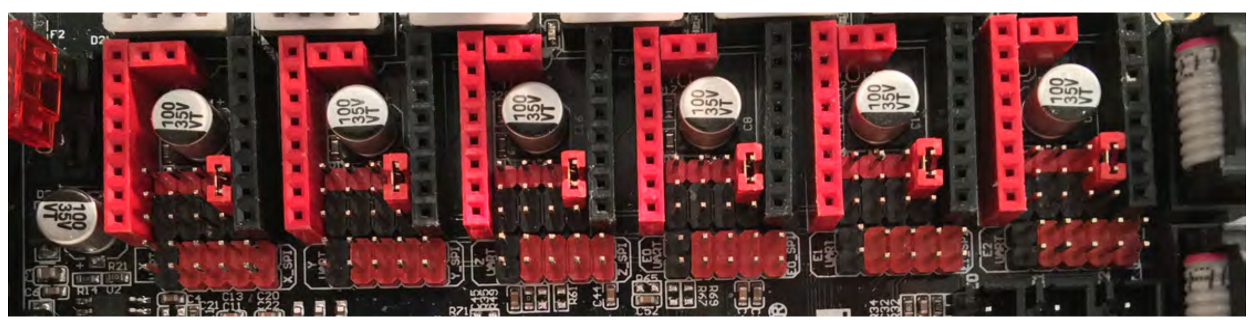
Stand-alone Mode

1 / 4

Interpolation: **1/256**

StealthChop





NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

BIQU TMC2100

Stand-alone Mode

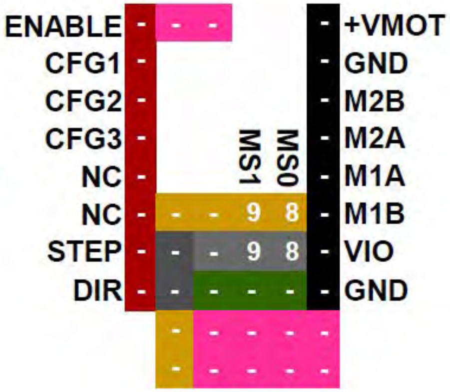
Stand-alone Mode

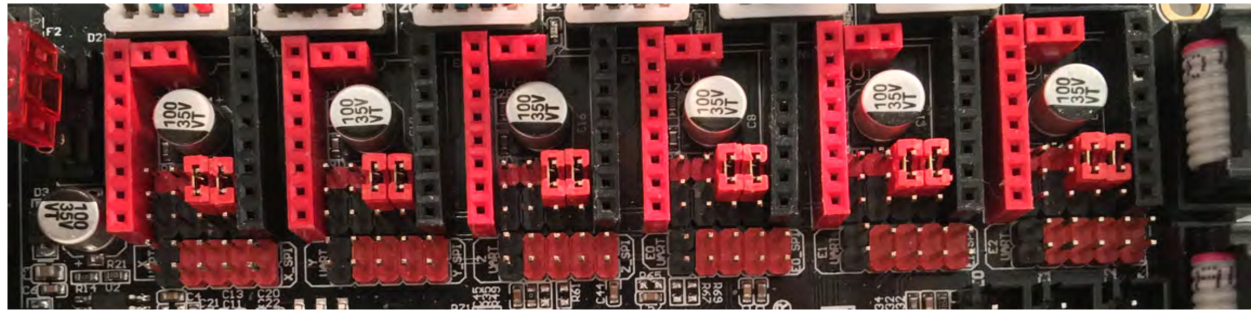
Stand-alone Mode

1 / 16

Interpolation: **none**

SpreadCycle



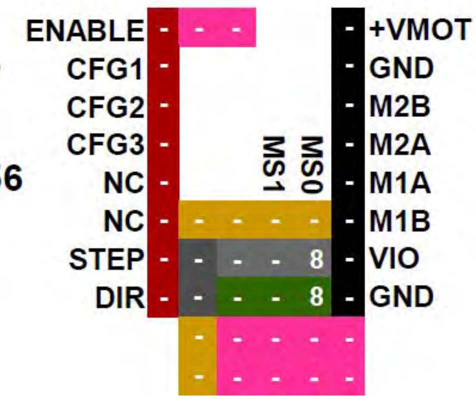


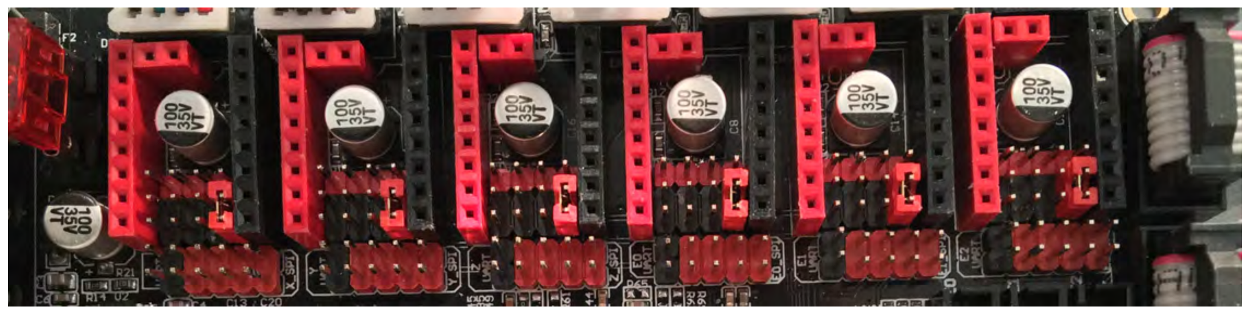
Stand-alone Mode

1 / 16

Interpolation: **1/256**

SpreadCycle



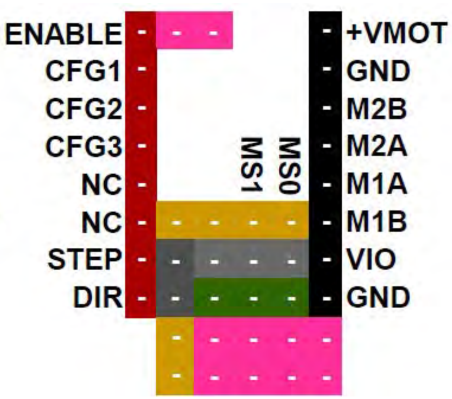


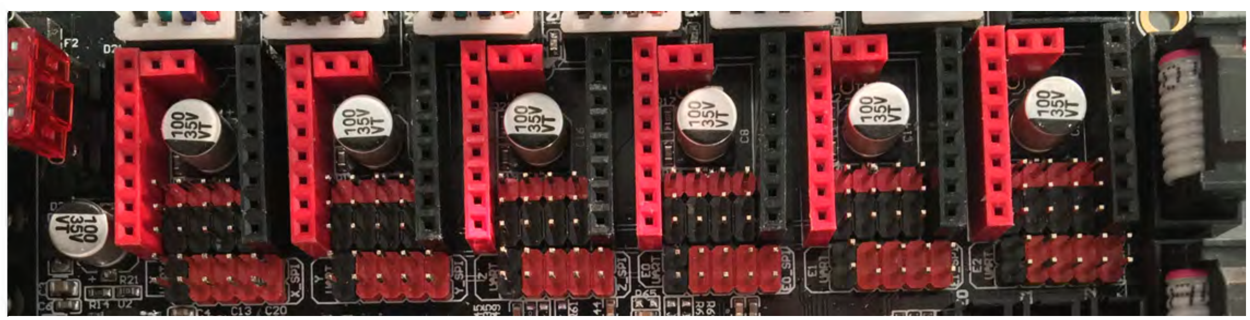
Stand-alone Mode

1 / 16

Interpolation: **1/256**

StealthChop



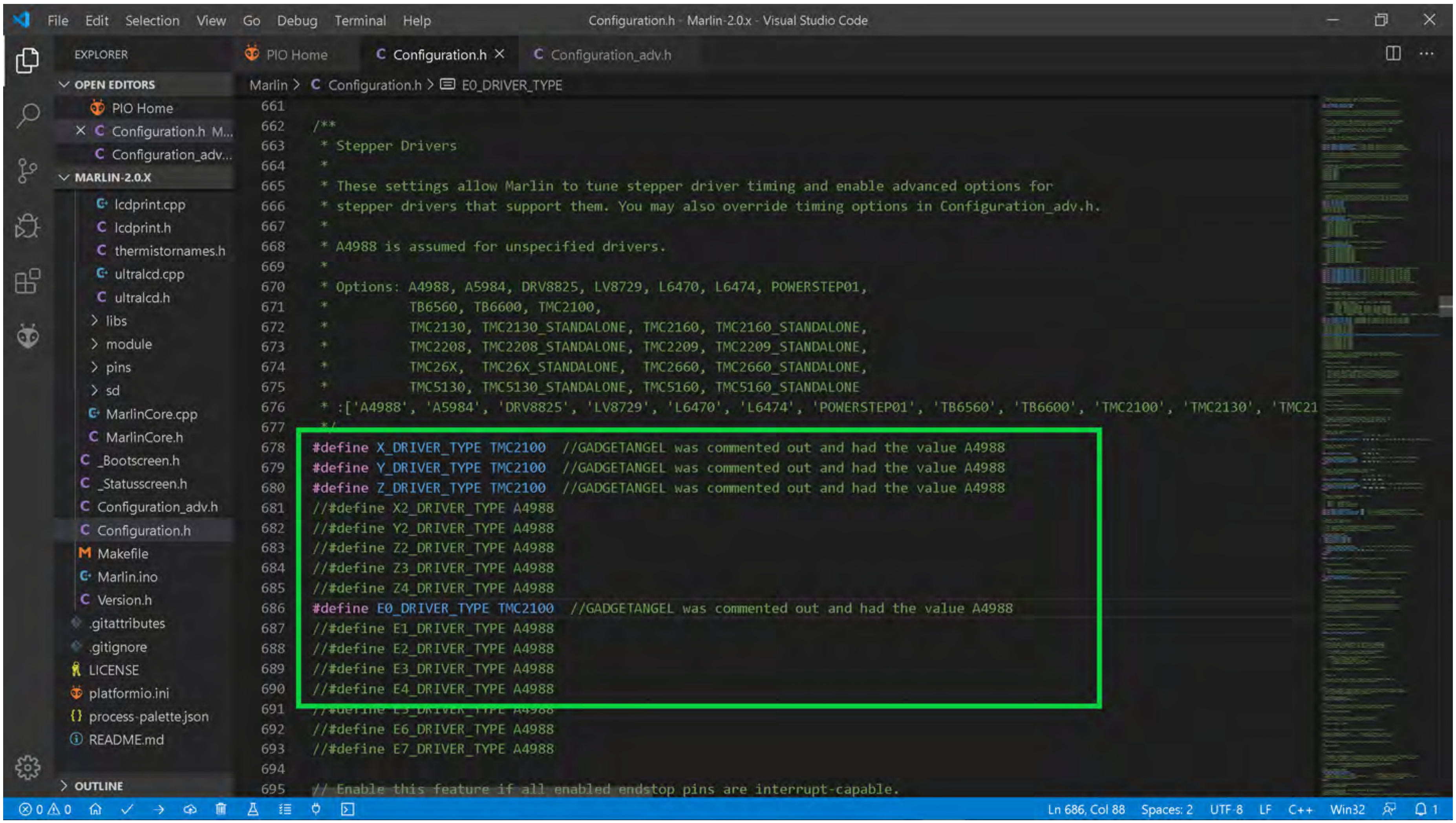


NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

The (latest release of) Marlin Setup for BIQU TMC2100 Drivers in Stand-alone Mode

NOTE: **Go to Appendix C,** and then come back here for the changes to Marlin for BIQU TMC2100 stepper motor drivers in stand-alone mode.

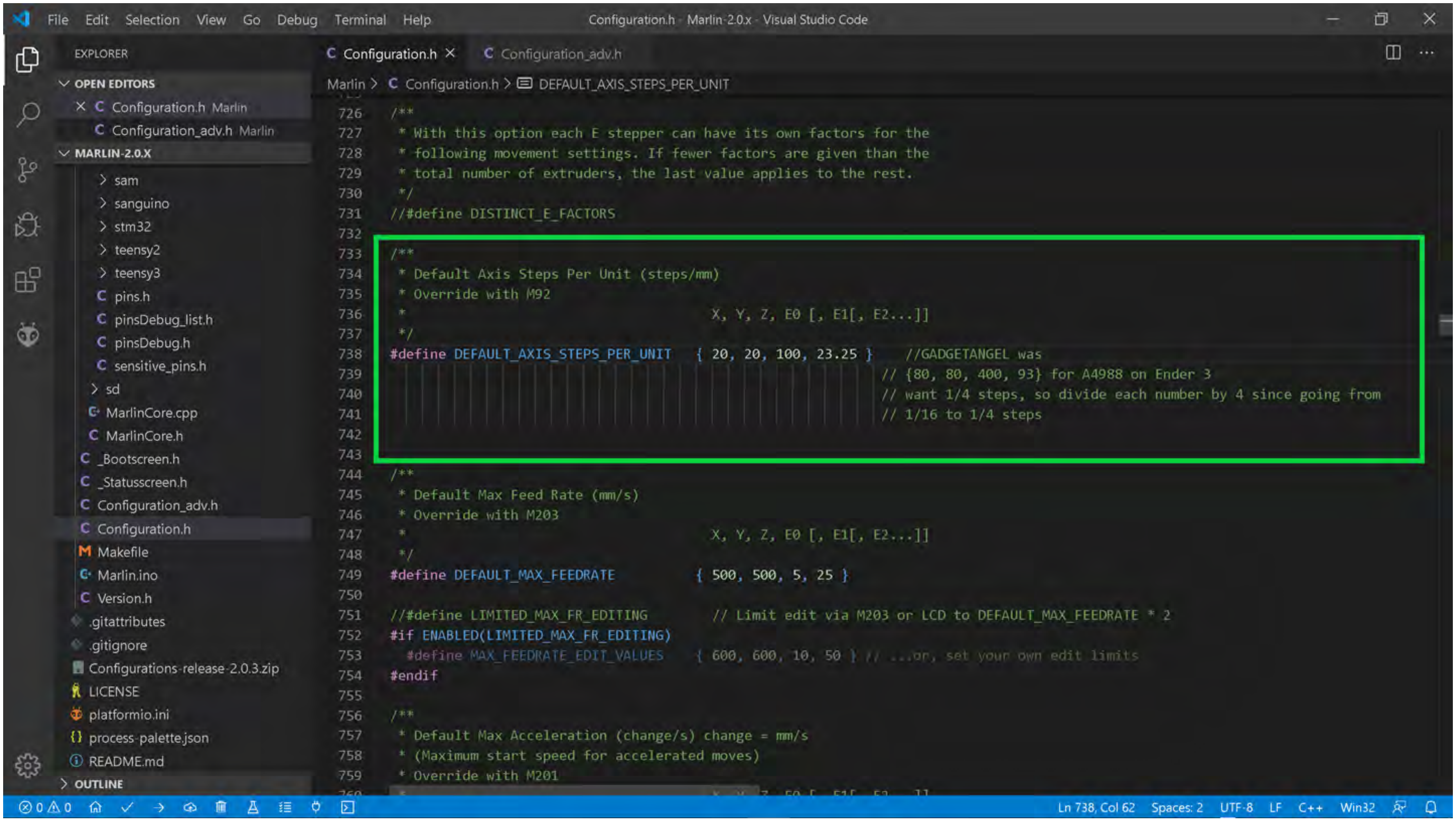
- Change the stepper motor drivers so that Marlin knows you are using BIQU TMC2100 drivers in stand-alone mode. Change one line for each axis and one for each extruder you will be using. See the picture below for an example of how to use BIQU TMC2100 drivers in stand-alone mode. When two "//" appear at the beginning of a line that means that line is commented out. To un-comment a line just remove the leading two forward slashes ("//").



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2100 Drivers in Stand-alone Mode

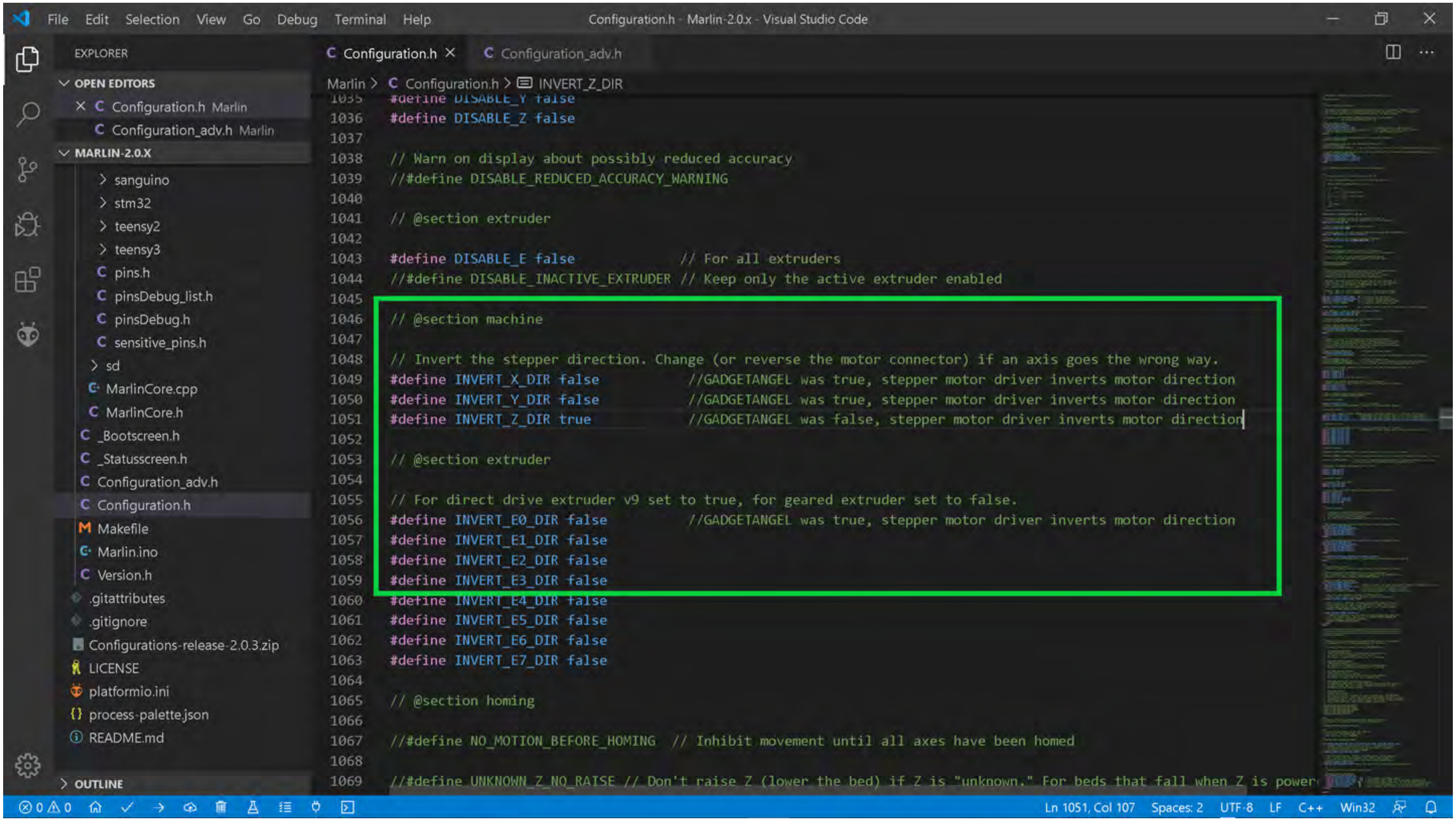
- Since I desire to use 1/4 stepping, and we are changing from A4988 stepper motor drivers on the Ender 3 to for BIQU TMC2100 stepper motor drivers for each axis and the extruder stepper motor driver, we will be going from 1/16 stepping to 1/4 stepping. So we are cutting our STEPS by one quarter. Therefore, we must adjust our "DEFAULT_AXIS_STEPS_PER_UNIT" anytime our STEPS are NOT 1/16. So change "DEFAULT_AXIS_STEPS_PER_UNIT" to {20, 20, 100, 23.25}, as shown in the **GREEN** box below.



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2100 Drivers in Stand-alone Mode

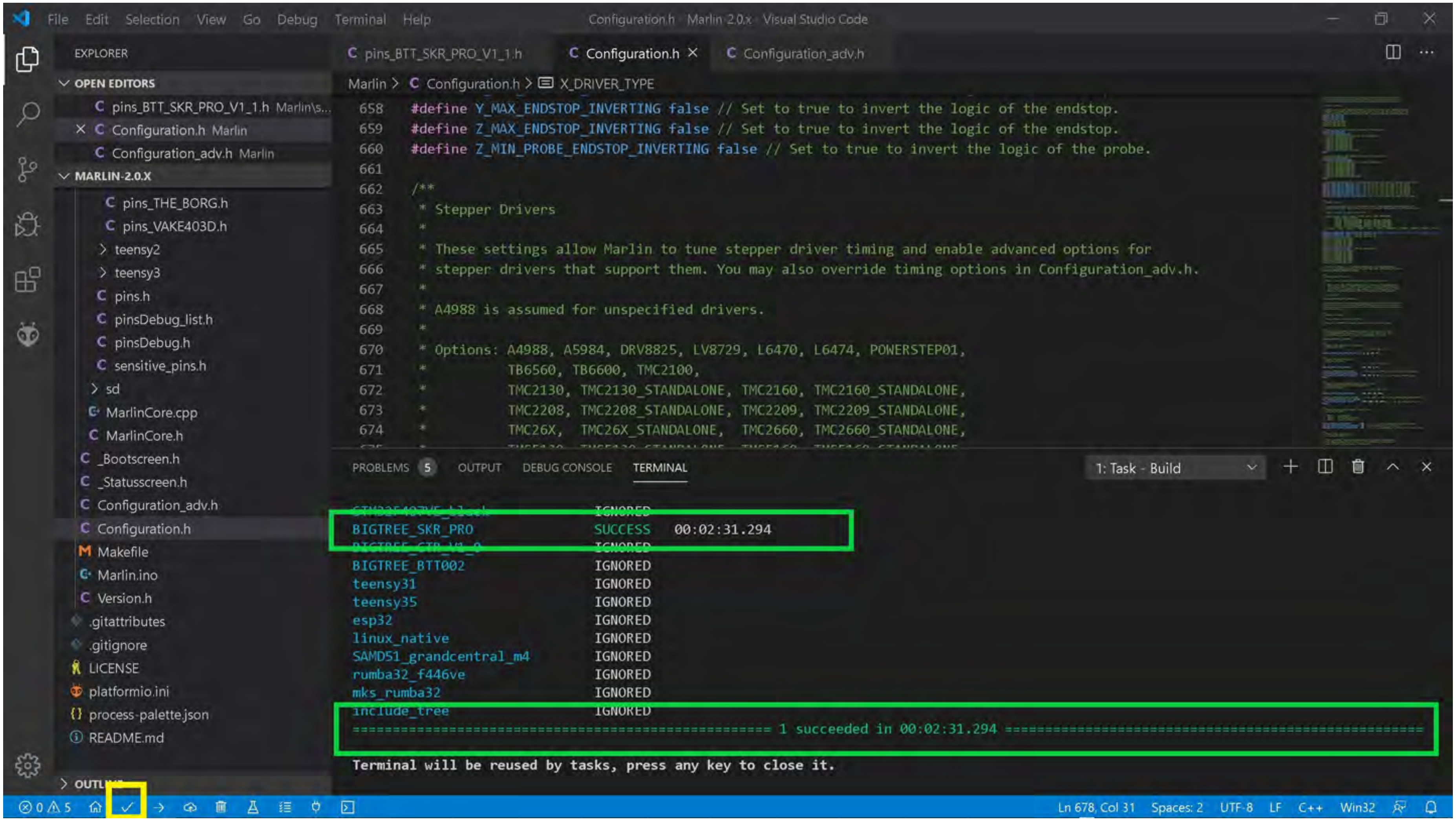
- Since the A4988 driver is what my Ender 3 used, but, now I want to use TMC2100 drivers, I must invert the stepper motor direction because the TMC2100 driver will turn the motors in the opposite direction than the A4988 driver's motor direction. So if the axis' setting you will be using the TMC2100 driver on was "true" change it to "false", as shown in the **GREEN** box below. If the setting was "false", now set it to "true", as show in the **GREEN** box below



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2100 Drivers in Stand-alone Mode

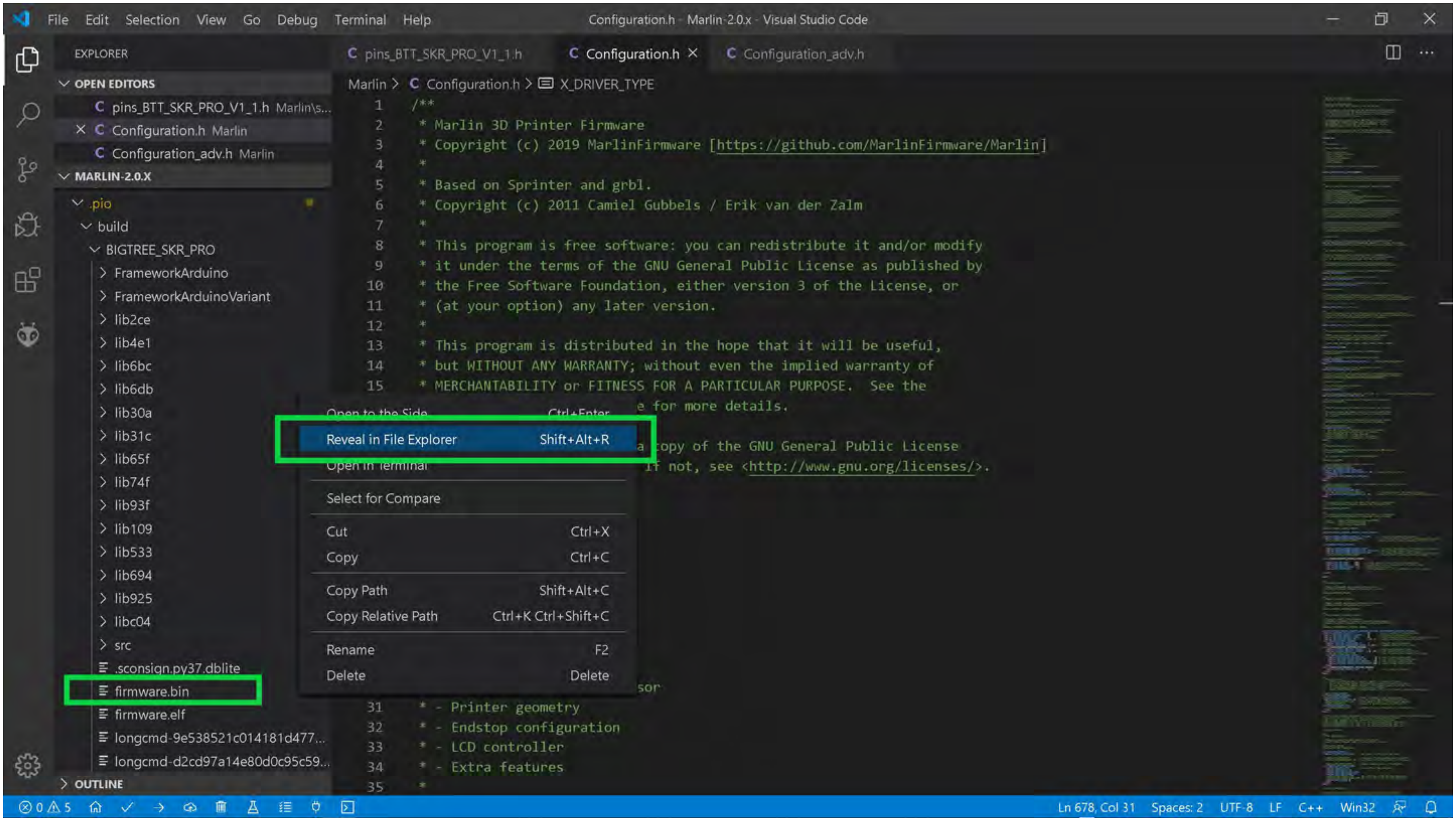
- The end of Marlin setup for BIQU TMC2100 drivers in stand-alone mode. Now, compile your code by clicking on the check mark, as seen in the **YELLOW** box below. Once the firmware has compiled successfully you will see the following messages, as shown in the **GREEN** boxes below.



- For the SKR PRO V1.1 board you must save the "firmware.bin" file to the micro SD card then place the micro SD card into the micro SD card reader of the SKR PRO V1.1 board and turn on the power to your printer for the NEW firmware to be automatically applied to the SKR PRO MCU. To locate the "firmware.bin" file, and learn how to transfer it to your micro SD card go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2100 Drivers in Stand-alone Mode

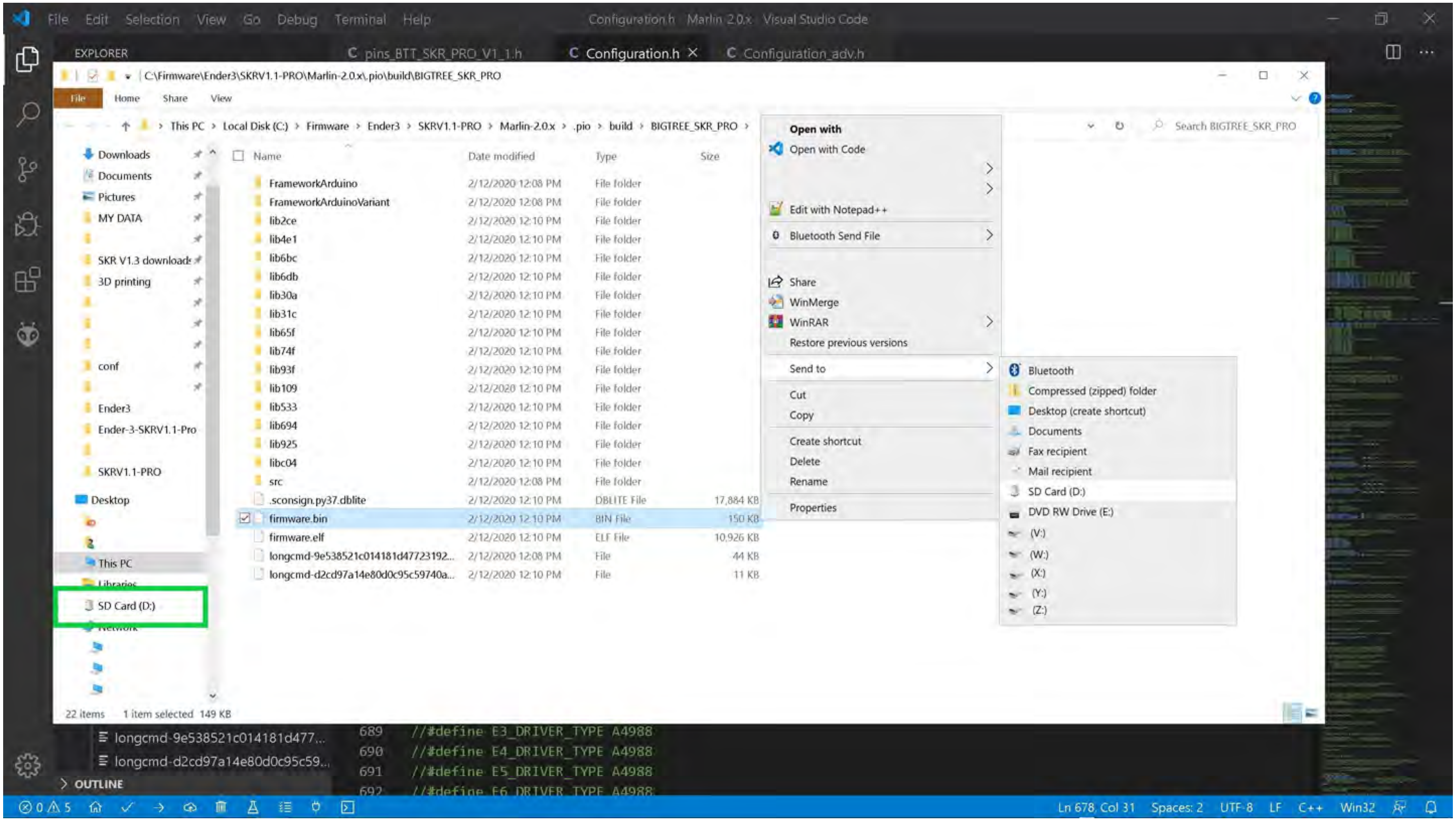
- The "firmware.bin" file can be found by using the left side of the VS code window and going to folder Marlin-2.0.X/.pio/build/ BIGTREE_SKR_PRO/ and right clicking on the "firmware.bin" file name to bring up the content menu. Select "Reveal in File Explorer" to have your Window's machine open a file explorer window.



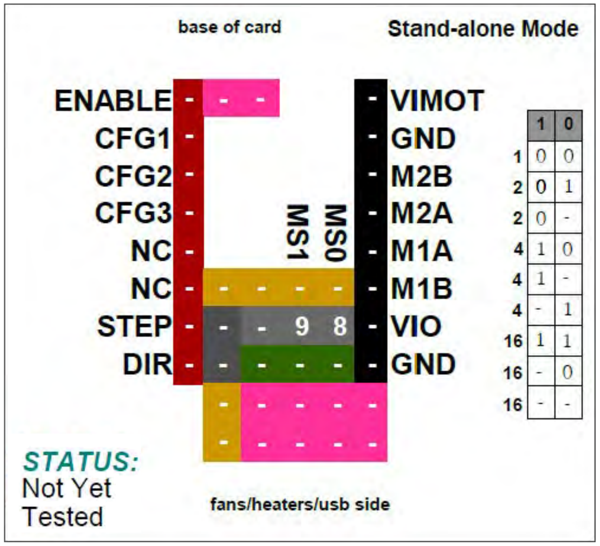
- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2100 Drivers in Stand-alone Mode

- Ensure your micro SD card, which came with your SKR PRO V1.1 board, with your SKR PRO V1.1 board is in a micro-to-SD-card adapter and plugged into your Window's SD card reader.
    - From the file explorer window right click on the "firmware.bin" file name and select "Send to" from the content menu. For my computer "D:" is my SD card reader. For your computer it may be a different drive letter. The "firmware.bin" file will be written to the SD card and renamed to "FIRMWARE.CUR".



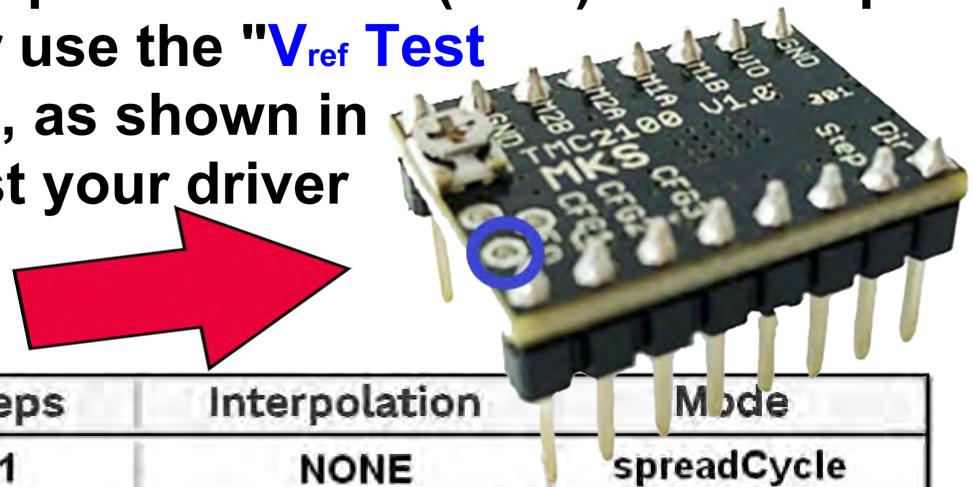
- Right click on the SD card reader, as seen in the **GREEN** box above, and select "Eject" from the context menu. Take the micro SD card out of the adapter and place it into the micro SD card reader of the SKR PRO V1.1 board. Turn on the power to the printer (i.e. the SKR PRO V1.1 board) and the "FIRMWARE.CUR" file will be uploaded to the board. After the upload the firmware file will be renamed to "firmware.bin" and stored on the micro SD card.



MKS TMC2100

Stand-alone Mode

NOTE: **Use the potentiometer (POT) on the top of the board or use the "V_{ref} Test point" location, as shown in BLUE, to adjust your driver board's V_{ref}**



Driver Chip

Makerbase

TMC2100

Stand Alone Mode

Maximum 16 Subdivision

46V DC
2.5A (peak)

MS1	MS0	Steps	Interpolation	Mode
GND	GND	1	NONE	spreadCycle
GND	VIO	1 / 2	NONE	spreadCycle
GND	OPEN	1 / 2	1 / 256	spreadCycle
VIO	GND	1 / 4	NONE	spreadCycle
VIO	OPEN	1 / 4	1 / 256	spreadCycle
OPEN	VIO	1 / 4	1 / 256	stealthChop
VIO	VIO	1 / 16	NONE	spreadCycle
OPEN	GND	1 / 16	1 / 256	spreadCycle
OPEN	OPEN	1 / 16	1 / 256	stealthChop

Driving Current Calculation Formula

R_S(Typical Sense Resistor)= 0.11Ω

$$I_{MAX}=V_{ref}$$

See Appendix B #1. Use 50% to 90% as shown below:

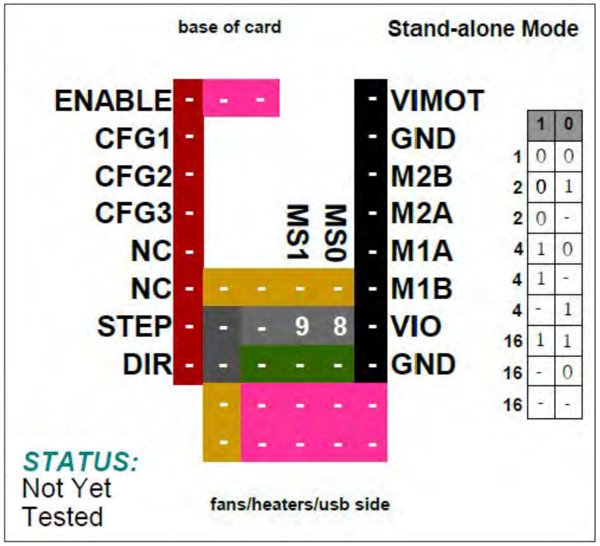
$$I_{MAX}=I_{MAX} * 0.90$$

$$V_{ref}=I_{MAX}$$

See Appendix B #1. Use 50% to 90% as shown below:

$$V_{ref}=V_{ref} * 0.90$$

NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

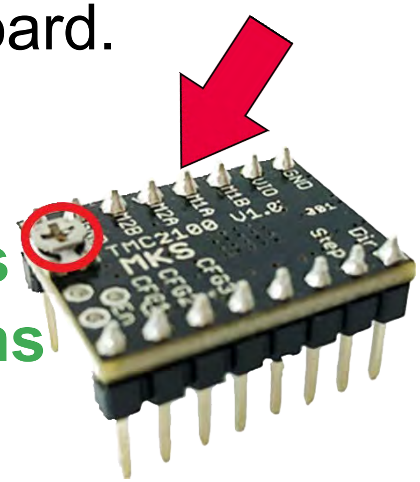


MKS TMC2100

Stand-alone Mode

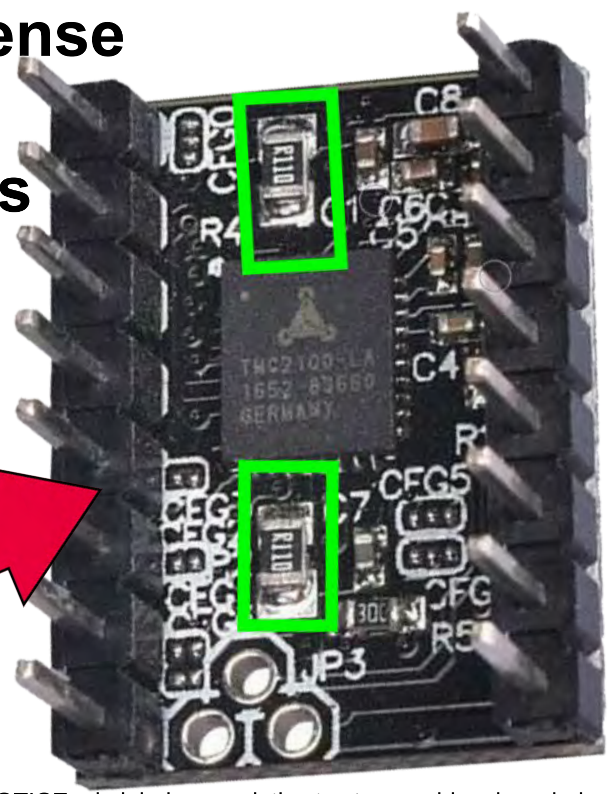
Note: Use 90% of the calculated V_{ref} when tuning the stepper driver board.

NOTE: Use the potentiometer (POT) on the top of the board (or use the board's "V_{ref} Test point") to set your V_{ref}. See **Appendix A** for instructions on how to set the V_{ref} on a driver board.



To learn more, please watch this YouTube video done by Teaching Tech and check out this link on the TMC2100 Driver

Note: Check your current sense resistors (R_s) values on the driver board, as shown in GREEN



R_s = R050 is 0.05 Ohms

R_s = R068 is 0.068 Ohms

R_s = R100 is 0.1 Ohms

R_s = R150 is 0.15 Ohms

R_s = R200 is 0.2 Ohms

R_s = R220 is 0.22 Ohms

Note: See this video about current sense resistors (R_s) and their possible locations: **https://youtu.be/8wk1elugv5A**

NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

MKS TMC2100

Stand-alone Mode

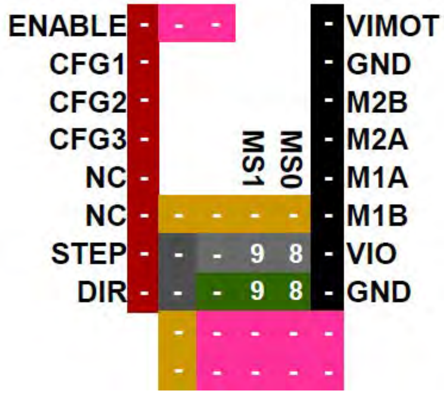
Stand-alone Mode

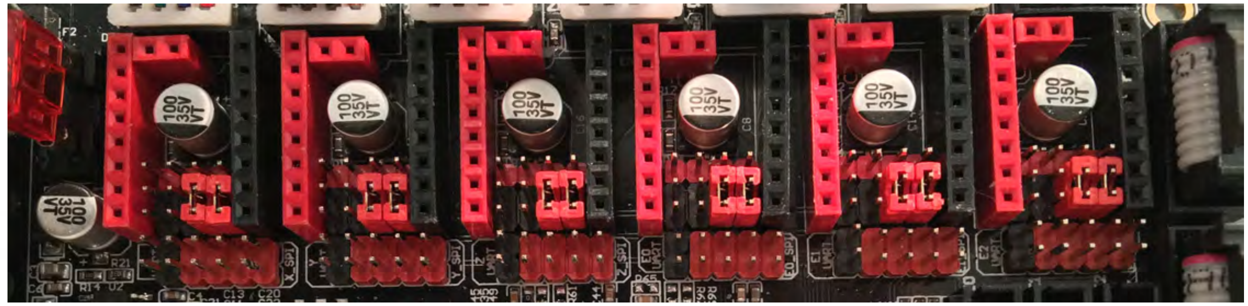
Stand-alone Mode

STEP

Interpolation: **none**

SpreadCycle



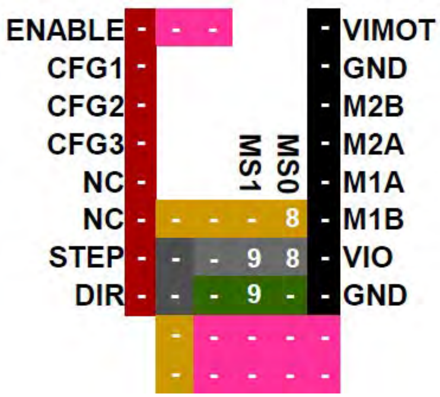


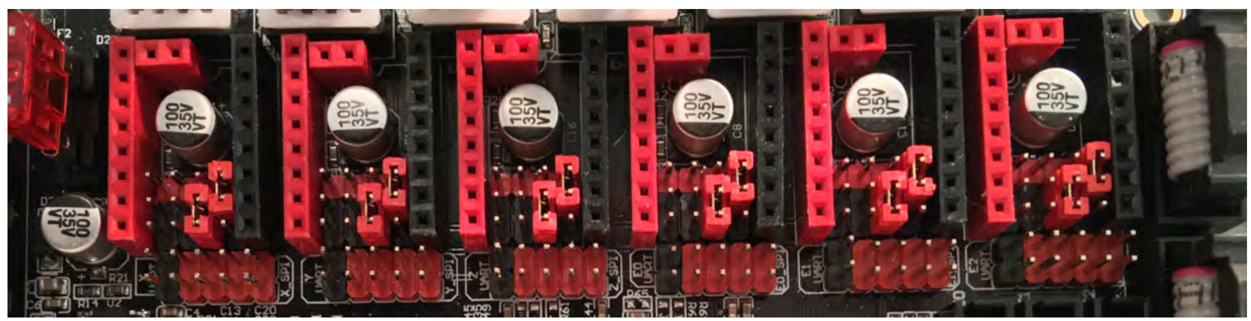
Stand-alone Mode

1 / 2

Interpolation: **none**

SpreadCycle



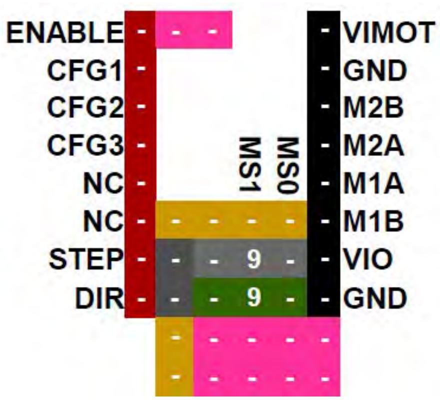


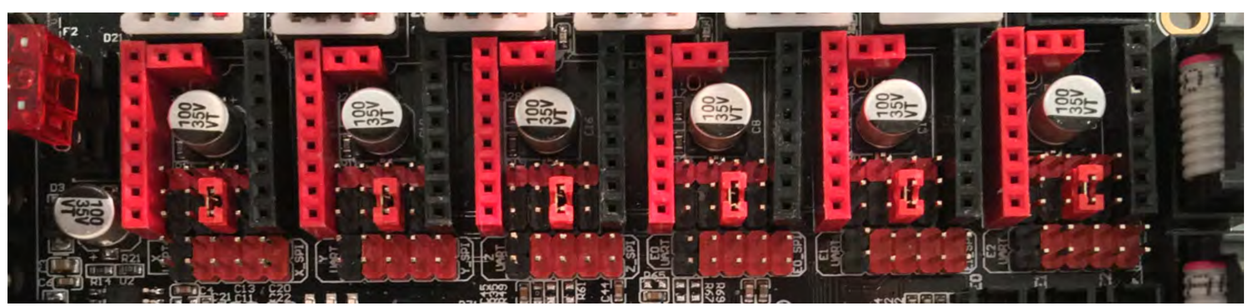
Stand-alone Mode

1 / 2

Interpolation: **1/256**

SpreadCycle





NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

MKS TMC2100

Stand-alone Mode

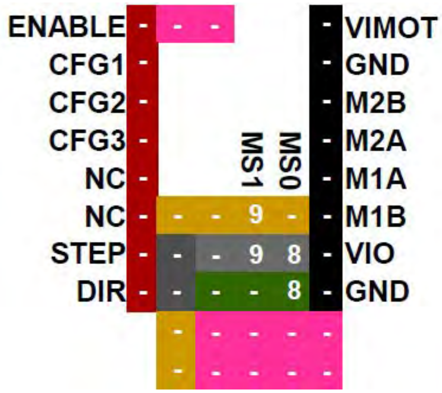
Stand-alone Mode

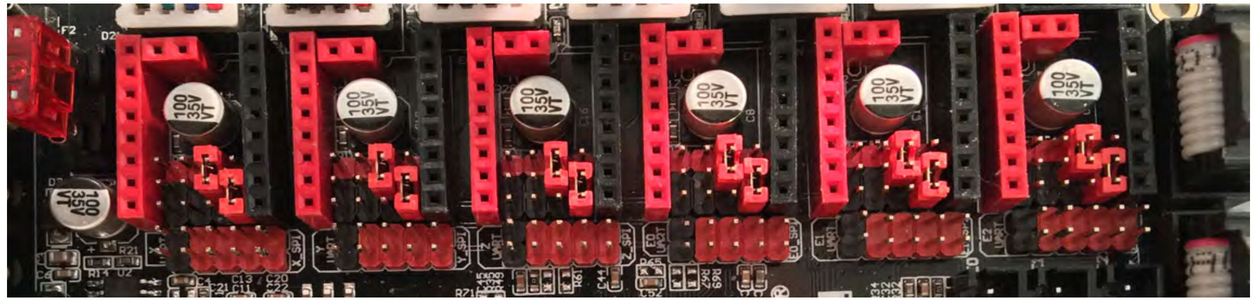
Stand-alone Mode

1 / 4

Interpolation: **none**

SpreadCycle



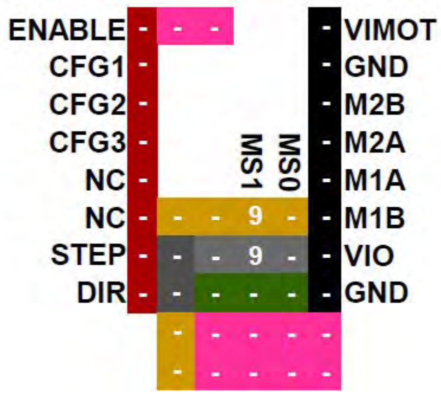


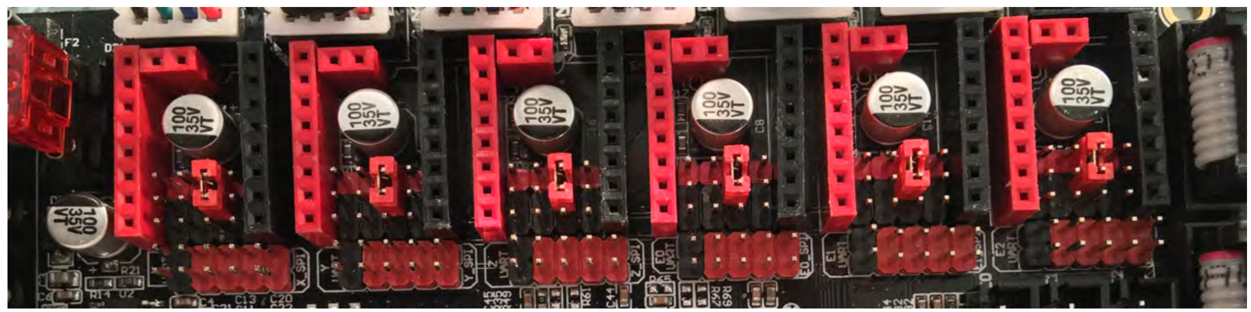
Stand-alone Mode

1 / 4

Interpolation: **1/256**

SpreadCycle



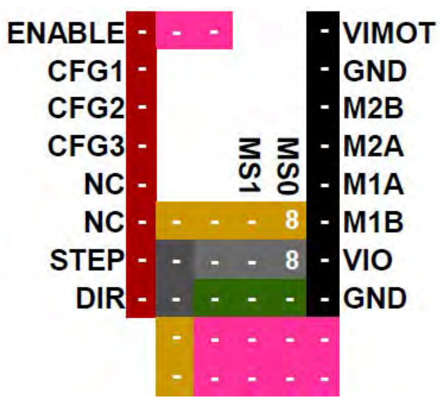


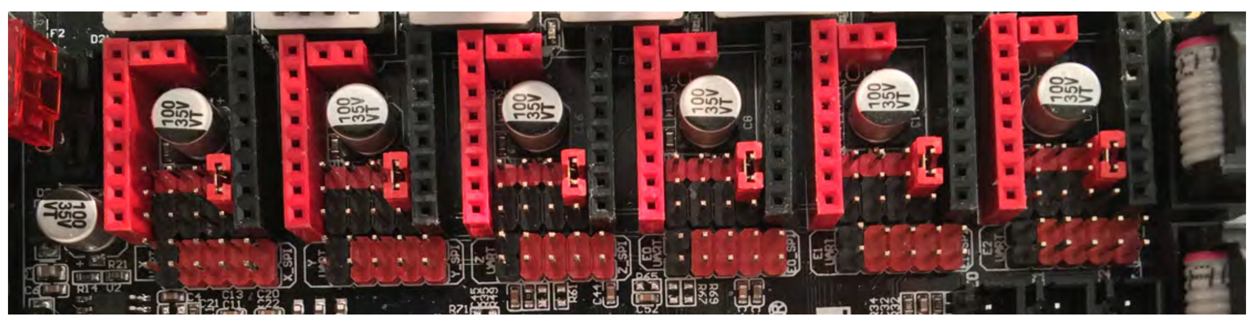
Stand-alone Mode

1 / 4

Interpolation: **1/256**

StealthChop





NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

MKS TMC2100

Stand-alone Mode

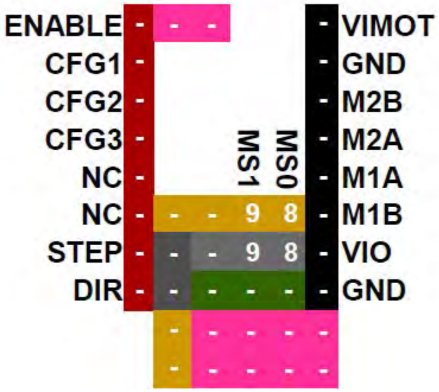
Stand-alone Mode

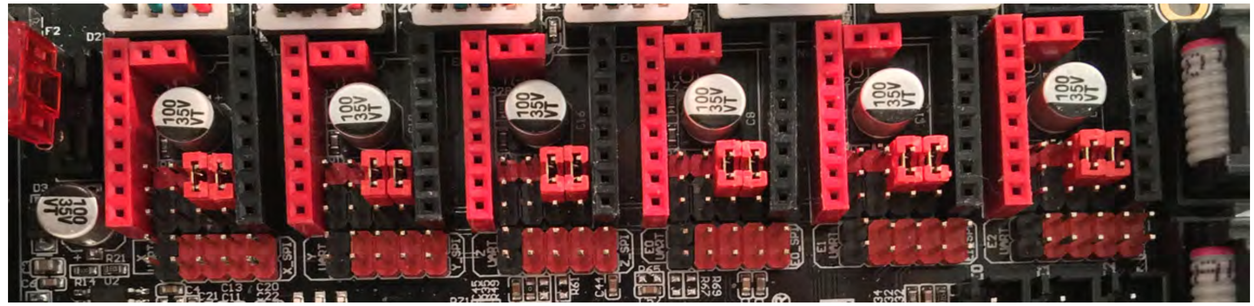
Stand-alone Mode

1 / 16

Interpolation: **none**

SpreadCycle



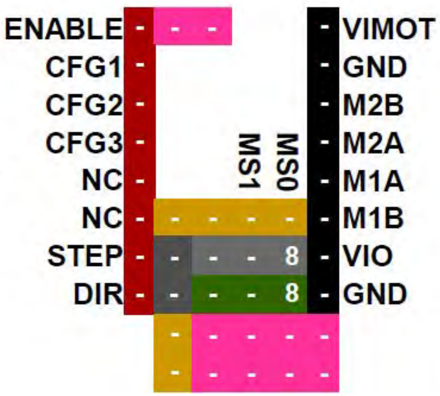


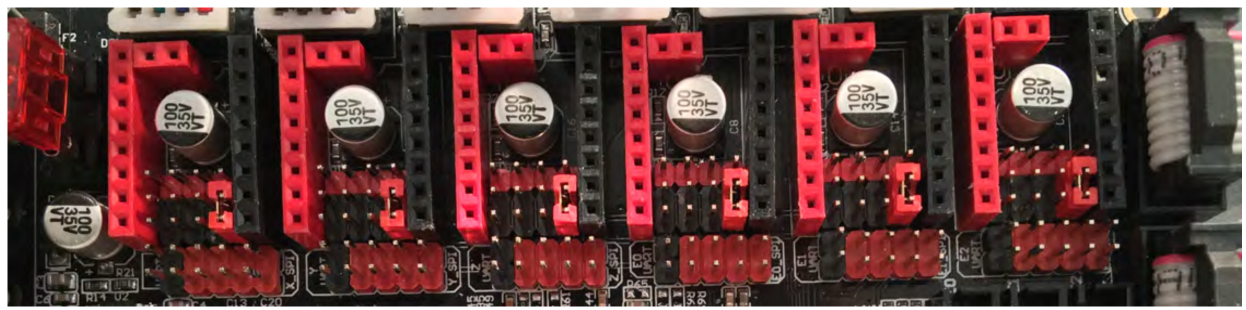
Stand-alone Mode

1 / 16

Interpolation: **1/256**

SpreadCycle



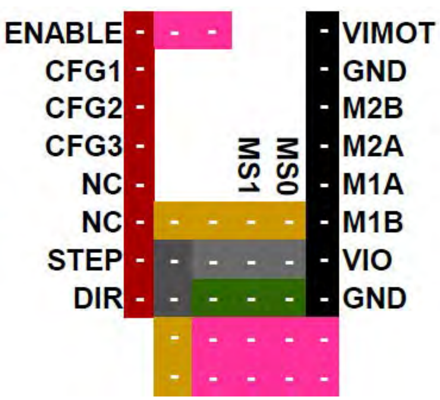


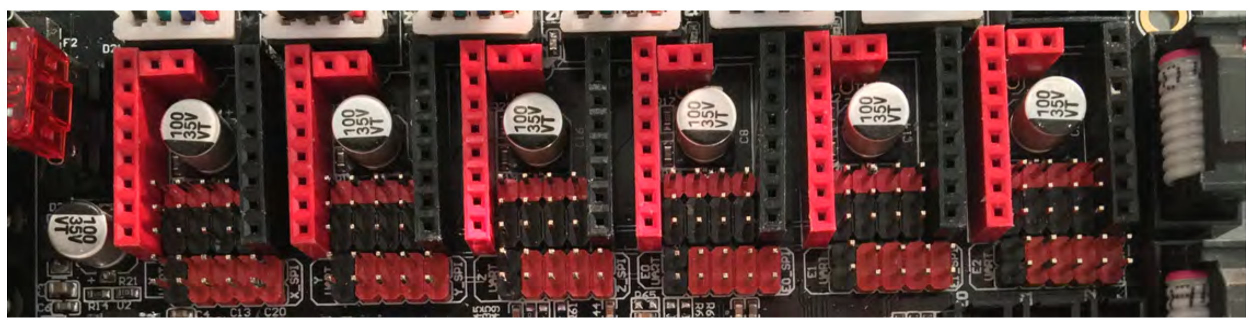
Stand-alone Mode

1 / 16

Interpolation: **1/256**

StealthChop



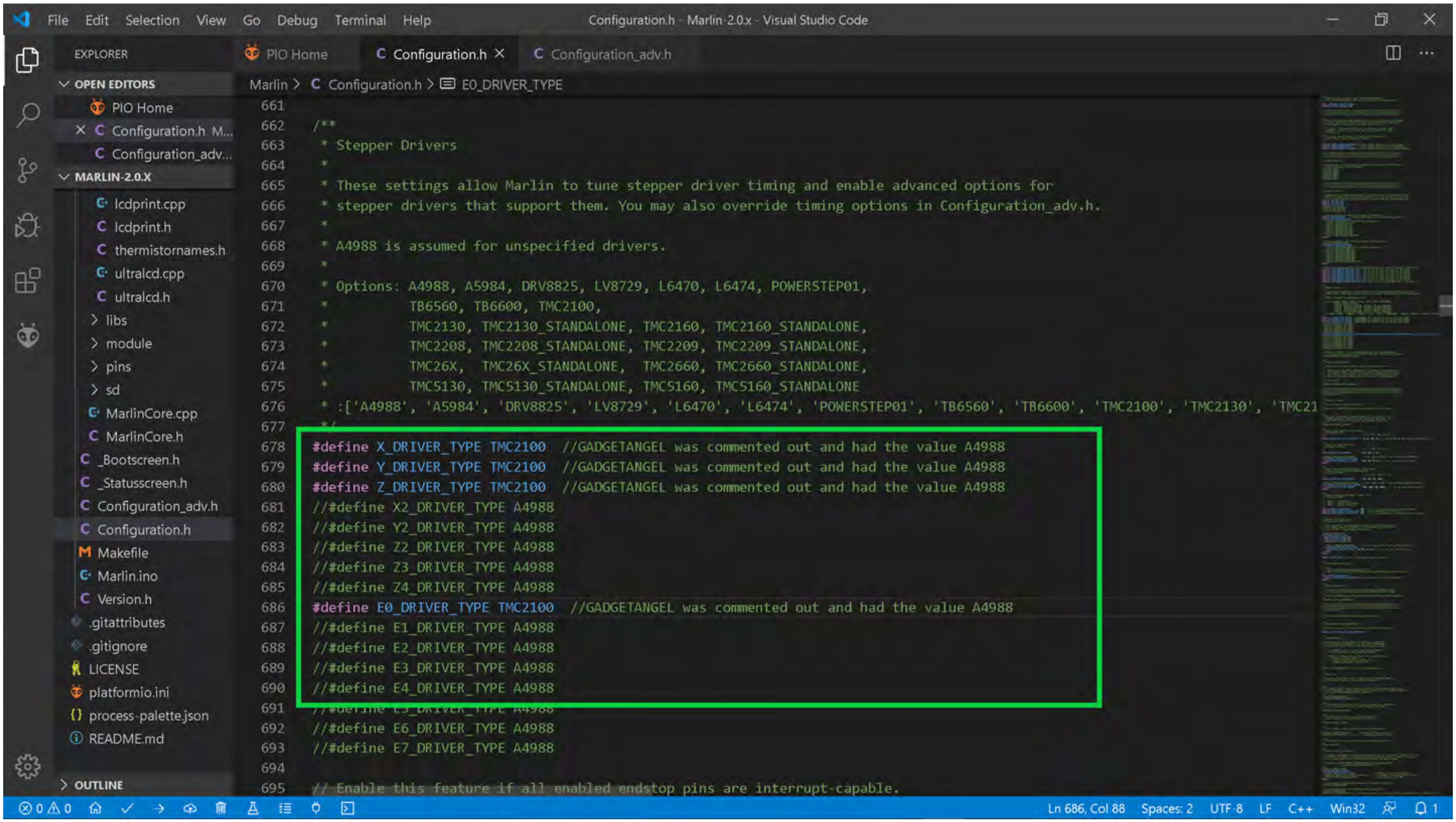


NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

The (latest release of) Marlin Setup for MKS TMC2100 Drivers in Stand-alone Mode

NOTE: **Go to Appendix C**, and then come back here for the changes to Marlin for MKS TMC2100 stepper motor drivers in stand-alone mode.

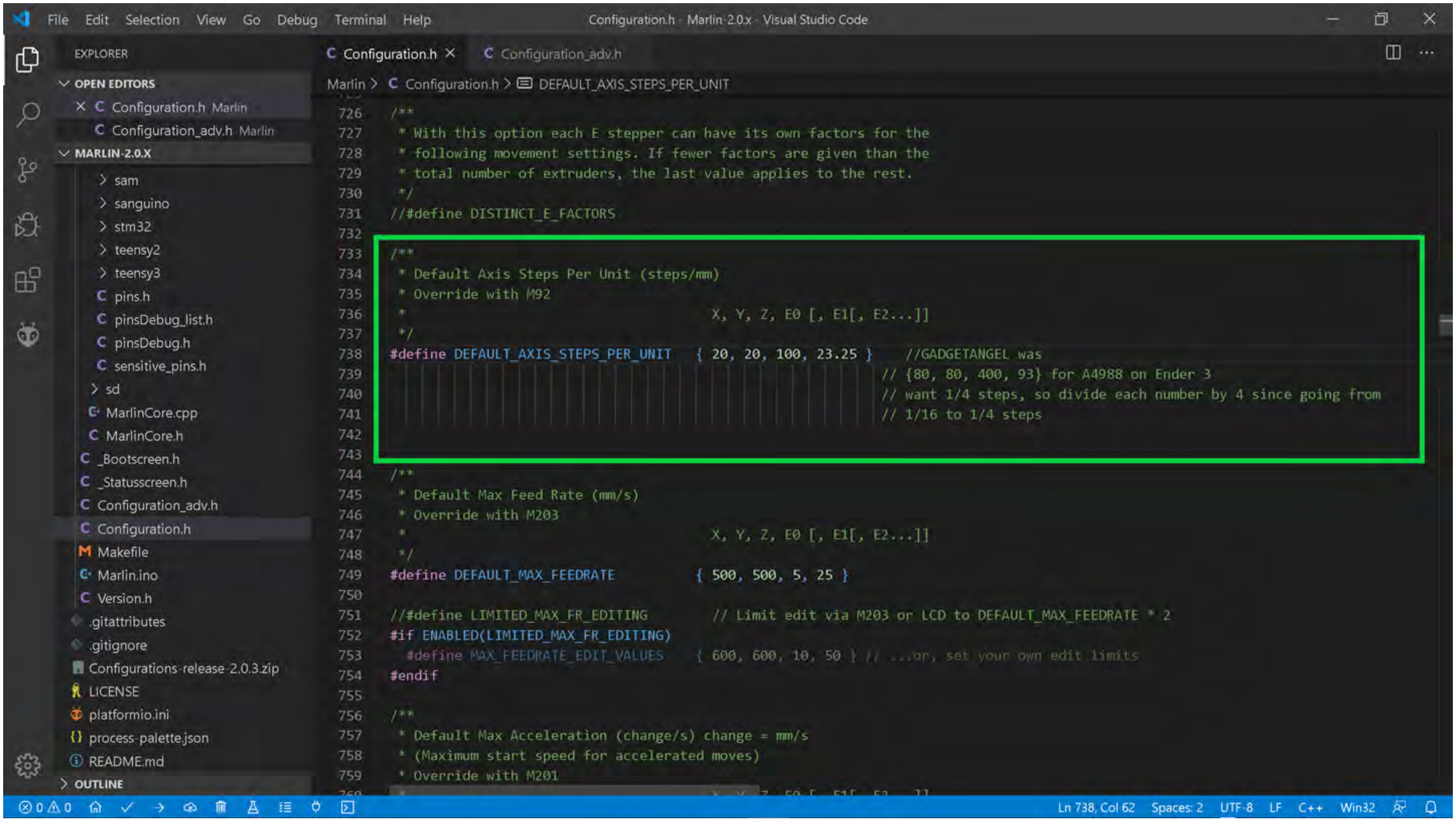
- Change the stepper motor drivers so that Marlin knows you are using TMC2100 drivers in stand-alone mode. Change one line for each axis and one for each extruder you will be using. See the picture below for an example of how to use TMC2100 drivers in stand-alone mode. When two "//" appear at the beginning of a line that means that line is commented out. To un-comment a line just remove the leading two forward slashes ("//").



- Go to the next page.

The (latest release of) Marlin Setup for MKS TMC2100 Drivers in Stand-alone Mode

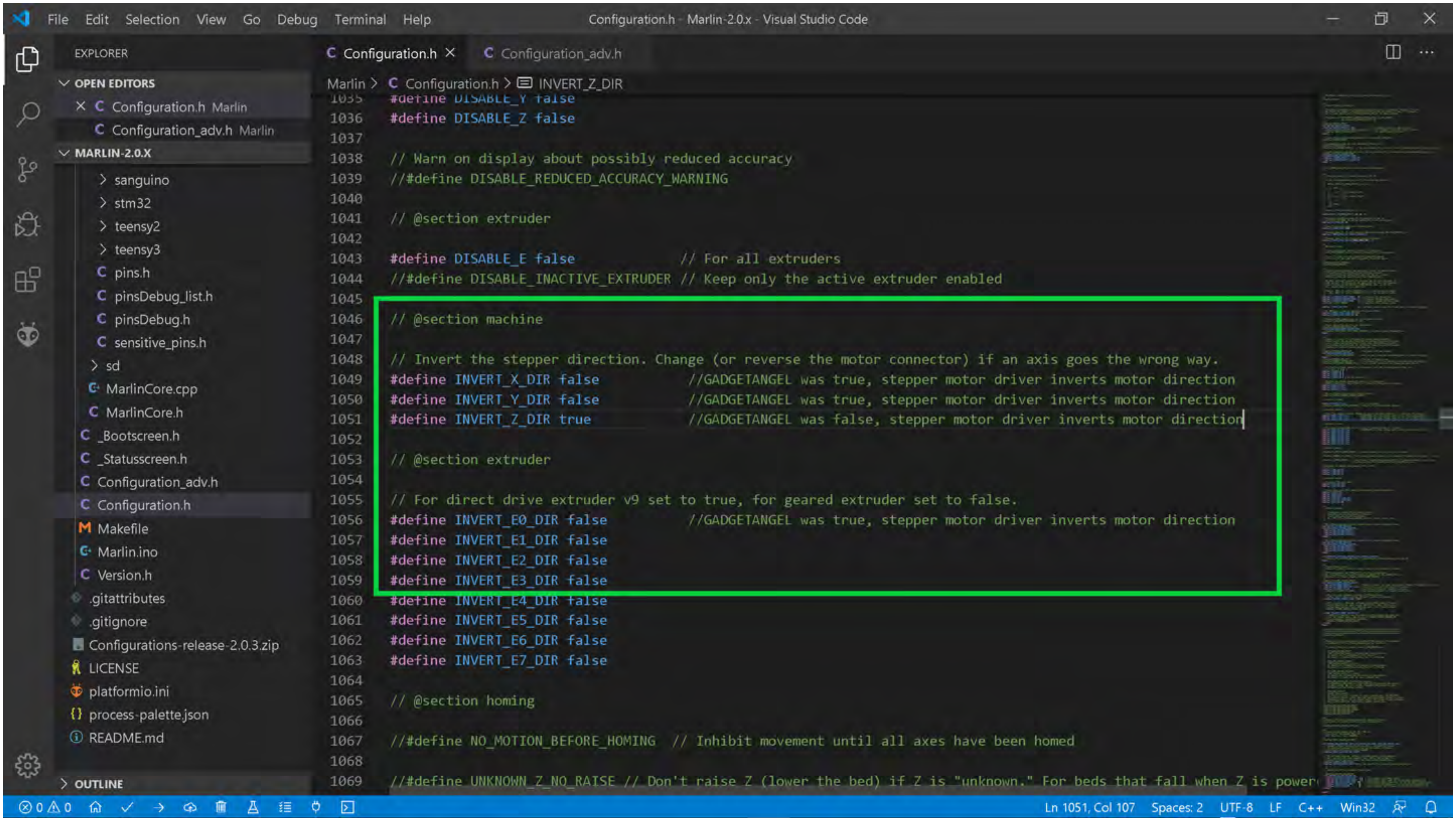
- Since I desire to use 1/4 stepping, and we are changing from A4988 stepper motor drivers on the Ender 3 to MKS TMC2100 stepper motor drivers for each axis and the extruder stepper motor driver, we will be going from 1/16 stepping to 1/4 stepping. So we are cutting our STEPS by one quarter. Therefore, we must adjust our "DEFAULT_AXIS_STEPS_PER_UNIT" anytime our STEPS are NOT 1/16. So change "DEFAULT_AXIS_STEPS_PER_UNIT" to {20, 20, 100, 23.25}, as seen in the **GREEN** box below.



- Go to the next page.

The (latest release of) Marlin Setup for MKS TMC2100 Drivers in Stand-alone Mode

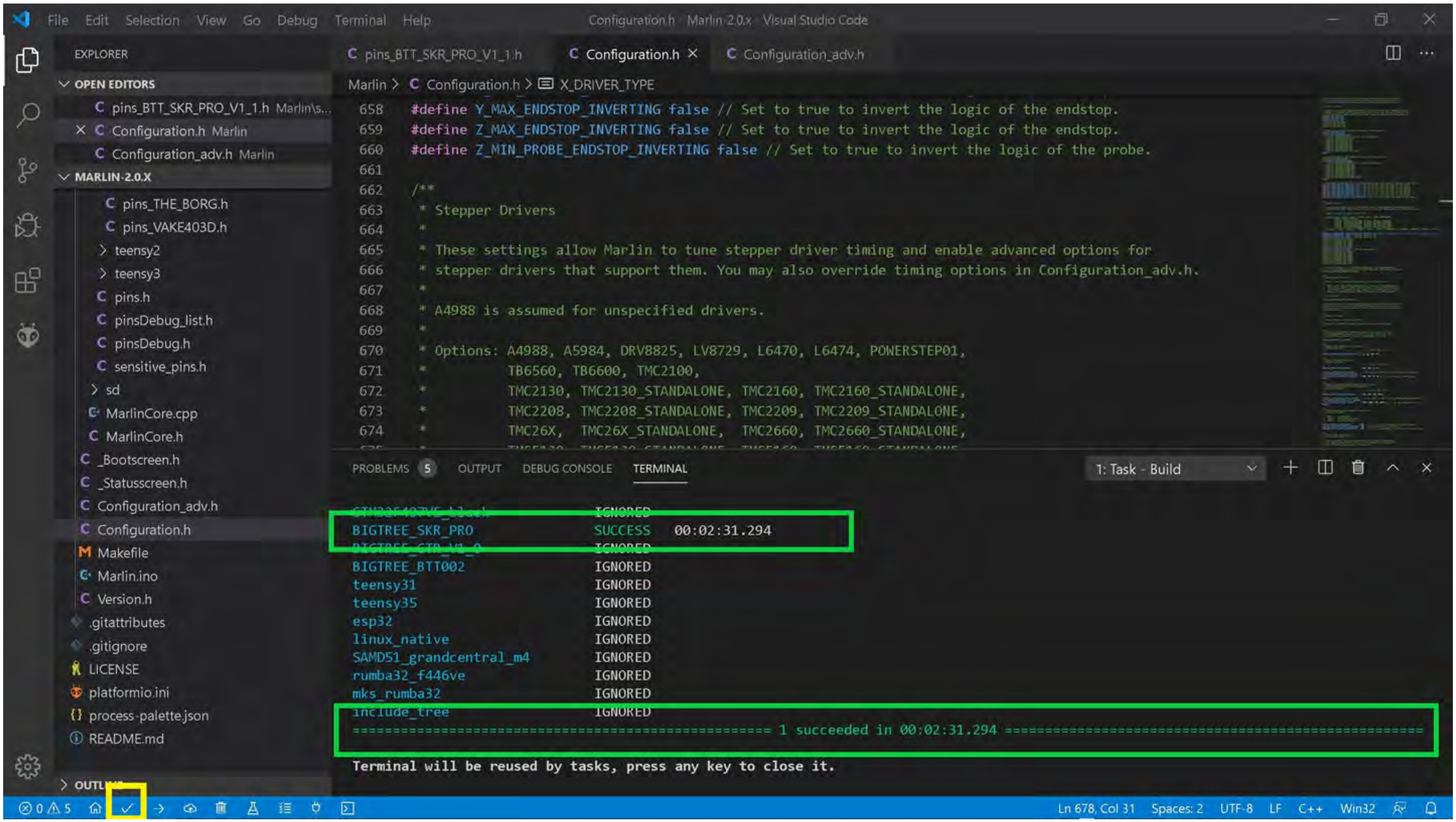
- Since the A4988 driver is what my Ender 3 used, but, now I want to use TMC2100 drivers, I must invert the stepper motor direction because the TMC2100 driver will turn the motors in the opposite direction than the A4988 driver's motor direction. So if the axis' setting you will be using the TMC2100 driver on was "true" change it to "false", as shown in the **GREEN** box below. If the setting was "false", now set it to "true", as show in the **GREEN** box below



- Go to the next page.

The (latest release of) Marlin Setup for MKS TMC2100 Drivers in Stand-alone Mode

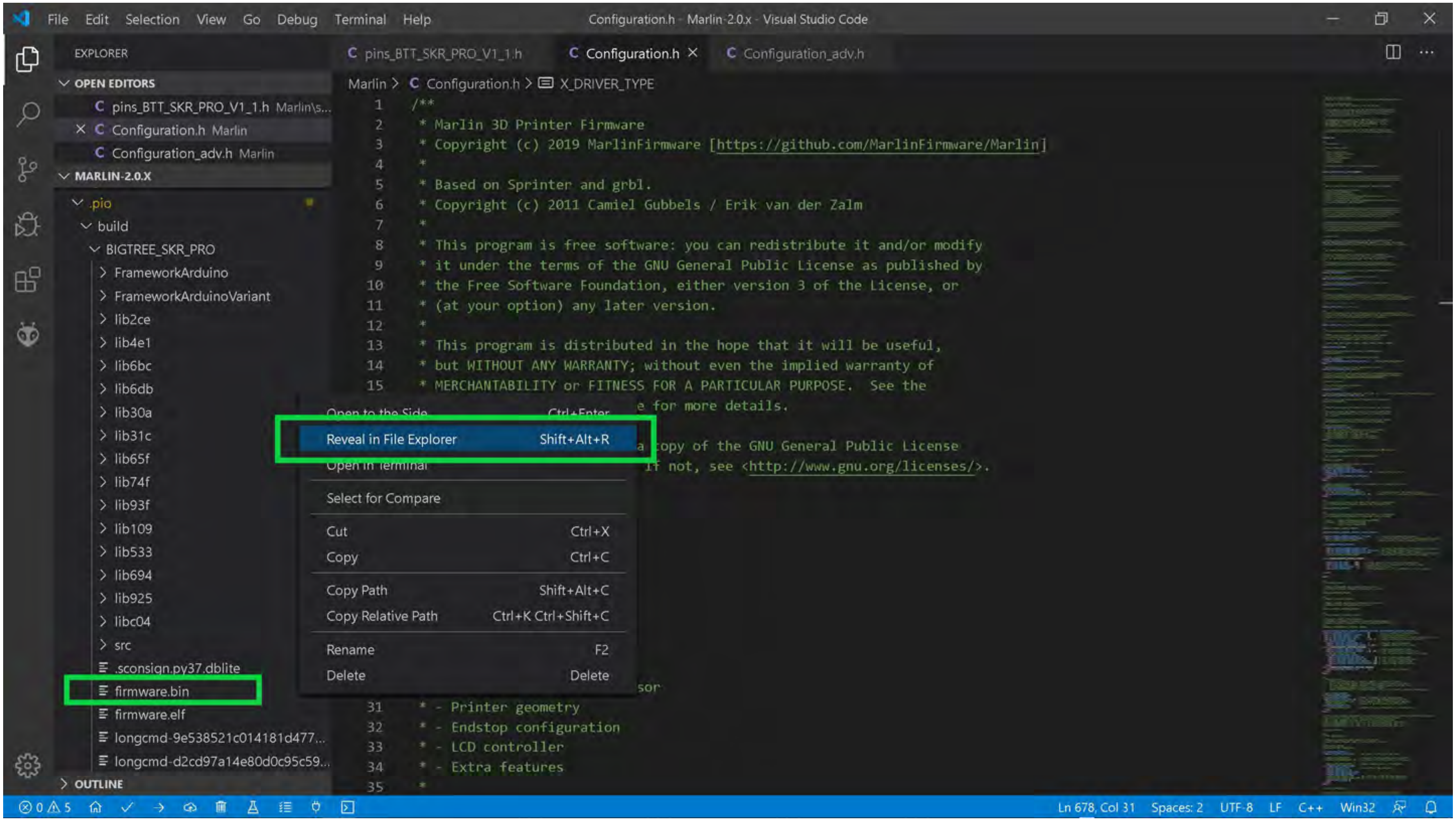
- The end of Marlin setup for MKS TMC2100 drivers in stand-alone mode. Now, compile your code by clicking on the check mark, as seen in the **YELLOW** box below. Once the firmware has compiled successfully you will see the following messages, as shown in the **GREEN** boxes below.



- For the SKR PRO V1.1 board you must save the "firmware.bin" file to the micro SD card then place the micro SD card into the micro SD card reader of the SKR PRO V1.1 board and turn on the power to your printer for the NEW firmware to be automatically applied to the SKR PRO MCU. To locate the "firmware.bin" file, and learn how to transfer it to your micro SD card go to the next page.

The (latest release of) Marlin Setup for MKS TMC2100 Drivers in Stand-alone Mode

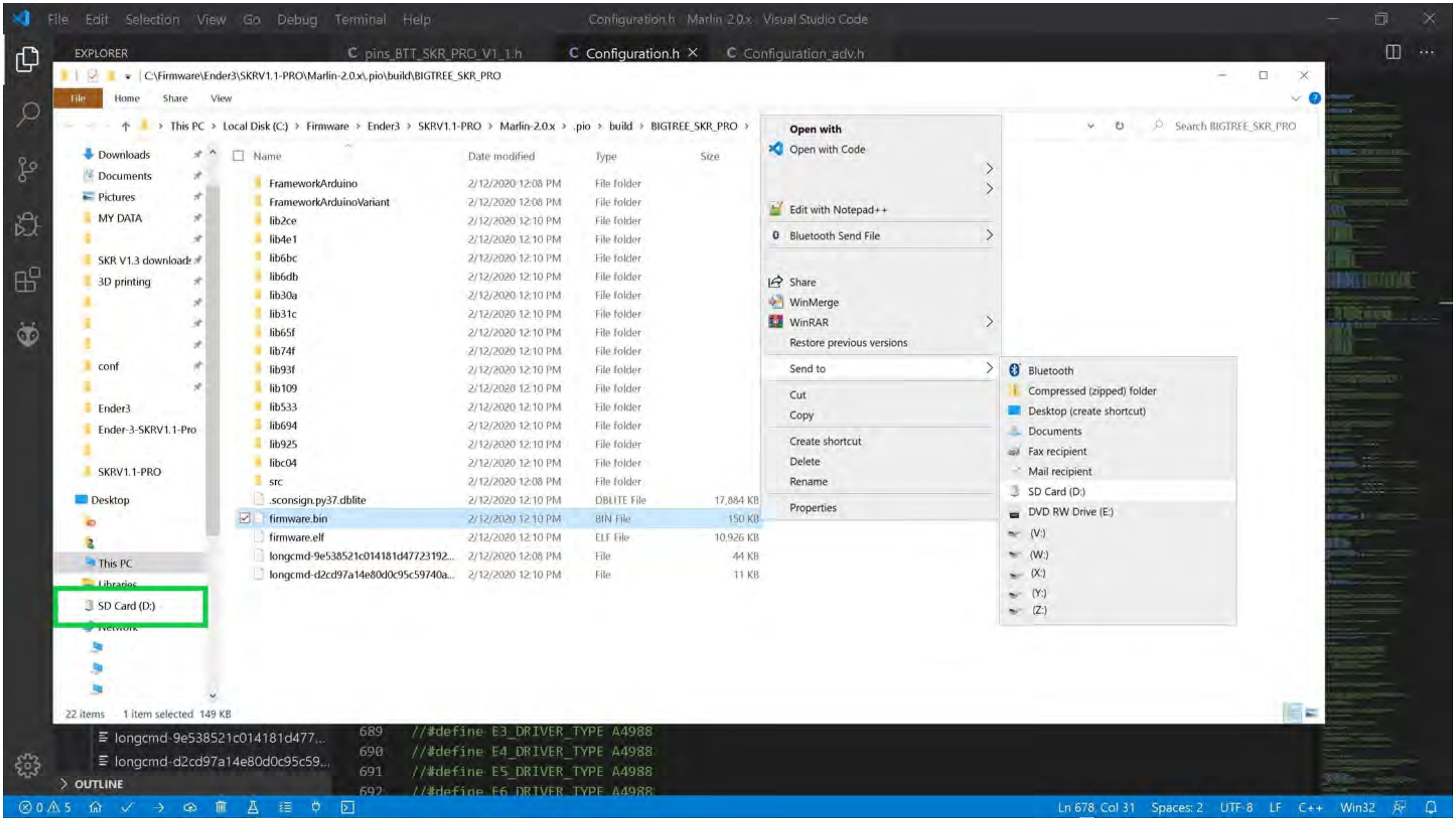
- The "firmware.bin" file can be found by using the left side of the VS code window and going to folder Marlin-2.0.X/.pio/build/ BIGTREE_SKR_PRO/ and right clicking on the "firmware.bin" file name to bring up the content menu. Select "Reveal in File Explorer" to have your Window's machine open a file explorer window.



- Go to the next page.

The (latest release of) Marlin Setup for MKS TMC2100 Drivers in Stand-alone Mode

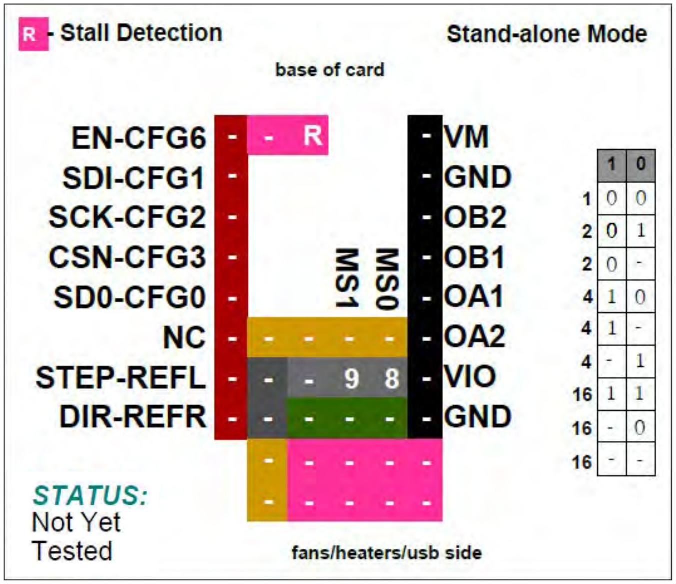
- Ensure your micro SD card, which came with your SKR PRO V1.1 board, with your SKR PRO V1.1 board is in a micro-to-SD-card adapter and plugged into your Window's SD card reader.
    - From the file explorer window right click on the "firmware.bin" file name and select "Send to" from the content menu. For my computer "D:" is my SD card reader. For your computer it may be a different drive letter. The "firmware.bin" file will be written to the SD card and renamed to "FIRMWARE.CUR".



- Right click on the SD card reader, as seen in the **GREEN** box above, and select "Eject" from the context menu. Take the micro SD card out of the adapter and place it into the micro SD card reader of the SKR PRO V1.1 board. Turn on the power to the printer (i.e. the SKR PRO V1.1 board) and the "FIRMWARE.CUR" file will be uploaded to the board. After the upload the firmware file will be renamed to "firmware.bin" and stored on the micro SD card.

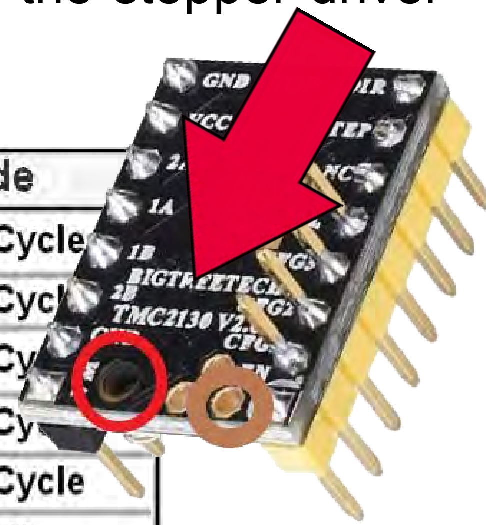
BIQU TMC2130

Stand-alone Mode



Note: Use 90% of the calculated V_{ref} when tuning the stepper driver board.

NOTE: Use the potentiometer (POT) on the top of the board to adjust your V_{ref}, as shown in **RED**; or use the "V_{ref} Test point" location on the top of the driver board, as shown in **BROWN**. See **Appendix A** for instructions on how to set the V_{ref} for the stepper driver board.



Driver Chip

BIQU®

TMC2130

Stand Alone Mode

Maximum 16 Subdivision

46V DC
2.5A (peak)

MS1	MS0	Steps	Interpolation	Mode
GND	GND	1	NONE	spreadCycle
GND	VIO	1 / 2	NONE	spreadCycle
GND	OPEN	1 / 2	1 / 256	spreadCycle
VIO	GND	1 / 4	NONE	spreadCycle
VIO	OPEN	1 / 4	1 / 256	spreadCycle
OPEN	VIO	1 / 4	1 / 256	stealthChop
VIO	VIO	1 / 16	NONE	spreadCycle
OPEN	GND	1 / 16	1 / 256	spreadCycle
OPEN	OPEN	1 / 16	1 / 256	stealthChop

Driving Current Calculation Formula

R_S (Typical Sense Resistor) = 0.11Ω

$$I_{MAX} = V_{ref}$$

See Appendix B #2. Use 50% to 90% as shown below:

$$I_{MAX} = I_{MAX} * 0.90$$

$$V_{ref} = I_{MAX}$$

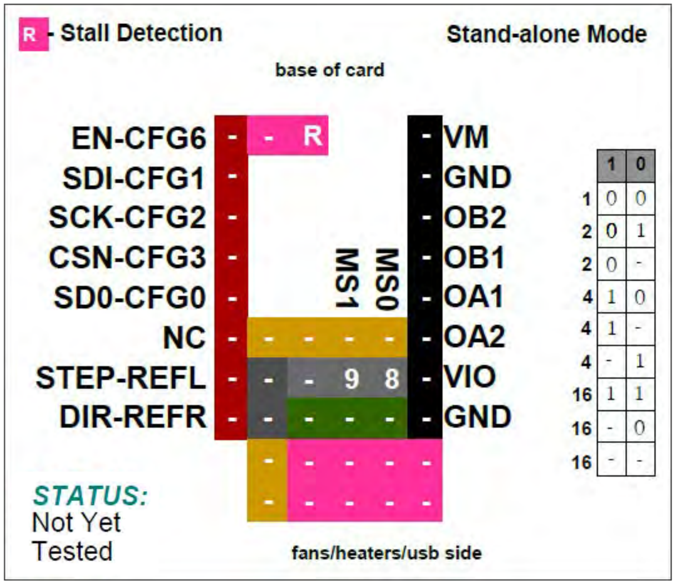
See Appendix B #2. Use 50% to 90% as shown below:

$$V_{ref} = V_{ref} * 0.90$$

NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

BIQU TMC2130

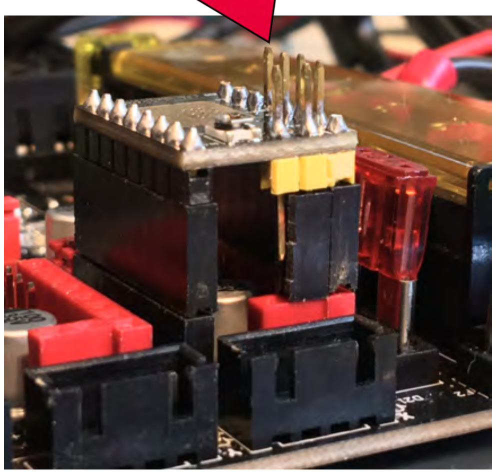
Stand-alone Mode

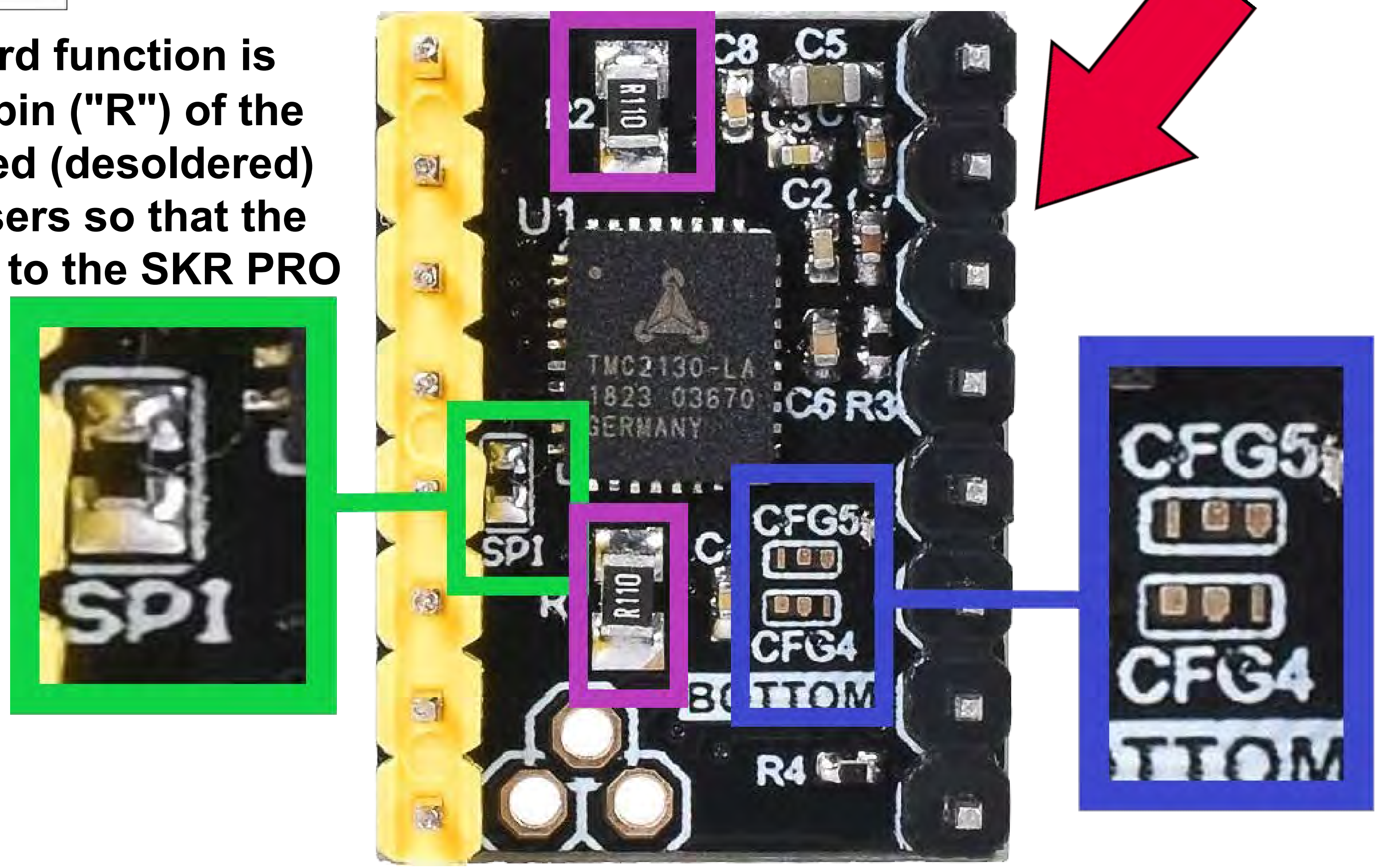


To place the BIQU TMC2130 into Stand-alone mode:

Solder the SPI Jumper together, on the bottom of the driver board to the adjacent pad, as shown in the **GREEN** box below. Ensure at CFG4 location and CFG5 location that those pads are NOT soldered together to form a bridge, as shown in **BLUE**. The **PURPLE box** shows the location of the current sense resistors (R_s).

Note: When the stall-guard function is not used, the stall-guard pin ("R") of the TMC2130 must be removed (desoldered) or use long pin header risers so that the "R" pin does not connect to the SKR PRO V1.1 board.





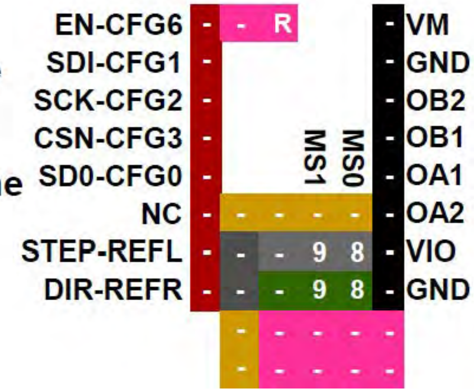
Stand-alone Mode

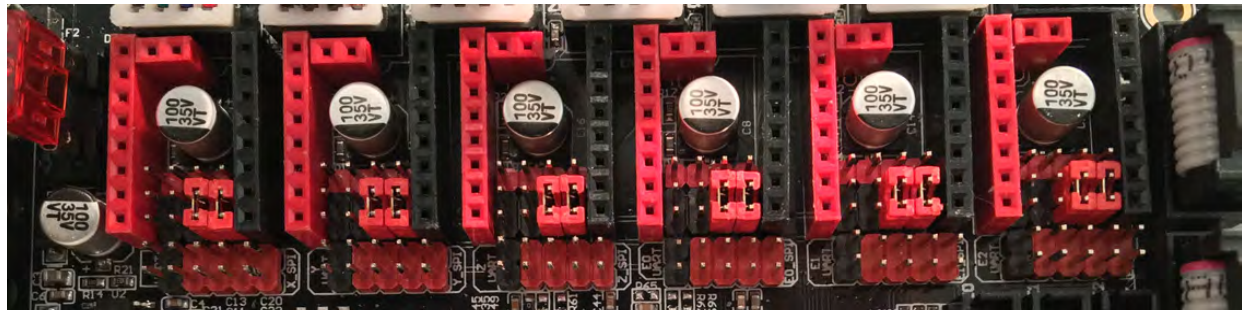
Stand-alone Mode

STEP

Interpolation: **none**

SpreadCycle



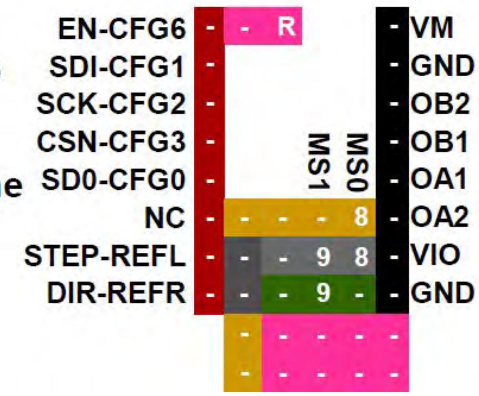


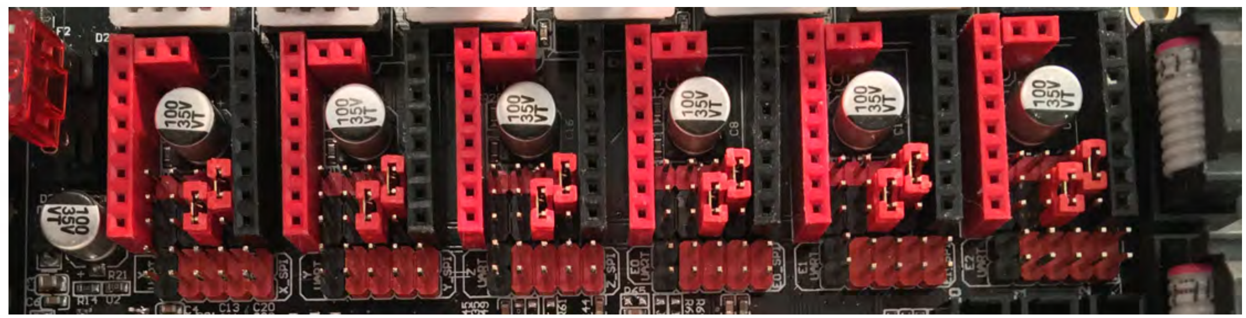
Stand-alone Mode

1 / 2

Interpolation: **none**

SpreadCycle



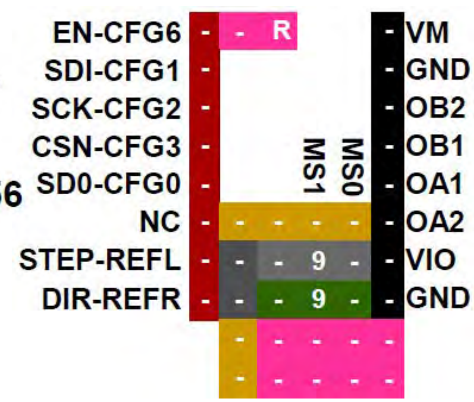


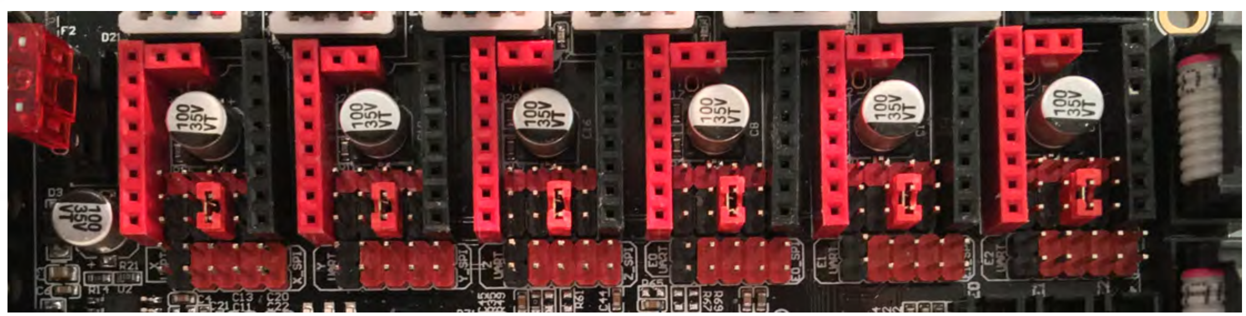
Stand-alone Mode

1 / 2

Interpolation: **1/256**

SpreadCycle





NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

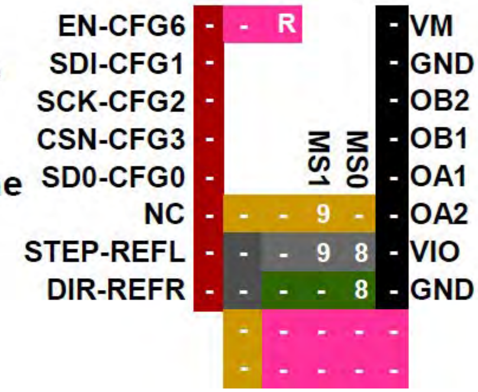
Stand-alone Mode

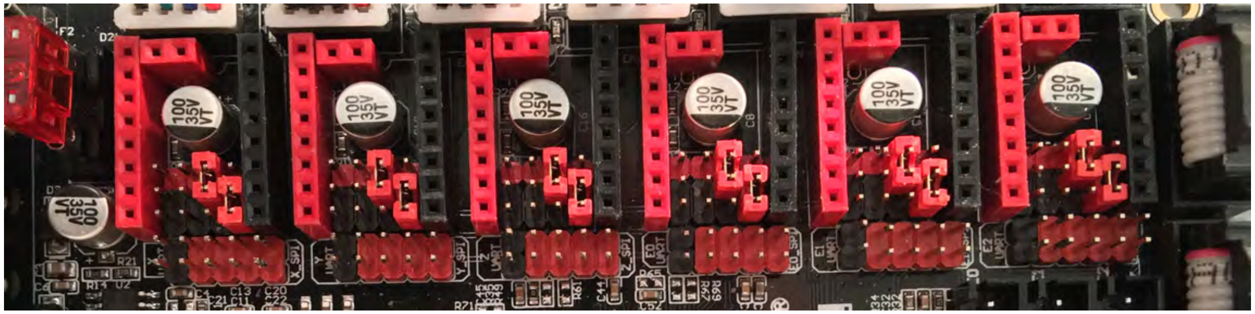
Stand-alone Mode

1 / 4

Interpolation: none

SpreadCycle



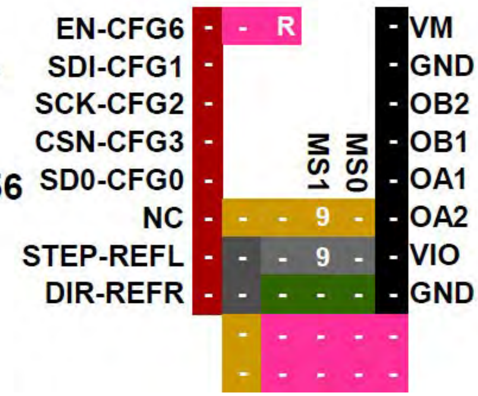


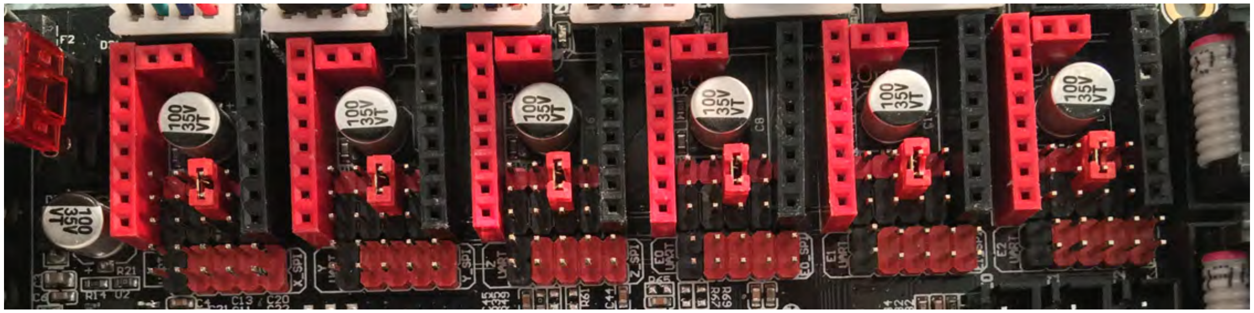
Stand-alone Mode

1 / 4

Interpolation: 1/256

SpreadCycle



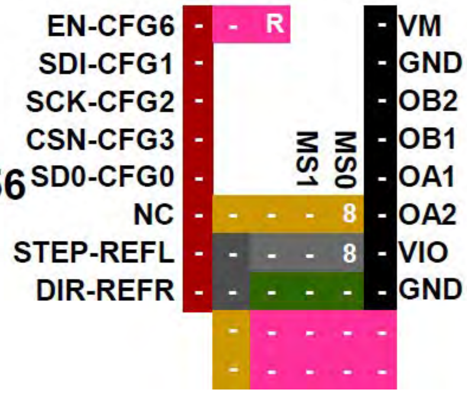


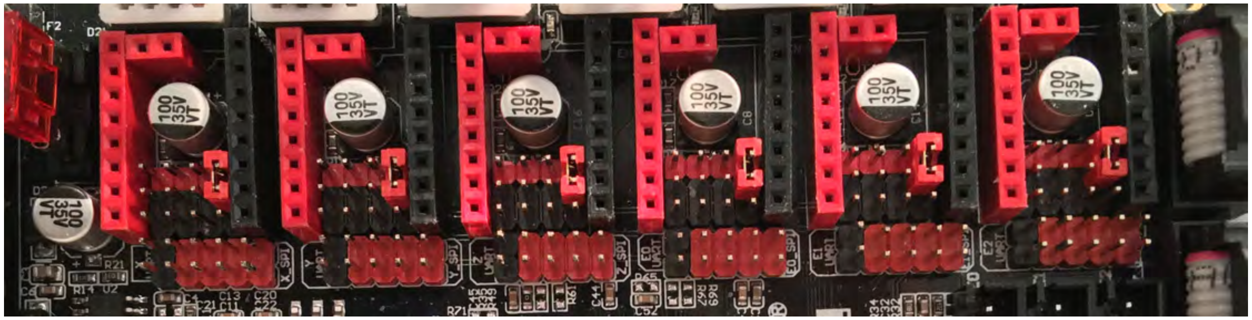
Stand-alone Mode

1 / 4

Interpolation: 1/256

StealthChop





NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

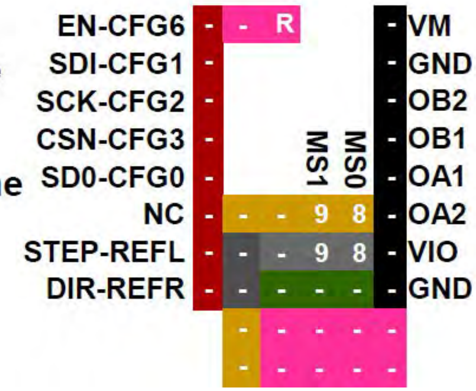
Stand-alone Mode

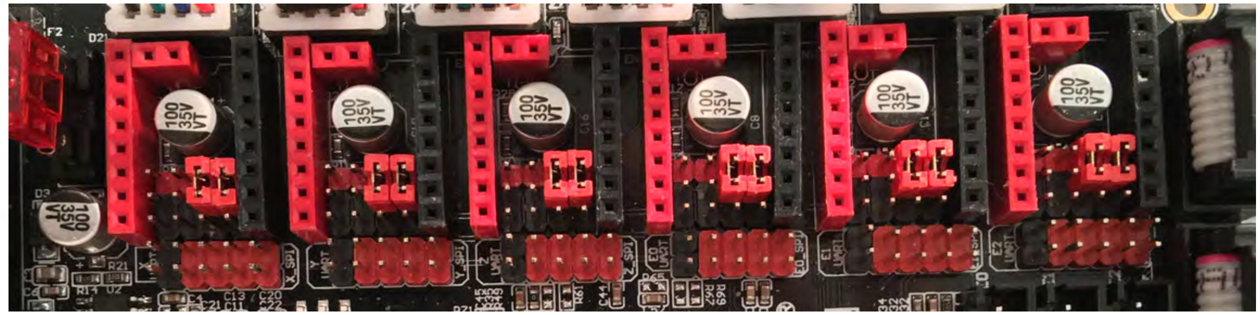
Stand-alone Mode

1 / 16

Interpolation: none

SpreadCycle



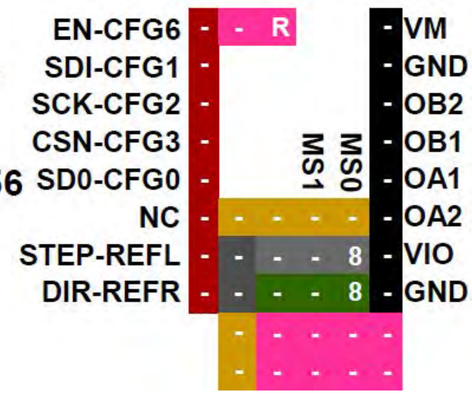


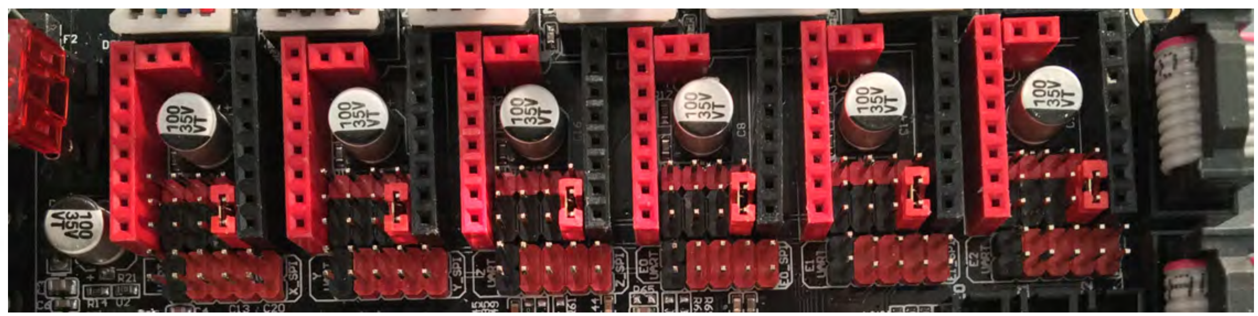
Stand-alone Mode

1 / 16

Interpolation: 1/256

SpreadCycle



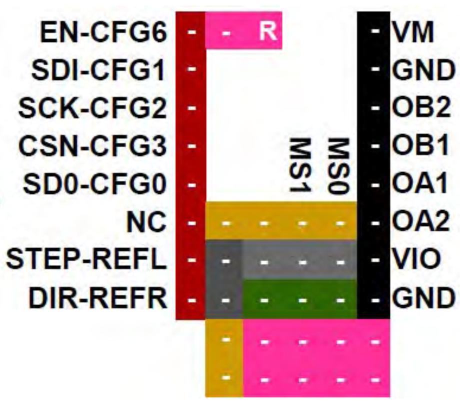


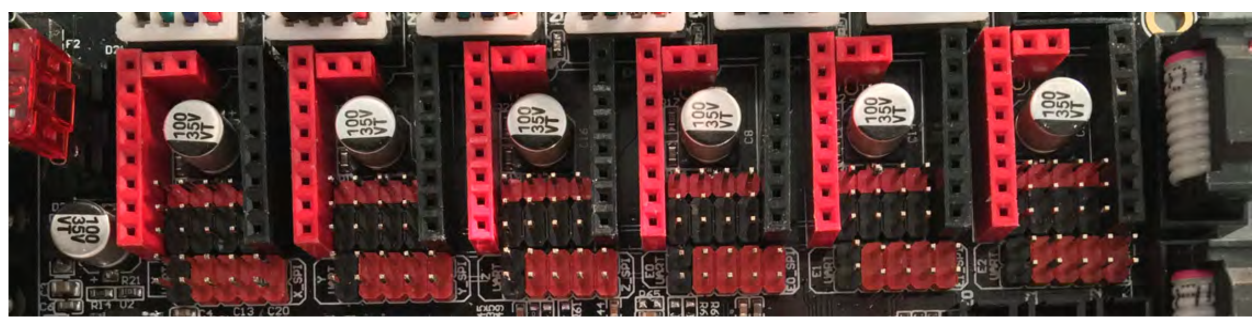
Stand-alone Mode

1 / 16

Interpolation: 1/256

StealthChop



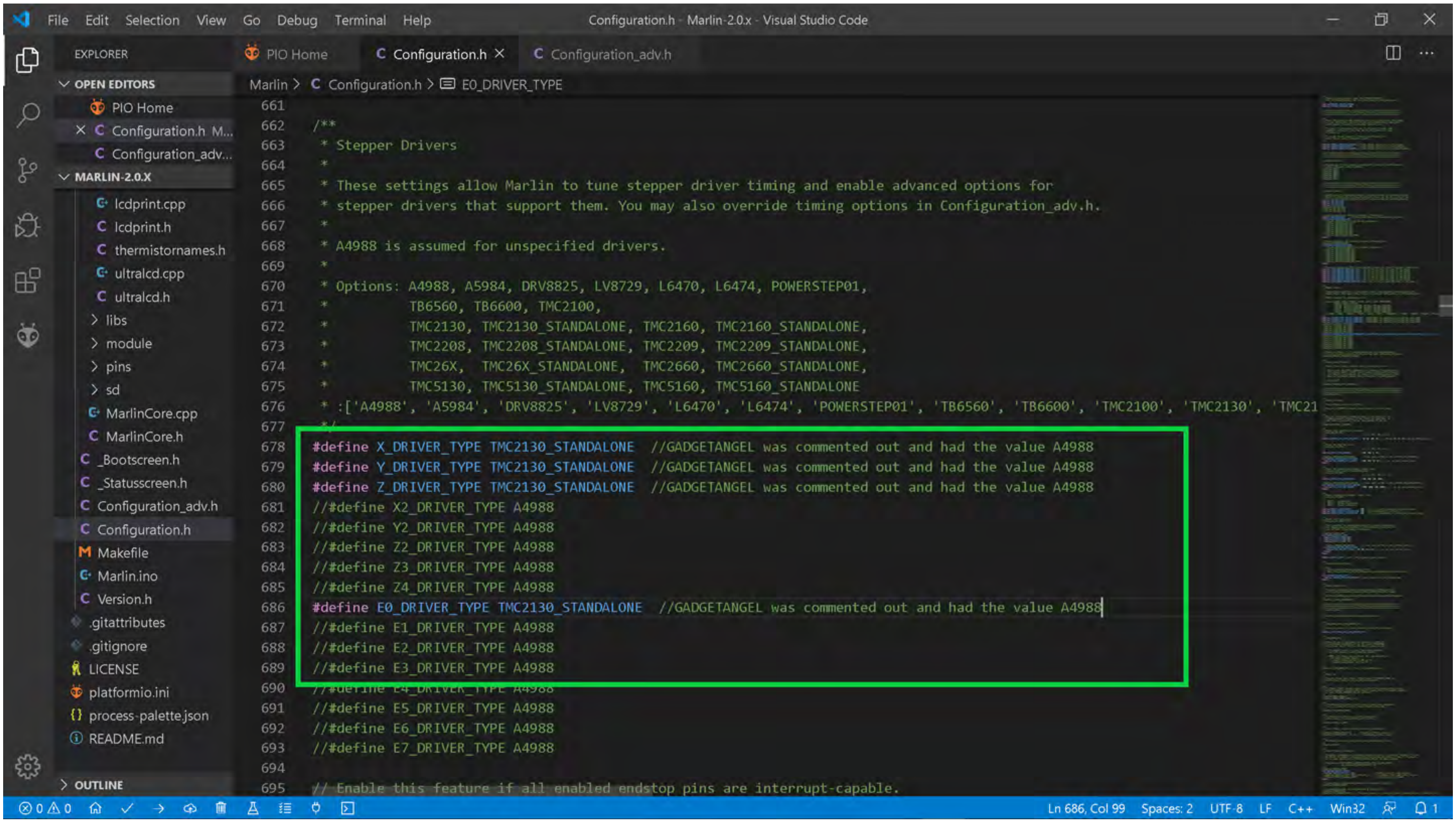


NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

The (latest release of) Marlin Setup for BIQU TMC2130 Drivers in Stand-alone Mode

NOTE: **Go to Appendix C**, and then come back here for the changes to Marlin for BIQU TMC2130 stepper motor drivers in stand-alone mode.

- Change the stepper motor drivers so that Marlin knows you are using TMC2130 drivers in stand-alone mode. Change one line for each axis and one for each extruder you will be using. See the picture below for an example of how to use TMC2130 drivers in stand-alone mode. When two "//" appear at the beginning of a line that means that line is commented out. To un-comment a line just remove the leading two forward slashes ("//").

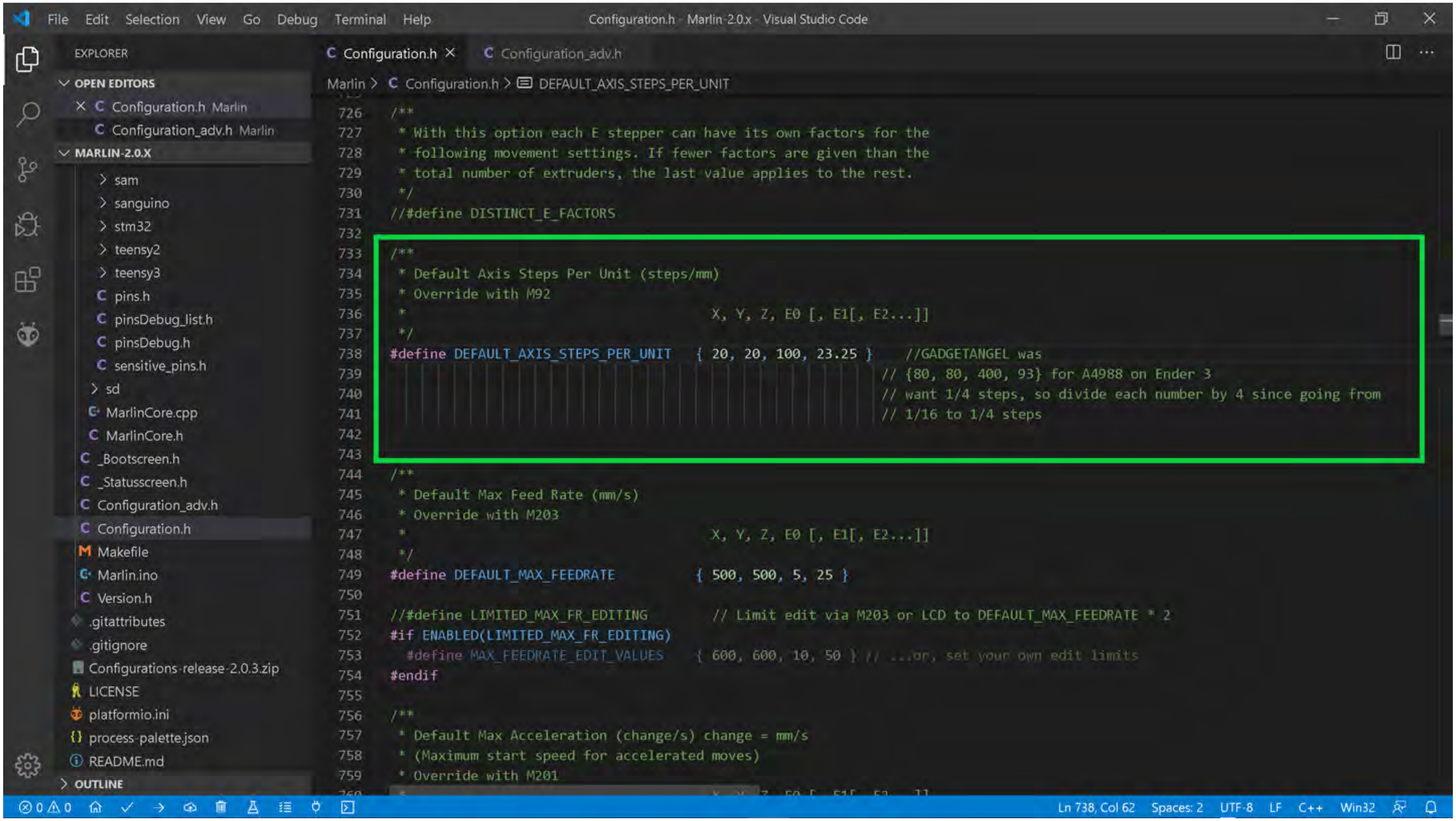


- Go to the next page.

NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

The (latest release of) Marlin Setup for BIQU TMC2130 Drivers in Stand-alone Mode

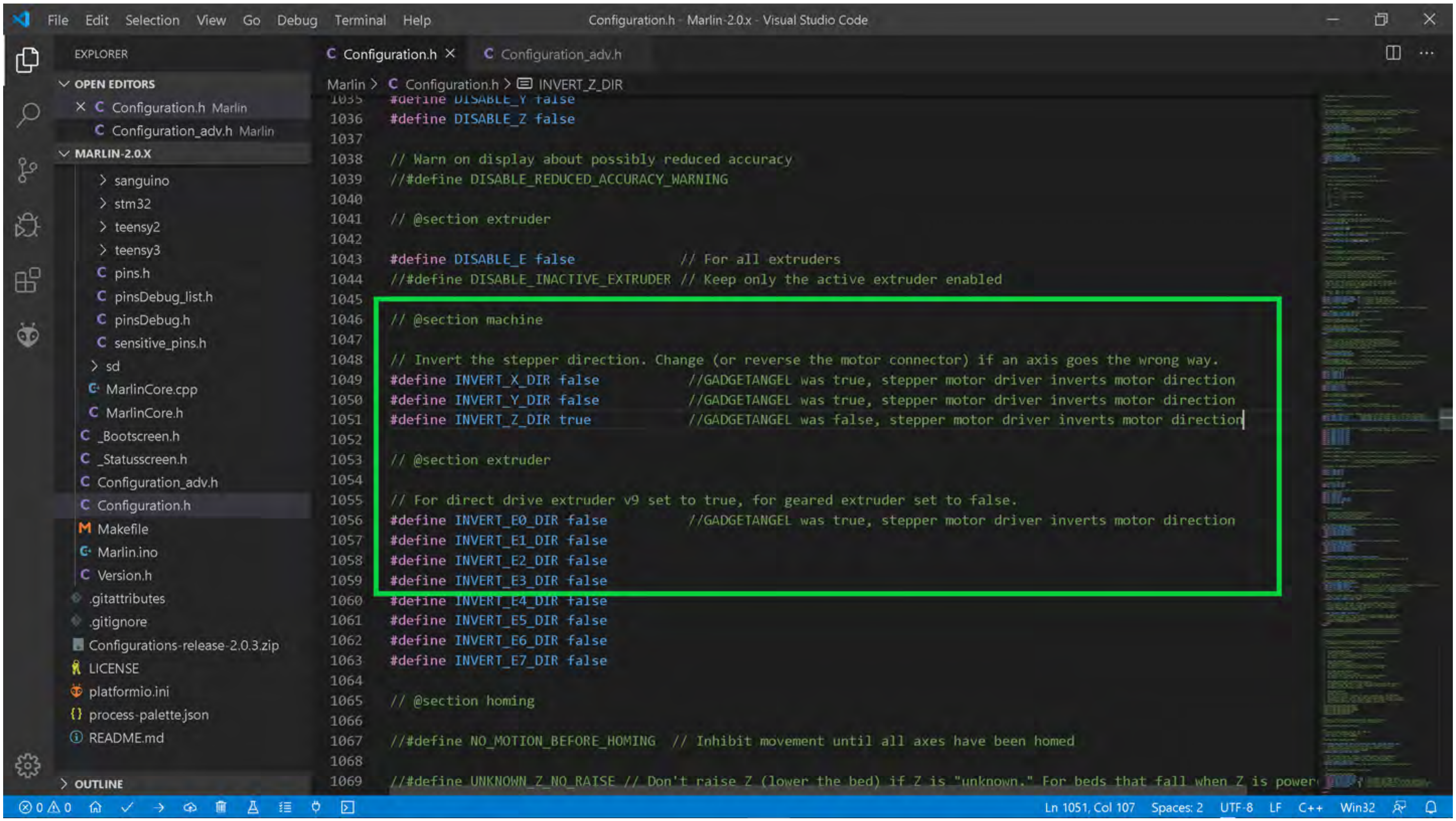
- Since I desire to use 1/4 stepping, and we are changing from A4988 stepper motor drivers on the Ender 3 to BIQU TMC2130 stepper motor drivers for each axis and the extruder stepper motor driver, we will be going from 1/16 stepping to 1/4 stepping. So we are cutting our STEPS by one quarter. Therefore, we must adjust our "DEFAULT_AXIS_STEPS_PER_UNIT" anytime our STEPS are NOT 1/16. So change "DEFAULT_AXIS_STEPS_PER_UNIT" to {20, 20, 100, 23.25}, as seen in the **GREEN** box below.



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2130 Drivers in Stand-alone Mode

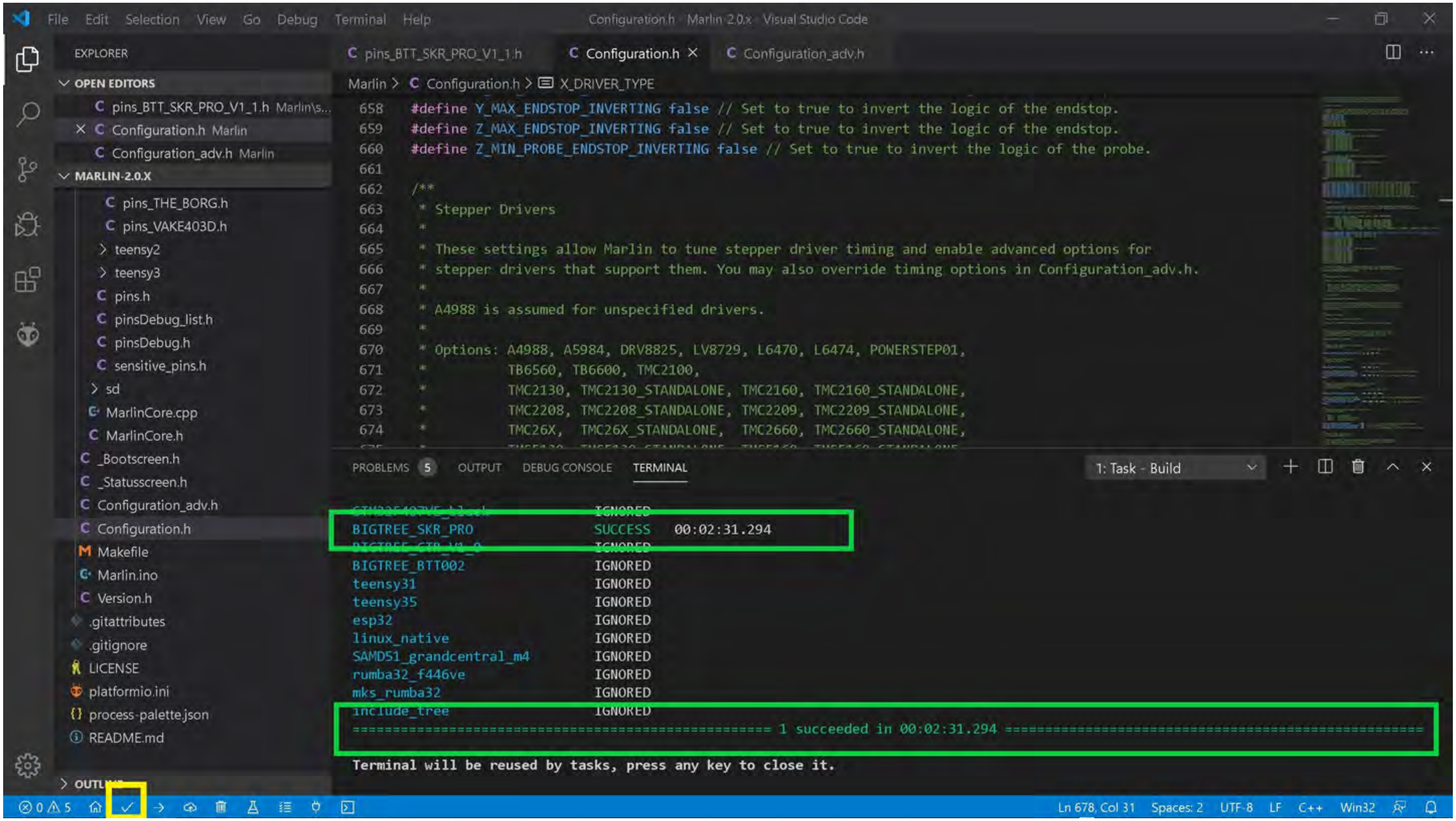
- Since the A4988 driver is what my Ender 3 used, but, now I want to use TMC2130 drivers, I must invert the stepper motor direction because the TMC2130 driver will turn the motors in the opposite direction than the A4988 driver's motor direction. So if the axis' setting you will be using the TMC2130 driver on was "true" change it to "false", as shown in the **GREEN** box below. If the setting was "false", now set it to "true", as show in the **GREEN** box below



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2130 Drivers in Stand-alone Mode

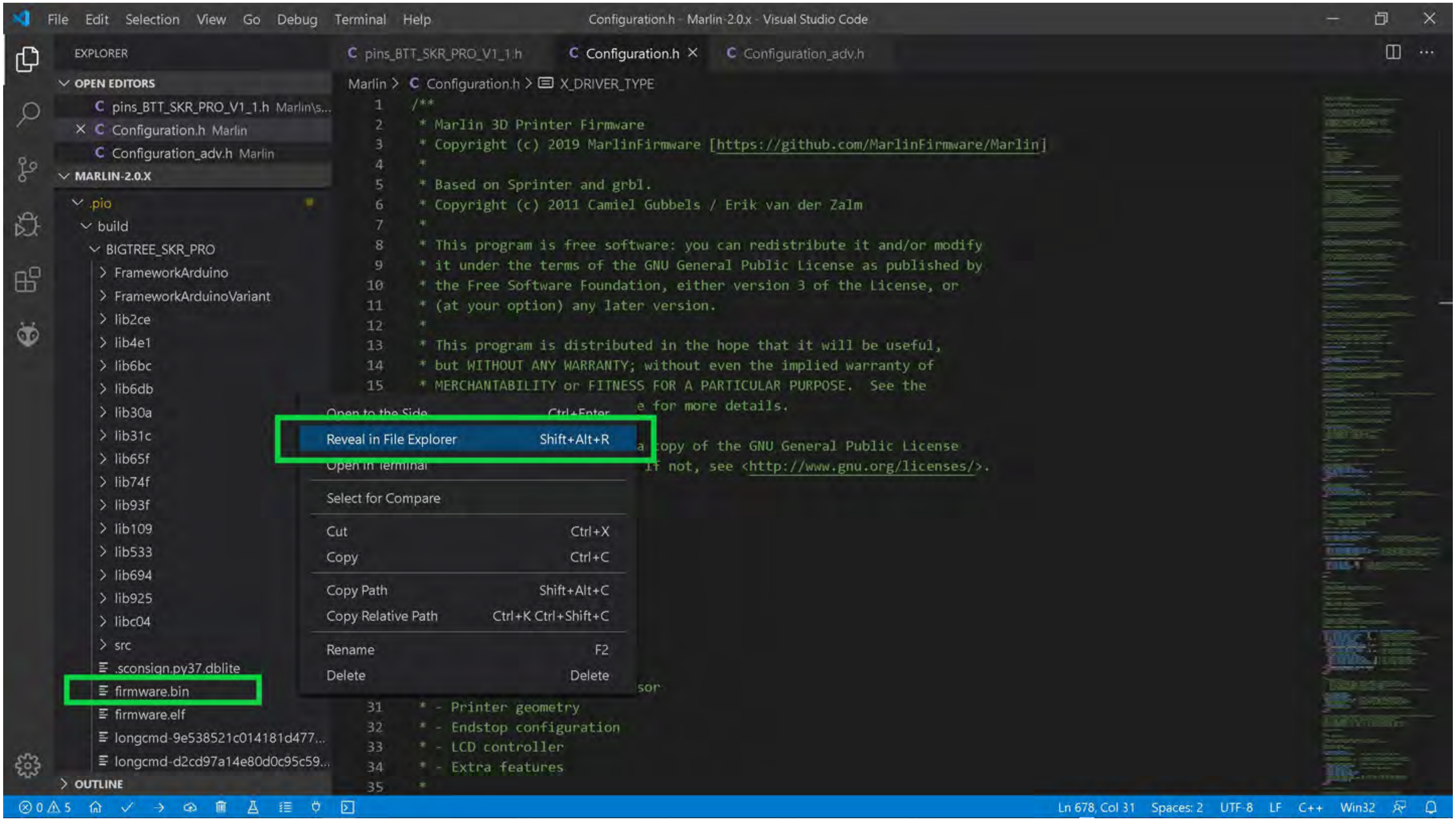
- The end of Marlin setup for BIQU TMC2130 drivers in stand-alone mode. Now, compile your code by clicking on the check mark, as seen in the **YELLOW** box below. Once the firmware has compiled successfully you will see the following messages, as shown in the **GREEN** boxes below.



- For the SKR PRO V1.1 board you must save the "firmware.bin" file to the micro SD card then place the micro SD card into the micro SD card reader of the SKR PRO V1.1 board and turn on the power to your printer for the NEW firmware to be automatically applied to the SKR PRO MCU. To locate the "firmware.bin" file, and learn how to transfer it to your micro SD card go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2130 Drivers in Stand-alone Mode

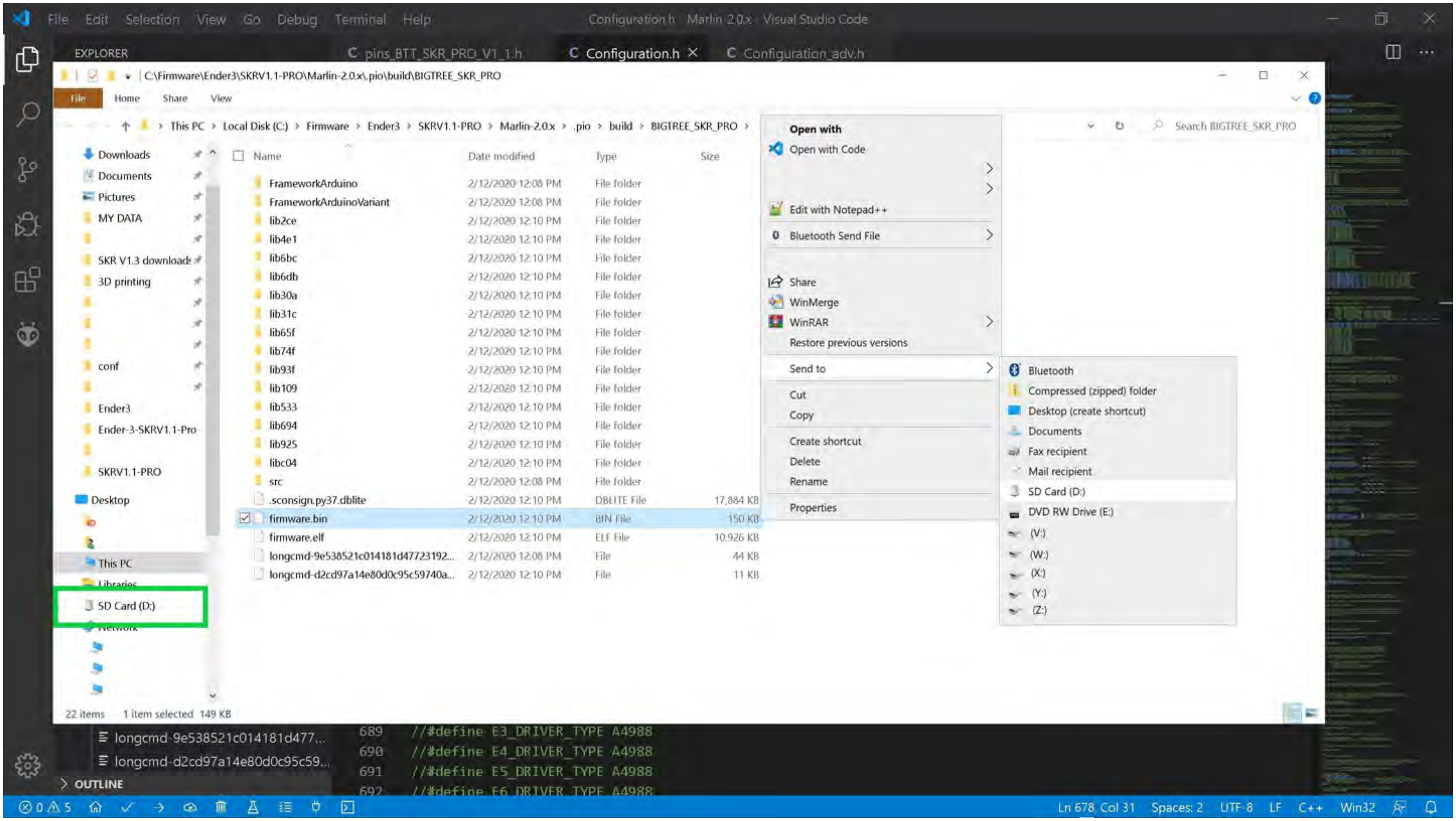
- The "firmware.bin" file can be found by using the left side of the VS code window and going to folder Marlin-2.0.X/.pio/build/ BIGTREE_SKR_PRO/ and right clicking on the "firmware.bin" file name to bring up the content menu. Select "Reveal in File Explorer" to have your Window's machine open a file explorer window.



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2130 Drivers in Stand-alone Mode

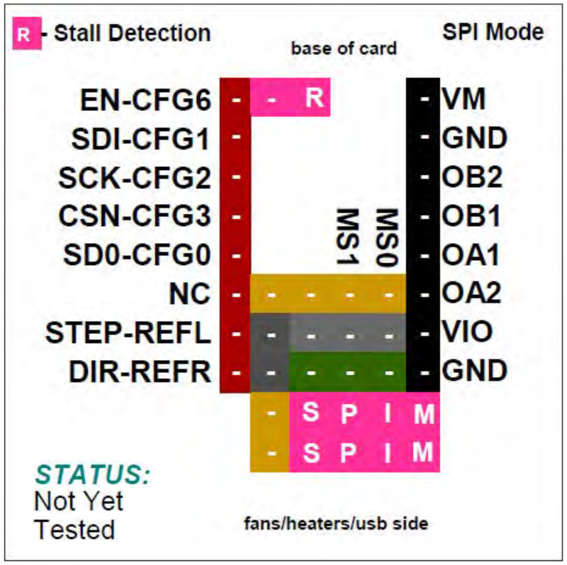
- Ensure your micro SD card, which came with your SKR PRO V1.1 board, with your SKR PRO V1.1 board is in a micro-to-SD-card adapter and plugged into your Window's SD card reader.
    - From the file explorer window right click on the "firmware.bin" file name and select "Send to" from the content menu. For my computer "D:" is my SD card reader. For your computer it may be a different drive letter. The "firmware.bin" file will be written to the SD card and renamed to "FIRMWARE.CUR".



- Right click on the SD card reader, as seen in the **GREEN** box above, and select "Eject" from the context menu. Take the micro SD card out of the adapter and place it into the micro SD card reader of the SKR PRO V1.1 board. Turn on the power to the printer (i.e. the SKR PRO V1.1 board) and the "FIRMWARE.CUR" file will be uploaded to the board. After the upload the firmware file will be renamed to "firmware.bin" and stored on the micro SD card.

BIQU TMC2130

SPI Mode



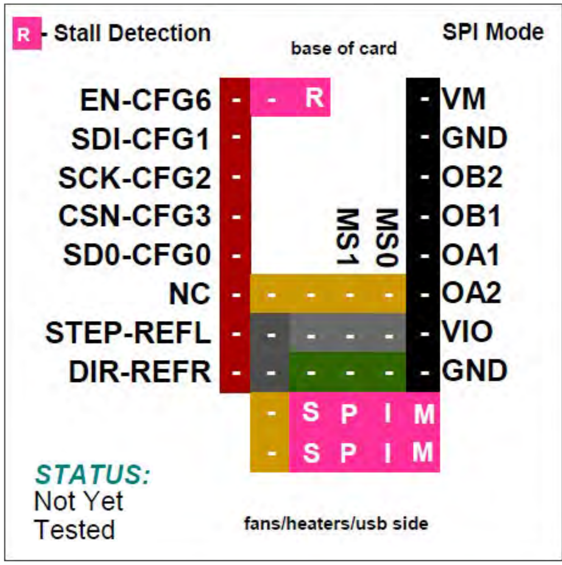
Note: You can use 50% to 90% of the calculated IRMS (I_{MAX}/1.414) when tuning ("X_CURRENT", "Y_CURRENT", etc.) the stepper motor driver in the firmware.

See the next page for further information.

Driver Chip	Steps are set inside of your Firmware	
BIQU® TMC2130 SPI Mode Maximum 256 Subdivision 46V DC 2.5A (peak)		
Driving Current Calculation Formula R_S(Typical Sense Resistor)= 0.11Ω	$I_{MAX}=V_{ref}$ See Appendix B #2. Use 50% to 90% as shown below: $I_{MAX}=I_{MAX} * 0.90$	$V_{ref}=I_{MAX}$ See Appendix B #2. Use 50% to 90% as shown below: $V_{ref}=V_{ref} * 0.90$

BIQU TMC2130

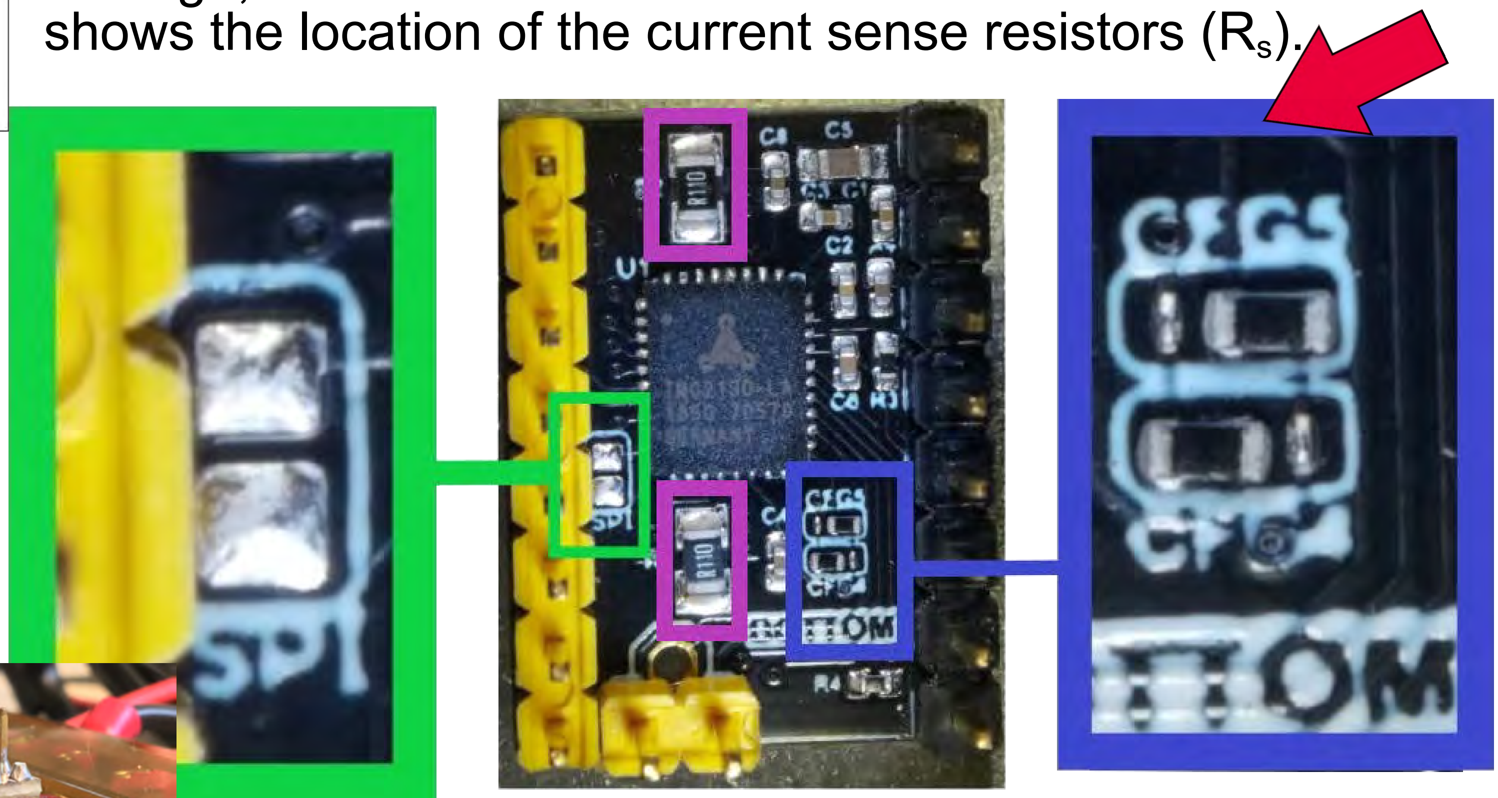
SPI Mode

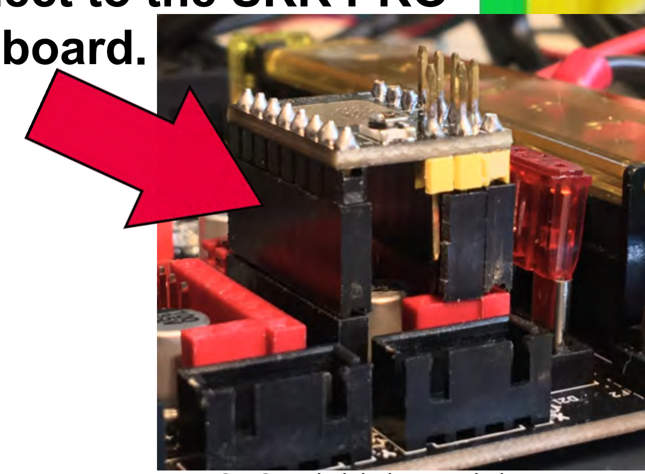


To place the BIQU TMC2130 into SPI Mode:

The SPI Jumper must have a gap between the two SPI pads, on the bottom of the driver board, as shown in the **GREEN** box below. Ensure that at CFG4 location and CFG5 location the correct two pads are soldered together to form a bridge, as shown in the **BLUE** box. The **PURPLE** box shows the location of the current sense resistors (R_s).

Note: **When the stall-guard function is not used, the stall-guard pin ("R") of the TMC2130 must be removed (desoldered) or use long pin header risers so that the "R" pin does not connect to the SKR PRO V1.1 board.**





NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

SPI Mode

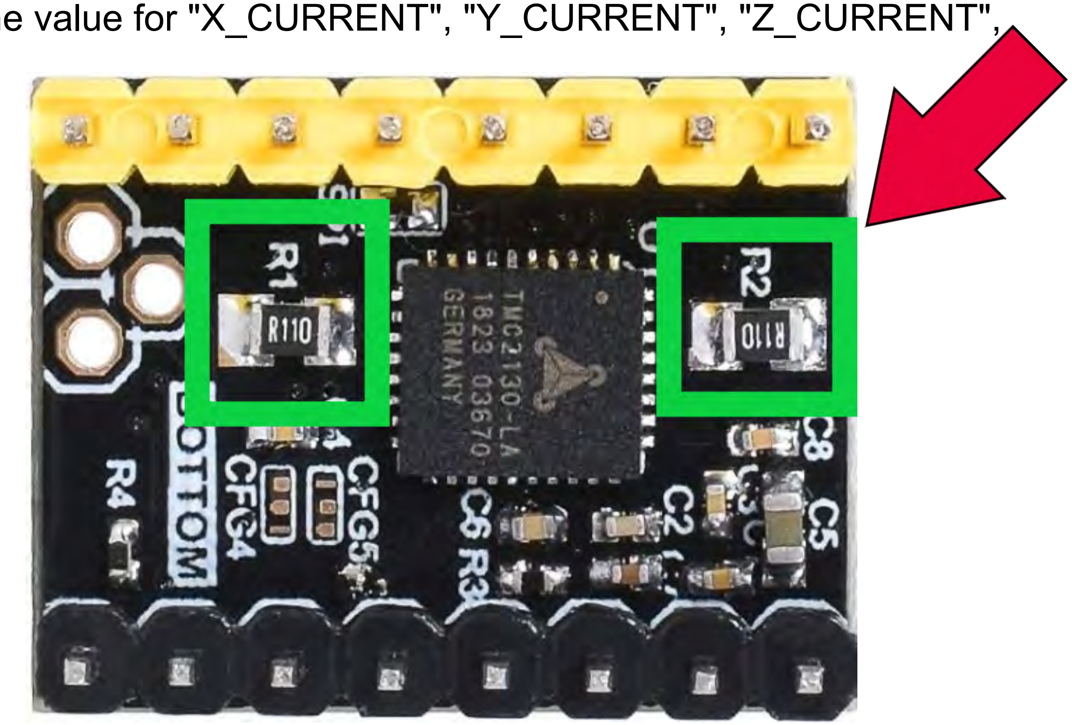
BIQU TMC2130

SPI Mode

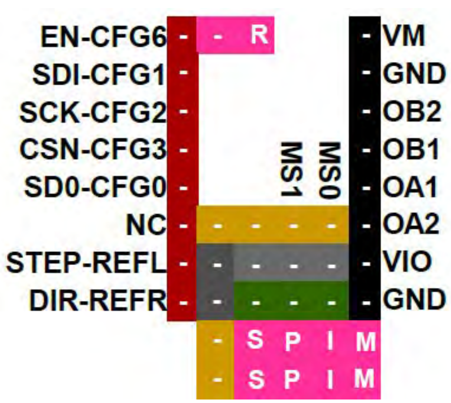
R - Stall Detection

Note: The location of the current sense resistors are shown in **GREEN**. Use the current sense resistors' value in the Marlin Firmware ("X_RSENSE", "Y_RSENSE", "Z_RSENSE" and/or "E0_RSENSE") so that the appropriate current limit can be sent to the driver board. If you do not want to use V_{ref} as the value for "X_CURRENT", "Y_CURRENT", "Z_CURRENT" and/or "E0_CURRENT", you should use I_{RMS} instead. You find I_{RMS} by taking I_{MAX} and dividing it by 1.414 ($I_{RMS}=I_{MAX}/1.414$). You use 50% to 90% of the calculated I_{RMS} as the value for "X_CURRENT", "Y_CURRENT", "Z_CURRENT", and/or "E0_CURRENT".

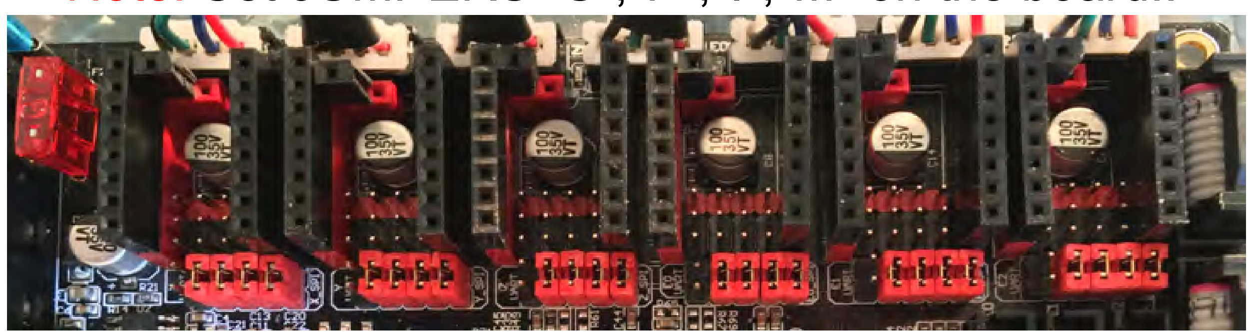
R_s = R050 is 0.05 Ohms
R_s = R062 is 0.062 Ohms
R_s = R068 is 0.068 Ohms
R_s = R075 is 0.075 Ohms
R_s = R100 is 0.1 Ohms
R_s = R110 is 0.11 Ohms
R_s = R150 is 0.15 Ohms
R_s = R200 is 0.2 Ohms
R_s = R220 is 0.22 Ohms



SPI



Note: Set JUMPERS "S","P","I","M" on the board!!

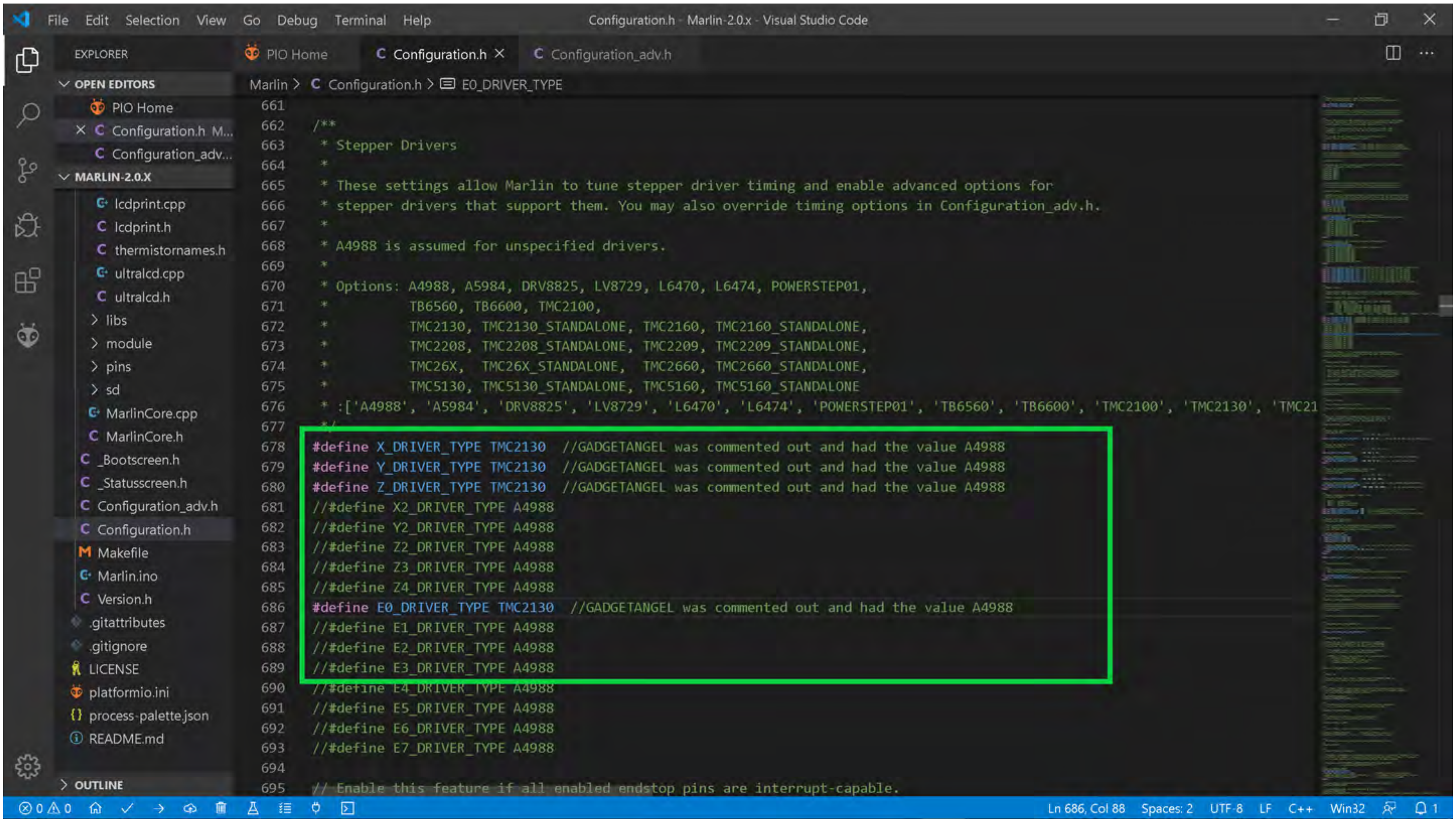


NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

The (latest release of) Marlin Setup for BIQU TMC2130 Drivers in SPI Mode

NOTE: **Go to Appendix C,** and then come back here for the changes to Marlin for BIQU TMC2130 stepper motor drivers in SPI. mode.

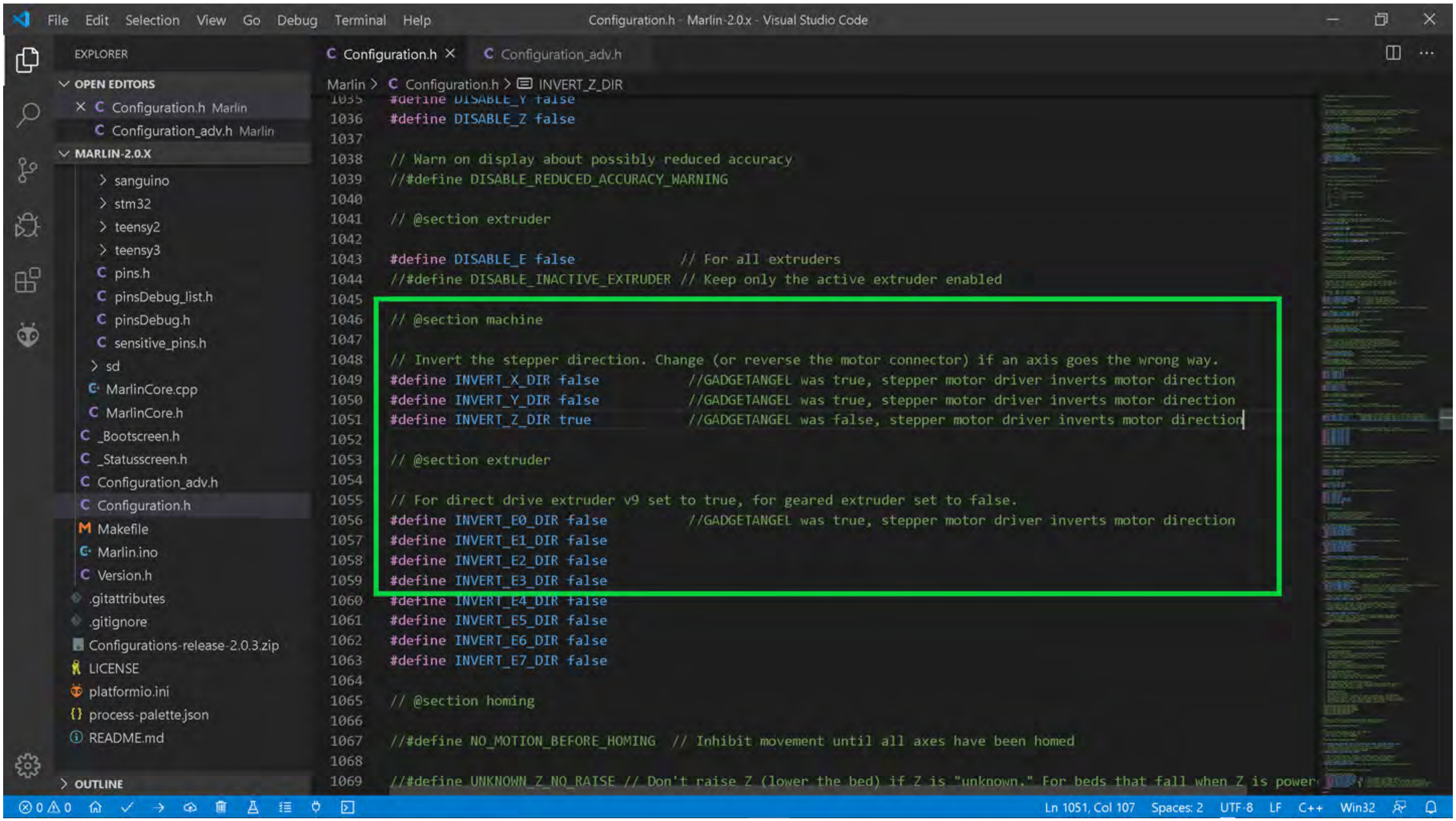
- Change the stepper motor drivers so that Marlin knows you are using BIQU TMC2130 drivers in SPI mode. Change one line for each axis and one for each extruder you will be using. See the picture below for an example of how to use TMC2130 drivers in SPI mode. When two "//" appear at the beginning of a line that means that line is commented out. To un-comment a line just remove the leading two forward slashes ("//").



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2130 Drivers in SPI Mode

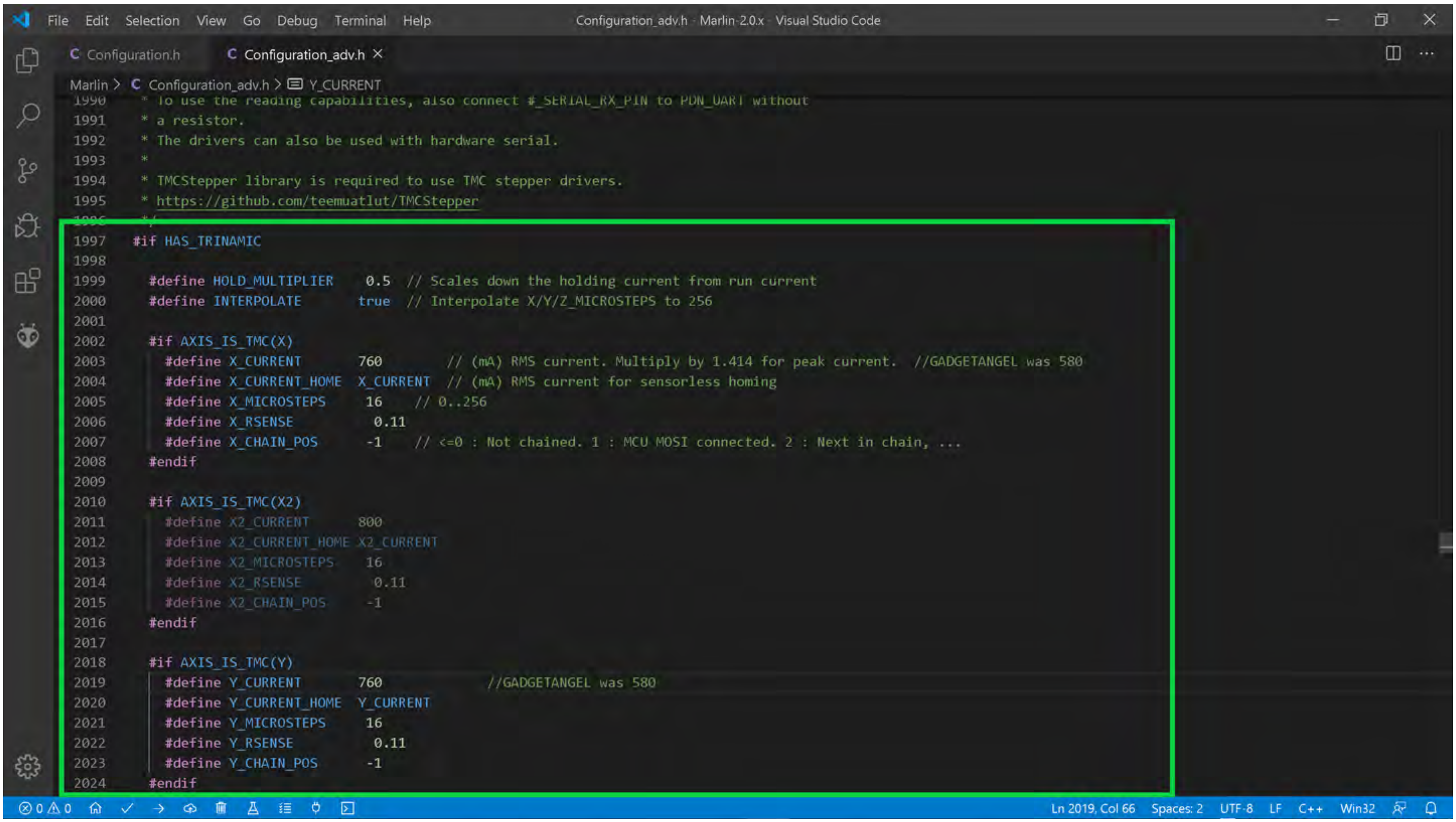
- Since the A4988 driver is what my Ender 3 used, but, now I want to use TMC2130 drivers, I must invert the stepper motor direction because the TMC2130 driver will turn the motors in the opposite direction than the A4988 driver's motor direction. So if the axis' setting you will be using the TMC2130 driver on was "true" change it to "false", as shown in the **GREEN** box below. If the setting was "false", now set it to "true", as shown in the **GREEN** box below



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2130 Drivers in SPI Mode

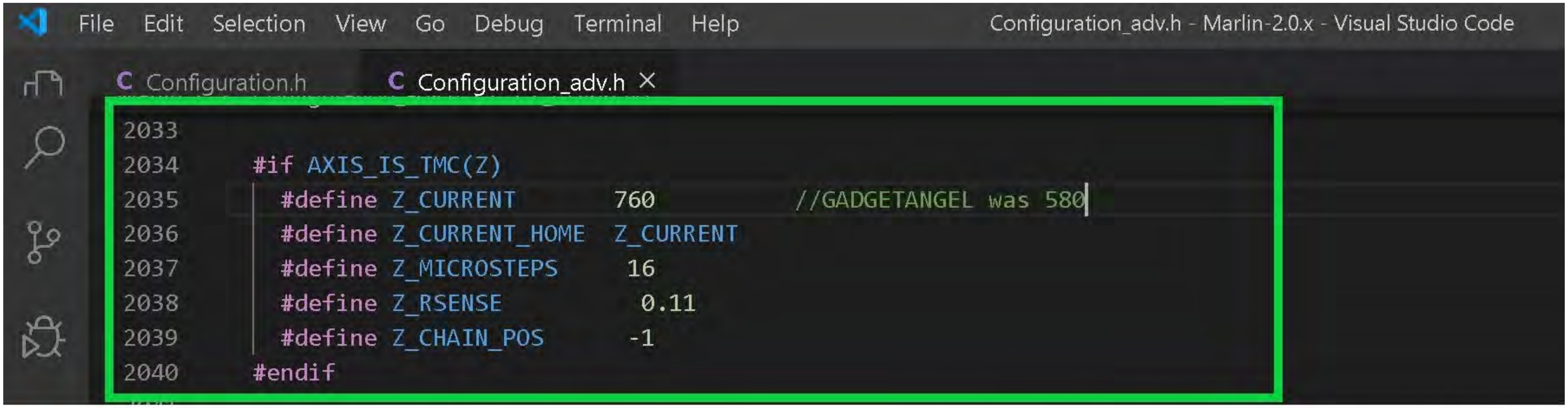
- Next you want to set your V_{ref} in the Marlin firmware for each axis that has the TMC2130 driver, as seen in the **GREEN** box below. I changed the "X_CURRENT" to be the calculated V_{ref} for my X-Axis, which is 760mV for an Ender 3. I changed the "Y_CURRENT" to be the calculated V_{ref} for my Y-Axis, which is 760mV on the Ender 3.
- Ensure "X_RSENSE" is set to 0.11. Ensure "Y_RSENSE" is set to 0.11.
- If you do not want to use V_{ref} as the value for "X_CURRENT" and/or "Y_CURRENT", you should use I_{RMS} instead. You find I_{RMS} by taking I_{MAX} and dividing it by 1.414 ($I_{RMS}=I_{MAX}/1.414$). You use 50% to 90% of the calculated I_{RMS} as the value for "X_CURRENT" and/or "Y_CURRENT".

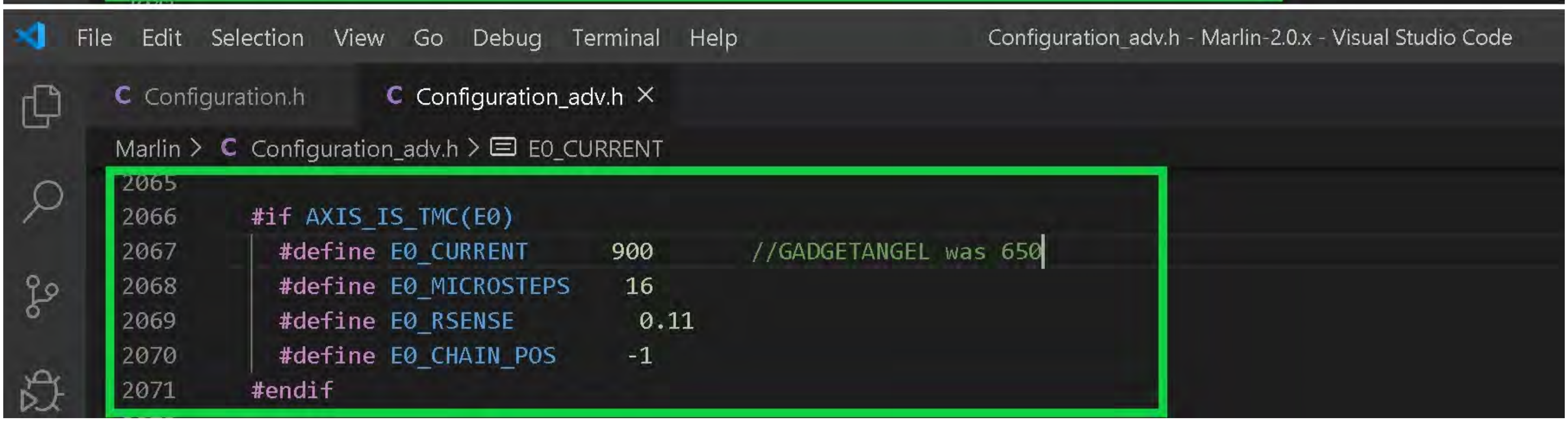


- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2130 Drivers in SPI Mode

- Now, I am setting the V_{ref} for Z-Axis and the extruder, as seen in the **GREEN** boxes below. I changed the "Z_CURRENT" to be the calculated V_{ref} for my Z-Axis, which is 760mV for an Ender 3. I changed the "E0_CURRENT" to be the calculated V_{ref} for my Extruder, which is 900mV on the Ender 3.
- Ensure "Z_RSENSE" is set to 0.11. Ensure "E0_RSENSE" is set to 0.11.
- If you do not want to use V_{ref} as the value for "Z_CURRENT" and/or "E0_CURRENT", you should use I_{RMS} instead. You find I_{RMS} by taking I_{MAX} and dividing it by 1.414 ($I_{RMS}=I_{MAX}/1.414$). You use 50% to 90% of the calculated I_{RMS} as the value for "Z_CURRENT" and/or "E0_CURRENT".

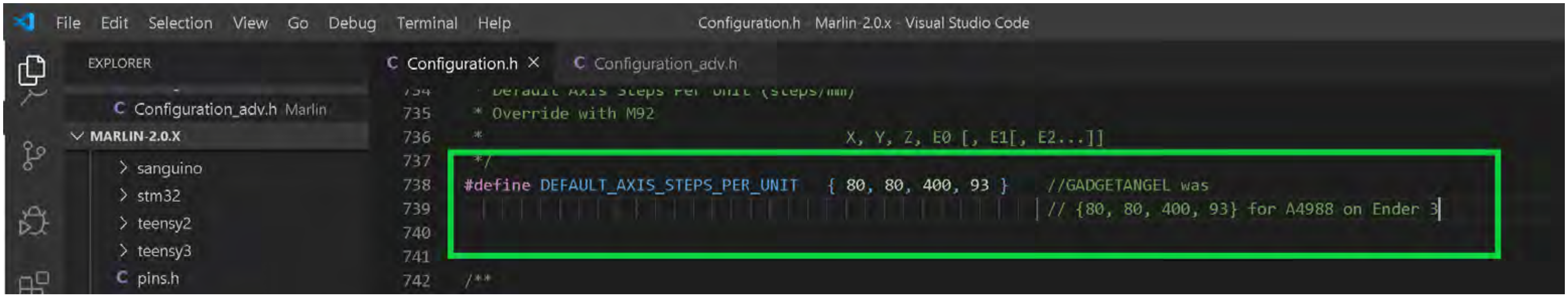




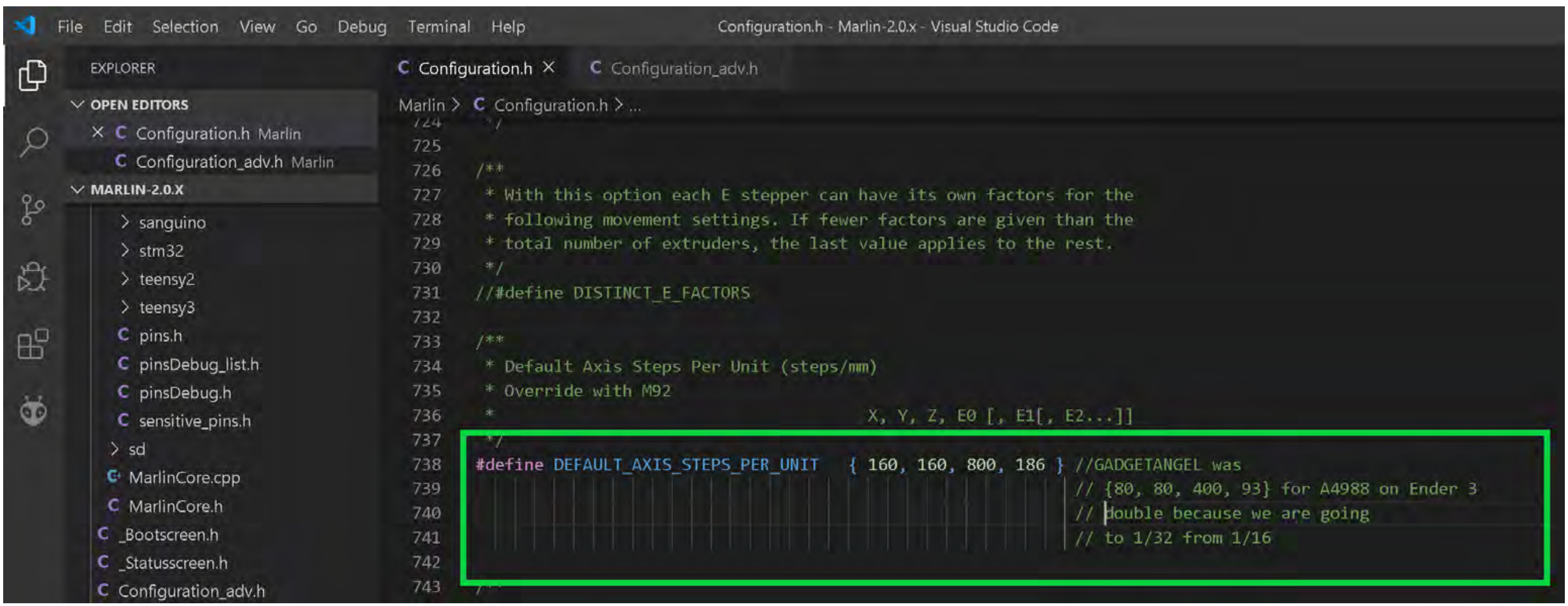
- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2130 Drivers in SPI Mode

- If you changed the "MICROSTEPS" for any of the axes then you will need to update "DEFAULT_AXIS_STEPS_PER_UNIT" to reflect your changes



- FOR EAMPLE if you wanted to use 1/32 stepping instead of the default 1/16, you would be **doubling** your STEPS. Therefore, we must adjust our "DEFAULT_AXIS_STEPS_PER_UNIT" anytime our STEPS are NOT 1/16. So change "DEFAULT_AXIS_STEPS_PER_UNIT" to {160, 160, 800, 186}, as seen in the **GREEN** box below.

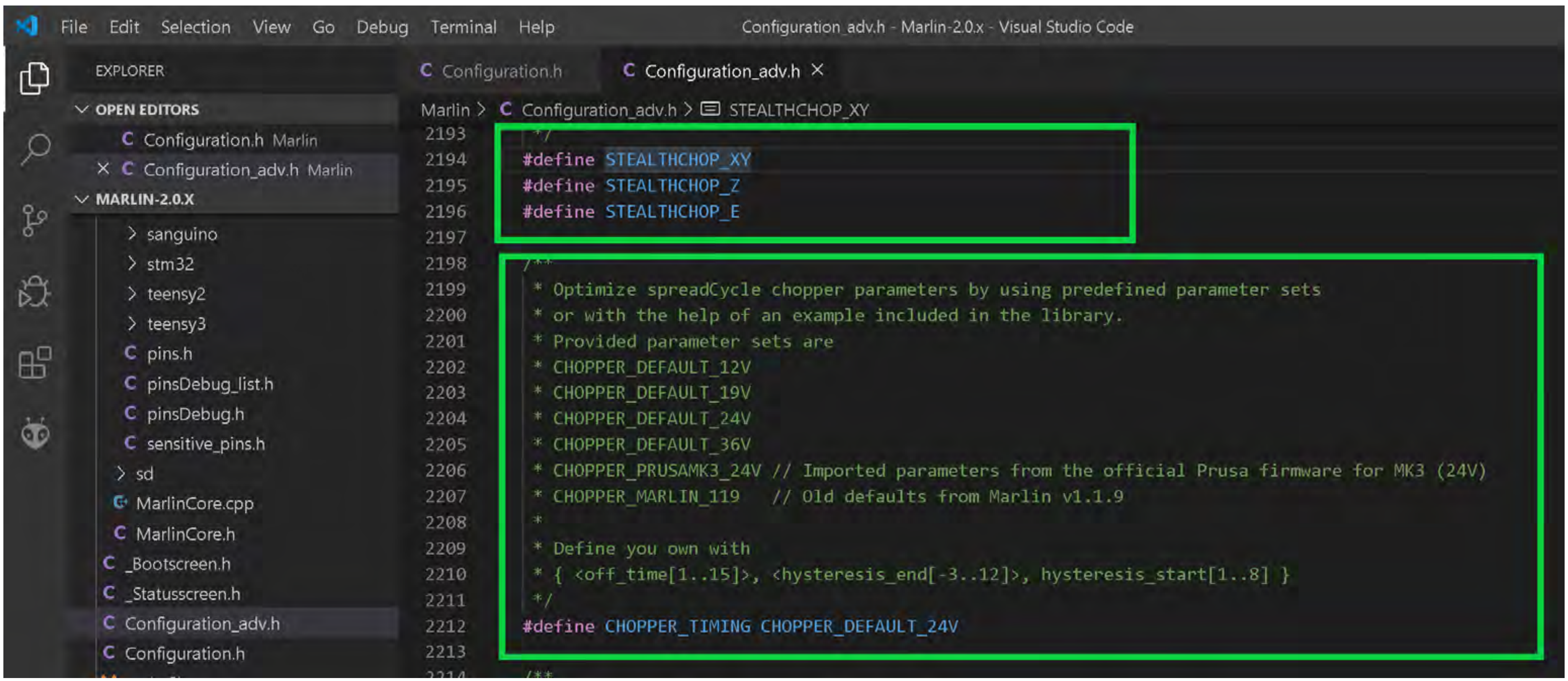


- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2130 Drivers in SPI Mode

- We need to uncomment out the "TMC_USE_SW_SPI" because the SKR PRO V1.1 pins file depends on this variable to define its SPI pins

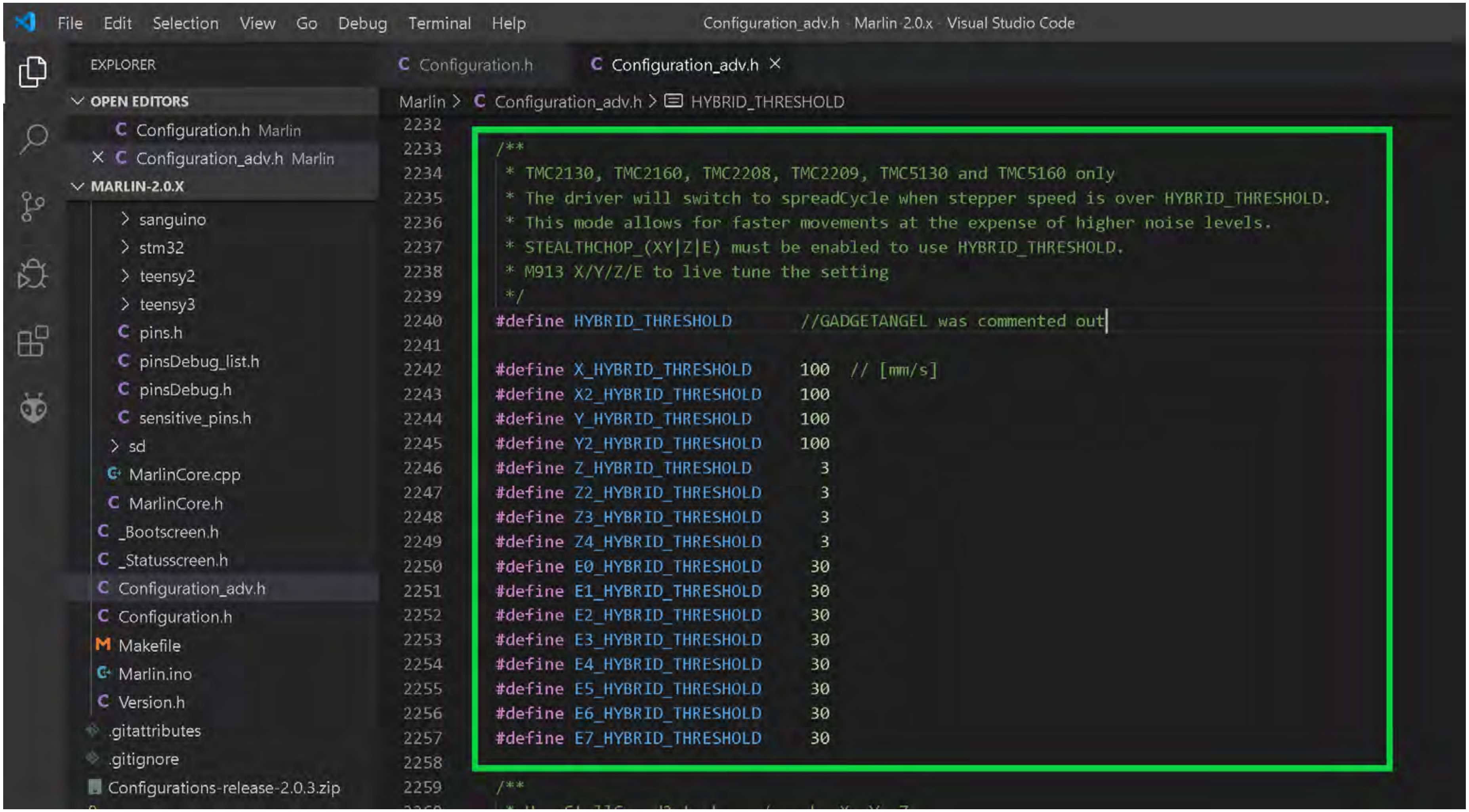
- By default stealthChop is enabled in the Marlin firmware. If you want spreadCycle ONLY then comment out the appropriate lines. I want stealthChop enabled so I want to make sure the lines are not commented out {"STEALTHCHOP_XY", "STEALTHCHOP_Z" and "STEALTHCHOP_E"}. You also want to check to see if the proper "CHOPPER_TIMING" is set for your printer. An Ender 3 is a 24VDC printer, my "CHOPPER_TIMING" is correct.



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2130 Drivers in SPI Mode

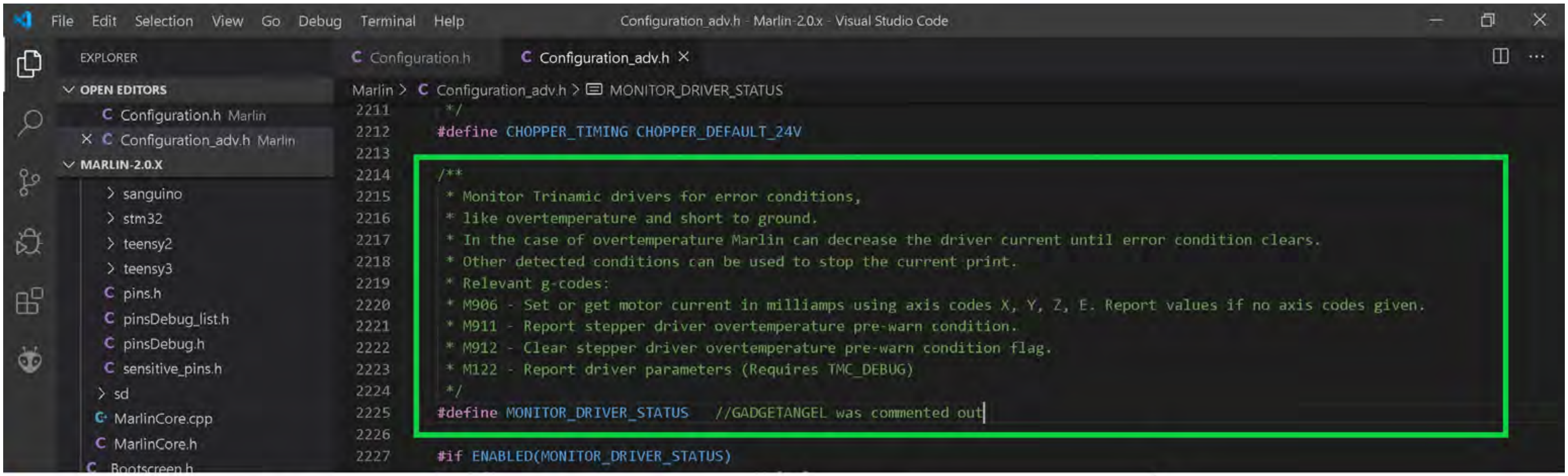
- Now you either enable "HYBRID_THRESHOLD" or disable it. By default it is disabled. "HYBRID_THERSHOLD" allows the printer to change between stealthChop and spreadCycle dynamically depending on the print speed. I want "HYBRID_THRESHOLD" enabled so I need to remove the two leading "//", which uncomments the line in the Marlin firmware.

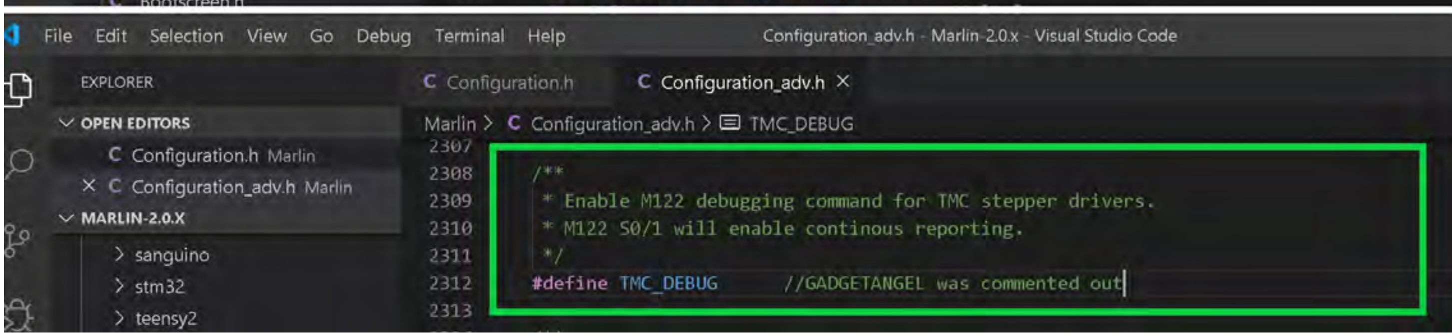


- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2130 Drivers in SPI Mode

- Now I want to enable some statements that allow me access to debugging the TMC drivers. I will uncomment "MONITOR_DRIVER_STATUS" and "TMC_DEBUG". "MONITOR_DRIVER_STATUS" will enable the following G-codes: M906, M911, and M912, "TMC_DEBUG" will enable the M122 G-code command. You can read about these from the comments in the firmware and in Marlin's documentation located on-line.

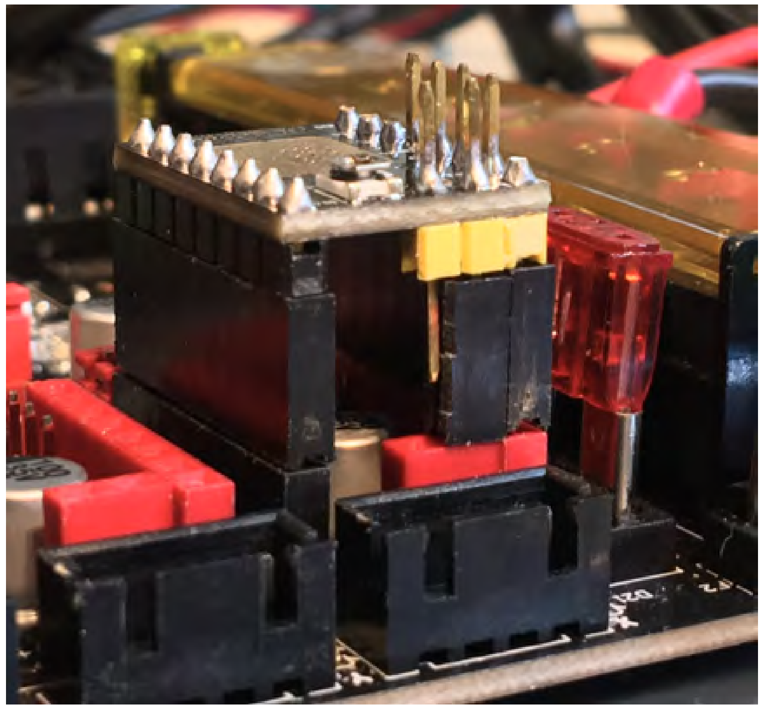




- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2130 Drivers in SPI Mode

- This next section covers sensor-less homing which is available for the TMC2130 in SPI mode. I want to enable it so I will be covering sensor-less homing for the X and Y axis only. I will not be using sensor-less homing on my Z axis on my Ender 3 printer. For sensor-less homing to work the DIAG1 pin on the TMC2130 driver has to be plugged into the SKR PRO V1.1 board. Since I am not using sensor-less homing on my Z axis I will need to ensure that my DIAG1 pin on the Z axis TMC2130 is NOT connected to the board. I plan to plug my Z axis TMC2130 as seen in the picture below

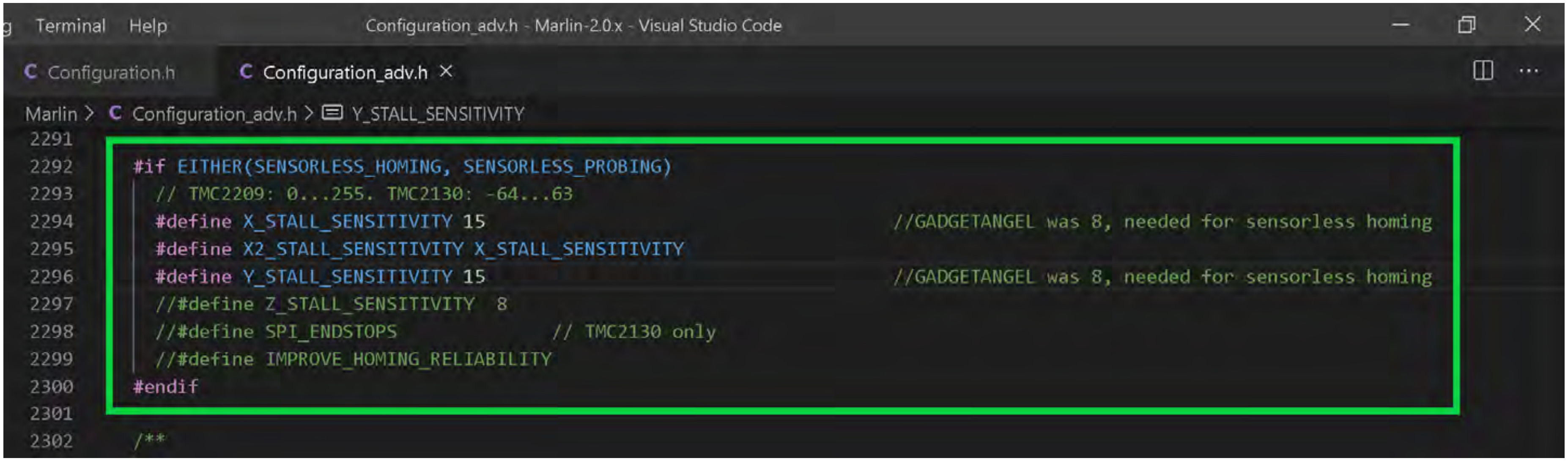


- Sensor-less homing is commented out by default. So I remove the two leading "//" to un-comment "SENSORLESS_HOMING"

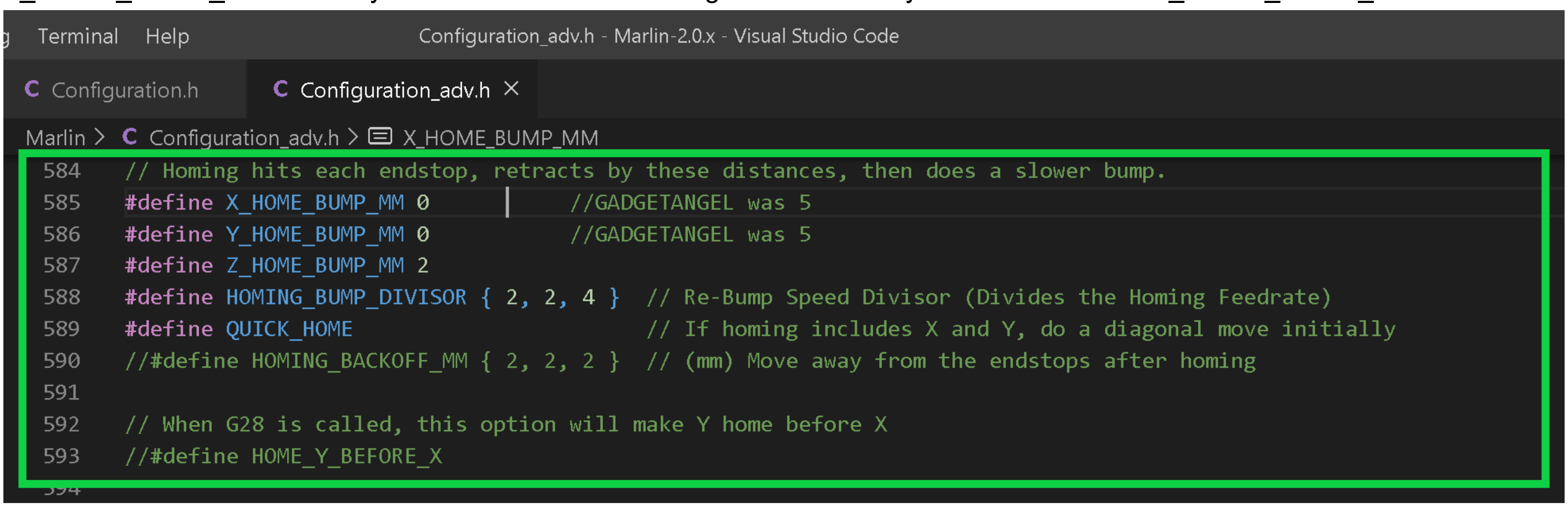
- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2130 Drivers in SPI Mode

- Next we set the "starting" stall sensitivity for sensor-less homing. I choose to make it 15. If the stall sensitivity is too high your motor will grind and not stop when it hits the end of travel on the axis. If the stall sensitivity is too low then the motor will barely move because it thinks it has hit the end of travel for the axis. Notice I only uncommented the "X_STALL_SENSITIVITY" and the "Y_STALL_SENSITIVITY". If you want sensor-less homing on the Z axis, then you will have to uncomment "Z_STALL_SENSITIVITY".



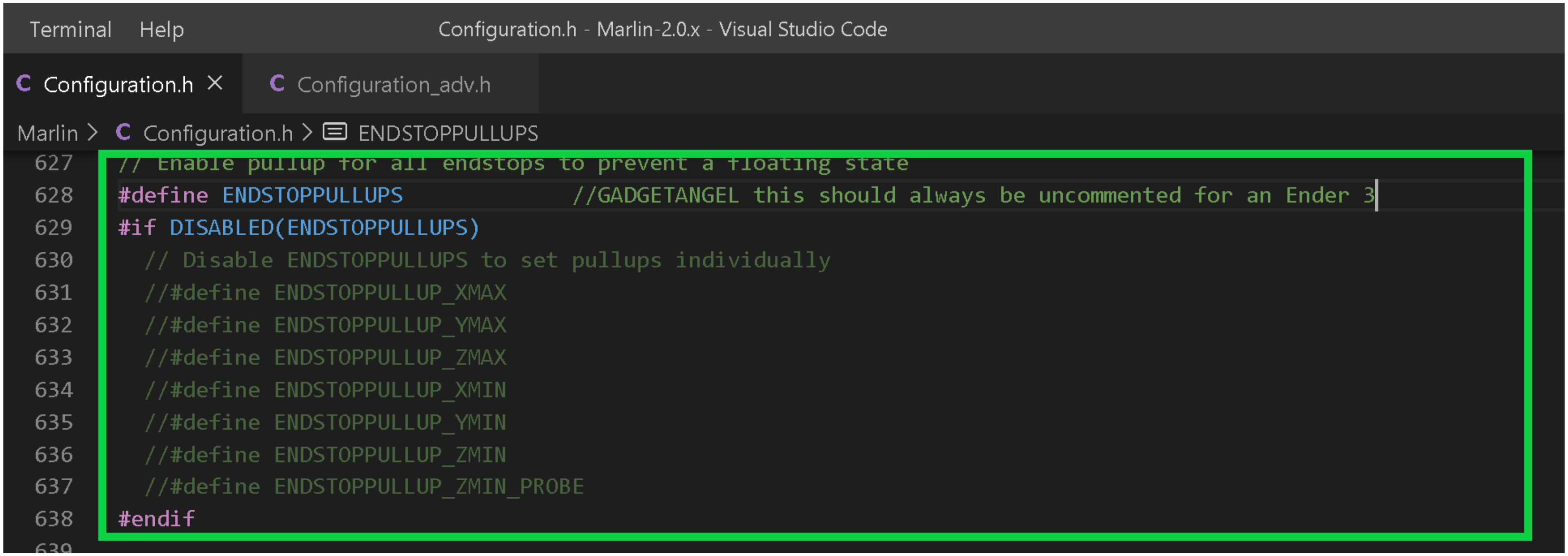
- We now have to set our home bump to 0 for each axis with sensor-less homing enabled. So I will set "X_HOME_BUMP_MM" to 0 and "Y_HOME_BUMP_MM" to 0. If you want sensor-less homing on Z axis then you will need to set "Z_HOME_BUMP_MM" to 0.



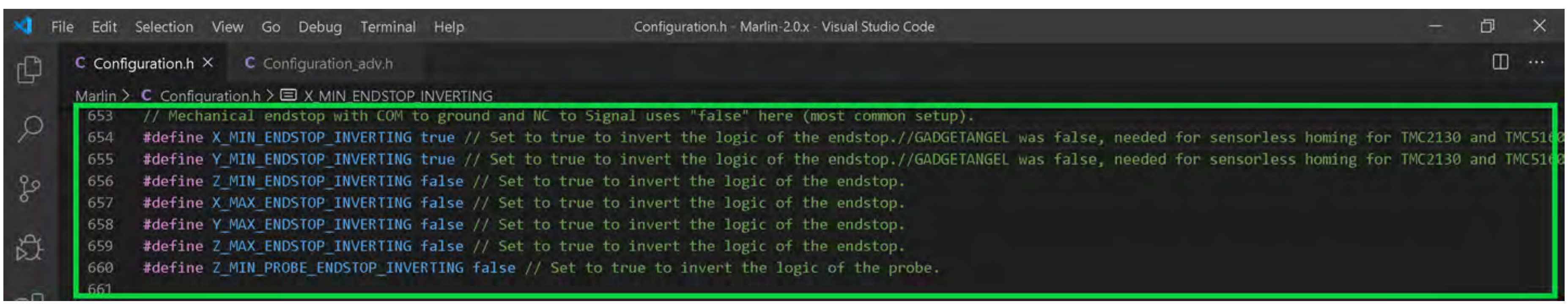
- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2130 Drivers in SPI Mode

- Let's check the firmware to ensure that "ENDSTOPPULLUPS" is enabled. It is by default.



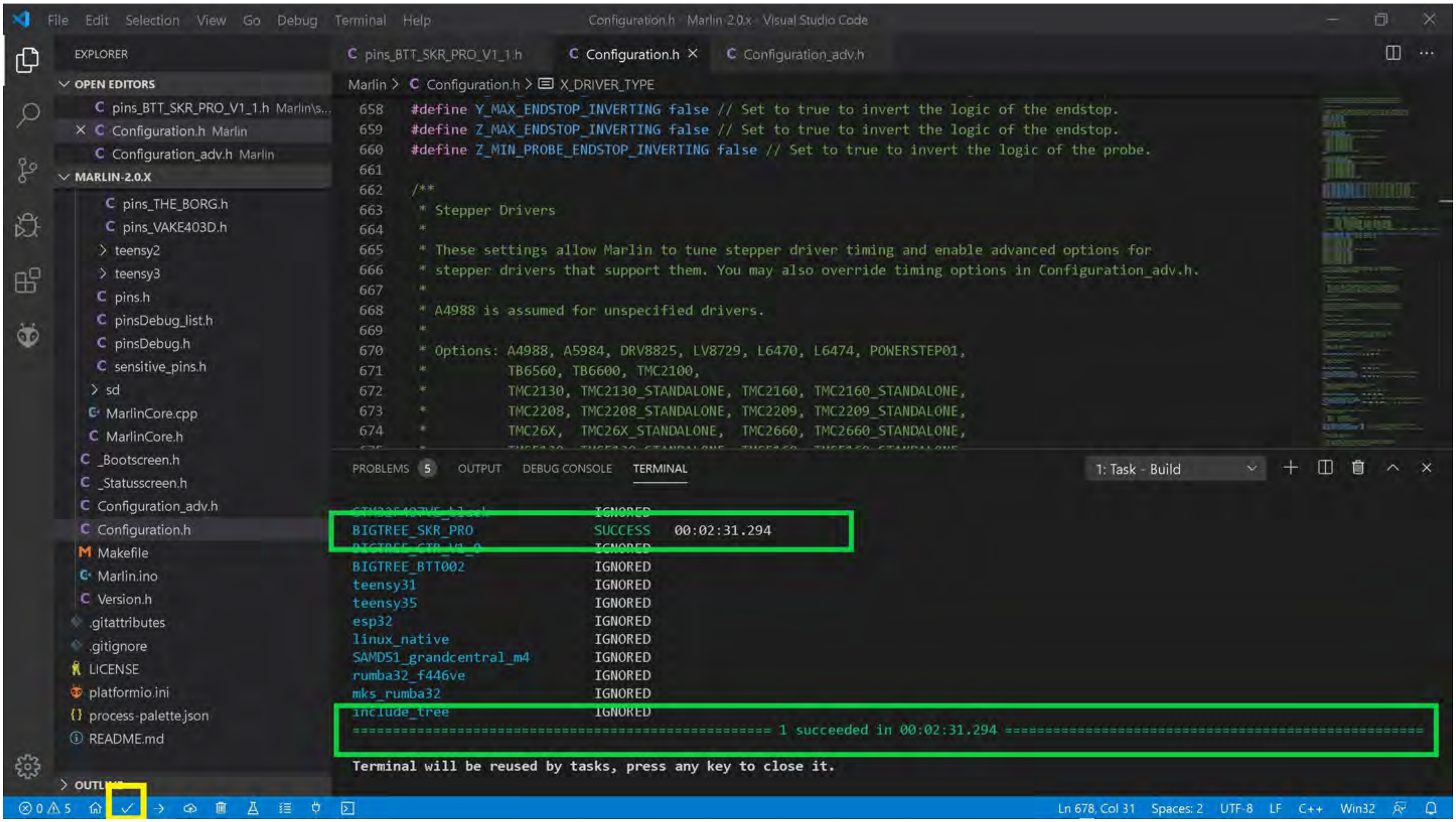
- Next to allow sensor-less homing to work (while using the BIQU TMC2130) we need to change our end stop logic. Therefore I set "X_MIN_ENDSTOP_INVERTING" to true and "Y_MIN_ENSTOP_INVERTING" to true. If you want sensor-less homing on the Z axis, you will need to set "Z_MIN_ENDSTOP_INVERTING" to true. But since I do not want sensor-less homing on the Z axis I will leave "Z_MIN_ENDSTOP_INVERTING" set to false.



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2130 Drivers in SPI Mode

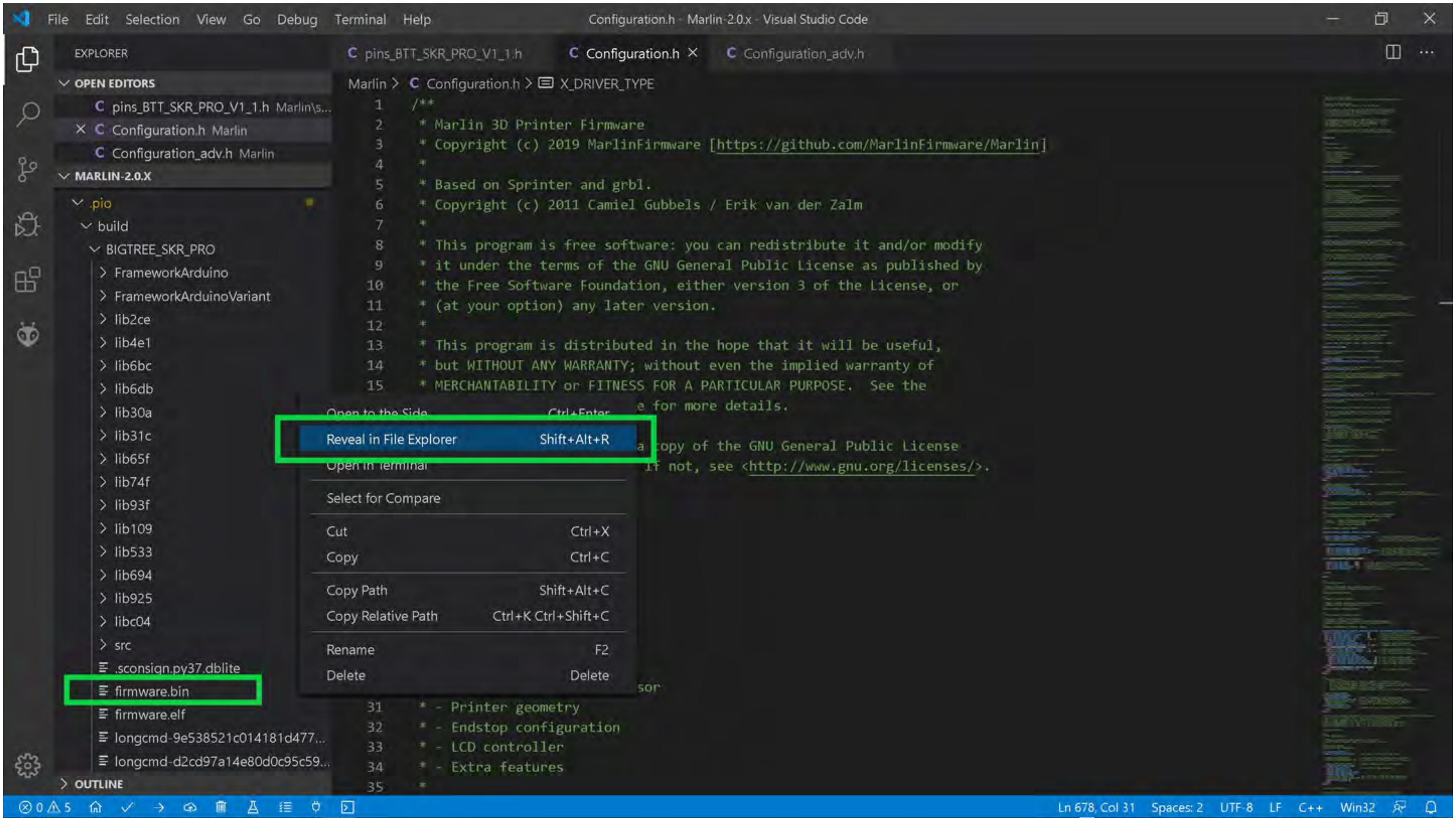
- The end of Marlin setup for BIQU TMC2130 drivers in SPI mode. Now, compile your code by clicking on the check mark, as seen in the **YELLOW** box below. Once the firmware has compiled successfully you will see the following messages, as shown in the **GREEN** boxes below.



- For the SKR PRO V1.1 board you must save the "firmware.bin" file to the micro SD card then place the micro SD card into the micro SD card reader of the SKR PRO V1.1 board and turn on the power to your printer for the NEW firmware to be automatically applied to the SKR PRO MCU. To locate the "firmware.bin" file, and learn how to transfer it to your micro SD card go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2130 Drivers in SPI Mode

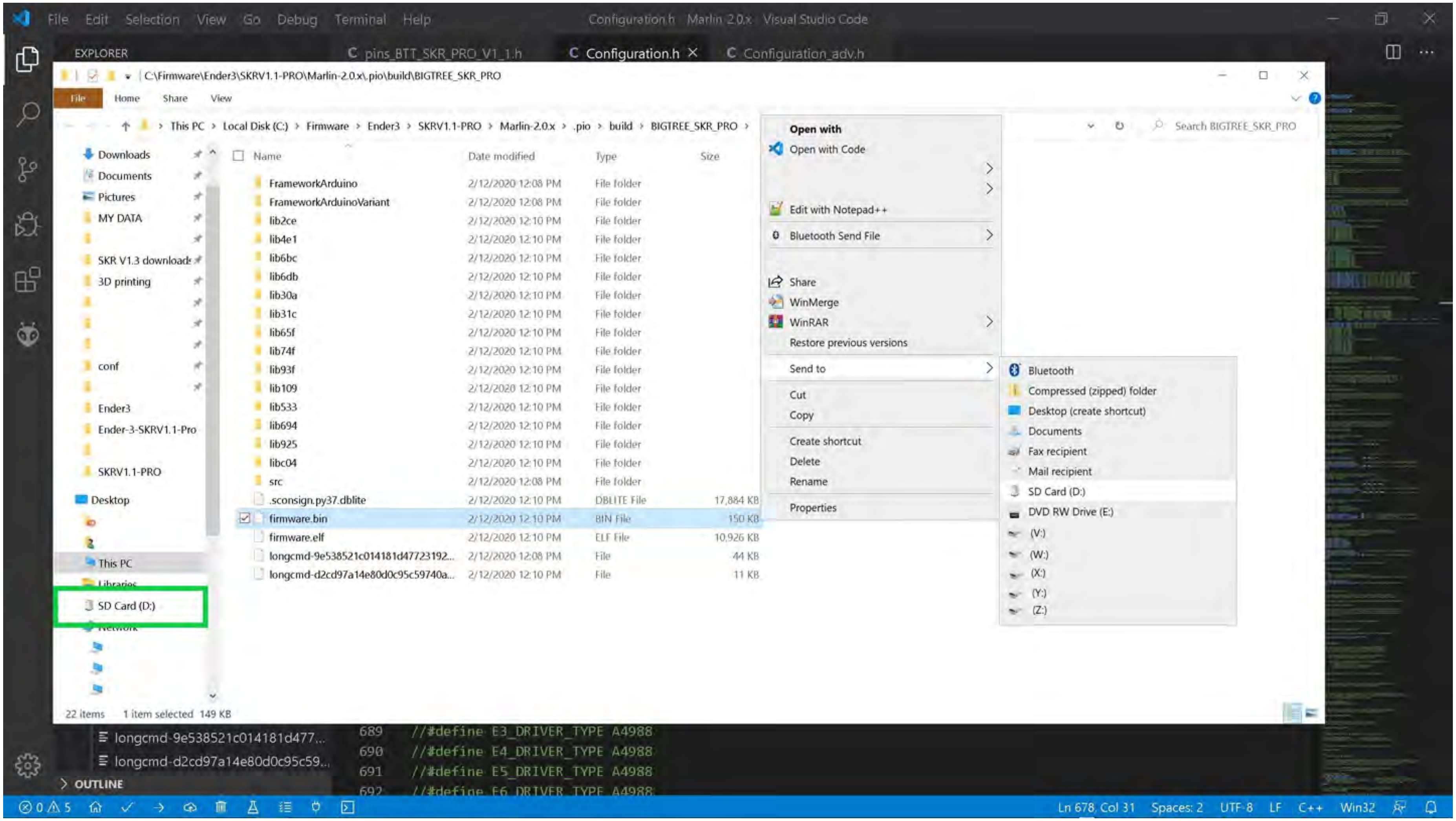
- The "firmware.bin" file can be found by using the left side of the VS code window and going to folder Marlin-2.0.X/.pio/build/ BIGTREE_SKR_PRO/ and right clicking on the "firmware.bin" file name to bring up the content menu. Select "Reveal in File Explorer" to have your Window's machine open a file explorer window.



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2130 Drivers in SPI Mode

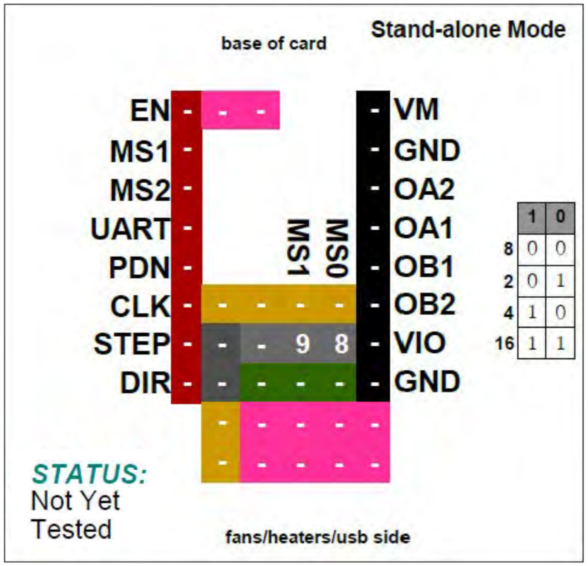
- Ensure your micro SD card, which came with your SKR PRO V1.1 board, with your SKR PRO V1.1 board is in a micro-to-SD-card adapter and plugged into your Window's SD card reader.
    - From the file explorer window right click on the "firmware.bin" file name and select "Send to" from the content menu. For my computer "D:" is my SD card reader. For your computer it may be a different drive letter. The "firmware.bin" file will be written to the SD card and renamed to "FIRMWARE.CUR".



- Right click on the SD card reader, as seen in the **GREEN** box above, and select "Eject" from the context menu. Take the micro SD card out of the adapter and place it into the micro SD card reader of the SKR PRO V1.1 board. Turn on the power to the printer (i.e. the SKR PRO V1.1 board) and the "FIRMWARE.CUR" file will be uploaded to the board. After the upload the firmware file will be renamed to "firmware.bin" and stored on the micro SD card.

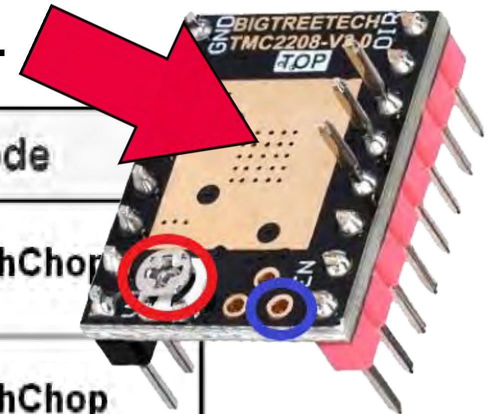
BIQU TMC2208 V3.0

Stand-alone Mode



Note: Use 90% of the calculated V_{ref} when tuning the stepper driver board.

NOTE: Use the potentiometer (POT) on the top of the board, as shown in **RED**; or use the board's "V_{ref} Test point" location, as shown in **BLUE**, to set your V_{ref}. See **Appendix A** for instructions on how to set the V_{ref} on a driver board.



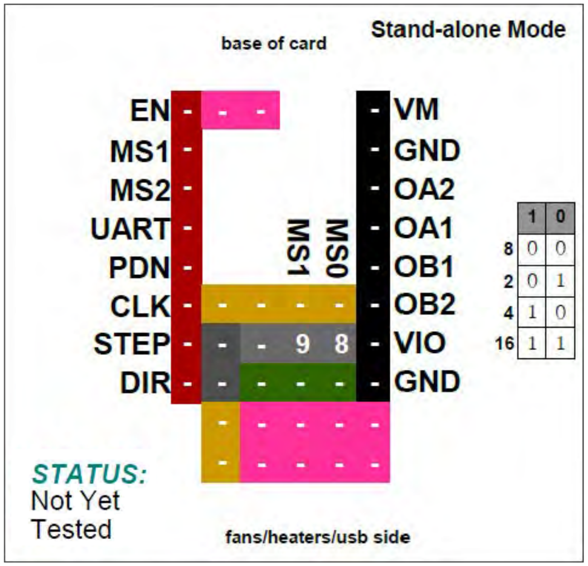
Driver Chip	MS1	MS0	Steps	Interpolation	Mode
BIQU® TMC2208 Stand Alone Mode Maximum 16 Subdivision 35V DC 2A (peak)	GND	GND	1 / 8	1 / 256	stealthChop
	GND	VIO	1 / 2	1 / 256	stealthChop
	VIO	GND	1 / 4	1 / 256	stealthChop
	VIO	VIO	1 / 16	1 / 256	stealthChop

Driving Current Calculation Formula		
R_S(Typical Sense Resistor)= 0.11Ω	$I_{MAX}=V_{ref}*0.9286$ See Appendix B #3. Use 50% to 90% as shown below: $I_{MAX}=(V_{ref}*0.9286)*0.90$	$V_{ref}=I_{MAX}*1.0769$ See Appendix B #3. Use 50% to 90% as shown below: $V_{ref}=(I_{MAX}*1.0769)*0.90$

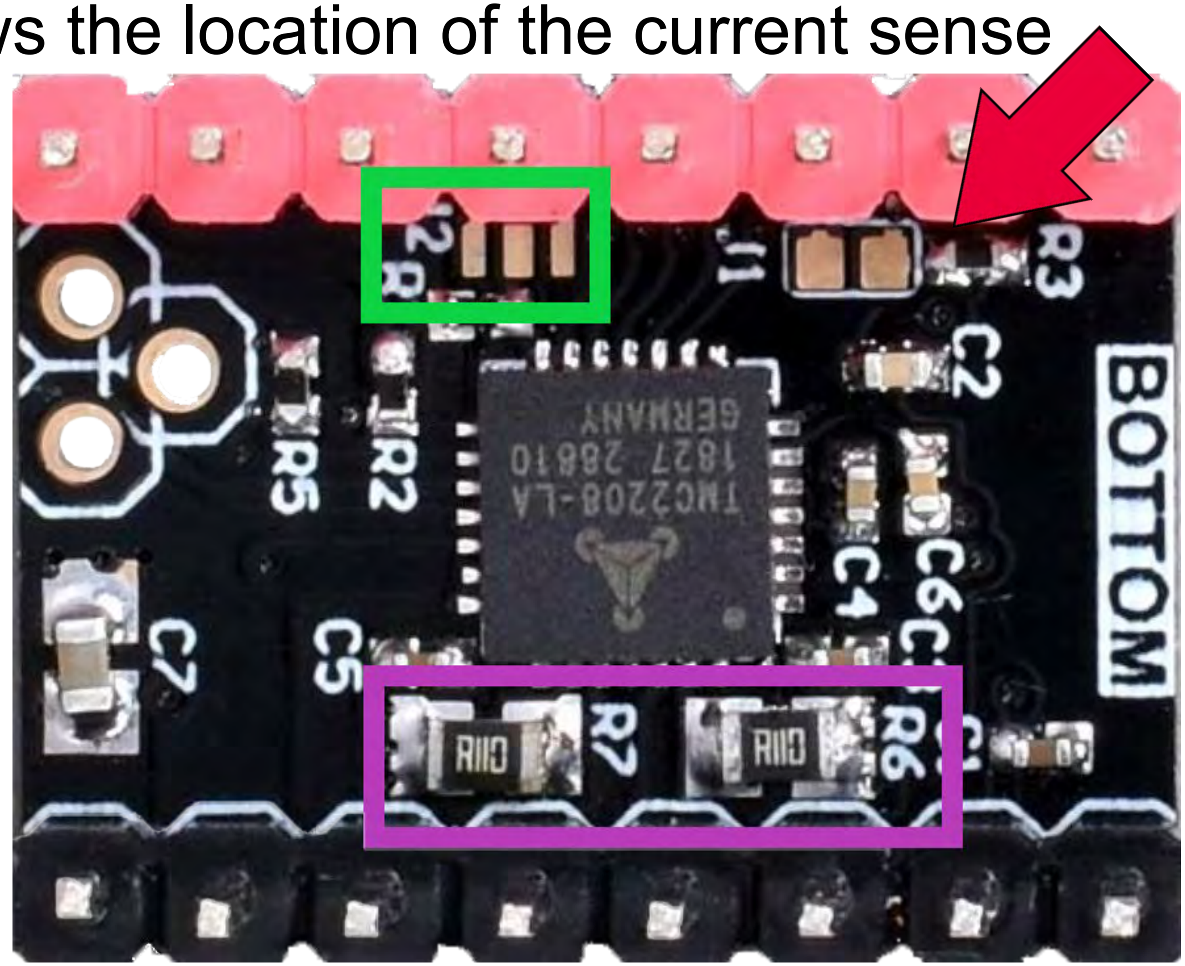
NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

BIQU TMC2208 V3.0

Stand-alone Mode



Note: The three pads (J2), on the bottom of the BIQU TMC2208 V3.0 driver boards, **MUST NOT** be connected. Again, a gap MUST be in place between all three J2 pads to obtain stand-alone mode for the TMC2208 V3.0, as seen in the **GREEN** box below. The **PURPLE** box shows the location of the current sense resistors (R_s).



Note: MOST BIQU TMC2208 V3.0 driver boards, when purchased for UART mode will have two adjacent J2 pads already soldered together (located on the bottom of the driver board).

NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

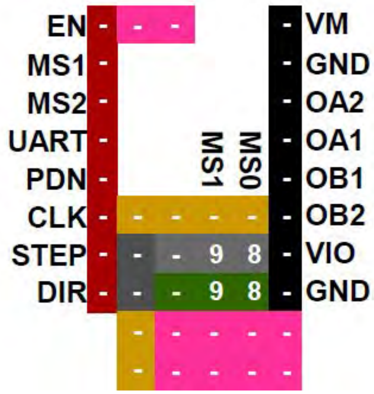
Stand-alone Mode

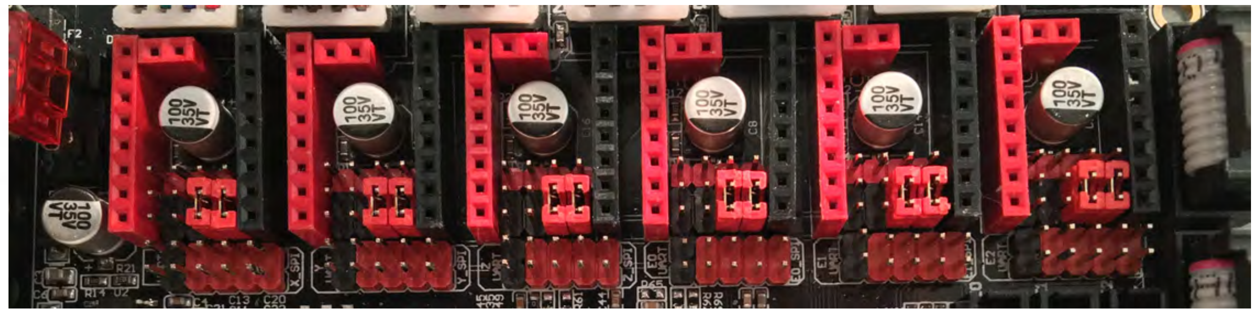
Stand-alone Mode

1 / 8

Interpolation: 1/256

StealthChop



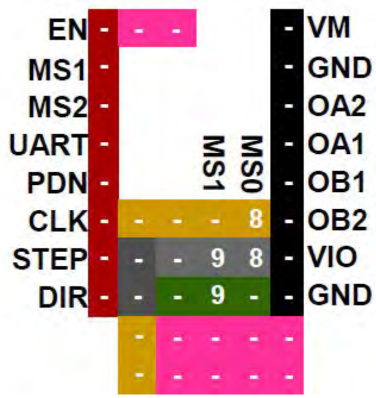


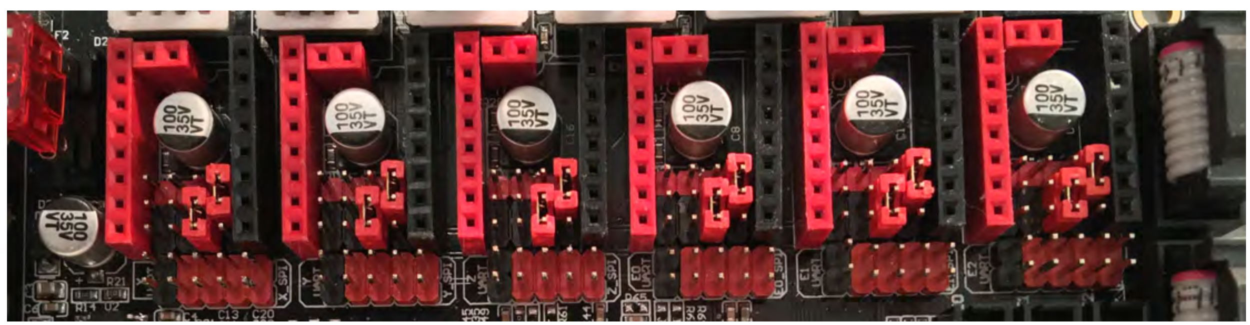
Stand-alone Mode

1 / 2

Interpolation: 1/256

StealthChop





NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

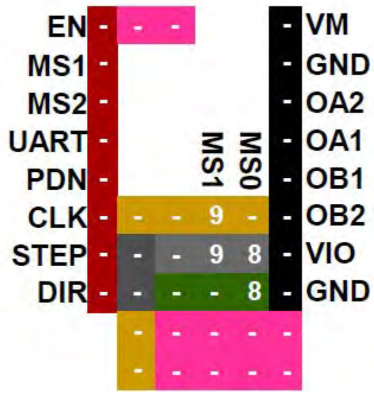
Stand-alone Mode

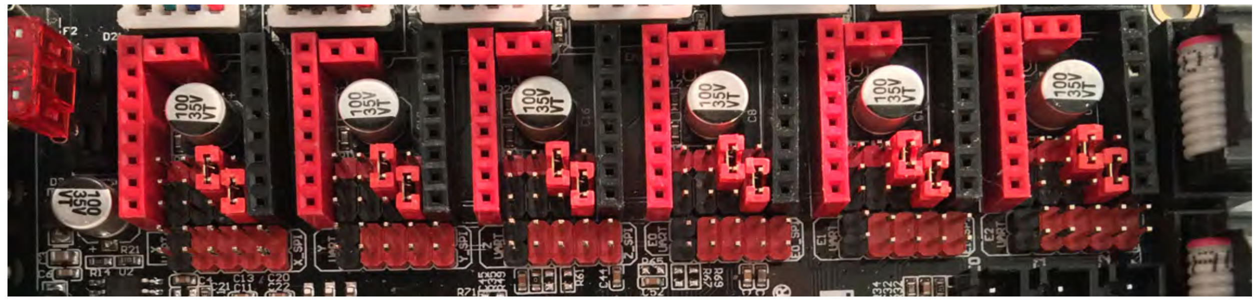
Stand-alone Mode

1 / 4

Interpolation: 1/256

StealthChop



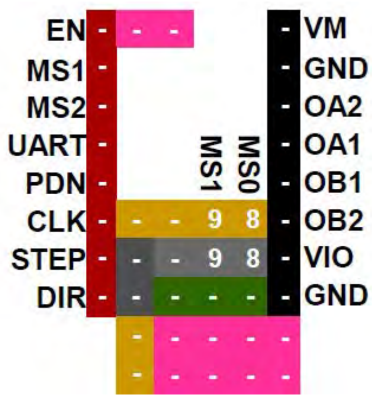


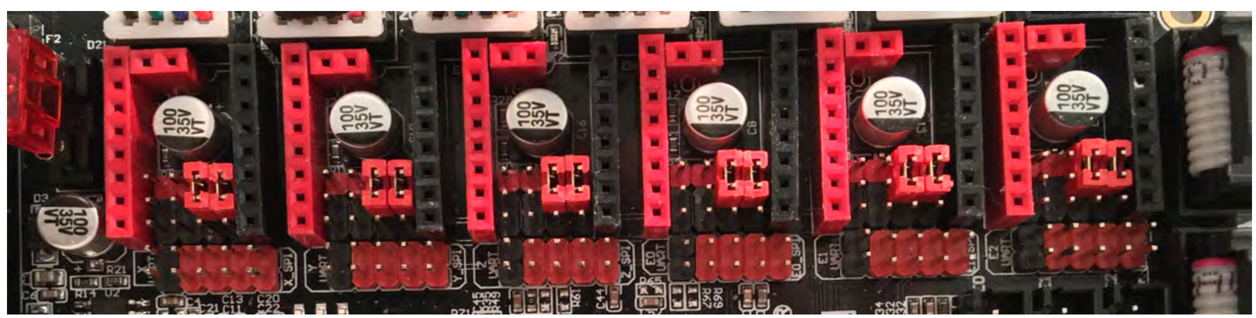
Stand-alone Mode

1 / 16

Interpolation: 1/256

StealthChop



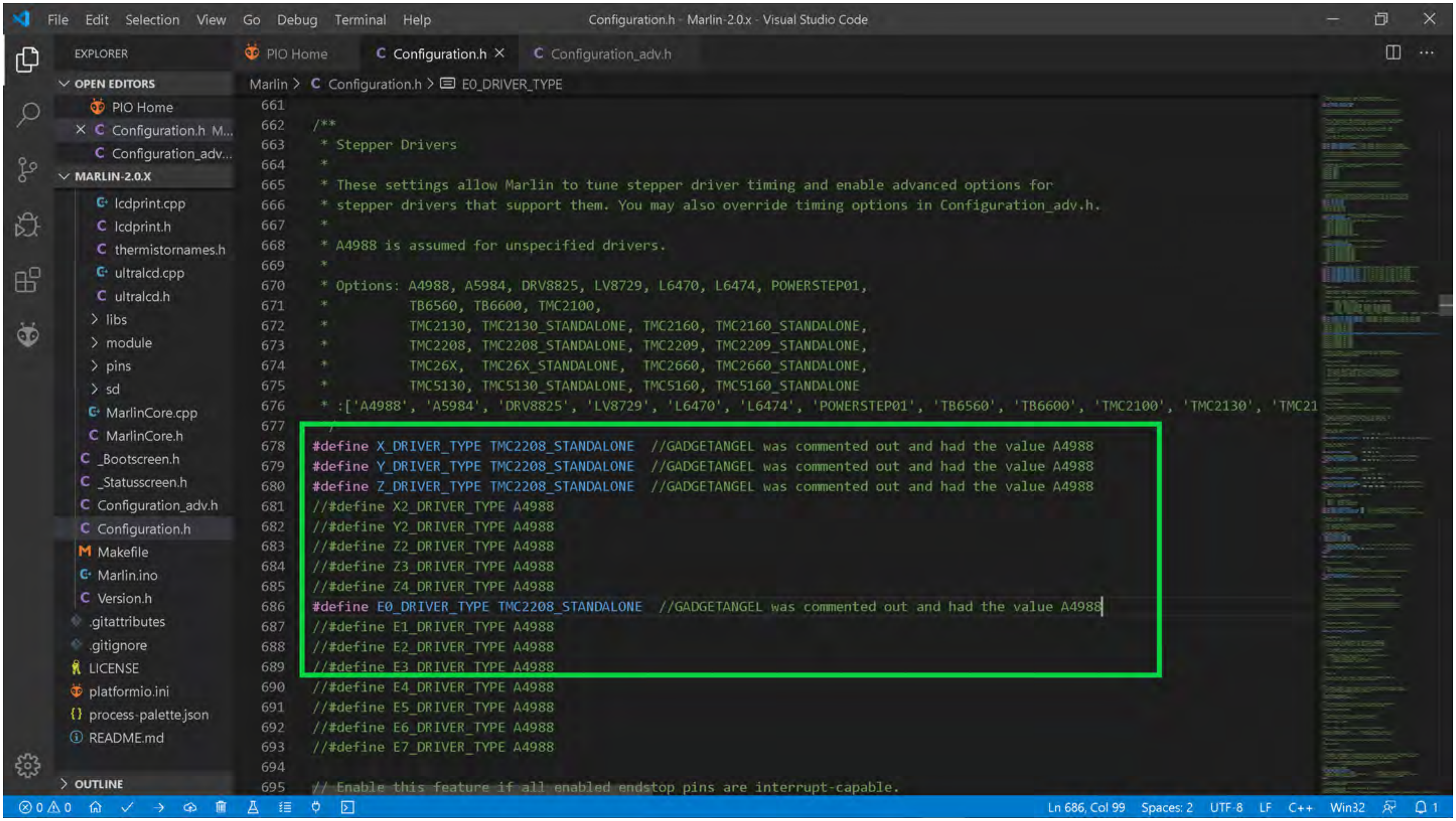


NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

The (latest release of) Marlin Setup for BIQU TMC2208 V3.0 Drivers in Stand-alone Mode

NOTE: **Go to Appendix C**, and then come back here for the changes to Marlin for BIQU TMC2208 stepper motor drivers in stand-alone mode.

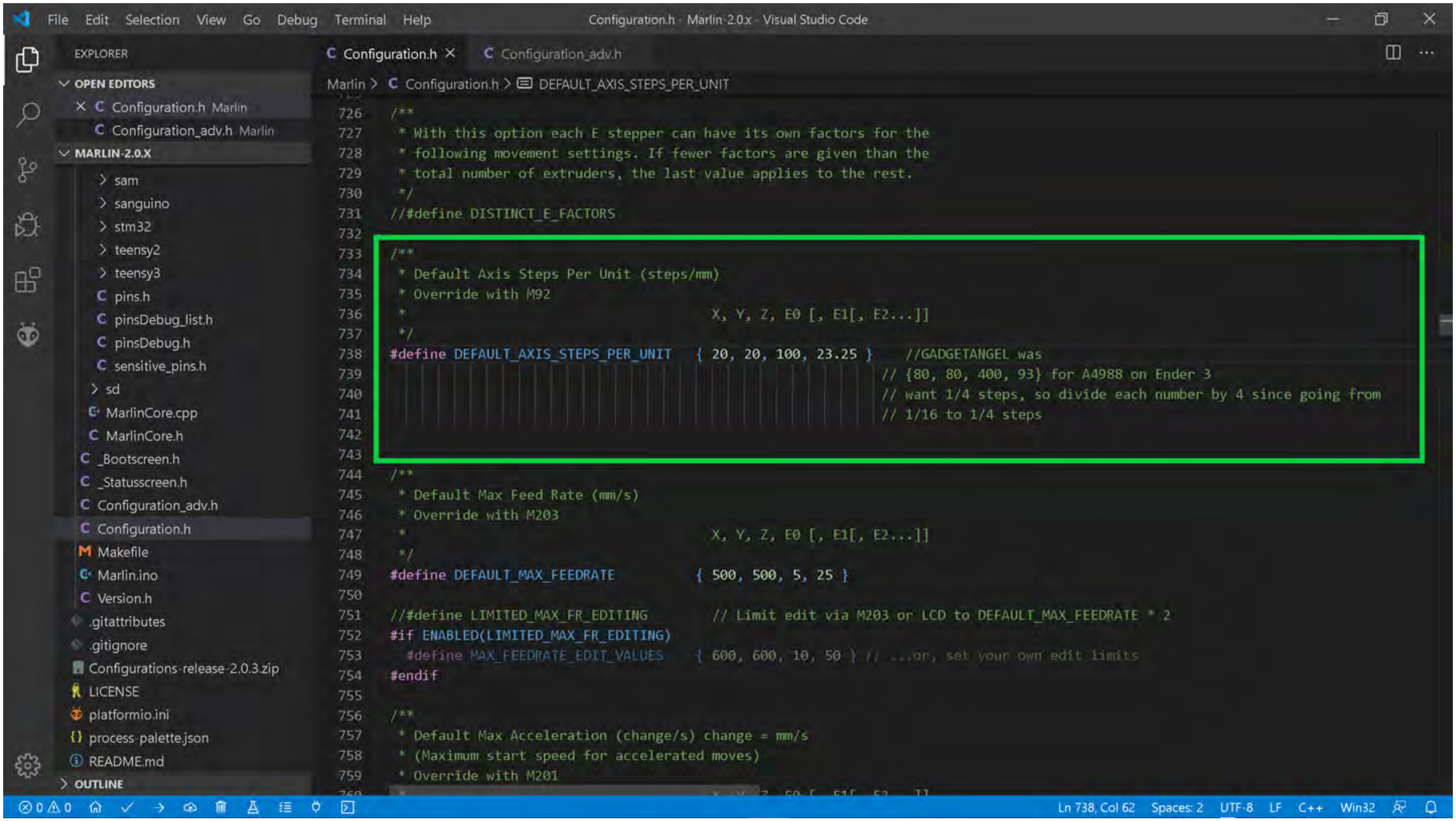
- Change the stepper motor drivers so that Marlin knows you are using TMC2208 drivers in stand-alone mode. Change one line for each axis and one for each extruder you will be using. See the picture below for an example of how to use TMC2208 drivers in stand-alone mode. When two "//" appear at the beginning of a line that means that line is commented out. To un-comment a line just remove the leading two forward slashes ("//").



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2208 V3.0 Drivers in Stand-alone Mode

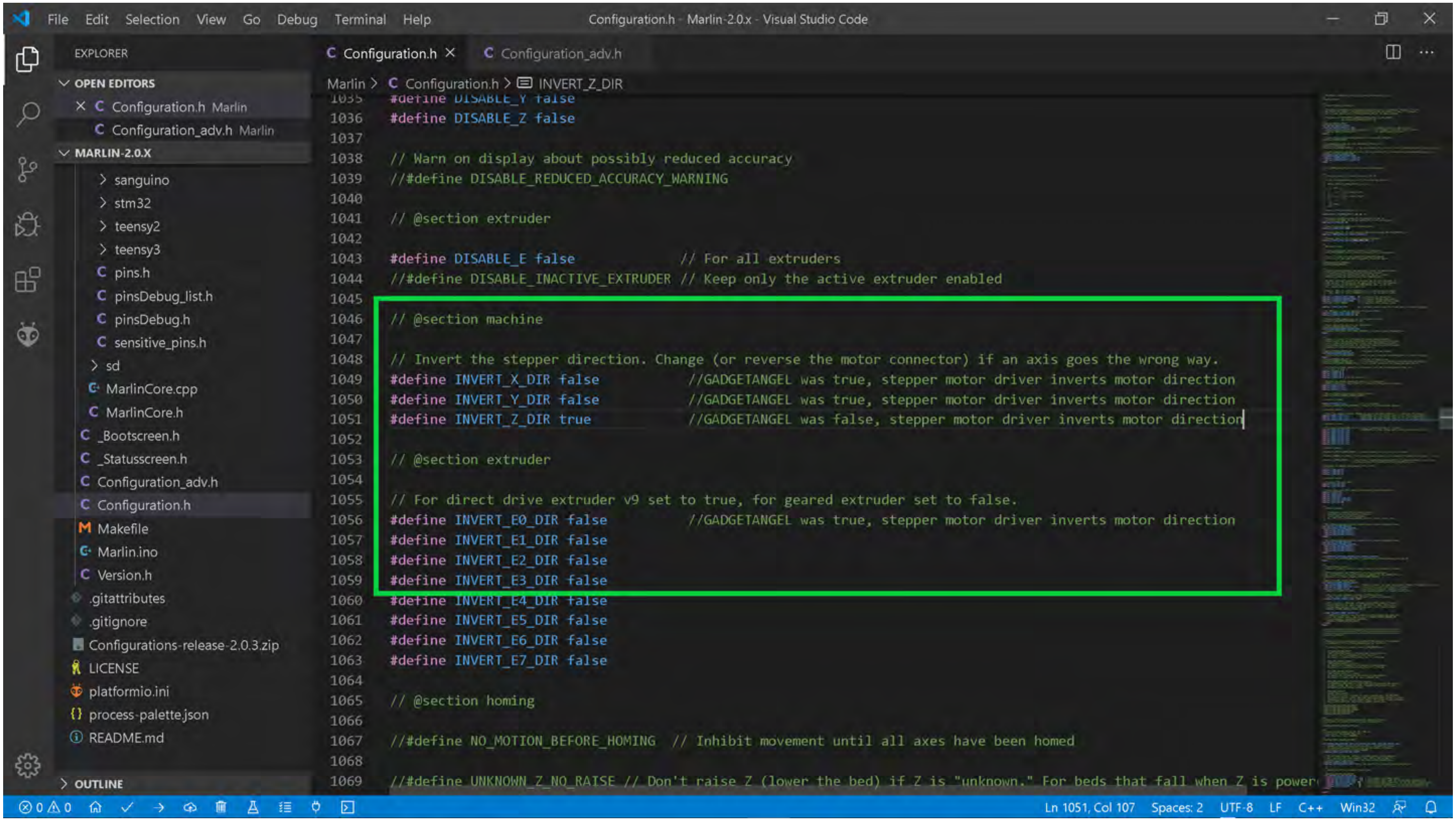
- Since I desire to use 1/4 stepping, and we are changing from A4988 stepper motor drivers on the Ender 3 to TMC2208 stepper motor drivers for each axis and the extruder stepper motor driver, we will be going from 1/16 stepping to 1/4 stepping. So we are cutting our STEPS by one quarter. Therefore, we must adjust our "DEFAULT_AXIS_STEPS_PER_UNIT" anytime our STEPS are NOT 1/16. So change "DEFAULT_AXIS_STEPS_PER_UNIT" to {20, 20, 100, 23.25}, as seen in the **GREEN** box below.



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2208 V3.0 Drivers in Stand-alone Mode

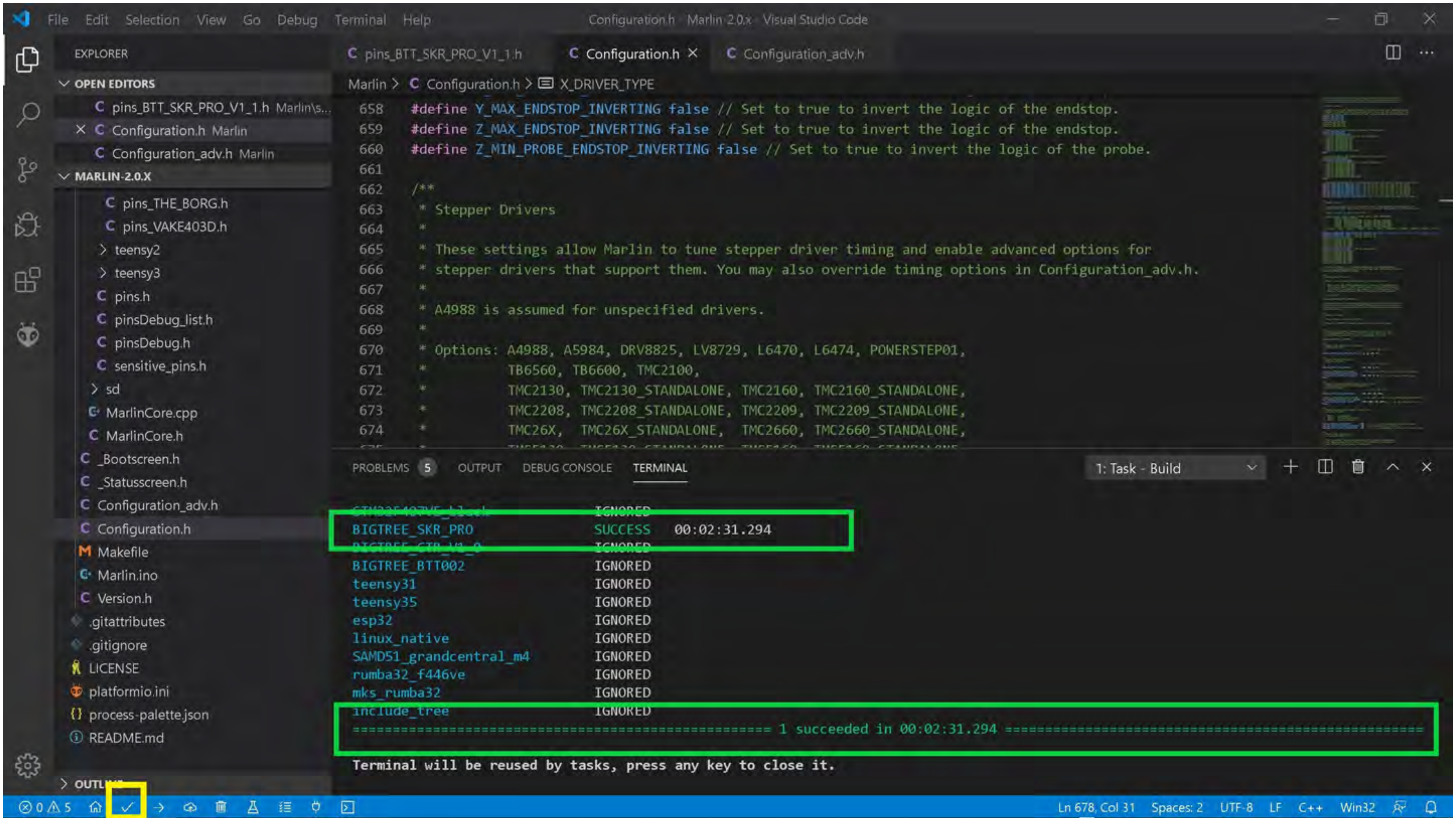
- Since the A4988 driver is what my Ender 3 used, but, now I want to use TMC2208 drivers, I must invert the stepper motor direction because the TMC2208 driver will turn the motors in the opposite direction than the A4988 driver's motor direction. So if the axis' setting you will be using the TMC2208 driver on was "true" change it to "false", as shown in the **GREEN** box below. If the setting was "false", now set it to "true", as shown in the **GREEN** box below



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2208 V3.0 Drivers in Stand-alone Mode

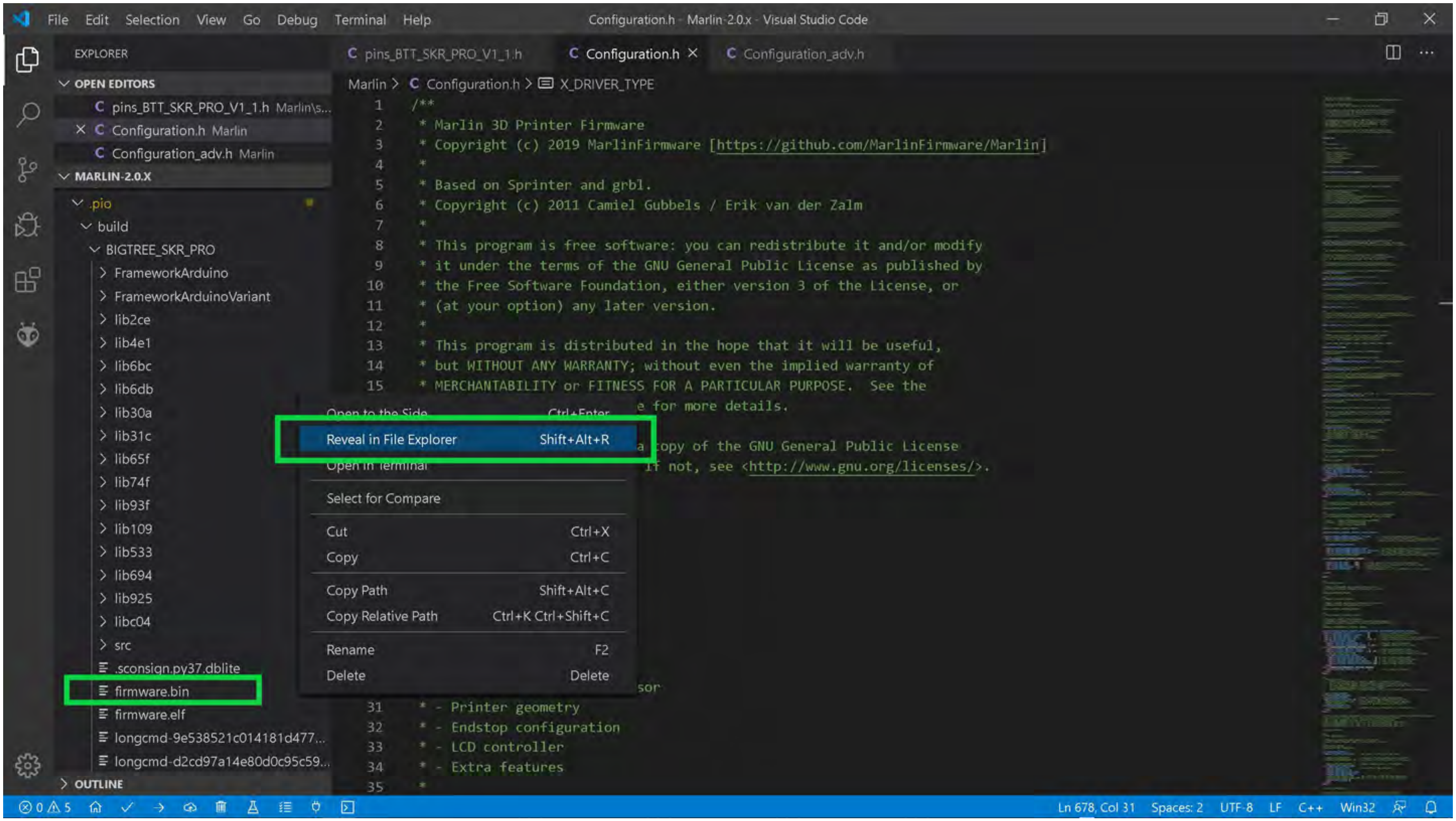
- The end of Marlin setup for BIQU TMC2208 V3.0 drivers in stand-alone mode. Now, compile your code by clicking on the check mark, as seen in the **YELLOW** box below. Once the firmware has compiled successfully you will see the following messages, as shown in the **GREEN** boxes below.



- For the SKR PRO V1.1 board you must save the "firmware.bin" file to the micro SD card then place the micro SD card into the micro SD card reader of the SKR PRO V1.1 board and turn on the power to your printer for the NEW firmware to be automatically applied to the SKR PRO MCU. To locate the "firmware.bin" file, and learn how to transfer it to your micro SD card go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2208 V3.0 Drivers in Stand-alone Mode

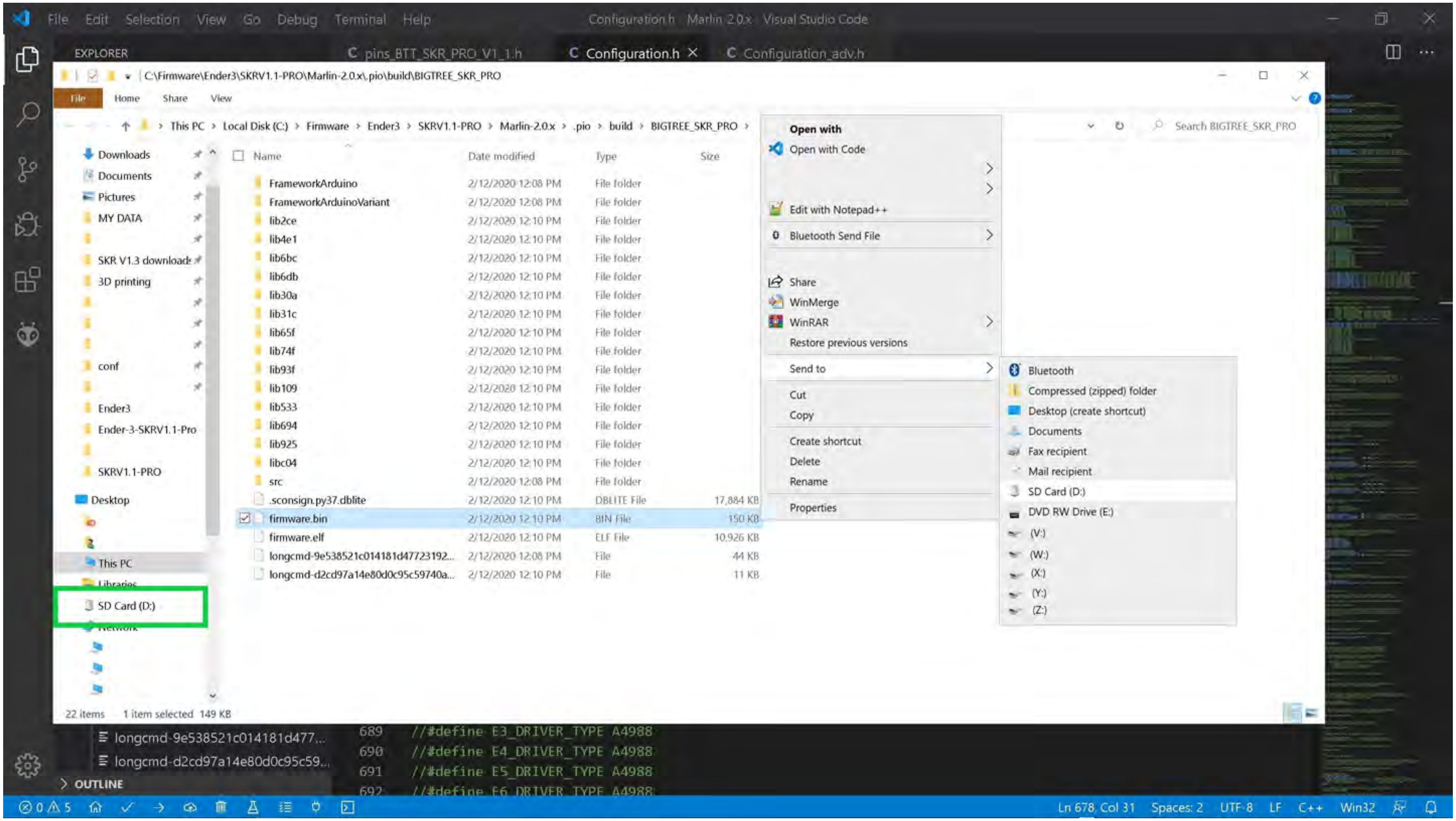
- The "firmware.bin" file can be found by using the left side of the VS code window and going to folder Marlin-2.0.X/.pio/build/BIGTREE_SKR_PRO/ and right clicking on the "firmware.bin" file name to bring up the content menu. Select "Reveal in File Explorer" to have your Window's machine open a file explorer window.



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2208 V3.0 Drivers in Stand-alone Mode

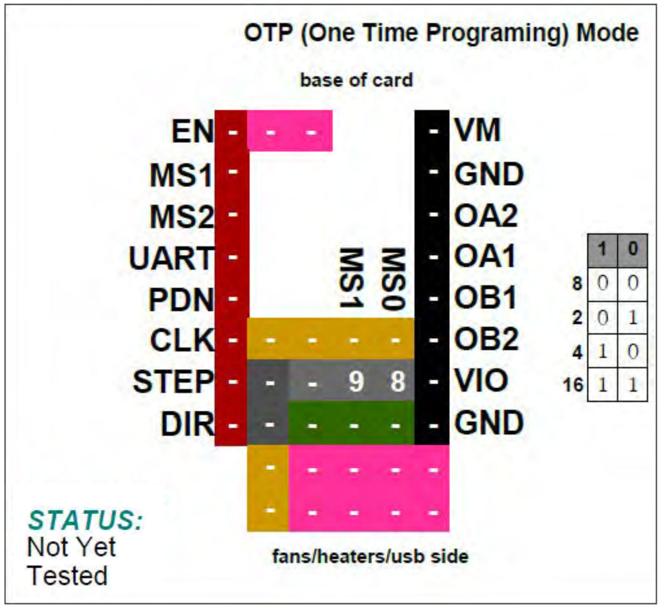
- Ensure your micro SD card, which came with your SKR PRO V1.1 board, with your SKR PRO V1.1 board is in a micro-to-SD-card adapter and plugged into your Window's SD card reader.
  - From the file explorer window right click on the "firmware.bin" file name and select "Send to" from the content menu. For my computer "D:" is my SD card reader. For your computer it may be a different drive letter. The "firmware.bin" file will be written to the SD card and renamed to "FIRMWARE.CUR".



- Right click on the SD card reader, as seen in the **GREEN** box above, and select "Eject" from the context menu. Take the micro SD card out of the adapter and place it into the micro SD card reader of the SKR PRO V1.1 board. Turn on the power to the printer (i.e. the SKR PRO V1.1 board) and the "FIRMWARE.CUR" file will be uploaded to the board. After the upload the firmware file will be renamed to "firmware.bin" and stored on the micro SD card.

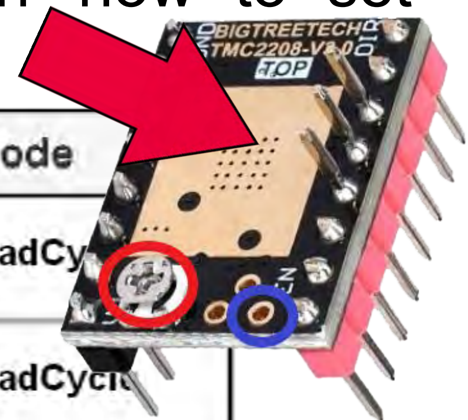
BIQU TMC2208 V3.0

One Time Programming (OTP) Mode



Note: Use 90% of the calculated V_{ref} when tuning the stepper driver board.

NOTE: Use the potentiometer (POT) on the top of the board, as shown in **RED**; or use the board's "V_{ref} Test point" location, as shown in **BLUE**, to set your V_{ref}. See **Appendix A** for instructions on how to set the V_{ref} on a driver board.

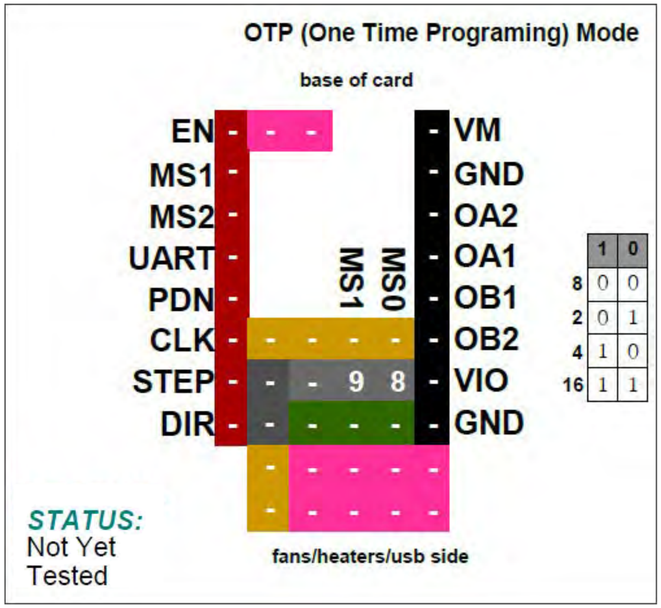


Driver Chip	MS1	MS0	Steps	Interpolation	Mode
BIQU® TMC2208 OTP Mode Maximum 16 Subdivision 35V DC 2A (peak)	GND	GND	1 / 8	1 / 256	spreadCy
	GND	VIO	1 / 2	1 / 256	spreadCycl
	VIO	GND	1 / 4	1 / 256	spreadCycle
	VIO	VIO	1 / 16	1 / 256	spreadCycle

Driving Current Calculation Formula		
R_S (Typical Sense Resistor) = 0.11Ω	$I_{MAX} = V_{ref} * 0.9286$ See Appendix B #3. Use 50% to 90% as shown below: $I_{MAX} = (V_{ref} * 0.9286) * 0.90$	$V_{ref} = I_{MAX} * 1.0769$ See Appendix B #3. Use 50% to 90% as shown below: $V_{ref} = (I_{MAX} * 1.0769) * 0.90$

BIQU TMC2208 V3.0

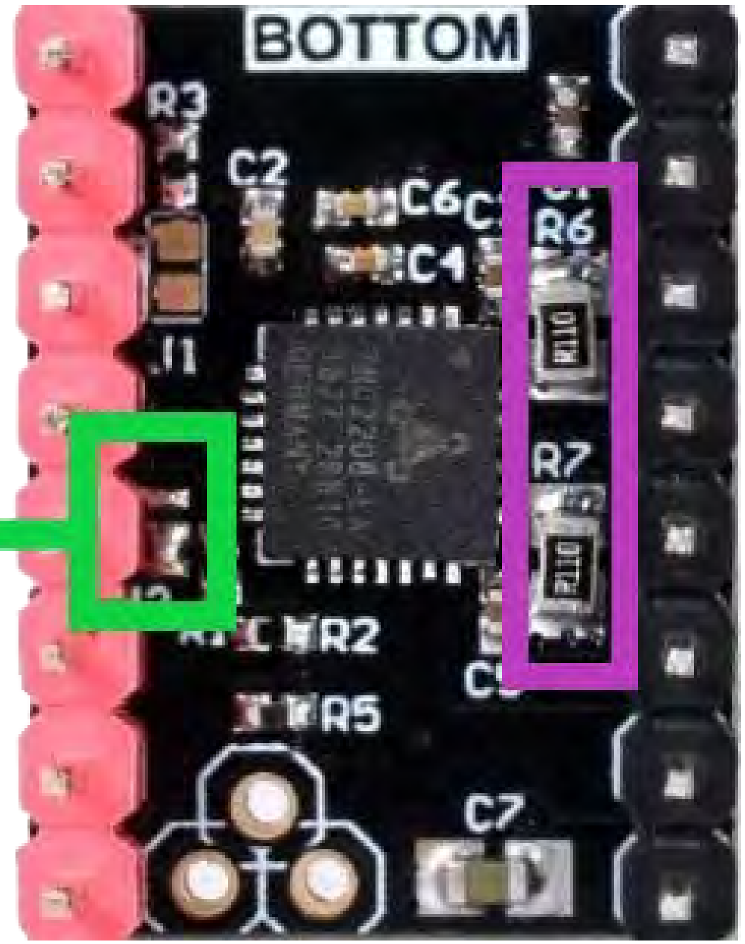
One Time Programming (OTP) Mode



NOTE: **Stand-alone Mode by default uses StealthChop, if you want SpreadCycle, you MUST use OTP mode. See TMC220x Configurator for One-Time-Programming Information: TMC220x Configurator.**

Important: To place BIQU TMC2208 V3.0 into OTP mode, at J2, you must solder the top two pads together OR solder the bottom two pads together on the bottom of the driver board.

As an example, the picture shows the TOP two pads soldered together, as shown in the **GREEN box**, so that OTP mode can be obtained. To do One-Time-Programming (OTP), the TMC2208 must be placed in UART mode to program it. The **PURPLE box** shows the location of the current sense resistors (Rs).



MOST BIQU TMC2208 V3.0 driver boards, when purchased for UART mode, will have two adjacent J2 pads already soldered together (located on the bottom of the driver board).

NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

BIQU TMC2208 V3.0

One Time Programming (OTP) Mode

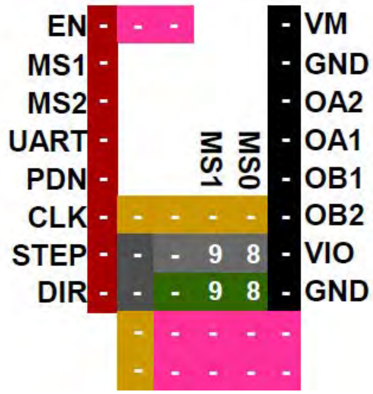
OTP (One Time Programing) Mode

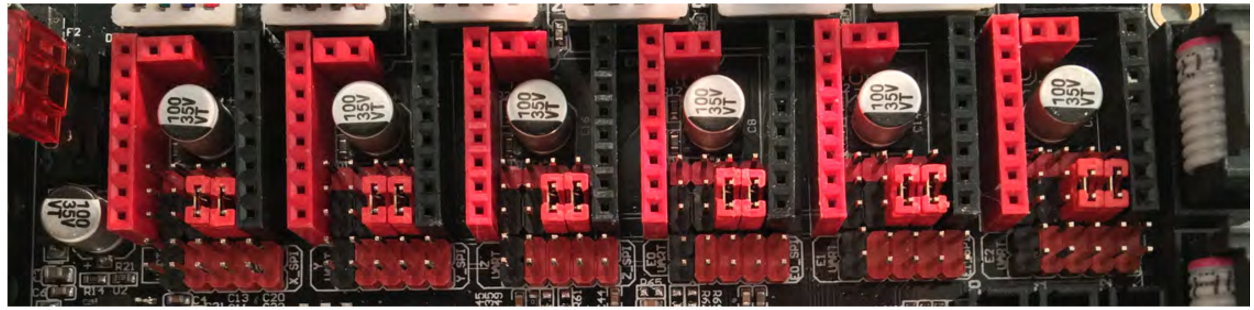
OTP

1 / 8

Interpolation: **1/256**

SpreadCycle



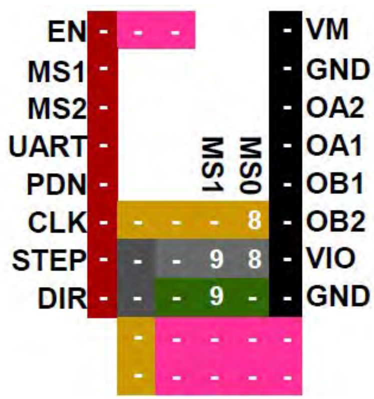


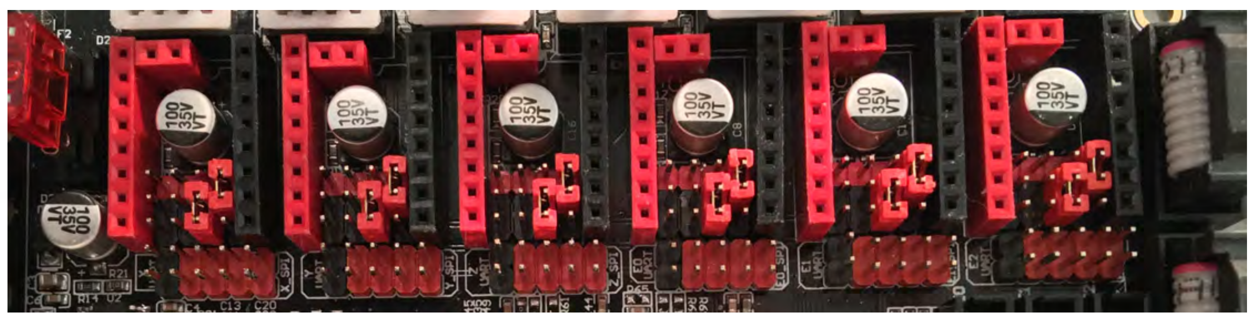
OTP

1 / 2

Interpolation: **1/256**

SpreadCycle





NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

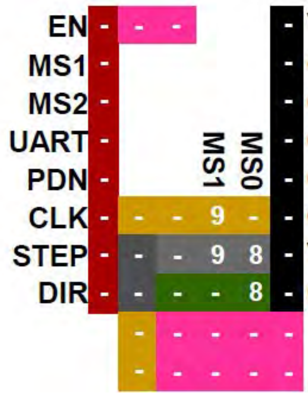
OTP (One Time Programing) Mode

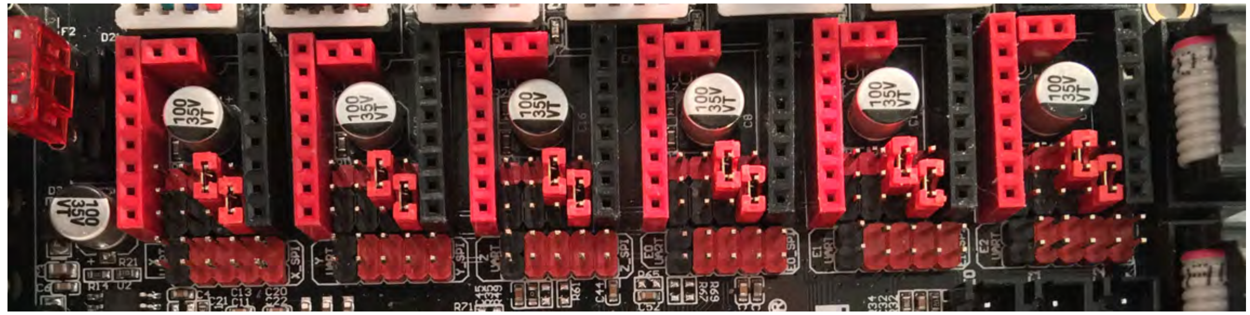
OTP

1 / 4

Interpolation: **1/256**

SpreadCycle



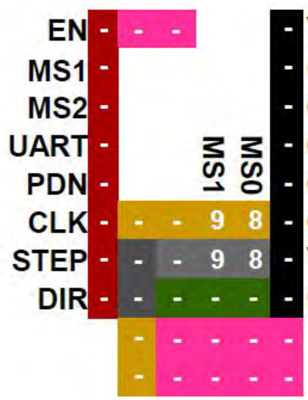


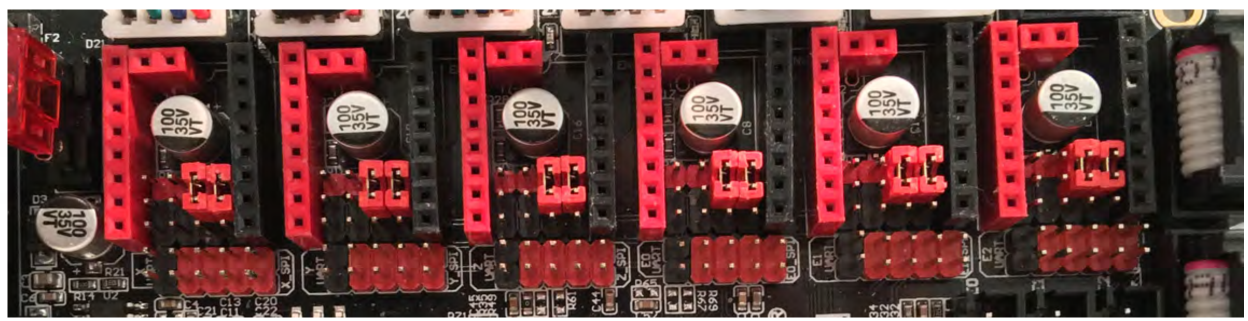
OTP

1 / 16

Interpolation: **1/256**

SpreadCycle

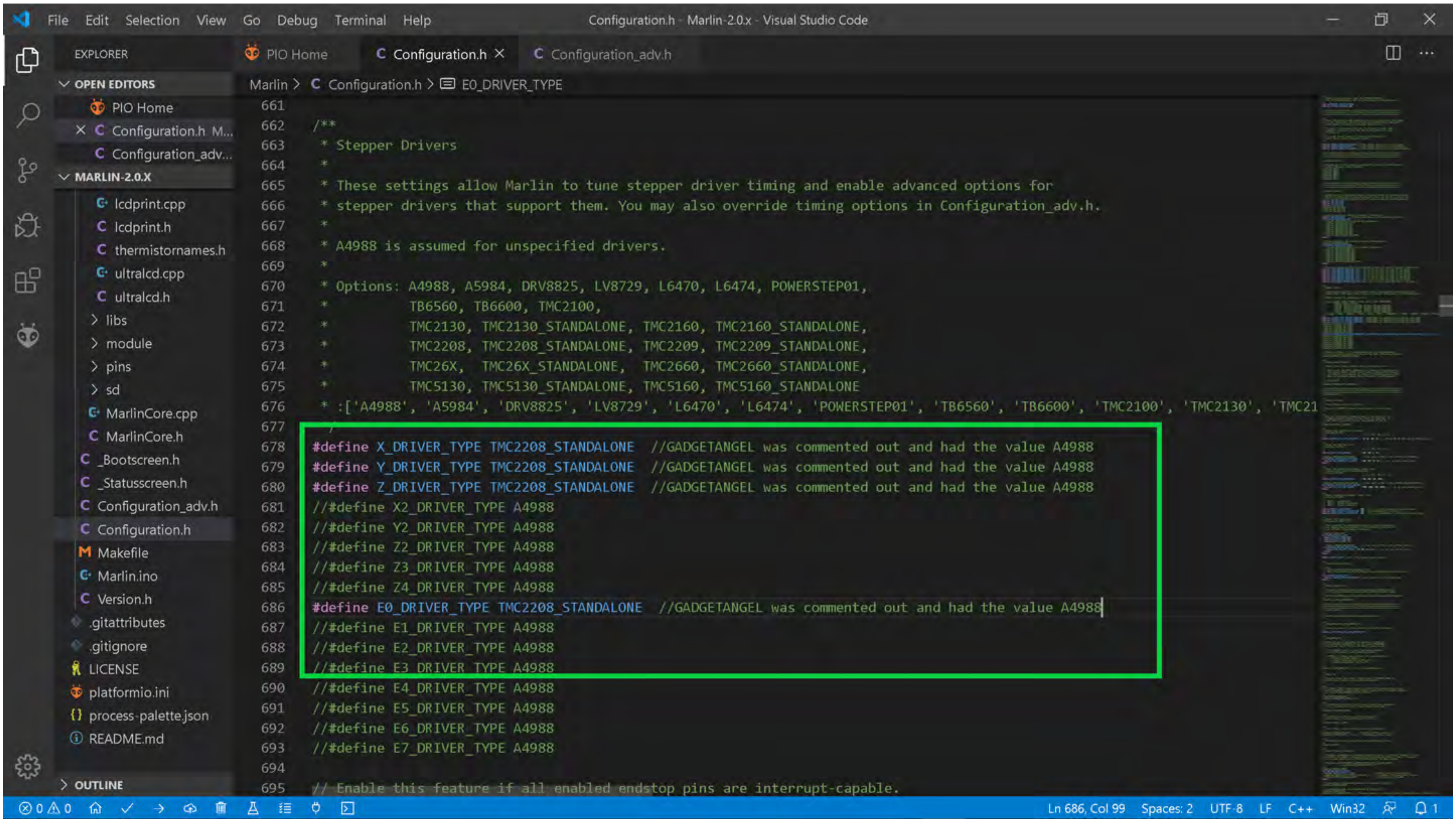




The (latest release of) Marlin Setup for BIQU TMC2208 V3.0 Drivers in One Time Programming (OTP) Mode

NOTE: **Go to Appendix C**, and then come back here for the changes to Marlin for BIQU TMC2208 stepper motor drivers in OTP mode.

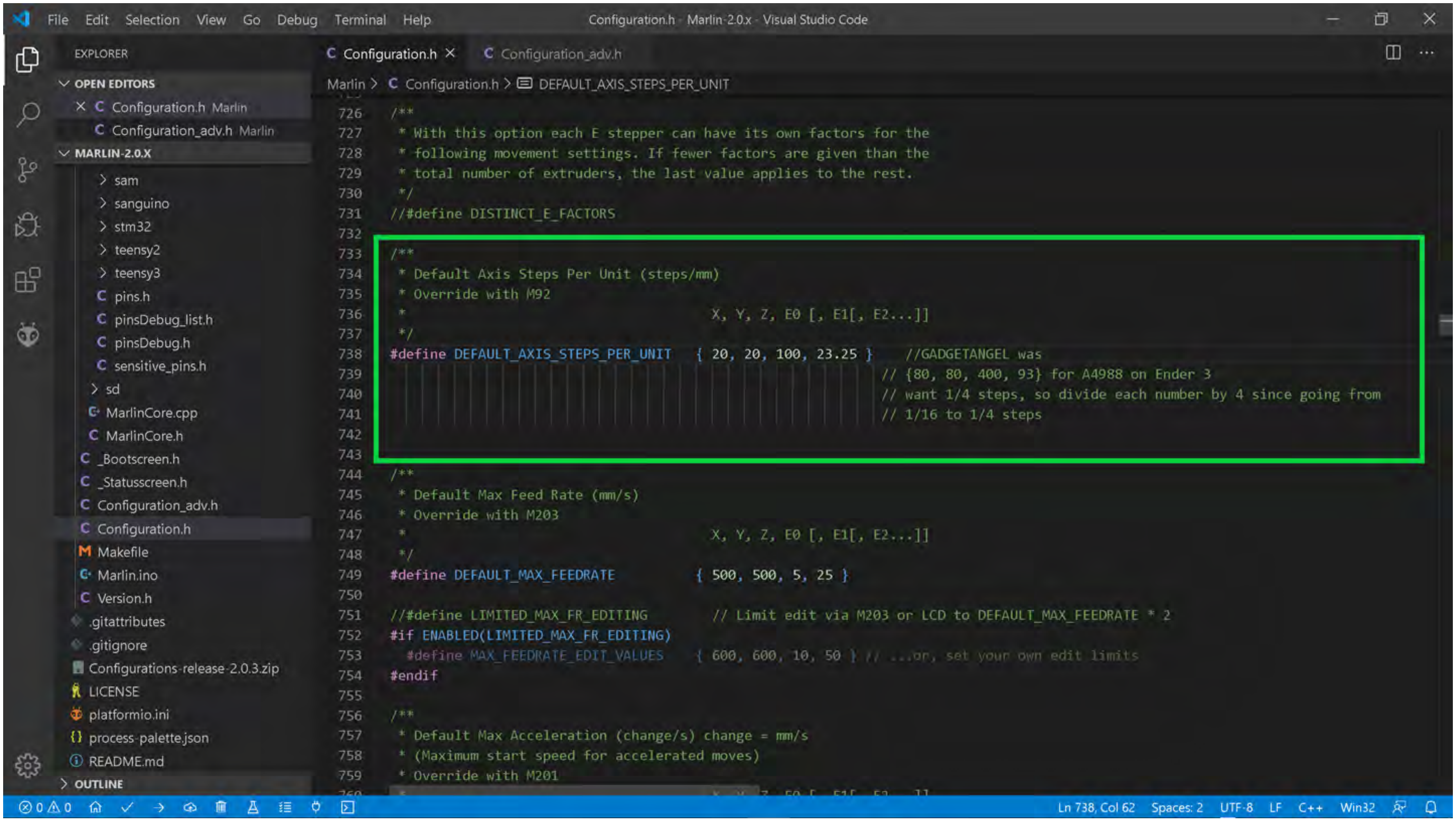
- Change the stepper motor drivers so that Marlin knows you are using TMC2208 drivers in OTP mode. Change one line for each axis and one for each extruder you will be using. See the picture below for an example of how to use TMC2208 drivers in OTP mode. When two "//" appear at the beginning of a line that means that line is commented out. To un-comment a line just remove the leading two forward slashes ("//").



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2208 V3.0 Drivers in One Time Programming (OTP) Mode

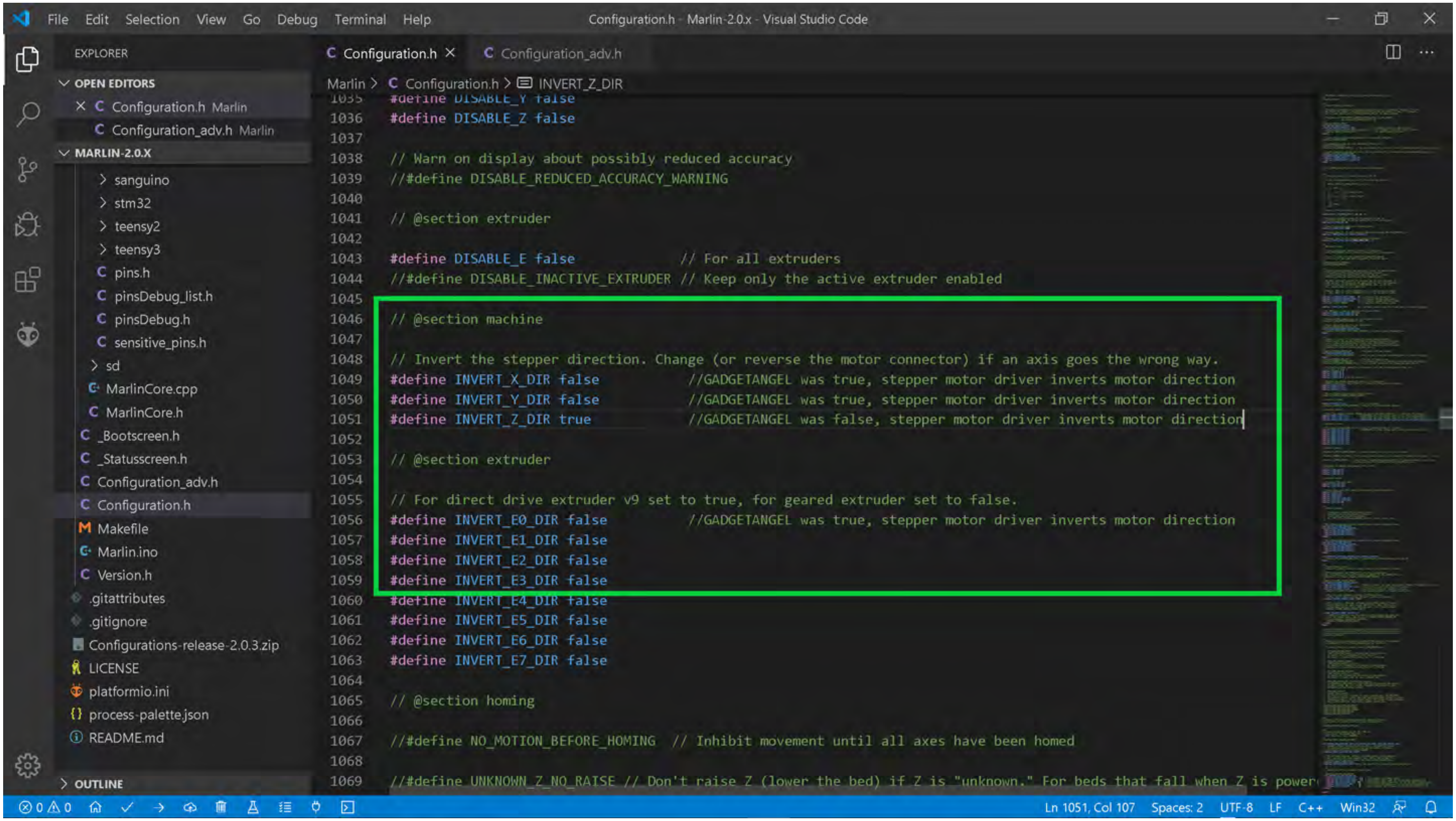
- Since I desire to use 1/4 stepping, and we are changing from A4988 stepper motor drivers on the Ender 3 to TMC2208 stepper motor drivers for each axis and the extruder stepper motor driver, we will be going from 1/16 stepping to 1/4 stepping. So we are cutting our STEPS by one quarter. Therefore, we must adjust our "DEFAULT_AXIS_STEPS_PER_UNIT" anytime our STEPS are NOT 1/16. So change "DEFAULT_AXIS_STEPS_PER_UNIT" to {20, 20, 100, 23.25}, as seen in the **GREEN** box below.



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2208 V3.0 Drivers in One Time Programming (OTP) Mode

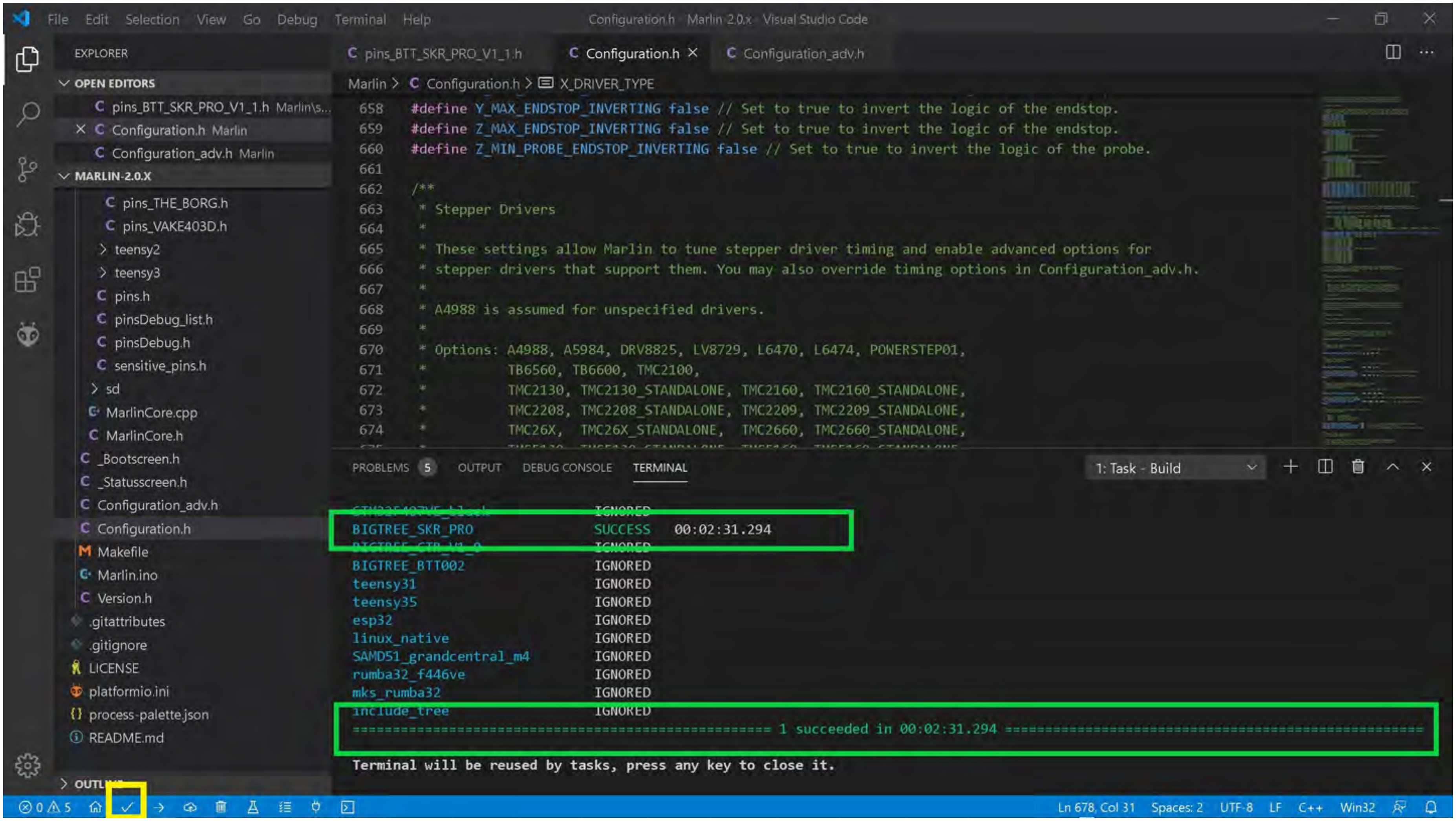
- Since the A4988 driver is what my Ender 3 used, but, now I want to use TMC2208 drivers, I must invert the stepper motor direction because the TMC2208 driver will turn the motors in the opposite direction than the A4988 driver's motor direction. So if the axis' setting you will be using the TMC2208 driver on was "true" change it to "false", as shown in the **GREEN** box below. If the setting was "false", now set it to "true", as shown in the **GREEN** box below



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2208 V3.0 Drivers in One Time Programming (OTP) Mode

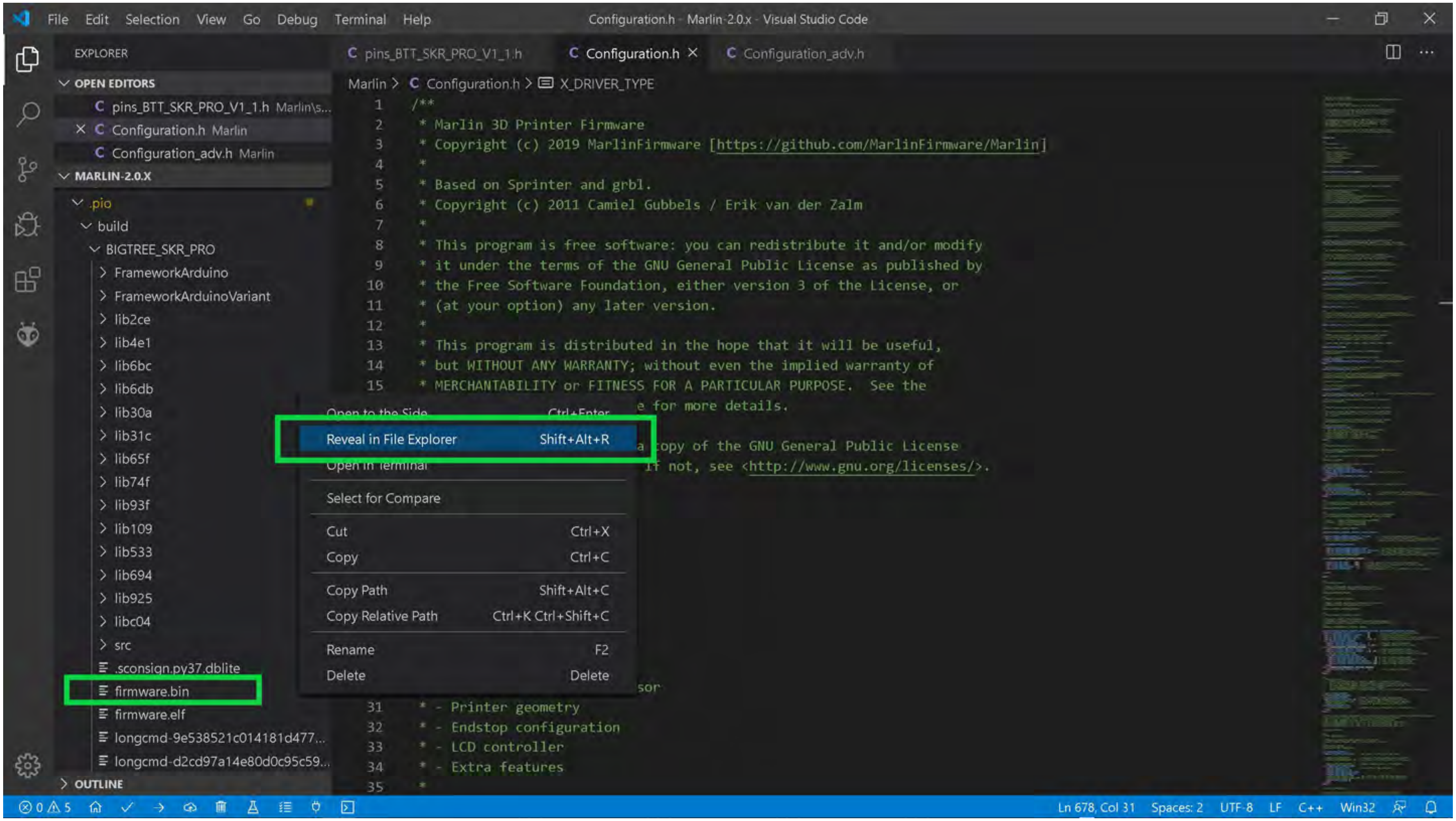
- The end of Marlin setup for BIQU TMC2208 V3.0 drivers in OTP mode. Now, compile your code by clicking on the check mark, as seen in the **YELLOW** box below. Once the firmware has compiled successfully you will see the following messages, as shown in the **GREEN** boxes below.



- For the SKR PRO V1.1 board you must save the "firmware.bin" file to the micro SD card then place the micro SD card into the micro SD card reader of the SKR PRO V1.1 board and turn on the power to your printer for the NEW firmware to be automatically applied to the SKR PRO MCU. To locate the "firmware.bin" file, and learn how to transfer it to your micro SD card go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2208 V3.0 Drivers in One Time Programming (OTP) Mode

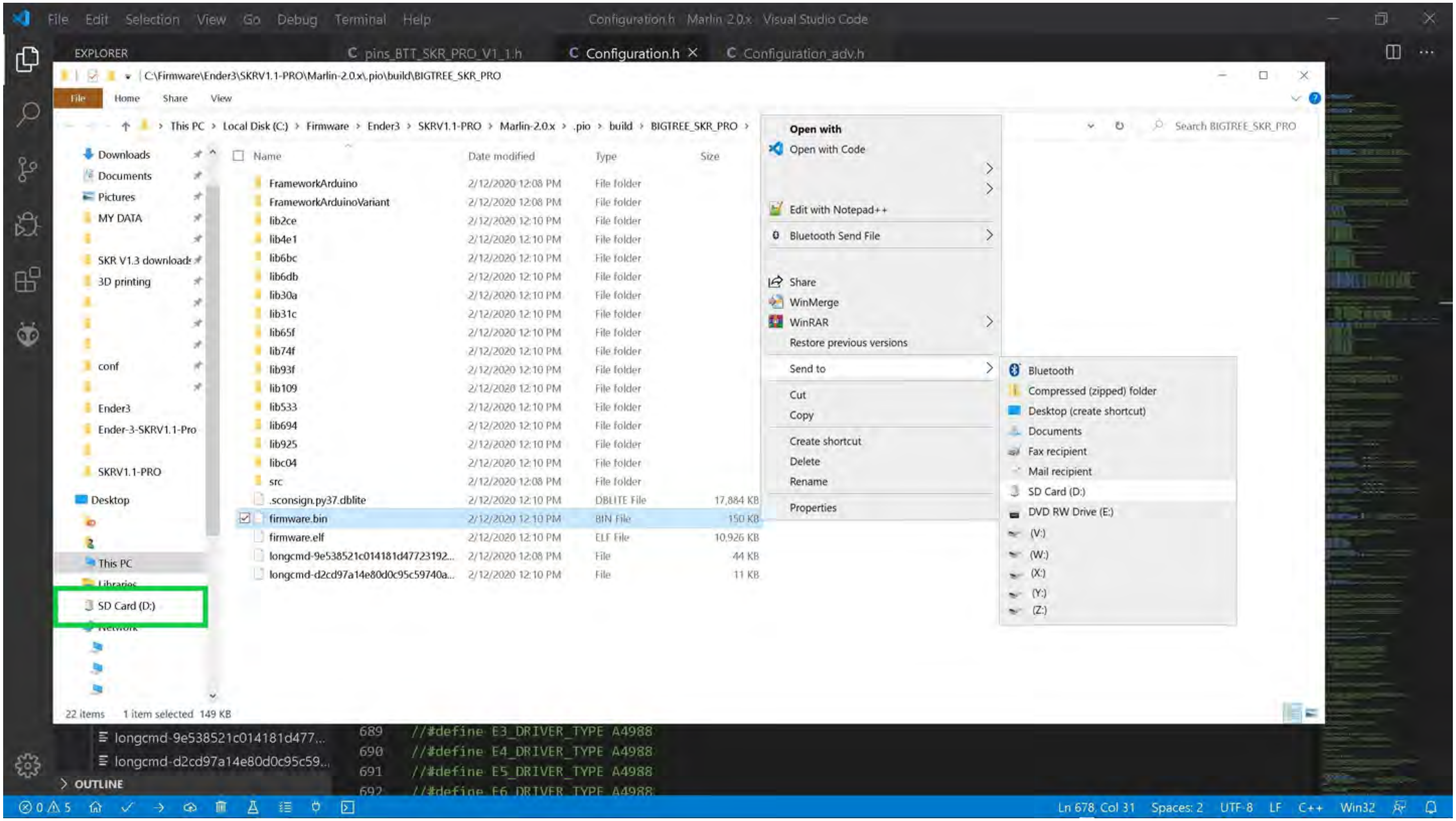
- The "firmware.bin" file can be found by using the left side of the VS code window and going to folder Marlin-2.0.X/.pio/build/ BIGTREE_SKR_PRO/ and right clicking on the "firmware.bin" file name to bring up the content menu. Select "Reveal in File Explorer" to have your Window's machine open a file explorer window.



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2208 V3.0 Drivers in One Time Programming (OTP) Mode

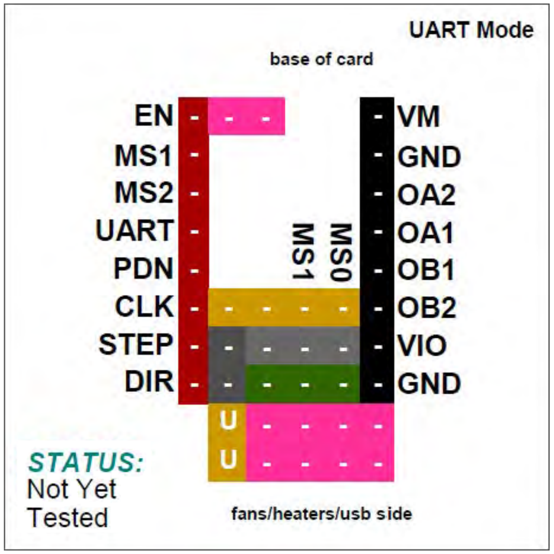
- Ensure your micro SD card, which came with your SKR PRO V1.1 board, with your SKR PRO V1.1 board is in a micro-to-SD-card adapter and plugged into your Window's SD card reader.
  - From the file explorer window right click on the "firmware.bin" file name and select "Send to" from the content menu. For my computer "D:" is my SD card reader. For your computer it may be a different drive letter. The "firmware.bin" file will be written to the SD card and renamed to "FIRMWARE.CUR".



- Right click on the SD card reader, as seen in the **GREEN** box above, and select "Eject" from the context menu. Take the micro SD card out of the adapter and place it into the micro SD card reader of the SKR PRO V1.1 board. Turn on the power to the printer (i.e. the SKR PRO V1.1 board) and the "FIRMWARE.CUR" file will be uploaded to the board. After the upload the firmware file will be renamed to "firmware.bin" and stored on the micro SD card.

BIQU TMC2208 V3.0

UART Mode



Note: **You can use 50% to 90% of the calculated I_{RMS} (I_{MAX}/1.414) when tuning ("X_CURRENT", "Y_CURRENT", etc. the stepper motor driver in the firmware.**

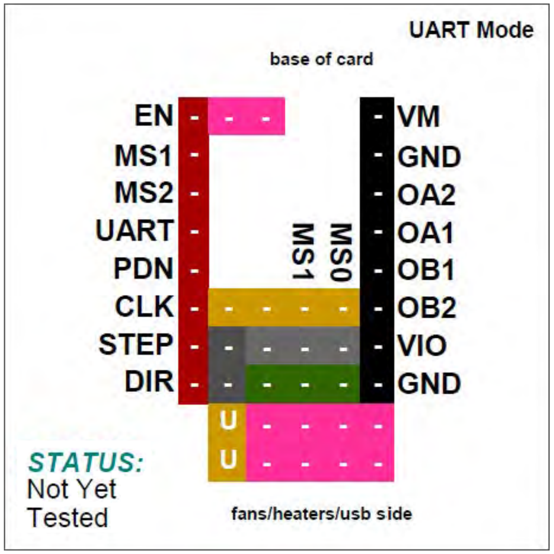
See the next page for further information.

Driver Chip	Steps are set inside of your Firmware	
BIQU® TMC2208 UART Mode Maximum 256 Subdivision 35V DC 2A (peak)		
Driving Current Calculation Formula R_S(Typical Sense Resistor)= 0.11Ω	$I_{MAX} = V_{ref}$ See Appendix B #4. Use 50% to 90% as shown below: $I_{MAX} = I_{MAX} * 0.90$	$V_{ref} = I_{MAX}$ See Appendix B #4. Use 50% to 90% as shown below: $V_{ref} = V_{ref} * 0.90$

NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

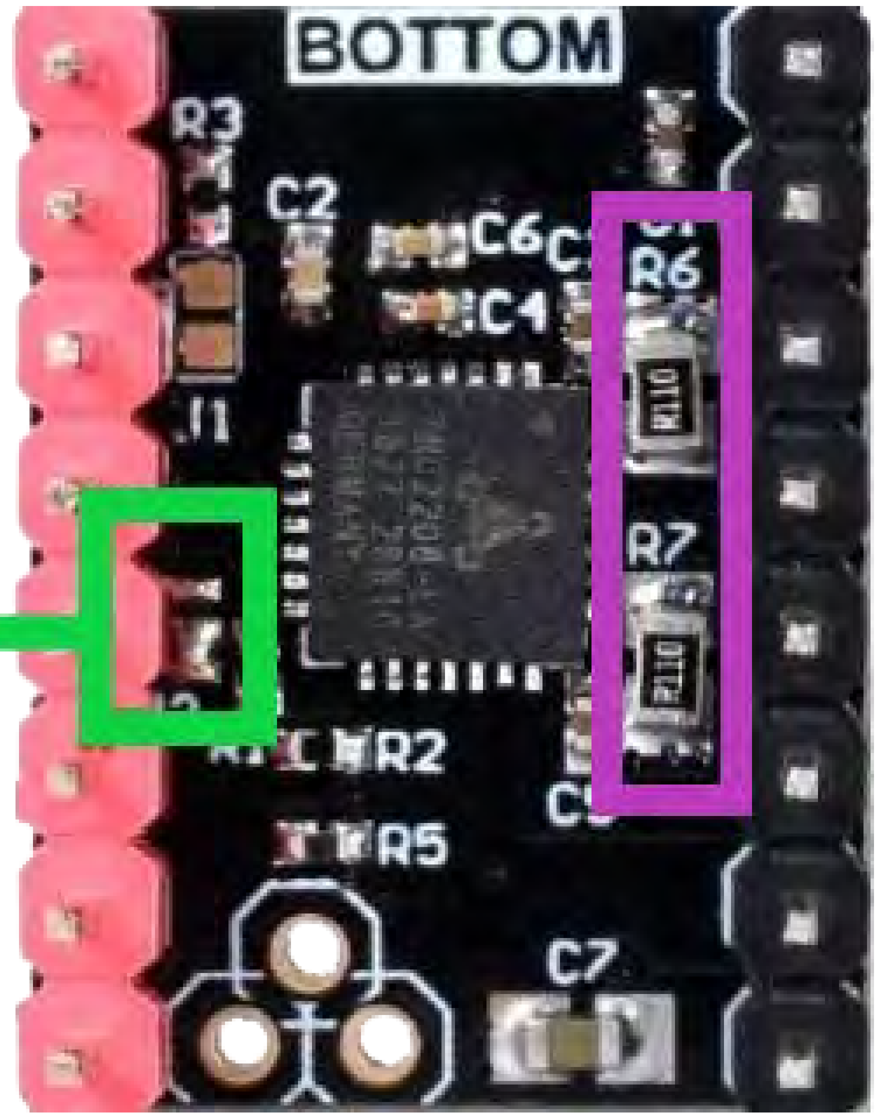
BIQU TMC2208 V3.0

UART Mode



Important: **To ensure that the BIQU TMC2208 V3.0 is in UART Mode, check to see if two adjacent J2 pads are soldered together on the bottom of the driver board, as seen in the GREEN box below. The PURPLE box shows the location of the current sense resistors (R_S).**

MOST BIQU TMC2208 V3.0 driver boards, when purchased for UART mode, will have two adjacent J2 pads already soldered together (located on the bottom of the driver board).



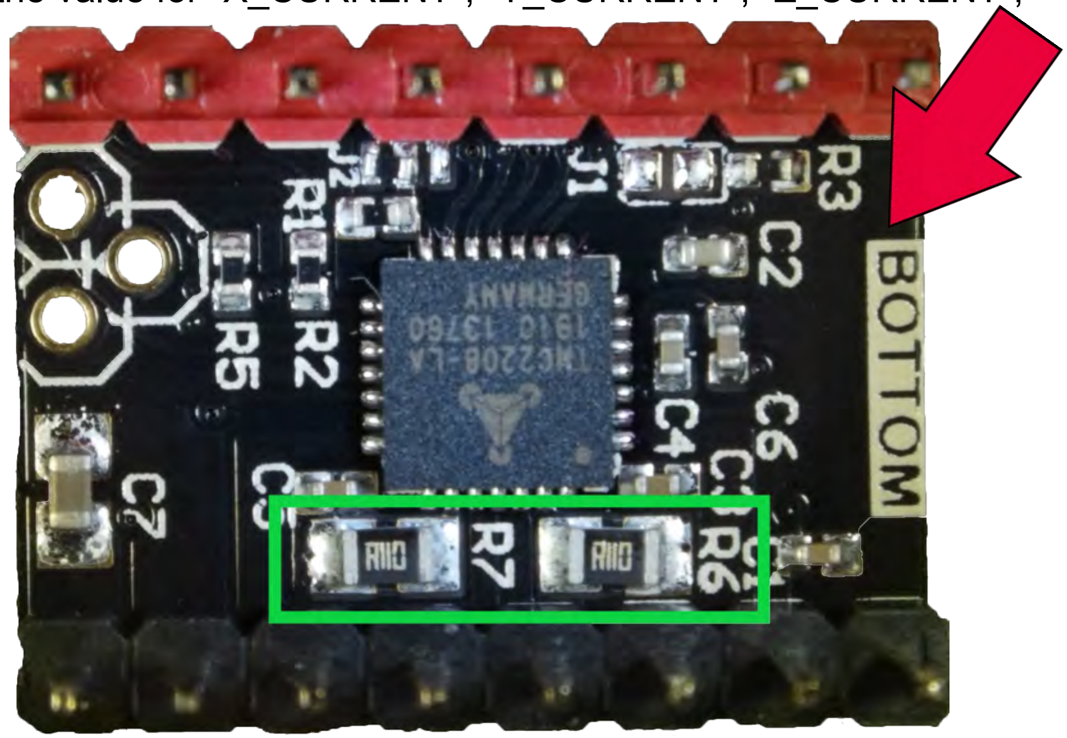
BIQU TMC2208 V3.0

UART Mode

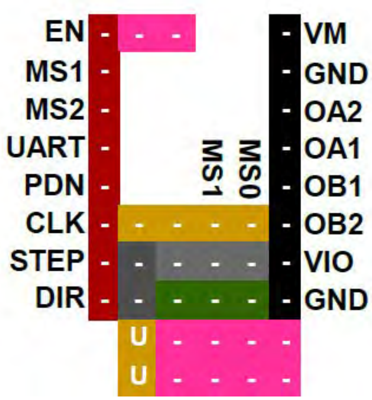
UART Mode

Note: The location of the current sense resistors are shown in **GREEN**. Use the current sense resistors' value in the Marlin Firmware ("X_RSENSE", "Y_RSENSE", "Z_RSENSE" and/or "E0_RSENSE") so that the appropriate current limit can be sent to the driver board. If you do not want to use V_{ref} as the value for "X_CURRENT", "Y_CURRENT", "Z_CURRENT" and/or "E0_CURRENT", you should use I_{RMS} instead. You find I_{RMS} by taking I_{MAX} and dividing it by 1.414 ($I_{RMS}=I_{MAX}/1.414$). You use 50% to 90% of the calculated I_{RMS} as the value for "X_CURRENT", "Y_CURRENT", "Z_CURRENT", and/or "E0_CURRENT".

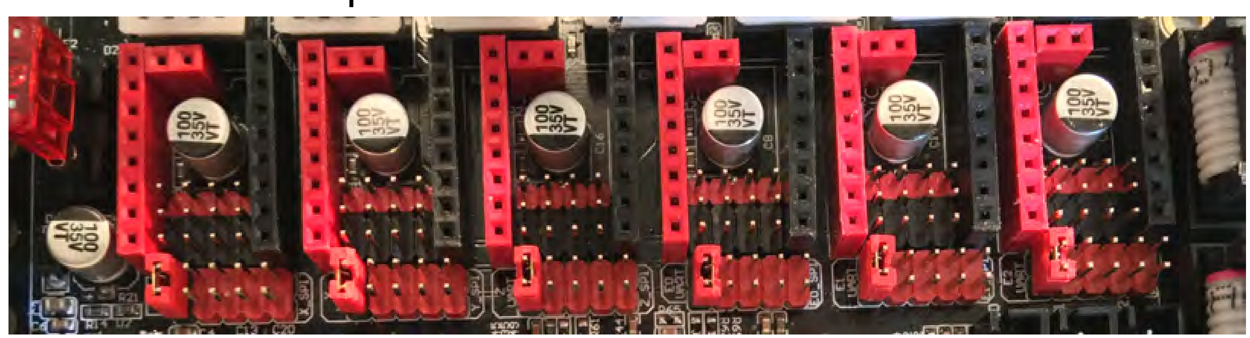
R_s = R050 is 0.05 Ohms
R_s = R062 is 0.062 Ohms
R_s = R068 is 0.068 Ohms
R_s = R075 is 0.075 Ohms
R_s = R100 is 0.1 Ohms
R_s = R110 is 0.11 Ohms
R_s = R150 is 0.15 Ohms
R_s = R200 is 0.2 Ohms
R_s = R220 is 0.22 Ohms



UART



Note: Set Jumper "U" for UART MODE!!

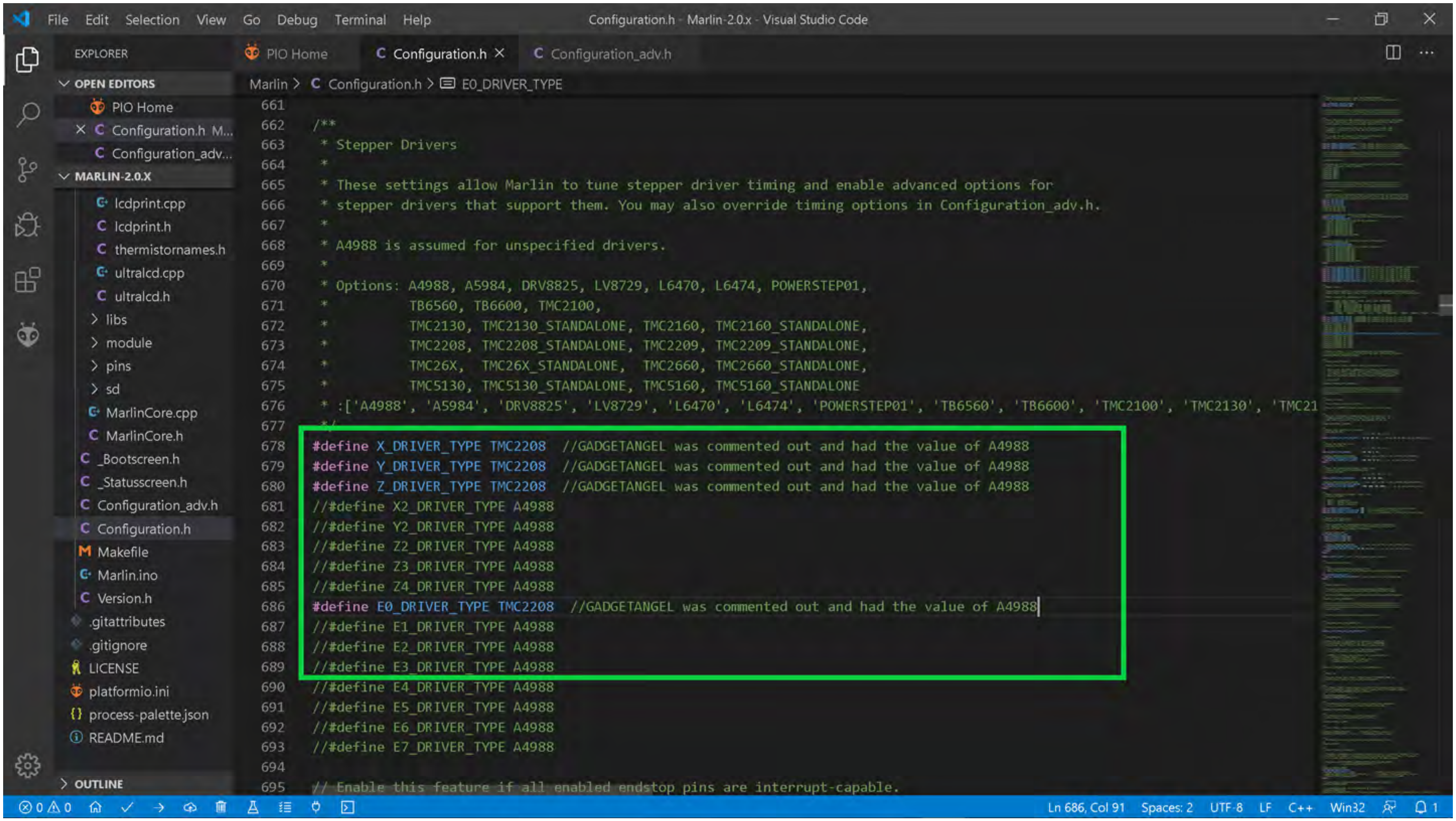


NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

The (latest release of) Marlin Setup for BIQU TMC2208 V3.0 Drivers in UART Mode

NOTE: **Go to Appendix C**, and then come back here for the changes to Marlin for BIQU TMC2208 V3.0 stepper motor drivers in UART mode.

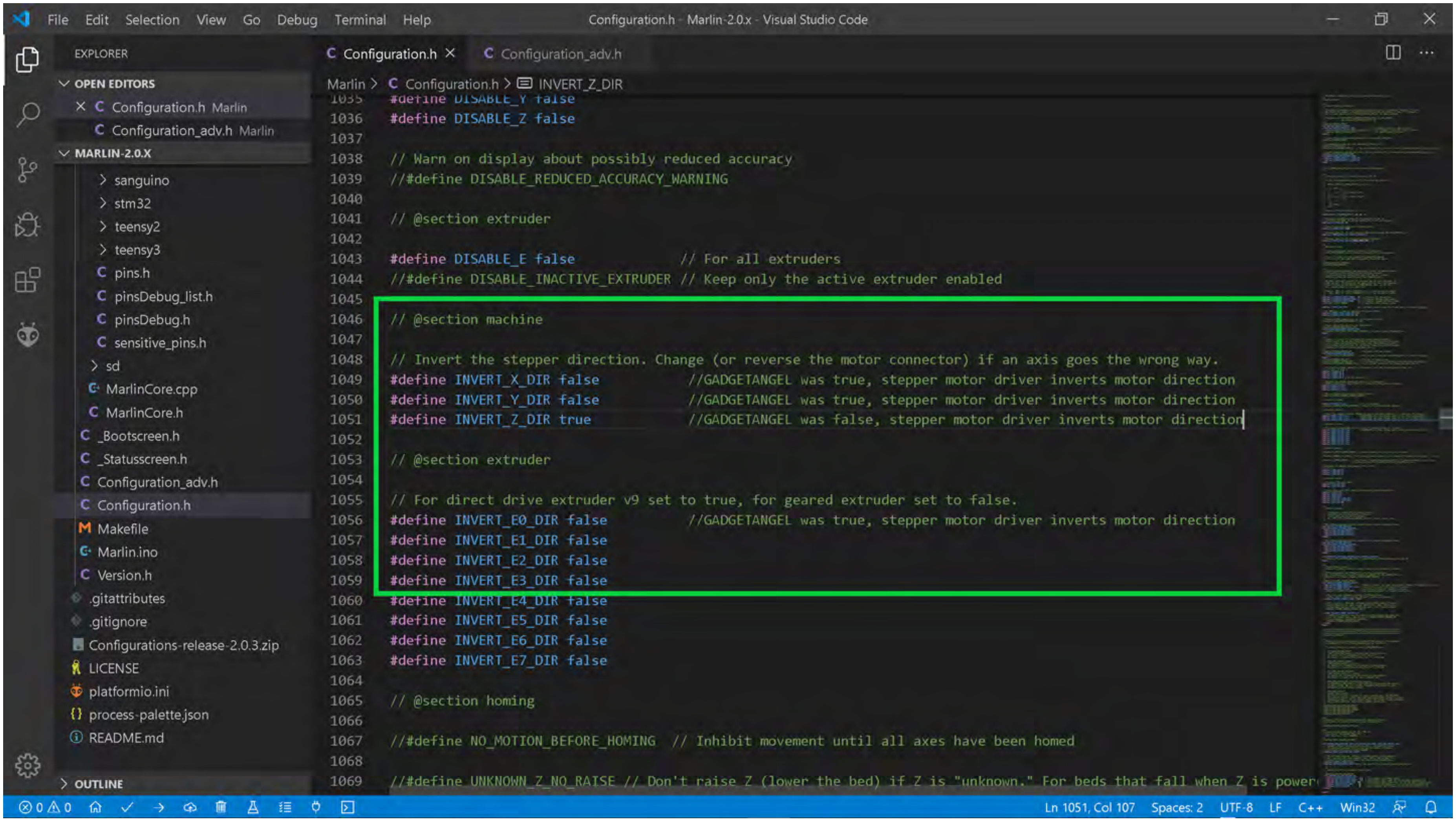
- Change the stepper motor drivers so that Marlin knows you are using TMC2208 drivers in UART mode. Change one line for each axis and one for each extruder you will be using. See the picture below for an example of how to use TMC2208 drivers in UART mode. When two "//" appear at the beginning of a line that means that line is commented out. To un-comment a line just remove the leading two forward slashes ("//").



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2208 V3.0 Drivers in UART Mode

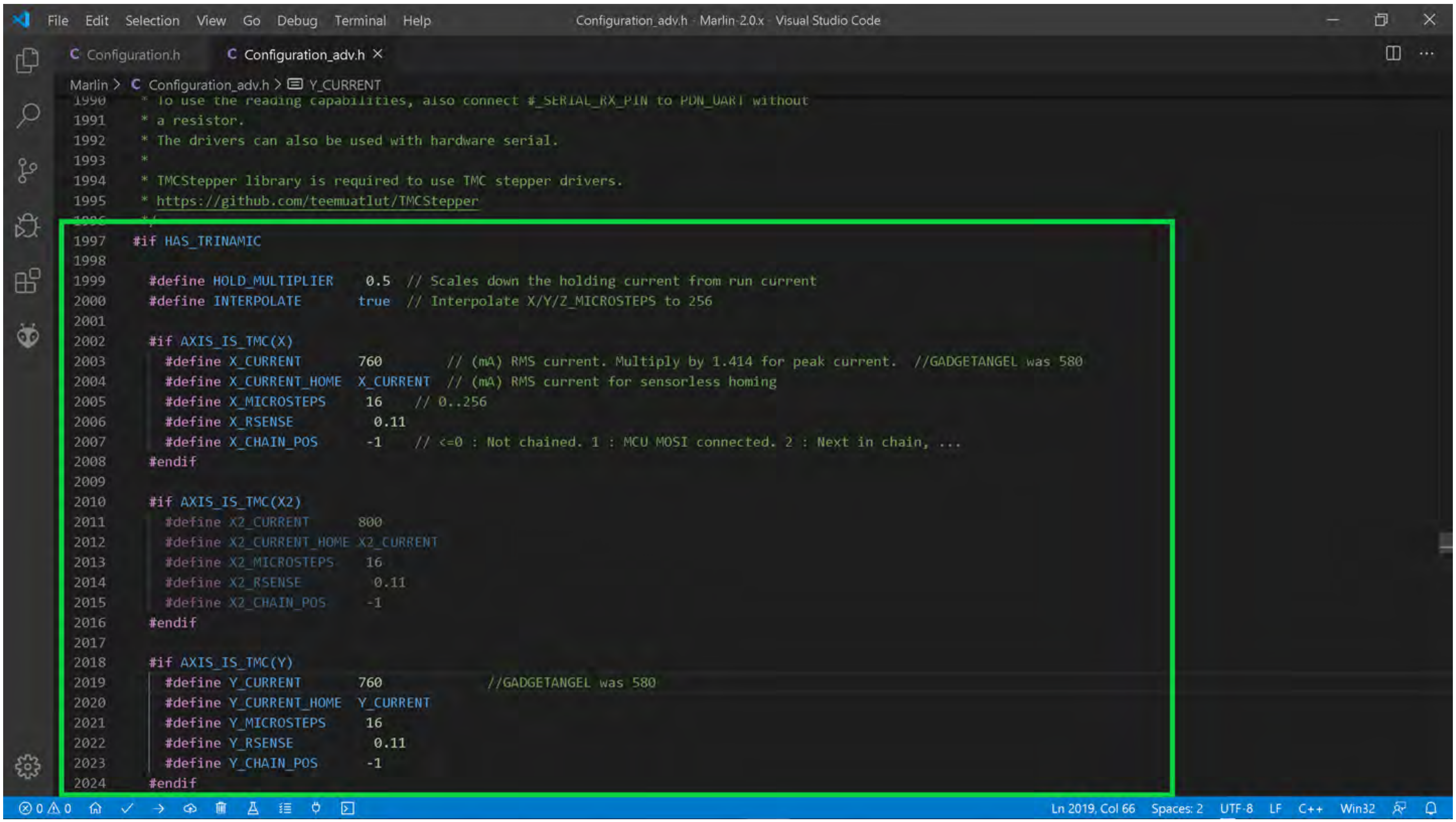
- Since the A4988 driver is what my Ender 3 used, but, now I want to use TMC2208 drivers, I must invert the stepper motor direction because the TMC2208 driver will turn the motors in the opposite direction than the A4988 driver's motor direction. So if the axis' setting you will be using the TMC2208 driver on was "true" change it to "false", as shown in the **GREEN** box below. If the setting was "false", now set it to "true", as shown in the **GREEN** box below



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2208 V3.0 Drivers in UART Mode

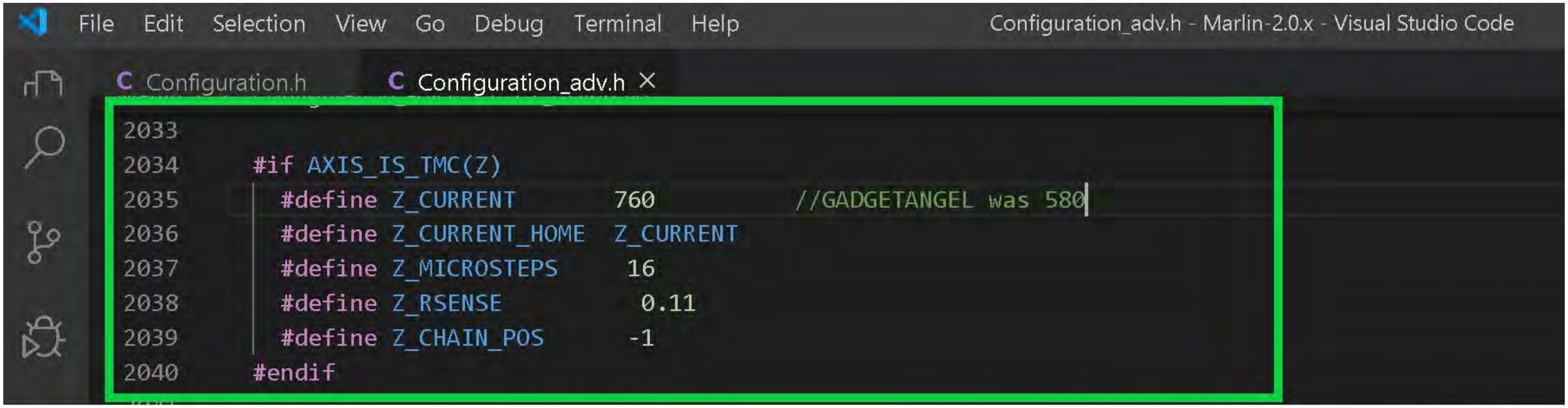
- Next you want to set your V_{ref} in the Marlin firmware for each axis that has the TMC2208 driver, as seen in the **GREEN** box below. I changed the "X_CURRENT" to be the calculated V_{ref} for my X-Axis, which is 760mV for an Ender 3. I changed the "Y_CURRENT" to be the calculated V_{ref} for my Y-Axis, which is 760mV on the Ender 3.
- Ensure "X_RSENSE" is set to 0.11. Ensure "Y_RSENSE" is set to 0.11.
- If you do not want to use V_{ref} as the value for "X_CURRENT" and/or "Y_CURRENT", you should use I_{RMS} instead. You find I_{RMS} by taking I_{MAX} and dividing it by 1.414 ($I_{RMS}=I_{MAX}/1.414$). You use 50% to 90% of the calculated I_{RMS} as the value for "X_CURRENT" and/or "Y_CURRENT".

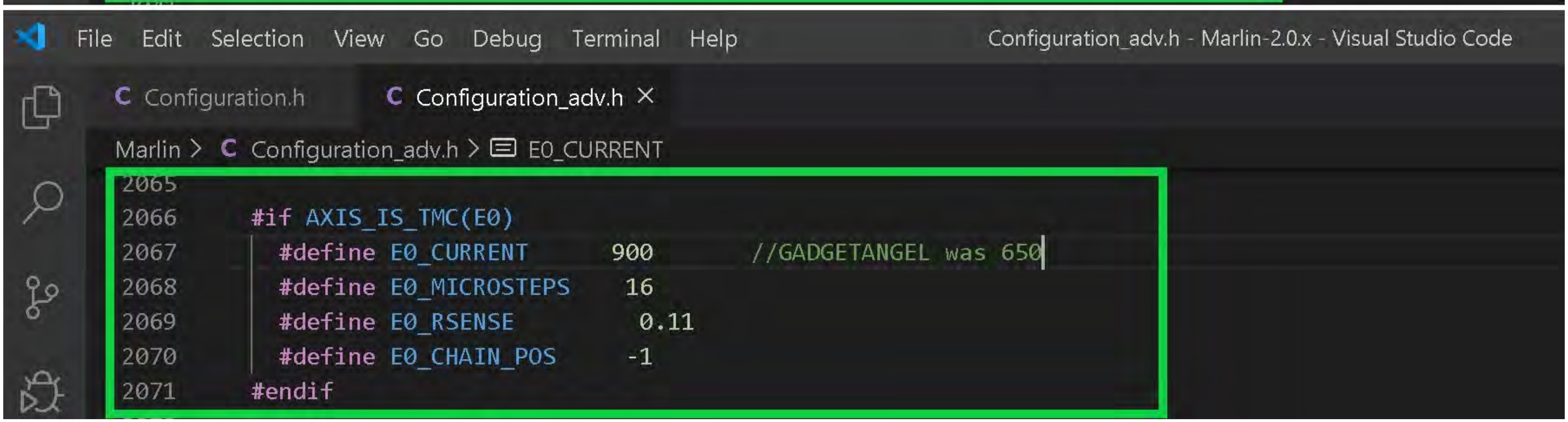


- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2208 V3.0 Drivers in UART Mode

- Now, I am setting the V_{ref} for Z-Axis and the extruder, as seen in the **GREEN** boxes below. I changed the "Z_CURRENT" to be the calculated V_{ref} for my Z-Axis, which is 760mV for an Ender 3. I changed the "E0_CURRENT" to be the calculated V_{ref} for my Extruder, which is 900mV on the Ender 3.
- Ensure "Z_RSENSE" is set to 0.11. Ensure "E0_RSENSE" is set to 0.11.
- If you do not want to use V_{ref} as the value for "Z_CURRENT" and/or "E0_CURRENT", you should use I_{RMS} instead. You find I_{RMS} by taking I_{MAX} and dividing it by 1.414 ($I_{RMS}=I_{MAX}/1.414$). You use 50% to 90% of the calculated I_{RMS} as the value for "Z_CURRENT" and/or "E0_CURRENT".

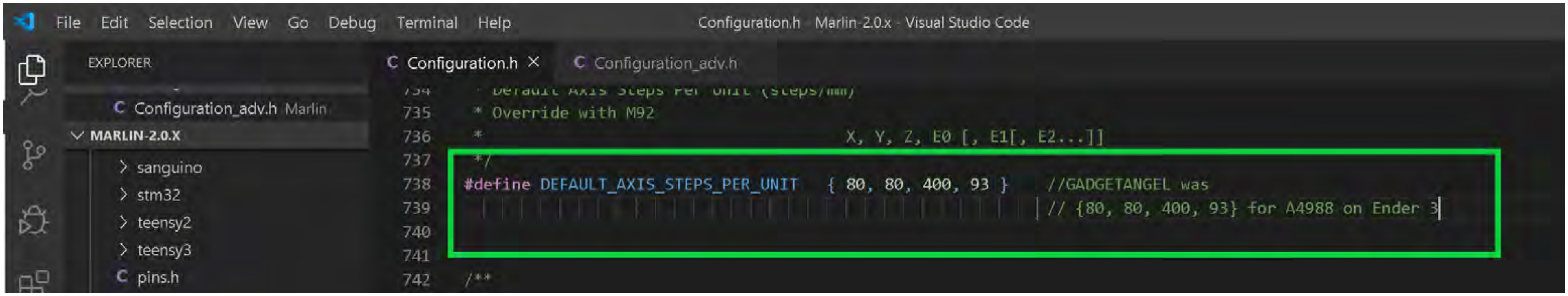




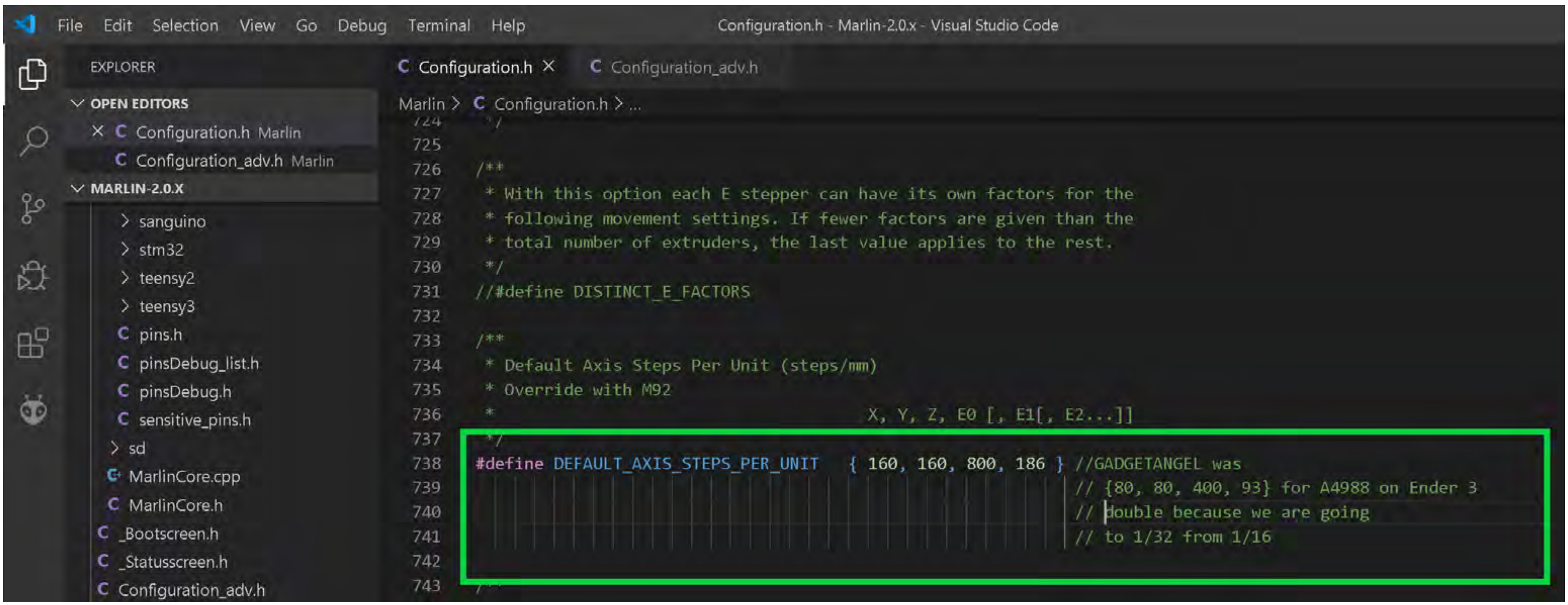
- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2208 V3.0 Drivers in UART Mode

- If you changed the "MICROSTEPS" for any of the axes then you will need to update "DEFAULT_AXIS_STEPS_PER_UNIT" to reflect your changes



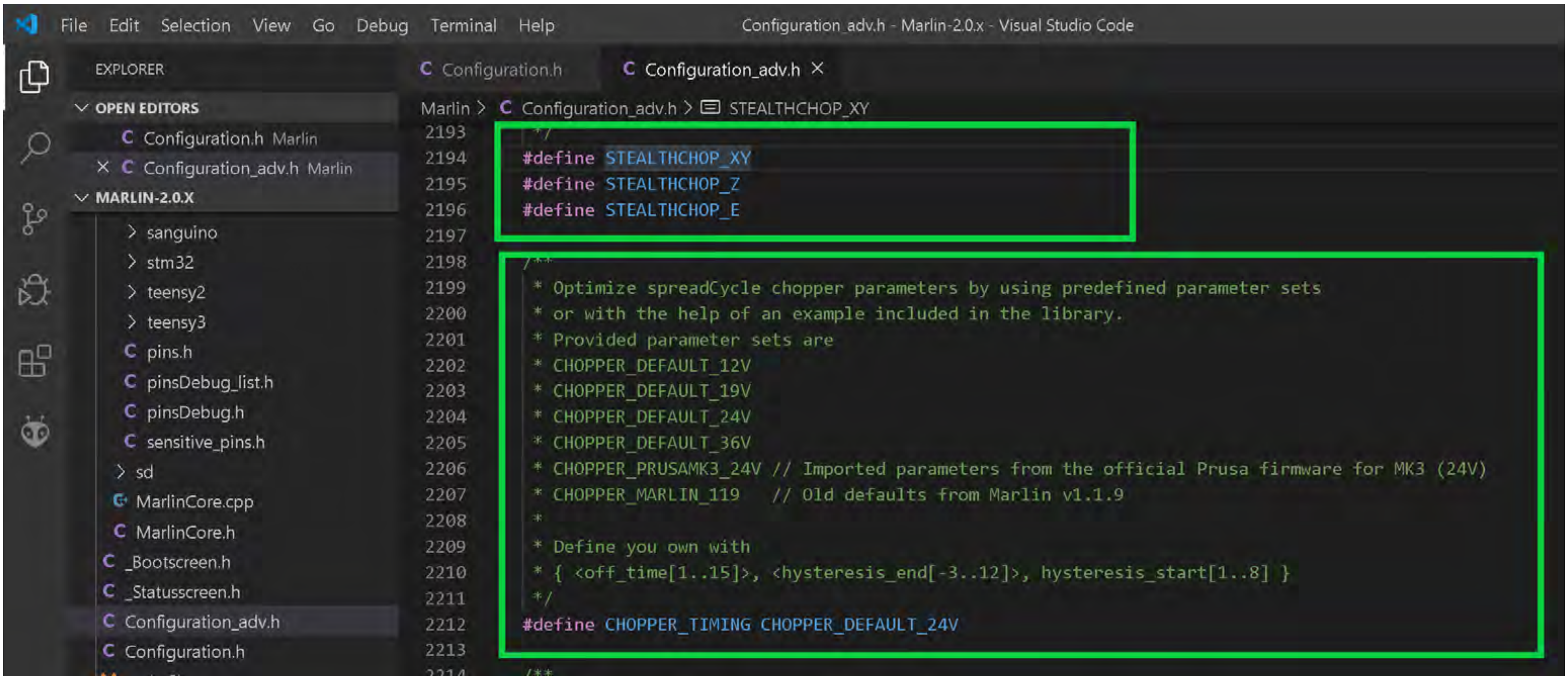
- FOR EAMPLE if you wanted to use 1/32 stepping instead of the default 1/16, you would be **doubling** your STEPS. Therefore, we must adjust our "DEFAULT_AXIS_STEPS_PER_UNIT" anytime our STEPS are NOT 1/16. So change "DEFAULT_AXIS_STEPS_PER_UNIT" to {160, 160, 800, 186}, as seen in the **GREEN** box below.



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2208 V3.0 Drivers in UART Mode

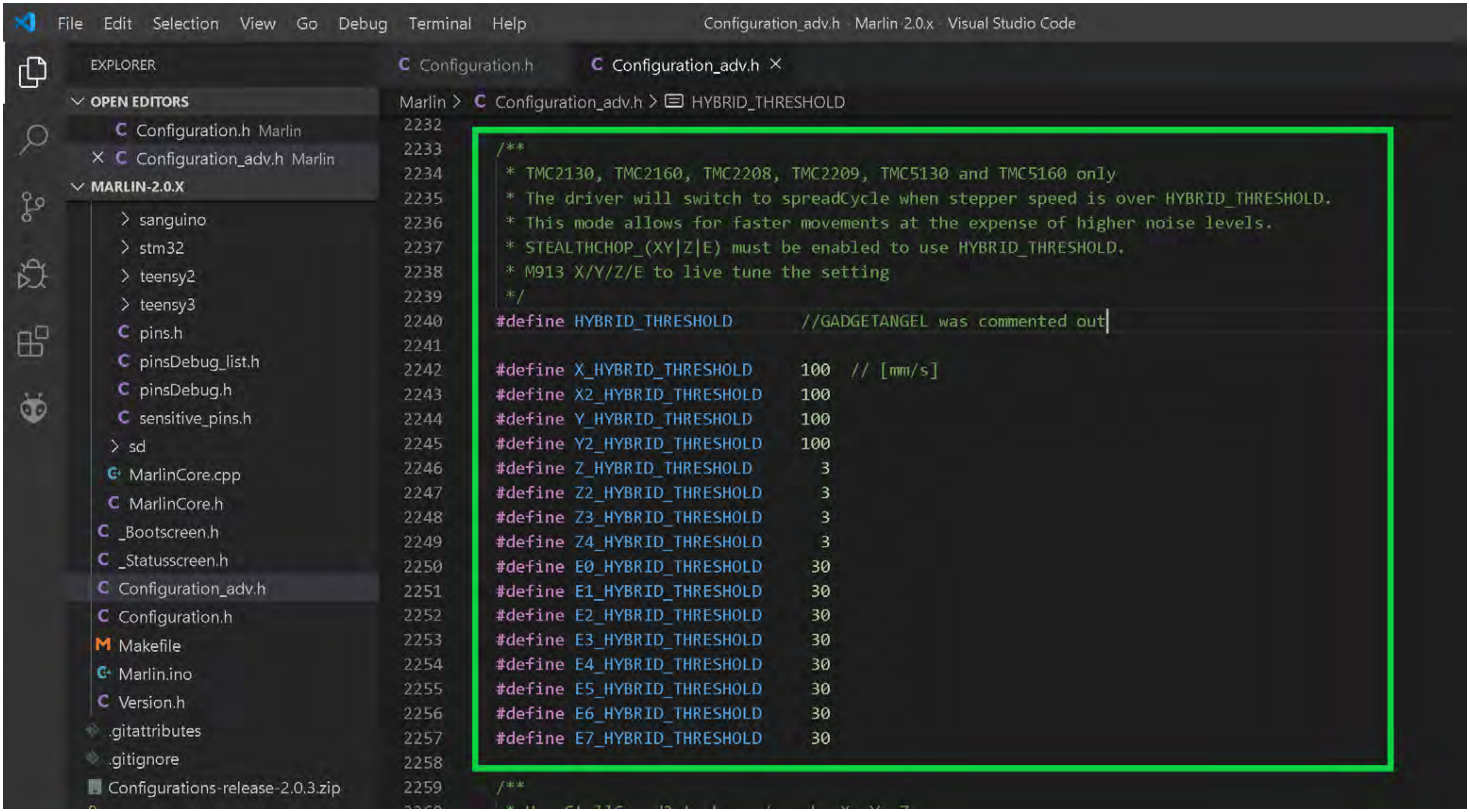
- By default stealthChop is enabled in the Marlin firmware. If you want spreadCycle ONLY then comment out the appropriate lines. I want stealthChop enabled so I want to make sure the lines are not commented out {"STEALTHCHOP_XY", "STEALTHCHOP_Z" and "STEALTHCHOP_E"}. You also want to check to see if the proper "CHOPPER_TIMING" is set for your printer. An Ender 3 is a 24VDC printer, my "CHOPPER_TIMING" is correct.



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2208 V3.0 Drivers in UART Mode

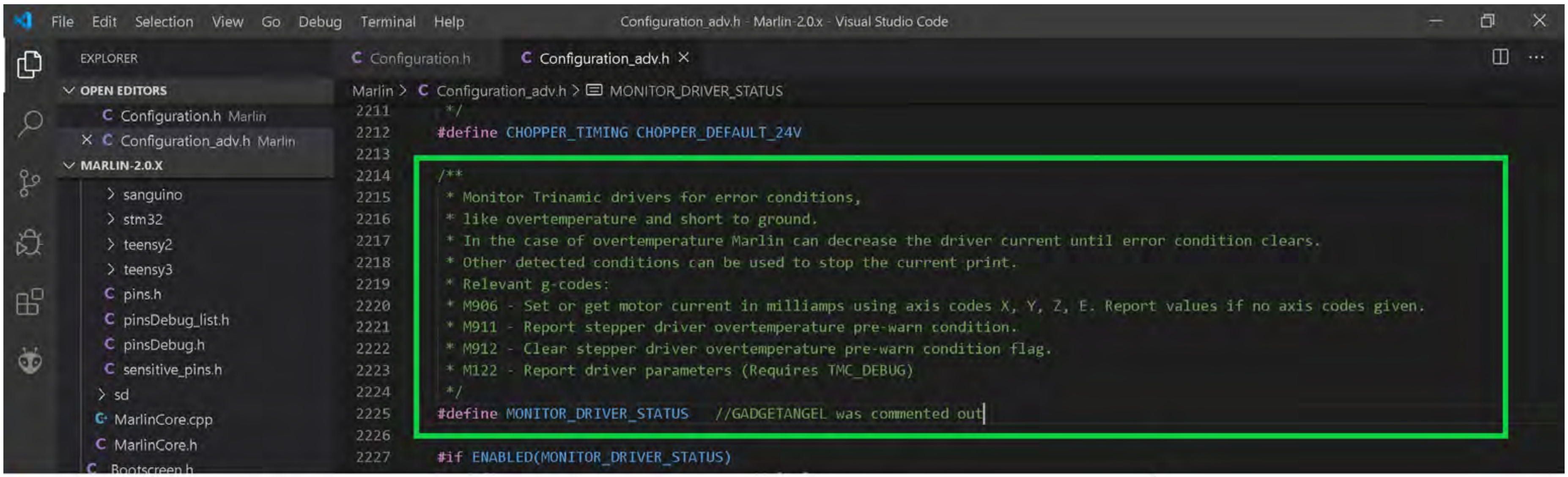
- Now you either enable "HYBRID_THRESHOLD" or disable it. By default, it is disabled. "HYBRID_THERSHOLD" allows the printer to change between stealthChop and spreadCycle dynamically depending on the print speed. I want "HYBRID_THRESHOLD" enabled so I need to remove the two leading "//", which uncomments the line in the Marlin firmware.

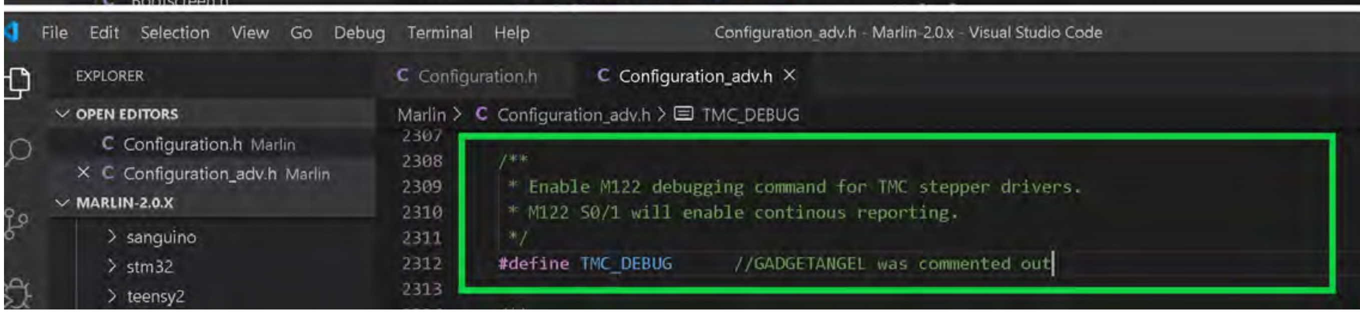


- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2208 V3.0 Drivers in UART Mode

- Now I want to enable some statements that allow me access to debugging the TMC drivers. I will uncomment "MONITOR_DRIVER_STATUS" and "TMC_DEBUG". "MONITOR_DRIVER_STATUS" will enable the following G-codes: M906, M911, and M912, "TMC_DEBUG" will enable the M122 G-code command. You can read about these from the comments in the firmware and in **Marlin's documentation located on-line.**

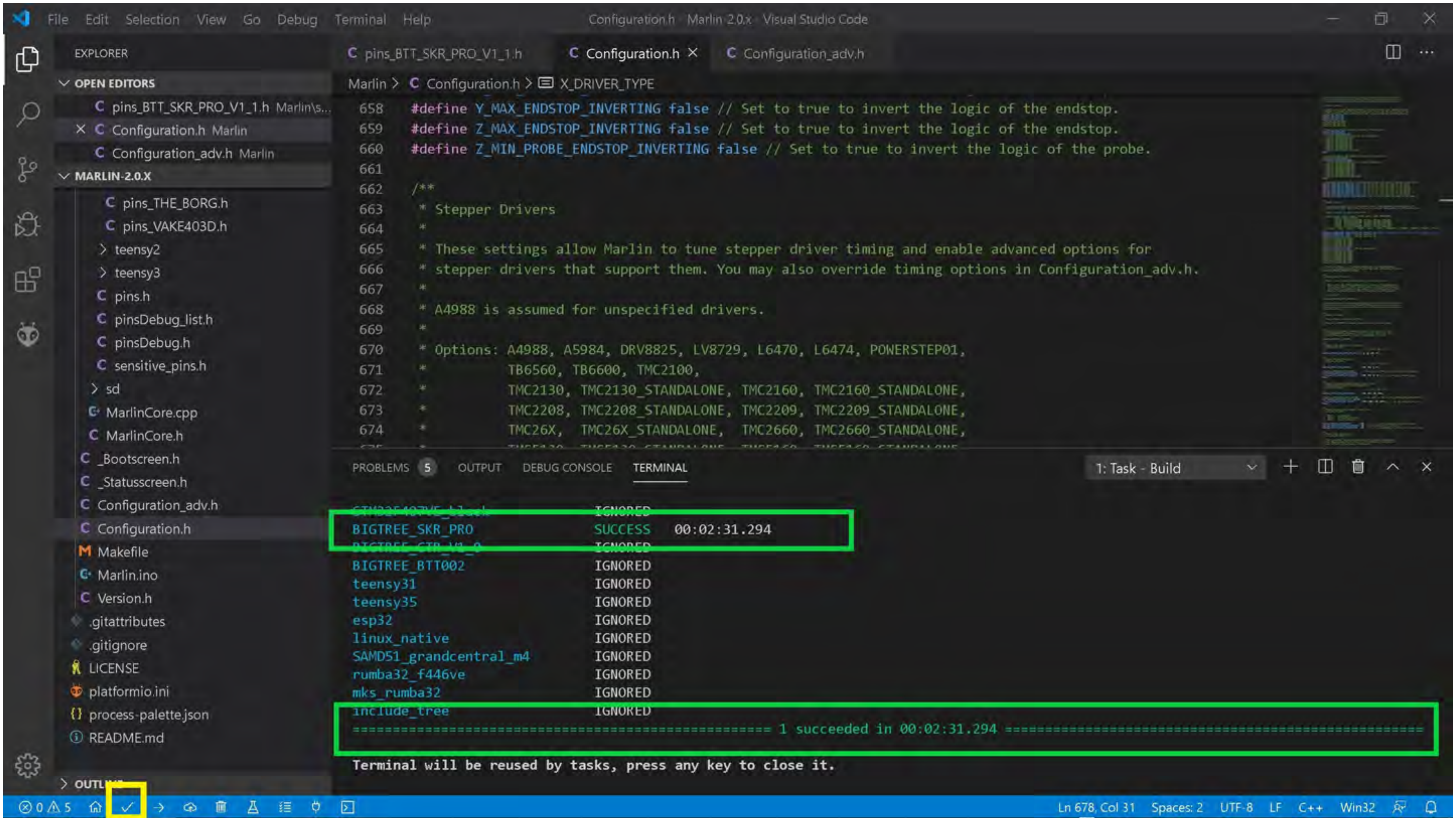




- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2208 V3.0 Drivers in UART Mode

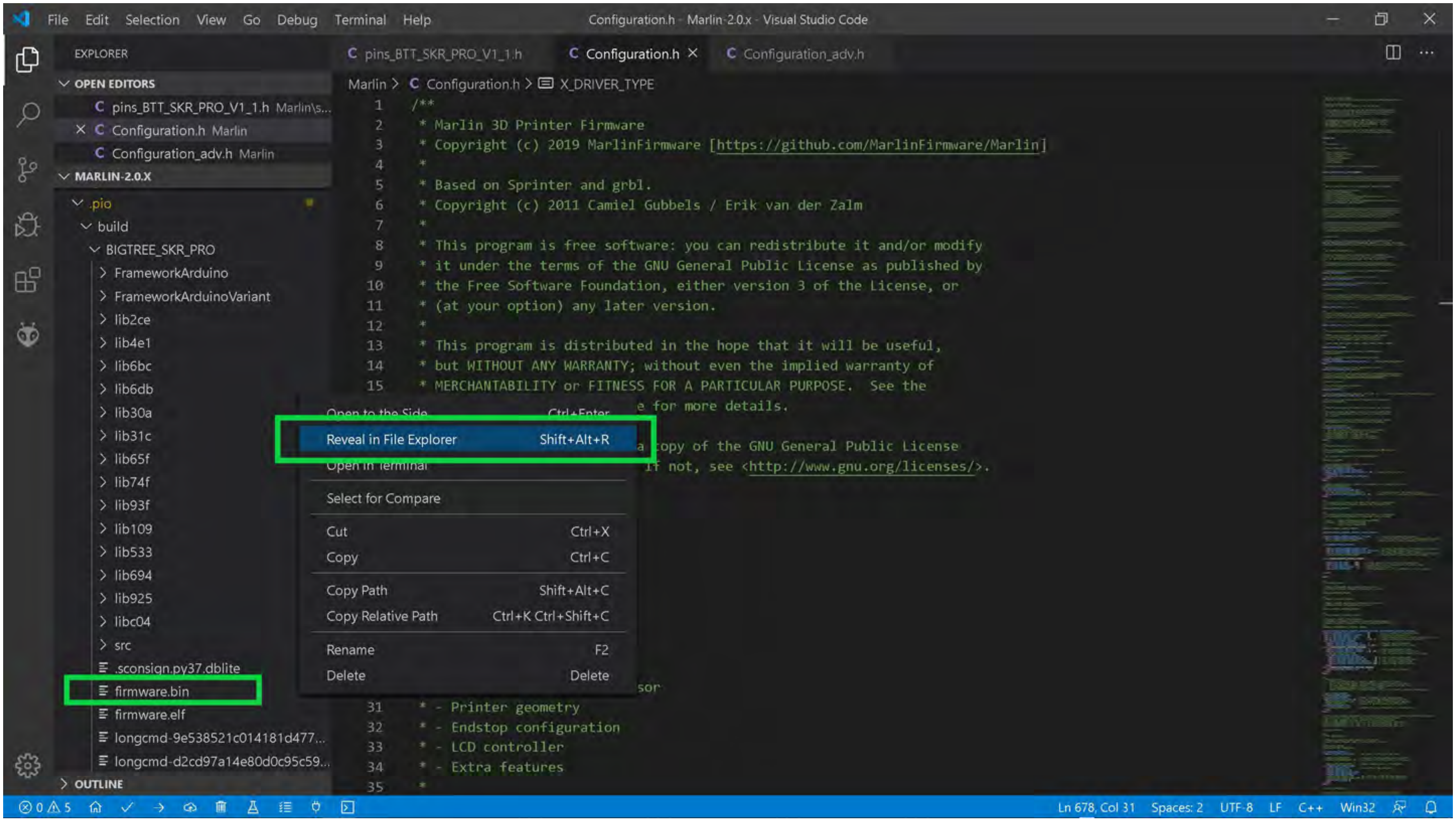
- The end of Marlin setup for BIQU TMC2208 V3.0 drivers in UART mode. Now, compile your code by clicking on the check mark, as seen in the **YELLOW** box below. Once the firmware has compiled successfully you will see the following messages, as shown in the **GREEN** boxes below.



- For the SKR PRO V1.1 board you must save the "firmware.bin" file to the micro SD card then place the micro SD card into the micro SD card reader of the SKR PRO V1.1 board and turn on the power to your printer for the NEW firmware to be automatically applied to the SKR PRO MCU. To locate the "firmware.bin" file, and learn how to transfer it to your micro SD card go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2208 V3.0 Drivers in UART Mode

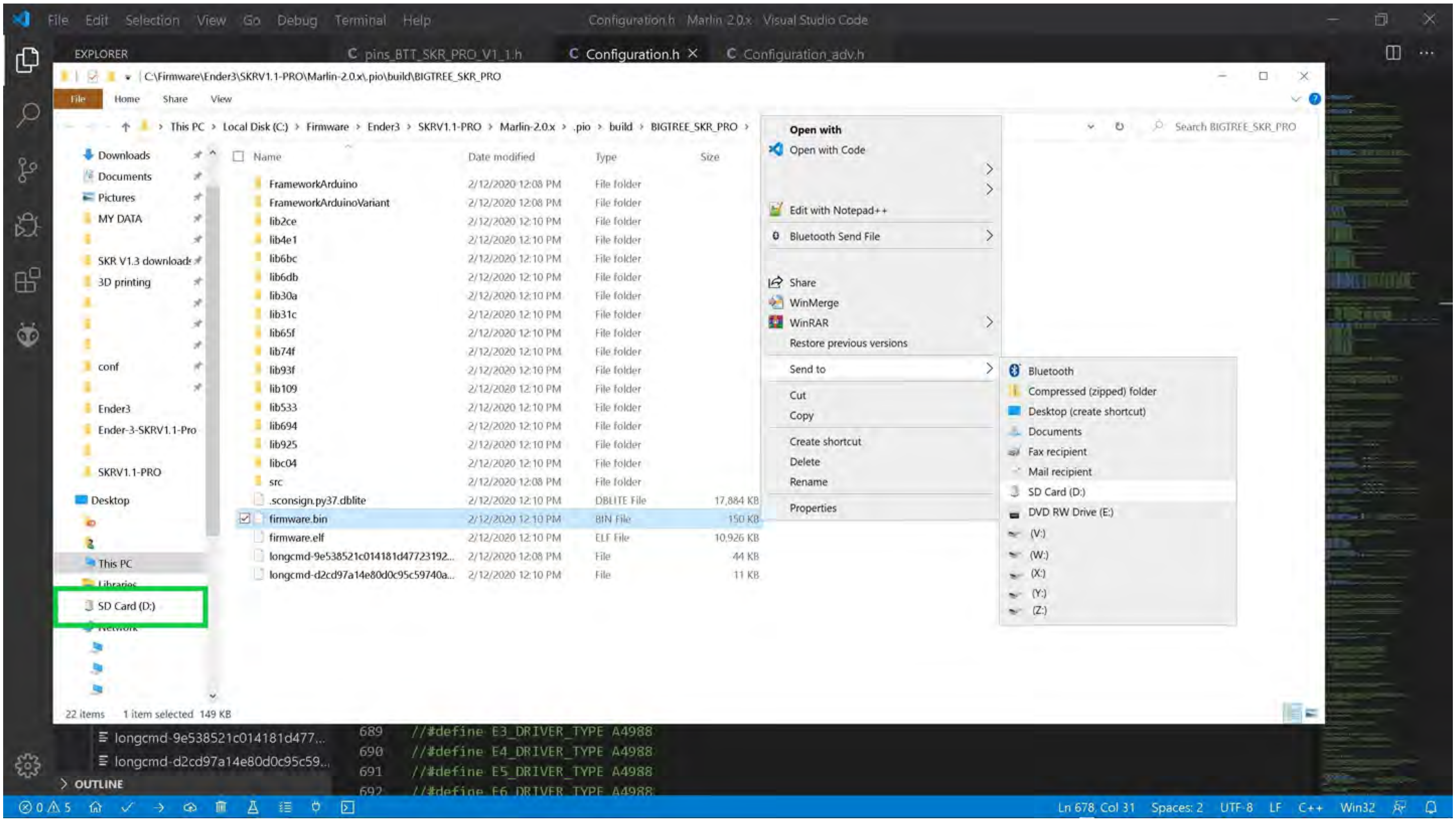
- The "firmware.bin" file can be found by using the left side of the VS code window and going to folder Marlin-2.0.X/.pio/build/ BIGTREE_SKR_PRO/ and right clicking on the "firmware.bin" file name to bring up the content menu. Select "Reveal in File Explorer" to have your Window's machine open a file explorer window.



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2208 V3.0 Drivers in UART Mode

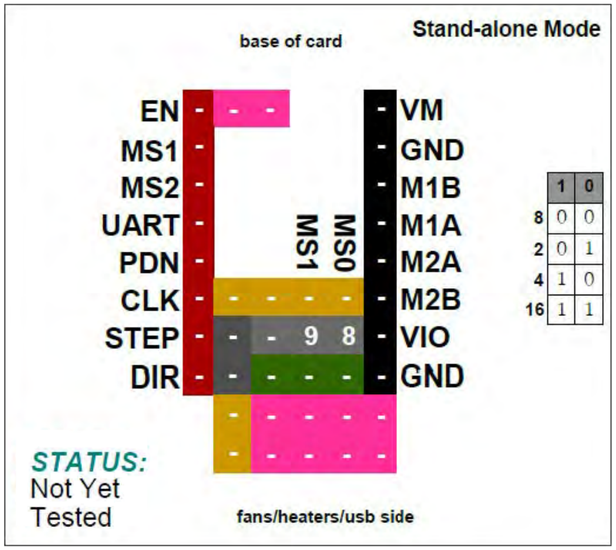
- Ensure your micro SD card, which came with your SKR PRO V1.1 board, with your SKR PRO V1.1 board is in a micro-to-SD-card adapter and plugged into your Window's SD card reader.
    - From the file explorer window right click on the "firmware.bin" file name and select "Send to" from the content menu. For my computer "D:" is my SD card reader. For your computer it may be a different drive letter. The "firmware.bin" file will be written to the SD card and renamed to "FIRMWARE.CUR".



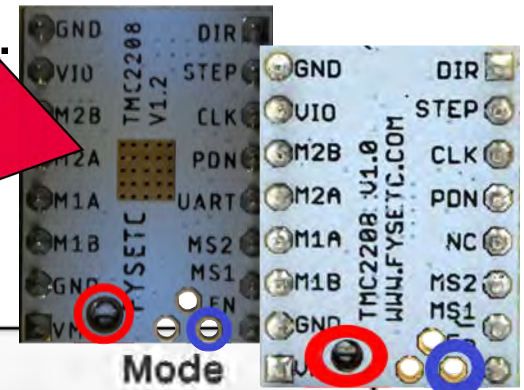
- Right click on the SD card reader, as seen in the **GREEN** box above, and select "Eject" from the context menu. Take the micro SD card out of the adapter and place it into the micro SD card reader of the SKR PRO V1.1 board. Turn on the power to the printer (i.e. the SKR PRO V1.1 board) and the "FIRMWARE.CUR" file will be uploaded to the board. After the upload the firmware file will be renamed to "firmware.bin" and stored on the micro SD card.

FYSETC TMC2208 V1.2

Stand-alone Mode



NOTE: Use the potentiometer (POT) on the top of the board, as shown in **RED**; or use the board's "V_{ref} Test point" location, as shown in **BLUE**, to set your V_{ref}. See **Appendix A** for instructions on how to set the V_{ref} on a driver board.



Note: Use 90% of the calculated V_{ref} when tuning the stepper driver board.

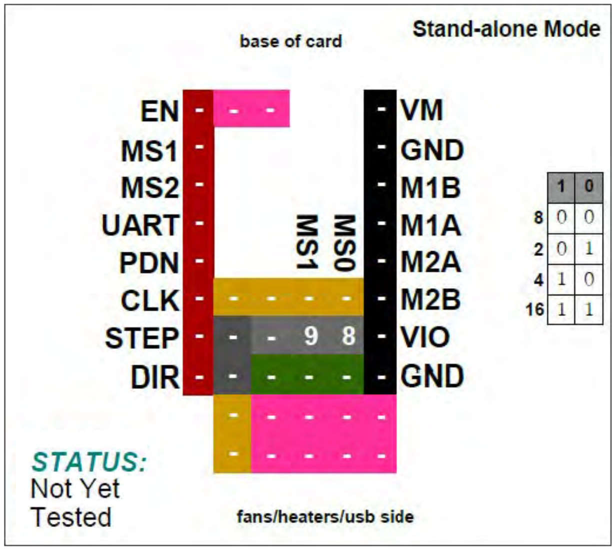
Driver Chip	MS1	MS0	Steps	Interpolation	Mode
FYSETC TMC2208 Stand Alone Mode Maximum 16 Subdivision 35V DC 2A (peak)	GND	GND	1 / 8	1 / 256	stealthChop
	GND	VIO	1 / 2	1 / 256	stealthChop
	VIO	GND	1 / 4	1 / 256	stealthChop
	VIO	VIO	1 / 16	1 / 256	stealthChop

Driving Current Calculation Formula		
R_S(Typical Sense Resistor)= 0.11Ω	$I_{MAX}=V_{ref}*0.9286$ See Appendix B #3. Use 50% to 90% as shown below: $I_{MAX}=(V_{ref}*0.9286)*0.90$	$V_{ref}=I_{MAX}*1.0769$ See Appendix B #3. Use 50% to 90% as shown below: $V_{ref}=(I_{MAX}*1.0769)*0.90$

NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

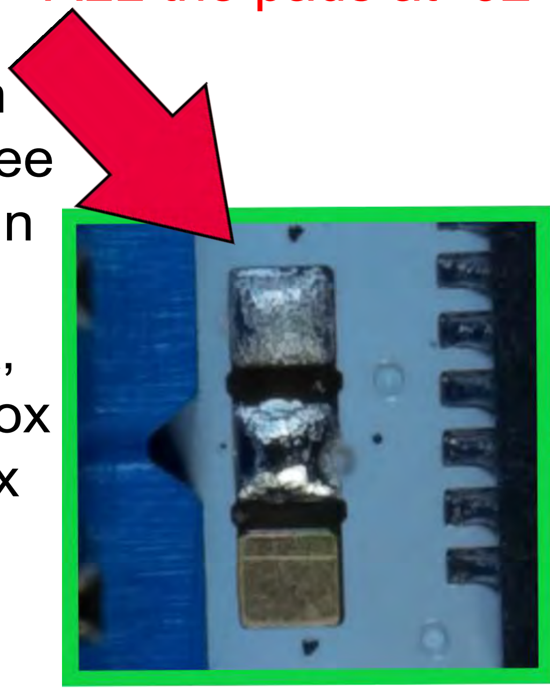
FYSETC TMC2208 V1.2

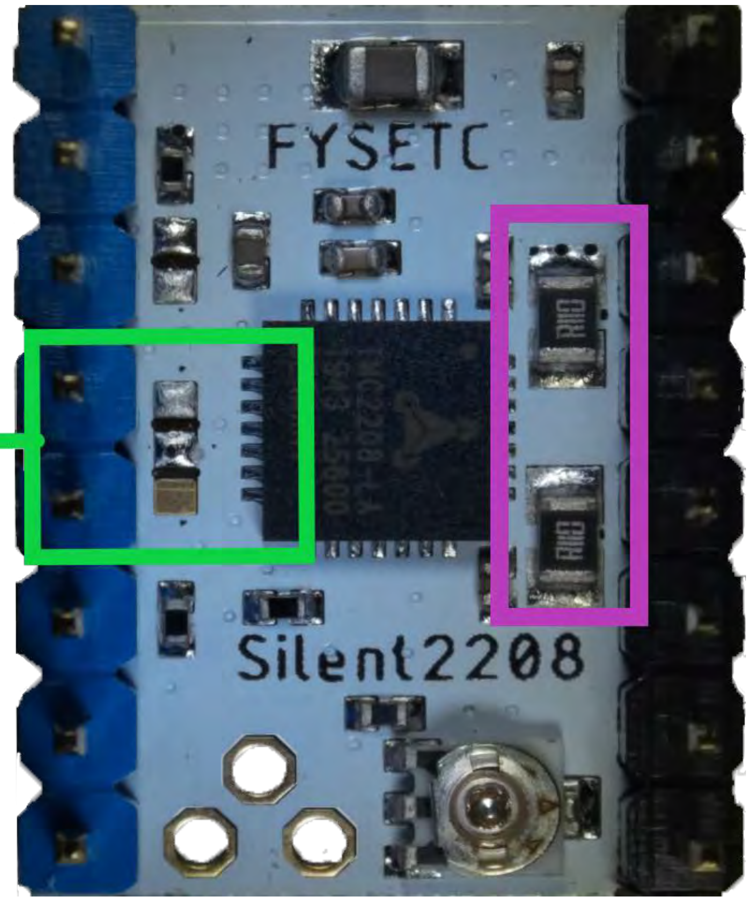
Stand-alone Mode



Note: To determine if your driver board is in UART mode, check the bottom of the driver board for three pads, located in the same position as shown in the picture below. There could be two or three pads located at this position (let's call it the "J2" position). To have the driver board in STANDALONE MODE, ALL the pads at "J2" MUST NOT be connected.

Again, a gap MUST be in place between on all (three or two) "J2" pads to obtain Standalone Mode for the FYSETC TMC2208 V1.X, as seen in the **GREEN** box below. The **PURPLE** box shows the location of the current sense resistors (R_S).





MOST FYSETC TMC2208 boards are sold as V1.2 driver boards. The V1.2 driver board might be setup to be in UART mode. BUT the V1.0 driver board is setup to be in STANDALONE mode by default. So please check your boards to ensure they are in the correct mode!!

FYSETC TMC2208 V1.2 for Standalone Mode

FYSETC TMC2208 V1.2

Stand-alone Mode

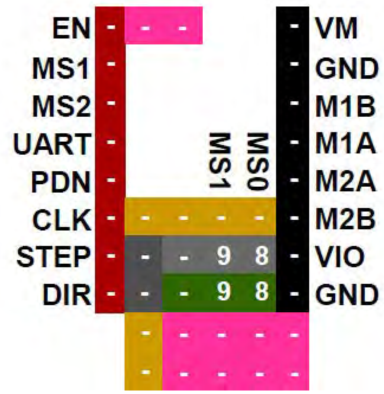
Stand-alone Mode

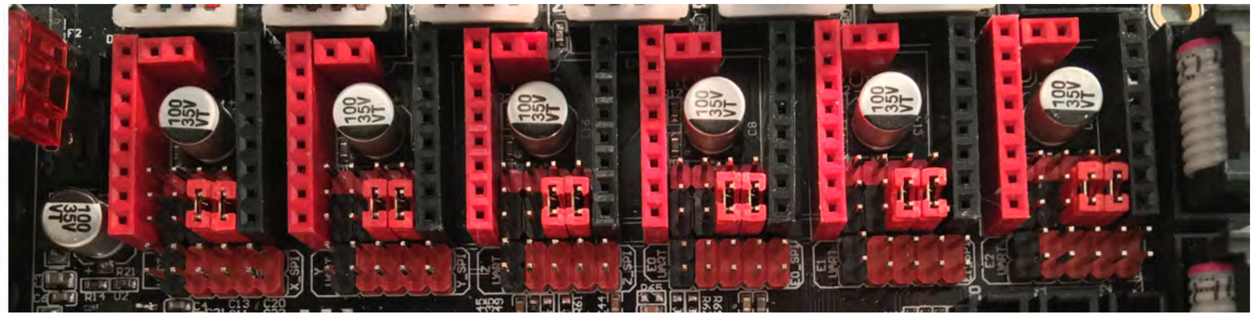
Stand-alone Mode

1 / 8

Interpolation: **1/256**

StealthChop



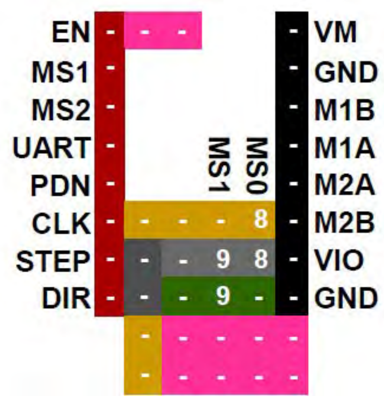


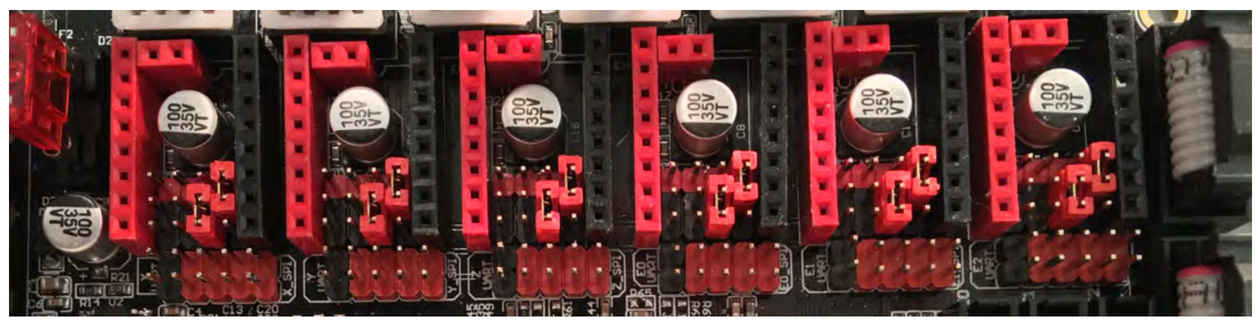
Stand-alone Mode

1 / 2

Interpolation: **1/256**

StealthChop





NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

FYSETC TMC2208 V1.2

Stand-alone Mode

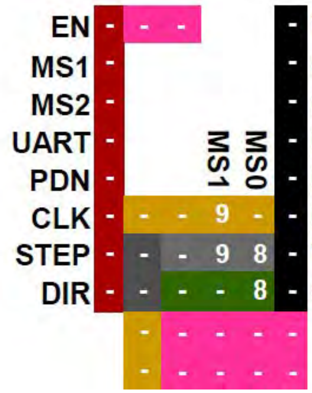
Stand-alone Mode

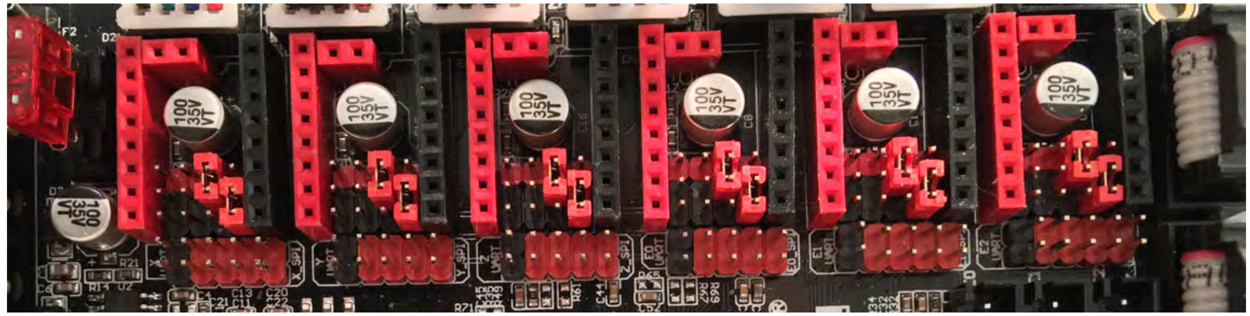
Stand-alone Mode

1 / 4

Interpolation: **1/256**

StealthChop



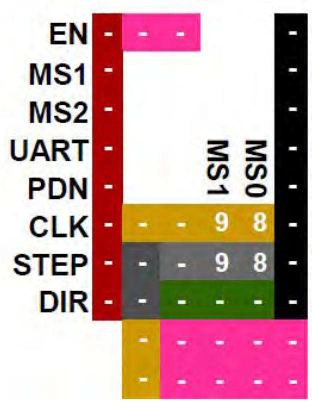


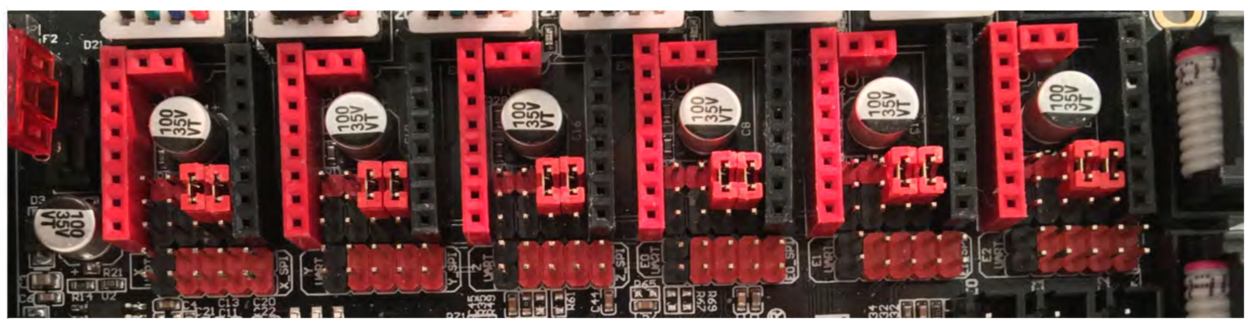
Stand-alone Mode

1 / 16

Interpolation: **1/256**

StealthChop

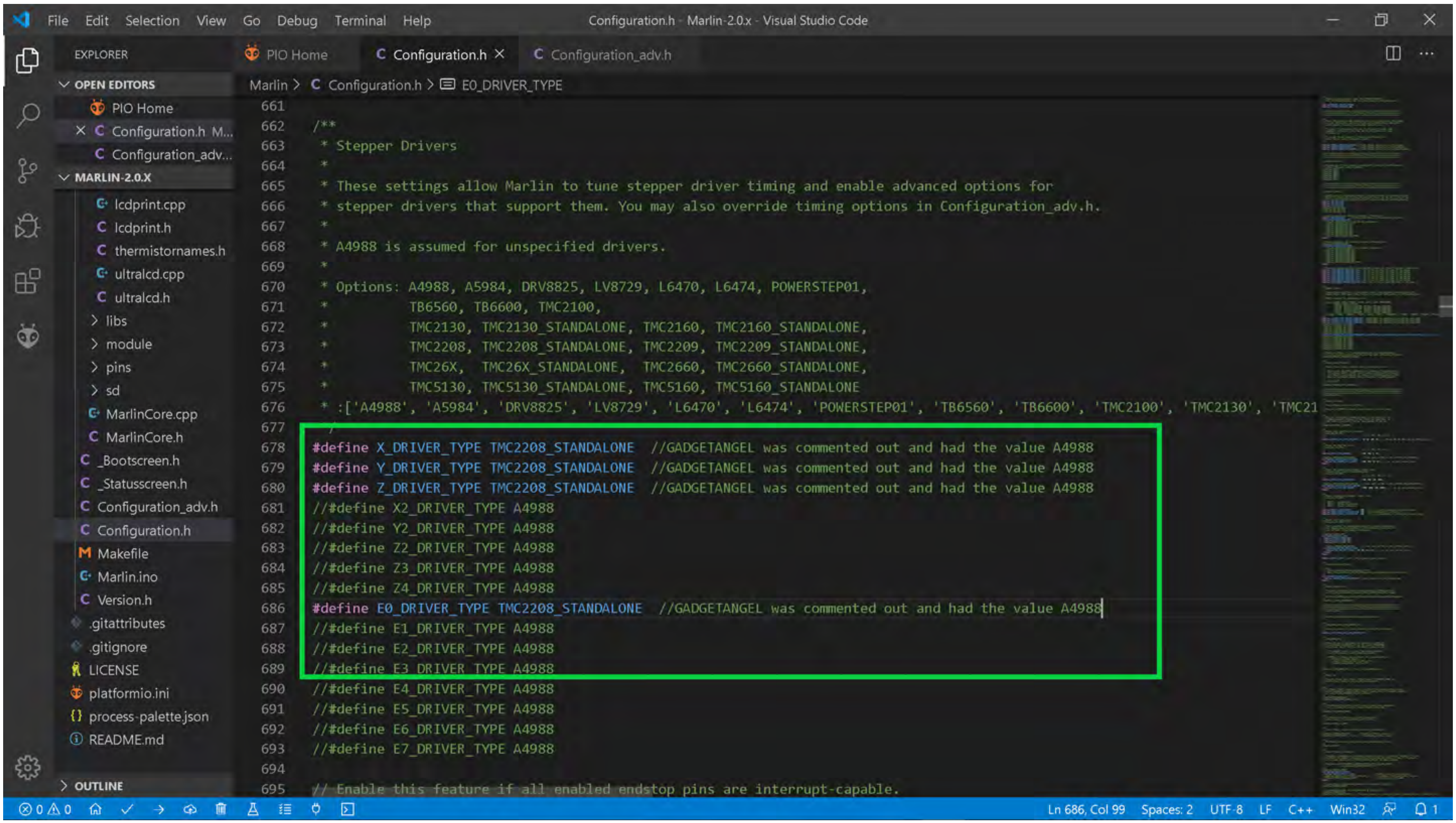




The (latest release of) Marlin Setup for FYSETC TMC2208 V1.2 Drivers in Stand-alone Mode

NOTE: **Go to Appendix C,** and then come back here for the changes to Marlin for FYSETC TMC2208 V1.2 stepper motor drivers in stand-alone mode.

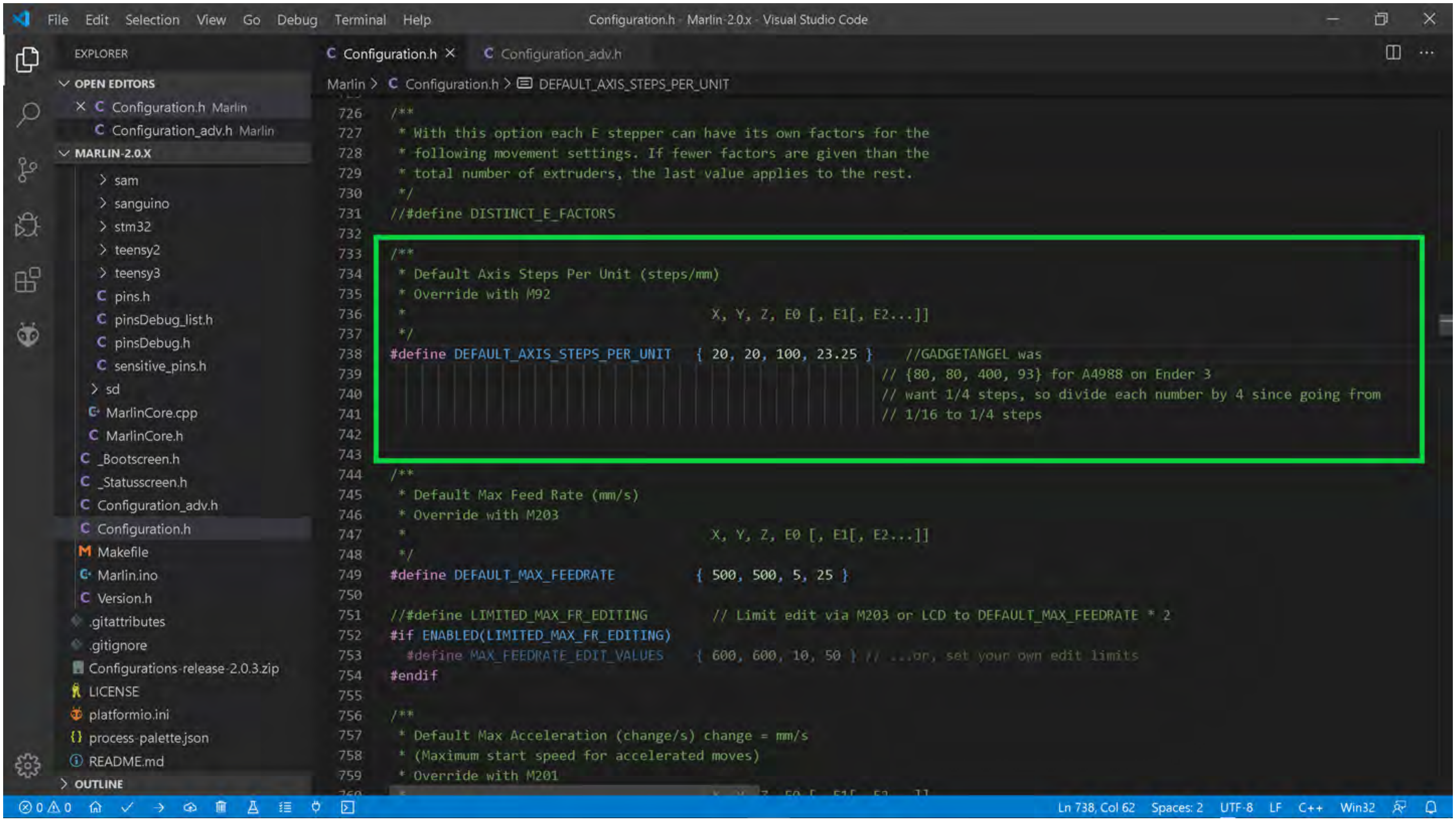
- Change the stepper motor drivers so that Marlin knows you are using TMC2208 drivers in stand-alone mode. Change one line for each axis and one for each extruder you will be using. See the picture below for an example of how to use TMC2208 drivers in stand-alone mode. When two "//" appear at the beginning of a line that means that line is commented out. To un-comment a line just remove the leading two forward slashes ("//").



- Go to the next page.

The (latest release of) Marlin Setup for FYSETC TMC2208 V1.2 Drivers in Stand-alone Mode

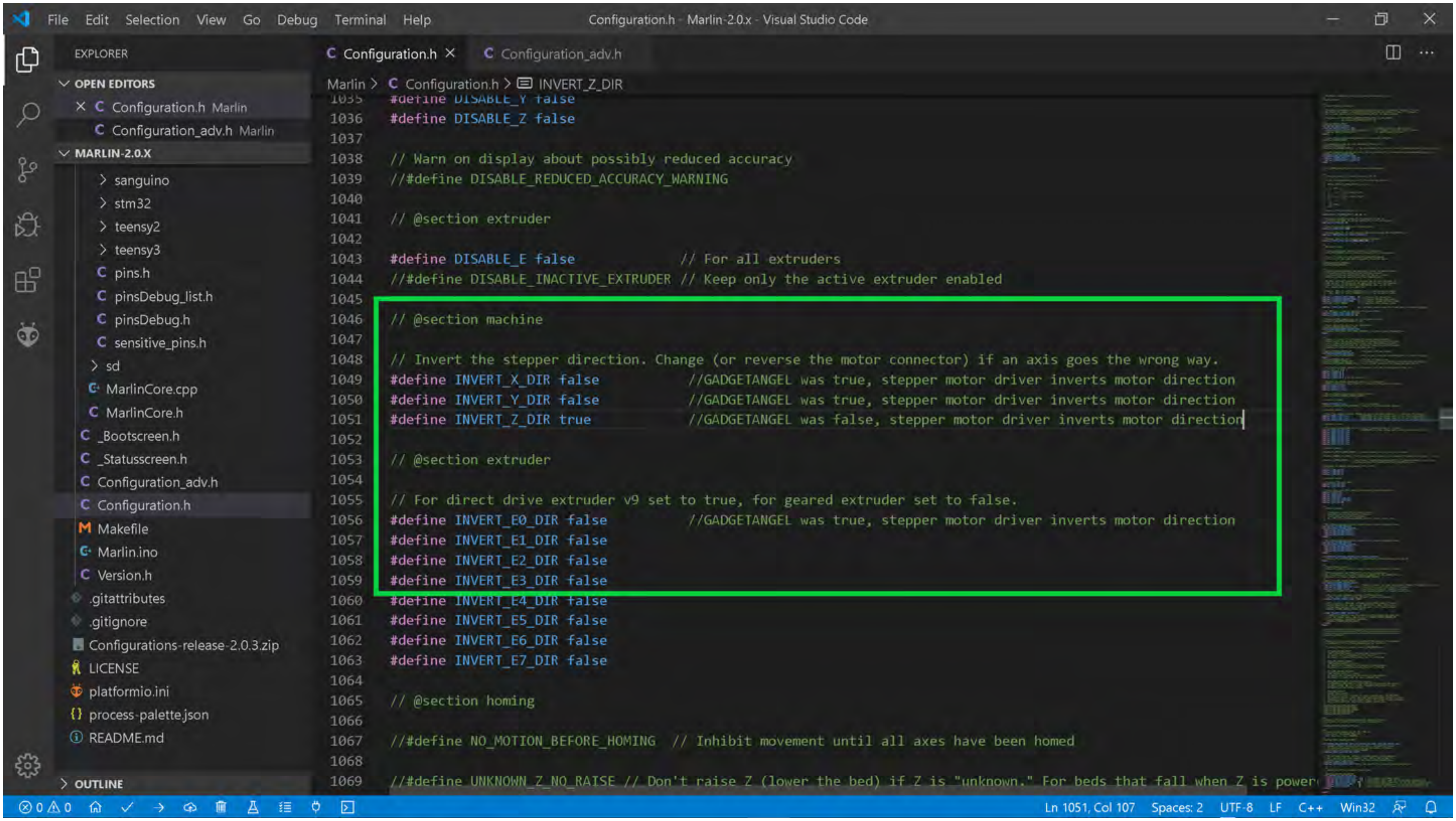
- Since I desire to use 1/4 stepping, and we are changing from A4988 stepper motor drivers on the Ender 3 to FYSETC TMC2208 stepper motor drivers for each axis and the extruder stepper motor driver, we will be going from 1/16 stepping to 1/4 stepping. So we are cutting our STEPS by one quarter. Therefore, we must adjust our "DEFAULT_AXIS_STEPS_PER_UNIT" anytime our STEPS are NOT 1/16. So change "DEFAULT_AXIS_STEPS_PER_UNIT" to {20, 20, 100, 23.25}, as seen in the **GREEN** box below.



- Go to the next page.

The (latest release of) Marlin Setup for FYSETC TMC2208 V1.2 Drivers in Stand-alone Mode

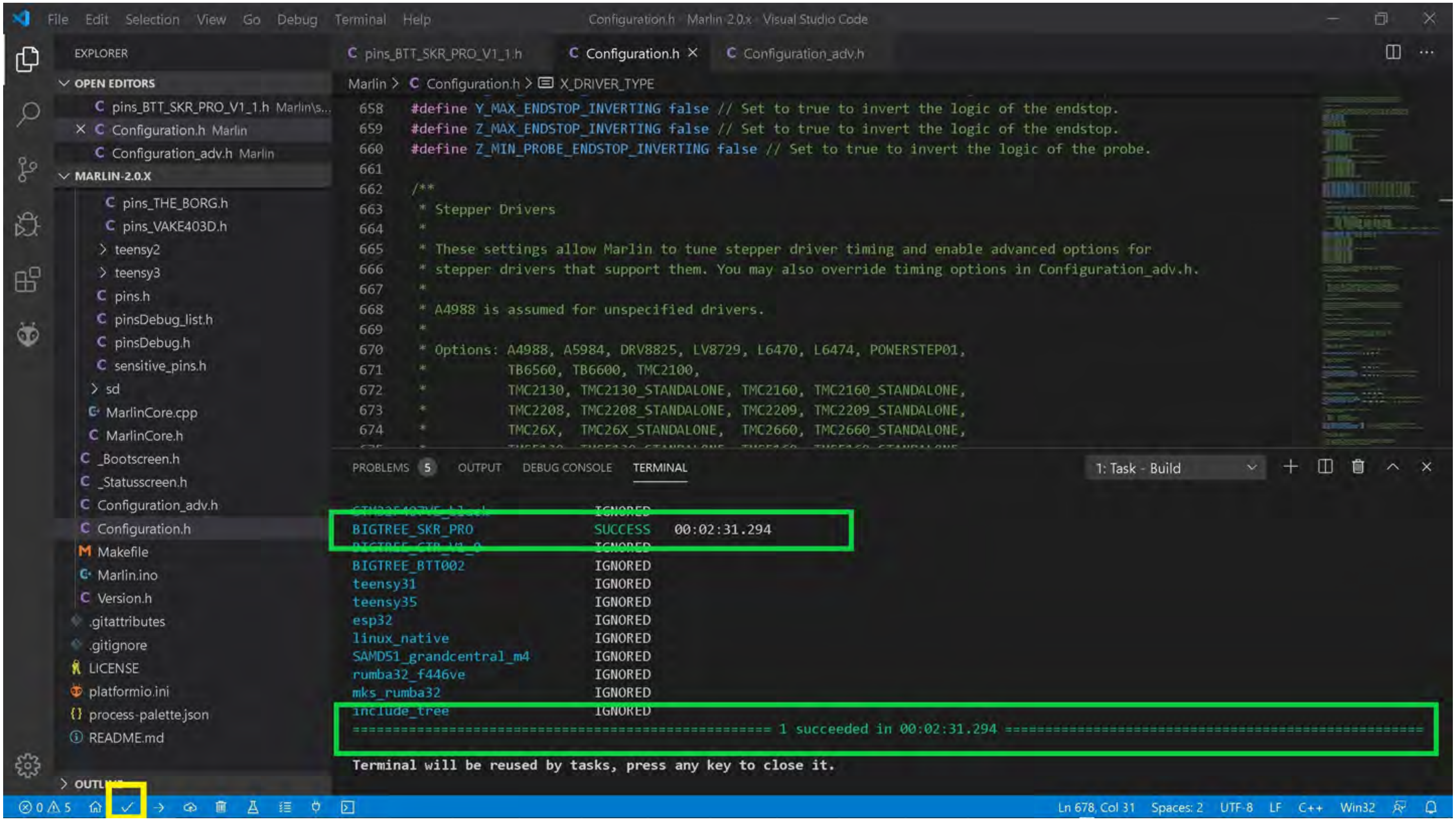
- Since the A4988 driver is what my Ender 3 used, but, now I want to use TMC2208 drivers, I must invert the stepper motor direction because the TMC2208 driver will turn the motors in the opposite direction than the A4988 driver's motor direction. So if the axis' setting you will be using the TMC2208 driver on was "true" change it to "false", as shown in the **GREEN** box below. If the setting was "false", now set it to "true", as shown in the **GREEN** box below



- Go to the next page.

The (latest release of) Marlin Setup for FYSETC TMC2208 V1.2 Drivers in Stand-alone Mode

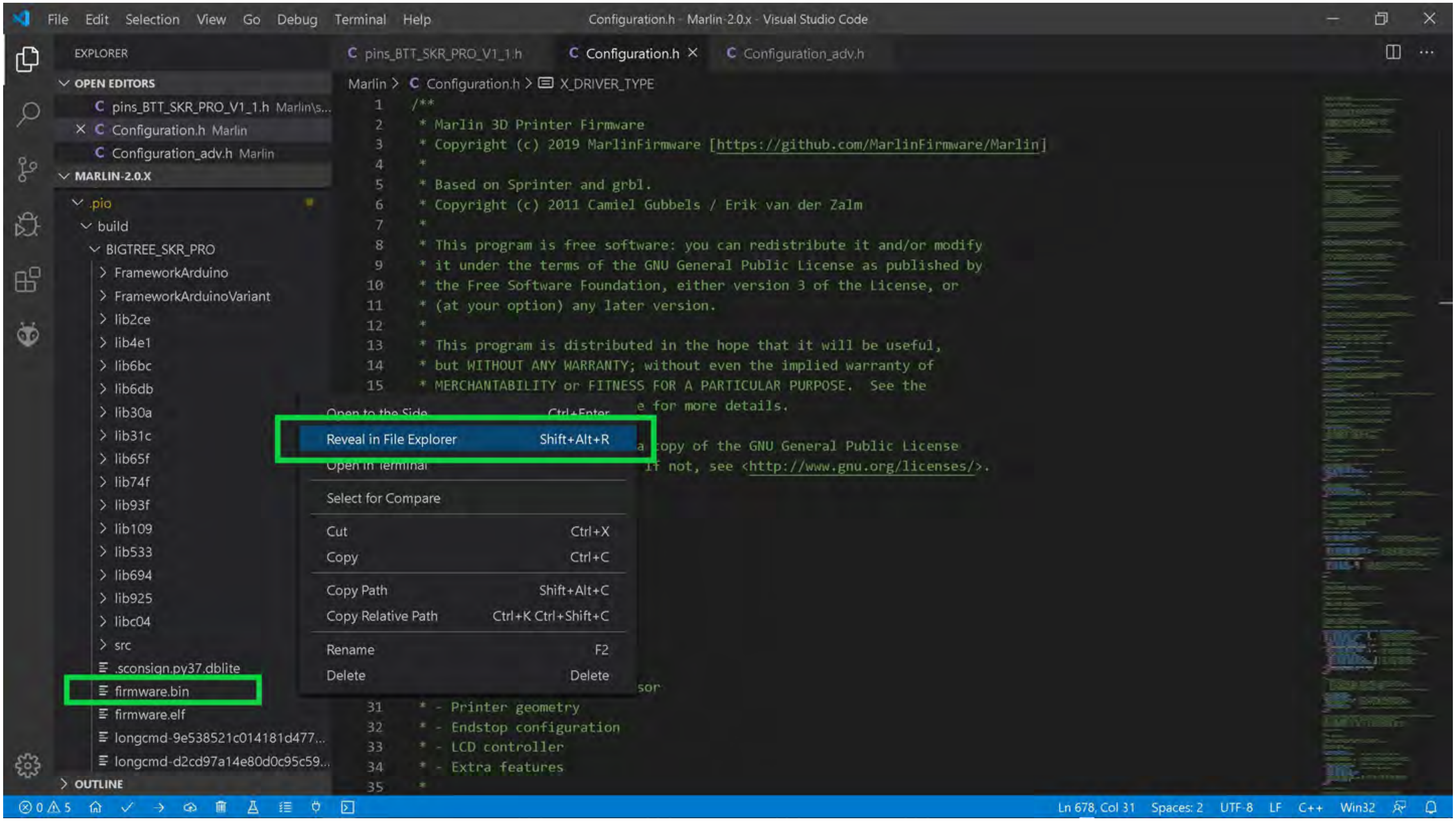
- The end of Marlin setup for FYSETC TMC2208 V1.2 drivers in stand-alone mode. Now, compile your code by clicking on the check mark, as seen in the **YELLOW** box below. Once the firmware has compiled successfully you will see the following messages, as shown in the **GREEN** boxes below.



- For the SKR PRO V1.1 board you must save the "firmware.bin" file to the micro SD card then place the micro SD card into the micro SD card reader of the SKR PRO V1.1 board and turn on the power to your printer for the NEW firmware to be automatically applied to the SKR PRO MCU. To locate the "firmware.bin" file, and learn how to transfer it to your micro SD card go to the next page.

The (latest release of) Marlin Setup for FYSETC TMC2208 V1.2 Drivers in Stand-alone Mode

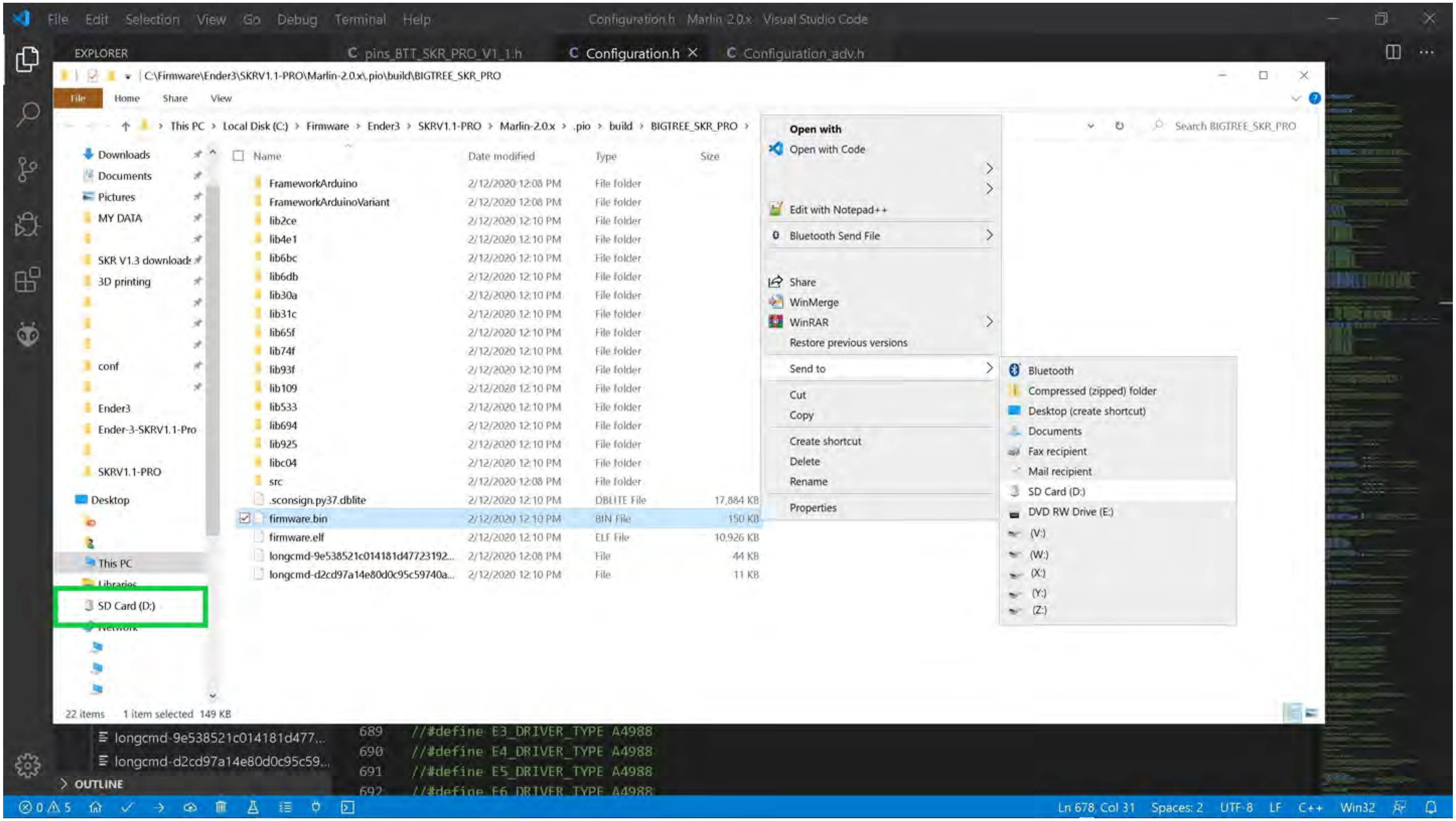
- The "firmware.bin" file can be found by using the left side of the VS code window and going to folder Marlin-2.0.X/.pio/build/ BIGTREE_SKR_PRO/ and right clicking on the "firmware.bin" file name to bring up the content menu. Select "Reveal in File Explorer" to have your Window's machine open a file explorer window.



- Go to the next page.

The (latest release of) Marlin Setup for FYSETC TMC2208 V1.2 Drivers in Stand-alone Mode

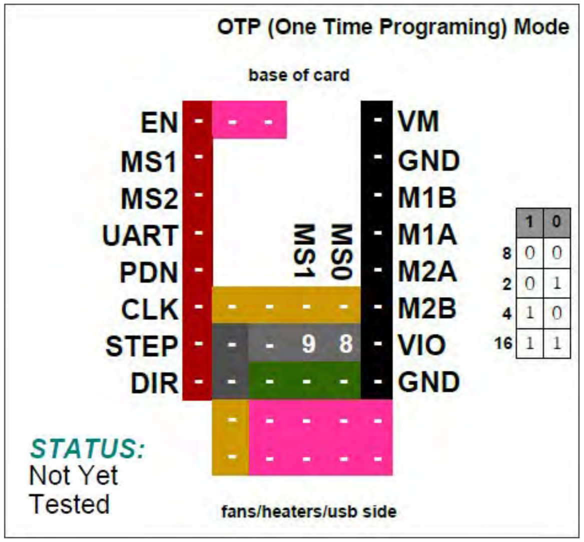
- Ensure your micro SD card, which came with your SKR PRO V1.1 board, with your SKR PRO V1.1 board is in a micro-to-SD-card adapter and plugged into your Window's SD card reader.
    - From the file explorer window right click on the "firmware.bin" file name and select "Send to" from the content menu. For my computer "D:" is my SD card reader. For your computer it may be a different drive letter. The "firmware.bin" file will be written to the SD card and renamed to "FIRMWARE.CUR".



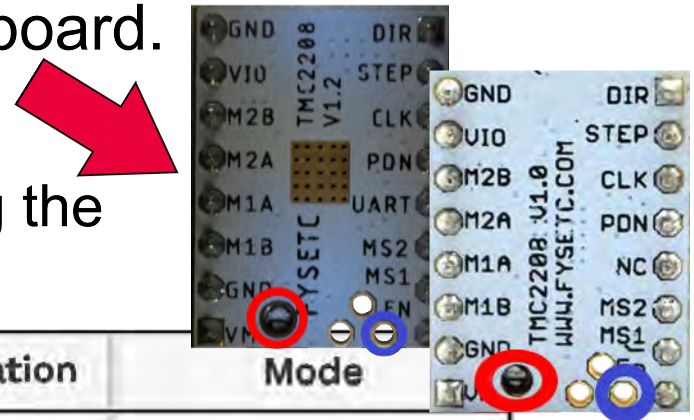
- Right click on the SD card reader, as seen in the **GREEN** box above, and select "Eject" from the context menu. Take the micro SD card out of the adapter and place it into the micro SD card reader of the SKR PRO V1.1 board. Turn on the power to the printer (i.e. the SKR PRO V1.1 board) and the "FIRMWARE.CUR" file will be uploaded to the board. After the upload the firmware file will be renamed to "firmware.bin" and stored on the micro SD card.

FYSETC TMC2208 V1.2

One Time Programming (OTP) Mode

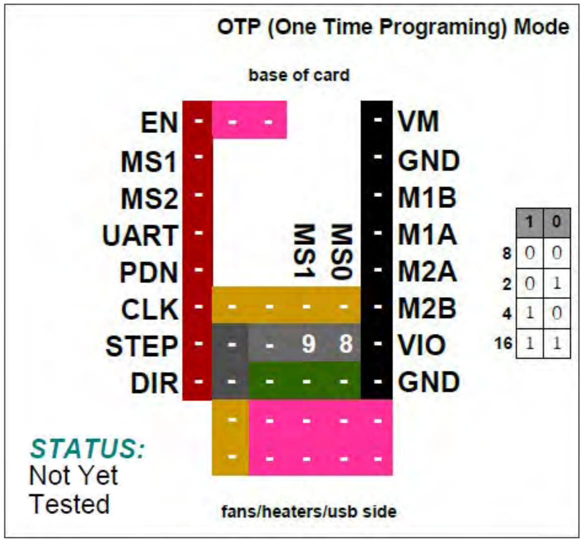


NOTE: Use the potentiometer (POT) on the top of the board, as shown in **RED**; or use the board's "V_{ref} Test point" location, as shown in **BLUE**, to set your V_{ref}. See **Appendix A** for instructions on how to set the V_{ref} on a driver board.



Note: Use 90% of the calculated V_{ref} when tuning the stepper driver board.

Driver Chip	MS1	MS0	Steps	Interpolation	Mode
FYSETC **TMC2208** OTP Mode, Maximum 16 Subdivision, 35V DC, 2A (peak)	GND	GND	1 / 8	1 / 256	spreadCycle
	GND	VIO	1 / 2	1 / 256	spreadCycle
	VIO	GND	1 / 4	1 / 256	spreadCycle
	VIO	VIO	1 / 16	1 / 256	spreadCycle
Driving Current Calculation Formula R_S(Typical Sense Resistor)= 0.11Ω	$I_{MAX}=V_{ref}*0.9286$ See Appendix B #3. Use 50% to 90% as shown below: $I_{MAX}=(V_{ref}*0.9286)*0.90$			$V_{ref}=I_{MAX}*1.0769$ See Appendix B #3. Use 50% to 90% as shown below: $V_{ref}=(I_{MAX}*1.0769)*0.90$	



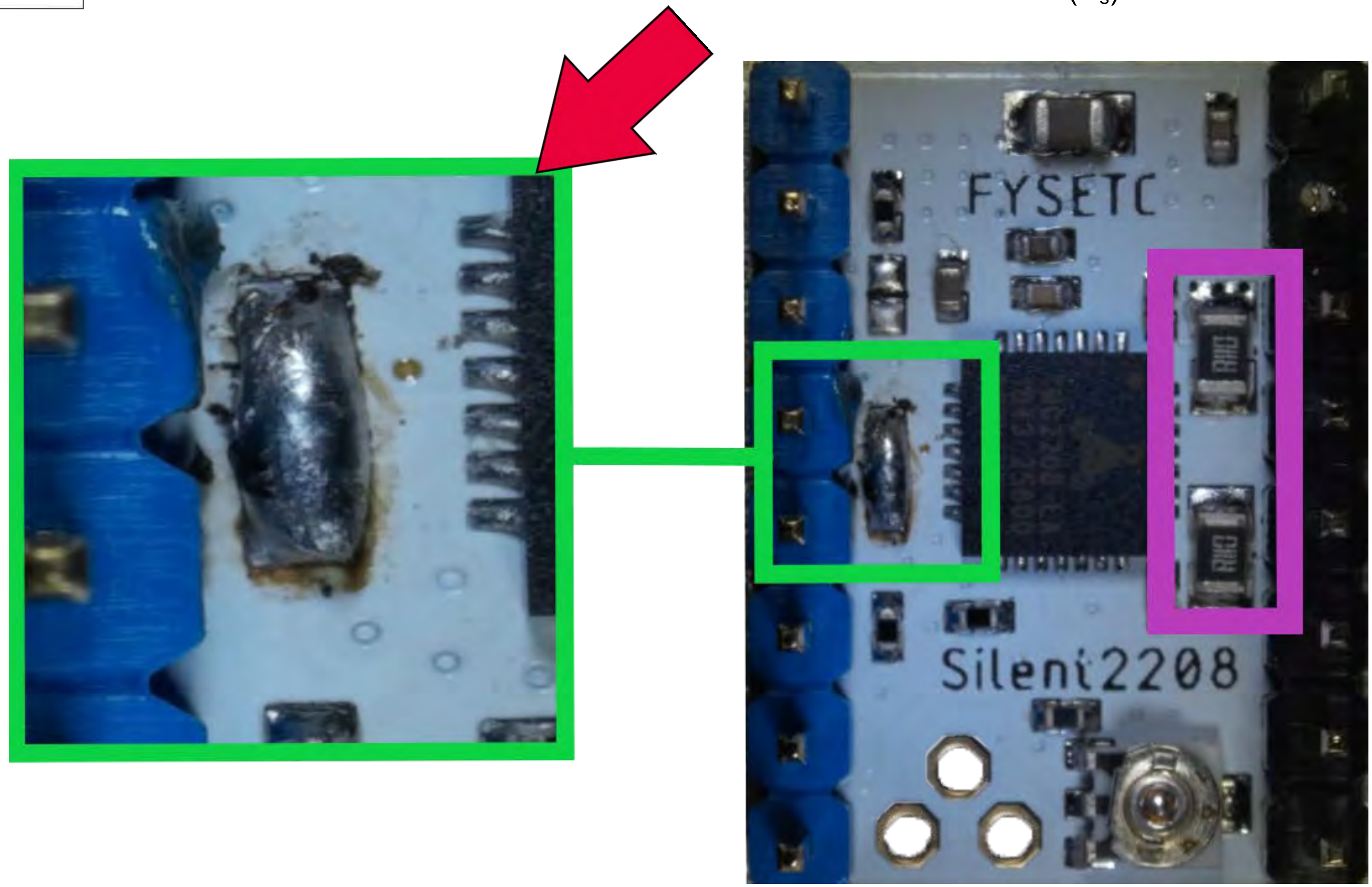
FYSETC TMC2208 V1.2

One Time Programming (OTP) Mode

NOTE: Stand-alone Mode by default uses StealthChop, if you want SpreadCycle, you MUST use OTP mode. Here are the directions for running the TMC220x Configurator: https://wiki.fysetc.com/TMC2208/#to-run-the-program. See TMC220x Configurator for One-Time-Programming Information: TMC220x Configurator.

Important: To place FYSETC TMC2208 V1.0 or V1.2 into OTP Mode you must solder all adjacent pads together on the bottom of the driver board. As an example, the picture (V1.2) below shows all the pads soldered together, as shown in **GREEN**. To do One-Time-Programming (OTP), the TMC2208 must be placed in UART mode to program it. The **PURPLE** box shows the location of the current sense resistors (R_s).

MOST FYSETC TMC2208 boards are sold as V1.2 driver boards. The V1.2 driver board might be setup to be in UART mode. BUT the V1.0 driver board is setup to be in STANDALONE mode by default. So please check your boards to ensure they are in the correct mode you desire! To be able to program the Chip for OTP mode, the FYSETC TMC2208 board **MUST be set in UART (all three pads bridged together)!**



FYSETC TMC2208 V1.2

One Time Programming (OTP) Mode

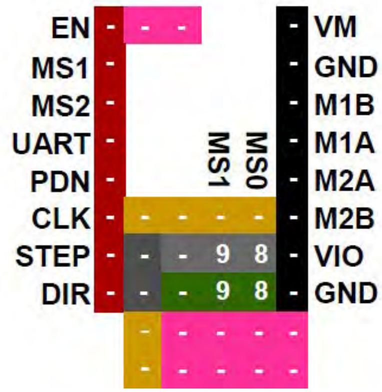
OTP (One Time Programing) Mode

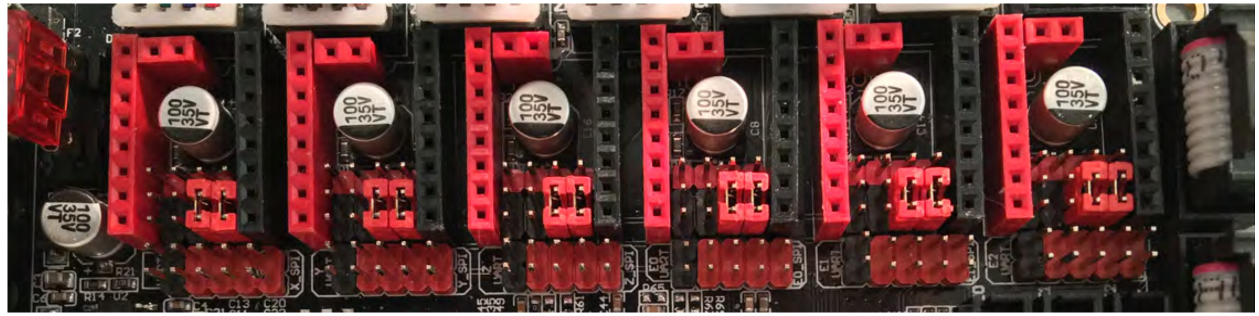
OTP

1 / 8

Interpolation: **1/256**

SpreadCycle



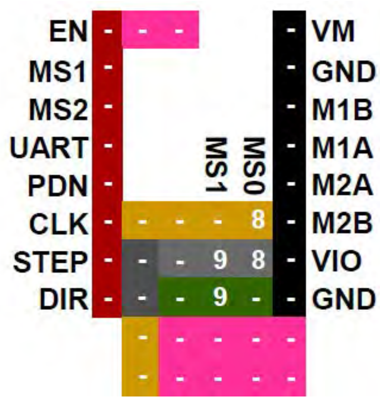


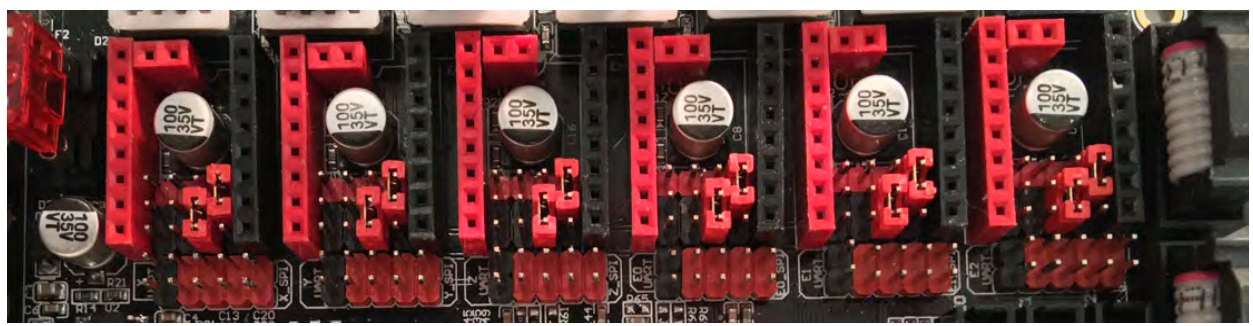
OTP

1 / 2

Interpolation: **1/256**

SpreadCycle





FYSETC TMC2208 V1.2

One Time Programming (OTP) Mode

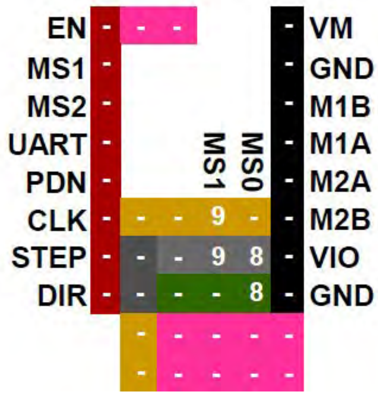
OTP (One Time Programing) Mode

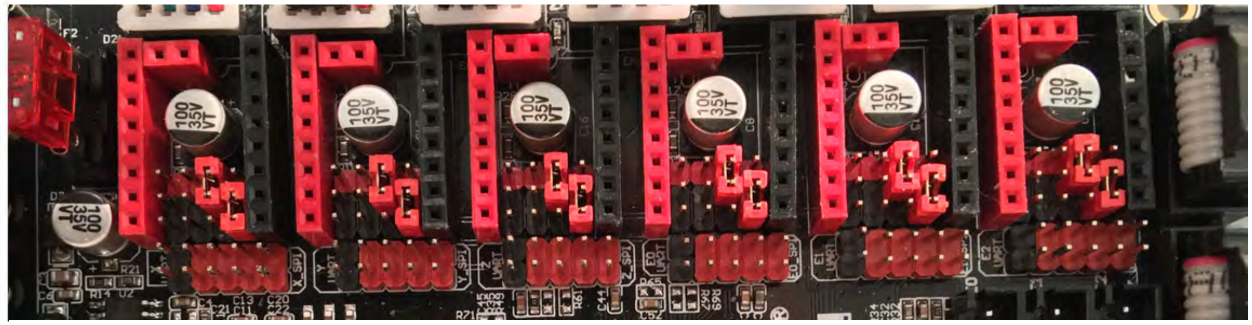
OTP

1 / 4

Interpolation: **1/256**

SpreadCycle



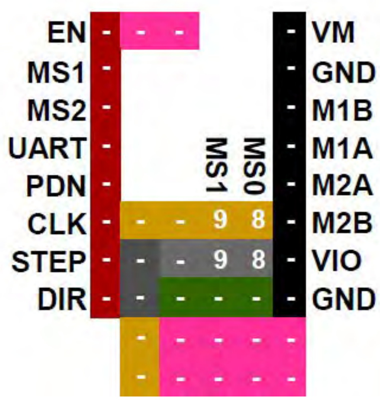


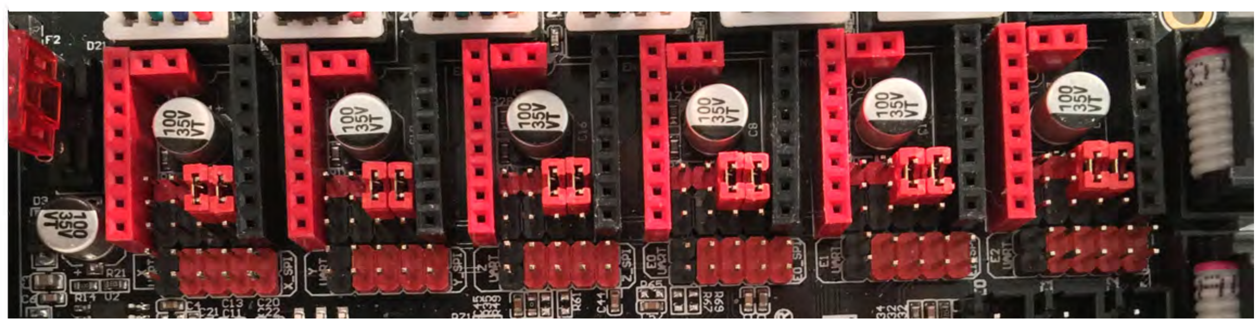
OTP

1 / 16

Interpolation: **1/256**

SpreadCycle



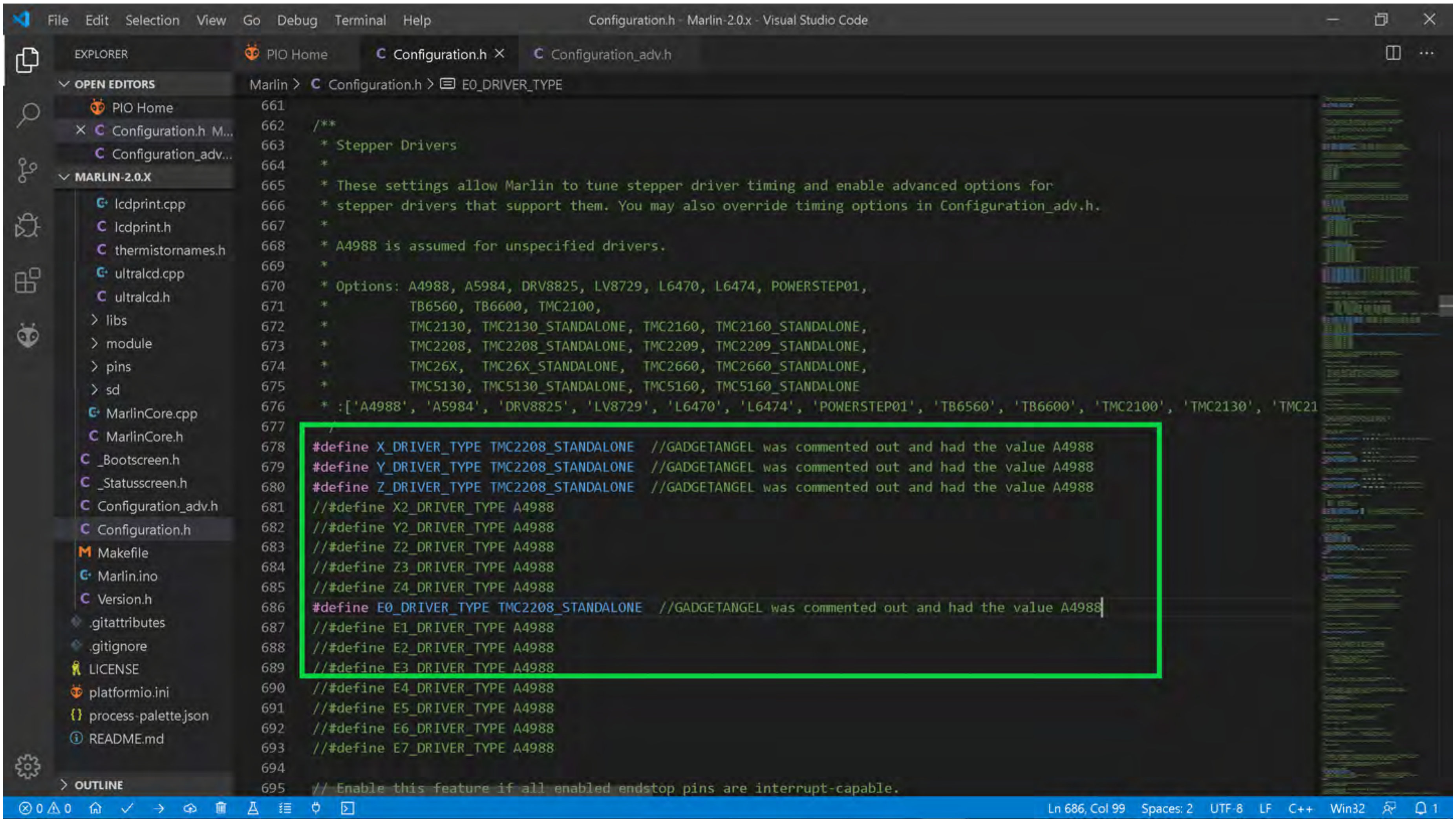


NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

The (latest release of) Marlin Setup for FYSETC TMC2208 V1.2 Drivers in One Time Programming (OTP) Mode

NOTE: **Go to Appendix C,** and then come back here for the changes to Marlin for FYSETC TMC2208 V1.2 stepper motor drivers in OTP mode.

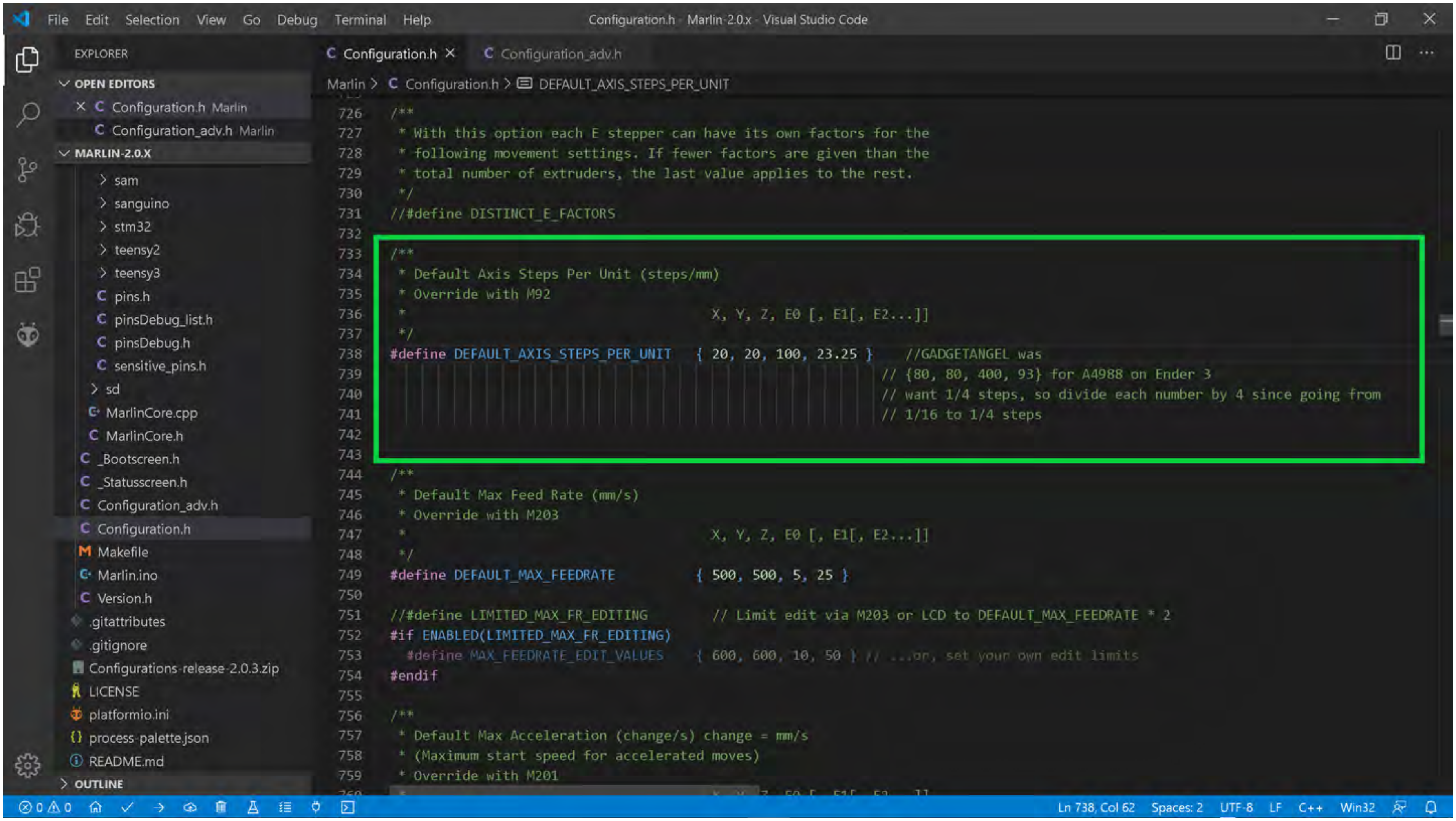
- Change the stepper motor drivers so that Marlin knows you are using TMC2208 drivers in OTP mode. Change one line for each axis and one for each extruder you will be using. See the picture below for an example of how to use TMC2208 drivers in OTP mode. When two "//" appear at the beginning of a line that means that line is commented out. To un-comment a line just remove the leading two forward slashes ("//").



- Go to the next page.

The (latest release of) Marlin Setup for FYSETC TMC2208 V1.2 Drivers in One Time Programming (OTP) Mode

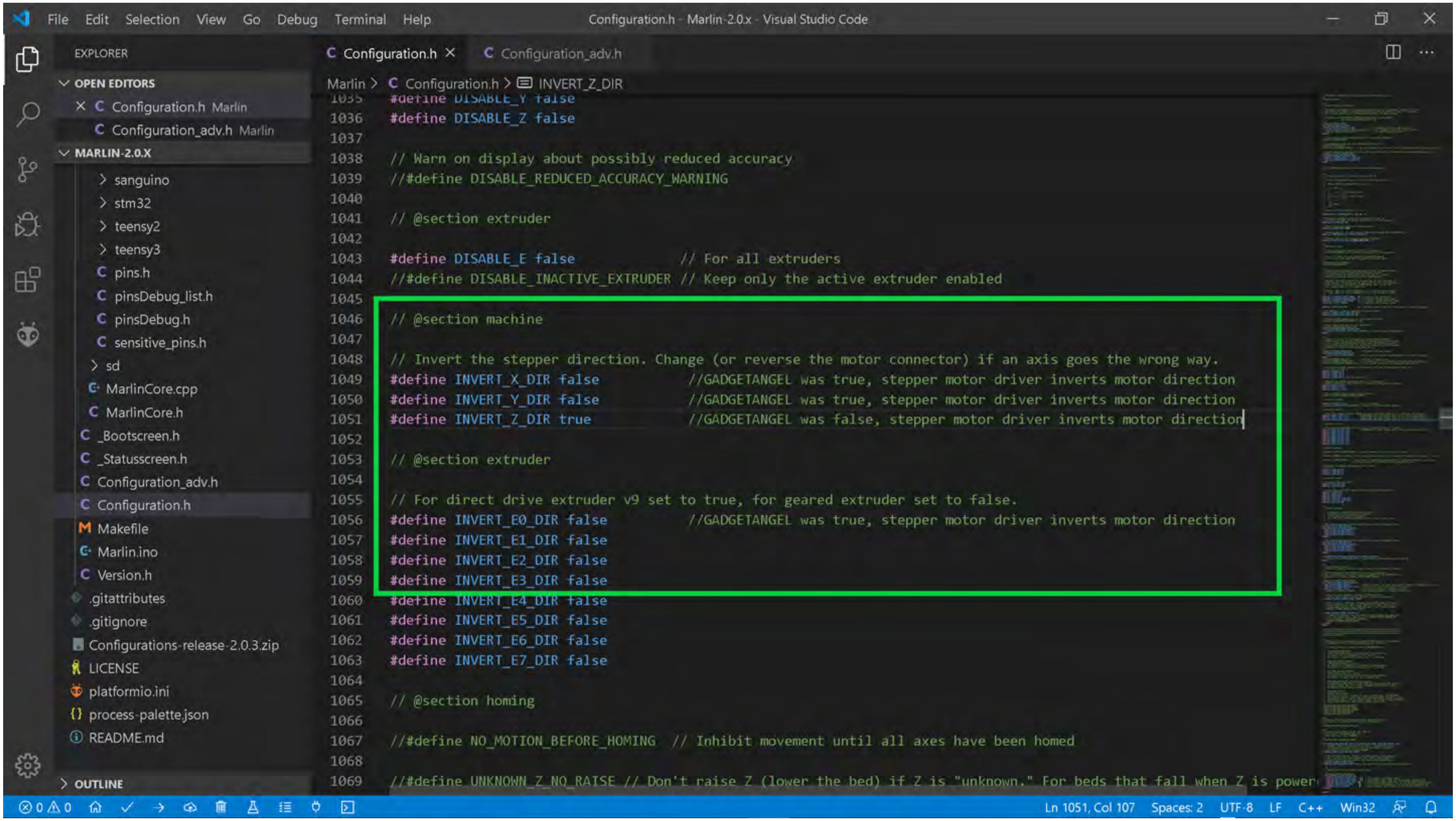
- Since I desire to use 1/4 stepping, and we are changing from A4988 stepper motor drivers on the Ender 3 to FYSETC TMC2208 stepper motor drivers for each axis and the extruder stepper motor driver, we will be going from 1/16 stepping to 1/4 stepping. So we are cutting our STEPS by one quarter. Therefore, we must adjust our "DEFAULT_AXIS_STEPS_PER_UNIT" anytime our STEPS are NOT 1/16. So change "DEFAULT_AXIS_STEPS_PER_UNIT" to {20, 20, 100, 23.25}, as seen in the **GREEN** box below.



- Go to the next page.

The (latest release of) Marlin Setup for FYSETC TMC2208 V1.2 Drivers in One Time Programming (OTP) Mode

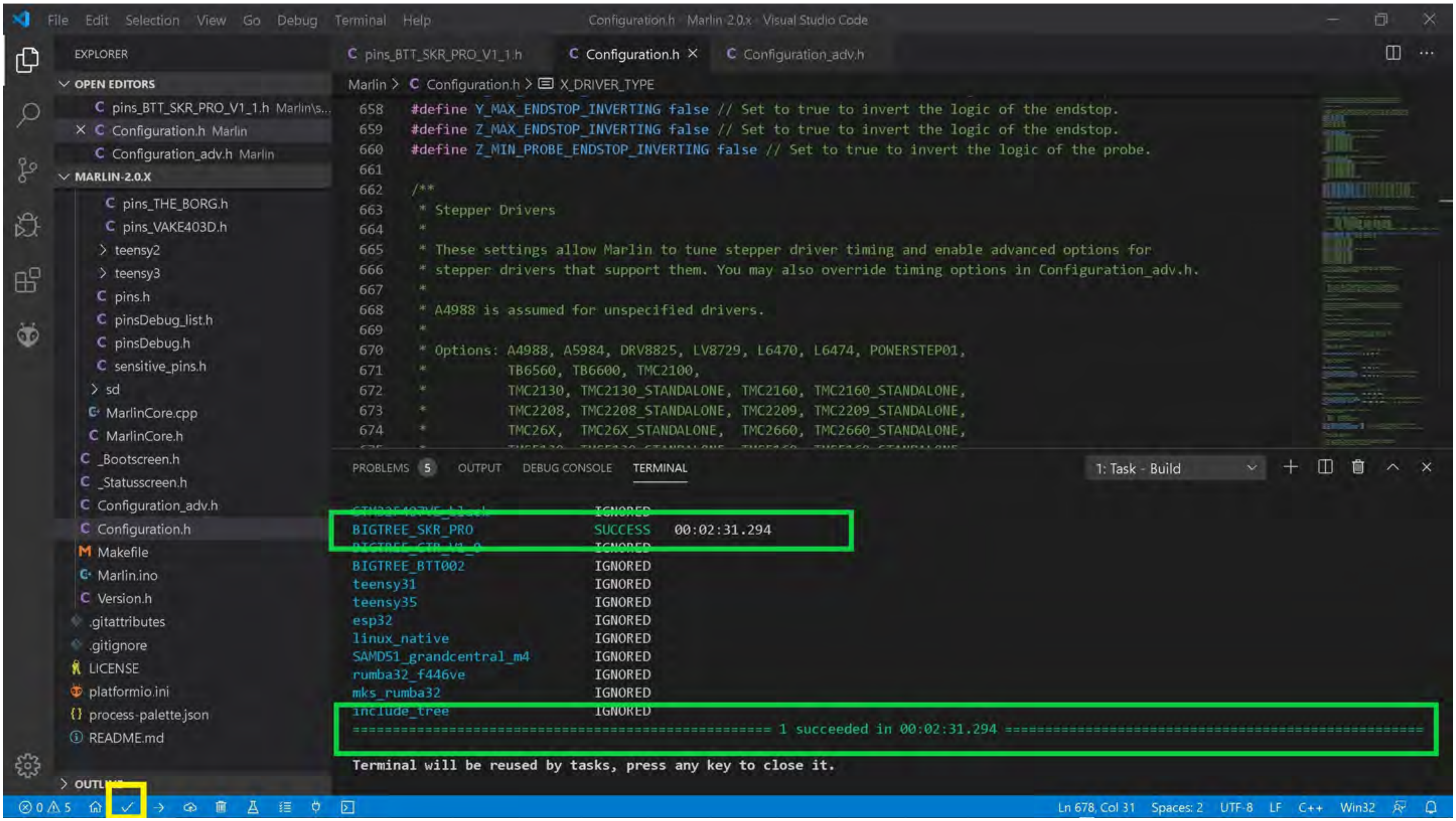
- Since the A4988 driver is what my Ender 3 used, but, now I want to use TMC2208 drivers, I must invert the stepper motor direction because the TMC2208 driver will turn the motors in the opposite direction than the A4988 driver's motor direction. So if the axis' setting you will be using the TMC2208 driver on was "true" change it to "false", as shown in the **GREEN** box below. If the setting was "false", now set it to "true", as shown in the **GREEN** box below



- Go to the next page.

The (latest release of) Marlin Setup for FYSETC TMC2208 V1.2 Drivers in One Time Programming (OTP) Mode

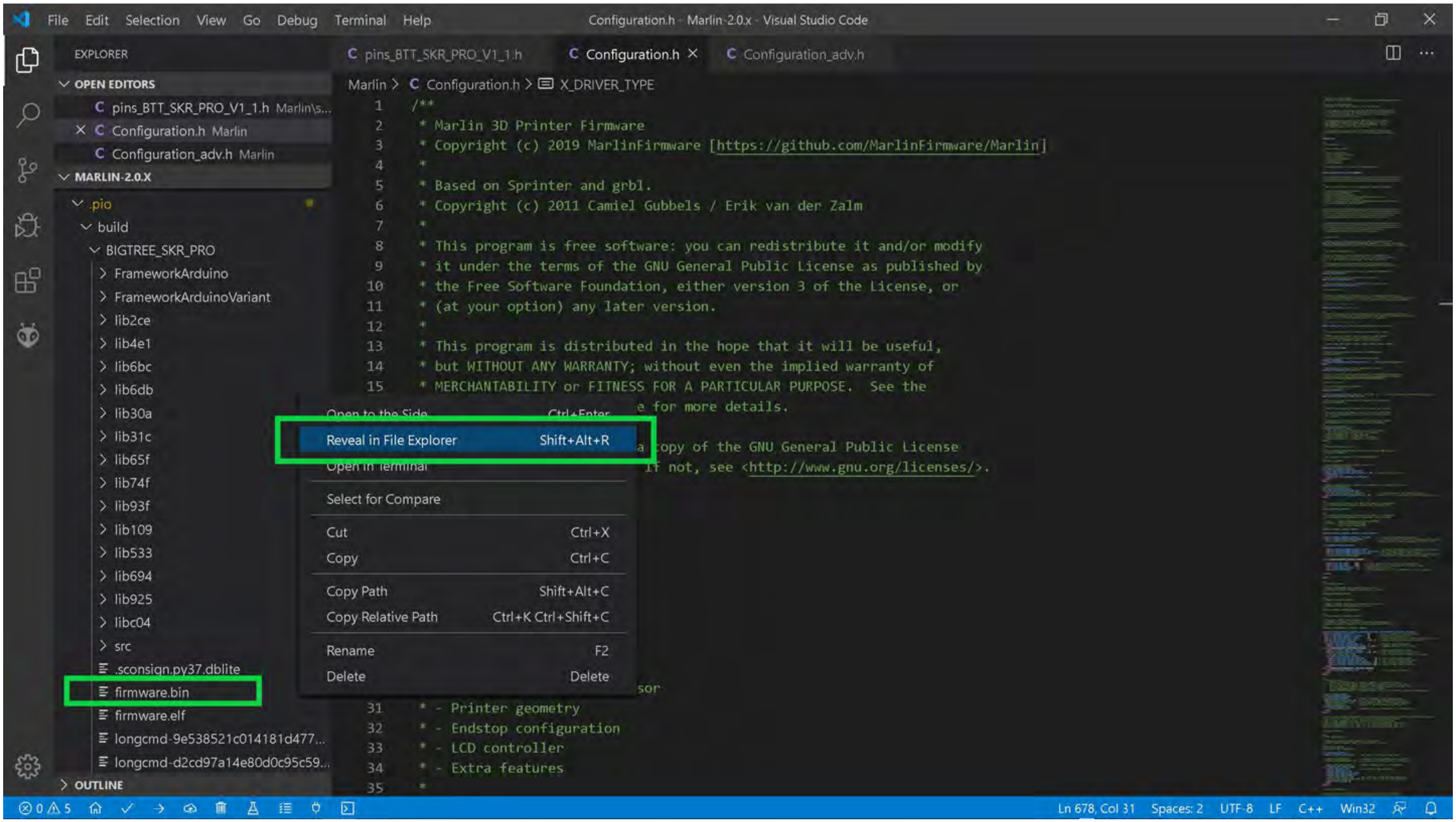
- The end of Marlin setup for FYSETC TMC2208 V1.2 drivers in OTP mode. Now, compile your code by clicking on the check mark, as seen in the **YELLOW** box below. Once the firmware has compiled successfully you will see the following messages, as shown in the **GREEN** boxes below.



- For the SKR PRO V1.1 board you must save the "firmware.bin" file to the micro SD card then place the micro SD card into the micro SD card reader of the SKR PRO V1.1 board and turn on the power to your printer for the NEW firmware to be automatically applied to the SKR PRO MCU. To locate the "firmware.bin" file, and learn how to transfer it to your micro SD card go to the next page.

The (latest release of) Marlin Setup for FYSETC TMC2208 V1.2 Drivers in One Time Programming (OTP) Mode

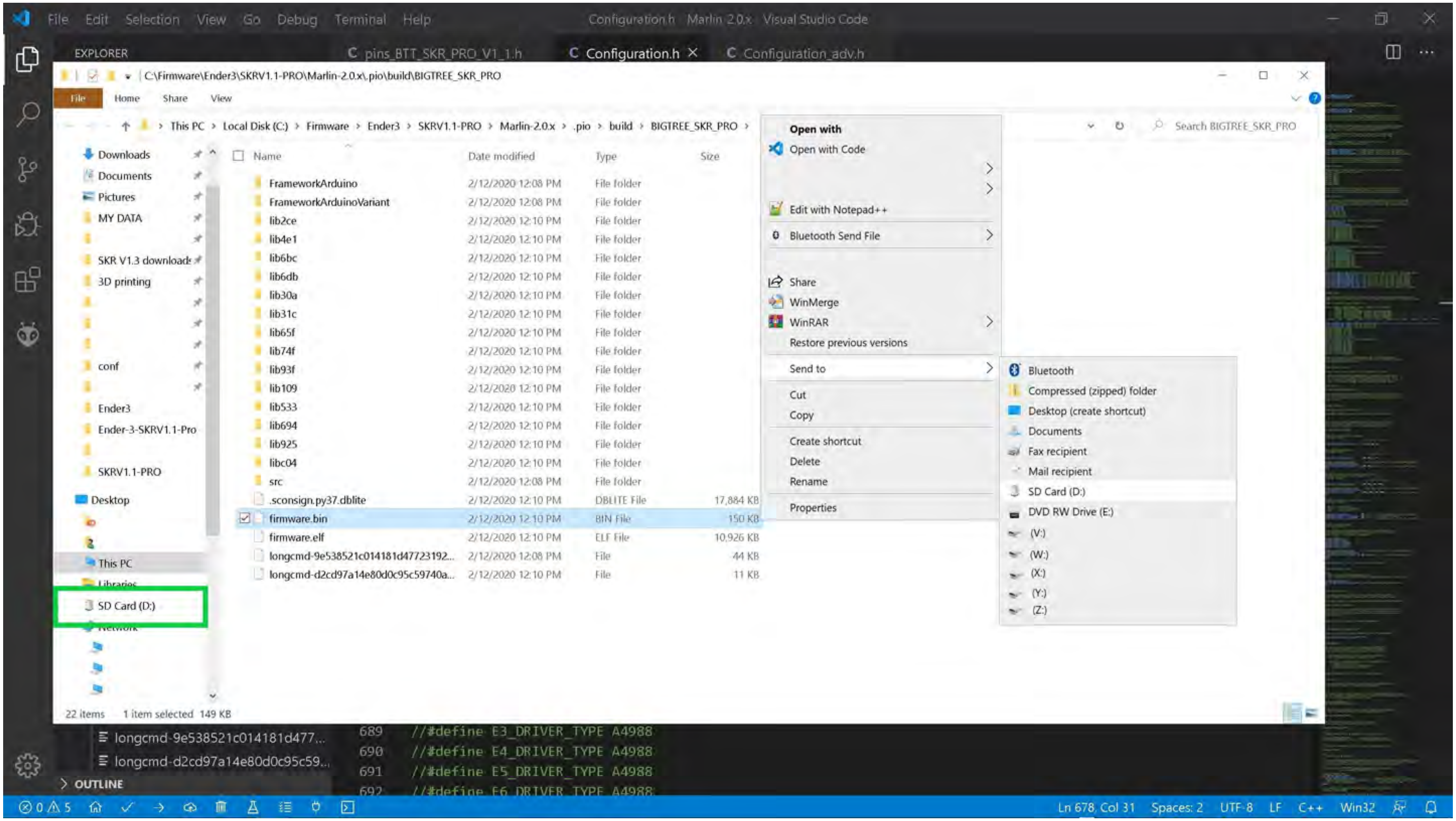
- The "firmware.bin" file can be found by using the left side of the VS code window and going to folder Marlin-2.0.X/.pio/build/ BIGTREE_SKR_PRO/ and right clicking on the "firmware.bin" file name to bring up the content menu. Select "Reveal in File Explorer" to have your Window's machine open a file explorer window.



- Go to the next page.

The (latest release of) Marlin Setup for FYSETC TMC2208 V1.2 Drivers in One Time Programming (OTP) Mode

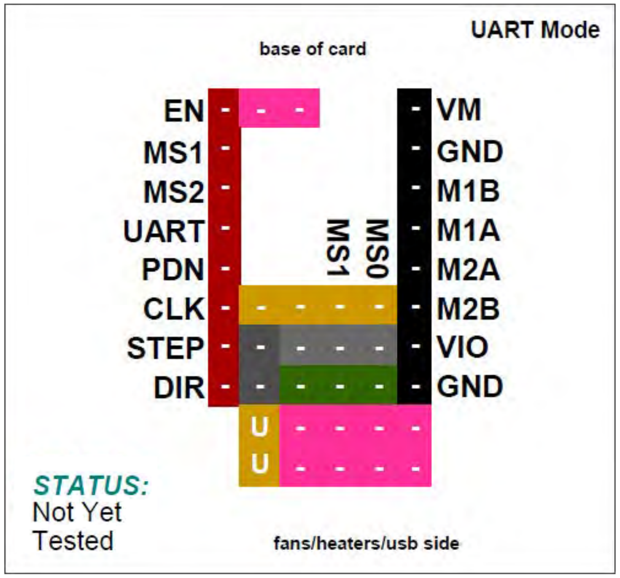
- Ensure your micro SD card, which came with your SKR PRO V1.1 board, with your SKR PRO V1.1 board is in a micro-to-SD-card adapter and plugged into your Window's SD card reader.
    - From the file explorer window right click on the "firmware.bin" file name and select "Send to" from the content menu. For my computer "D:" is my SD card reader. For your computer it may be a different drive letter. The "firmware.bin" file will be written to the SD card and renamed to "FIRMWARE.CUR".



- Right click on the SD card reader, as seen in the GREEN box above, and select "Eject" from the context menu. Take the micro SD card out of the adapter and place it into the micro SD card reader of the SKR PRO V1.1 board. Turn on the power to the printer (i.e. the SKR PRO V1.1 board) and the "FIRMWARE.CUR" file will be uploaded to the board. After the upload the firmware file will be renamed to "firmware.bin" and stored on the micro SD card.

FYSETC TMC2208 V1.2

UART Mode



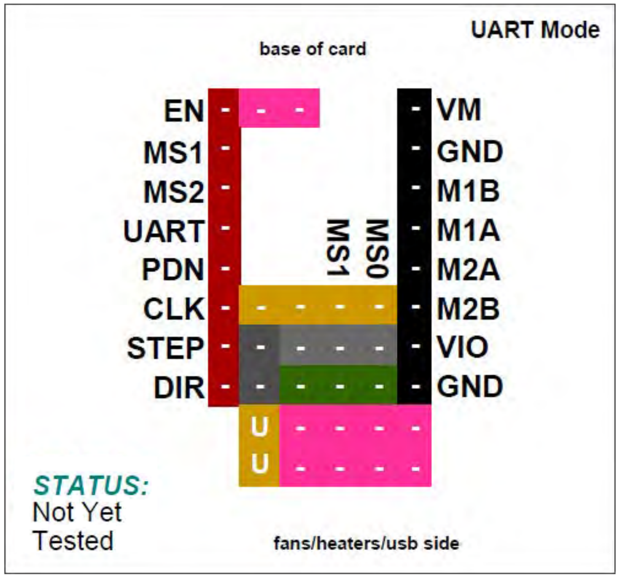
Note: You can use 50% to 90% of the calculated I_{RMS} (I_{MAX}/1.414) when tuning ("X_CURRENT", "Y_CURRENT", etc. the stepper motor driver in the firmware.

See the next page for further information.

Driver Chip	Steps are set inside of your Firmware	
FYSETC TMC2208 UART Mode Maximum 256 Subdivision 35V DC 2A (peak)		
Driving Current Calculation Formula R_S(Typical Sense Resistor)= 0.11Ω	$I_{MAX}=V_{ref}$ See Appendix B #4. Use 50% to 90% as shown below: $I_{MAX}=I_{MAX} * 0.90$	$V_{ref}=I_{MAX}$ See Appendix B #4. Use 50% to 90% as shown below: $V_{ref}=V_{ref} * 0.90$

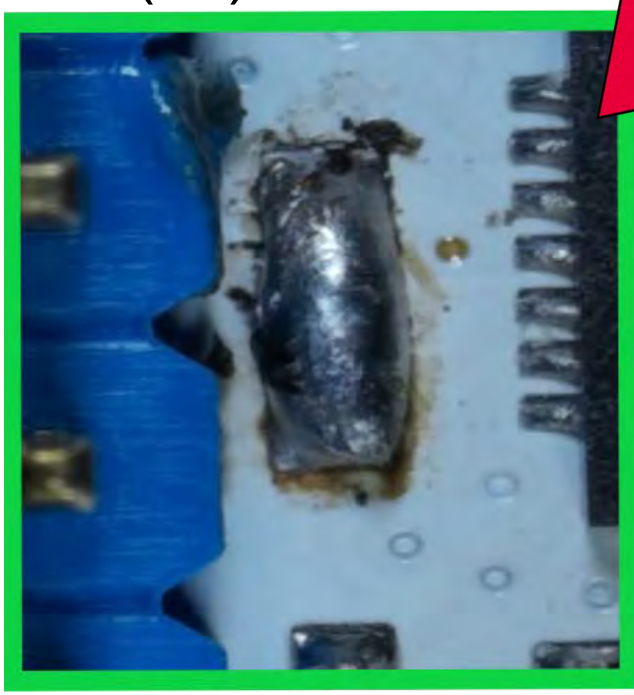
FYSETC TMC2208 V1.2

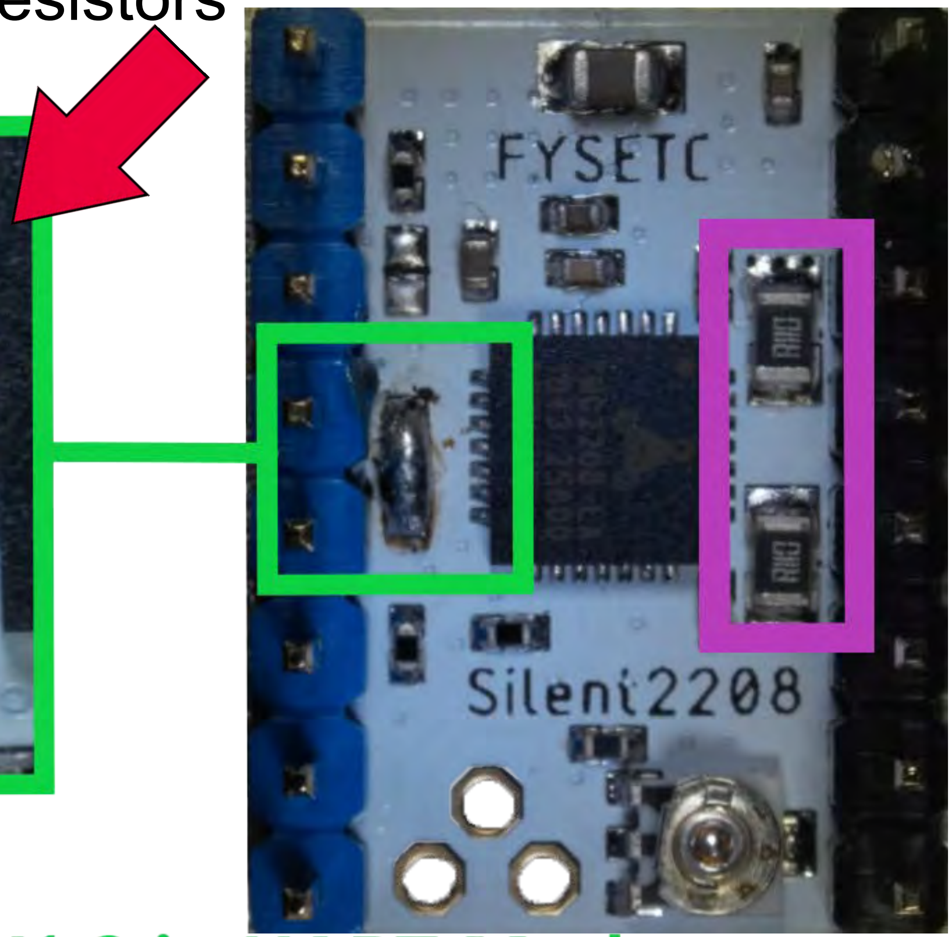
UART Mode



Important: To ensure that the FYSETC TMC2208 V1.0 or V1.2 is in UART Mode, check to see if all the adjacent pads are soldered together on the bottom of the driver board, as shown in the **GREEN** box. The **PURPLE** box shows the location of the current sense resistors (R_s).

MOST FYSETC TMC2208 boards are sold as V1.2 driver boards. The V1.2 driver board might be setup to be in UART mode. BUT the V1.0 driver board is setup to be in stand-alone mode by default. So please check your boards to ensure they are in the correct mode you desire!





FYSETC TMC2208 V1.2 in UART Mode

NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

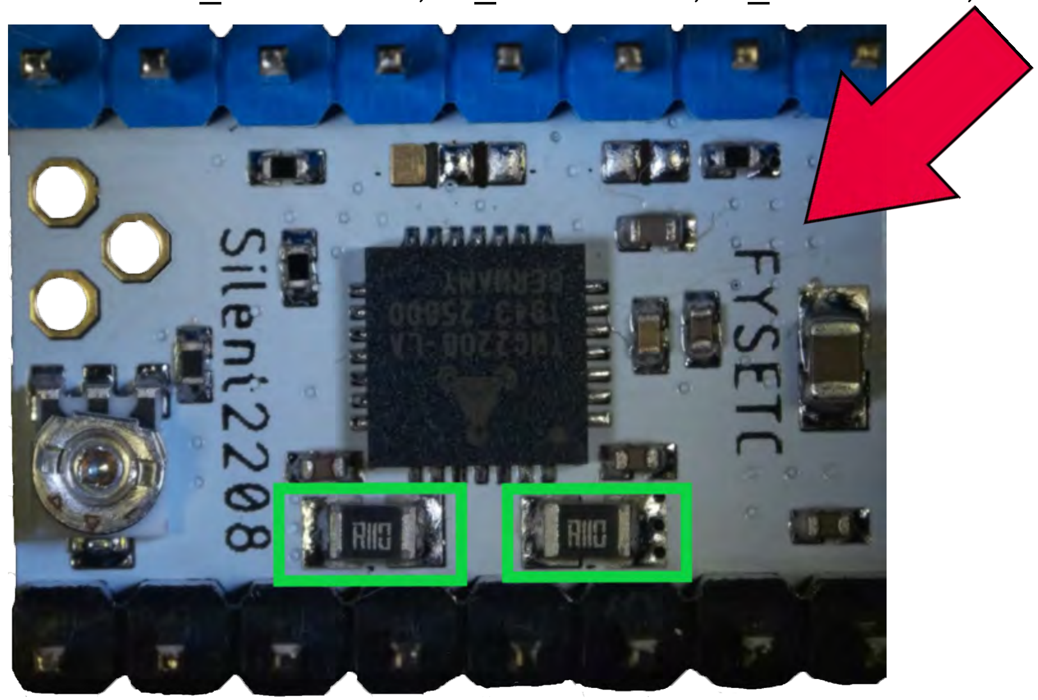
FYSETC TMC2208 V1.2

UART Mode

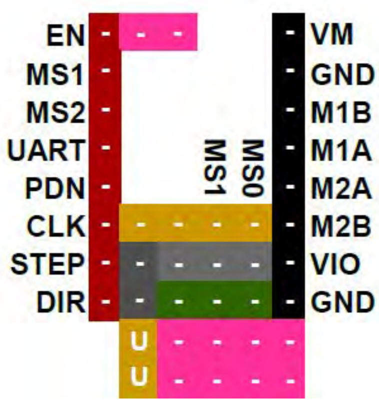
UART Mode

Note: The location of the current sense resistors are shown in **GREEN**. Use the current sense resistors' value in the Marlin Firmware ("X_RSENSE", "Y_RSENSE", "Z_RSENSE" and/or "E0_RSENSE") so that the appropriate current limit can be sent to the driver board. If you do not want to use V_{ref} as the value for "X_CURRENT", "Y_CURRENT", "Z_CURRENT" and/or "E0_CURRENT", you should use I_{RMS} instead. You find I_{RMS} by taking I_{MAX} and dividing it by 1.414 ($I_{RMS}=I_{MAX}/1.414$). You use 50% to 90% of the calculated I_{RMS} as the value for "X_CURRENT", "Y_CURRENT", "Z_CURRENT", and/or "E0_CURRENT".

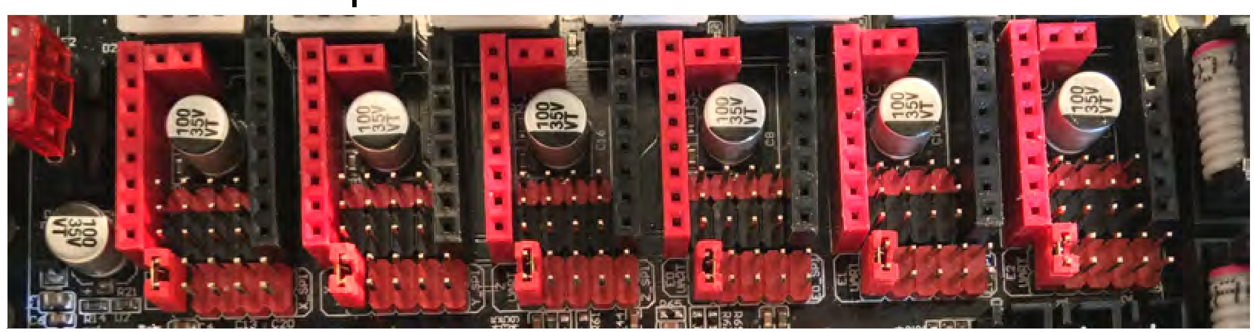
R_s = R050 is 0.05 Ohms
R_s = R062 is 0.062 Ohms
R_s = R068 is 0.068 Ohms
R_s = R075 is 0.075 Ohms
R_s = R100 is 0.1 Ohms
R_s = R110 is 0.11 Ohms
R_s = R150 is 0.15 Ohms
R_s = R200 is 0.2 Ohms
R_s = R220 is 0.22 Ohms



UART



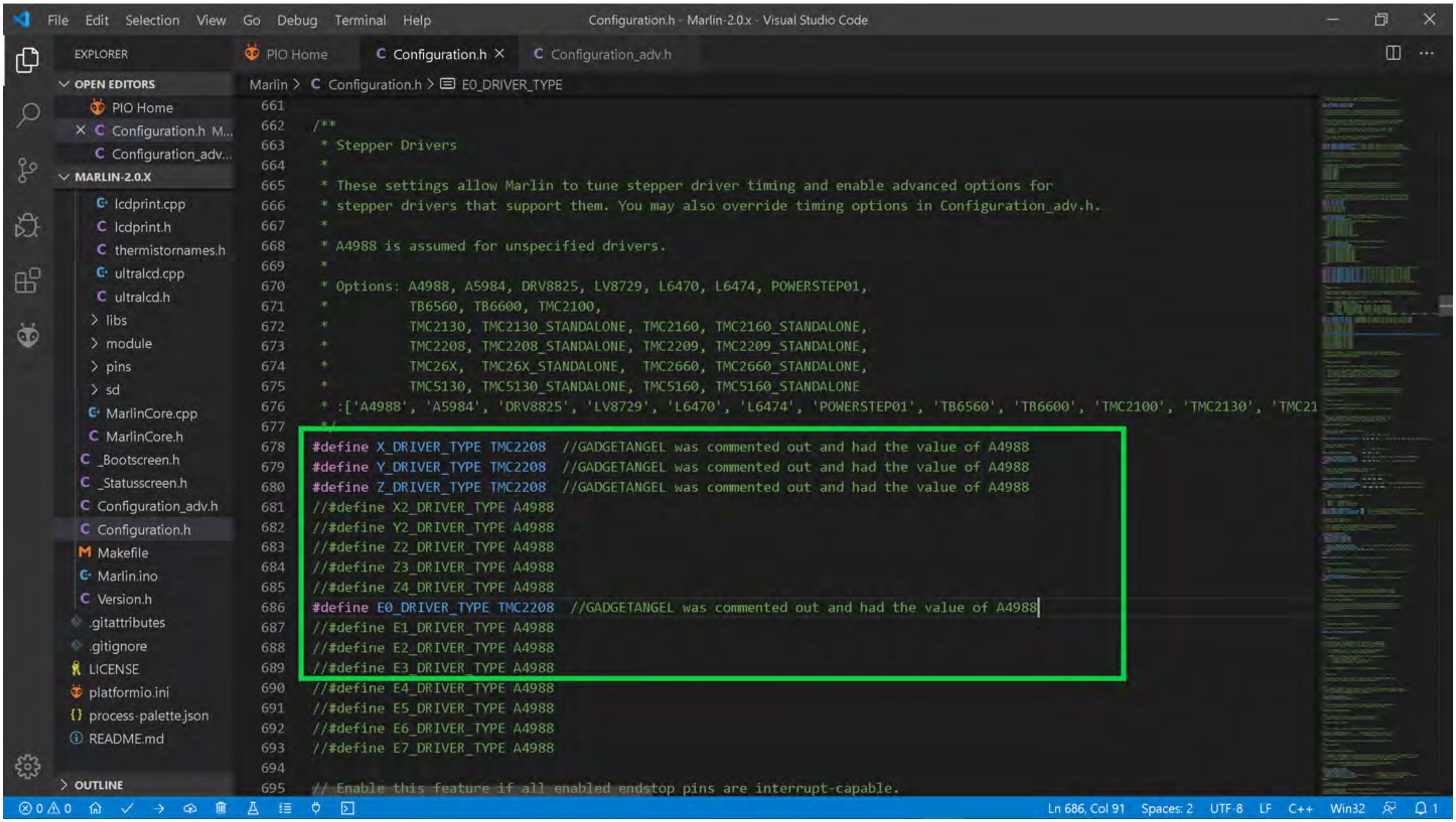
Note: Set Jumper "U" for UART MODE!!



The (latest release of) Marlin Setup for FYSETC TMC2208 V1.2 Drivers in UART Mode

NOTE: **Go to Appendix C**, and then come back here for the changes to Marlin for FYSETC TMC2208 stepper motor drivers in UART mode.

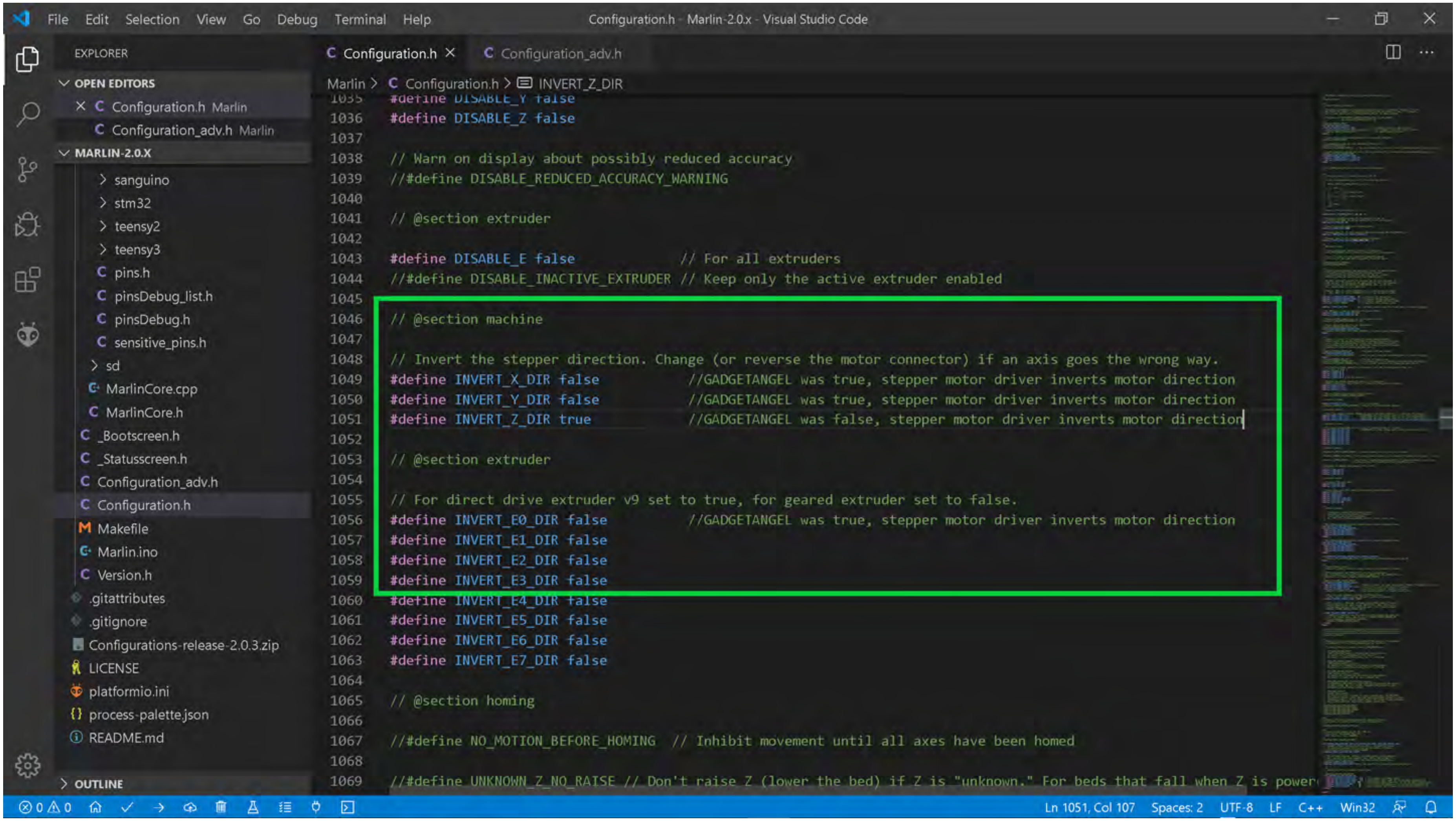
- Change the stepper motor drivers so that Marlin knows you are using TMC2208 drivers in UART mode. Change one line for each axis and one for each extruder you will be using. See the picture below for an example of how to use TMC2208 drivers in UART mode. When two "//" appear at the beginning of a line that means that line is commented out. To un-comment a line just remove the leading two forward slashes ("//").



- Go to the next page.

The (latest release of) Marlin Setup for FYSETC TMC2208 V1.2 Drivers in UART Mode

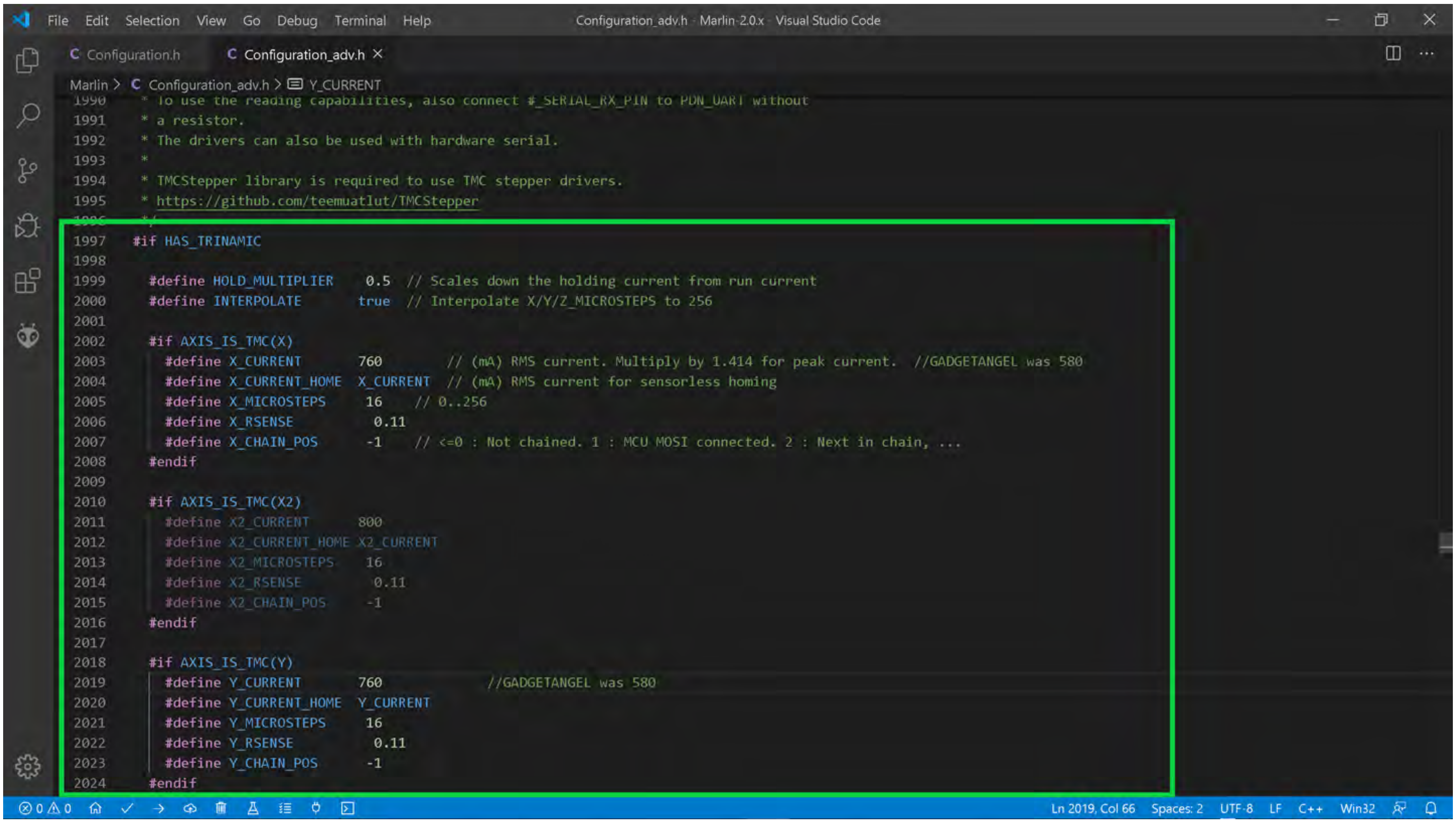
- Since the A4988 driver is what my Ender 3 used, but, now I want to use TMC2208 drivers, I must invert the stepper motor direction because the TMC2208 driver will turn the motors in the opposite direction than the A4988 driver's motor direction. So if the axis' setting you will be using the TMC2208 driver on was "true" change it to "false", as shown in the **GREEN** box below. If the setting was "false", now set it to "true", as shown in the **GREEN** box below



- Go to the next page.

The (latest release of) Marlin Setup for FYSETC TMC2208 V1.2 Drivers in UART Mode

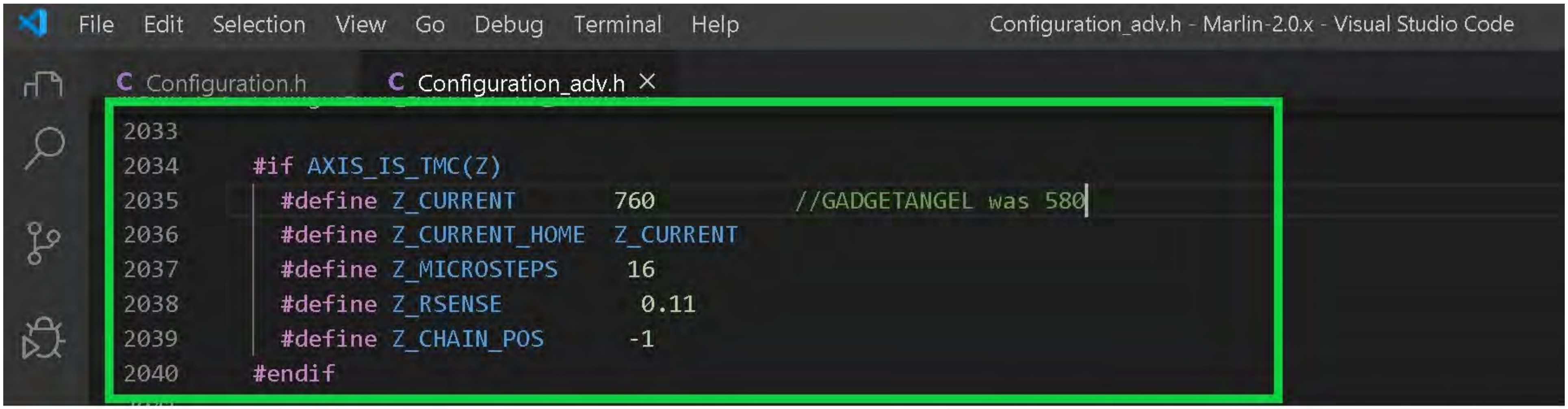
- Next you want to set your V_{ref} in the Marlin firmware for each axis that has the TMC2208 driver, as seen in the **GREEN** box below. I changed the "X_CURRENT" to be the calculated V_{ref} for my X-Axis, which is 760mV for an Ender 3. I changed the "Y_CURRENT" to be the calculated V_{ref} for my Y-Axis, which is 760mV on the Ender 3.
- Ensure "X_RSENSE" is set to 0.11. Ensure "Y_RSENSE" is set to 0.11.
- If you do not want to use V_{ref} as the value for "X_CURRENT" and/or "Y_CURRENT", you should use I_{RMS} instead. You find I_{RMS} by taking I_{MAX} and dividing it by 1.414 ($I_{RMS}=I_{MAX}/1.414$). You use 50% to 90% of the calculated I_{RMS} as the value for "X_CURRENT" and/or "Y_CURRENT".

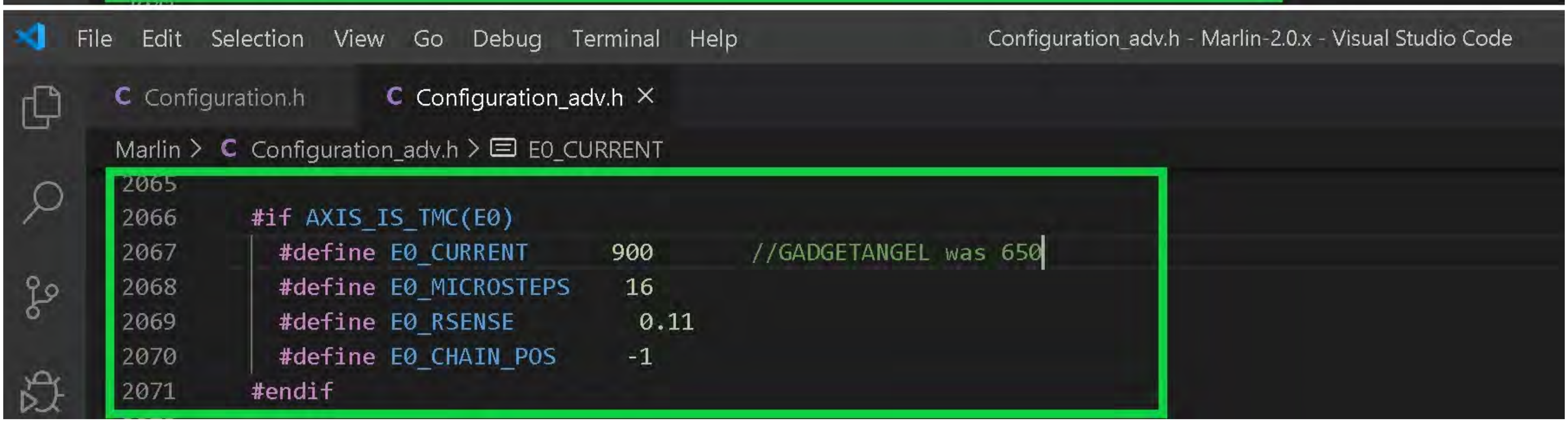


- Go to the next page.

The (latest release of) Marlin Setup for FYSETC TMC2208 V1.2 Drivers in UART Mode

- Now, I am setting the V_{ref} for Z-Axis and the extruder, as seen in the **GREEN** boxes below. I changed the "Z_CURRENT" to be the calculated V_{ref} for my Z-Axis, which is 760mV for an Ender 3. I changed the "E0_CURRENT" to be the calculated V_{ref} for my Extruder, which is 900mV on the Ender 3.
- Ensure "Z_RSENSE" is set to 0.11. Ensure "E0_RSENSE" is set to 0.11.
- If you do not want to use V_{ref} as the value for "Z_CURRENT" and/or "E0_CURRENT", you should use I_{RMS} instead. You find I_{RMS} by taking I_{MAX} and dividing it by 1.414 ($I_{RMS}=I_{MAX}/1.414$). You use 50% to 90% of the calculated I_{RMS} as the value for "Z_CURRENT" and/or "E0_CURRENT".

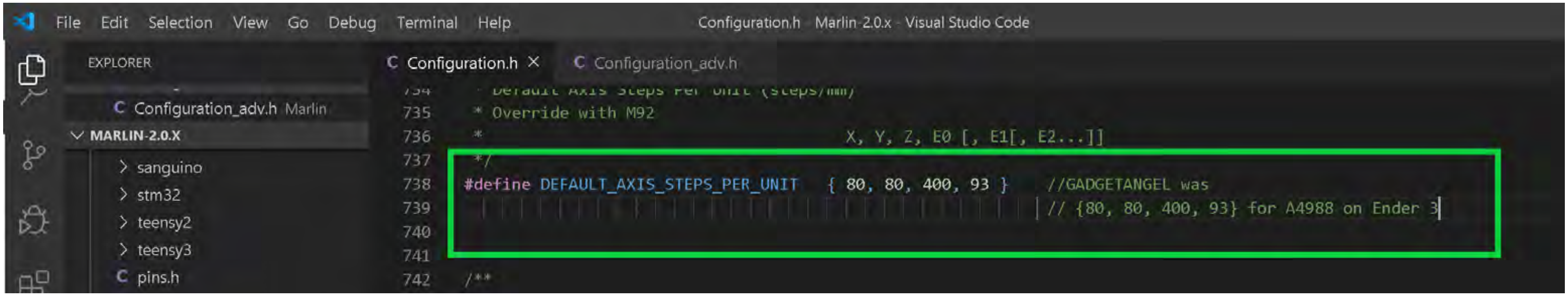




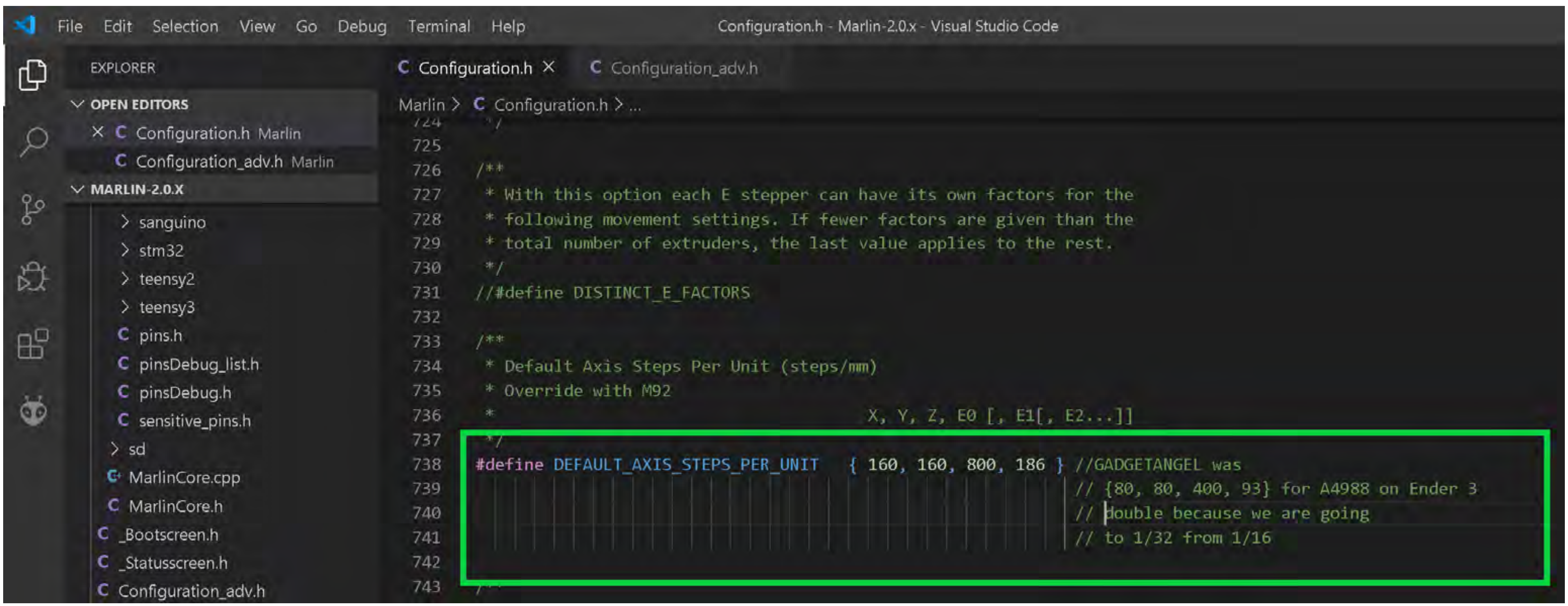
- Go to the next page.

The (latest release of) Marlin Setup for FYSETC TMC2208 V1.2 Drivers in UART Mode

- If you changed the "MICROSTEPS" for any of the axes then you will need to update "DEFAULT_AXIS_STEPS_PER_UNIT" to reflect your changes



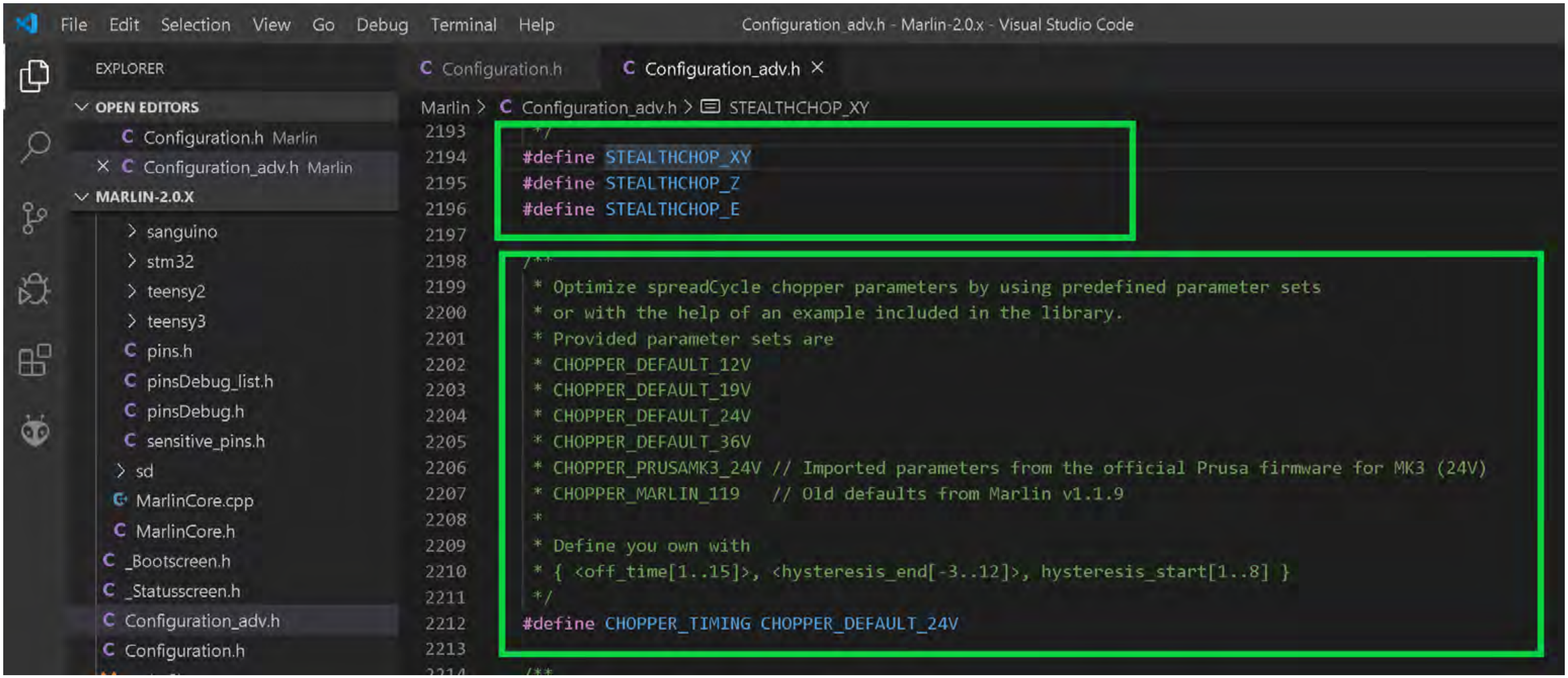
- FOR EAMPLE if you wanted to use 1/32 stepping instead of the default 1/16, you would be **doubling** your STEPS. Therefore, we must adjust our "DEFAULT_AXIS_STEPS_PER_UNIT" anytime our STEPS are NOT 1/16. So change "DEFAULT_AXIS_STEPS_PER_UNIT" to {160, 160, 800, 186}, as seen in the **GREEN** box below.



- Go to the next page.

The (latest release of) Marlin Setup for FYSETC TMC2208 V1.2 Drivers in UART Mode

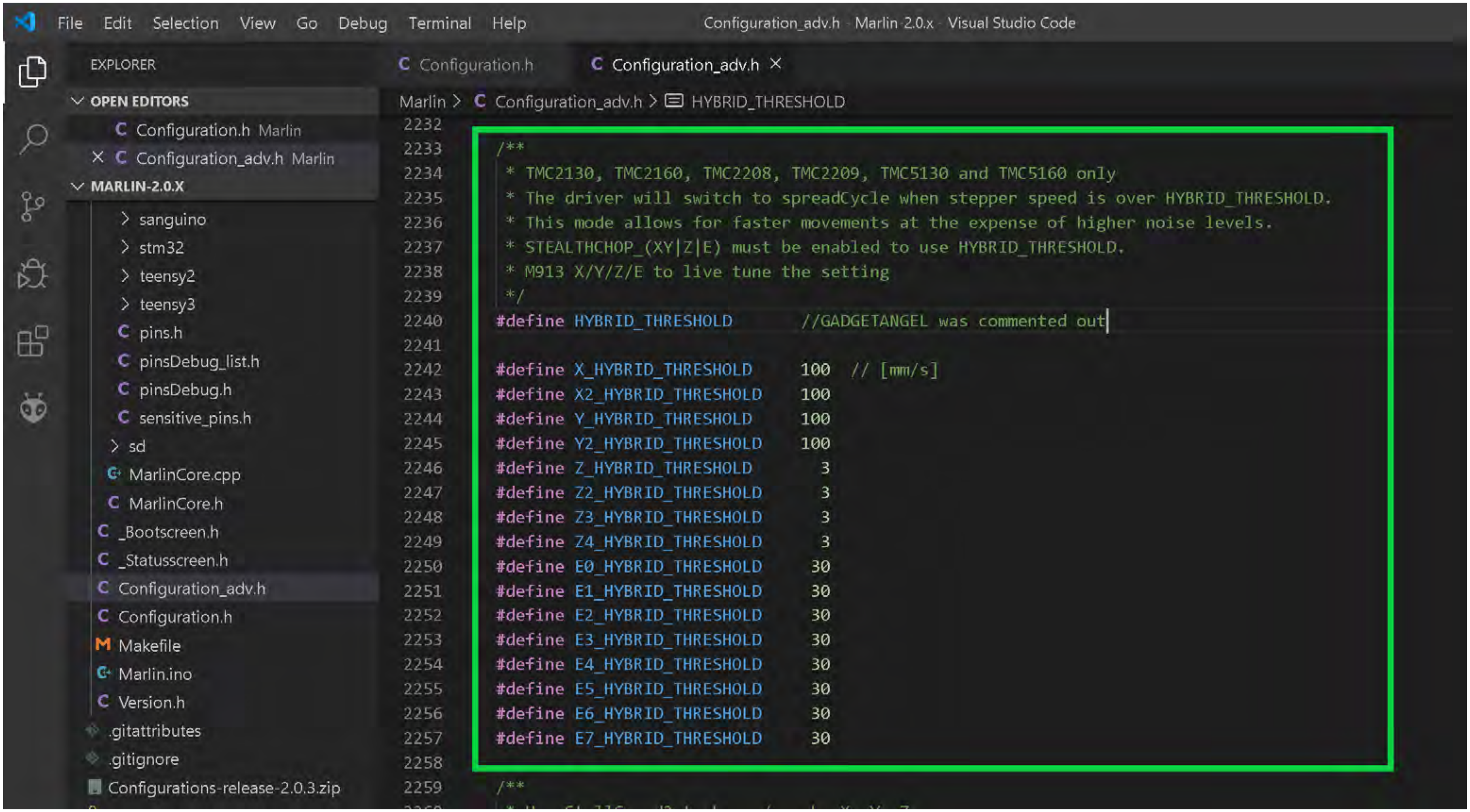
- By default stealthChop is enabled in the Marlin firmware. If you want spreadCycle ONLY then comment out the appropriate lines. I want stealthChop enabled so I want to make sure the lines are not commented out {"STEALTHCHOP_XY", "STEALTHCHOP_Z" and "STEALTHCHOP_E"}. You also want to check to see if the proper "CHOPPER_TIMING" is set for your printer. An Ender 3 is a 24VDC printer, my "CHOPPER_TIMING" is correct.



- Go to the next page.

The (latest release of) Marlin Setup for FYSETC TMC2208 V1.2 Drivers in UART Mode

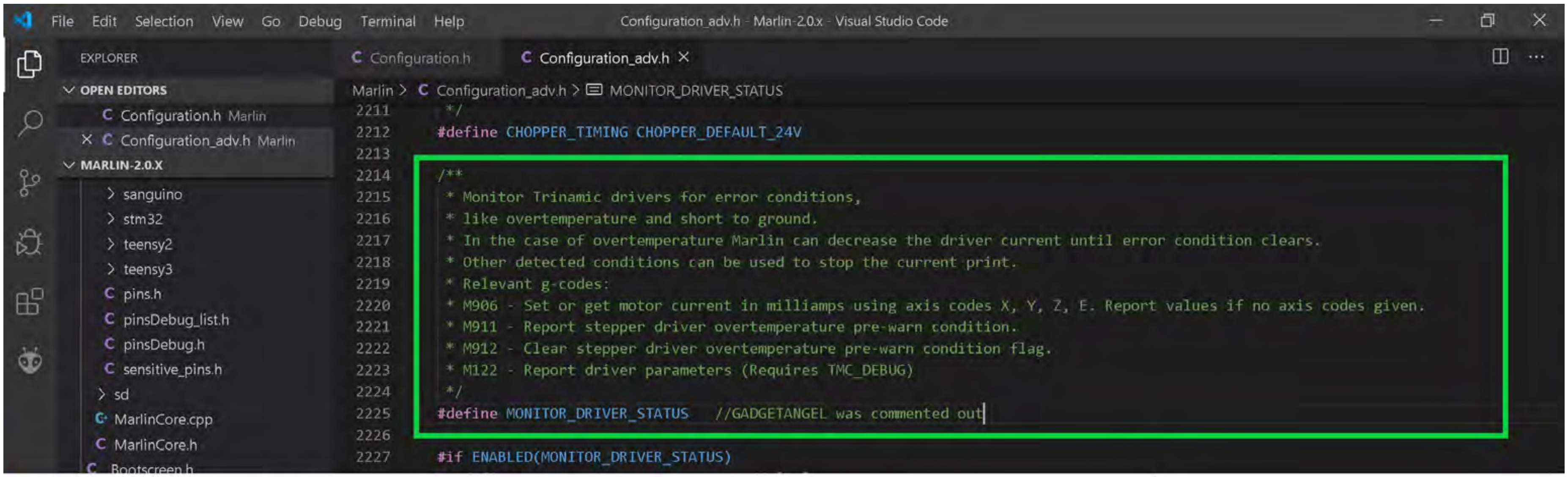
- Now you either enable "HYBRID_THRESHOLD" or disable it. By default it is disabled. "HYBRID_THERSHOLD" allows the printer to change between stealthChop and spreadCycle dynamically depending on the print speed. I want "HYBRID_THRESHOLD" enabled so I need to remove the two leading "//", which uncomments the line in the Marlin firmware.

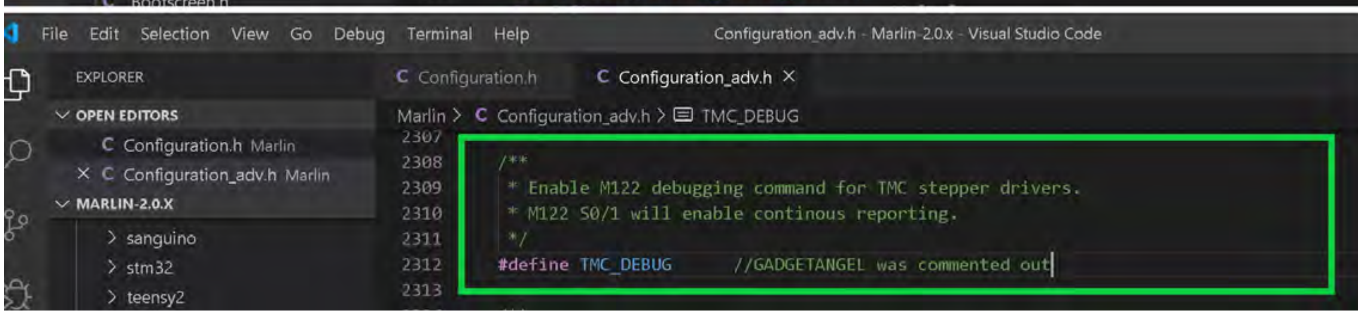


- Go to the next page.

The (latest release of) Marlin Setup for FYSETC TMC2208 V1.2 Drivers in UART Mode

- Now I want to enable some statements that allow me access to debugging the TMC drivers. I will uncomment "MONITOR_DRIVER_STATUS" and "TMC_DEBUG". "MONITOR_DRIVER_STATUS" will enable the following G-codes: M906, M911, and M912, "TMC_DEBUG" will enable the M122 G-code command. You can read about these from the comments in the firmware and in **Marlin's documentation located on-line**.

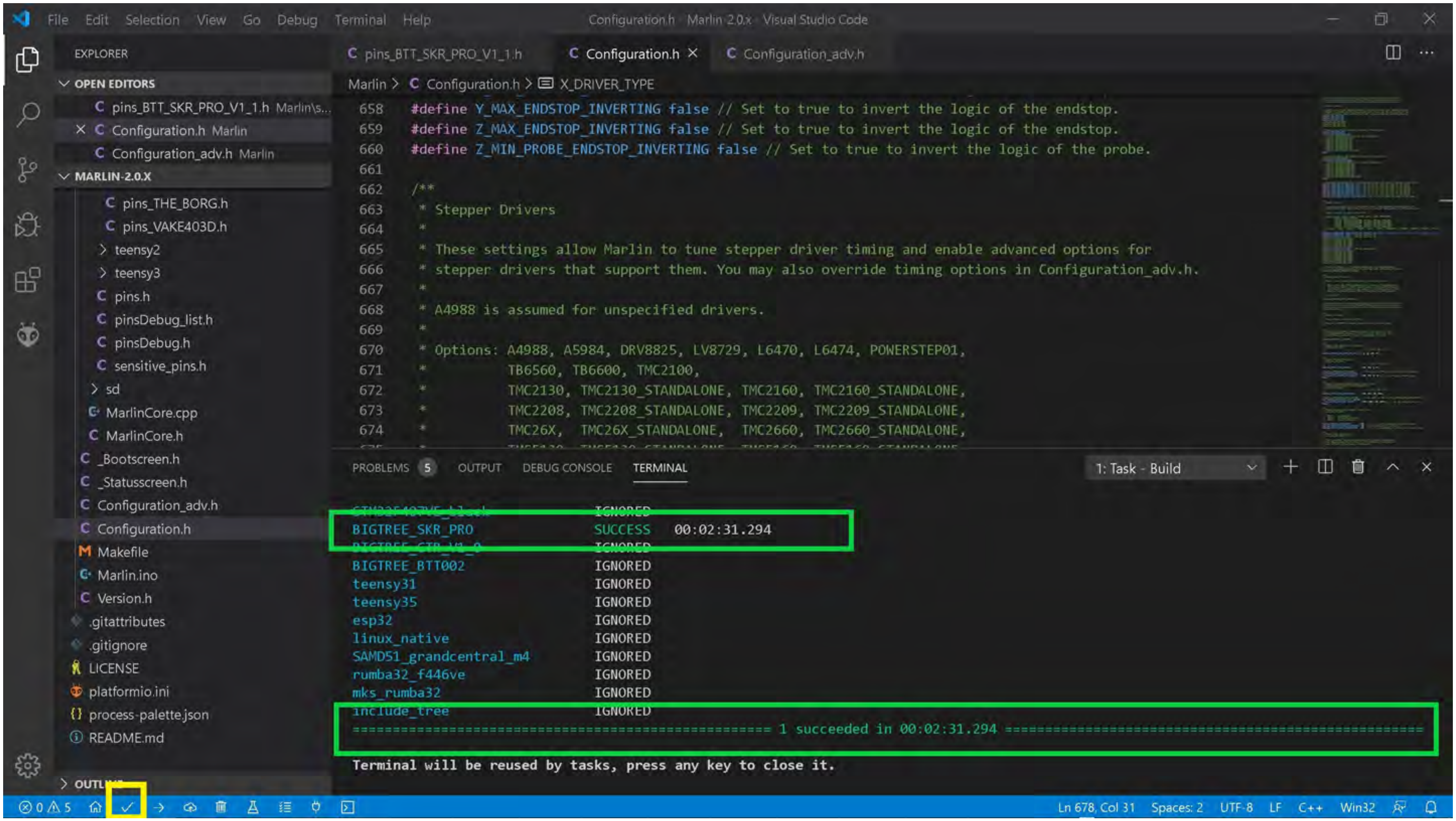




- Go to the next page.

The (latest release of) Marlin Setup for FYSETC TMC2208 V1.2 Drivers in UART Mode

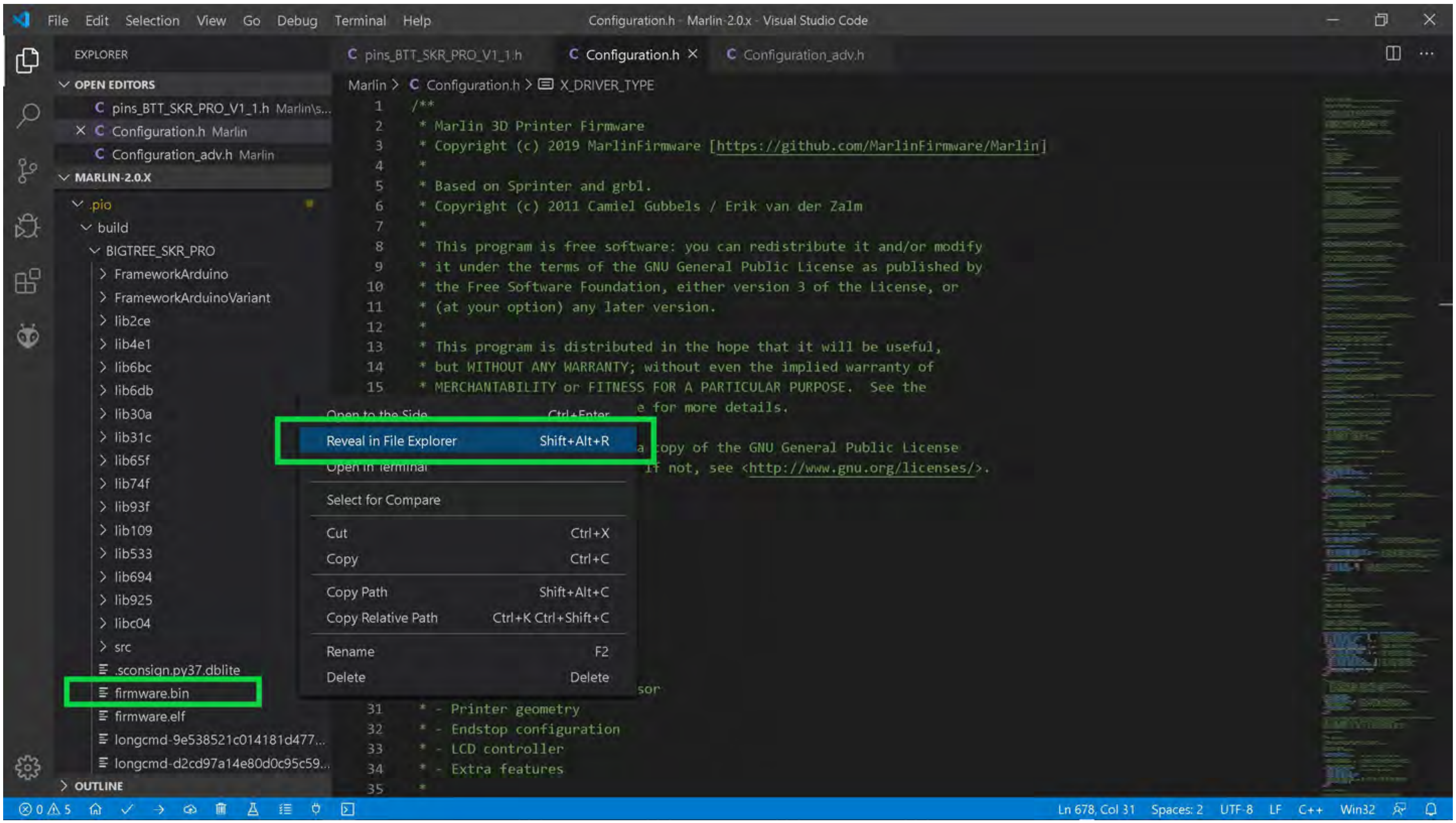
- The end of Marlin setup for FYSETC TMC2208 V1.2 drivers in UART mode. Now, compile your code by clicking on the check mark, as seen in the **YELLOW** box below. Once the firmware has compiled successfully you will see the following messages, as shown in the **GREEN** boxes below.



- For the SKR PRO V1.1 board you must save the "firmware.bin" file to the micro SD card then place the micro SD card into the micro SD card reader of the SKR PRO V1.1 board and turn on the power to your printer for the NEW firmware to be automatically applied to the SKR PRO MCU. To locate the "firmware.bin" file, and learn how to transfer it to your micro SD card go to the next page.

The (latest release of) Marlin Setup for FYSETC TMC2208 V1.2 Drivers in UART Mode

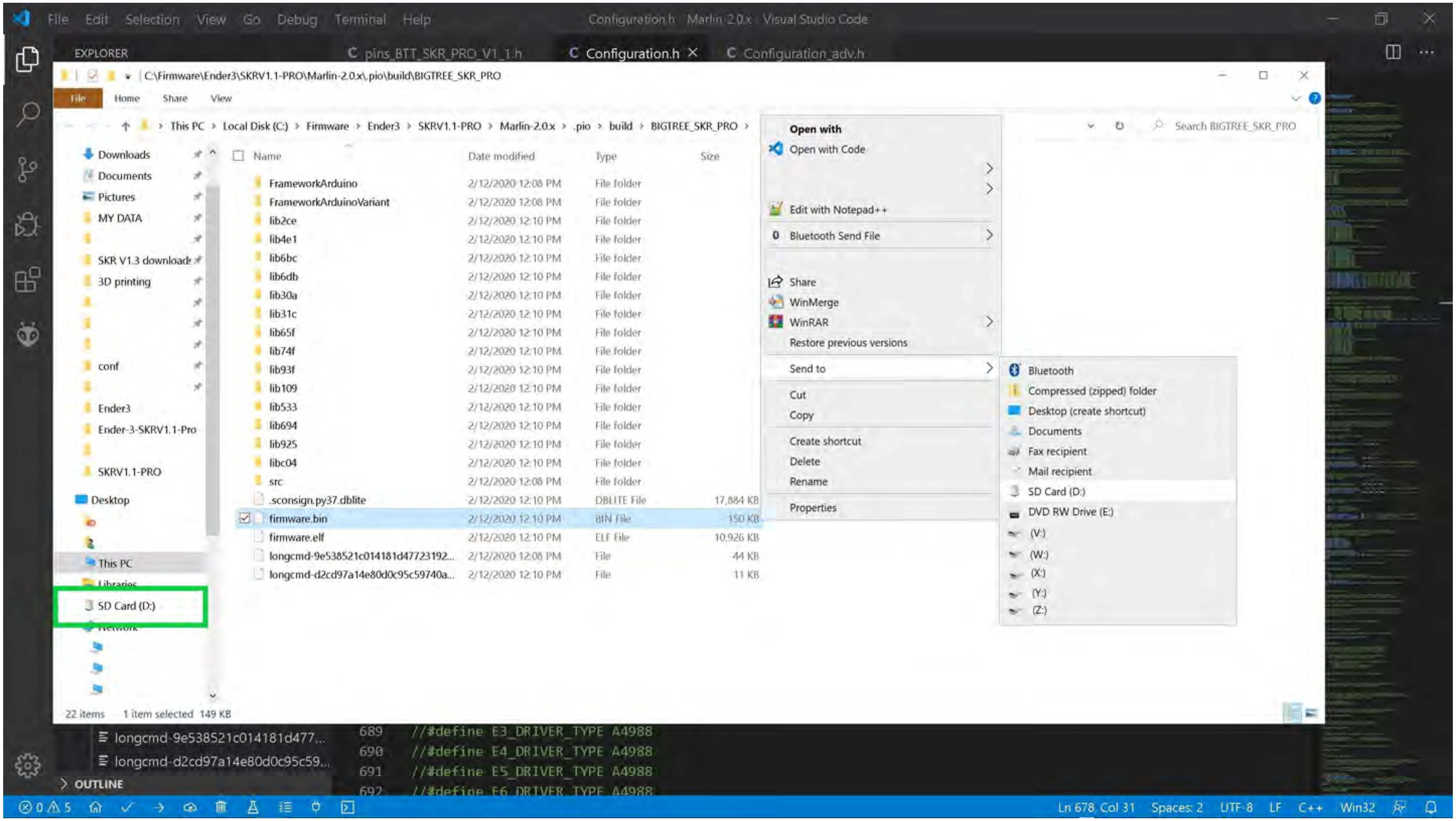
- The "firmware.bin" file can be found by using the left side of the VS code window and going to folder Marlin-2.0.X/.pio/build/ BIGTREE_SKR_PRO/ and right clicking on the "firmware.bin" file name to bring up the content menu. Select "Reveal in File Explorer" to have your Window's machine open a file explorer window.



- Go to the next page.

The (latest release of) Marlin Setup for FYSETC TMC2208 V1.2 Drivers in UART Mode

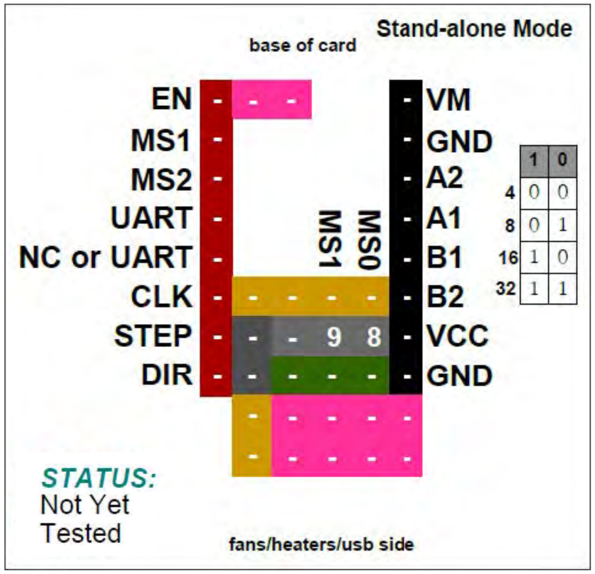
- Ensure your micro SD card, which came with your SKR PRO V1.1 board, with your SKR PRO V1.1 board is in a micro-to-SD-card adapter and plugged into your Window's SD card reader.
    - From the file explorer window right click on the "firmware.bin" file name and select "Send to" from the content menu. For my computer "D:" is my SD card reader. For your computer it may be a different drive letter. The "firmware.bin" file will be written to the SD card and renamed to "FIRMWARE.CUR".



- Right click on the SD card reader, as seen in the **GREEN** box above, and select "Eject" from the context menu. Take the micro SD card out of the adapter and place it into the micro SD card reader of the SKR PRO V1.1 board. Turn on the power to the printer (i.e. the SKR PRO V1.1 board) and the "FIRMWARE.CUR" file will be uploaded to the board. After the upload the firmware file will be renamed to "firmware.bin" and stored on the micro SD card.

BIQU TMC2225 V1.0

Stand-alone Mode



NOTE: Use the potentiometer (POT) on the top of the board, as shown in **RED**; or use the board's "V_{ref} Test point" location, as shown in **BLUE**, to set your V_{ref}. See **Appendix A** for instructions on how to set the V_{ref} on a driver board.

Note: Use 90% of the calculated V_{ref} when tuning the stepper driver board.

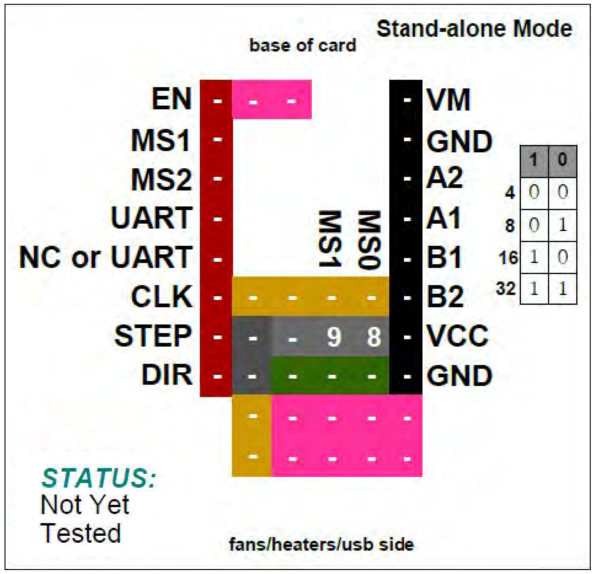
Driver Chip	MS1	MS0	Steps	Interpolation	Mode
BIQU® TMC2225 Stand Alone Mode Maximum 32 Subdivision 35V DC 2A (peak)	GND	GND	1 / 4	1 / 256	stealthChop
	GND	VIO	1 / 8	1 / 256	stealthChop
	VIO	GND	1 / 16	1 / 256	stealthChop
	VIO	VIO	1 / 32	1 / 256	stealthChop

Driving Current Calculation Formula R_S(Typical Sense Resistor)=0.15Ω	$I_{MAX}=V_{ref}*0.7222$ See Appendix B#10.Use 50% to 90% as shown below: $I_{MAX}=(V_{ref}*0.7222)*0.90$	$V_{ref}=I_{MAX}*1.3846$ See Appendix B#10.Use 50% to 90% as shown below: $V_{ref}=(I_{MAX}*1.3846)*0.90$

NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

BIQU TMC2225 V1.0

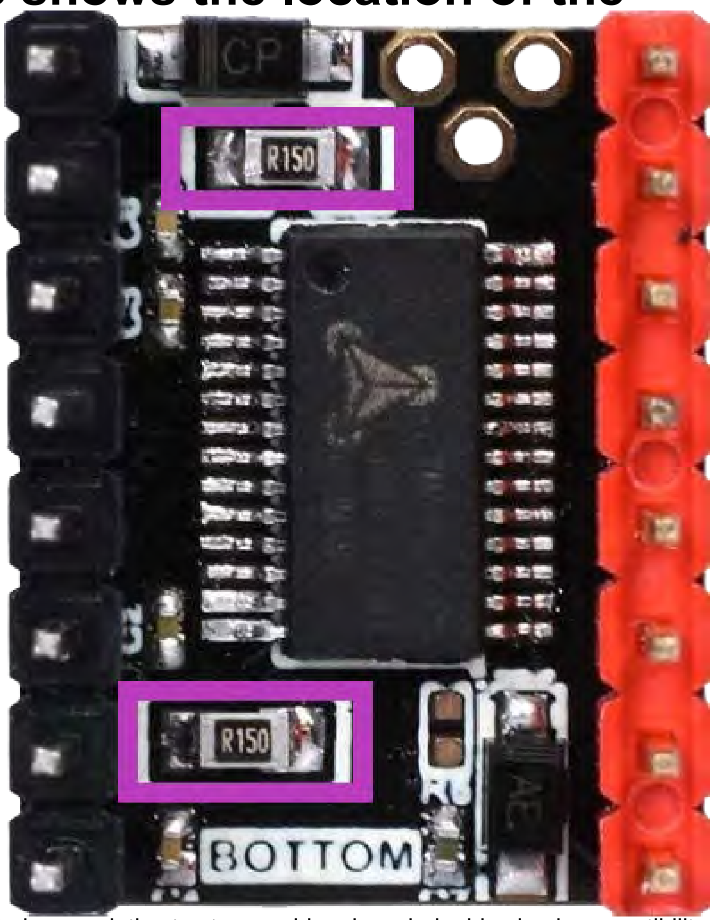
Stand-alone Mode

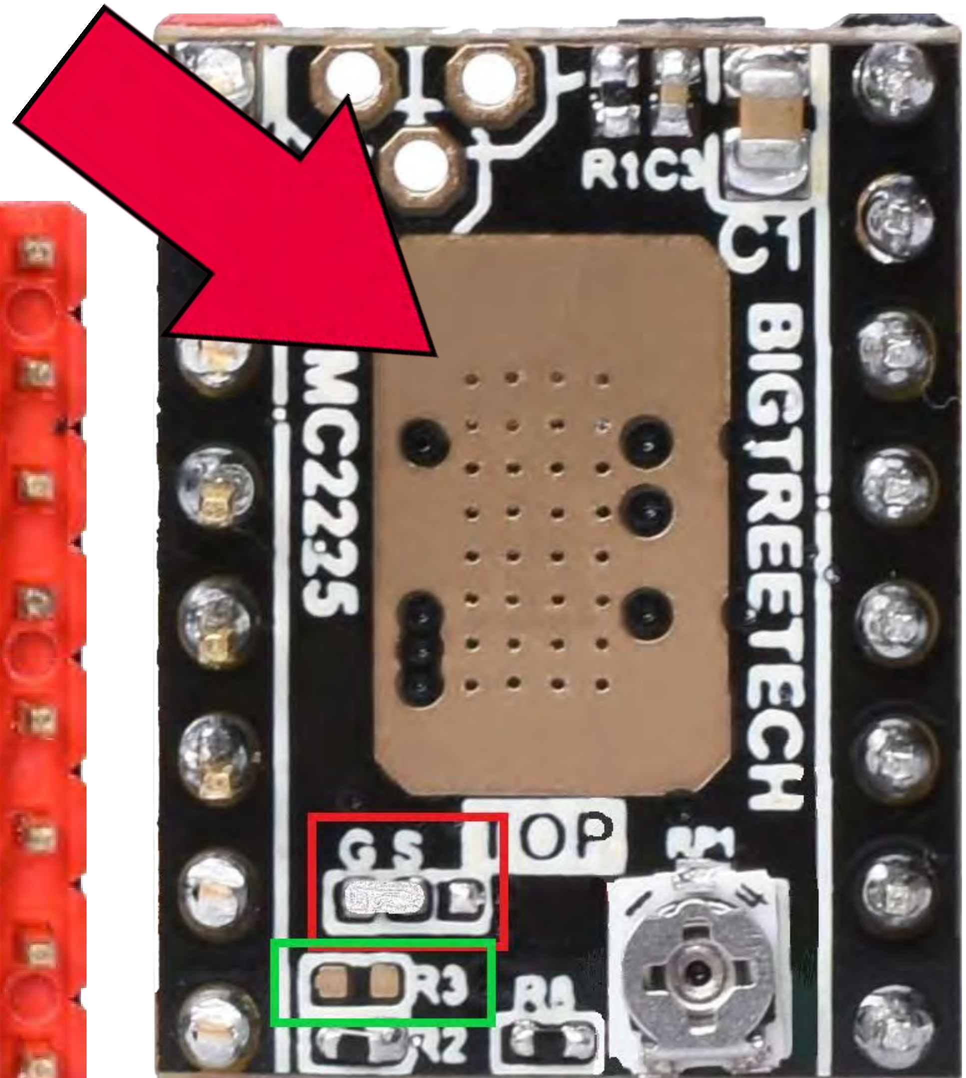


Note: To obtain **stand-alone mode** for the BIQU TMC2225 V1.0, the two pads located at R3 must have a gap between them, as seen in **GREEN** below, and the two pads at "G S" (located on the top of the driver board) must be set for StealthChop as seen in **RED** below.

The PURPLE boxes shows the location of the current sense resistors (R_s).

Note: MOST BIQU TMC2225 V1.0 driver boards, when purchased **for UART mode, will have two R3 pads (located on the top of the driver board), which are NOT soldered together.** This indicates the driver board can use the UART pin for the UART single wire interface (if the UART,"U", jumper is in place on the SKR PRO V1.1 board)





NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

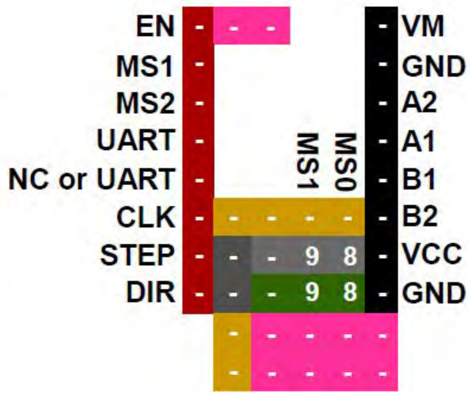
Stand-alone Mode

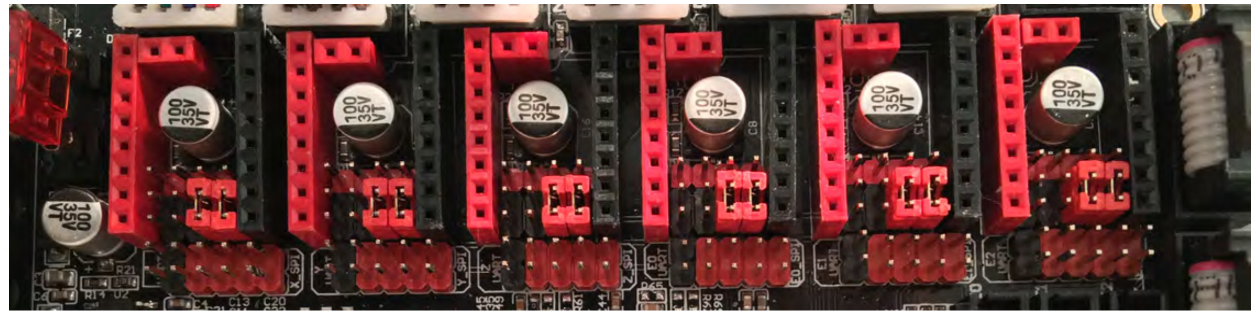
Stand-alone Mode

1 / 4

Interpolation: **1/256**

StealthChop



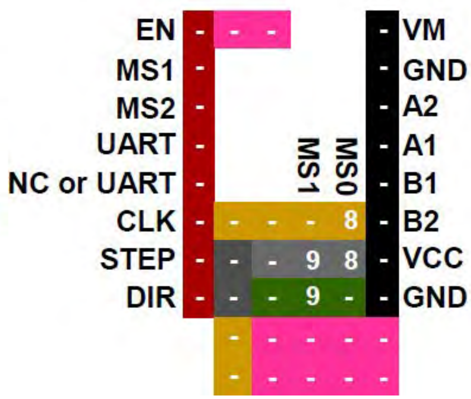


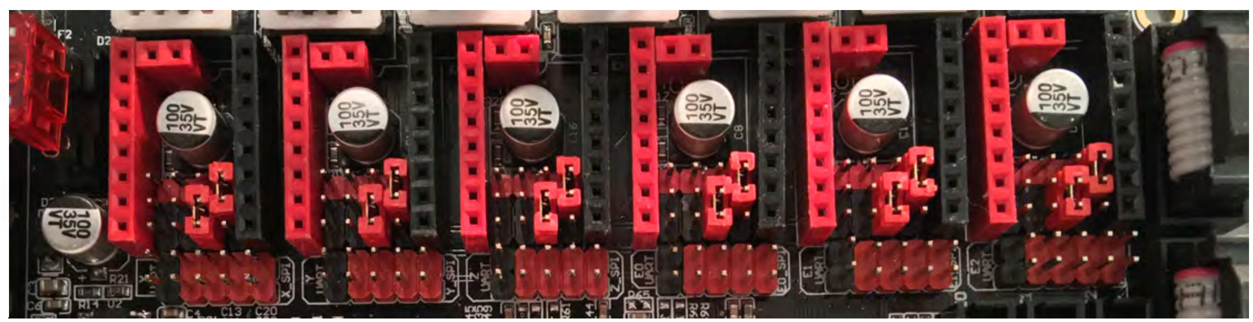
Stand-alone Mode

1 / 8

Interpolation: **1/256**

StealthChop





NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

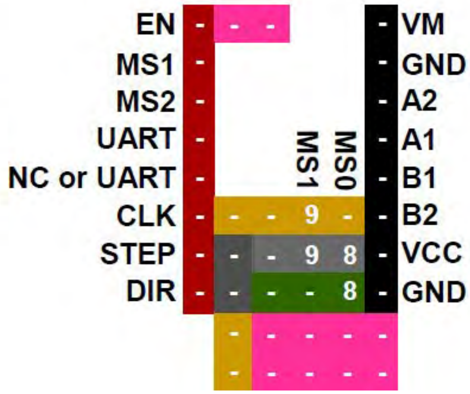
Stand-alone Mode

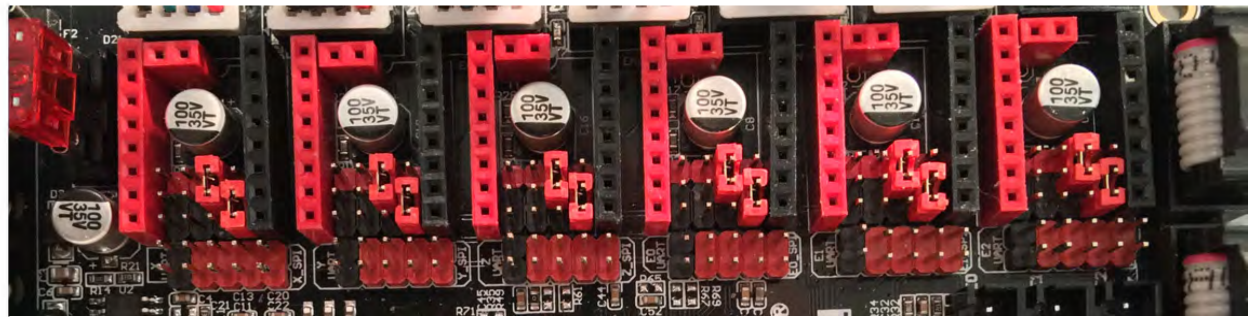
Stand-alone Mode

1 / 16

Interpolation: 1/256

StealthChop



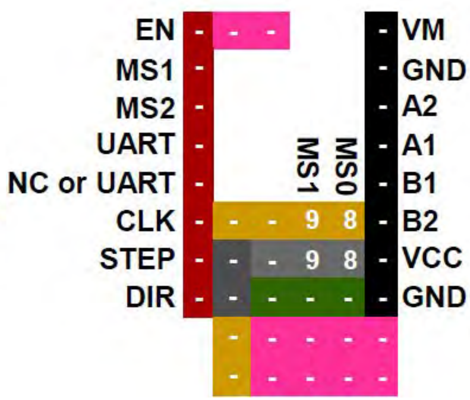


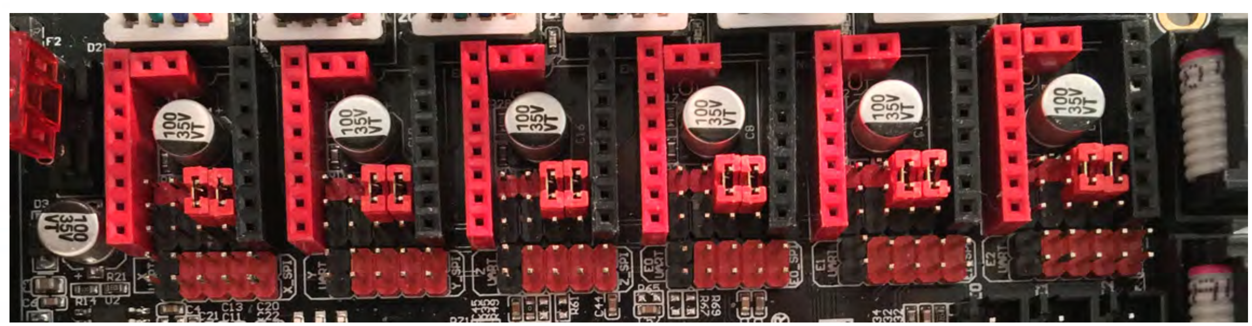
Stand-alone Mode

1 / 32

Interpolation: 1/256

StealthChop



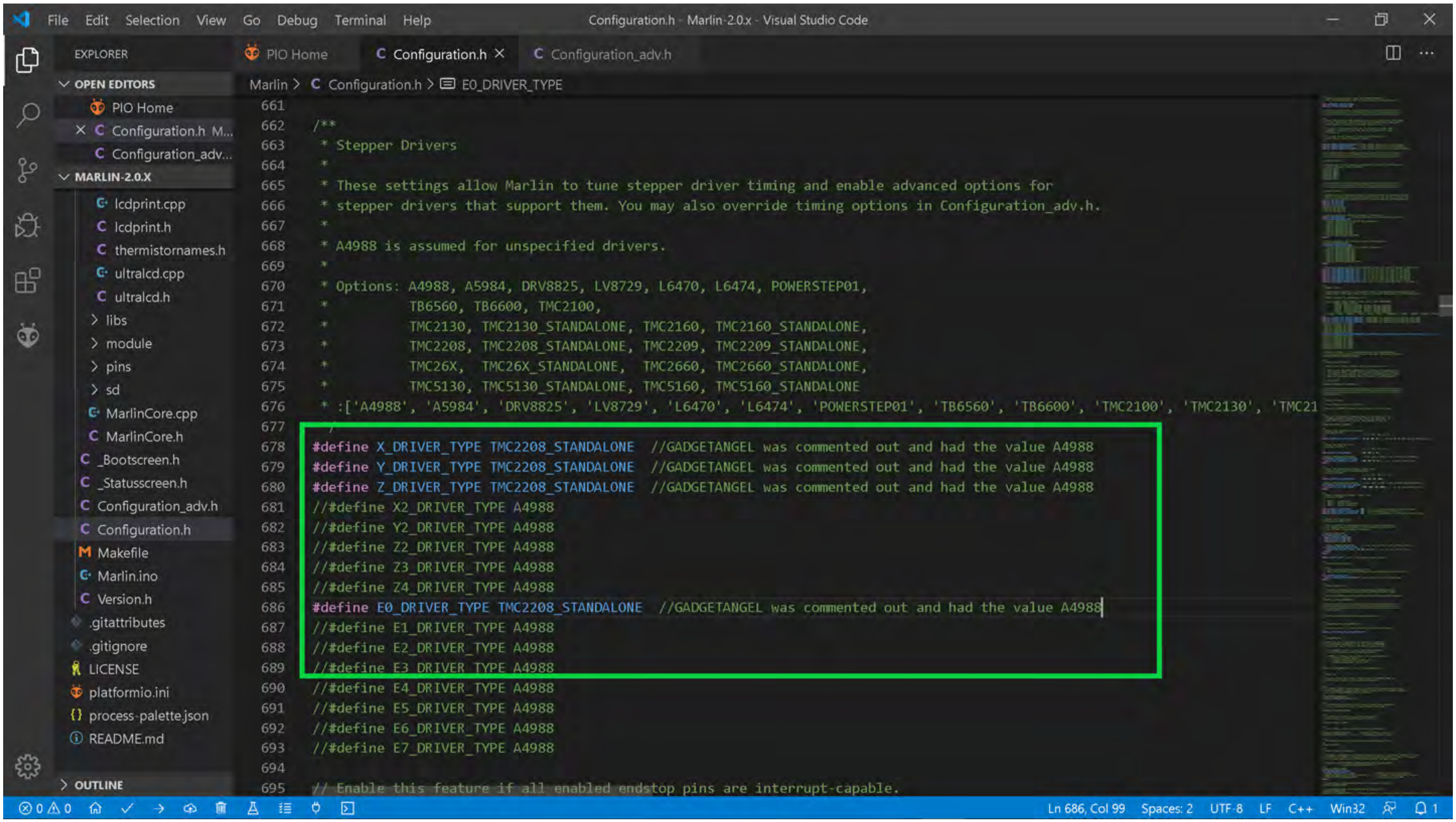


NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

The (latest release of) Marlin Setup for BIQU TMC2225 V1.0 Drivers in Stand-alone Mode

NOTE: **Go to Appendix C**, and then come back here for the changes to Marlin for BIQU TMC2225 V1.0 stepper motor drivers in stand-alone mode.

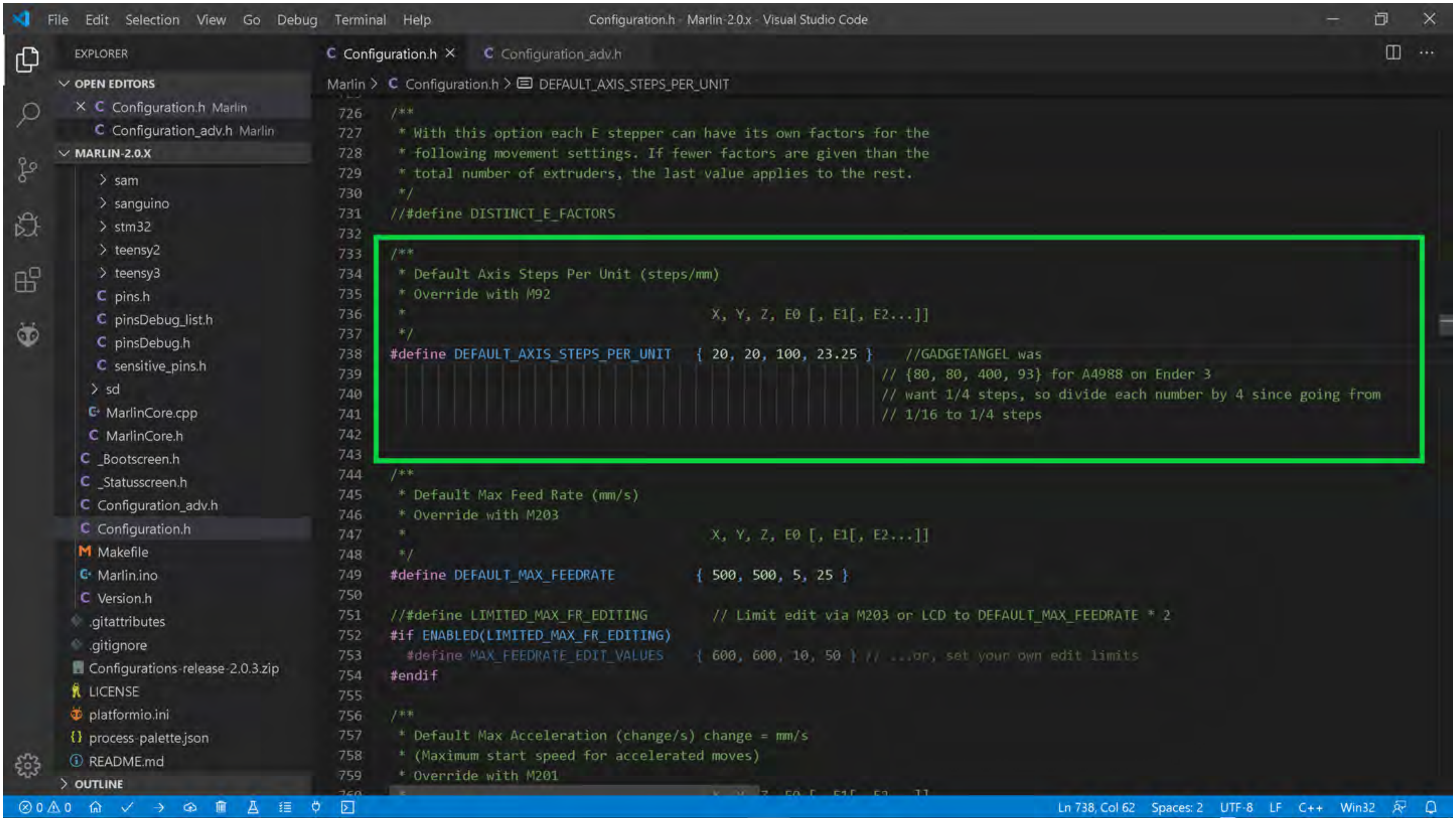
- Change the stepper motor drivers so that Marlin knows you are using BIQU TMC2225 drivers in stand-alone mode. Change one line for each axis and one for each extruder you will be using. See the picture below for an example of how to use TMC2225 drivers in stand-alone mode. Since Marlin does not have an option for TMC2225 drivers we will use "TMC2208_STANDALONE" for the DRIVER_TYPE. When two "//" appear at the beginning of a line that means that line is commented out. To un-comment a line just remove the leading two forward slashes ("//").



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2225 V1.0 Drivers in Stand-alone Mode

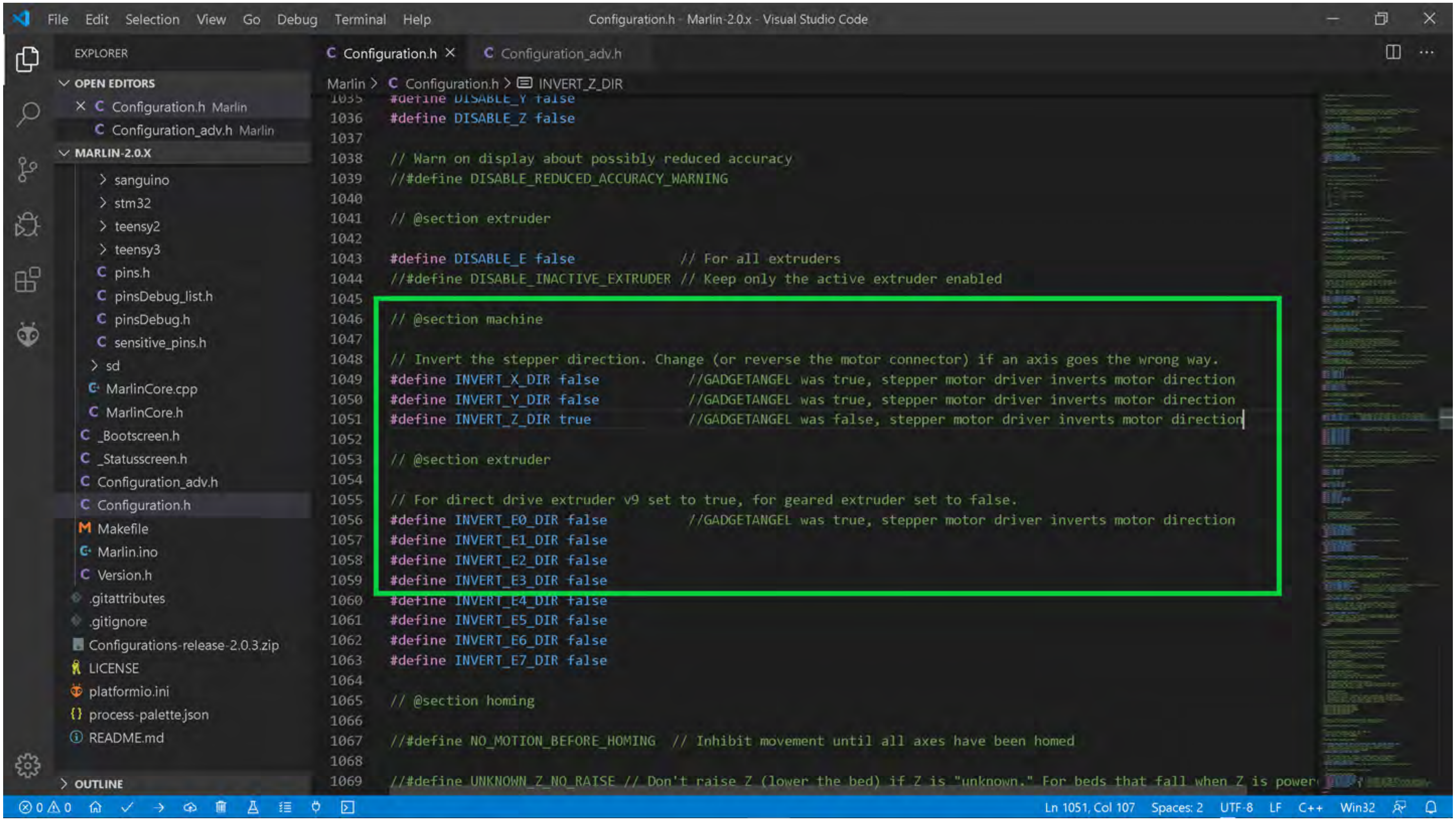
- Since I desire to use 1/4 stepping, and we are changing from A4988 stepper motor drivers on the Ender 3 to TMC2225 (which are exactly like the TMC2208) stepper motor drivers for each axis and the extruder stepper motor driver, we will be going from 1/16 stepping to 1/4 stepping. So we are cutting our STEPS by one quarter. Therefore, we must adjust our "DEFAULT_AXIS_STEPS_PER_UNIT" anytime our STEPS are NOT 1/16. So change "DEFAULT_AXIS_STEPS_PER_UNIT" to {20, 20, 100, 23.25}, as seen in the **GREEN** box below.



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2225 V1.0 Drivers in Stand-alone Mode

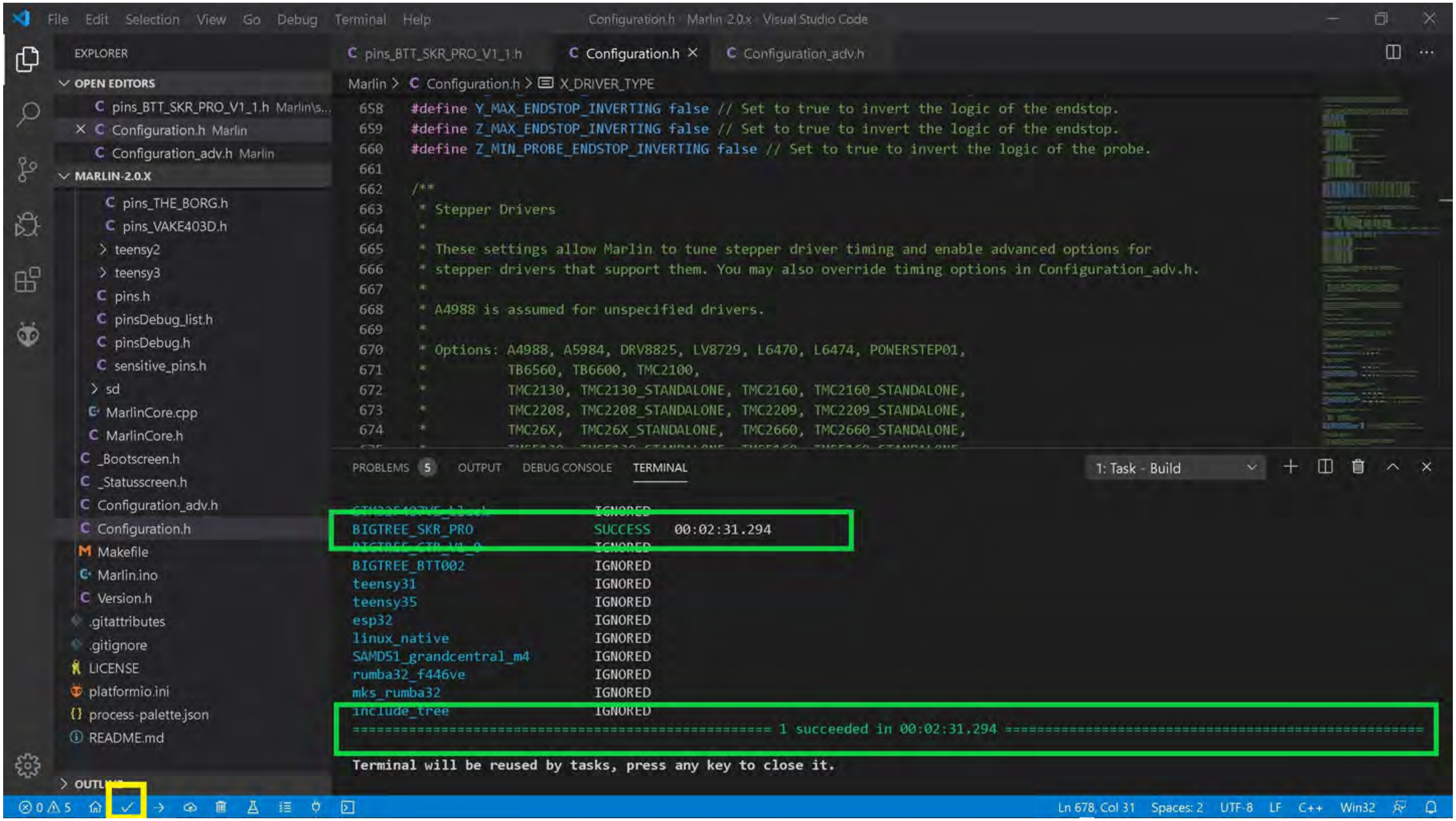
- Since the A4988 driver is what my Ender 3 used, but, now I want to use TMC2225 drivers, I must invert the stepper motor direction because the TMC2225 driver will turn the motors in the opposite direction than the A4988 driver's motor direction. So if the axis' setting you will be using the TMC2225 driver on was "true" change it to "false", as shown in the **GREEN** box below. If the setting was "false", now set it to "true", as shown in the **GREEN** box below



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2225 V1.0 Drivers in Stand-alone Mode

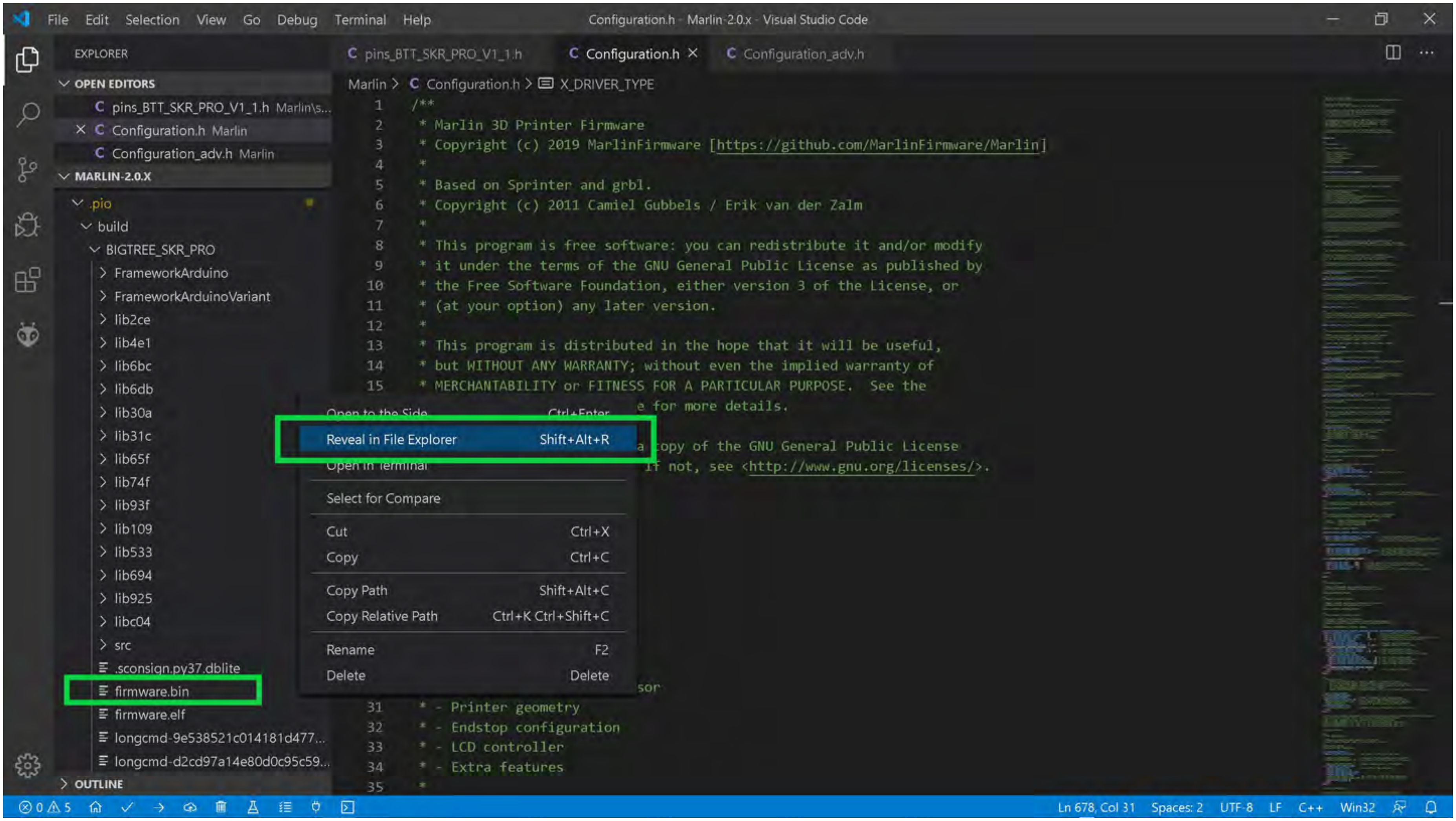
- The end of Marlin setup for BIQU TMC2225 V1.0 drivers in stand-alone mode. Now, compile your code by clicking on the check mark, as seen in the **YELLOW** box below. Once the firmware has compiled successfully you will see the following messages, as shown in the **GREEN** boxes below.



- For the SKR PRO V1.1 board you must save the "firmware.bin" file to the micro SD card then place the micro SD card into the micro SD card reader of the SKR PRO V1.1 board and turn on the power to your printer for the NEW firmware to be automatically applied to the SKR PRO MCU. To locate the "firmware.bin" file, and learn how to transfer it to your micro SD card go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2225 V1.0 Drivers in Stand-alone Mode

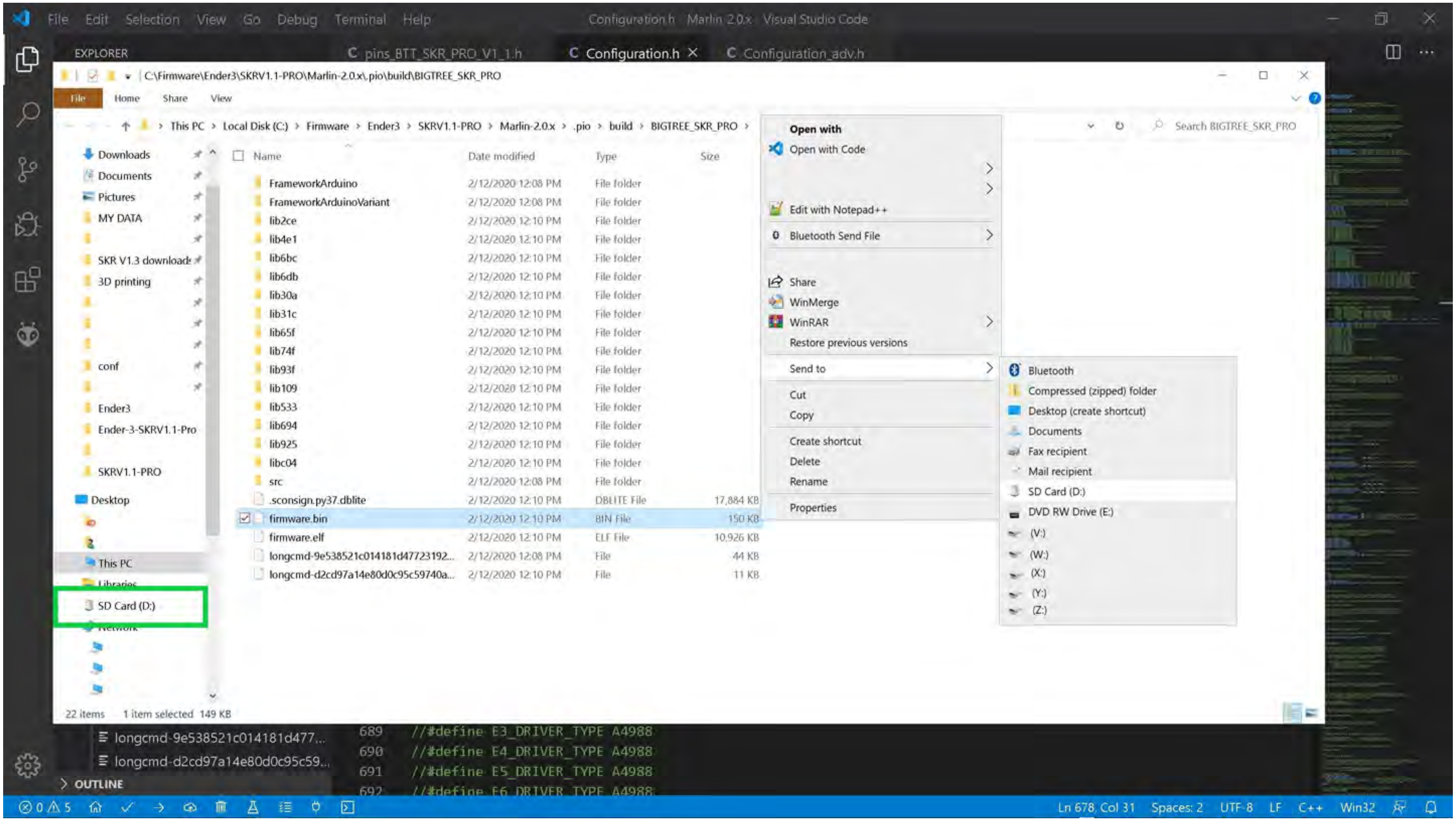
- The "firmware.bin" file can be found by using the left side of the VS code window and going to folder Marlin-2.0.X/.pio/build/ BIGTREE_SKR_PRO/ and right clicking on the "firmware.bin" file name to bring up the content menu. Select "Reveal in File Explorer" to have your Window's machine open a file explorer window.



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2225 V1.0 Drivers in Stand-alone Mode

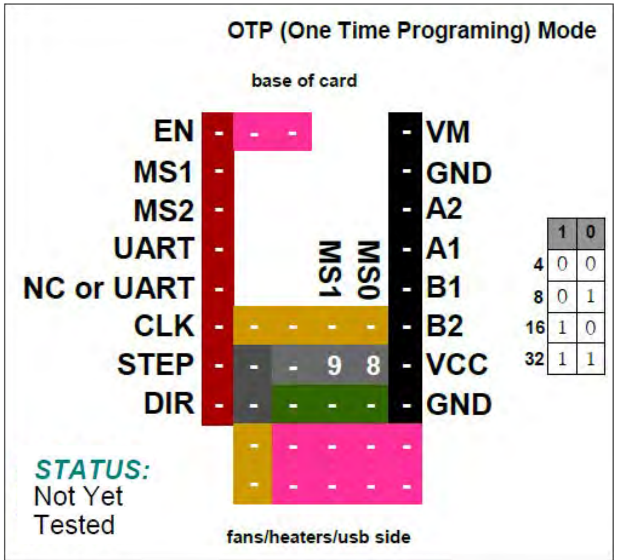
- Ensure your micro SD card, which came with your SKR PRO V1.1 board, with your SKR PRO V1.1 board is in a micro-to-SD-card adapter and plugged into your Window's SD card reader.
  - From the file explorer window right click on the "firmware.bin" file name and select "Send to" from the content menu. For my computer "D:" is my SD card reader. For your computer it may be a different drive letter. The "firmware.bin" file will be written to the SD card and renamed to "FIRMWARE.CUR".



- Right click on the SD card reader, as seen in the **GREEN** box above, and select "Eject" from the context menu. Take the micro SD card out of the adapter and place it into the micro SD card reader of the SKR PRO V1.1 board. Turn on the power to the printer (i.e. the SKR PRO V1.1 board) and the "FIRMWARE.CUR" file will be uploaded to the board. After the upload the firmware file will be renamed to "firmware.bin" and stored on the micro SD card.

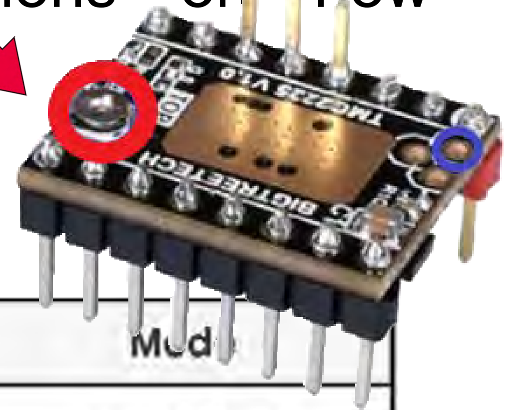
BIQU TMC2225 V1.0

One Time Programming (OTP) Mode



NOTE: Use the potentiometer (POT) on the top of the board, as shown in **RED**; or use the board's "V_{ref} Test point" location, as shown in **BLUE**, to set your V_{ref}. See Appendix A for instructions on how to set the V_{ref} on a driver board.

Note: Use 90% of the calculated V_{ref} when tuning the stepper driver board.

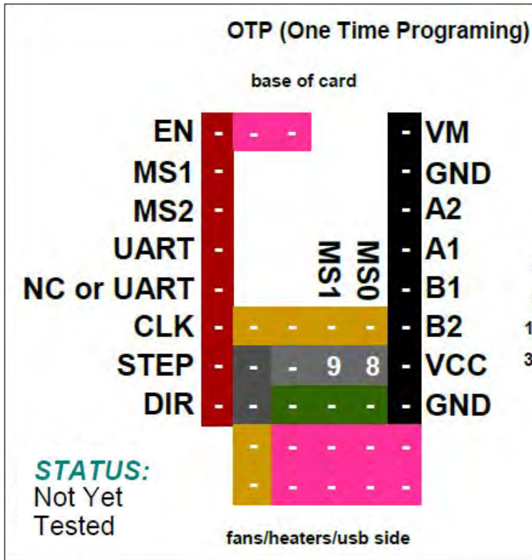


Driver Chip	MS1	MS0	Steps	Interpolation	Mode
BIQU® TMC2225 OTP Mode Maximum 32 Subdivision 35V DC 2A (peak)	GND	GND	1 / 4	1 / 256	spreadCycle
	GND	VIO	1 / 8	1 / 256	spreadCycle
	VIO	GND	1 / 16	1 / 256	spreadCycle
	VIO	VIO	1 / 32	1 / 256	spreadCycle

Driving Current Calculation Formula R_S(Typical Sense Resistor)=0.15Ω	$I_{MAX}=V_{ref}*0.7222$ See Appendix B#10.Use 50% to 90% as shown below: $I_{MAX}=(V_{ref}*0.7222)*0.90$	$V_{ref}=I_{MAX}*1.3846$ See Appendix B#10.Use 50% to 90% as shown below: $V_{ref}=(I_{MAX}*1.3846)*0.90$

BIQU TMC2225 V1.0

One Time Programming (OTP) Mode

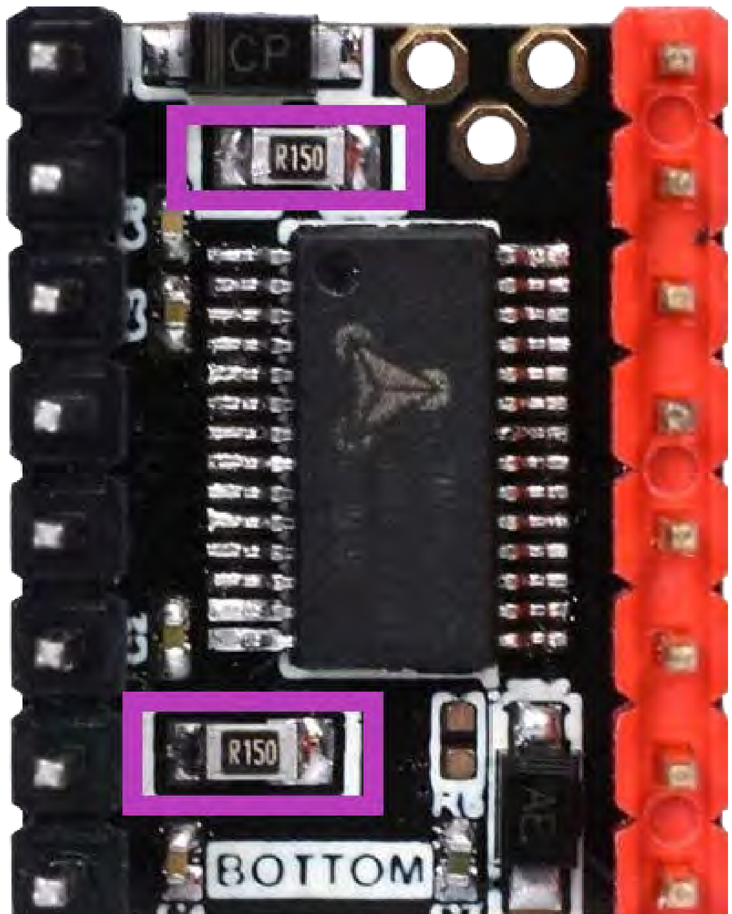


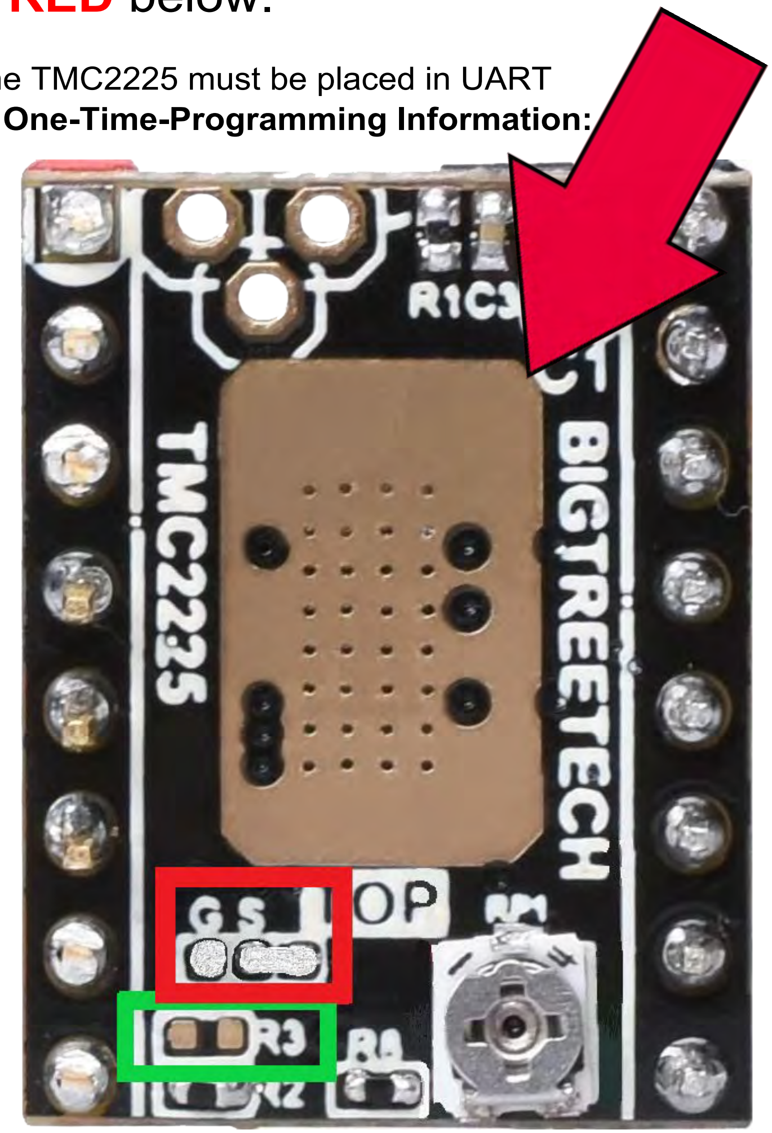
	1	0
4	0	0
8	0	1
16	1	0
32	1	1

Note: Stand-alone mode by default uses stealthChop, if you want spreadCycle, you MUST use OTP mode. To obtain **One Time Programming (OTP) mode,** for the BIQU TMC2225 V1.0, the two pads located at R3 must have a gap between them, as sown in **GREEN** below, and the two pads at "G S" (located on the top of the driver board) must be set for spread Cycle as seen in **RED** below.

To do One-Time-Programming (OTP), the TMC2225 must be placed in UART mode. **See TMC220x Configurator for One-Time-Programming Information:** **TMC220x Configurator.**

The PURPLE boxes shows the location of the current sense resistors (R_s).





Note: **MOST BIQU TMC2225 V1.0 driver boards, when purchased for UART mode, will have two R3 pads (located on the top of the driver board), which are NOT soldered together. This indicates the driver board can use the UART pin for the UART single wire interface (if the UART,"U", jumper is in place on the SKR PRO V1.1 board)**

NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

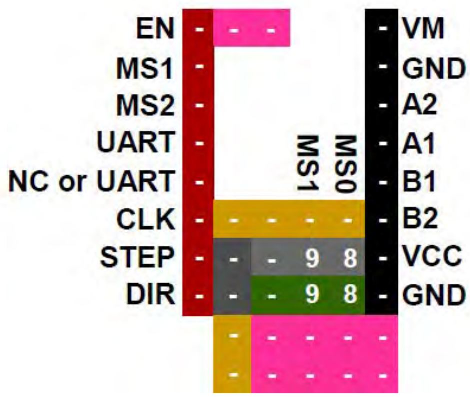
OTP (One Time Programing) Mode

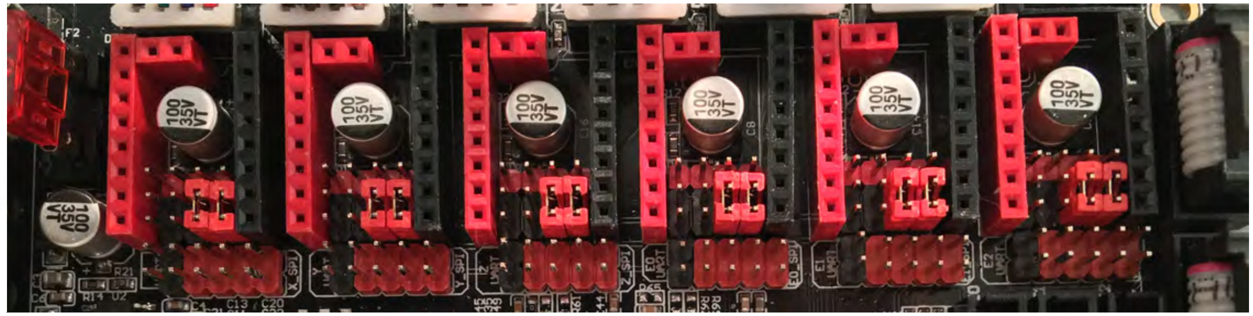
OTP

1 / 4

Interpolation: **1/256**

SpreadCycle



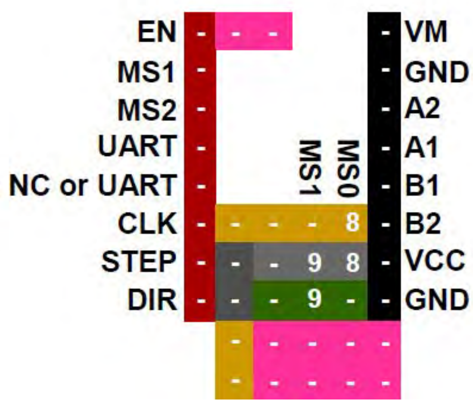


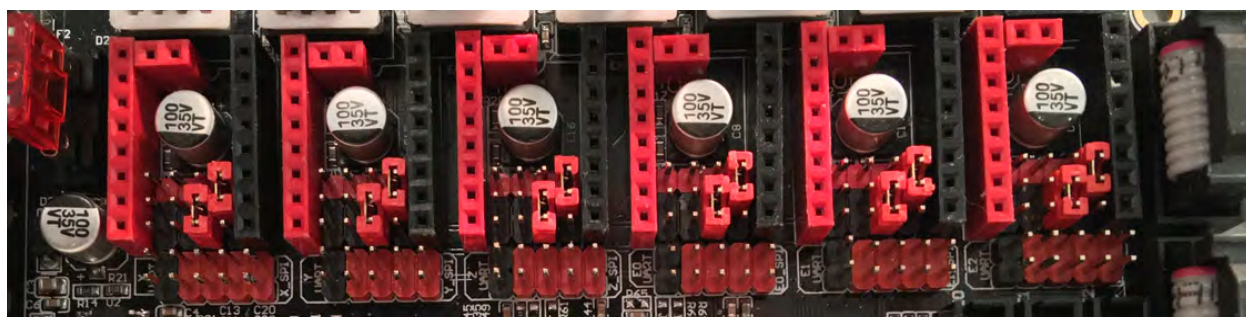
OTP

1 / 8

Interpolation: **1/256**

SpreadCycle





NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

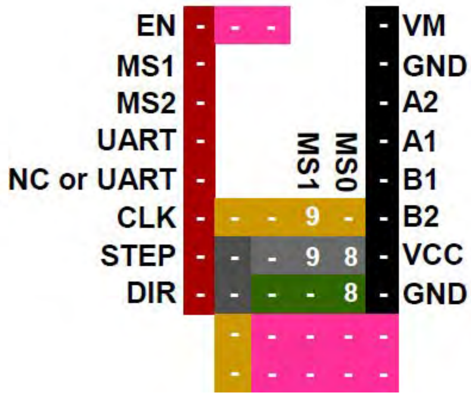
OTP (One Time Programing) Mode

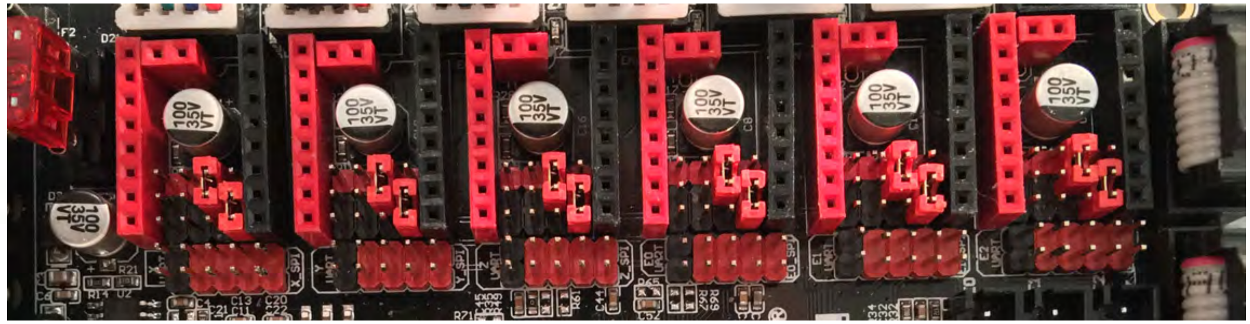
OTP

1 / 16

Interpolation: **1/256**

SpreadCycle



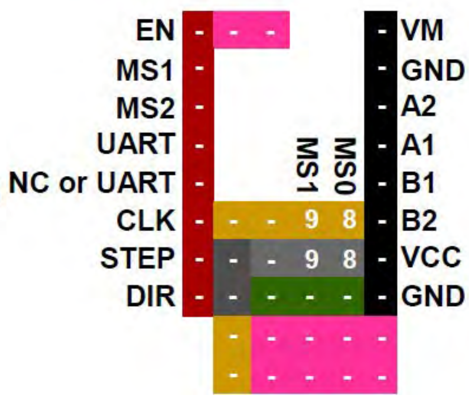


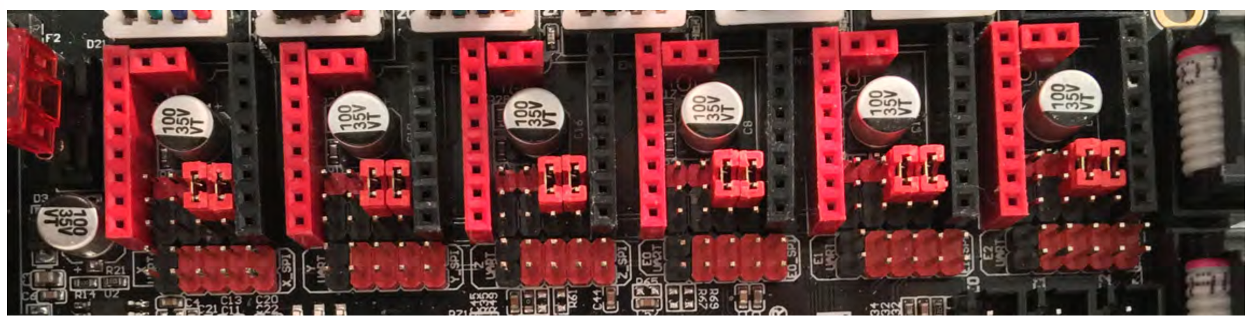
OTP

1 / 32

Interpolation: **1/256**

SpreadCycle

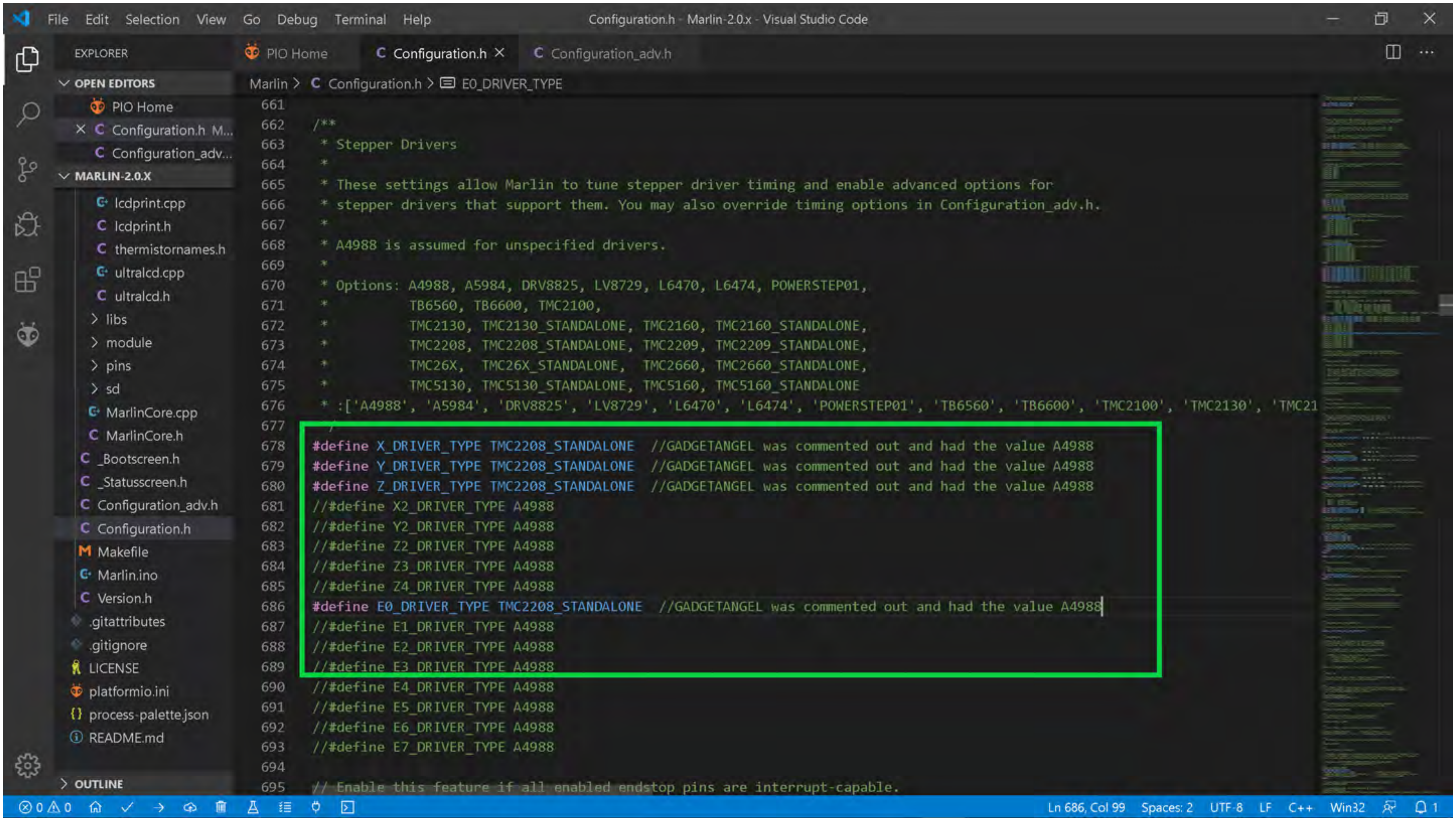




The (latest release of) Marlin Setup for BIQU TMC2225 V1.0 Drivers in One Time Programming (OTP) Mode

NOTE: **Go to Appendix C**, and then come back here for the changes to Marlin for BIQU TMC2225 V1.0 stepper motor drivers in OTP mode.

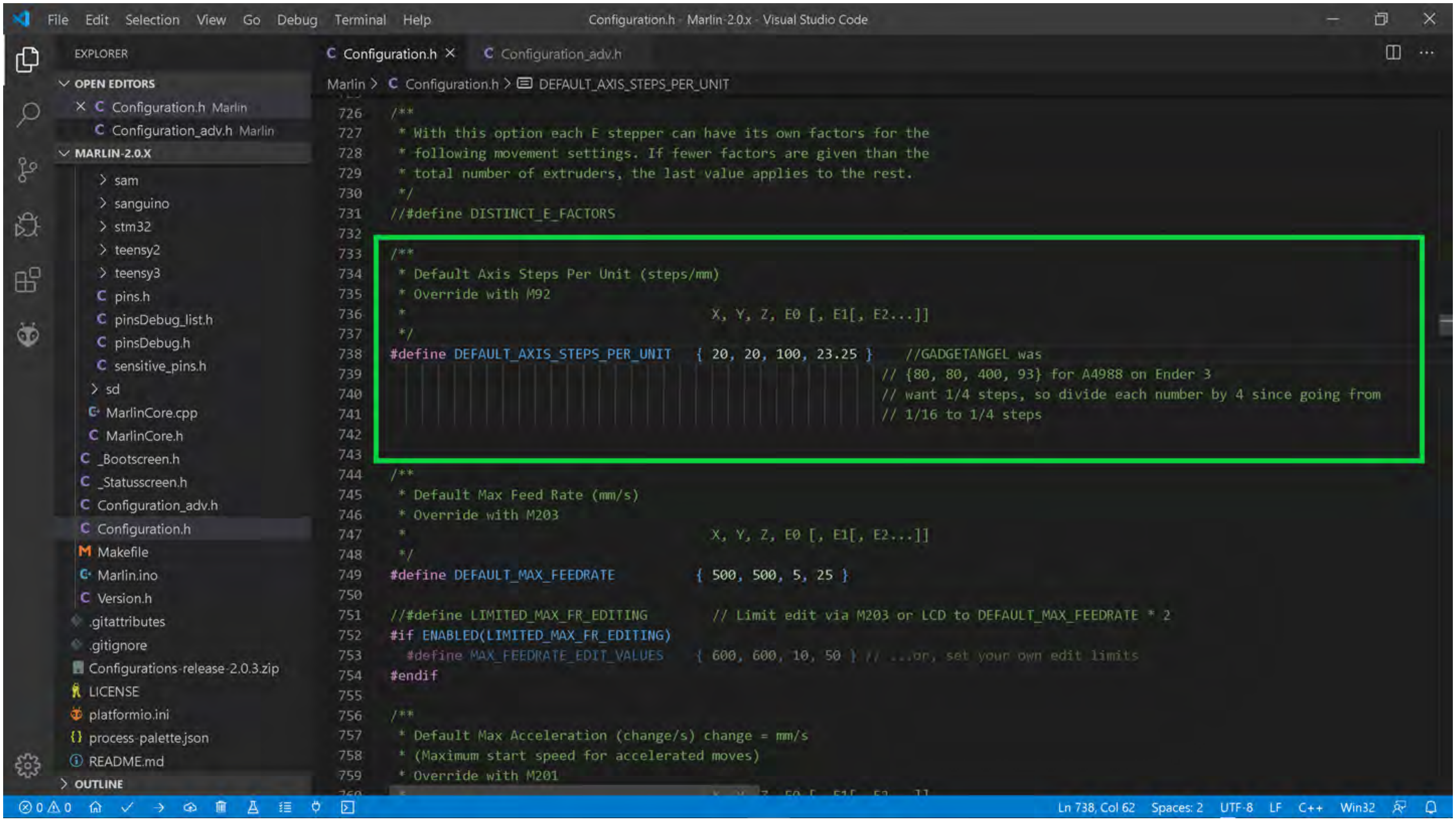
- Change the stepper motor drivers so that Marlin knows you are using BIQU TMC2225 drivers in OTP mode. Change one line for each axis and one for each extruder you will be using. See the picture below for an example of how to use BIQU TMC2225 drivers in OTP mode. Since Marlin does not have an option for TMC2225 drivers we will use "TMC2208_STANDALONE" for the DRIVER_TYPE. When two "//" appear at the beginning of a line that means that line is commented out. To un-comment a line just remove the leading two forward slashes ("//").



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2225 V1.0 Drivers in One Time Programming (OTP) Mode

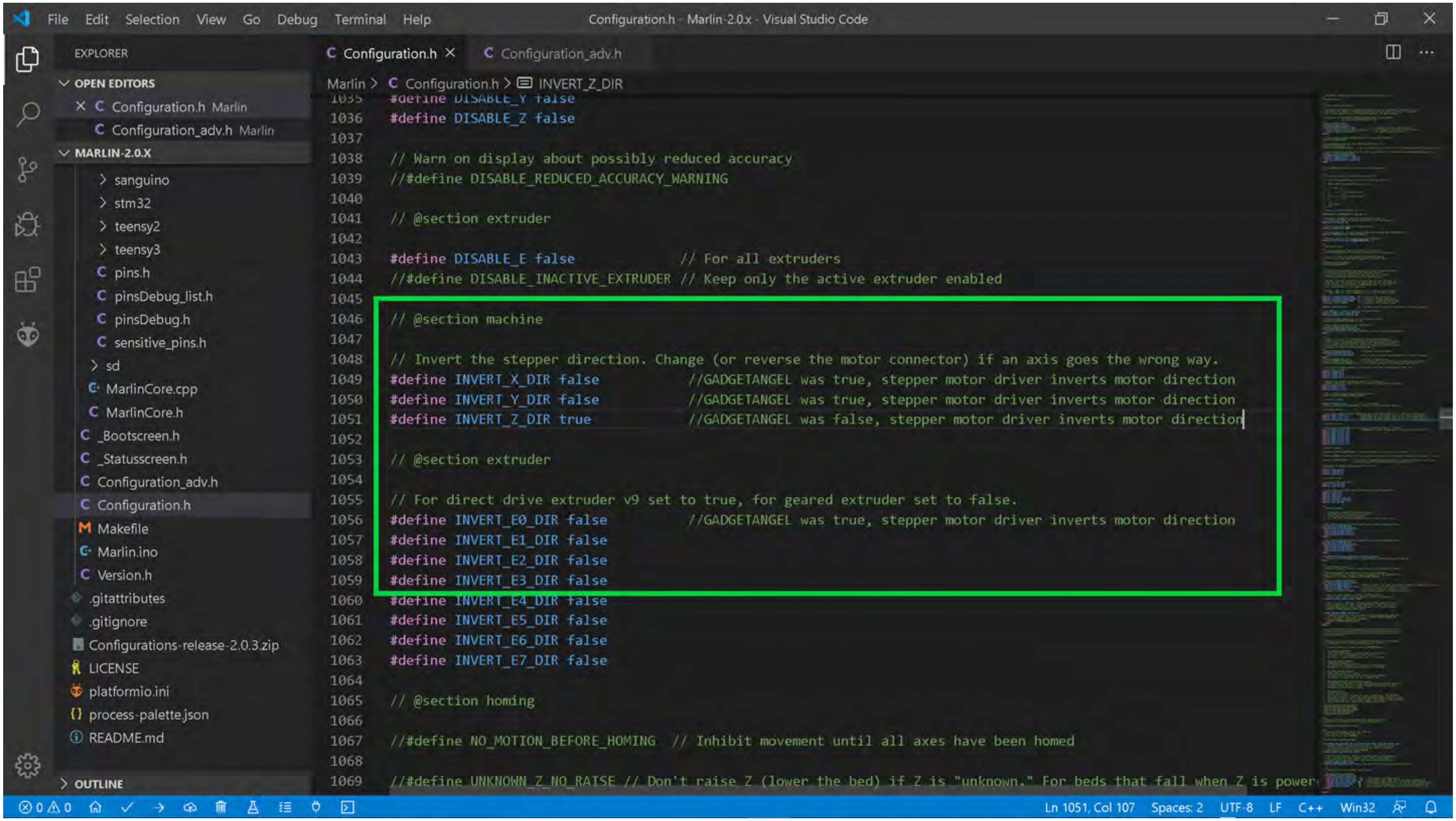
- Since I desire to use 1/4 stepping, and we are changing from A4988 stepper motor drivers on the Ender 3 to TMC2225 (which are exactly like the TMC2208) stepper motor drivers for each axis and the extruder stepper motor driver, we will be going from 1/16 stepping to 1/4 stepping. So we are cutting our STEPS by one quarter. Therefore, we must adjust our "DEFAULT_AXIS_STEPS_PER_UNIT" anytime our STEPS are NOT 1/16. So change "DEFAULT_AXIS_STEPS_PER_UNIT" to {20, 20, 100, 23.25}, as seen in the **GREEN** box below.



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2225 V1.0 Drivers in One Time Programming (OTP) Mode

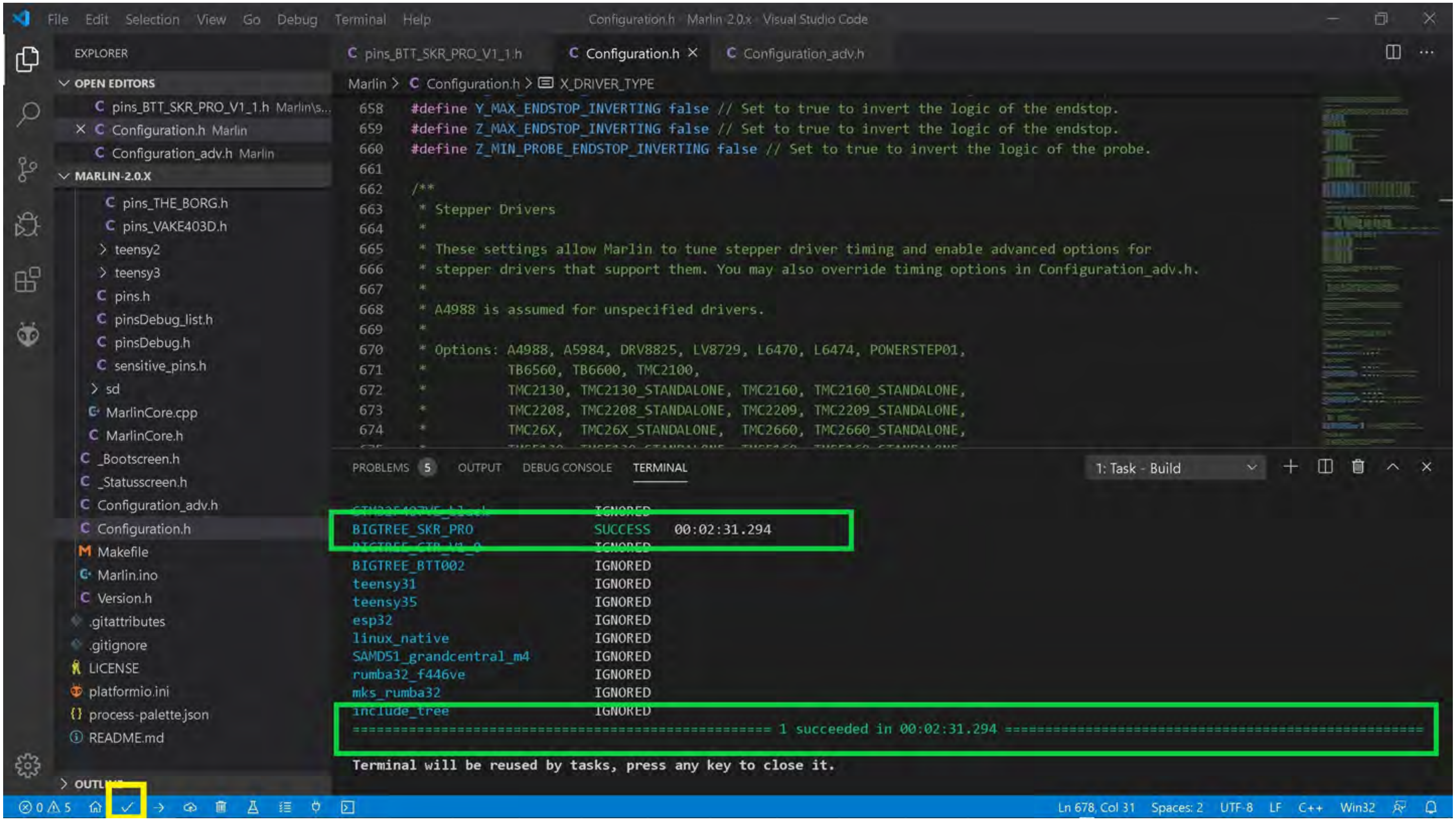
- Since the A4988 driver is what my Ender 3 used, but, now I want to use TMC2225 drivers, I must invert the stepper motor direction because the TMC2225 driver will turn the motors in the opposite direction than the A4988 driver's motor direction. So if the axis' setting you will be using the driver on was "true" change it to "false", as shown in the **GREEN** box below. If the setting was "false", now set it to "true", as shown in the **GREEN** box below



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2225 V1.0 Drivers in One Time Programming (OTP) Mode

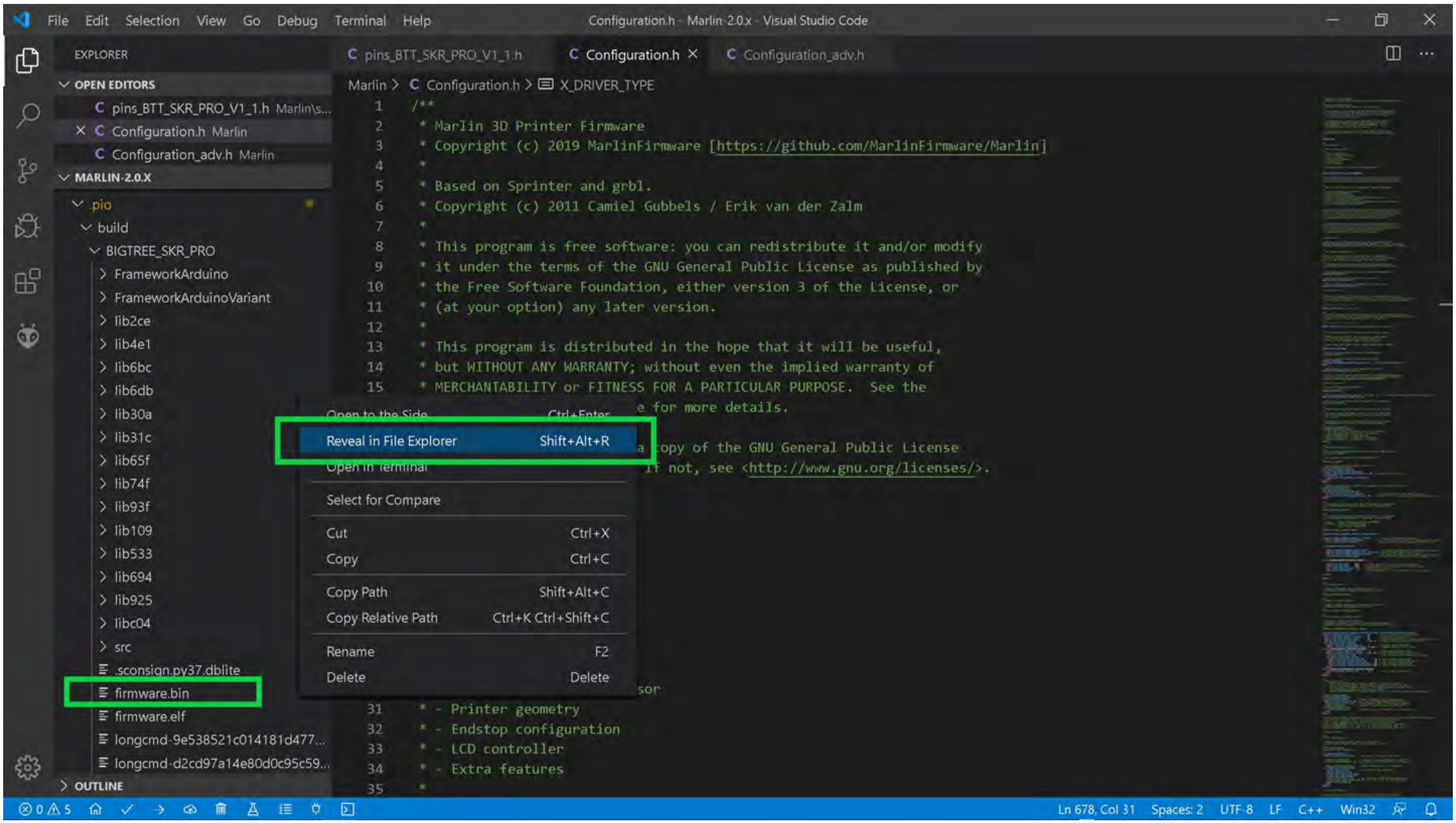
- The end of Marlin setup for BIQU TMC2225 V1.0 drivers in OTP mode. Now, compile your code by clicking on the check mark, as seen in the **YELLOW** box below. Once the firmware has compiled successfully you will see the following messages, as shown in the **GREEN** boxes below.



- For the SKR PRO V1.1 board you must save the "firmware.bin" file to the micro SD card then place the micro SD card into the micro SD card reader of the SKR PRO V1.1 board and turn on the power to your printer for the NEW firmware to be automatically applied to the SKR PRO MCU. To locate the "firmware.bin" file, and learn how to transfer it to your micro SD card go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2225 V1.0 Drivers in One Time Programming (OTP) Mode

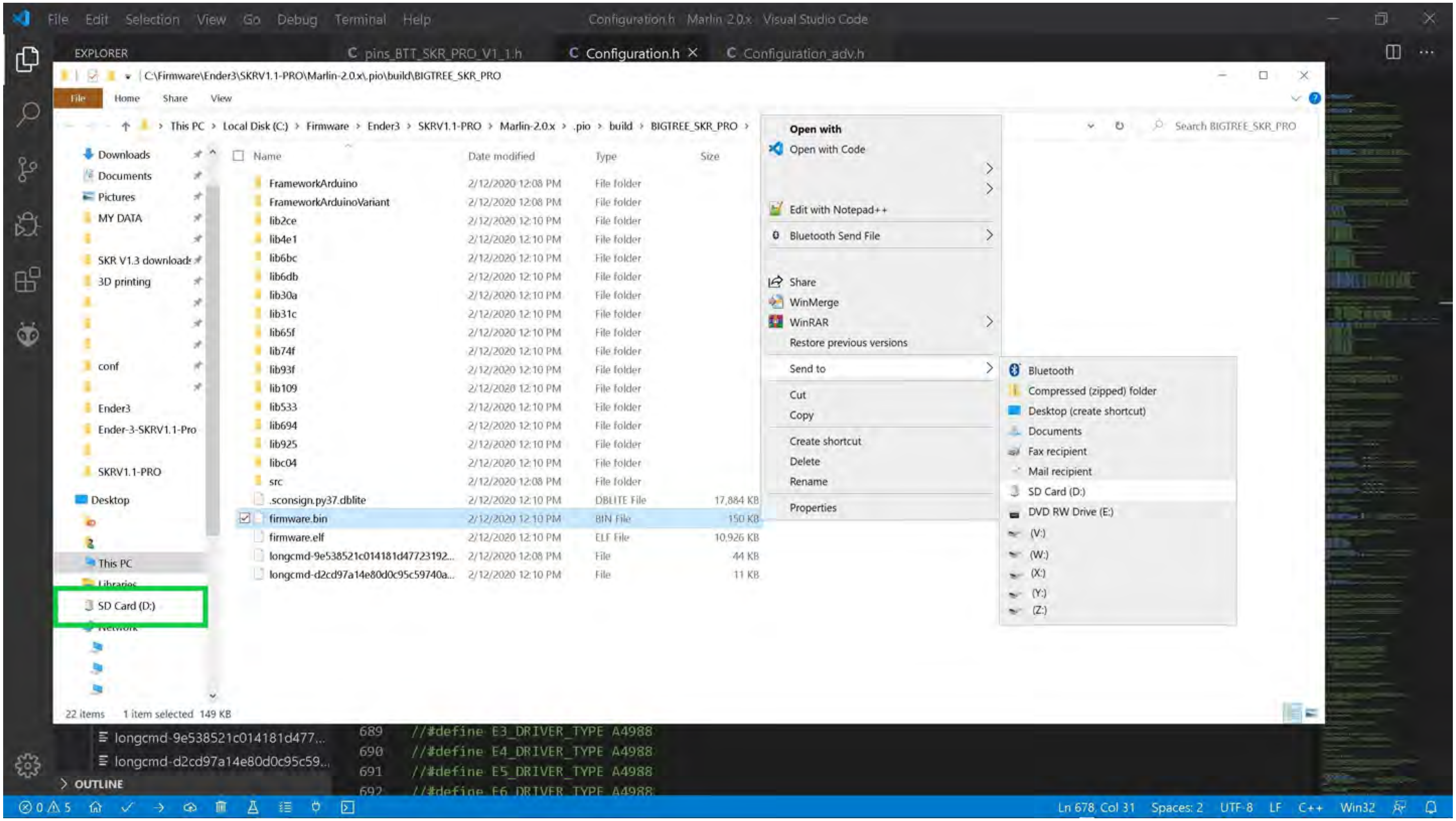
- The "firmware.bin" file can be found by using the left side of the VS code window and going to folder Marlin-2.0.X/.pio/build/ BIGTREE_SKR_PRO/ and right clicking on the "firmware.bin" file name to bring up the content menu. Select "Reveal in File Explorer" to have your Window's machine open a file explorer window.



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2225 V1.0 Drivers in One Time Programming (OTP) Mode

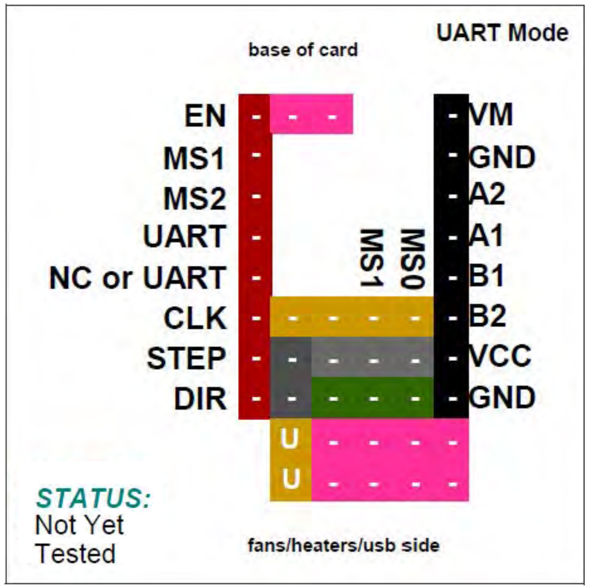
- Ensure your micro SD card, which came with your SKR PRO V1.1 board, with your SKR PRO V1.1 board is in a micro-to-SD-card adapter and plugged into your Window's SD card reader.
    - From the file explorer window right click on the "firmware.bin" file name and select "Send to" from the content menu. For my computer "D:" is my SD card reader. For your computer it may be a different drive letter. The "firmware.bin" file will be written to the SD card and renamed to "FIRMWARE.CUR".



- Right click on the SD card reader, as seen in the **GREEN** box above, and select "Eject" from the context menu. Take the micro SD card out of the adapter and place it into the micro SD card reader of the SKR PRO V1.1 board. Turn on the power to the printer (i.e. the SKR PRO V1.1 board) and the "FIRMWARE.CUR" file will be uploaded to the board. After the upload the firmware file will be renamed to "firmware.bin" and stored on the micro SD card.

BIQU TMC2225 V1.0

UART Mode



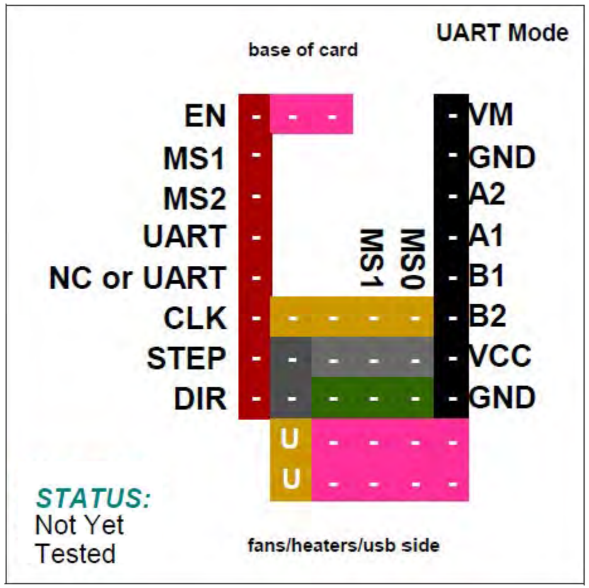
Note: You can use 50% to 90% of the calculated I_{RMS} (I_{MAX}/1.414) when tuning ("X_CURRENT", "Y_CURRENT", etc. the stepper motor driver in the firmware.

See the next page for further information.

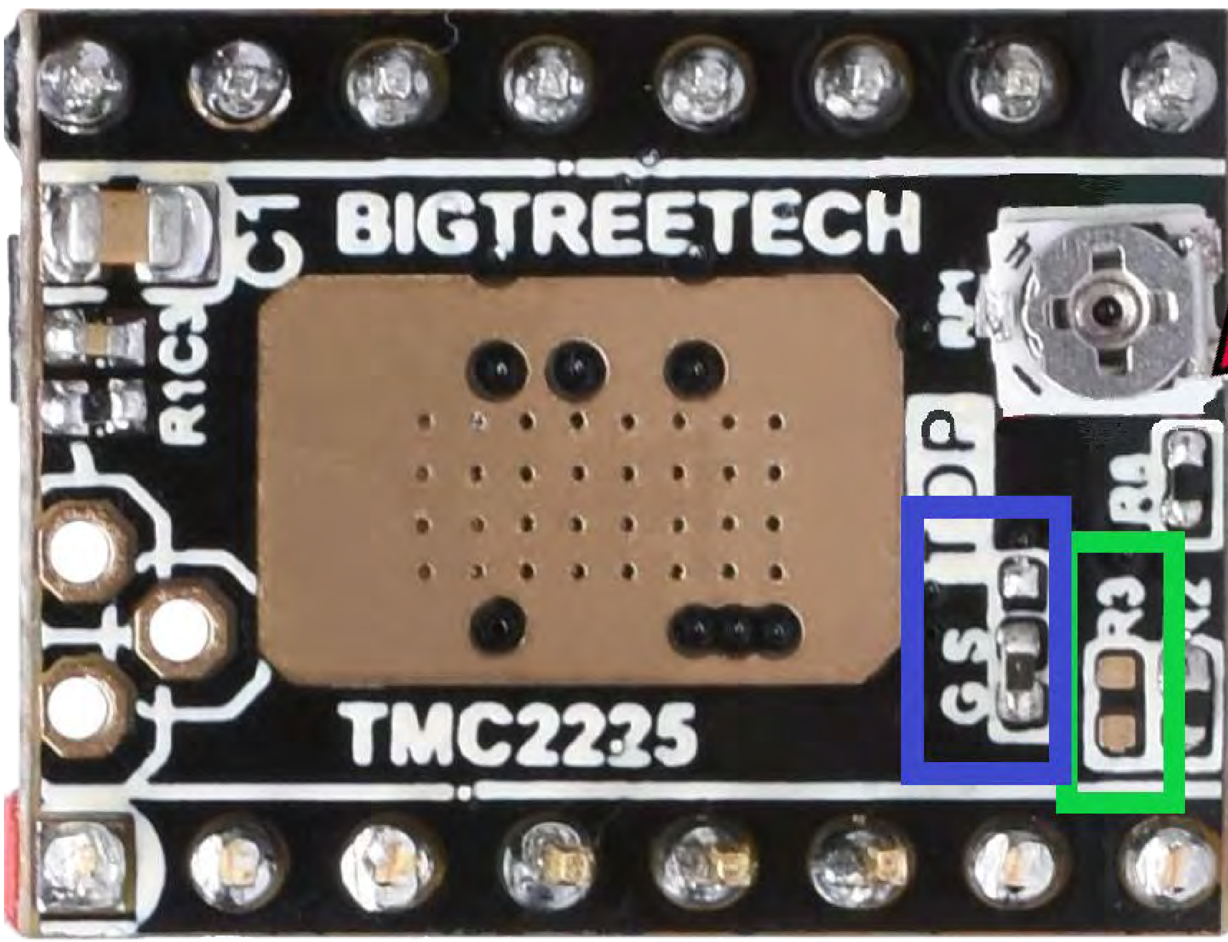
<table>
<tr><td>Driver Chip</td><td colspan="2" rowspan="2">Steps are set inside of your Firmware</td></tr>
<tr><td>BIQU®
TMC2225
UART Mode
Maximum 256 Subdivision
35V DC
2A (peak)</td></tr>
<tr><td>Driving Current Calculation Formula
R_S(Typical Sense Resistor)=0.15Ω</td><td>$I_{MAX}=V_{ref}*0.7647$
See Appendix B #7. Use 50% to 90% as shown below:
$I_{MAX}=(V_{ref}*0.7647)*0.90$</td><td>$V_{ref}=I_{MAX}*1.3077$
See Appendix B #7. Use 50% to 90% as shown below:
$V_{ref}=(I_{MAX}*1.3077)*0.90$</td></tr>
</table>

BIQU TMC2225 V1.0

UART Mode



Important: To ensure that the BIQU TMC2225 V1.0 is in UART Mode, check to see if the two pads located at R3 have a gap between them, as seen in GREEN below. Ensure the "U" jumper is in place on the SKR PRO V1.1 board.



The **BLUE** box show the device has StealthChop capability.

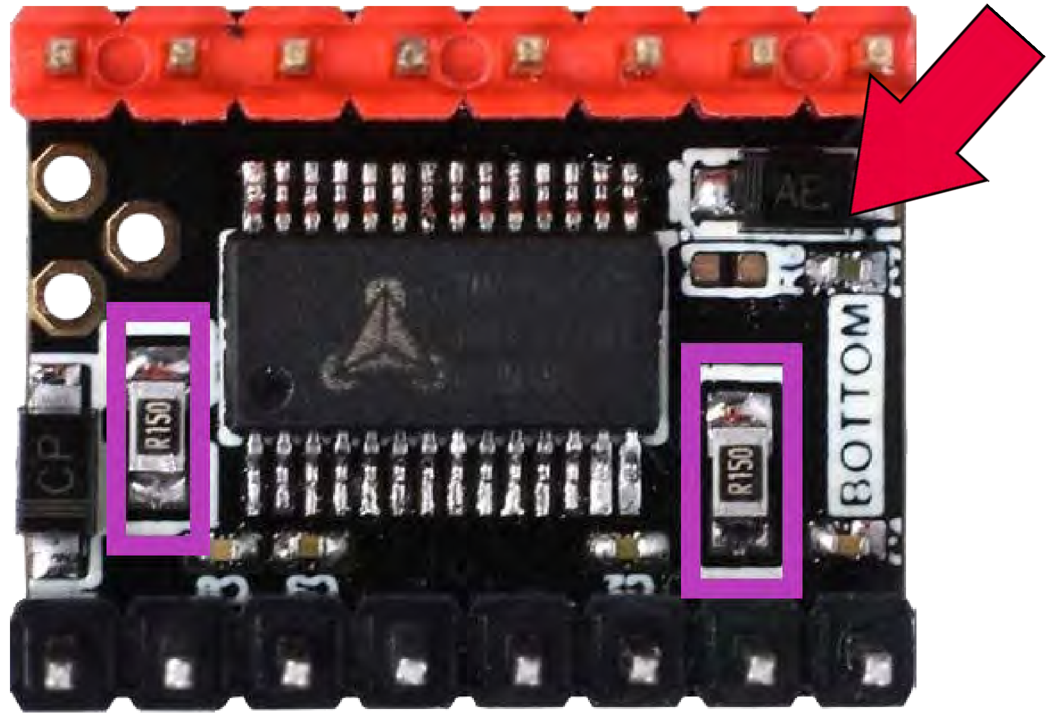
BIQU TMC2225 V1.0

UART Mode

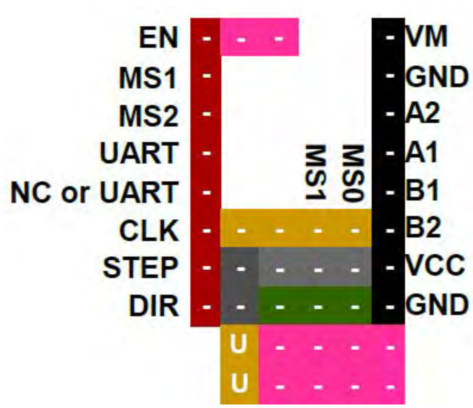
UART Mode

Note: The location of the current sense resistors are shown in **GREEN**. Use the current sense resistors' value in the Marlin Firmware ("X_RSENSE", "Y_RSENSE", "Z_RSENSE" and/or "E0_RSENSE") so that the appropriate current limit can be sent to the driver board. If you do not want to use V_{ref} as the value for "X_CURRENT", "Y_CURRENT", "Z_CURRENT" and/or "E0_CURRENT", you should use I_{RMS} instead. You find I_{RMS} by taking I_{MAX} and dividing it by 1.414 ($I_{RMS}=I_{MAX}/1.414$). You use 50% to 90% of the calculated I_{RMS} as the value for "X_CURRENT", "Y_CURRENT", "Z_CURRENT", and/or "E0_CURRENT".

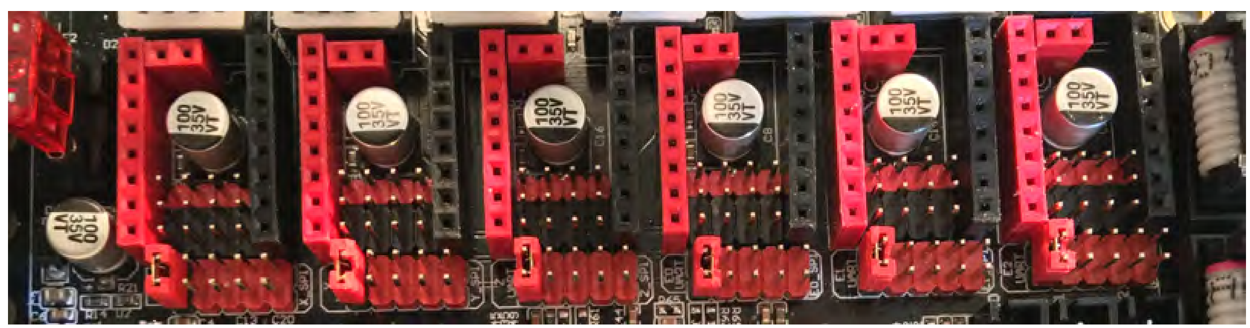
R_s = R050 is 0.05 Ohms
R_s = R062 is 0.062 Ohms
R_s = R068 is 0.068 Ohms
R_s = R075 is 0.075 Ohms
R_s = R100 is 0.1 Ohms
R_s = R110 is 0.11 Ohms
R_s = R150 is 0.15 Ohms
R_s = R200 is 0.2 Ohms
R_s = R220 is 0.22 Ohms



UART



Note: Set Jumper "U" for UART MODE!!

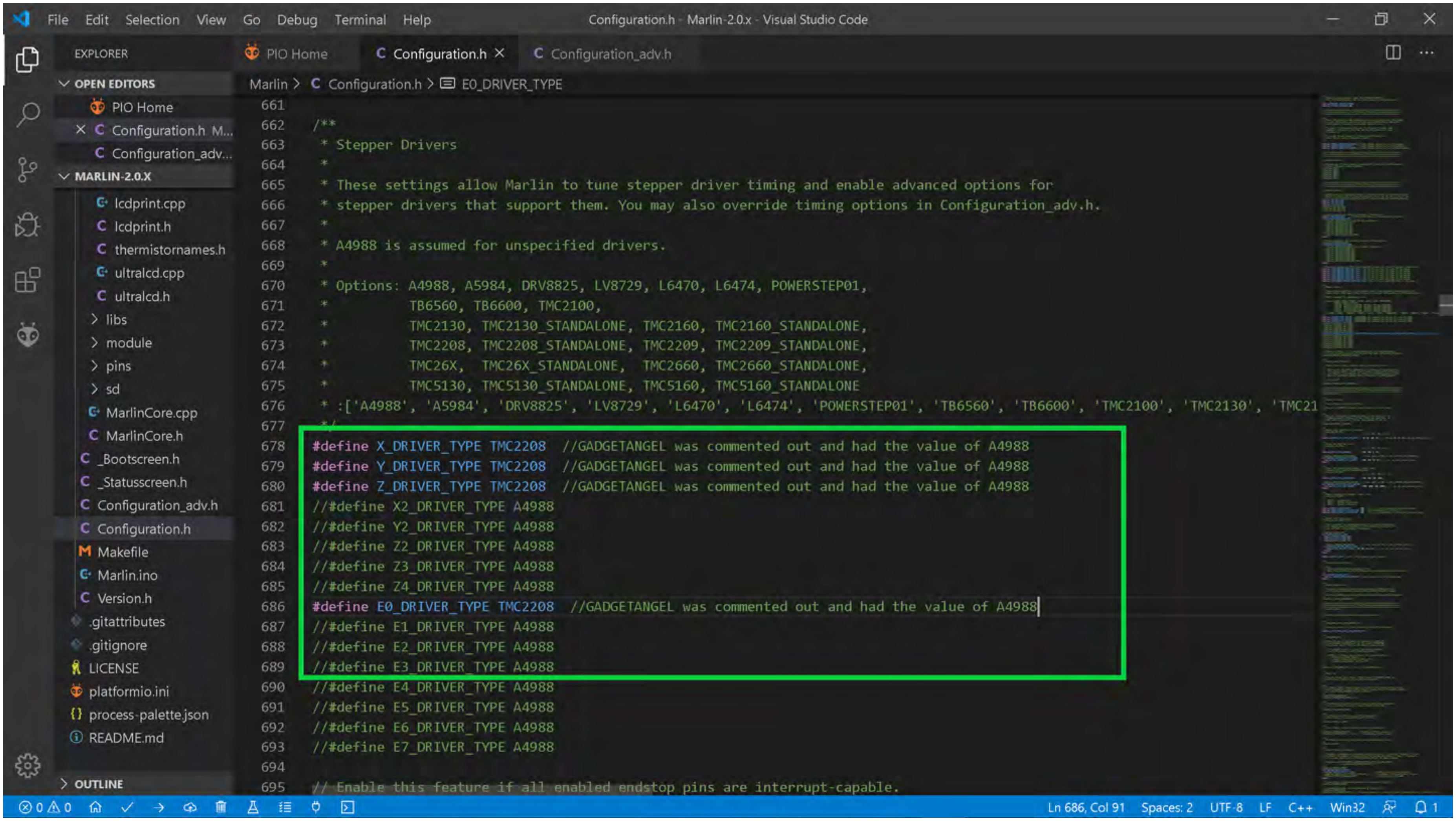


NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

The (latest release of) Marlin Setup for BIQU TMC2225 V1.0 Drivers in UART Mode

NOTE: **Go to Appendix C**, and then come back here for the changes to Marlin for BIQU TMC2225 V1.0 stepper motor drivers in UART mode.

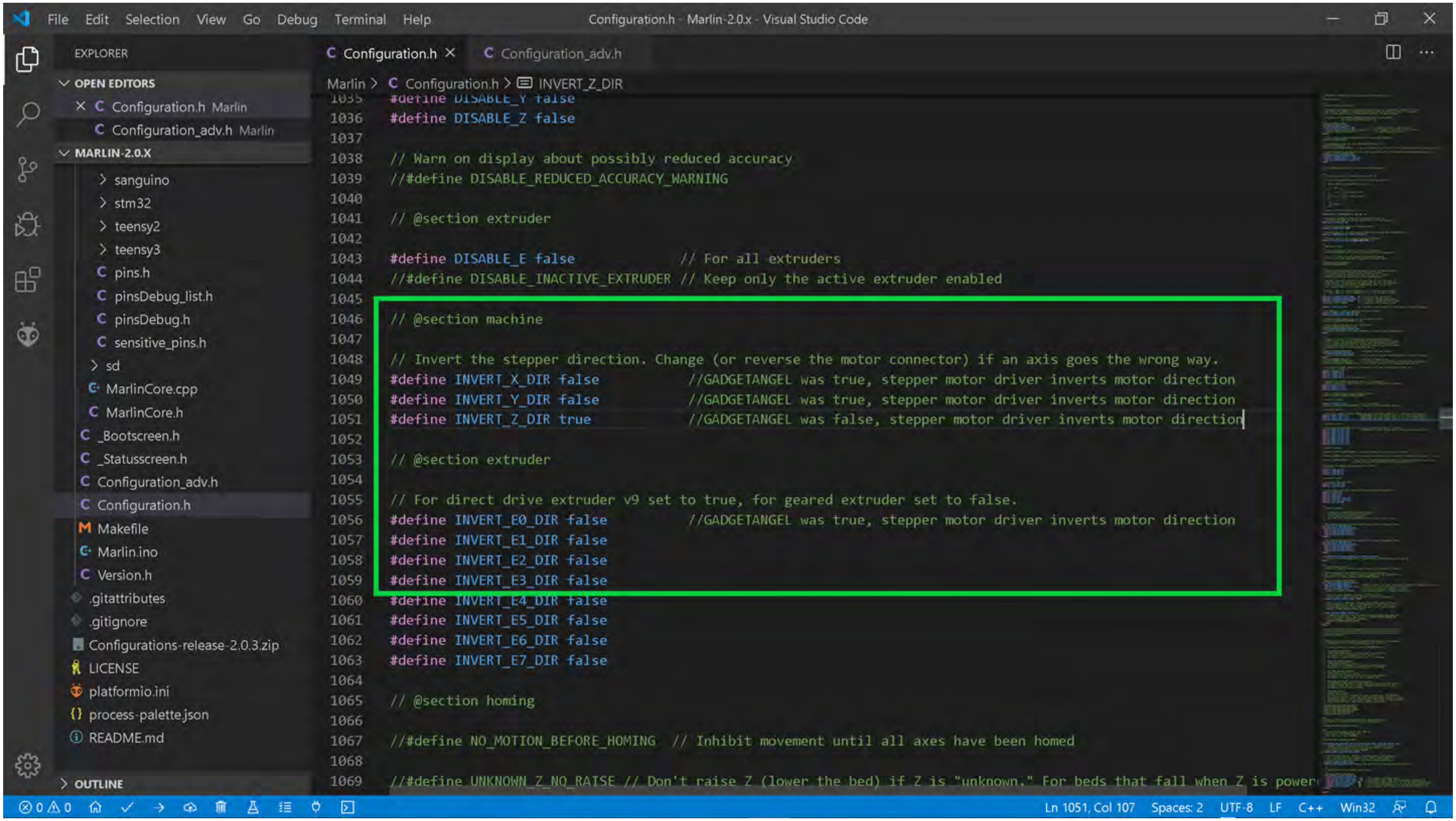
- Change the stepper motor drivers so that Marlin knows you are using TMC2225 drivers in UART mode. Change one line for each axis and one for each extruder you will be using. See the picture below for an example of how to use TMC2225 drivers in UART mode. Since Marlin does not have an option for TMC2225 drivers we will use "TMC2208" for the DRIVER_TYPE. When two "//" appear at the beginning of a line that means that line is commented out. To un-comment a line just remove the leading two forward slashes ("//").



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2225 V1.0 Drivers in UART Mode

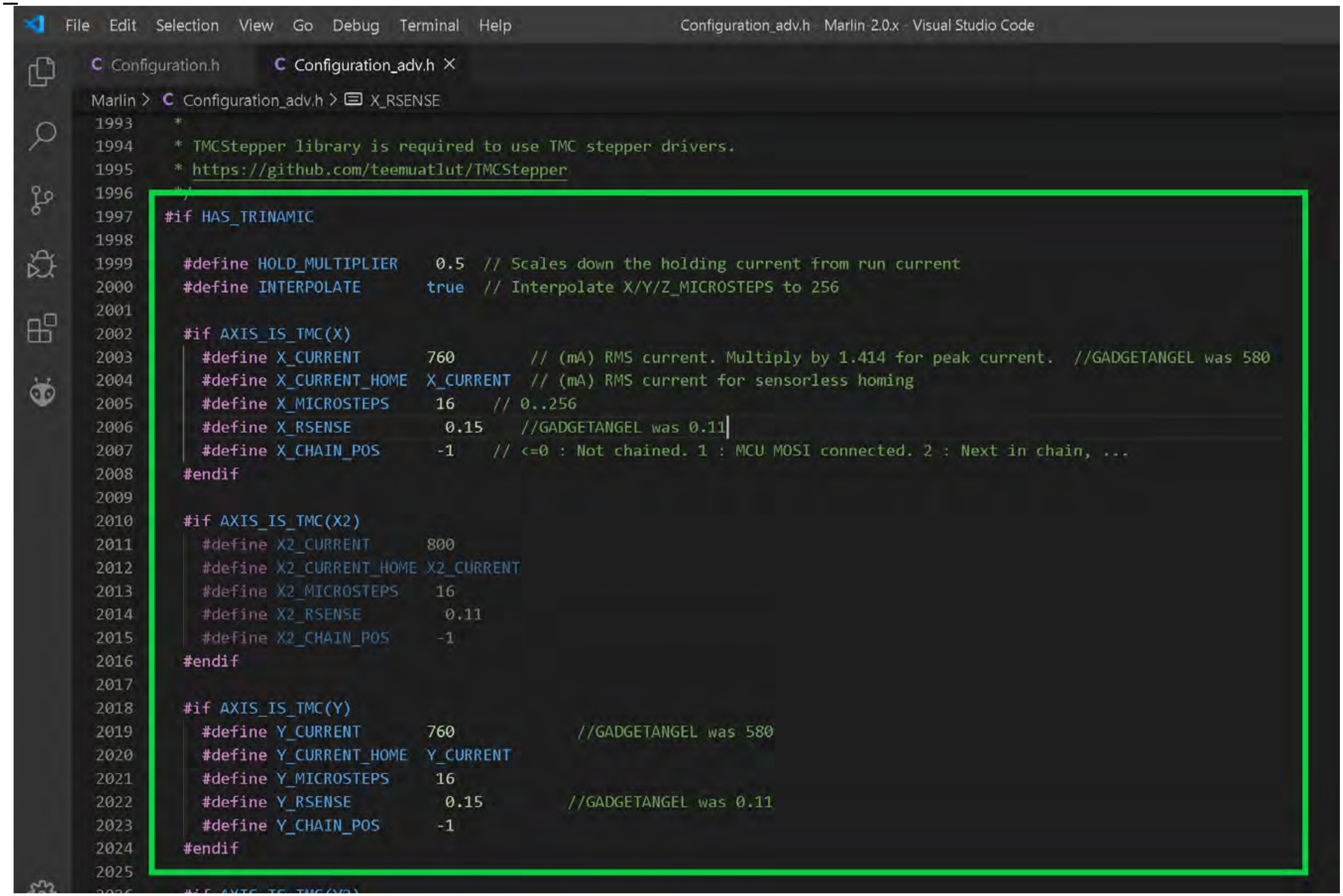
- Since the A4988 driver is what my Ender 3 used, but, now I want to use TMC2225 drivers, I must invert the stepper motor direction because the TMC2225 driver will turn the motors in the opposite direction than the A4988 driver's motor direction. So if the axis' setting you will be using the TMC2225 driver on was "true" change it to "false", as shown in the **GREEN** box below. If the setting was "false", now set it to "true", as shown in the **GREEN** box below



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2225 V1.0 Drivers in UART Mode

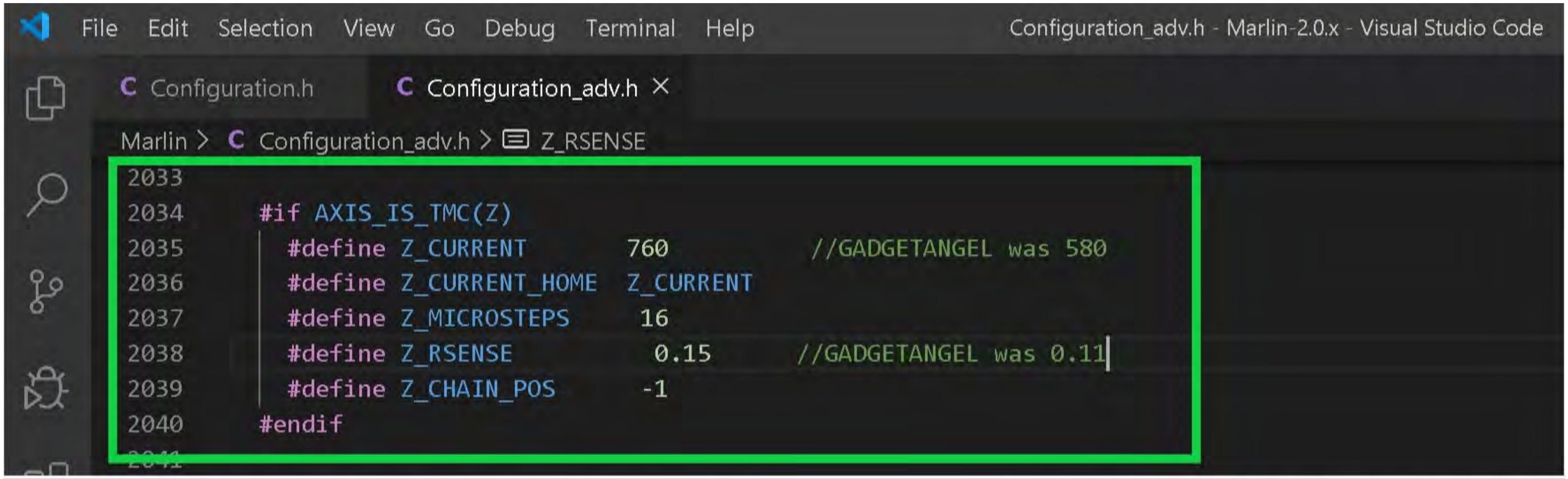
- Next you want to set your V_{ref} in the Marlin firmware for each axis that has the TMC2225 driver, as seen in the **GREEN** box below. I changed the "X_CURRENT" to be the calculated V_{ref} for my X-Axis, which is 760mV for an Ender 3. I changed the "Y_CURRENT" to be the calculated V_{ref} for my Y-Axis, which is 760mV on the Ender 3.
- Ensure "X_RSENSE" is set to 0.15. Ensure "Y_RSENSE" is set to 0.15.
- If you do not want to use V_{ref} as the value for "X_CURRENT" and/or "Y_CURRENT", you should use I_{RMS} instead. You find I_{RMS} by taking I_{MAX} and dividing it by 1.414 ($I_{RMS}=I_{MAX}/1.414$). You use 50% to 90% of the calculated I_{RMS} as the value for "X_CURRENT" and/or "Y_CURRENT".

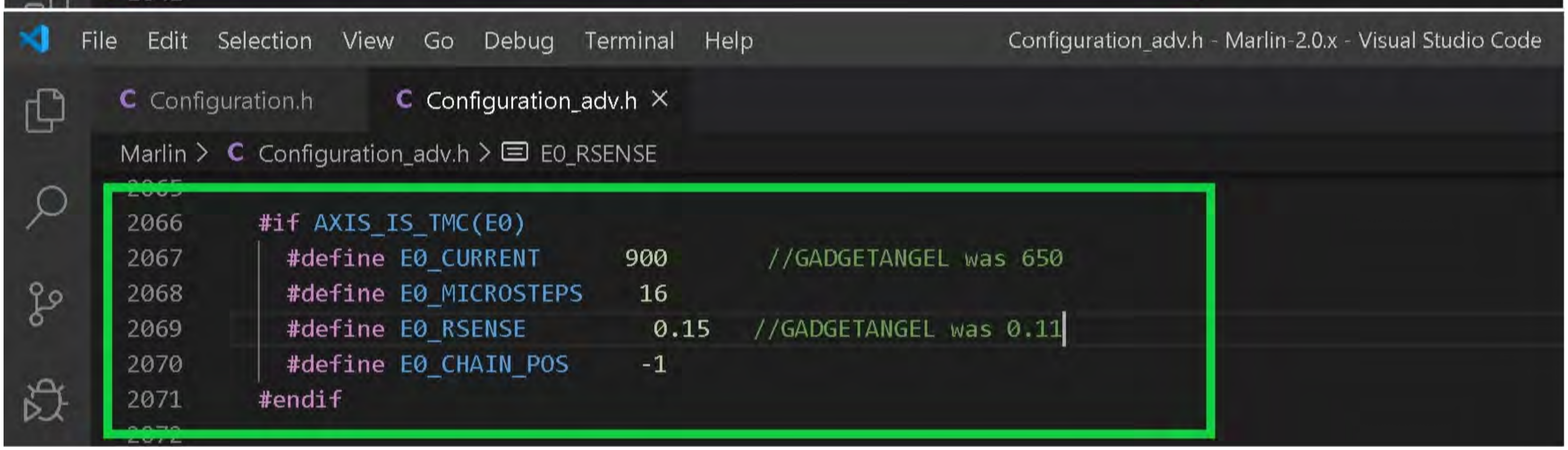


- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2225 V1.0 Drivers in UART Mode

- Now, I am setting the V_{ref} for Z-Axis and the extruder, as seen in the **GREEN** boxes below. I changed the "Z_CURRENT" to be the calculated V_{ref} for my Z-Axis, which is 760mV for an Ender 3. I changed the "E0_CURRENT" to be the calculated V_{ref} for my Extruder, which is 900mV on the Ender 3.
- Ensure "Z_RSENSE" is set to 0.15. Ensure "E0_RSENSE" is set to 0.15.
- If you do not want to use V_{ref} as the value for "Z_CURRENT" and/or "E0_CURRENT", you should use I_{RMS} instead. You find I_{RMS} by taking I_{MAX} and dividing it by 1.414 ($I_{RMS}=I_{MAX}/1.414$). You use 50% to 90% of the calculated I_{RMS} as the value for "Z_CURRENT" and/or "E0_CURRENT".

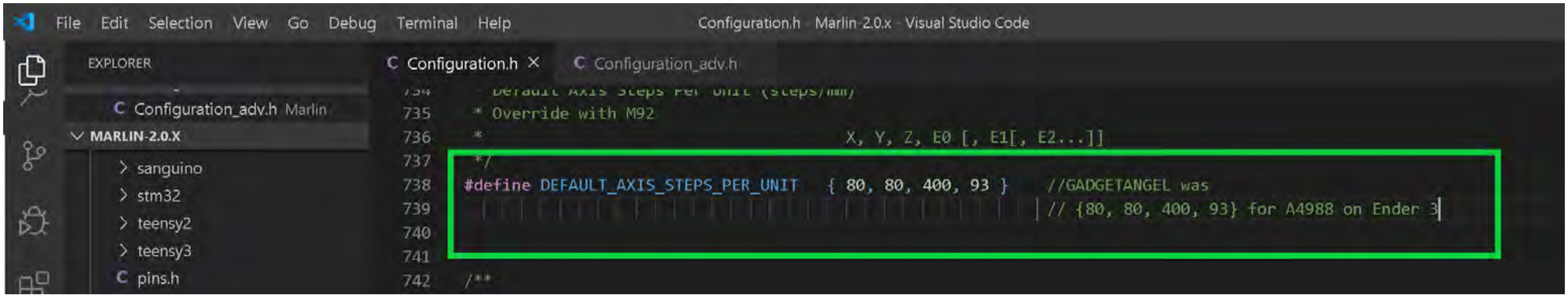




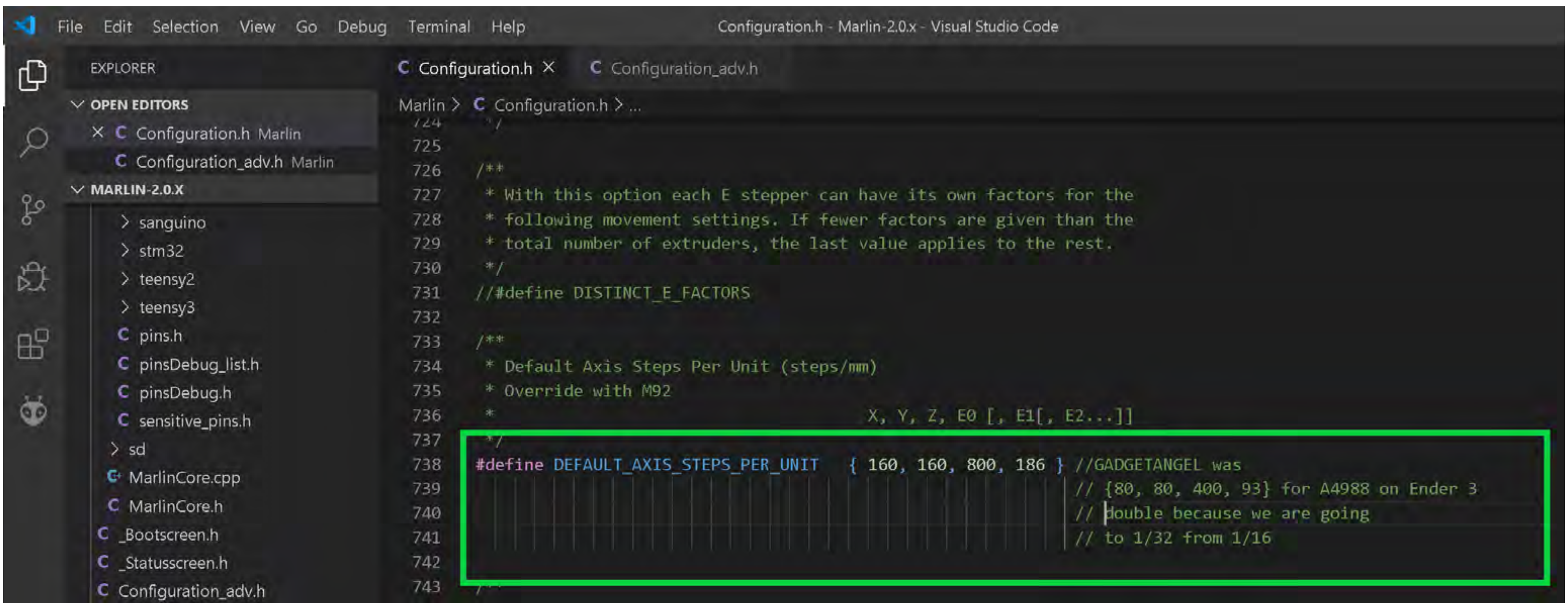
- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2225 V1.0 Drivers in UART Mode

- If you changed the "MICROSTEPS" for any of the axes then you will need to update "DEFAULT_AXIS_STEPS_PER_UNIT" to reflect your changes



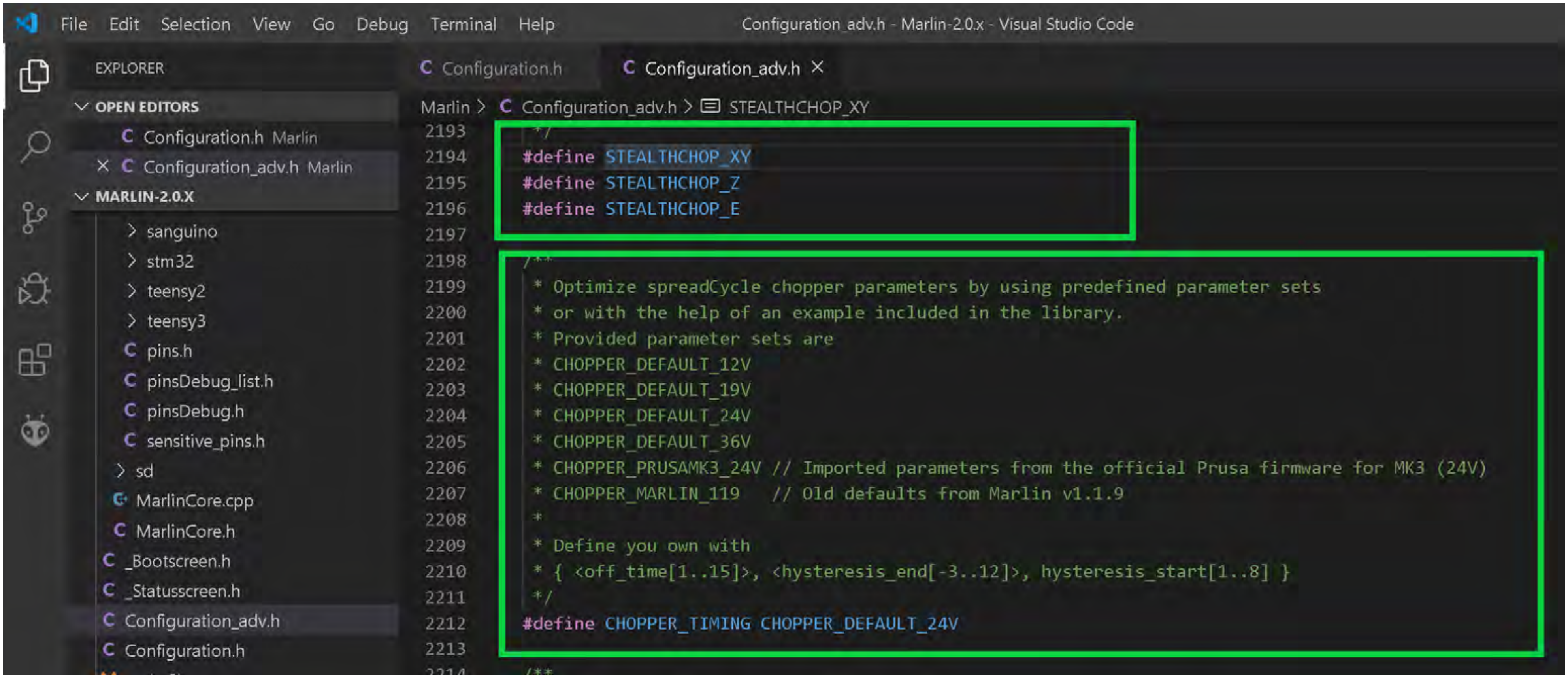
- FOR EAMPLE if you wanted to use 1/32 stepping instead of the default 1/16, you would be **doubling** your STEPS. Therefore, we must adjust our "DEFAULT_AXIS_STEPS_PER_UNIT" anytime our STEPS are NOT 1/16. So change "DEFAULT_AXIS_STEPS_PER_UNIT" to {160, 160, 800, 186}, as seen in the **GREEN** box below.



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2225 V1.0 Drivers in UART Mode

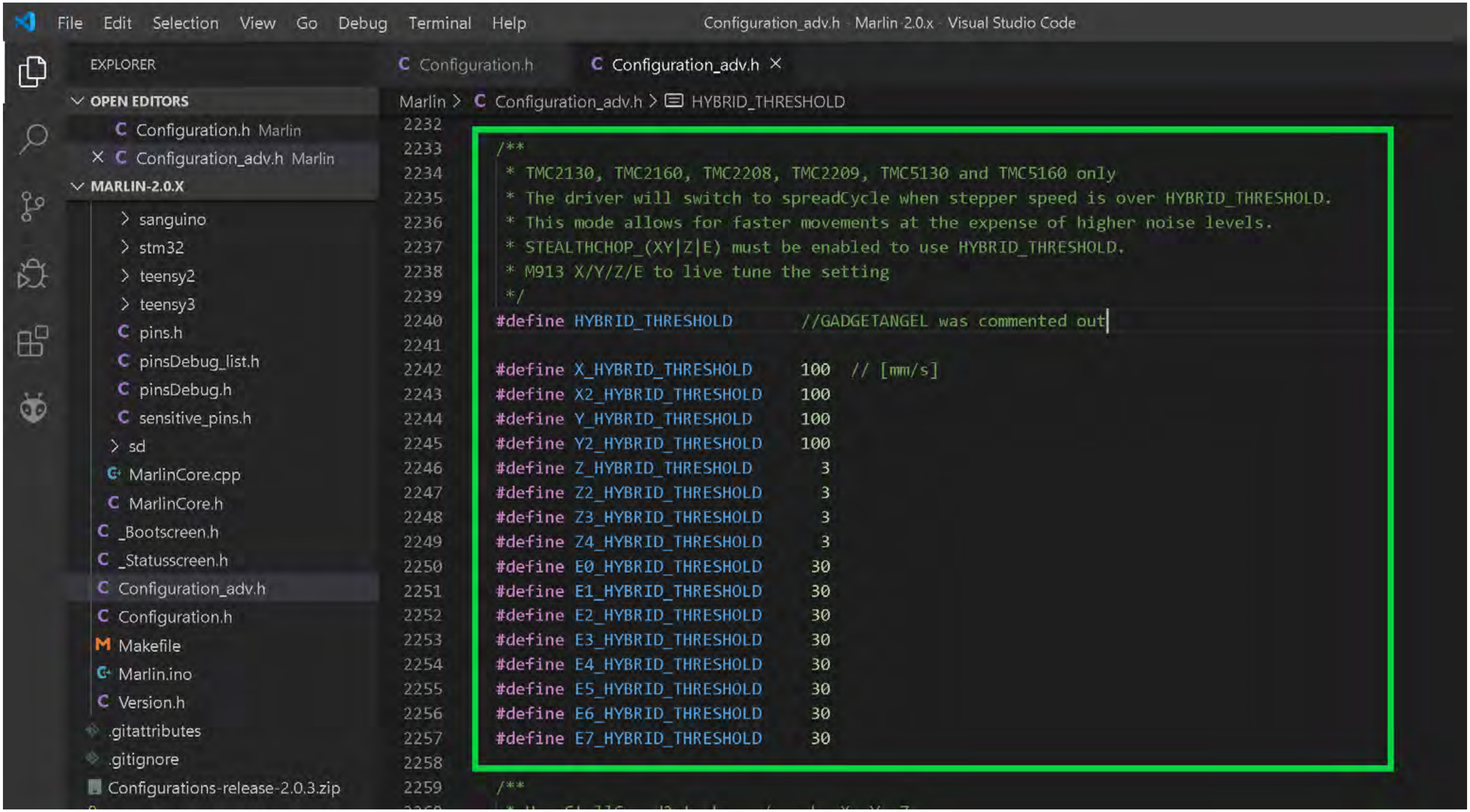
- By default stealthChop is enabled in the Marlin firmware. If you want spreadCycle ONLY then comment out the appropriate lines. I want stealthChop enabled so I want to make sure the lines are not commented out {"STEALTHCHOP_XY", "STEALTHCHOP_Z" and "STEALTHCHOP_E"}. You also want to check to see if the proper "CHOPPER_TIMING" is set for your printer. An Ender 3 is a 24VDC printer, my "CHOPPER_TIMING" is correct.



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2225 V1.0 Drivers in UART Mode

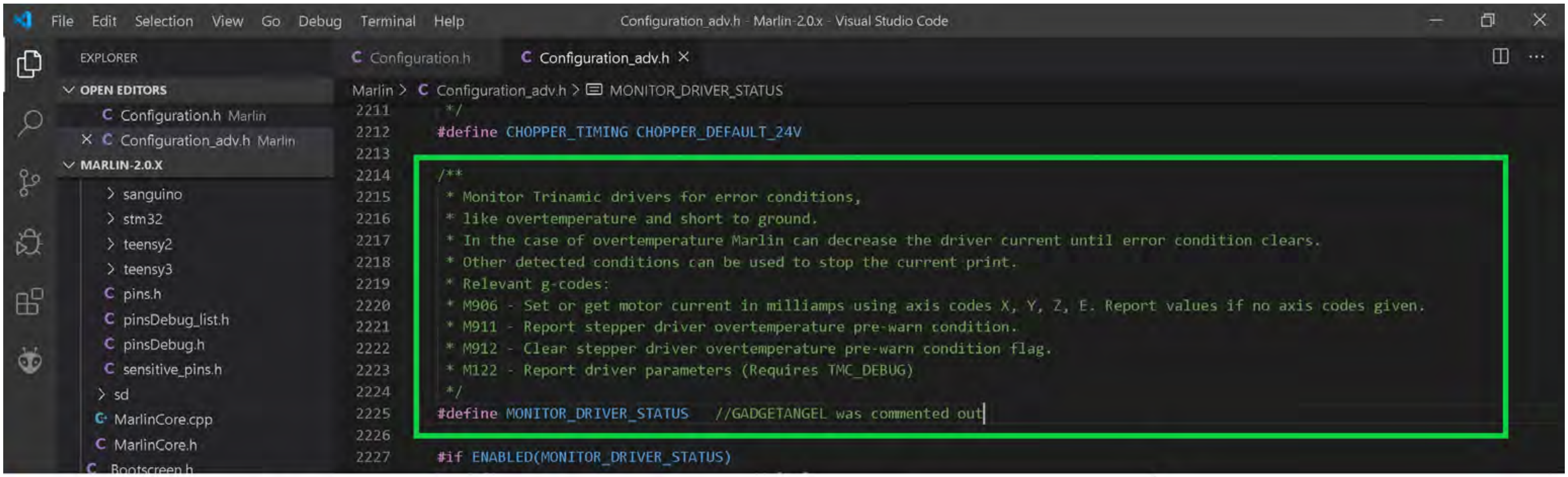
- Now you either enable "HYBRID_THRESHOLD" or disable it. By default, it is disabled. "HYBRID_THERSHOLD" allows the printer to change between stealthChop and spreadCycle dynamically depending on the print speed. I want "HYBRID_THRESHOLD" enabled so I need to remove the two leading "//", which uncomments the line in the Marlin firmware.

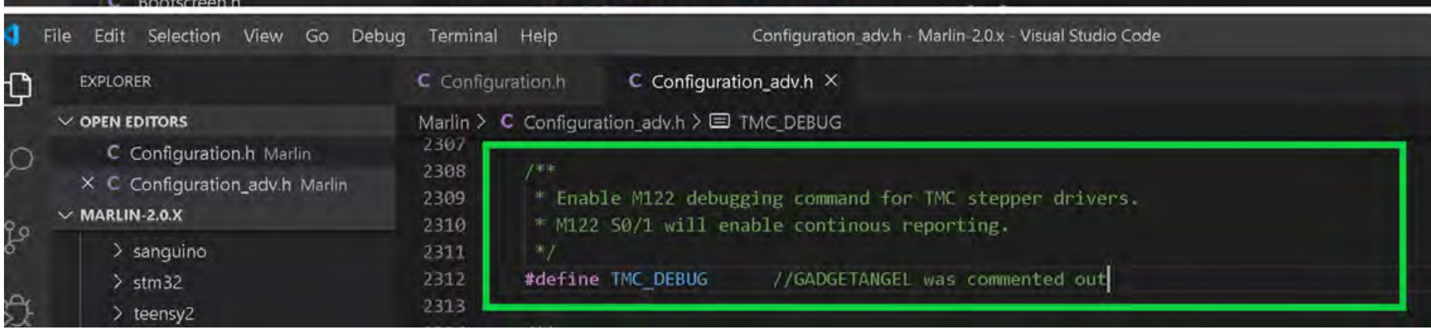


- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2225 V1.0 Drivers in UART Mode

- Now I want to enable some statements that allow me access to debugging the TMC drivers. I will uncomment "MONITOR_DRIVER_STATUS" and "TMC_DEBUG". "MONITOR_DRIVER_STATUS" will enable the following G-codes: M906, M911, and M912, "TMC_DEBUG" will enable the M122 G-code command. You can read about these from the comments in the firmware and in **Marlin's documentation located on-line.**

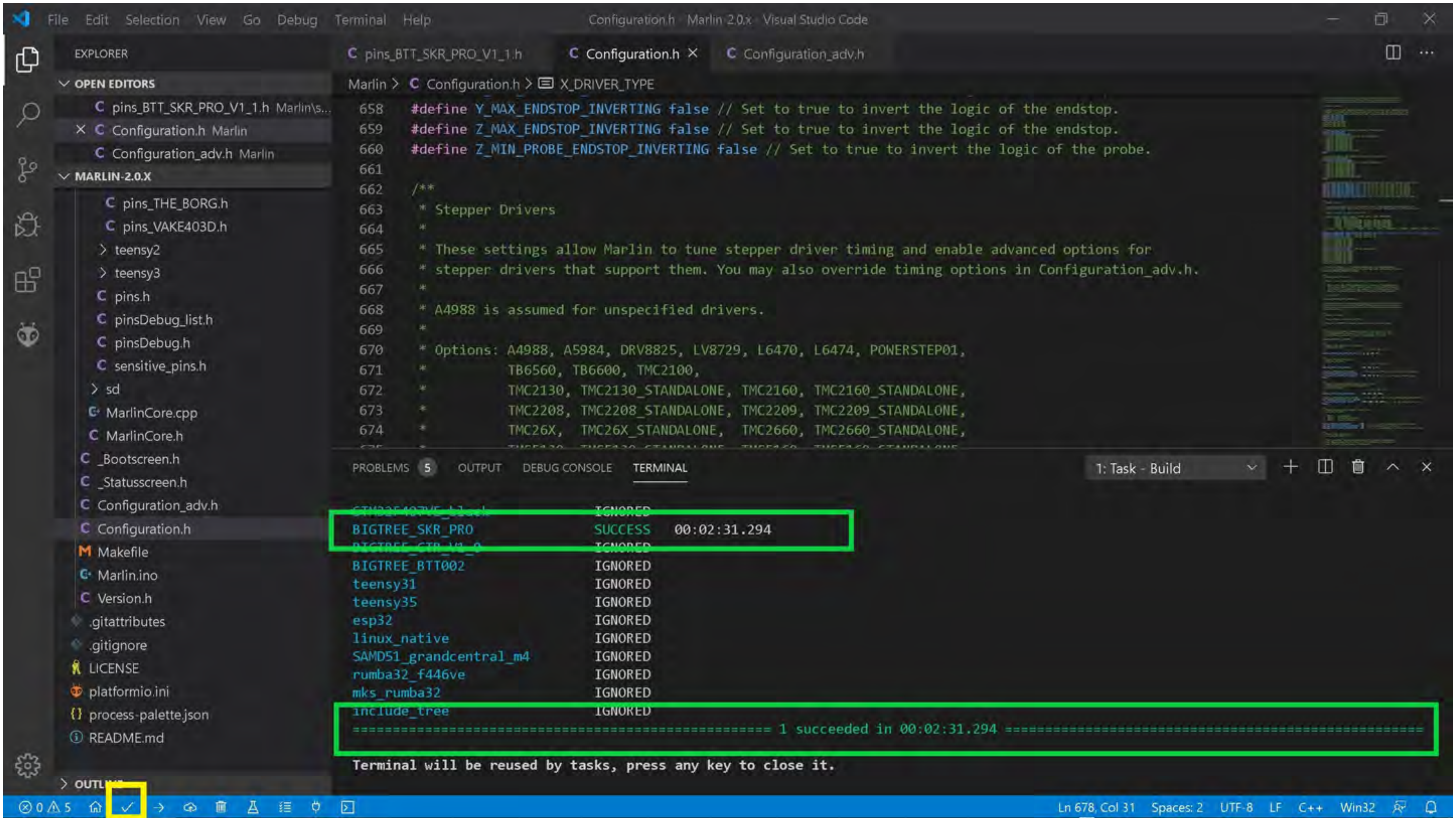




- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2225 V1.0 Drivers in UART Mode

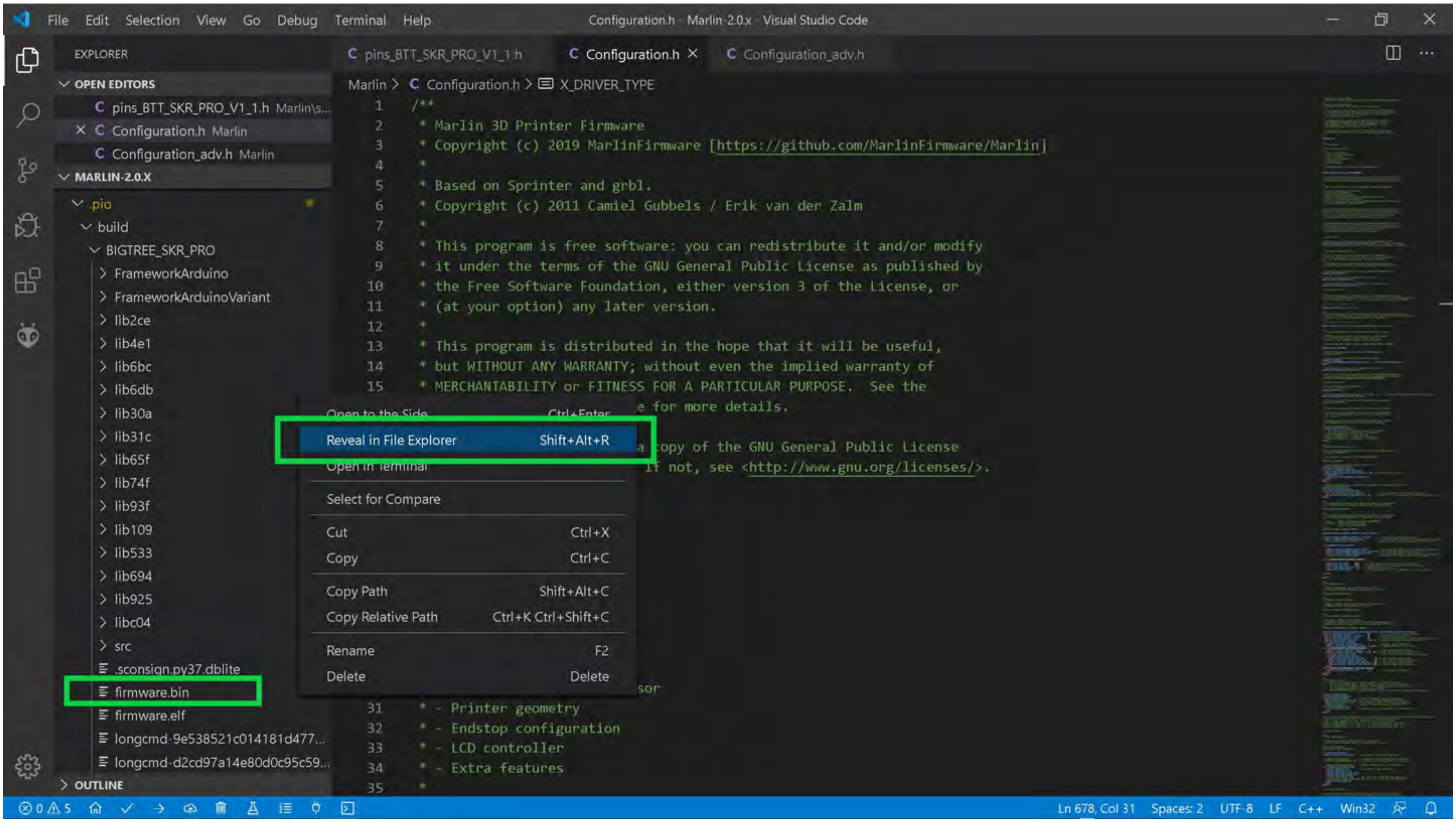
- The end of Marlin setup for BIQU TMC2225 V1.0 drivers in UART mode. Now, compile your code by clicking on the check mark, as seen in the **YELLOW** box below. Once the firmware has compiled successfully you will see the following messages, as shown in the **GREEN** boxes below.



- For the SKR PRO V1.1 board you must save the "firmware.bin" file to the micro SD card then place the micro SD card into the micro SD card reader of the SKR PRO V1.1 board and turn on the power to your printer for the NEW firmware to be automatically applied to the SKR PRO MCU. To locate the "firmware.bin" file, and learn how to transfer it to your micro SD card go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2225 V1.0 Drivers in UART Mode

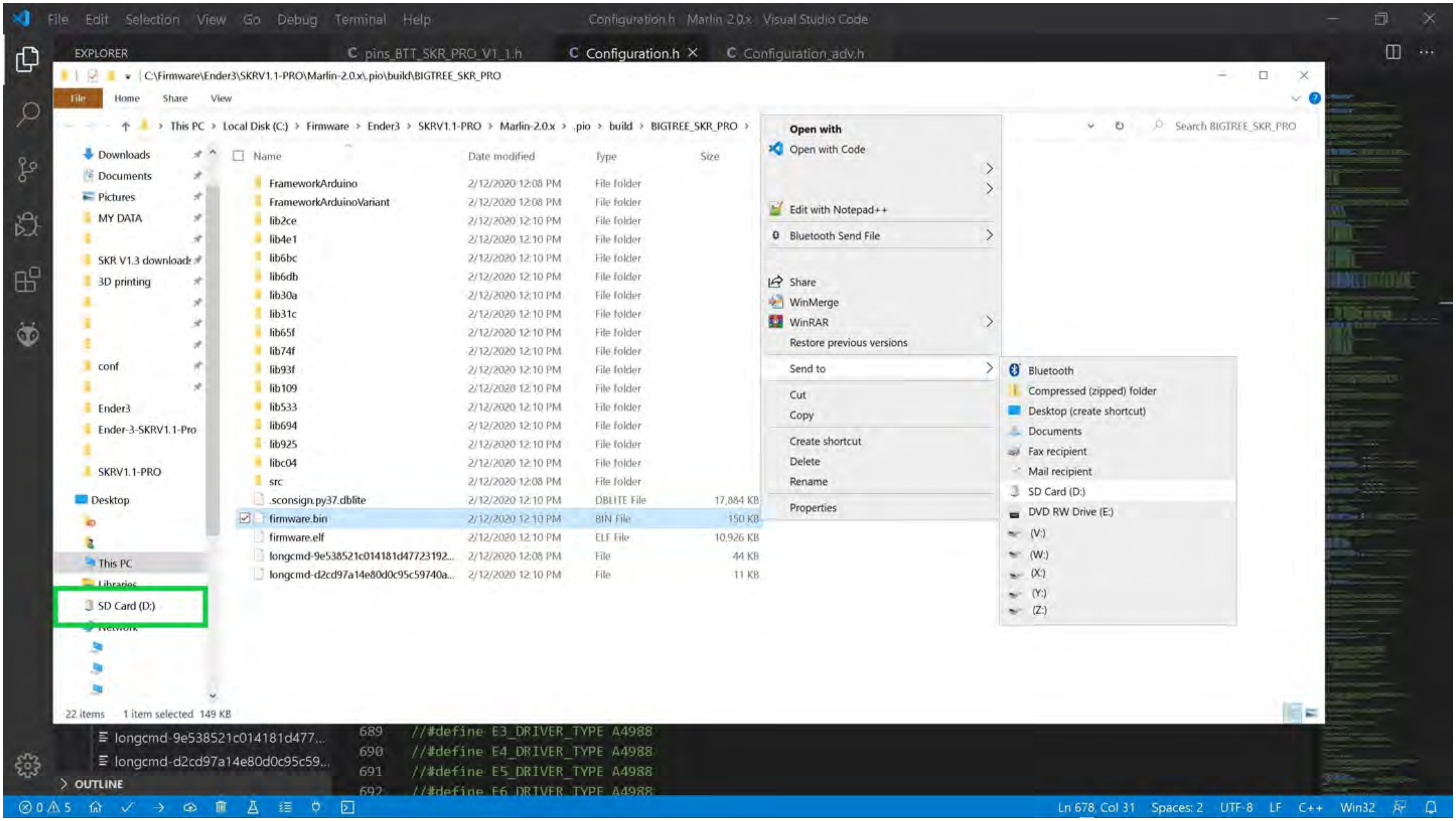
- The "firmware.bin" file can be found by using the left side of the VS code window and going to folder Marlin-2.0.X/.pio/build/BIGTREE_SKR_PRO/ and right clicking on the "firmware.bin" file name to bring up the content menu. Select "Reveal in File Explorer" to have your Window's machine open a file explorer window.



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2225 V1.0 Drivers in UART Mode

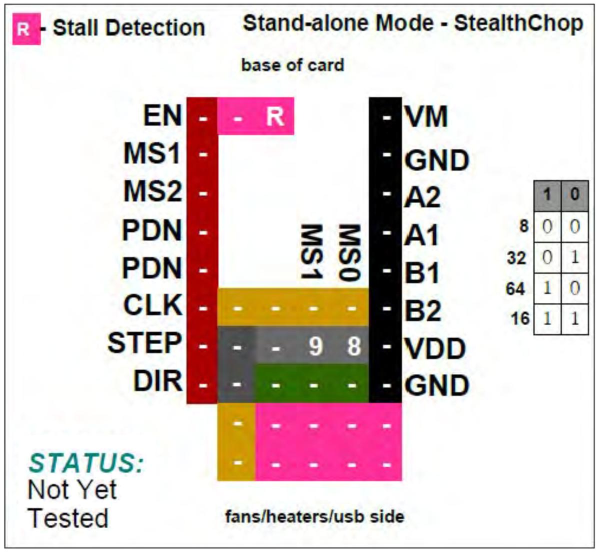
- Ensure your micro SD card, which came with your SKR PRO V1.1 board, with your SKR PRO V1.1 board is in a micro-to-SD-card adapter and plugged into your Window's SD card reader.
    - From the file explorer window right click on the "firmware.bin" file name and select "Send to" from the content menu. For my computer "D:" is my SD card reader. For your computer it may be a different drive letter. The "firmware.bin" file will be written to the SD card and renamed to "FIRMWARE.CUR".



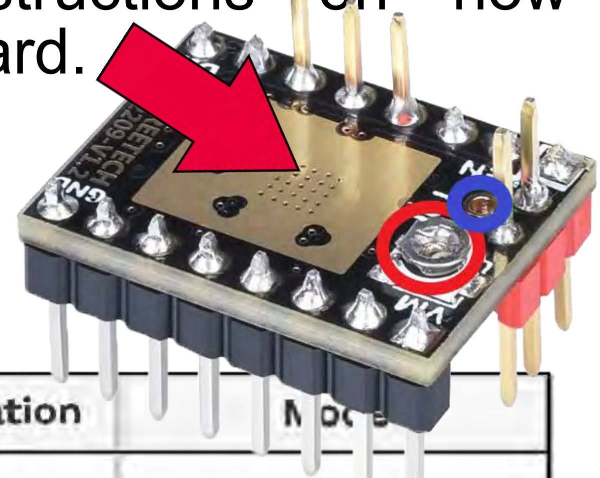
- Right click on the SD card reader, as seen in the **GREEN** box above, and select "Eject" from the context menu. Take the micro SD card out of the adapter and place it into the micro SD card reader of the SKR PRO V1.1 board. Turn on the power to the printer (i.e. the SKR PRO V1.1 board) and the "FIRMWARE.CUR" file will be uploaded to the board. After the upload the firmware file will be renamed to "firmware.bin" and stored on the micro SD card.

BIQU TMC2209 V1.2

Stand-alone Mode for StealthChop



NOTE: Use the potentiometer (POT) on the top of the board, as shown in **RED**; or use the board's "V_{ref} Test point" location, as shown in **BLUE**, to set your V_{ref}. See **Appendix A** for instructions on how to set the V_{ref} on a driver board.



Note: Use 90% of the calculated V_{ref} when tuning the stepper driver board.

Driver Chip	MS1	MS0	Steps	Interpolation	Mode
BIQU® TMC2209 Stand Alone Mode Maximum 64 Subdivision 28V DC 2.8A (peak)	GND	GND	1 / 8	1 / 256	stealthChop
	GND	VIO	1 / 32	1 / 256	stealthChop
	VIO	GND	1 / 64	1 / 256	stealthChop
	VIO	VIO	1 / 16	1 / 256	stealthChop

Driving Current Calculation Formula

R_S(Typical Sense Resistor)= 0.11Ω

$$I_{MAX}=V_{ref}$$

See Appendix B #5. Use 50% to 90% as shown below:

$$I_{MAX}=I_{MAX} * 0.90$$

$$V_{ref}=I_{MAX}$$

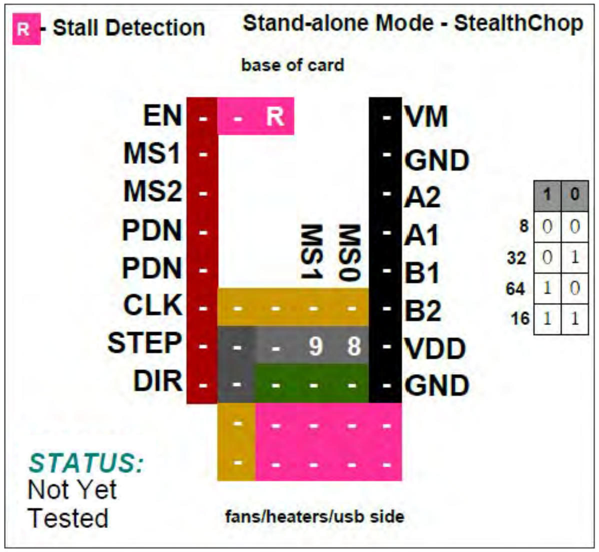
See Appendix B #5. Use 50% to 90% as shown below:

$$V_{ref}=V_{ref} * 0.90$$

NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

BIQU TMC2209 V1.2

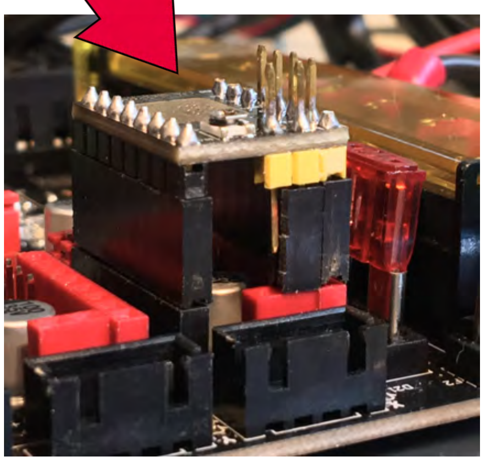
Stand-alone Mode for StealthChop



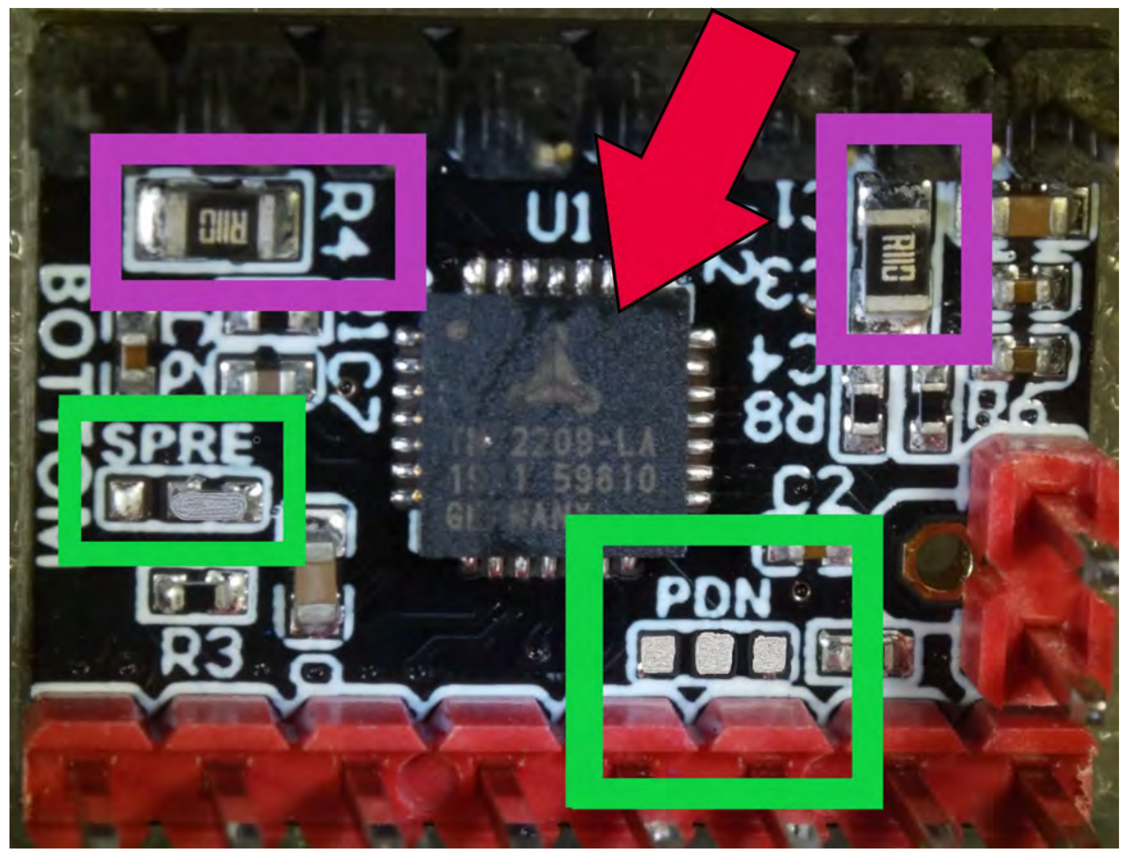
NOTE: The SPRE jumper is located on the bottom of the driver board. In stand-alone mode, the default setting is wired for StealthChop; i.e. the SPRE jumper is set to GND. To switch to Standalone with SpreadCycle, one needs to change the SPRE jumper on the bottom of the driver board. The **PURPLE boxes** below show the location of the current sense resistors (R_s) for the TMC2209.

Note: To switch to stand-alone mode, none of the PDN pads should be bridged. The picture below, as seen in the **GREEN** boxes, shows stand-alone mode with StealthChop.

Note: When the stall-guard function is not used, the stall-guard pin ("R") of the TMC2209 must be removed (desoldered) or use long pin header risers so that the "R" pin does not connect to the SKR PRO V1.1 board



MOST BIQU TMC2209 V1.2 driver boards, when purchased for UART mode, will have the correct PDN pads already soldered together, located on the bottom of the driver board.



BIQU TMC2209 V1.2

Stand-alone Mode for StealthChop

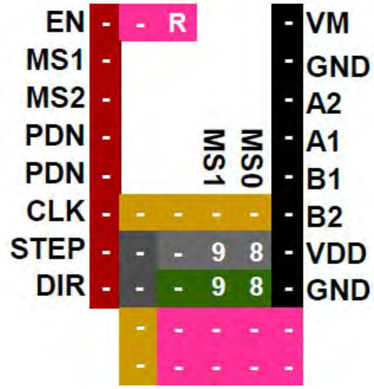
Stand-alone Mode - StealthChop

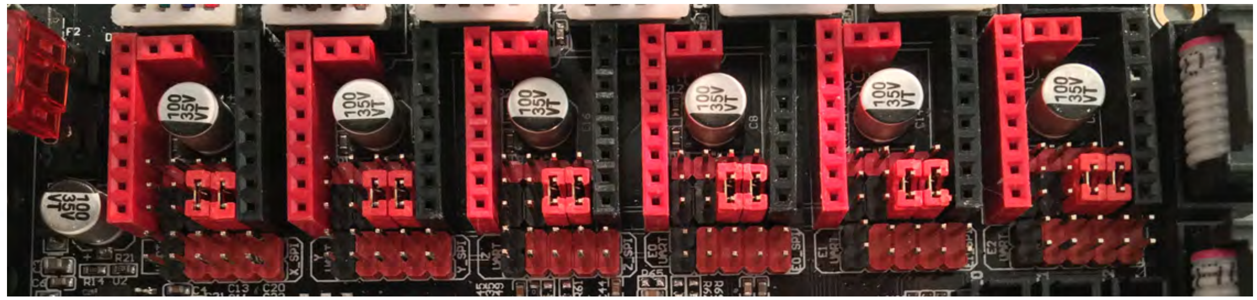
Stand-alone Mode

1 / 8

Interpolation: **1/256**

StealthChop



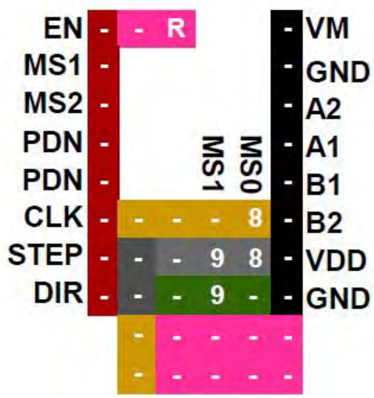


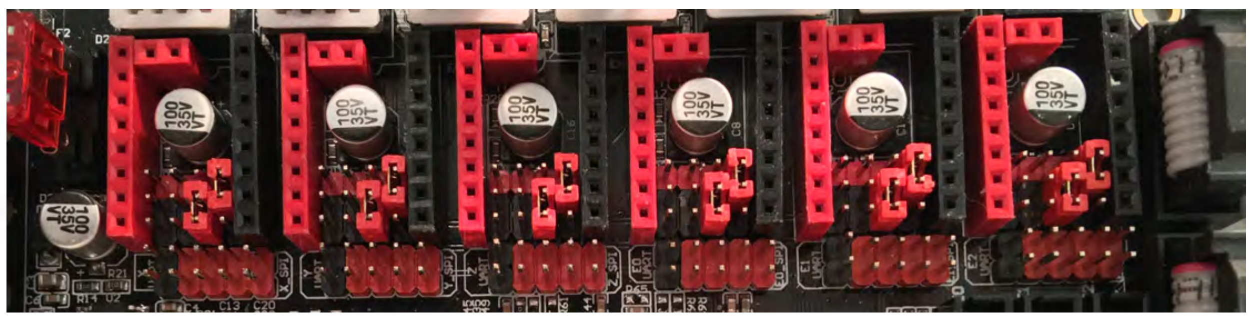
Stand-alone Mode

1 / 32

Interpolation: **1/256**

StealthChop





NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

BIQU TMC2209 V1.2

Stand-alone Mode for StealthChop

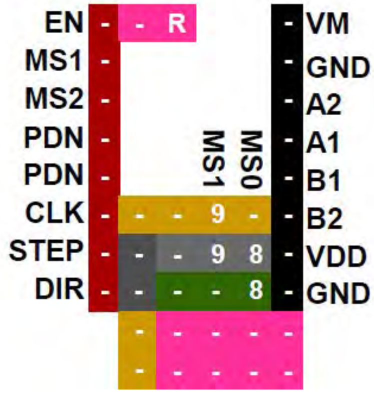
Stand-alone Mode - StealthChop

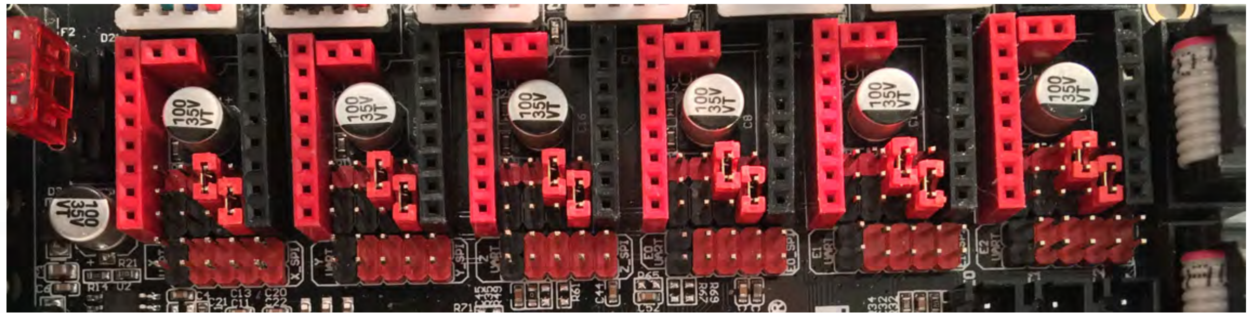
Stand-alone Mode

1 / 64

Interpolation: **1/256**

StealthChop



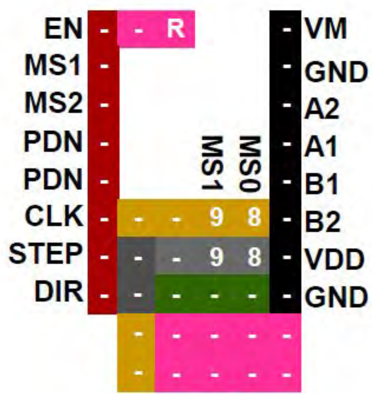


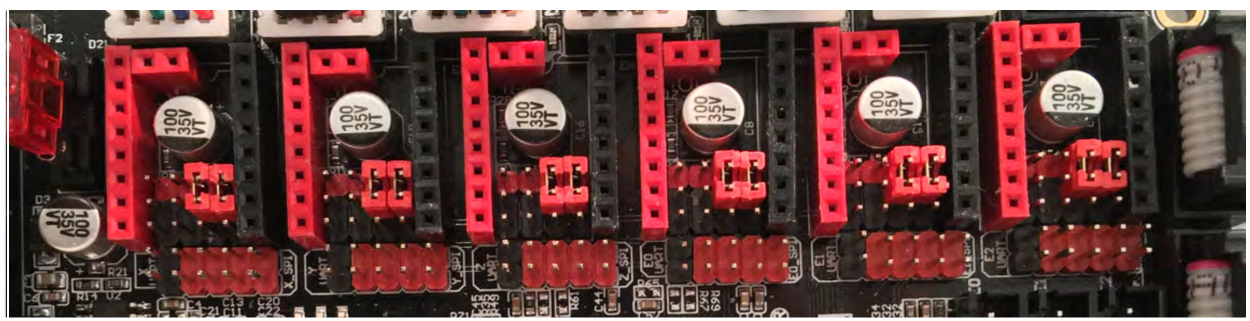
Stand-alone Mode

1 / 16

Interpolation: **1/256**

StealthChop



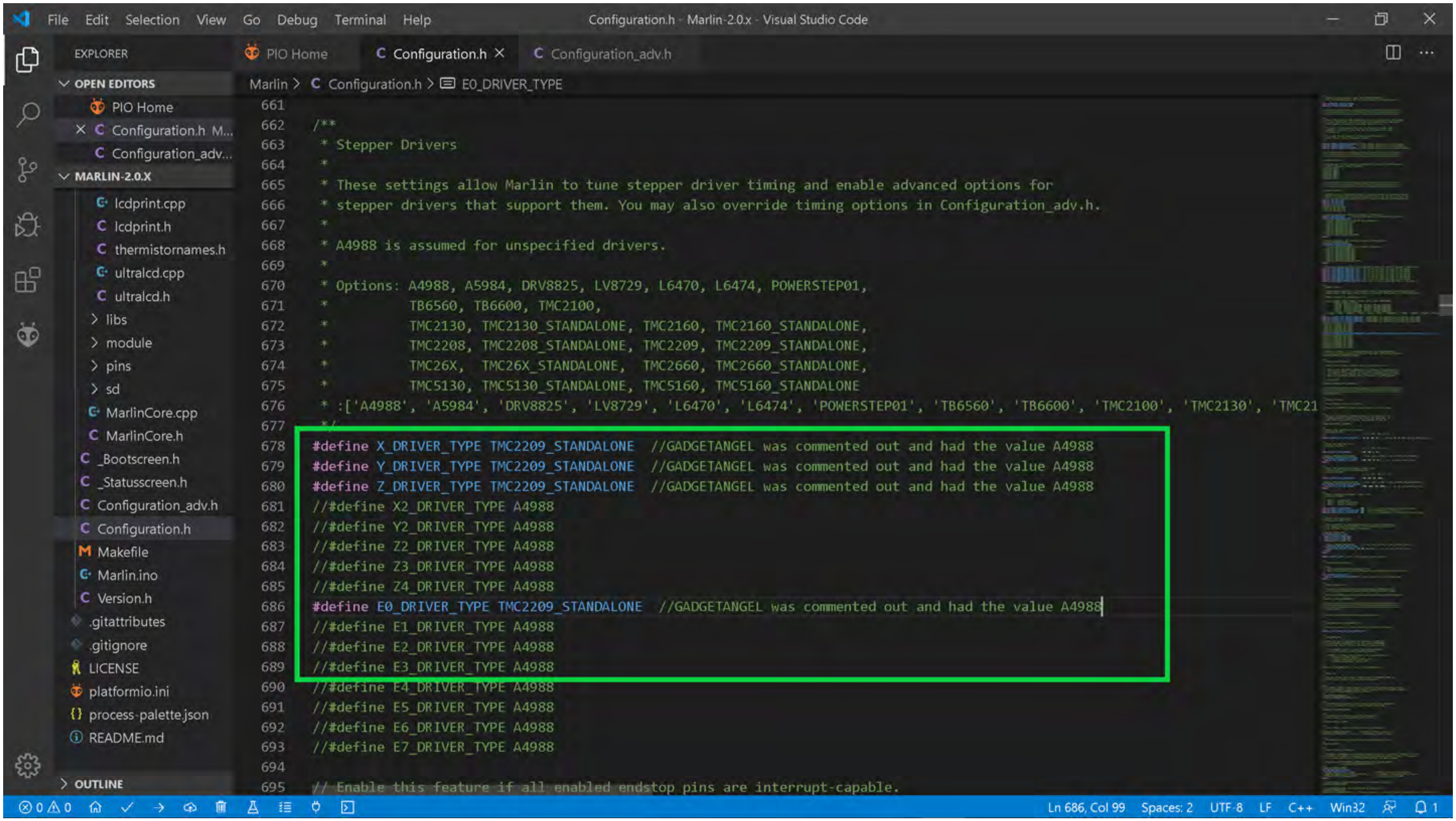


NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

The (latest release of) Marlin Setup for BIQU TMC2209 V1.2 Drivers in Stand-alone Mode for StealthChop

NOTE: **Go to Appendix C**, and then come back here for the changes to Marlin for BIQU TMC2209 V1.2 stepper motor drivers in stand-alone mode for stealthChop.

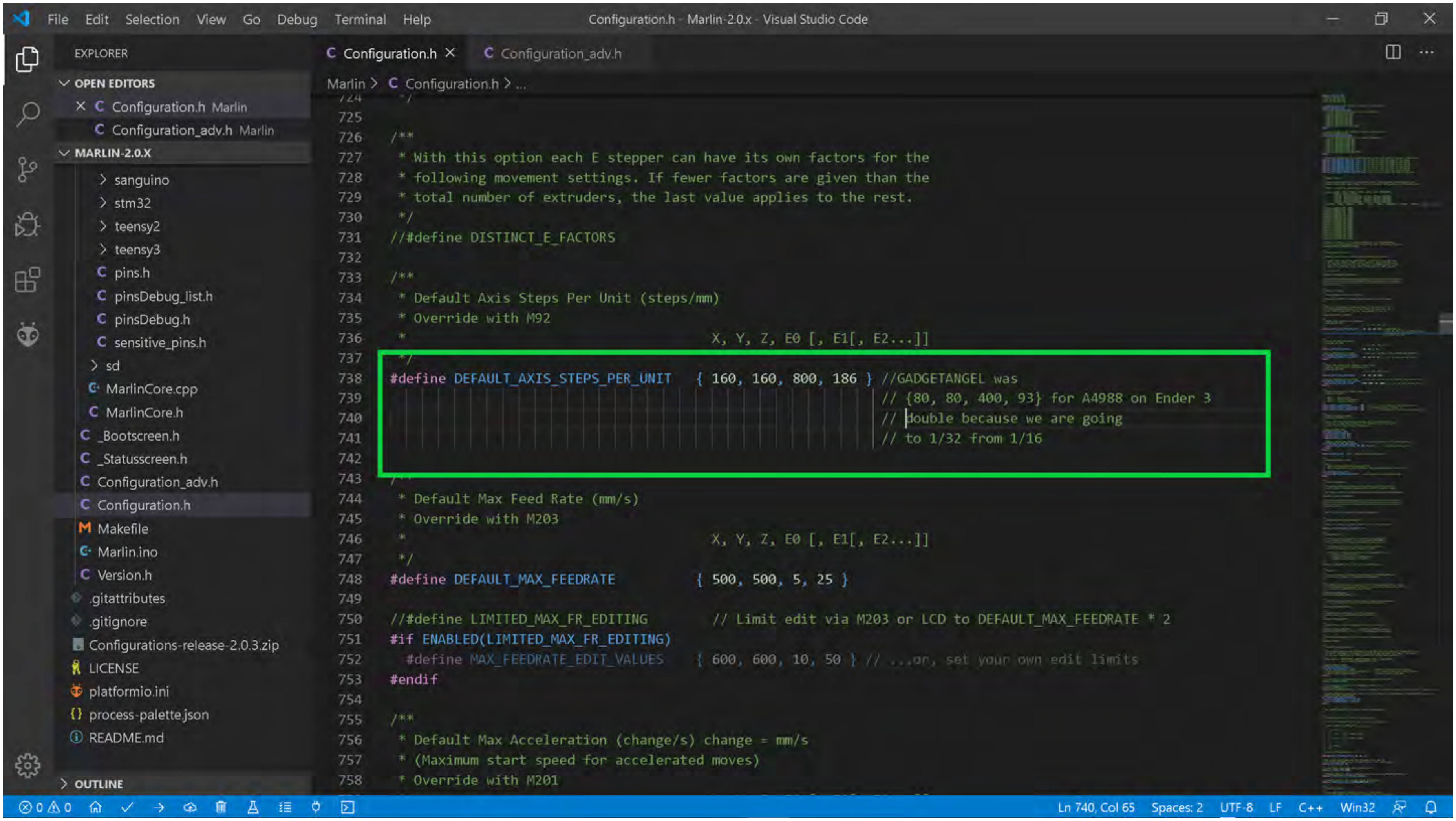
- Change the stepper motor drivers so that Marlin knows you are using TMC2209 drivers in stand-alone mode. Change one line for each axis and one for each extruder you will be using. See the picture below for an example of how to use TMC2209 drivers in stand-alone mode. When two "//" appear at the beginning of a line that means that line is commented out. To un-comment a line just remove the leading two forward slashes ("//").



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2209 V1.2 Drivers in Stand-alone Mode for StealthChop

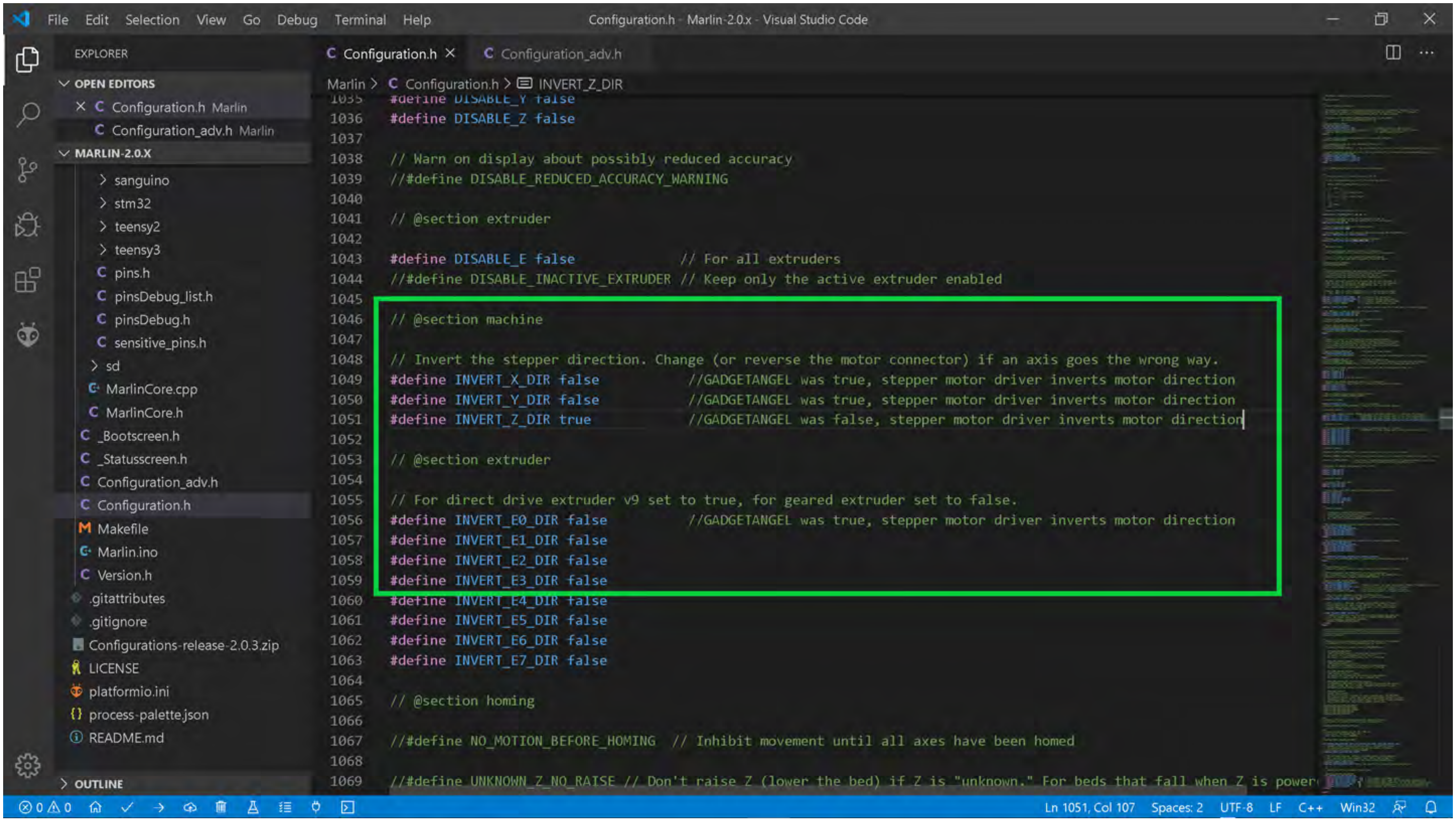
- Since I desire to use 1/32 stepping, and we are changing from A4988 stepper motor drivers on the Ender 3 to TMC2209 stepper motor drivers for each axis and the extruder stepper motor driver, we will be going from 1/16 stepping to 1/32 stepping. So we are doubling our STEPS. Therefore, we must adjust our "DEFAULT_AXIS_STEPS_PER_UNIT" anytime our STEPS are NOT 1/16. So change "DEFAULT_AXIS_STEPS_PER_UNIT" to {160, 160, 800, 186}, as seen in the **GREEN** box below.



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2209 V1.2 Drivers in Stand-alone Mode for StealthChop

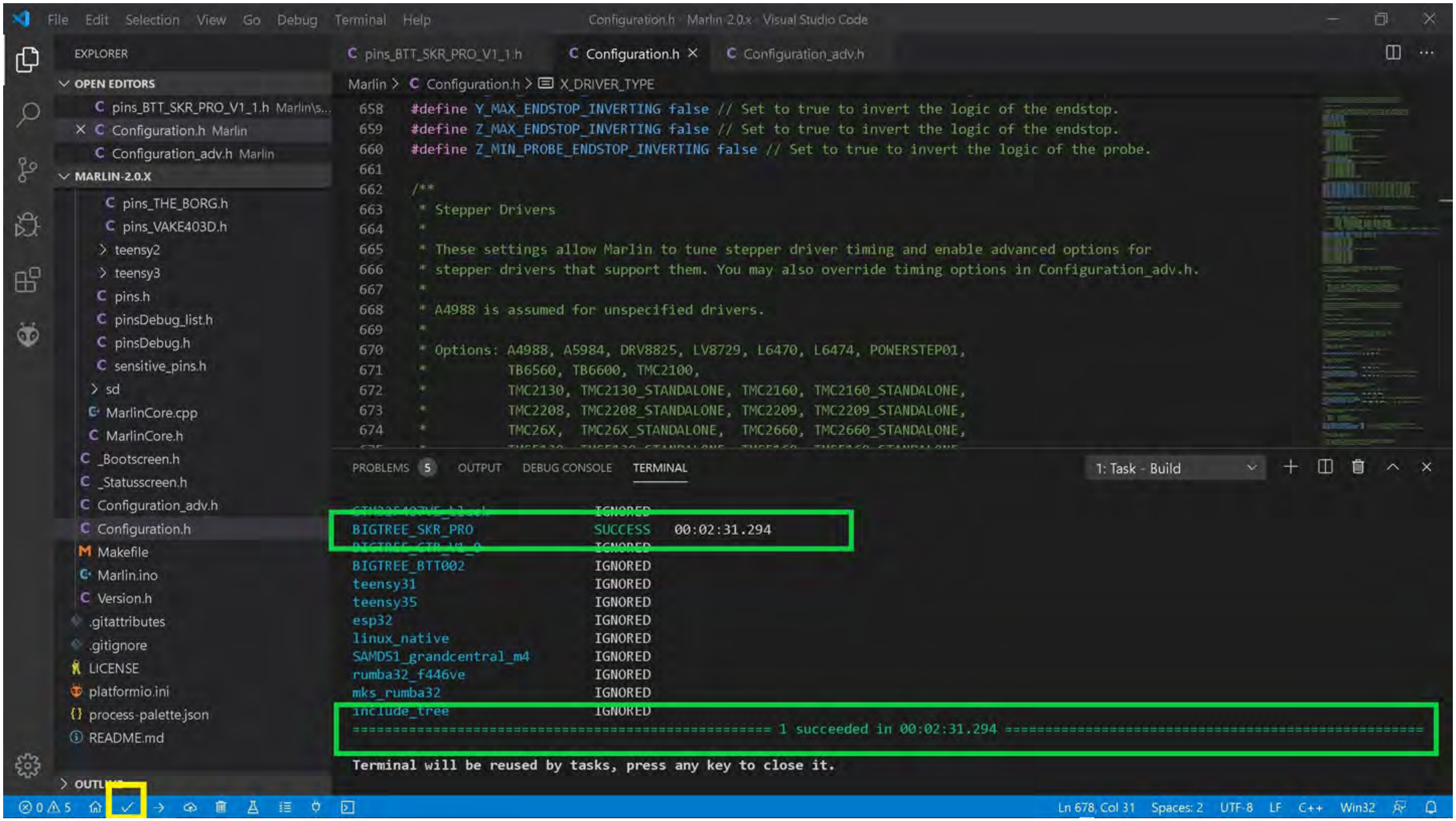
- Since the A4988 driver is what my Ender 3 used, but, now I want to use TMC2209 drivers, I must invert the stepper motor direction because the TMC2209 driver will turn the motors in the opposite direction than the A4988 driver's motor direction. So if the axis' setting you will be using the TMC2209 driver on was "true" change it to "false", as shown in the **GREEN** box below. If the setting was "false", now set it to "true", as shown in the **GREEN** box below



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2209 V1.2 Drivers in Stand-alone Mode for StealthChop

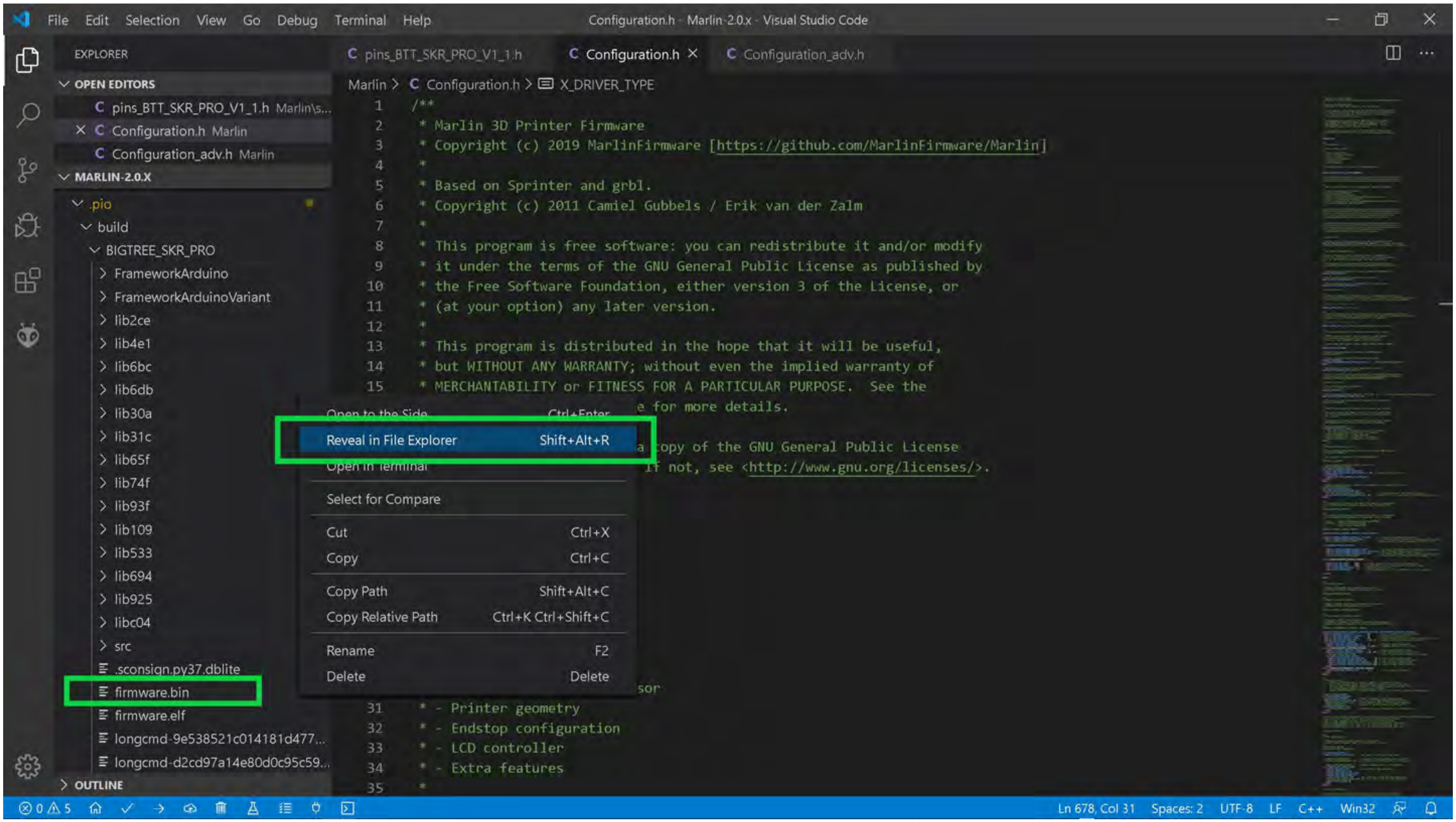
- The end of Marlin setup for BIQU TMC2209 V1.2 drivers in stand-alone mode for stealthChop. Now, compile your code by clicking on the check mark, as seen in the **YELLOW** box below. Once the firmware has compiled successfully you will see the following messages, as shown in the **GREEN** boxes below.



- For the SKR PRO V1.1 board you must save the "firmware.bin" file to the micro SD card then place the micro SD card into the micro SD card reader of the SKR PRO V1.1 board and turn on the power to your printer for the NEW firmware to be automatically applied to the SKR PRO MCU. To locate the "firmware.bin" file, and learn how to transfer it to your micro SD card go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2209 V1.2 Drivers in Stand-alone Mode for StealthChop

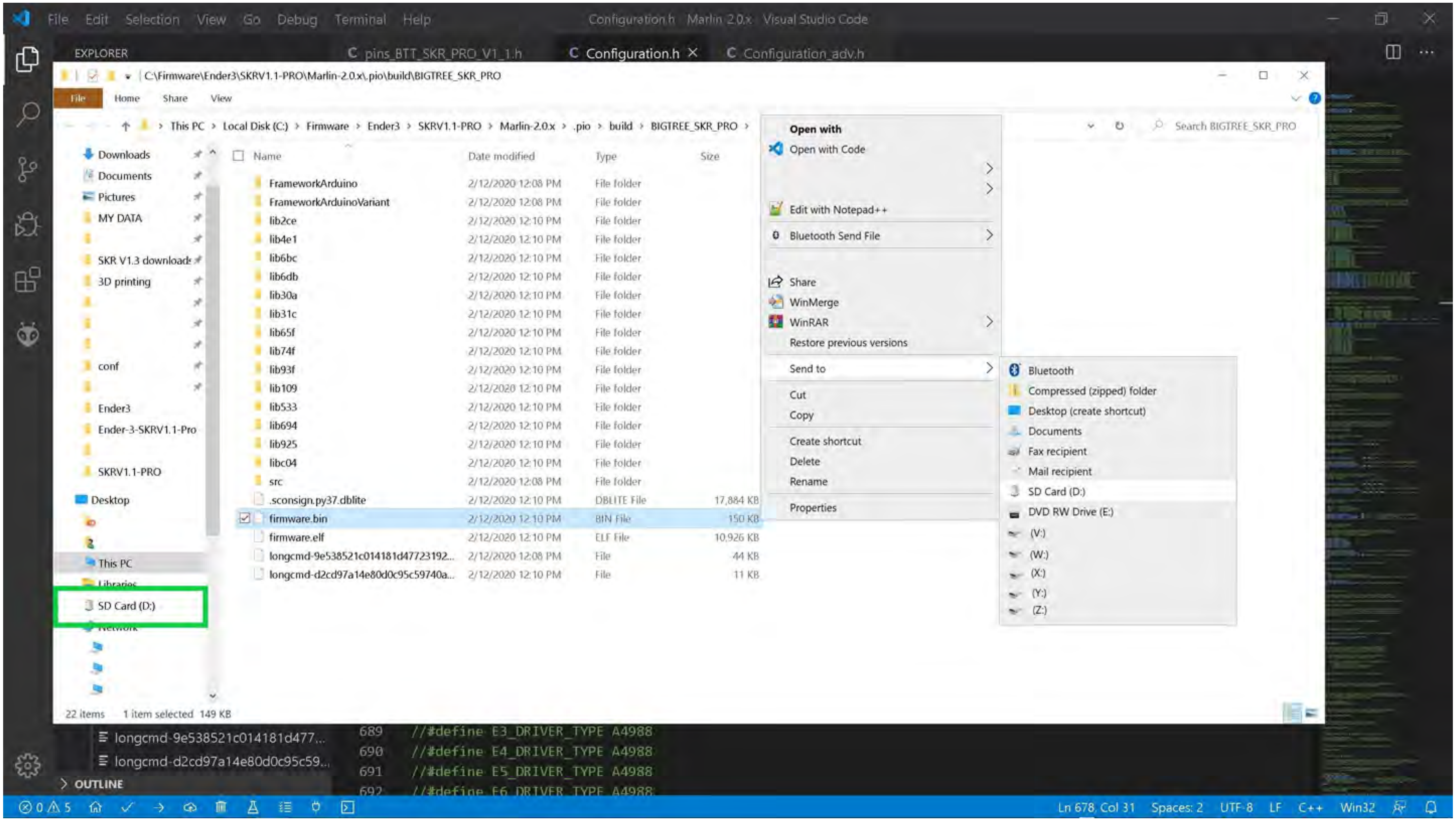
- The "firmware.bin" file can be found by using the left side of the VS code window and going to folder Marlin-2.0.X/.pio/build/ BIGTREE_SKR_PRO/ and right clicking on the "firmware.bin" file name to bring up the content menu. Select "Reveal in File Explorer" to have your Window's machine open a file explorer window.



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2209 V1.2 Drivers in Stand-alone Mode for StealthChop

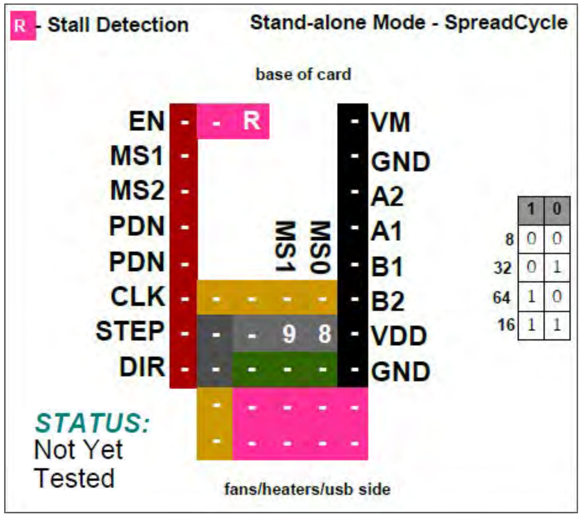
- Ensure your micro SD card, which came with your SKR PRO V1.1 board, with your SKR PRO V1.1 board is in a micro-to-SD-card adapter and plugged into your Window's SD card reader.
    - From the file explorer window right click on the "firmware.bin" file name and select "Send to" from the content menu. For my computer "D:" is my SD card reader. For your computer it may be a different drive letter. The "firmware.bin" file will be written to the SD card and renamed to "FIRMWARE.CUR".



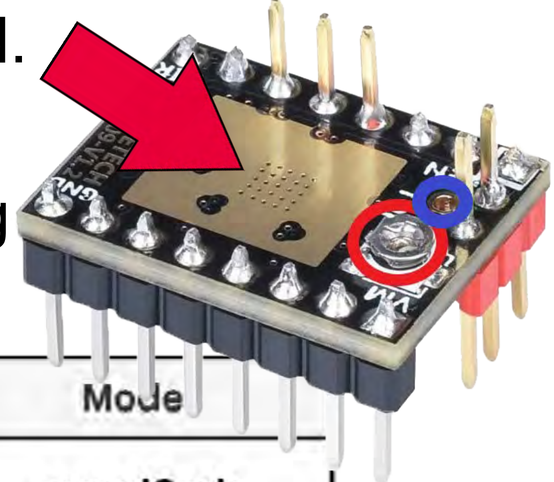
- Right click on the SD card reader, as seen in the **GREEN** box above, and select "Eject" from the context menu. Take the micro SD card out of the adapter and place it into the micro SD card reader of the SKR PRO V1.1 board. Turn on the power to the printer (i.e. the SKR PRO V1.1 board) and the "FIRMWARE.CUR" file will be uploaded to the board. After the upload the firmware file will be renamed to "firmware.bin" and stored on the micro SD card.

BIQU TMC2209 V1.2

Stand-alone Mode for SpreadCycle



NOTE: Use the potentiometer (POT) on the top of the board, as shown in **RED**; or use the board's "V_{ref} Test point" location, as shown in **BLUE**, to set your V_{ref}. See **Appendix A** for instructions on how to set the V_{ref} on a driver board.



Note: Use 90% of the calculated V_{ref} when tuning the stepper driver board.

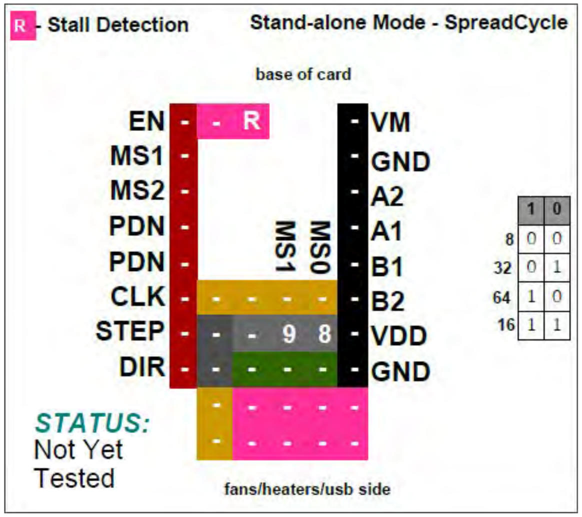
Driver Chip	MS1	MS0	Steps	Interpolation	Mode
BIQU® TMC2209 Stand Alone Mode Maximum 64 Subdivision 28V DC 2.8A (peak)	GND	GND	1 / 8	1 / 256	spreadCycle
	GND	VIO	1 / 32	1 / 256	spreadCycle
	VIO	GND	1 / 64	1 / 256	spreadCycle
	VIO	VIO	1 / 16	1 / 256	spreadCycle

Driving Current Calculation Formula R_S(Typical Sense Resistor)= 0.11Ω	$I_{MAX}=V_{ref}$ See Appendix B #5. Use 50% to 90% as shown below: $I_{MAX}=I_{MAX} * 0.90$	$V_{ref}=I_{MAX}$ See Appendix B #5. Use 50% to 90% as shown below: $V_{ref}=V_{ref} * 0.90$

NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

BIQU TMC2209 V1.2

Stand-alone Mode for SpreadCycle

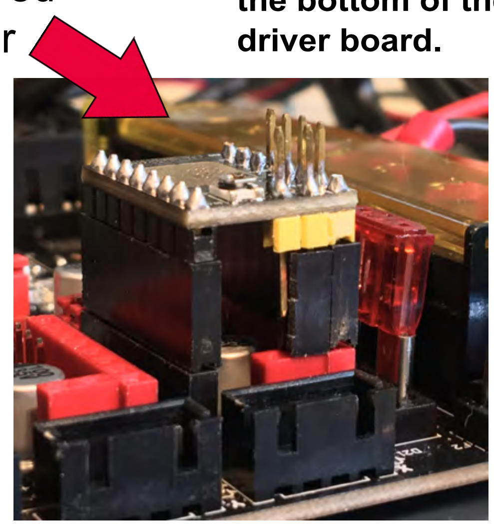


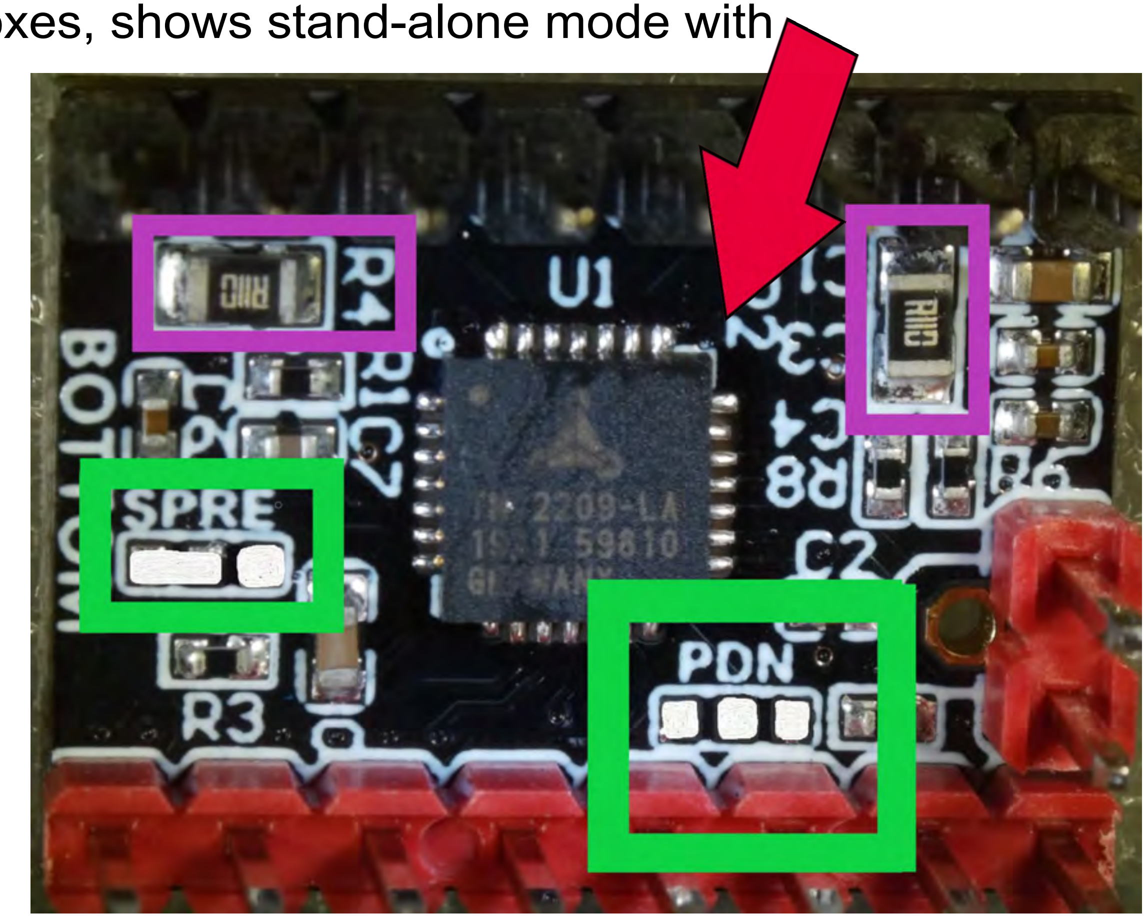
NOTE: The SPRE jumper is located on the bottom of the driver board. In Standalone Mode, the default setting is wired for StealthChop; i.e. the SPRE jumper is set to GND. To switch to Standalone with SpreadCycle, one needs to change the SPRE jumper on the bottom of the driver board. The **PURPLE boxes** below show the location of the current sense resistors (R_S) for the TMC2209.

Note: To switch to stand-alone mode, none of the PDN pads should be bridged. The picture below, as seen in the **GREEN** boxes, shows stand-alone mode with SpreadCycle.

Note: When the stall-guard function is not used, the stall-guard pin ("R") of the TMC2209 must be removed (desoldered) or use long pin header risers so that the "R" pin does not connect to the SKR PRO V1.1 board

MOST BIQU TMC2209 V1.2 driver boards, when purchased for UART mode, will have the correct PDN pads already soldered together, located on the bottom of the driver board.





NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

BIQU TMC2209 V1.2

Stand-alone Mode for SpreadCycle

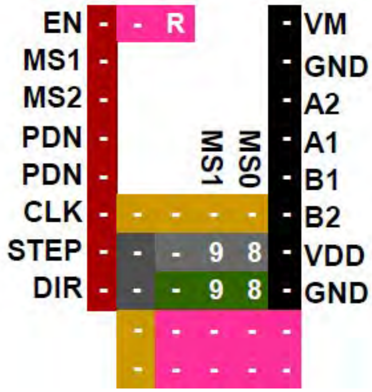
Stand-alone Mode - SpreadCycle

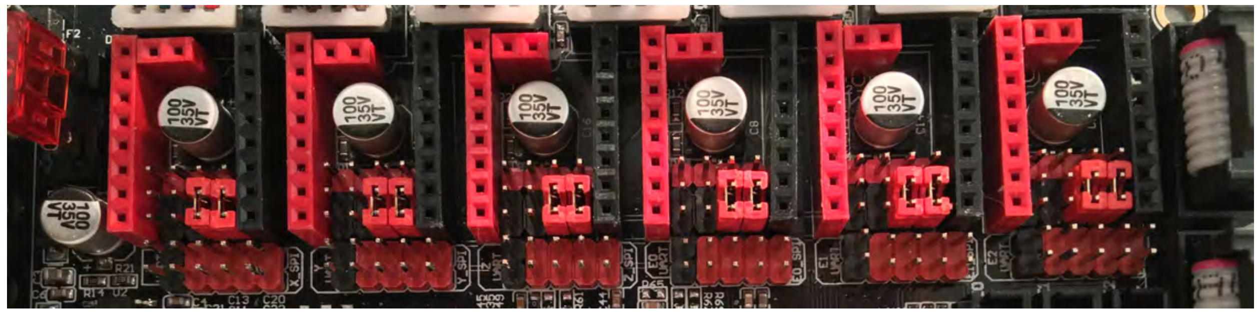
Stand-alone Mode

1 / 8

Interpolation: 1/256

SpreadCycle



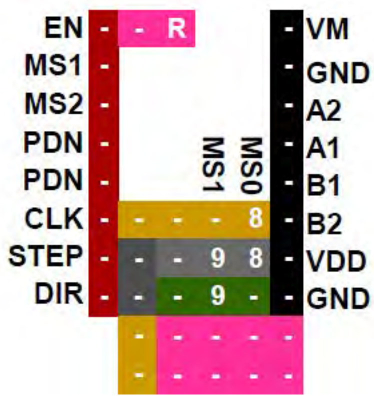


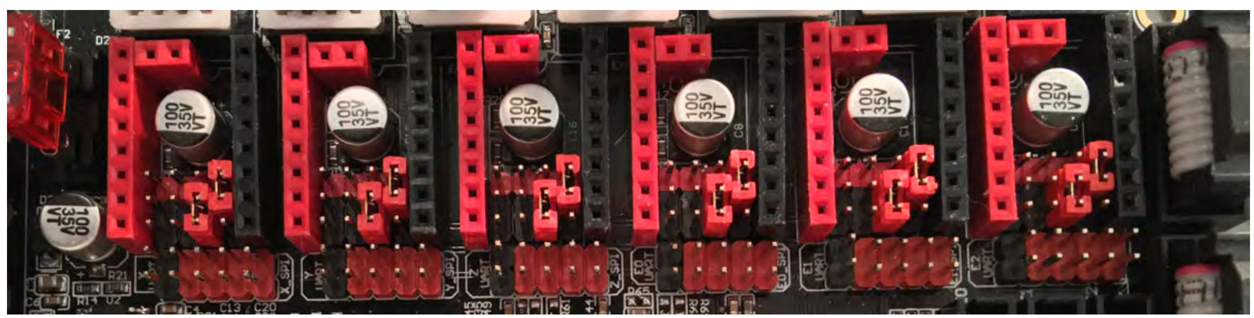
Stand-alone Mode

1 / 32

Interpolation: 1/256

SpreadCycle





NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

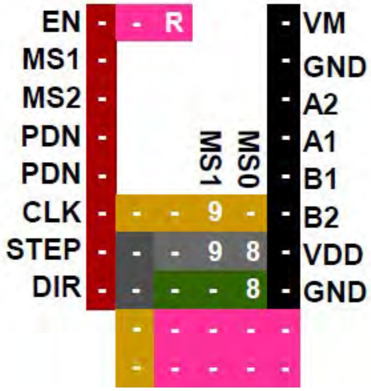
Stand-alone Mode - SpreadCycle

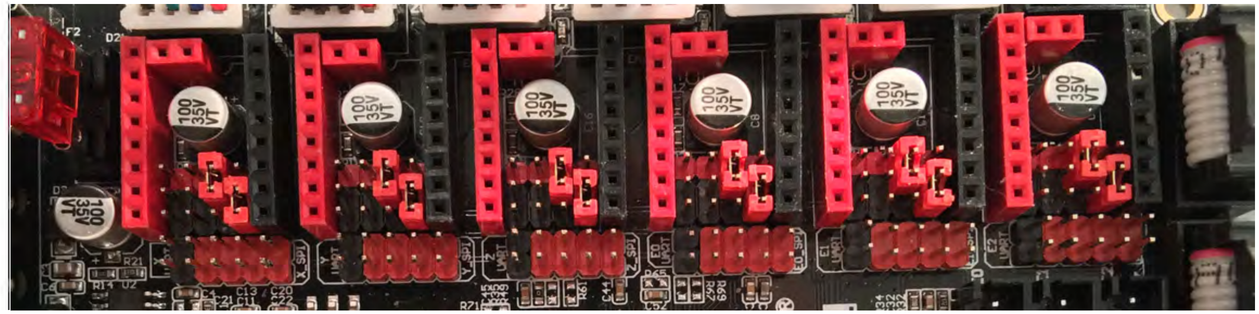
Stand-alone Mode

1 / 64

Interpolation: 1/256

SpreadCycle



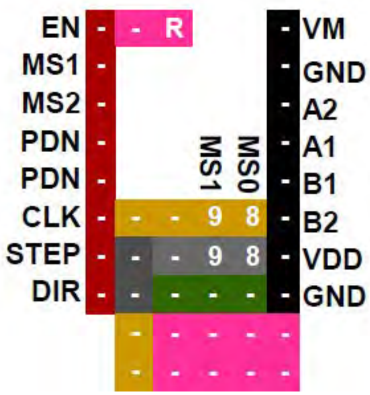


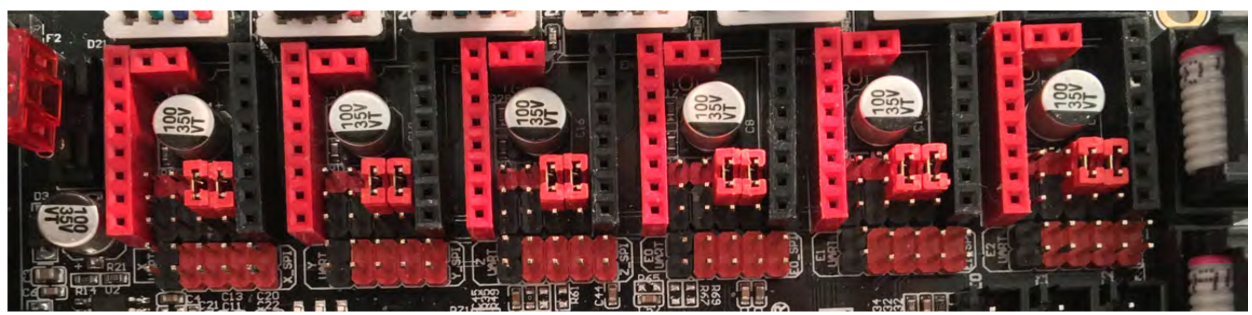
Stand-alone Mode

1 / 16

Interpolation: 1/256

SpreadCycle



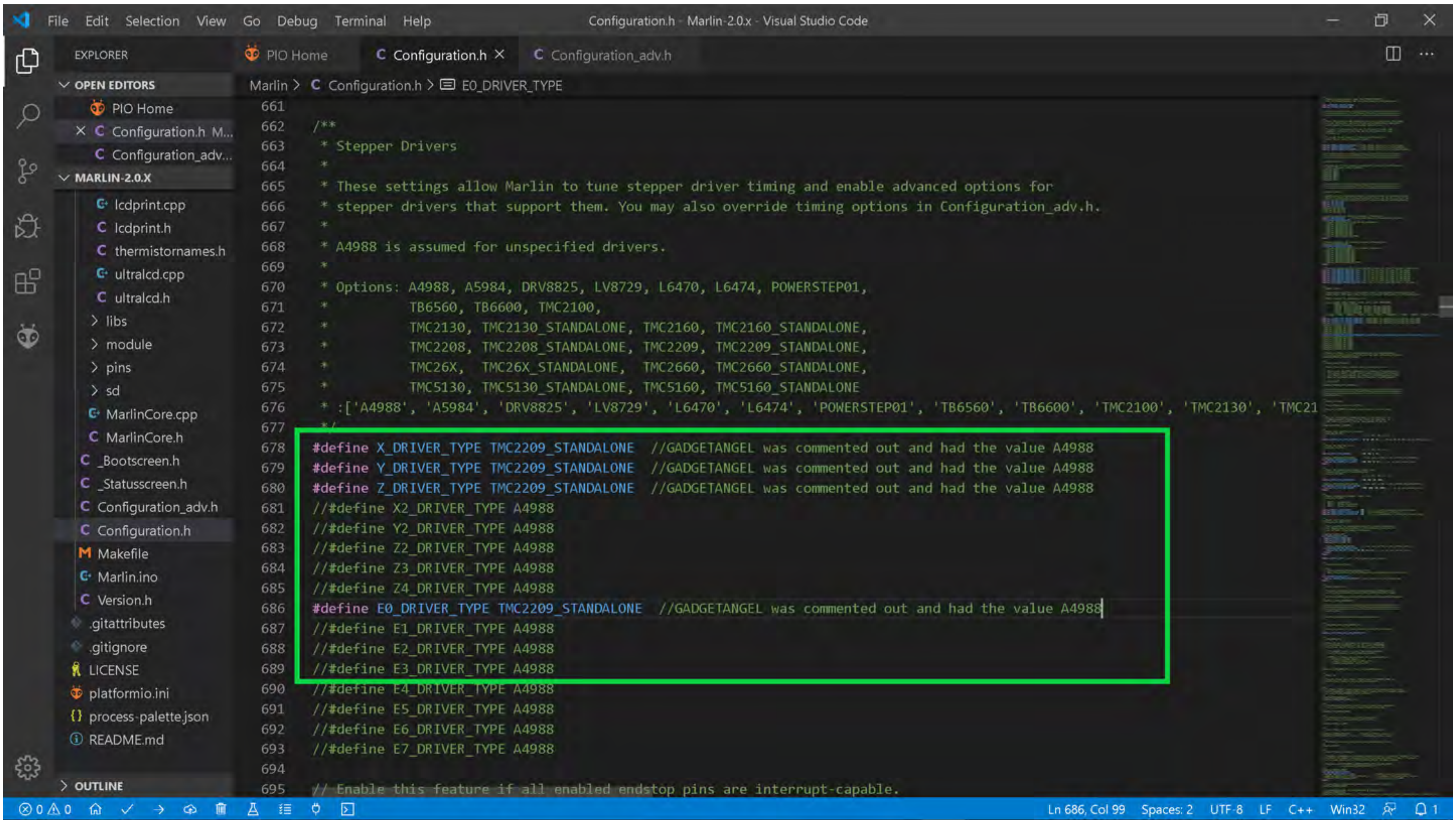


NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

The (latest release of) Marlin Setup for BIQU TMC2209 V1.2 Drivers in Stand-alone Mode for SpreadCycle

NOTE: **Go to Appendix C**, and then come back here for the changes to Marlin for BIQU TMC2209 V1.2 stepper motor drivers in stand-alone mode for spreadCycle.

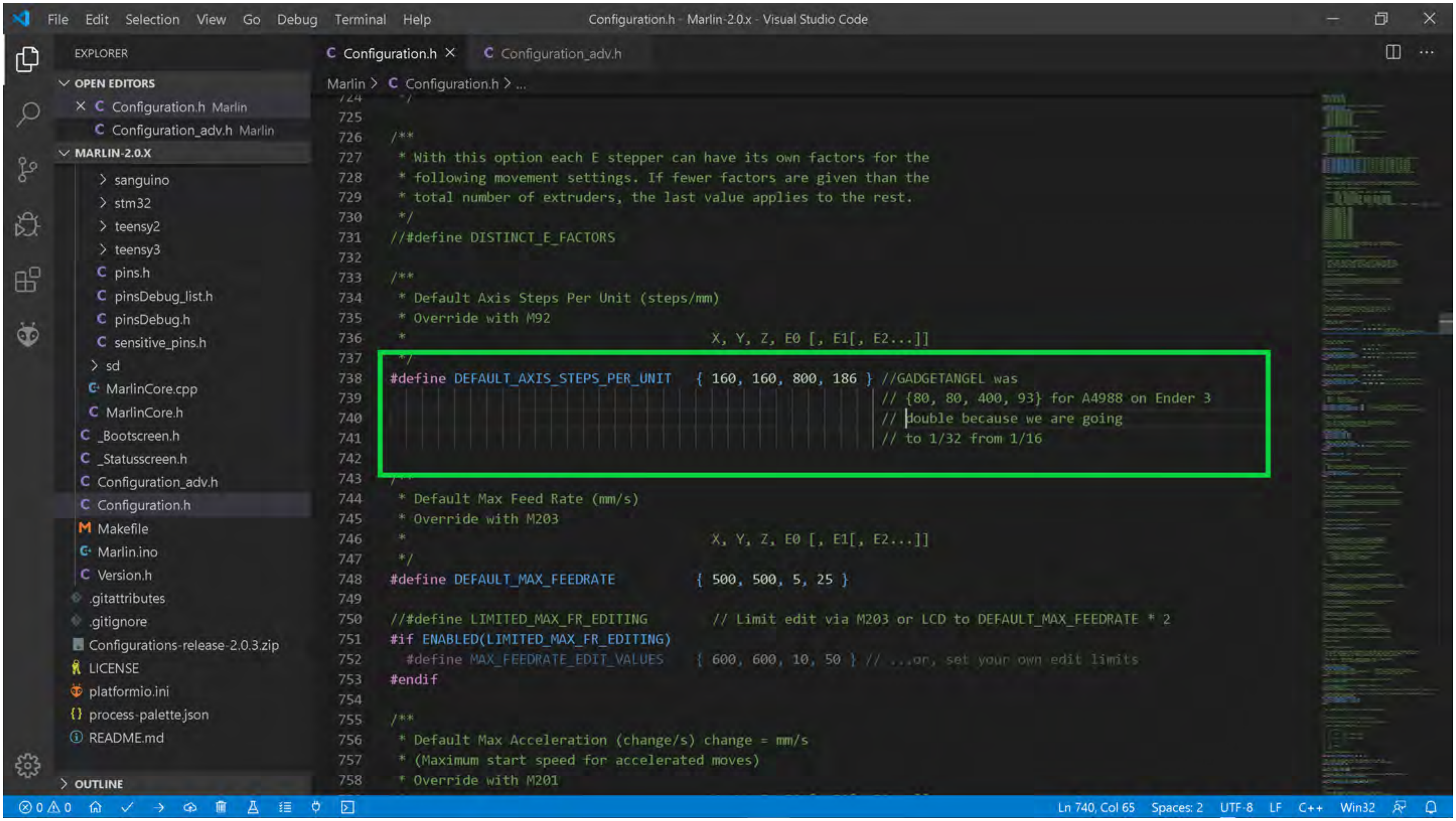
- Change the stepper motor drivers so that Marlin knows you are using TMC2209 drivers in stand-alone mode. Change one line for each axis and one for each extruder you will be using. See the picture below for an example of how to use TMC2209 drivers in stand-alone mode. When two "//" appear at the beginning of a line that means that line is commented out. To un-comment a line just remove the leading two forward slashes ("//").



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2209 V1.2 Drivers in Stand-alone Mode for SpreadCycle

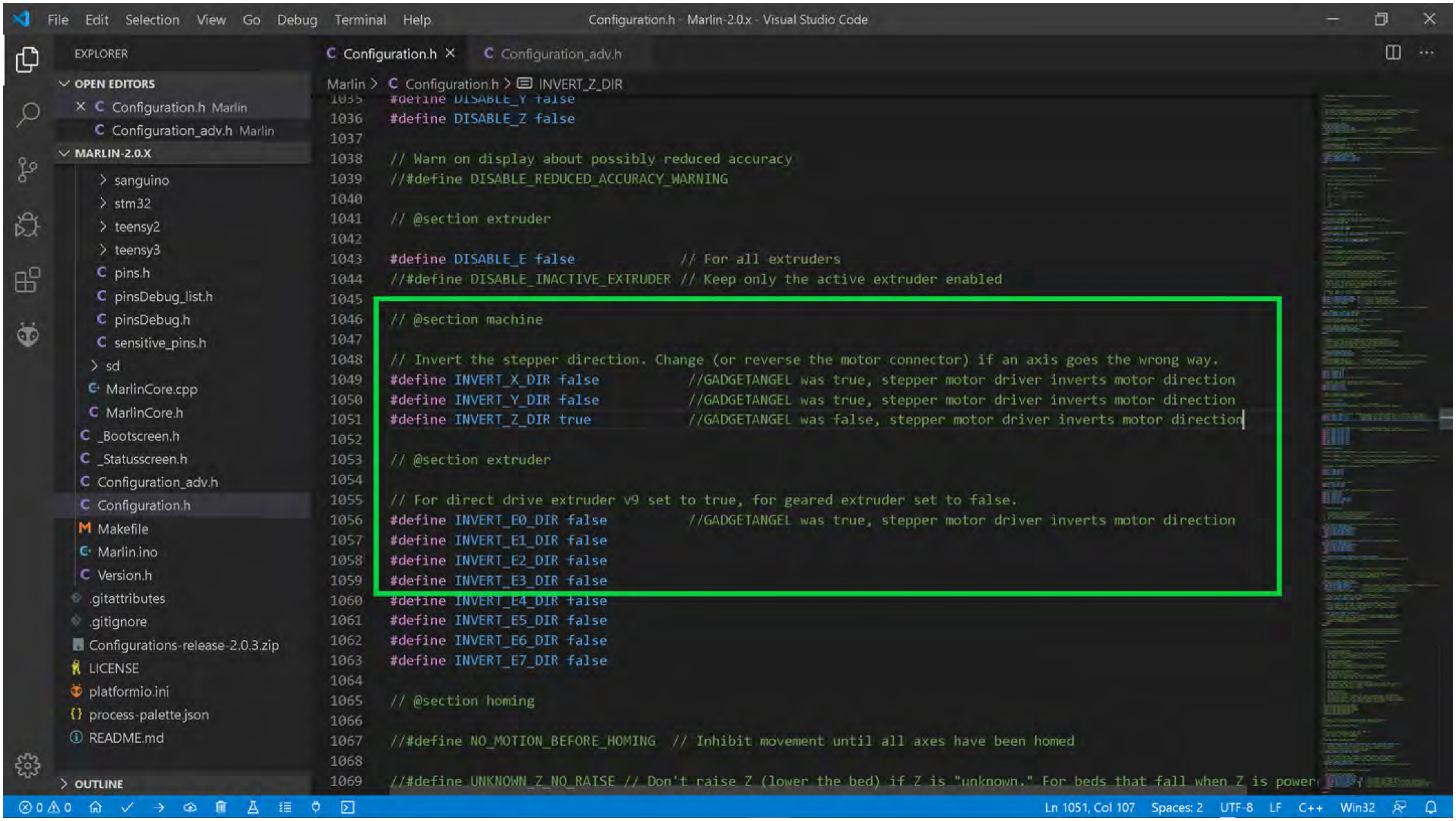
- Since I desire to use 1/32 stepping, and we are changing from A4988 stepper motor drivers on the Ender 3 to TMC2209 stepper motor drivers for each axis and the extruder stepper motor driver, we will be going from 1/16 stepping to 1/32 stepping. So we are doubling our STEPS. Therefore, we must adjust our "DEFAULT_AXIS_STEPS_PER_UNIT" anytime our STEPS are NOT 1/16. So change "DEFAULT_AXIS_STEPS_PER_UNIT" to {160, 160, 800, 186}, as seen in the **GREEN** box below.



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2209 V1.2 Drivers in Stand-alone Mode for SpreadCycle

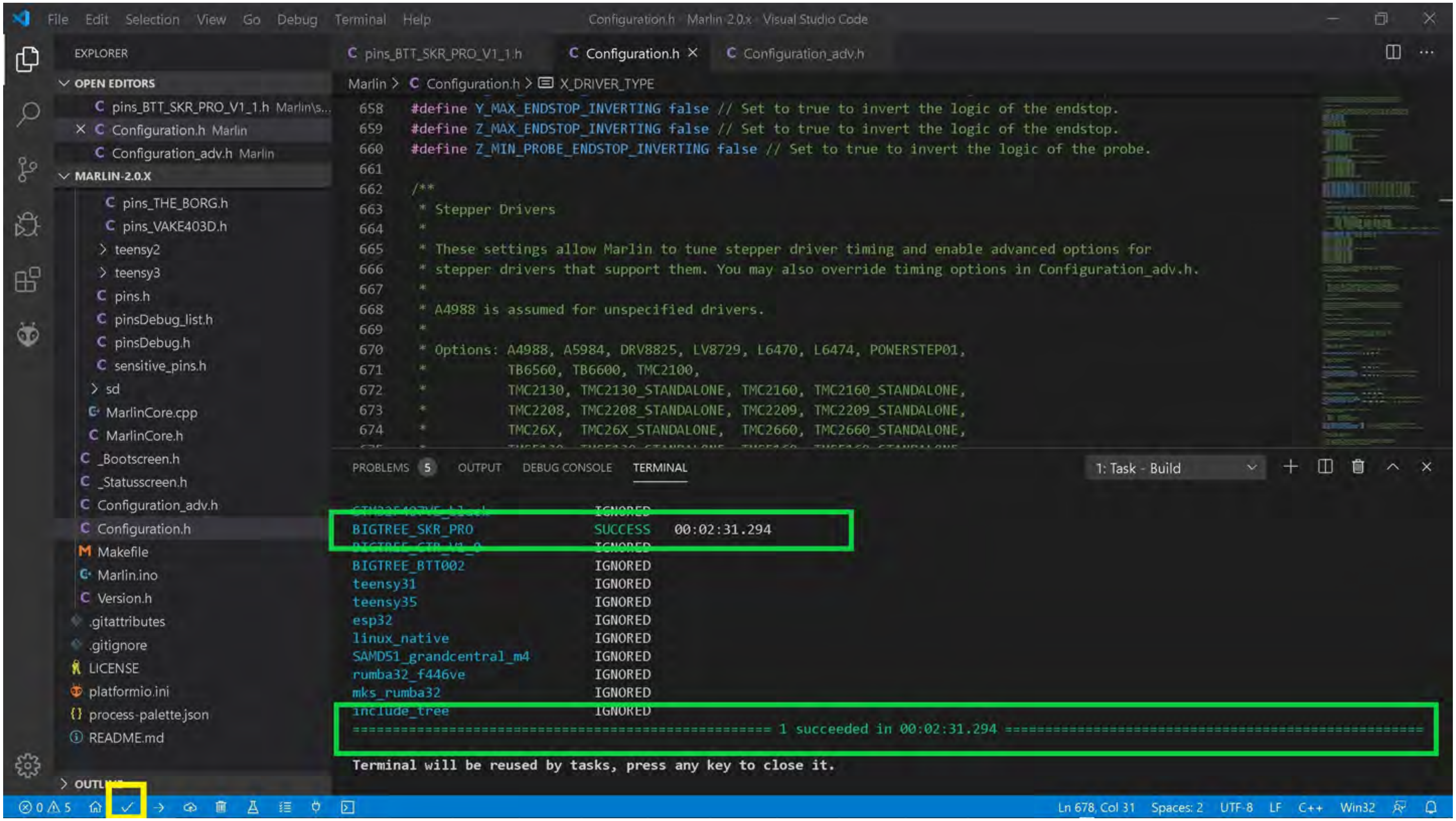
- Since the A4988 driver is what my Ender 3 used, but, now I want to use TMC2209 drivers, I must invert the stepper motor direction because the TMC2209 driver will turn the motors in the opposite direction than the A4988 driver's motor direction. So if the axis' setting you will be using the TMC2209 driver on was "true" change it to "false", as shown in the **GREEN** box below. If the setting was "false", now set it to "true", as shown in the **GREEN** box below



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2209 V1.2 Drivers in Stand-alone Mode for SpreadCycle

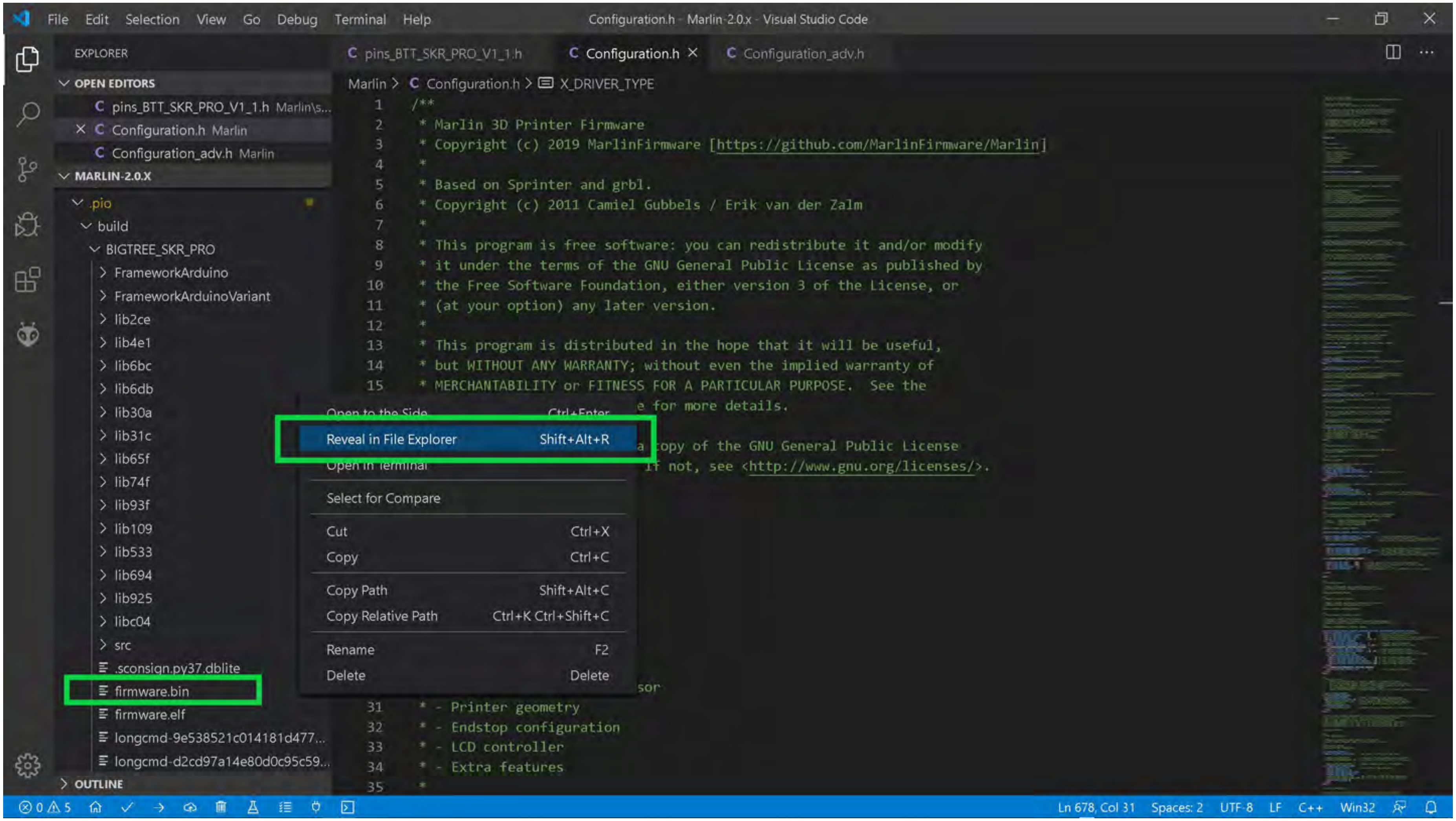
- The end of Marlin setup for BIQU TMC2209 V1.2 drivers in stand-alone mode for spreadCycle. Now, compile your code by clicking on the check mark, as seen in the **YELLOW** box below. Once the firmware has compiled successfully you will see the following messages, as shown in the **GREEN** boxes below.



- For the SKR PRO V1.1 board you must save the "firmware.bin" file to the micro SD card then place the micro SD card into the micro SD card reader of the SKR PRO V1.1 board and turn on the power to your printer for the NEW firmware to be automatically applied to the SKR PRO MCU. To locate the "firmware.bin" file, and learn how to transfer it to your micro SD card go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2209 V1.2 Drivers in Stand-alone Mode for SpreadCycle

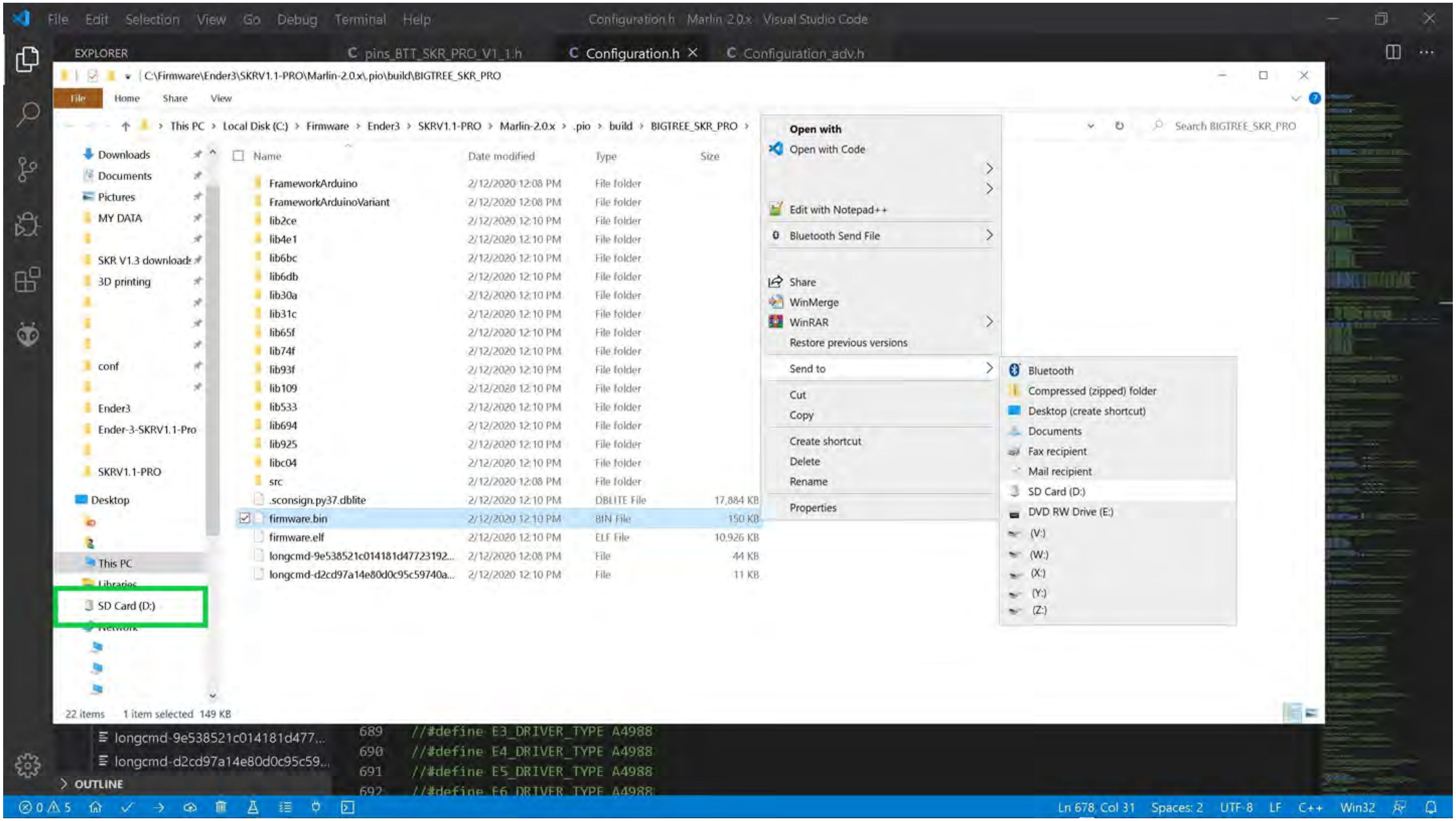
- The "firmware.bin" file can be found by using the left side of the VS code window and going to folder Marlin-2.0.X/.pio/build/ BIGTREE_SKR_PRO/ and right clicking on the "firmware.bin" file name to bring up the content menu. Select "Reveal in File Explorer" to have your Window's machine open a file explorer window.



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2209 V1.2 Drivers in Stand-alone Mode for SpreadCycle

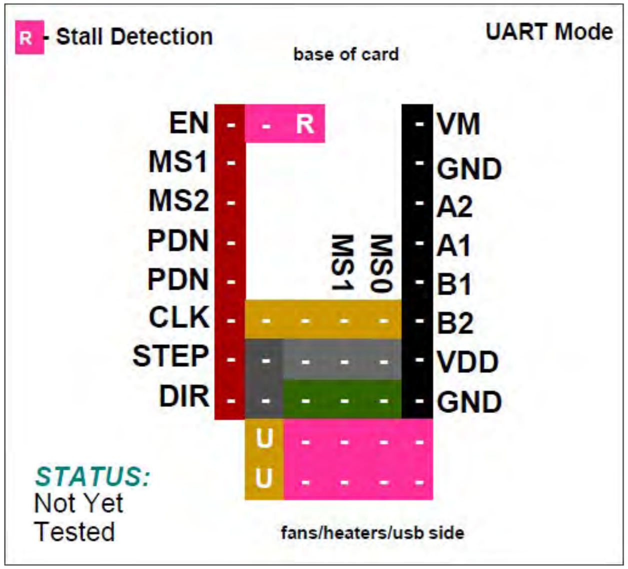
- Ensure your micro SD card, which came with your SKR PRO V1.1 board, with your SKR PRO V1.1 board is in a micro-to-SD-card adapter and plugged into your Window's SD card reader.
    - From the file explorer window right click on the "firmware.bin" file name and select "Send to" from the content menu. For my computer "D:" is my SD card reader. For your computer it may be a different drive letter. The "firmware.bin" file will be written to the SD card and renamed to "FIRMWARE.CUR".



- Right click on the SD card reader, as seen in the **GREEN** box above, and select "Eject" from the context menu. Take the micro SD card out of the adapter and place it into the micro SD card reader of the SKR PRO V1.1 board. Turn on the power to the printer (i.e. the SKR PRO V1.1 board) and the "FIRMWARE.CUR" file will be uploaded to the board. After the upload the firmware file will be renamed to "firmware.bin" and stored on the micro SD card.

BIQU TMC2209 V1.2

UART Mode



Note: You can use 50% to 90% of the calculated I_{RMS} ($I_{MAX}/1.414$) when tuning ("X_CURRENT", "Y_CURRENT", etc. the stepper motor driver in the firmware.

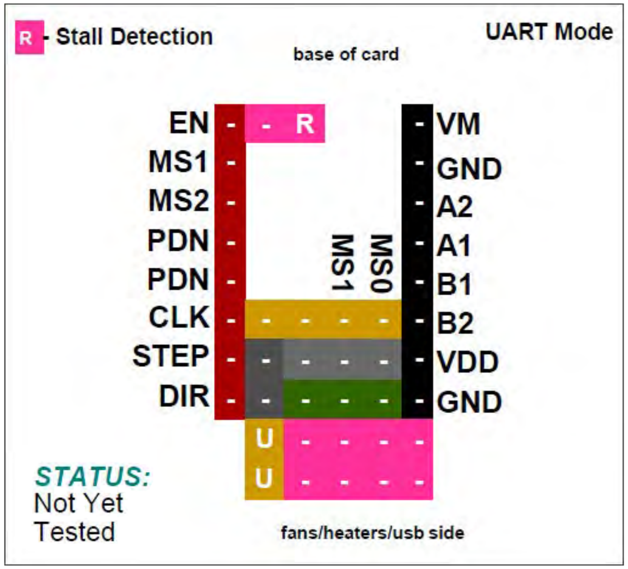
See the next page for further information.

Driver Chip	Steps are set inside of your Firmware	
BIQU® TMC2209 UART Mode Maximum 256 Subdivision 28V DC 2.8A (peak)		
Driving Current Calculation Formula R_S(Typical Sense Resistor)= 0.11Ω	$I_{MAX}=V_{ref}$ See Appendix B #6. Use 50% to 90% as shown below: $I_{MAX}=I_{MAX} * 0.90$	$V_{ref}=I_{MAX}$ See Appendix B #6. Use 50% to 90% as shown below: $V_{ref}=V_{ref} * 0.90$

NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

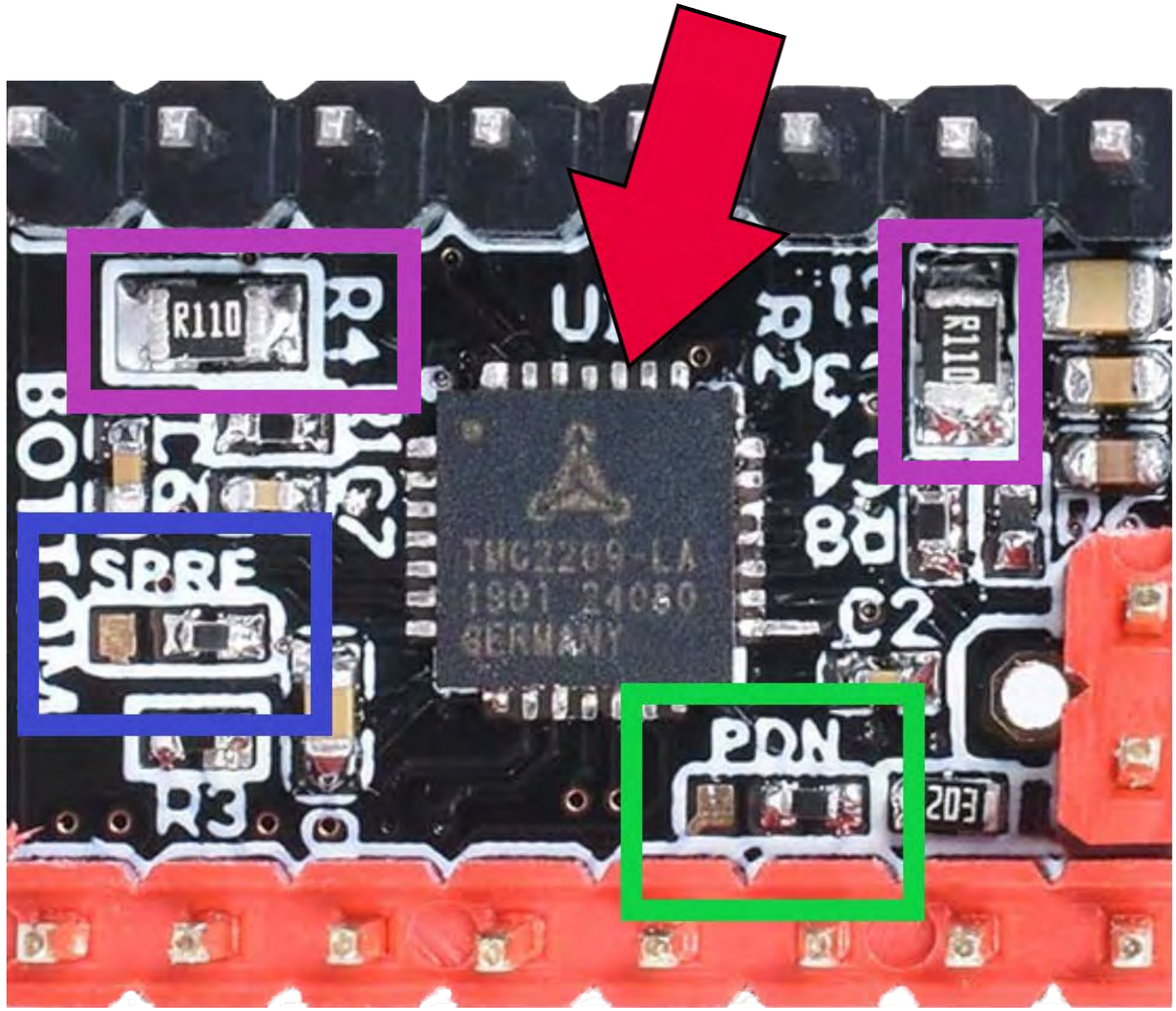
BIQU TMC2209 V1.2

UART Mode



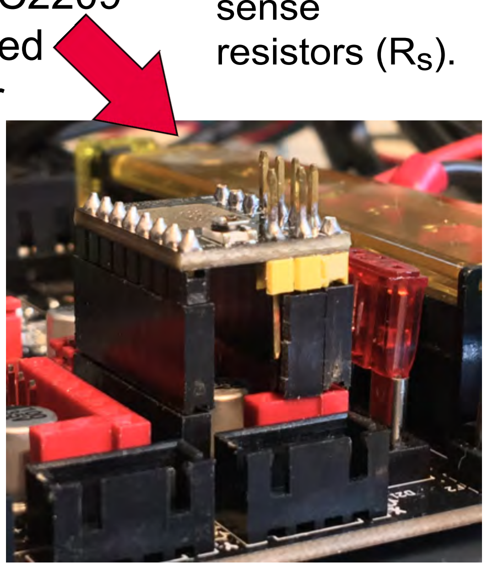
Note: To ensure your TMC2209 is in UART mode, look on the bottom of the driver board for the PDN pads. Two of the three pads should be bridged together. If a bridge exists then the device is in UART Mode, as seen in the **GREEN** box.

The **BLUE** box shows the device has StealthChop capability.



MOST BIQU TMC2209 V1.2 driver boards, when purchased for UART mode, will have the correct PDN pads already soldered together, located on the bottom of the driver board.

Note: When the stall-guard function is not used, the stall-guard pin ("R") of the TMC2209 must be removed (desoldered) or use long pin header risers so that the "R" pin does not connect to the SKR PRO V1.1 board



The **PURPLE** boxes show the location of the current sense resistors (R_S).

NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

BIQU TMC2209 V1.2

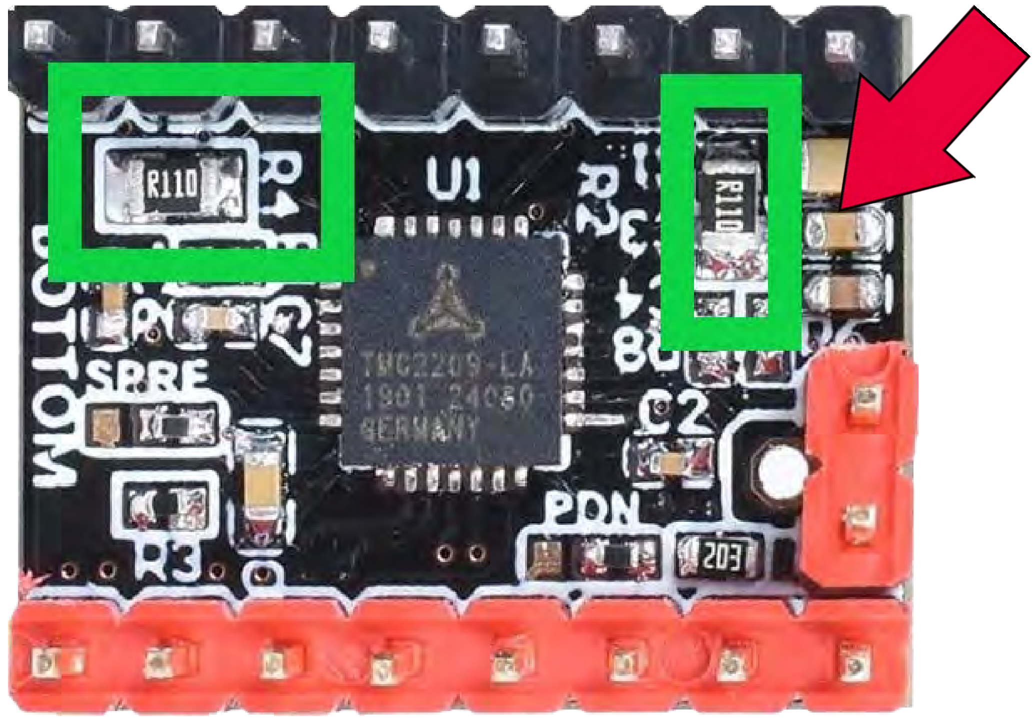
UART Mode

UART Mode

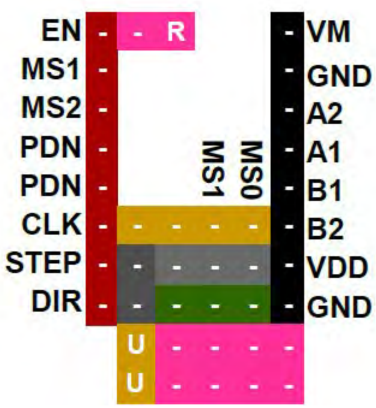
R - Stall Detection

Note: The location of the current sense resistors are shown in **GREEN**. Use the current sense resistors' value in the Marlin Firmware ("X_RSENSE", "Y_RSENSE", "Z_RSENSE" and/or "E0_RSENSE") so that the appropriate current limit can be sent to the driver board. If you do not want to use V_{ref} as the value for "X_CURRENT", "Y_CURRENT", "Z_CURRENT" and/or "E0_CURRENT", you should use I_{RMS} instead. You find I_{RMS} by taking I_{MAX} and dividing it by 1.414 ($I_{RMS}=I_{MAX}/1.414$). You use 50% to 90% of the calculated I_{RMS} as the value for "X_CURRENT", "Y_CURRENT", "Z_CURRENT", and/or "E0_CURRENT".

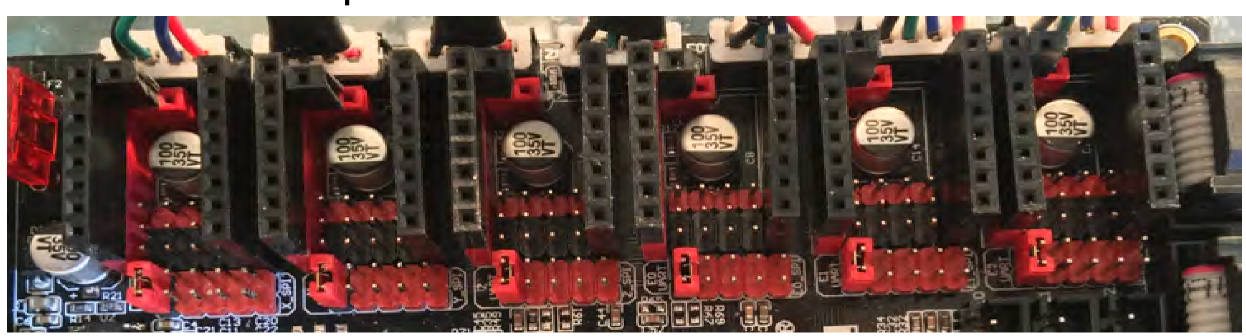
R_S = R050 is 0.05 Ohms
R_S = R062 is 0.062 Ohms
R_S = R068 is 0.068 Ohms
R_S = R075 is 0.075 Ohms
R_S = R100 is 0.1 Ohms
R_S = R110 is 0.11 Ohms
R_S = R150 is 0.15 Ohms
R_S = R200 is 0.2 Ohms
R_S = R220 is 0.22 Ohms



UART



Note: Set Jumper "U" for UART MODE!!

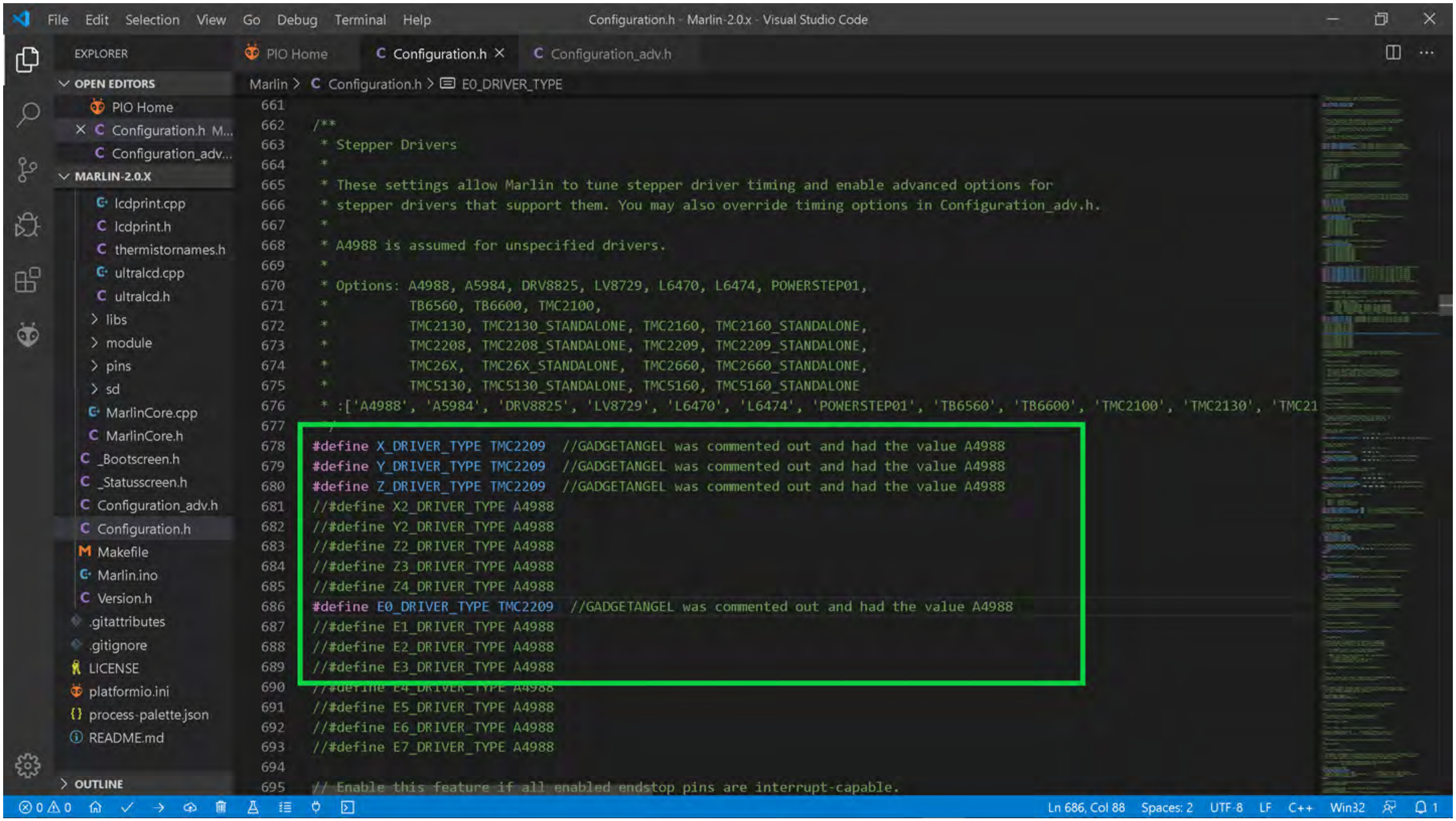


NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

The (latest release of) Marlin Setup for BIQU TMC2209 V1.2 Drivers in UART Mode

NOTE: **Go to Appendix C**, and then come back here for the changes to Marlin for BIQU TMC2209 V1.2 stepper motor drivers in UART mode.

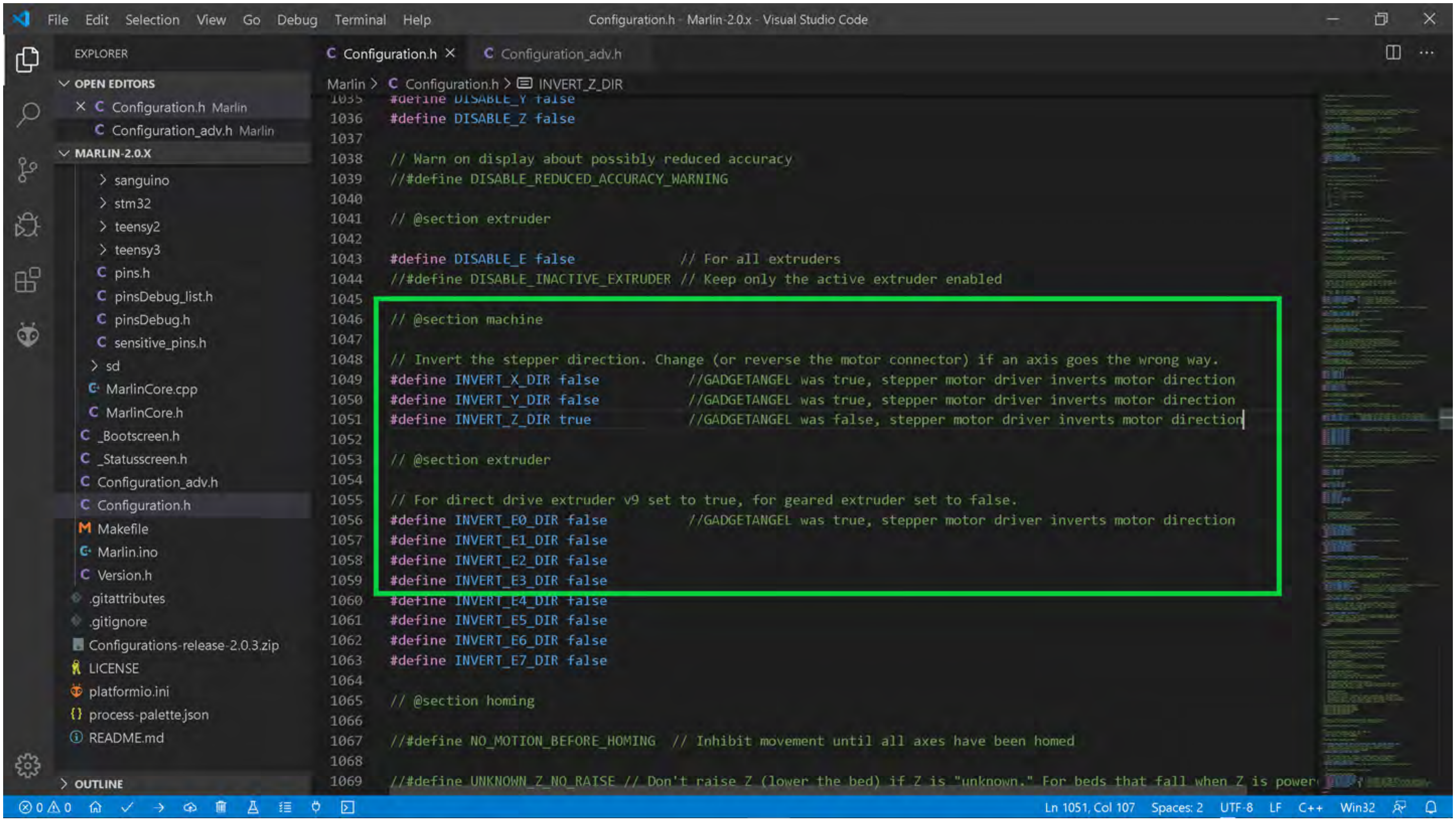
- Change the stepper motor drivers so that Marlin knows you are using TMC2209 drivers in UART mode. Change one line for each axis and one for each extruder you will be using. See the picture below for an example of how to use TMC2209 drivers in UART mode. When two "//" appear at the beginning of a line that means that line is commented out. To un-comment a line just remove the leading two forward slashes ("//").



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2209 V1.2 Drivers in UART Mode

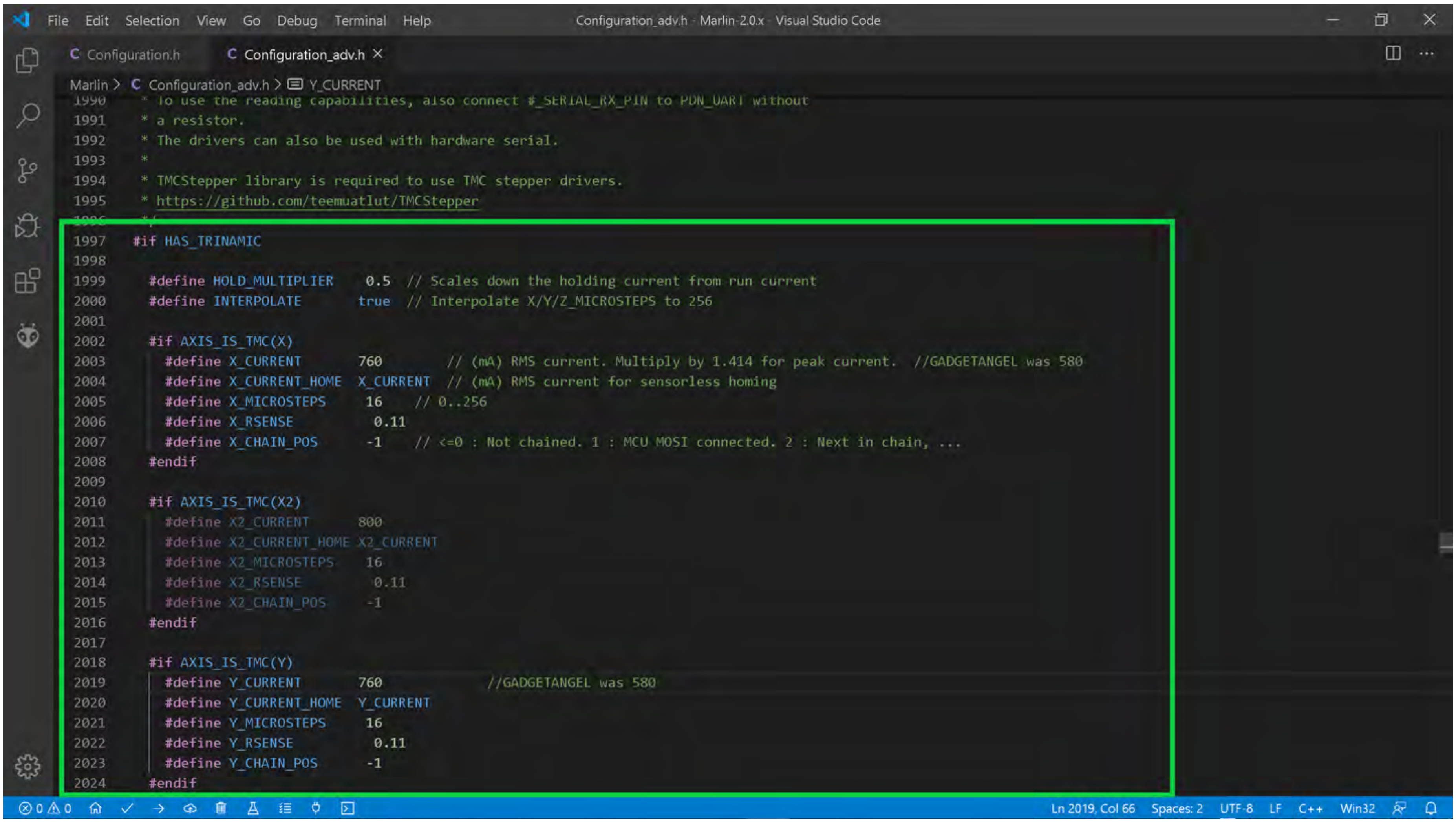
- Since the A4988 driver is what my Ender 3 used, but, now I want to use TMC2209 drivers, I must invert the stepper motor direction because the TMC2209 driver will turn the motors in the opposite direction than the A4988 driver's motor direction. So if the axis' setting you will be using the TMC2209 driver on was "true" change it to "false", as shown in the **GREEN** box below. If the setting was "false", now set it to "true", as shown in the **GREEN** box below



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2209 V1.2 Drivers in UART Mode

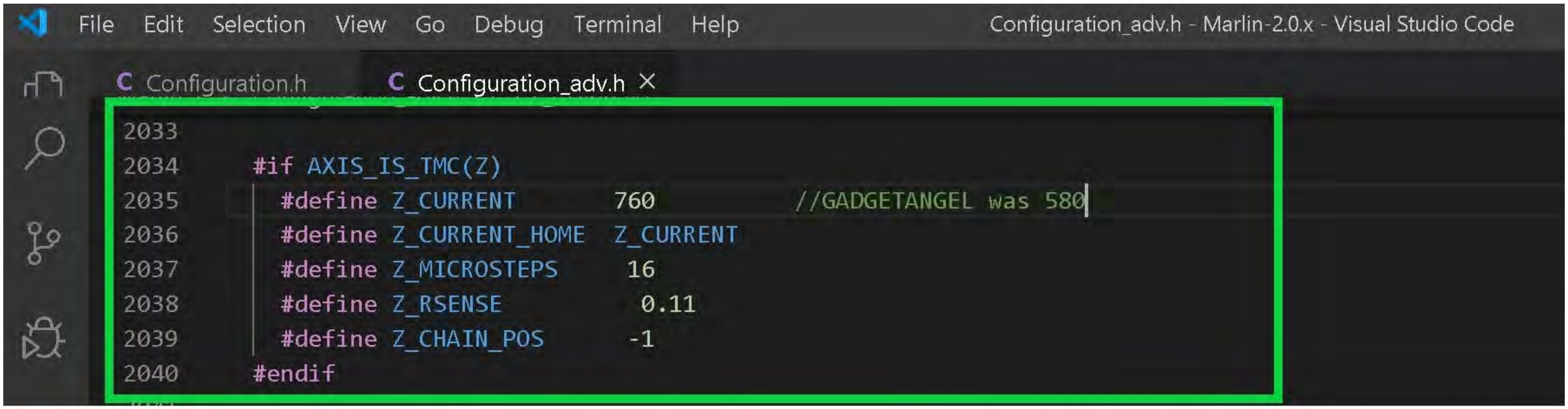
- Next you want to set your V_{ref} in the Marlin firmware for each axis that has the TMC2209 driver, as seen in the **GREEN** box below. I changed the "X_CURRENT" to be the calculated V_{ref} for my X-Axis, which is 760mV for an Ender 3. I changed the "Y_CURRENT" to be the calculated V_{ref} for my Y-Axis, which is 760mV on the Ender 3.
- Ensure "X_RSENSE" is set to 0.11. Ensure "Y_RSENSE" is set to 0.11.
- If you do not want to use V_{ref} as the value for "X_CURRENT" and/or "Y_CURRENT", you should use I_{RMS} instead. You find I_{RMS} by taking I_{MAX} and dividing it by 1.414 ($I_{RMS}=I_{MAX}/1.414$). You use 50% to 90% of the calculated I_{RMS} as the value for "X_CURRENT" and/or "Y_CURRENT".

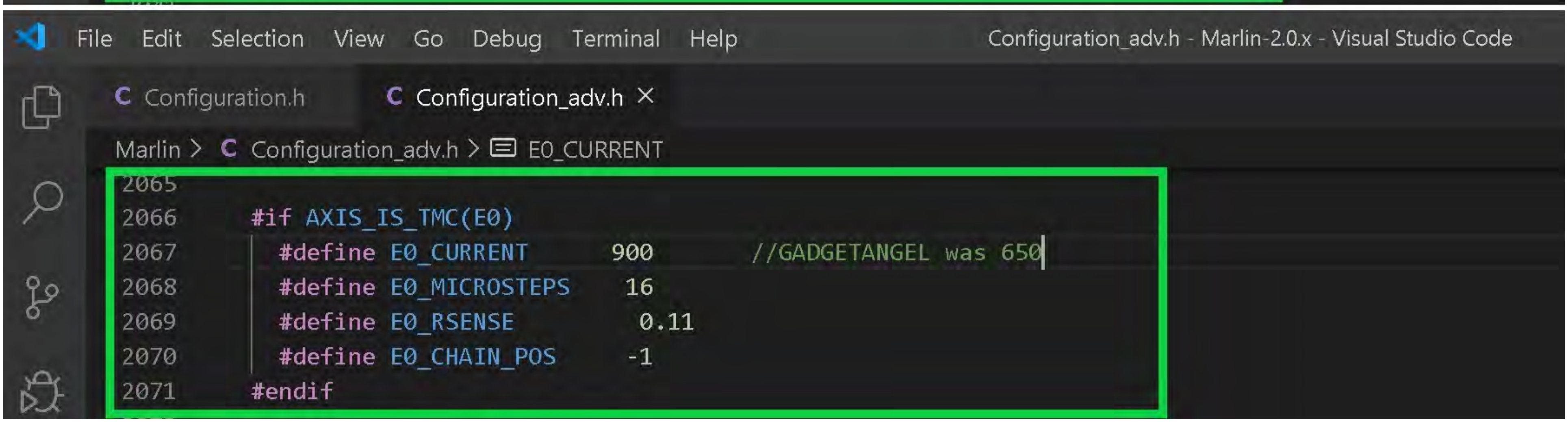


- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2209 V1.2 Drivers in UART Mode

- Now, I am setting the V_{ref} for Z-Axis and the extruder, as seen in the **GREEN** boxes below. I changed the "Z_CURRENT" to be the calculated V_{ref} for my Z-Axis, which is 760mV for an Ender 3. I changed the "E0_CURRENT" to be the calculated V_{ref} for my Extruder, which is 900mV on the Ender 3.
- Ensure "Z_RSENSE" is set to 0.11. Ensure "E0_RSENSE" is set to 0.11.
- If you do not want to use V_{ref} as the value for "Z_CURRENT" and/or "E0_CURRENT", you should use I_{RMS} instead. You find I_{RMS} by taking I_{MAX} and dividing it by 1.414 ($I_{RMS}=I_{MAX}/1.414$). You use 50% to 90% of the calculated I_{RMS} as the value for "Z_CURRENT" and/or "E0_CURRENT".

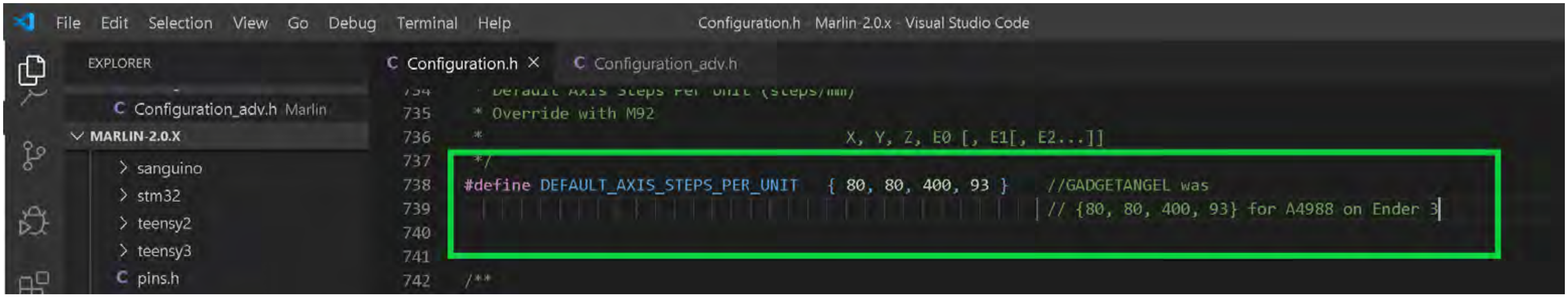




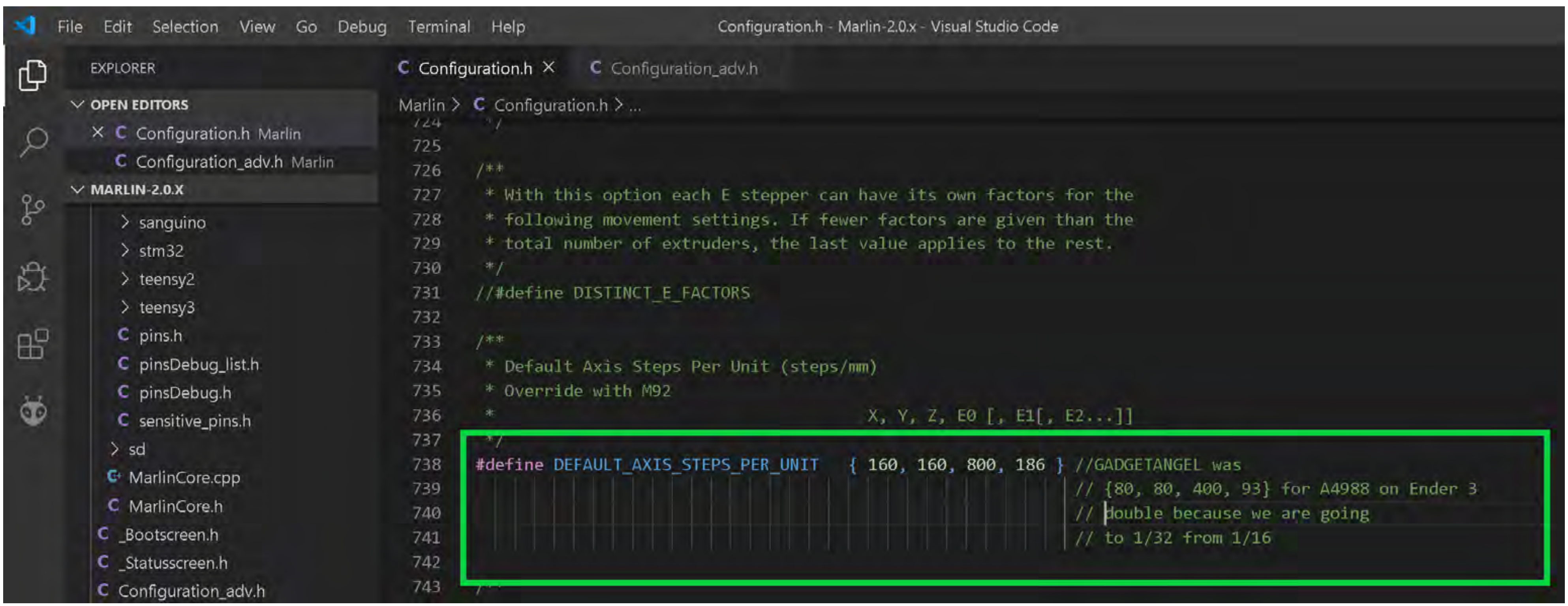
- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2209 V1.2 Drivers in UART Mode

- If you changed the "MICROSTEPS" for any of the axes then you will need to update "DEFAULT_AXIS_STEPS_PER_UNIT" to reflect your changes



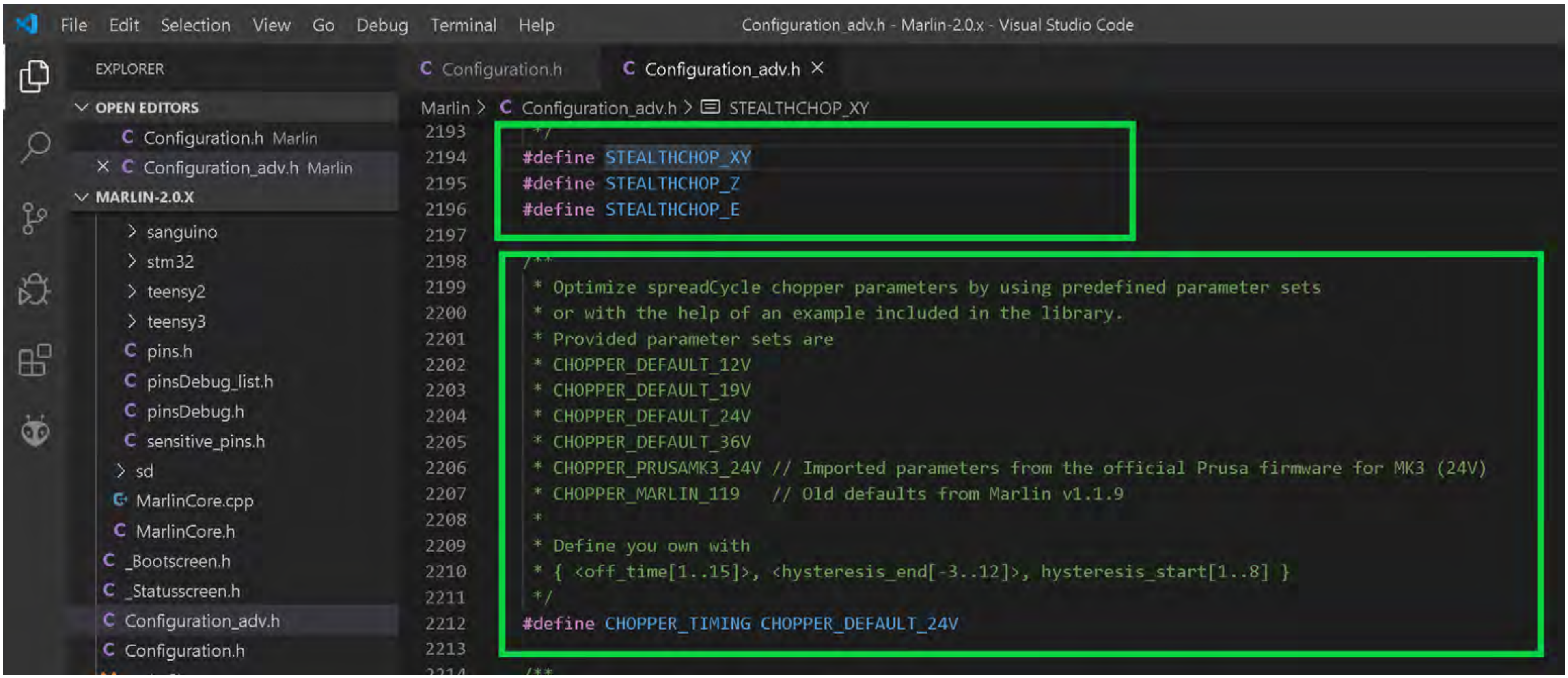
- FOR EAMPLE if you wanted to use 1/32 stepping instead of the default 1/16, you would be **doubling** your STEPS. Therefore, we must adjust our "DEFAULT_AXIS_STEPS_PER_UNIT" anytime our STEPS are NOT 1/16. So change "DEFAULT_AXIS_STEPS_PER_UNIT" to {160, 160, 800, 186}, as seen in the **GREEN** box below.



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2209 V1.2 Drivers in UART Mode

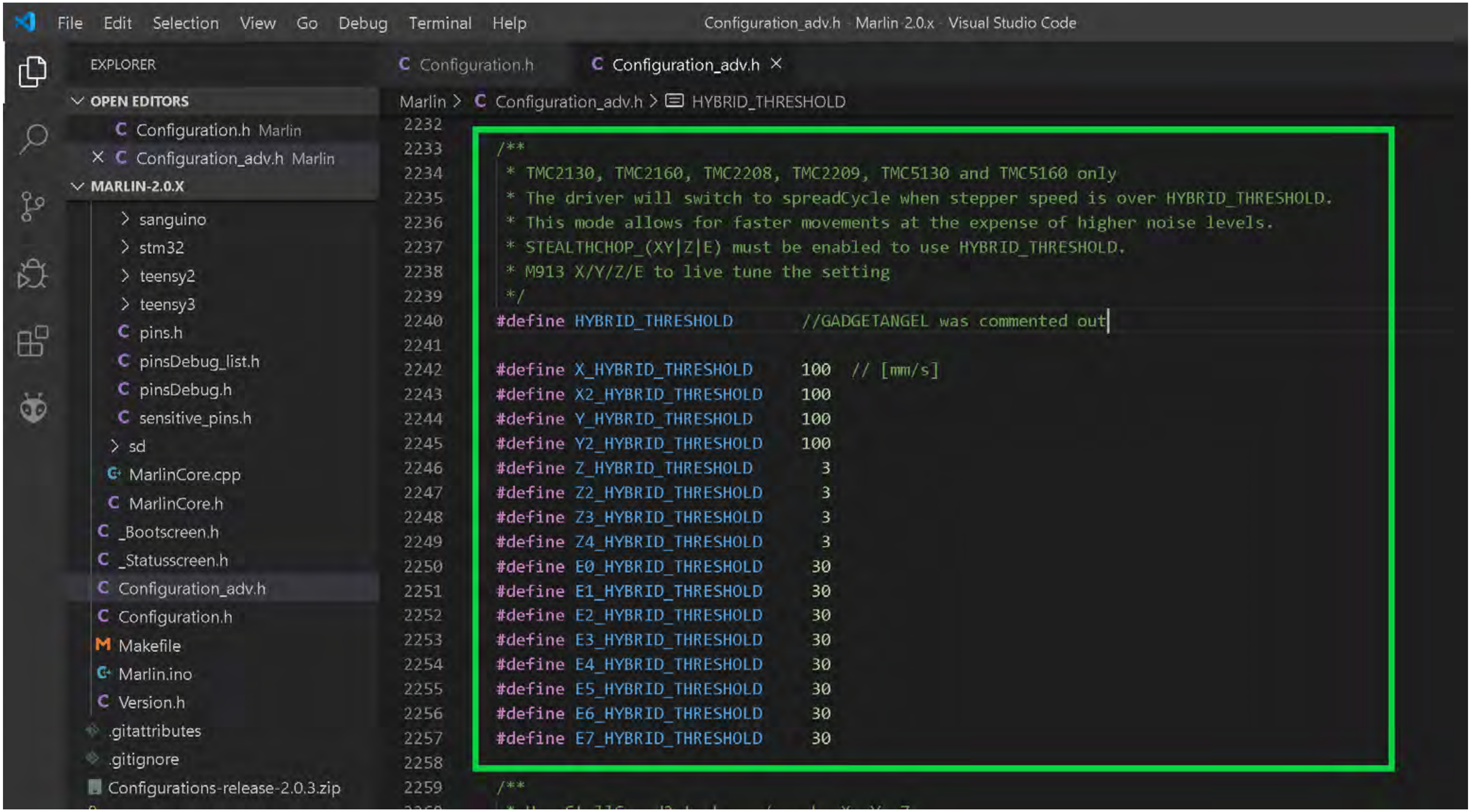
- By default stealthChop is enabled in the Marlin firmware. If you want spreadCycle ONLY then comment out the appropriate lines. I want stealthChop enabled so I want to make sure the lines are not commented out {"STEALTHCHOP_XY", "STEALTHCHOP_Z" and "STEALTHCHOP_E"}. You also want to check to see if the proper "CHOPPER_TIMING" is set for your printer. An Ender 3 is a 24VDC printer, my "CHOPPER_TIMING" is correct.



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2209 V1.2 Drivers in UART Mode

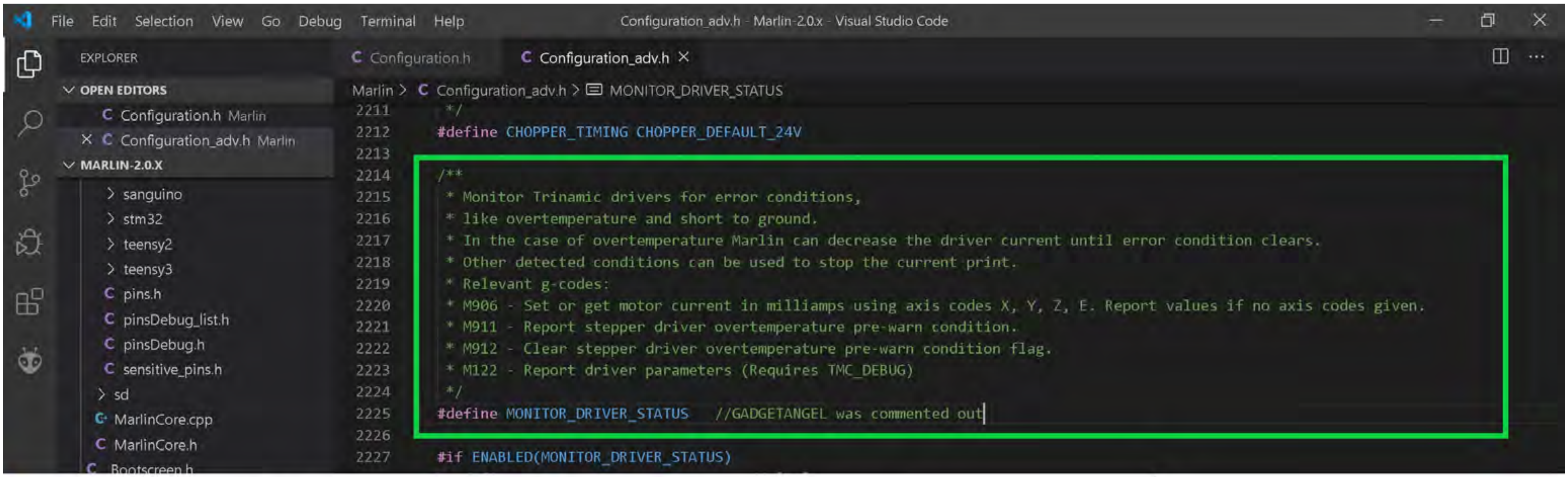
- Now you either enable "HYBRID_THRESHOLD" or disable it. By default, it is disabled. "HYBRID_THERSHOLD" allows the printer to change between stealthChop and spreadCycle dynamically depending on the print speed. I want "HYBRID_THRESHOLD" enabled so I need to remove the two leading "//", which uncomments the line in the Marlin firmware.

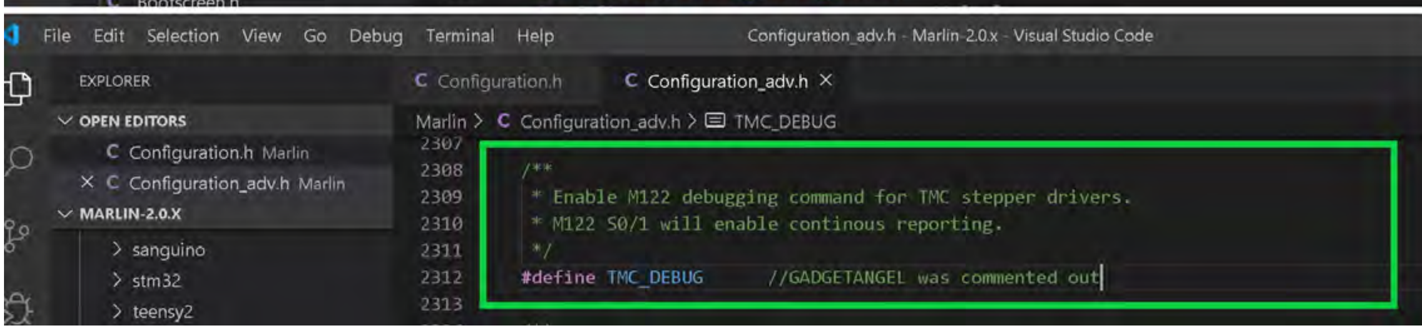


- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2209 V1.2 Drivers in UART Mode

- Now I want to enable some statements that allow me access to debugging the TMC drivers. I will uncomment "MONITOR_DRIVER_STATUS" and "TMC_DEBUG". "MONITOR_DRIVER_STATUS" will enable the following G-codes: M906, M911, and M912, "TMC_DEBUG" will enable the M122 G-code command. You can read about these from the comments in the firmware and in **Marlin's documentation located on-line.**

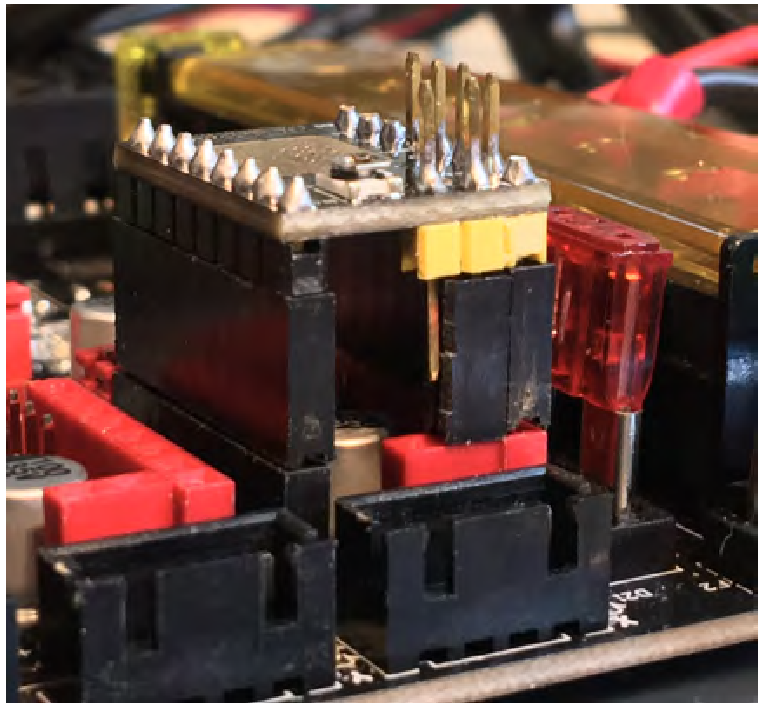




- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2209 V1.2 Drivers in UART Mode

- This next section covers sensor-less homing which is available for the TMC2209 in UART mode. I want to enable it so I will be covering sensor-less homing for the X and Y axis only. I will not be using sensor-less homing on my Z axis on my Ender 3 printer. For sensor-less homing to work the DIAG pin on the TMC2209 driver has to be plugged into the SKR PRO V1.1 board. Since I am not using sensor-less homing on my Z axis I will need to ensure that my DIAG pin on the Z axis' TMC2209 is NOT connected to the board. I plan to plug my Z axis TMC2209 into my SKR PRO V1.1 board by using long stackable header pin risers, as seen in the picture below.

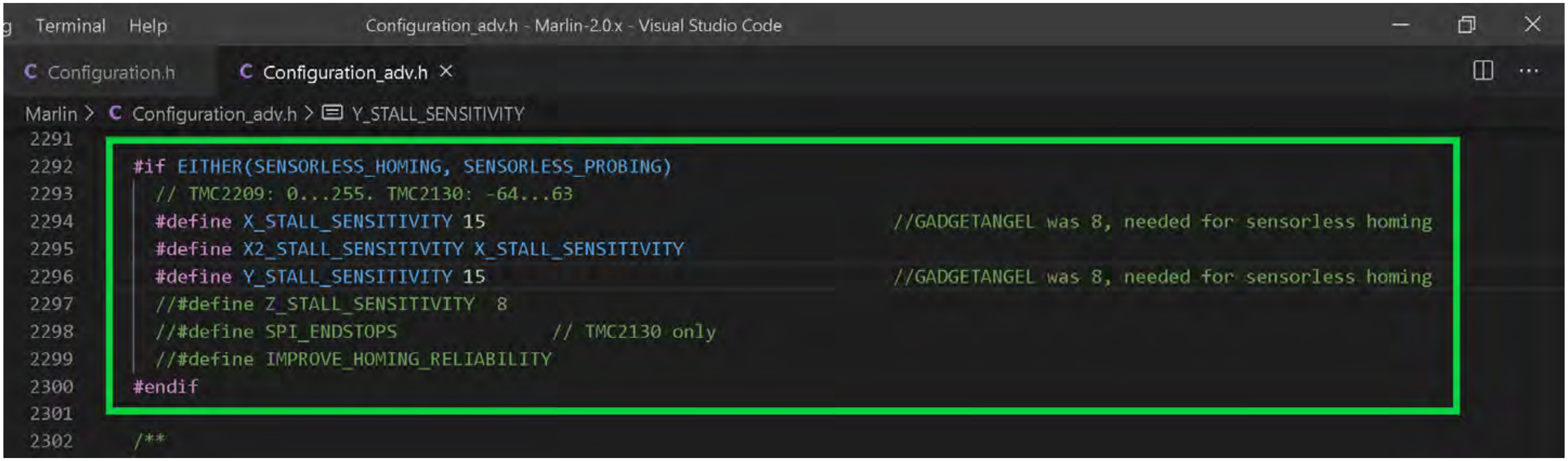


- Sensor-less homing is commented out by default. So I remove the two leading "//" to un-comment "SENSORLESS_HOMING"

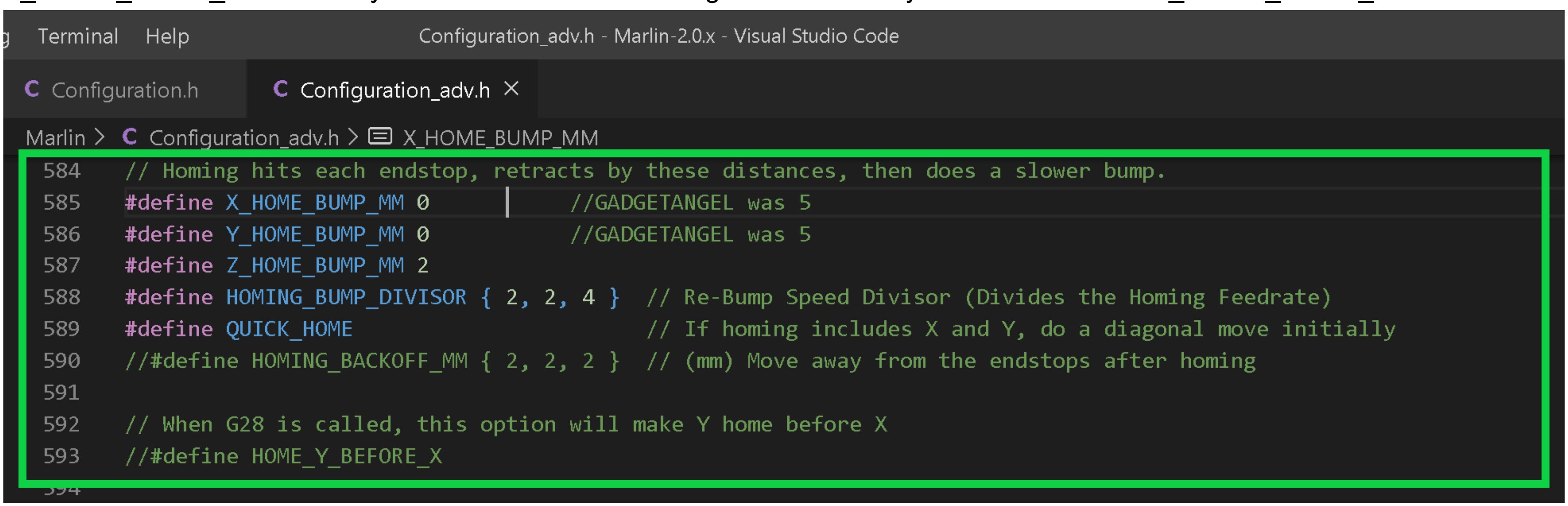
- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2209 V1.2 Drivers in UART Mode

- Next we set the "starting" stall sensitivity for sensor-less homing. I choose to make it 15. If the stall sensitivity is too high your motor will grind and not stop when it hits the end of travel on the axis. If the stall sensitivity is too low then the motor will barely move because it thinks it has hit the end of travel for the axis. Notice I only uncommented the "X_STALL_SENSITIVITY" and the "Y_STALL_SENSITIVITY". If you want sensor-less homing on the Z axis, then you will have to uncomment "Z_STALL_SENSITIVITY".



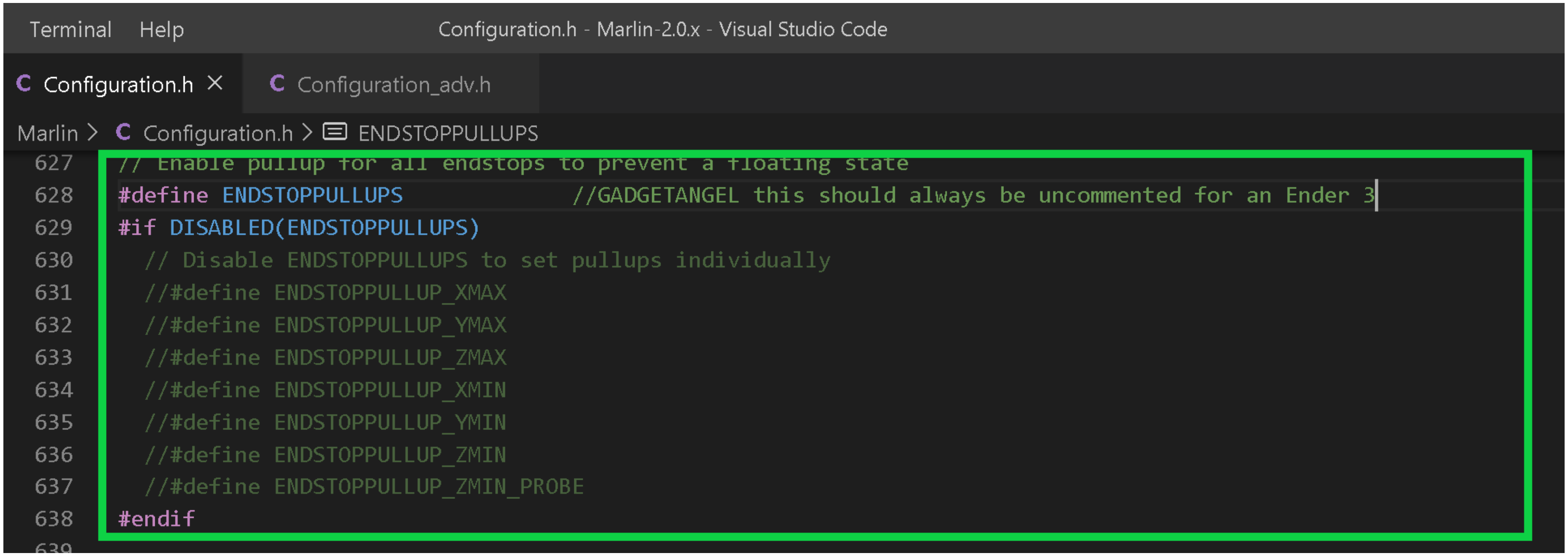
- We now have to set our home bump to 0 for each axis with sensor-less homing enabled. So I will set "X_HOME_BUMP_MM" to 0 and "Y_HOME_BUMP_MM" to 0. If you want sensor-less homing on Z axis then you will need to set "Z_HOME_BUMP_MM" to 0.



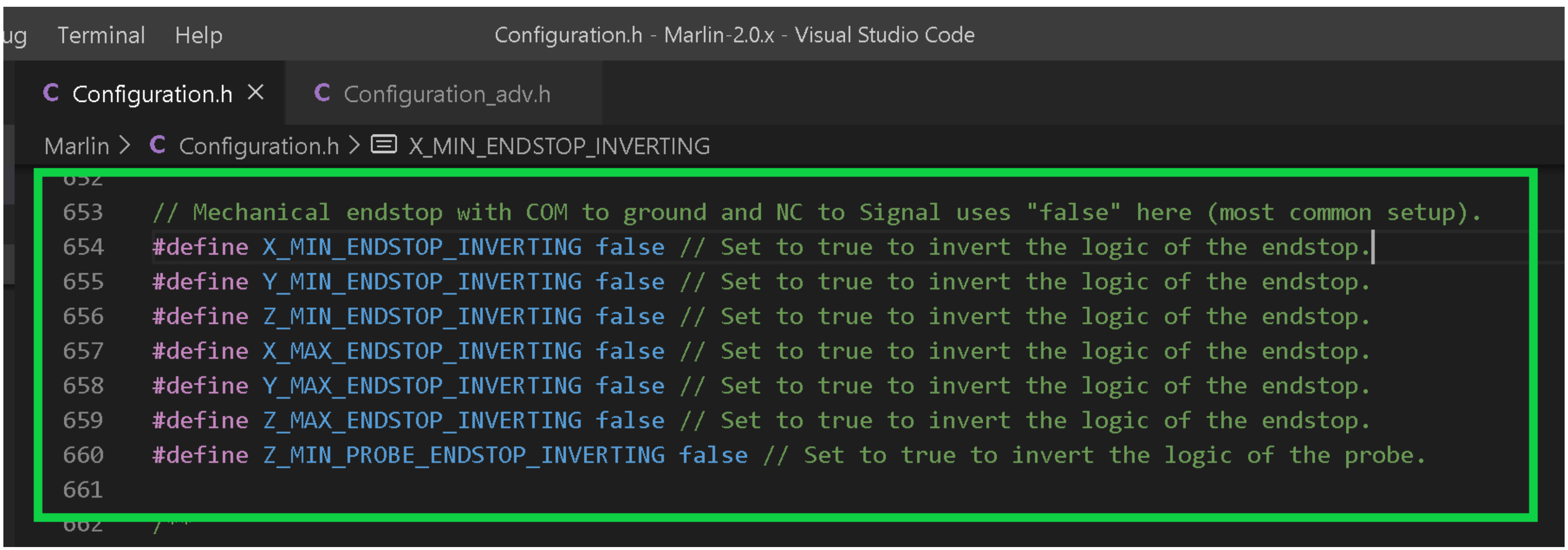
- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2209 V1.2 Drivers in UART Mode

- Let's check the firmware to ensure that "ENDSTOPPULLUPS" is enabled. It is by default. I also want to check to see how our MIN_ENDSTOP_INVERTINGs are set the right way. For and Ender 3 using TMC2209 drivers the "X_MIN_ENDSTOP_INVERTING" should be false, the "Y_MIN_ENDSTOP_INVERTING" should be false, and the "Z_MIN_ENDSTOP_INVERTING" should be false.




```
// Enable pullup for all endstops to prevent a floating state
#define ENDSTOPPULLUPS            //GADGETANGEL this should always be uncommented for an Ender 3
#if DISABLED(ENDSTOPPULLUPS)
  // Disable ENDSTOPPULLUPS to set pullups individually
  //#define ENDSTOPPULLUP_XMAX
  //#define ENDSTOPPULLUP_YMAX
  //#define ENDSTOPPULLUP_ZMAX
  //#define ENDSTOPPULLUP_XMIN
  //#define ENDSTOPPULLUP_YMIN
  //#define ENDSTOPPULLUP_ZMIN
  //#define ENDSTOPPULLUP_ZMIN_PROBE
#endif
```

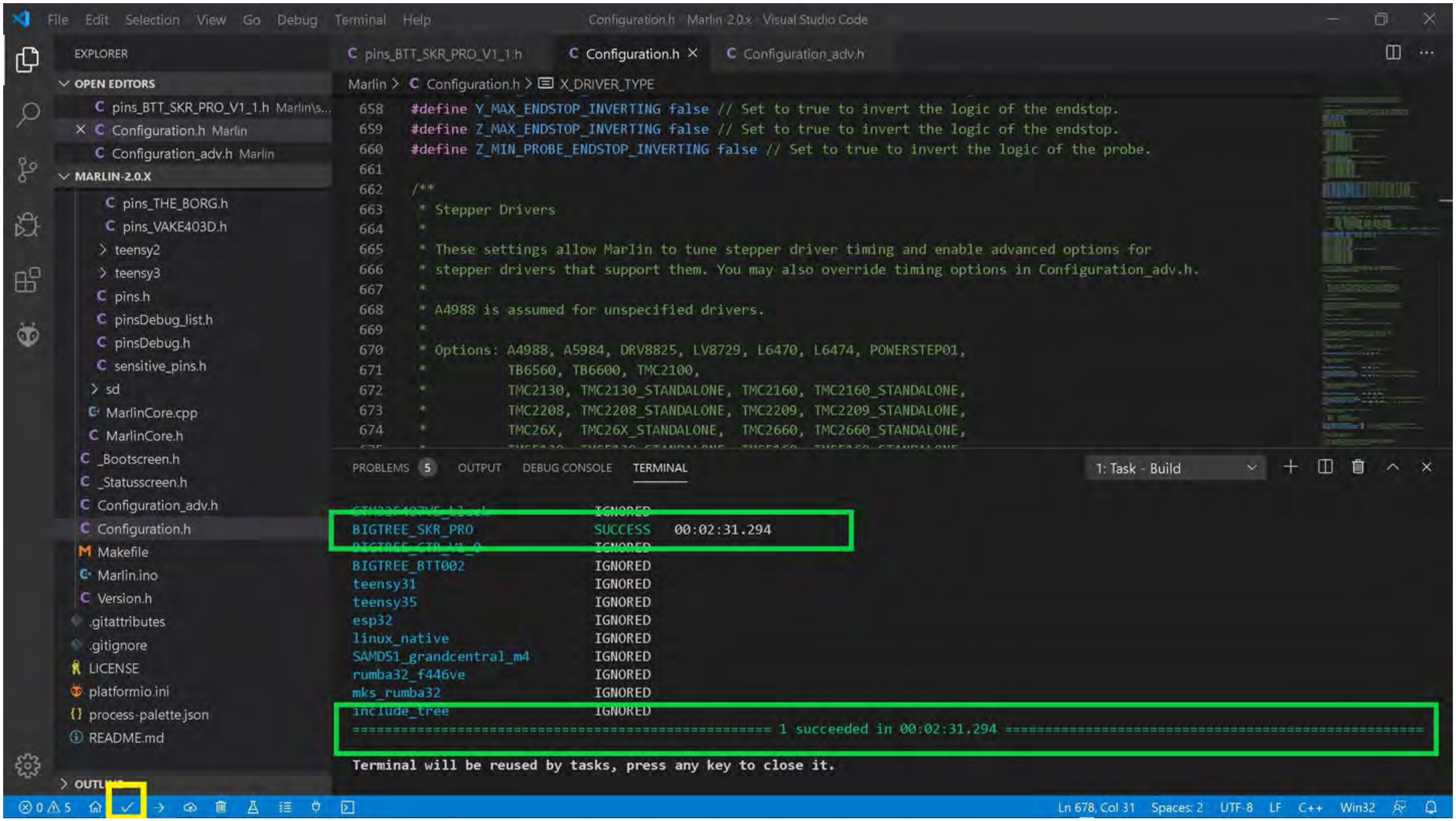



```
// Mechanical endstop with COM to ground and NC to Signal uses "false" here (most common setup).
#define X_MIN_ENDSTOP_INVERTING false // Set to true to invert the logic of the endstop.
#define Y_MIN_ENDSTOP_INVERTING false // Set to true to invert the logic of the endstop.
#define Z_MIN_ENDSTOP_INVERTING 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 false // Set to true to invert the logic of the probe.
```

- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2209 V1.2 Drivers in UART Mode

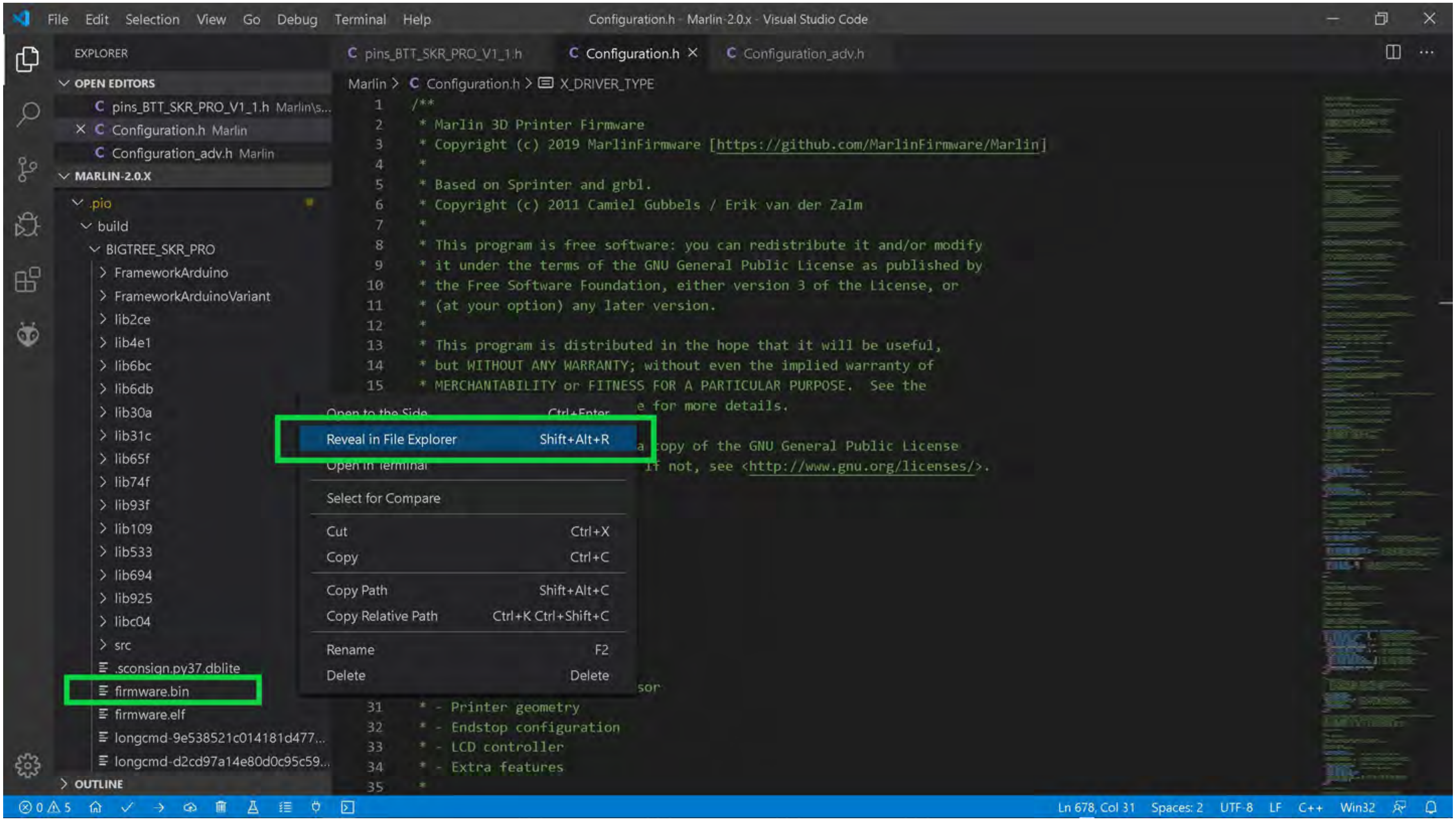
- The end of Marlin setup for BIQU TMC2209 drivers in UART mode. Now, compile your code by clicking on the check mark, as seen in the **YELLOW** box below. Once the firmware has compiled successfully you will see the following messages, as shown in the **GREEN** boxes below.



- For the SKR PRO V1.1 board you must save the "firmware.bin" file to the micro SD card then place the micro SD card into the micro SD card reader of the SKR PRO V1.1 board and turn on the power to your printer for the NEW firmware to be automatically applied to the SKR PRO MCU. To locate the "firmware.bin" file, and learn how to transfer it to your micro SD card go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2209 V1.2 Drivers in UART Mode

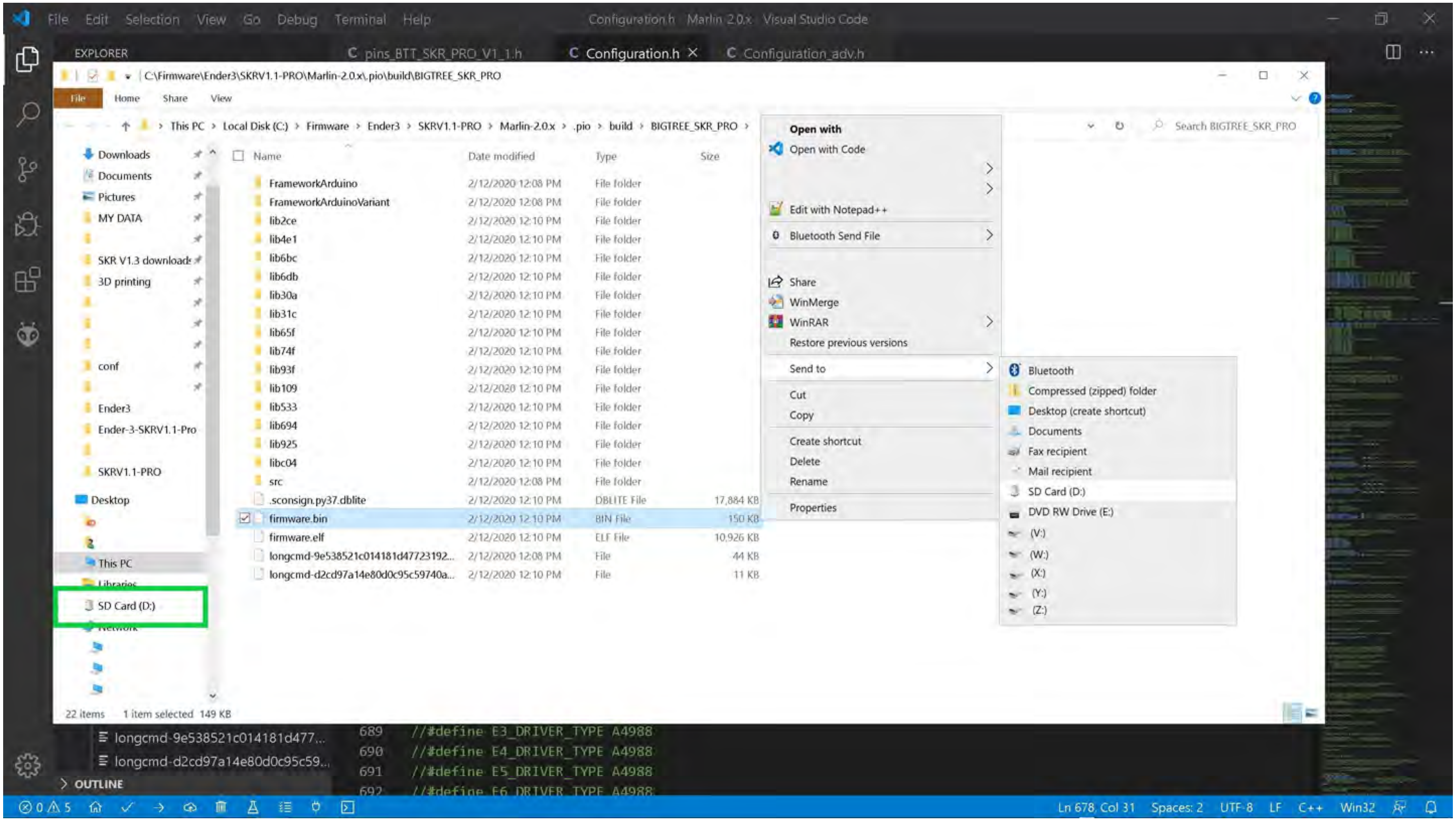
- The "firmware.bin" file can be found by using the left side of the VS code window and going to folder Marlin-2.0.X/.pio/build/ BIGTREE_SKR_PRO/ and right clicking on the "firmware.bin" file name to bring up the content menu. Select "Reveal in File Explorer" to have your Window's machine open a file explorer window.



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC2209 V1.2 Drivers in UART Mode

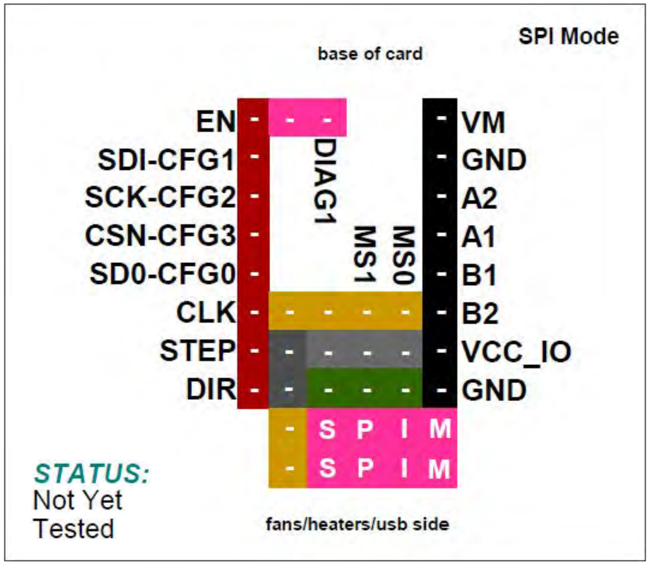
- Ensure your micro SD card, which came with your SKR PRO V1.1 board, with your SKR PRO V1.1 board is in a micro-to-SD-card adapter and plugged into your Window's SD card reader.
  - From the file explorer window right click on the "firmware.bin" file name and select "Send to" from the content menu. For my computer "D:" is my SD card reader. For your computer it may be a different drive letter. The "firmware.bin" file will be written to the SD card and renamed to "FIRMWARE.CUR".



- Right click on the SD card reader, as seen in the **GREEN** box above, and select "Eject" from the context menu. Take the micro SD card out of the adapter and place it into the micro SD card reader of the SKR PRO V1.1 board. Turn on the power to the printer (i.e. the SKR PRO V1.1 board) and the "FIRMWARE.CUR" file will be uploaded to the board. After the upload the firmware file will be renamed to "firmware.bin" and stored on the micro SD card.

BIQU TMC5160 V1.2

SPI Mode



Note: You can use 50% to 90% of the calculated I_{RMS} (I_{MAX}/1.414) when tuning ("X_CURRENT", "Y_CURRENT", etc. the stepper motor driver in the firmware.

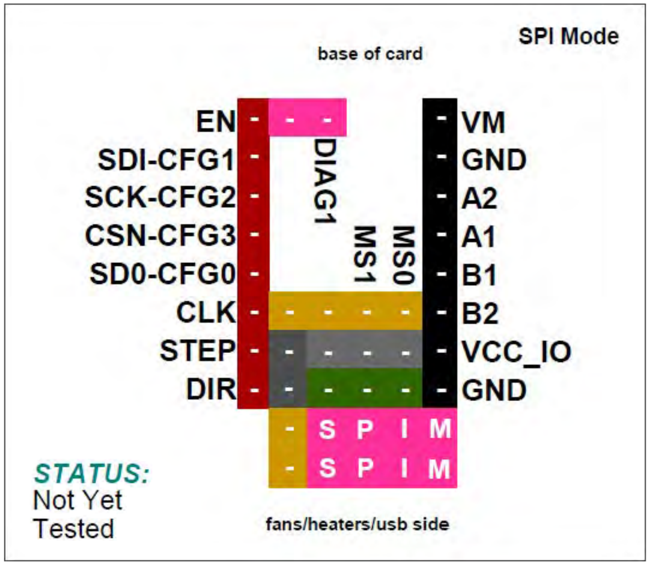
See the next page for further information.

Driver Chip	Steps are set inside of your Firmware	
BIQU® TMC5160 SPI Mode Maximum 256 Subdivision 40V DC 4.3A (peak)		
Driving Current Calculation Formula R_S(Typical Sense Resistor)=0.075 Ω	I_{MAX}=4.333 See Appendix B #8. Use 50% to 90% as shown below: $I_{MAX}=I_{MAX}*0.90=3.900$	Current Limit is set by the current sense resistors (R_s). Use 50% - 90% of I_{MAX}.

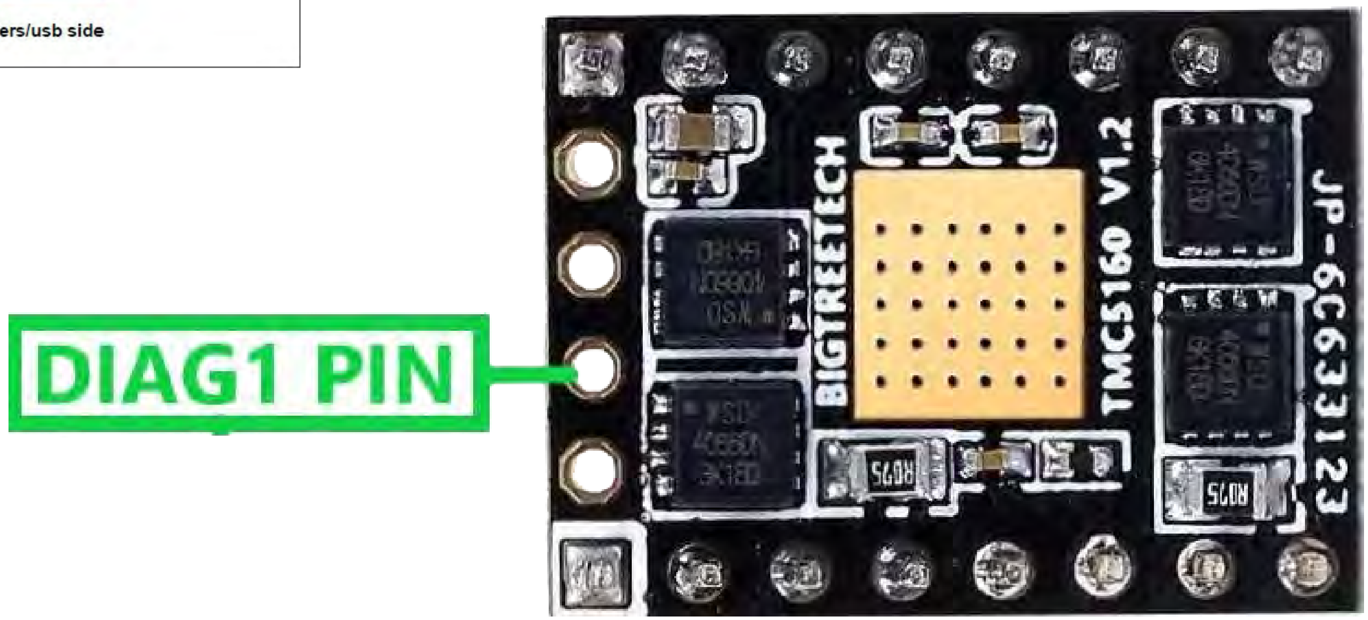
NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

BIQU TMC5160 V1.2

SPI Mode



Note: The TMC5160 V1.2 by default comes in SPI mode. The BIQU TMC5160 does NOT come with a POT or "V_{ref} Test point" location because the I_{RMS} is set inside of the Firmware.



NOTE: BIQU TMC5160 has the ability to do sensor-less homing. By default the DIAG1 pin is NOT soldered onto the driver board. Therefore, for any axis you want sensor-less homing enabled, YOU WILL HAVE to solder on the DIAG1 pin.

NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

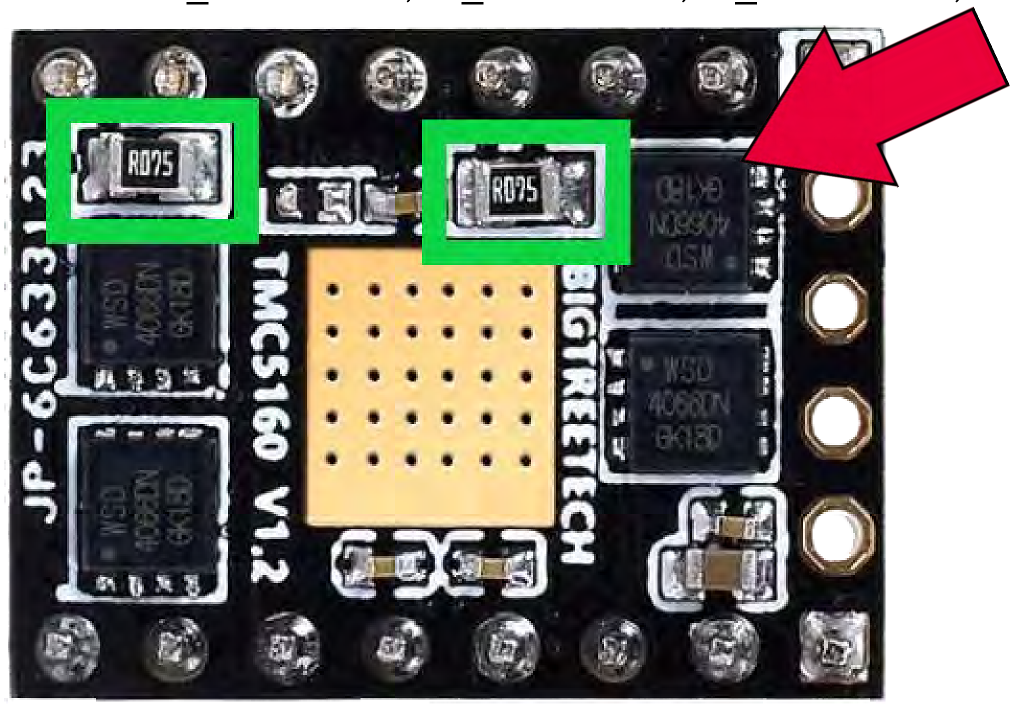
BIQU TMC5160 V1.2

SPI Mode

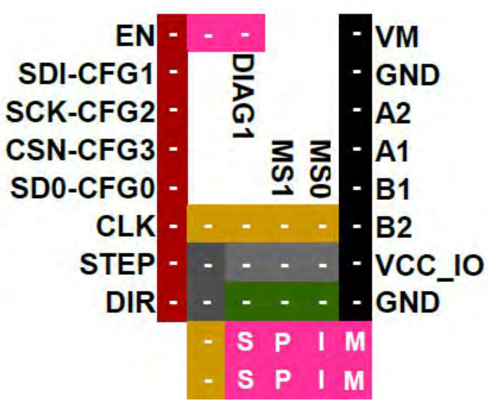
SPI Mode

Note: The location of the current sense resistors are shown in **GREEN**. Use the current sense resistors' value in the Marlin Firmware ("X_RSENSE", "Y_RSENSE", "Z_RSENSE" and/or "E0_RSENSE") so that the appropriate current limit can be sent to the driver board. If you do not want to use V_{ref} as the value for "X_CURRENT", "Y_CURRENT", "Z_CURRENT" and/or "E0_CURRENT", you should use I_{RMS} instead. You find I_{RMS} by taking I_{MAX} and dividing it by 1.414 ($I_{RMS}=I_{MAX}/1.414$). You use 50% to 90% of the calculated I_{RMS} as the value for "X_CURRENT", "Y_CURRENT", "Z_CURRENT", and/or "E0_CURRENT".

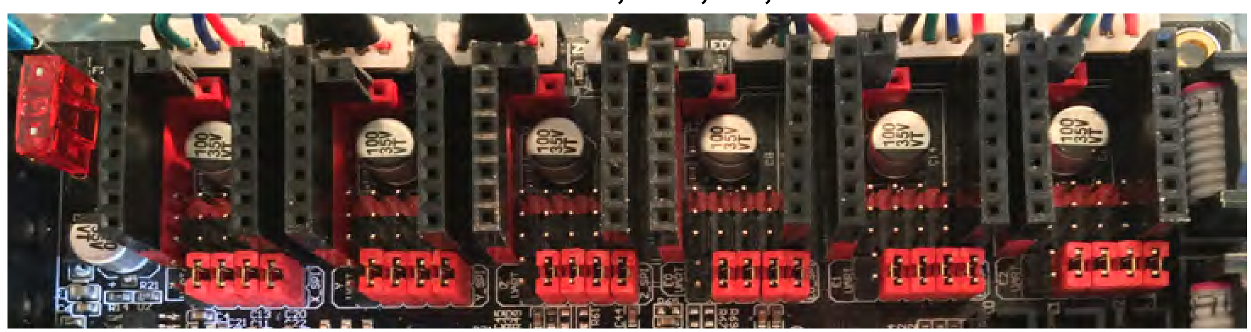
R_S = R050 is 0.05 Ohms
R_S = R062 is 0.062 Ohms
R_S = R068 is 0.068 Ohms
R_S = R075 is 0.075 Ohms
R_S = R100 is 0.1 Ohms
R_S = R110 is 0.11 Ohms
R_S = R150 is 0.15 Ohms
R_S = R200 is 0.2 Ohms
R_S = R220 is 0.22 Ohms



SPI



Note: Set JUMPERS "S","P","I","M" on the board!!

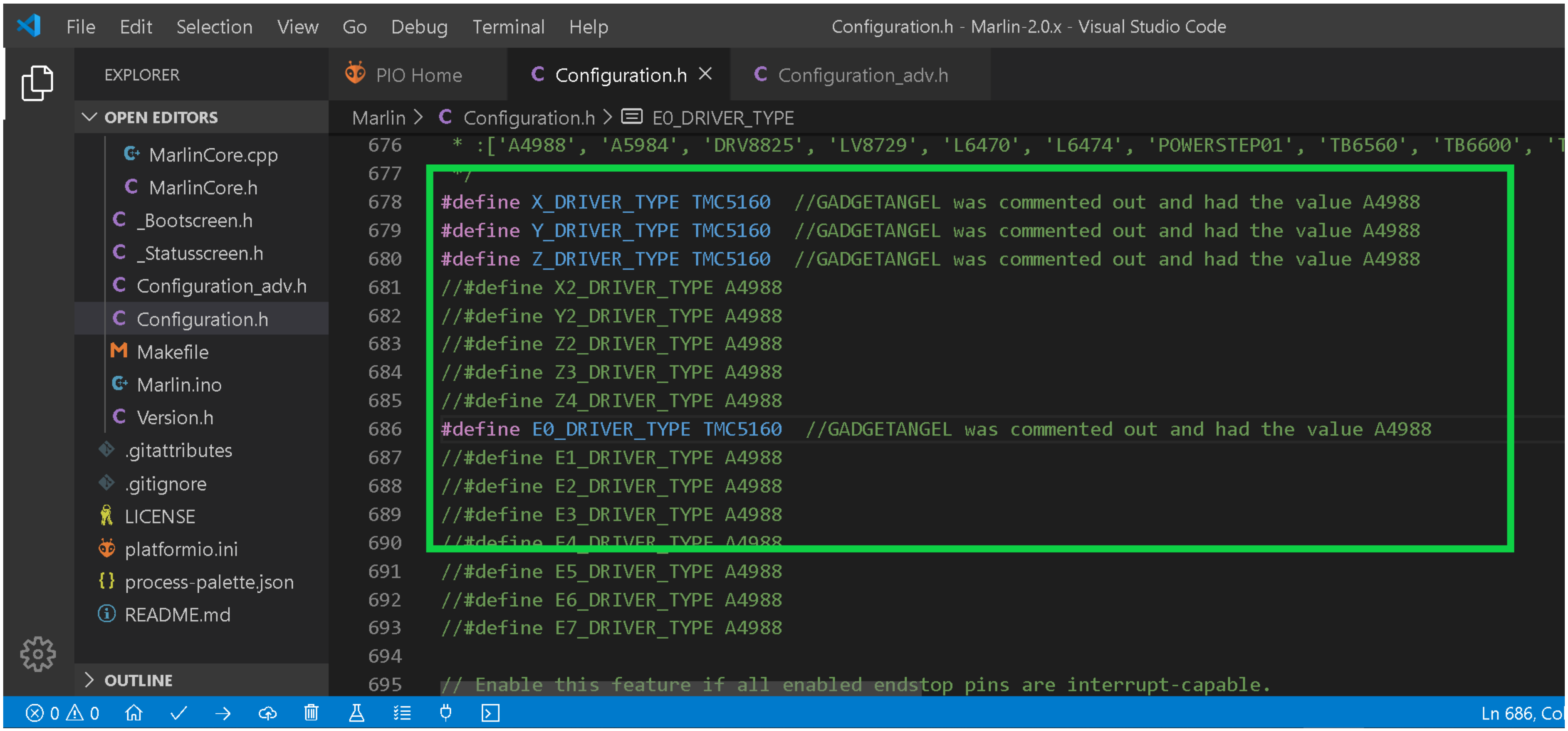


NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

The (latest release of) Marlin Setup for BIQU TMC5160 V1.2 Drivers in SPI Mode

NOTE: **Go to Appendix C**, and then come back here for the changes to Marlin for BIQU TMC5160 V1.2 stepper motor drivers in SPI mode.

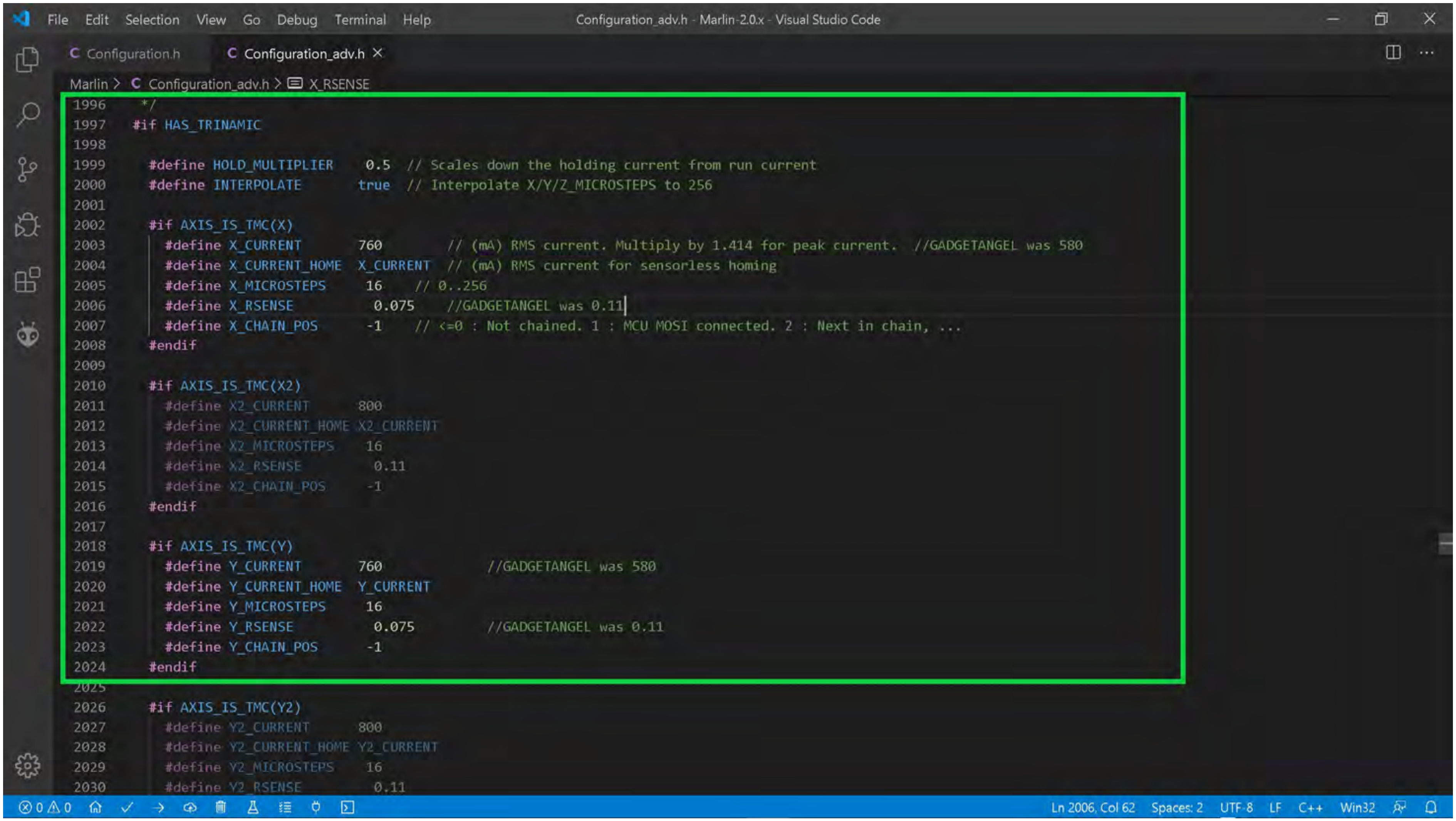
- Change the stepper motor drivers so that Marlin knows you are using TMC5160 drivers in SPI mode. Change one line for each axis and one for each extruder you will be using. See the picture below for an example of how to use TMC5160 drivers in SPI mode. When two "//" appear at the beginning of a line that means that line is commented out. To un-comment a line just remove the leading two forward slashes ("//").



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC5160 V1.2 Drivers in SPI Mode

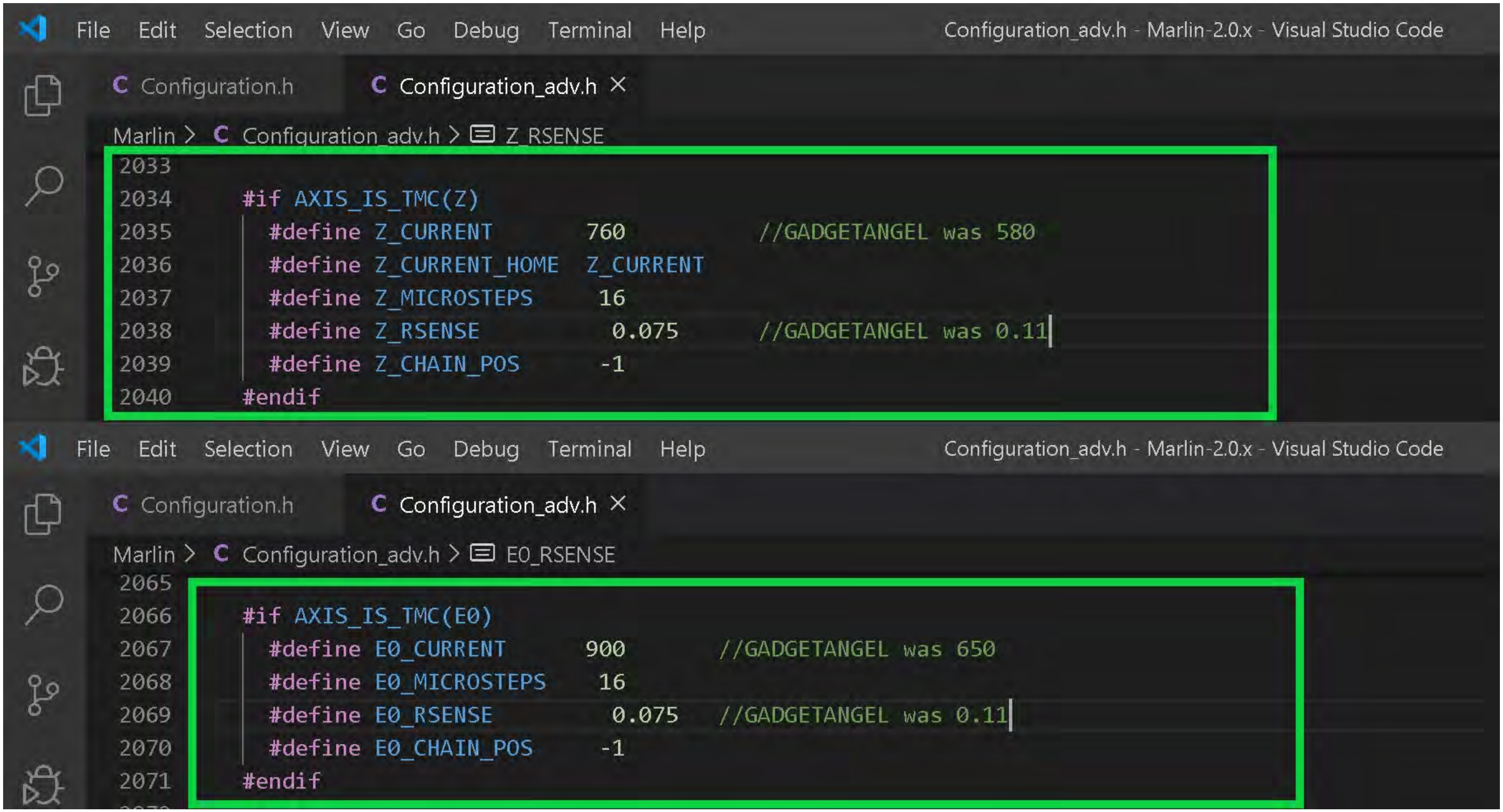
- Next you want to set your V_{ref} in the Marlin firmware for each axis that has the TMC5160 driver, as seen in the **GREEN** box below. I changed the "X_CURRENT" to be the calculated V_{ref} for my X-Axis, which is 760mV for an Ender 3. I changed the "Y_CURRENT" to be the calculated V_{ref} for my Y-Axis, which is 760mV on the Ender 3.
- Ensure "X_RSENSE" is set to 0.075. Ensure "Y_RSENSE" is set to 0.075.
- If you do not want to use V_{ref} as the value for "X_CURRENT" and/or "Y_CURRENT", you should use I_{RMS} instead. You find I_{RMS} by taking I_{MAX} and dividing it by 1.414 ($I_{RMS}=I_{MAX}/1.414$). You use 50% to 90% of the calculated I_{RMS} as the value for "X_CURRENT" and/or "Y_CURRENT".



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC5160 V1.2 Drivers in SPI Mode

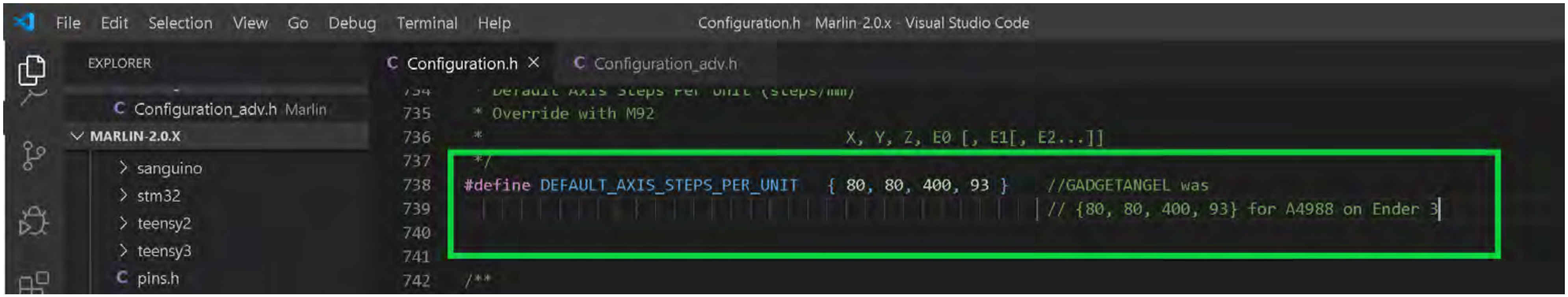
- Now, I am setting the V_{ref} for Z-Axis and the extruder, as seen in the **GREEN** boxes below. I changed the "Z_CURRENT" to be the calculated V_{ref} for my Z-Axis, which is 760mV for an Ender 3. I changed the "E0_CURRENT" to be the calculated V_{ref} for my Extruder, which is 900mV on the Ender 3.
- Ensure "Z_RSENSE" is set to 0.075. Ensure "E0_RSENSE" is set to 0.075.
- If you do not want to use V_{ref} as the value for "Z_CURRENT" and/or "E0_CURRENT", you should use I_{RMS} instead. You find I_{RMS} by taking I_{MAX} and dividing it by 1.414 ($I_{RMS}=I_{MAX}/1.414$). You use 50% to 90% of the calculated I_{RMS} as the value for "Z_CURRENT" and/or "E0_CURRENT".



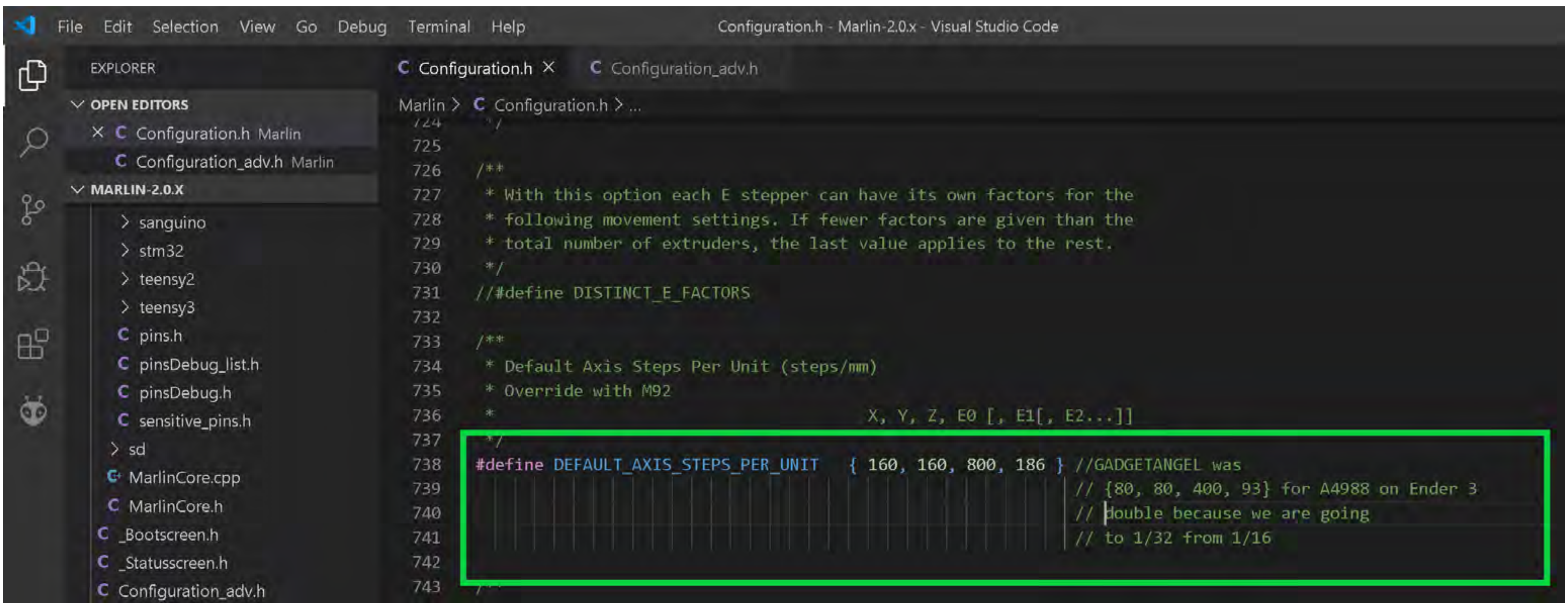
- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC5160 V1.2 Drivers in SPI Mode

- If you changed the "MICROSTEPS" for any of the axes then you will need to update "DEFAULT_AXIS_STEPS_PER_UNIT" to reflect your changes



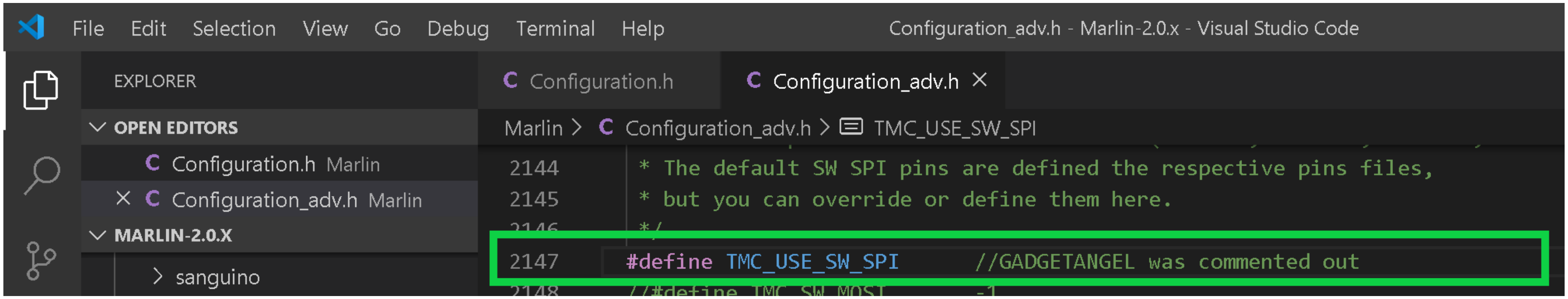
- FOR EAMPLE if you wanted to use 1/32 stepping instead of the default 1/16, you would be **doubling** your STEPS. Therefore, we must adjust our "DEFAULT_AXIS_STEPS_PER_UNIT" anytime our STEPS are NOT 1/16. So change "DEFAULT_AXIS_STEPS_PER_UNIT" to {160, 160, 800, 186}, as seen in the **GREEN** box below.



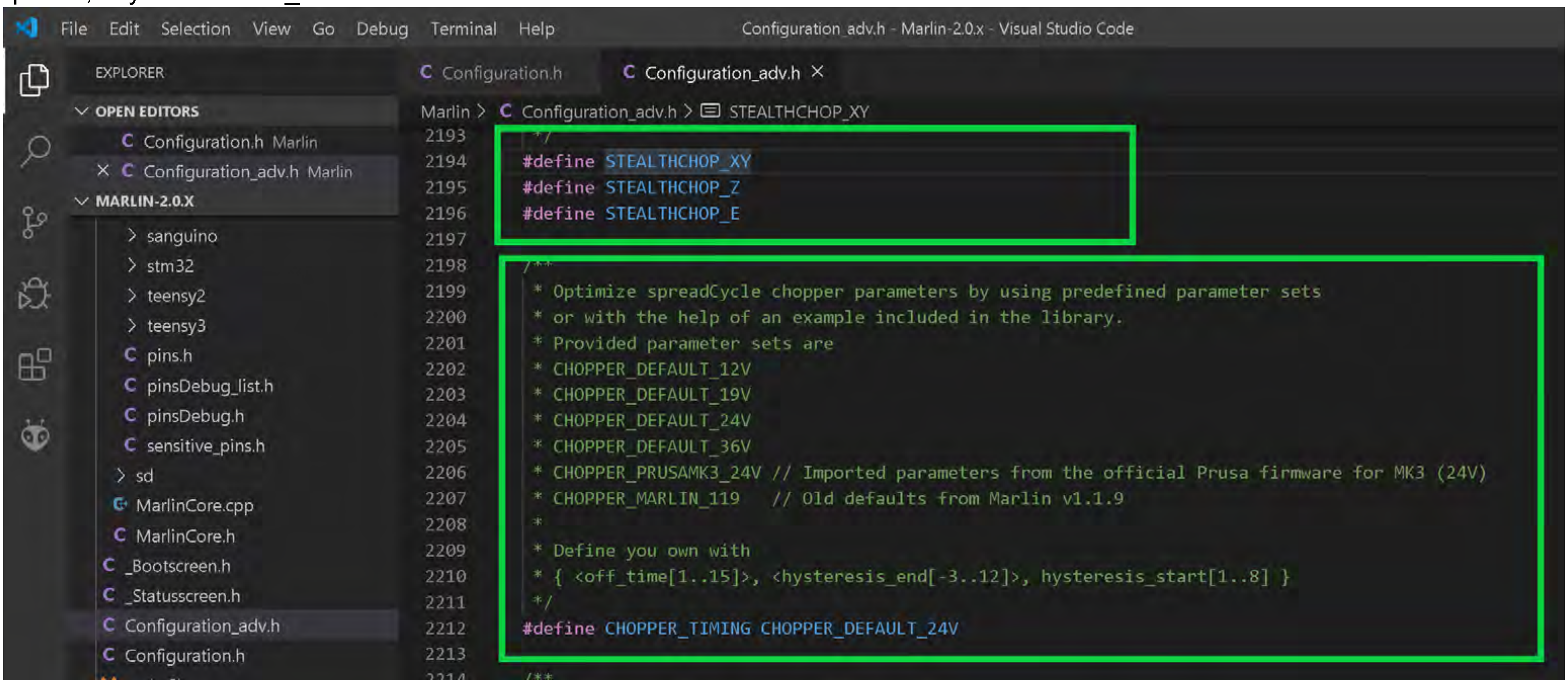
- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC5160 V1.2 Drivers in SPI Mode

- We need to uncomment out the "TMC_USE_SW_SPI" because the SKR PRO V1.1 pins file depends on this variable to define its SPI pins



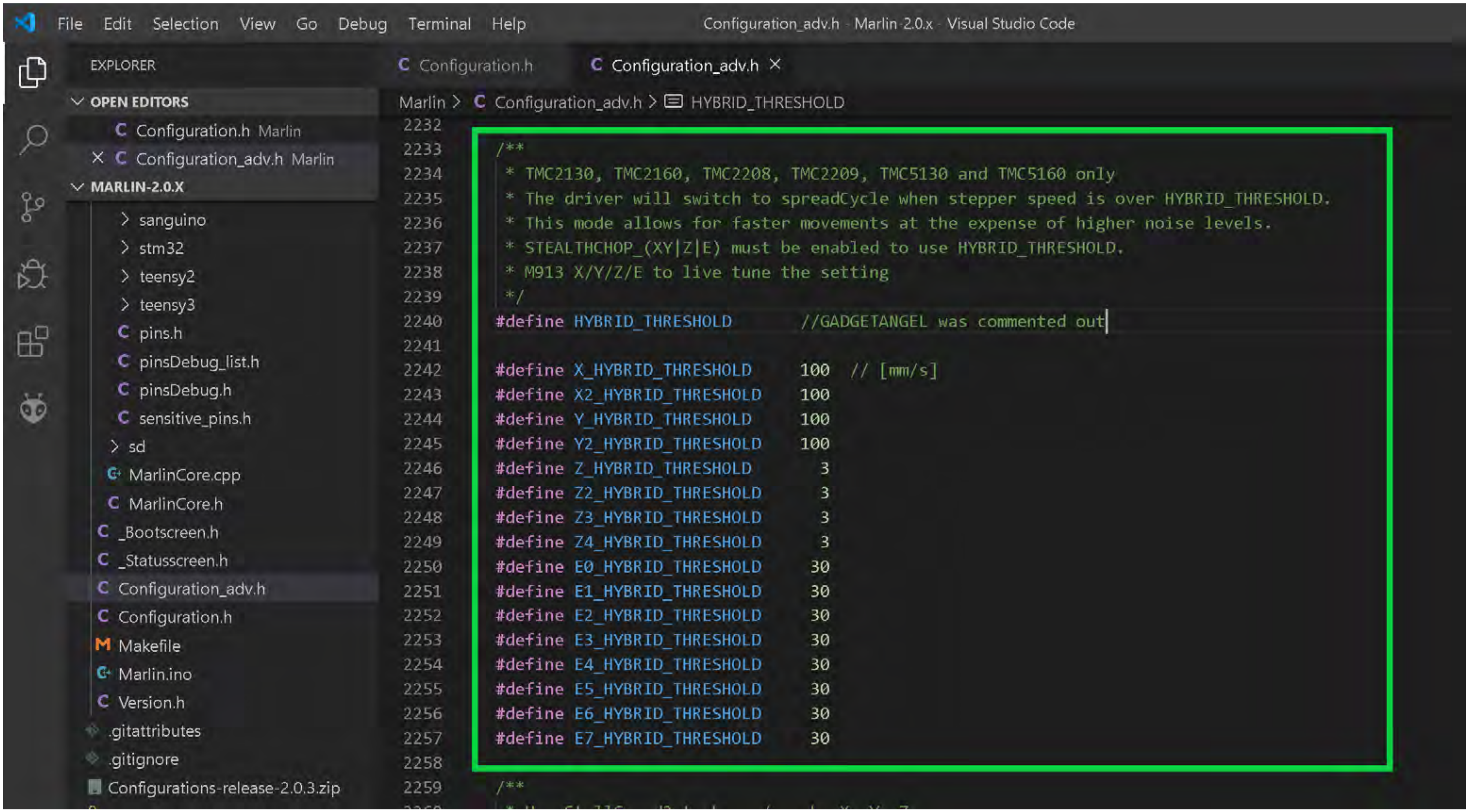
- By default stealthChop is enabled in the Marlin firmware. If you want spreadCycle ONLY then comment out the appropriate lines. I want stealthChop enabled so I want to make sure the lines are not commented out {"STEALTHCHOP_XY", "STEALTHCHOP_Z" and "STEALTHCHOP_E"}. You also want to check to see if the proper "CHOPPER_TIMING" is set for your printer. An Ender 3 is a 24VDC printer, my "CHOPPER_TIMING" is correct.



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC5160 V1.2 Drivers in SPI Mode

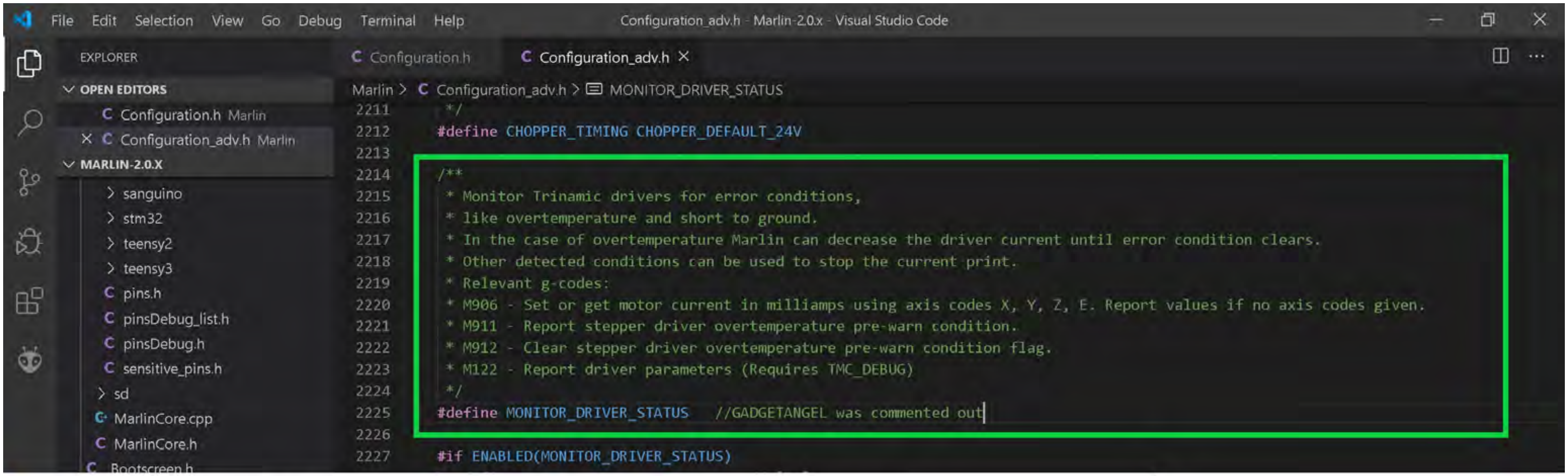
- Now you either enable "HYBRID_THRESHOLD" or disable it. By default, it is disabled. "HYBRID_THERSHOLD" allows the printer to change between stealthChop and spreadCycle dynamically depending on the print speed. I want "HYBRID_THRESHOLD" enabled so I need to remove the two leading "//", which uncomments the line in the Marlin firmware.

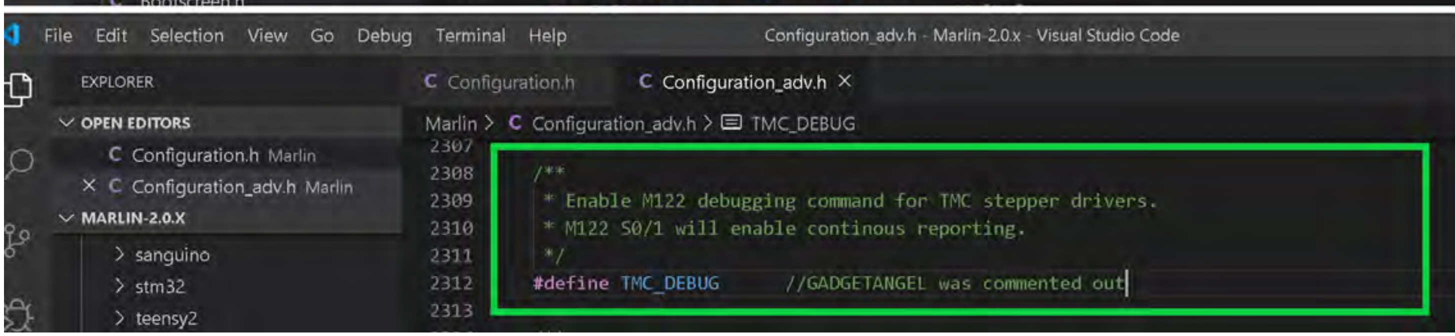


- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC5160 V1.2 Drivers in SPI Mode

- Now I want to enable some statements that allow me access to debugging the TMC drivers. I will uncomment "MONITOR_DRIVER_STATUS" and "TMC_DEBUG". "MONITOR_DRIVER_STATUS" will enable the following G-codes: M906, M911, and M912, "TMC_DEBUG" will enable the M122 G-code command. You can read about these from the comments in the firmware and in **Marlin's documentation located on-line.**

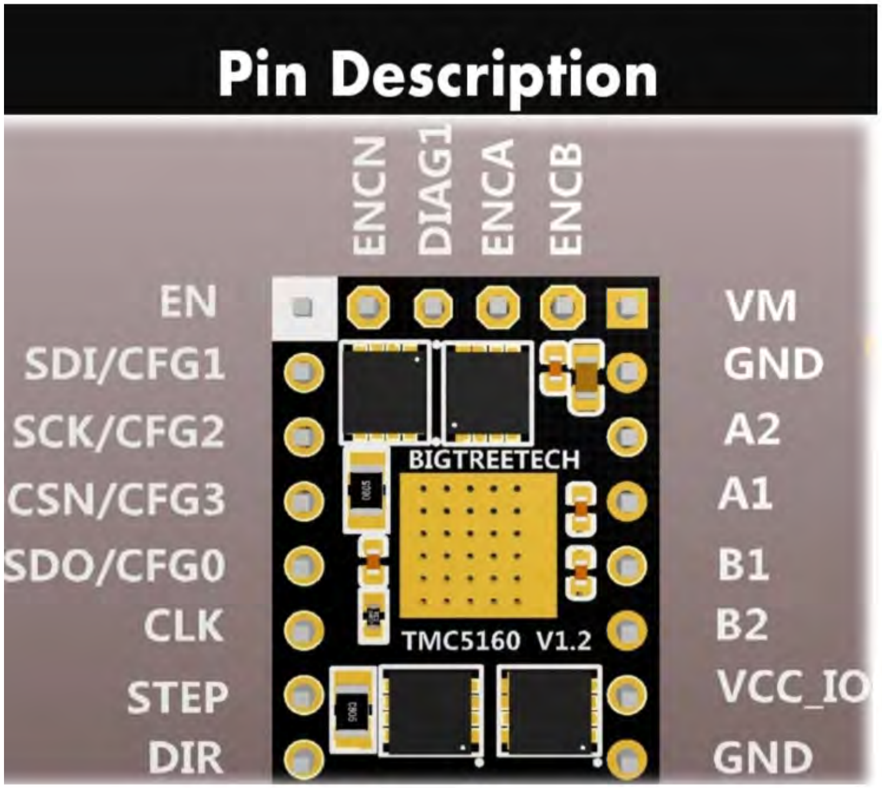


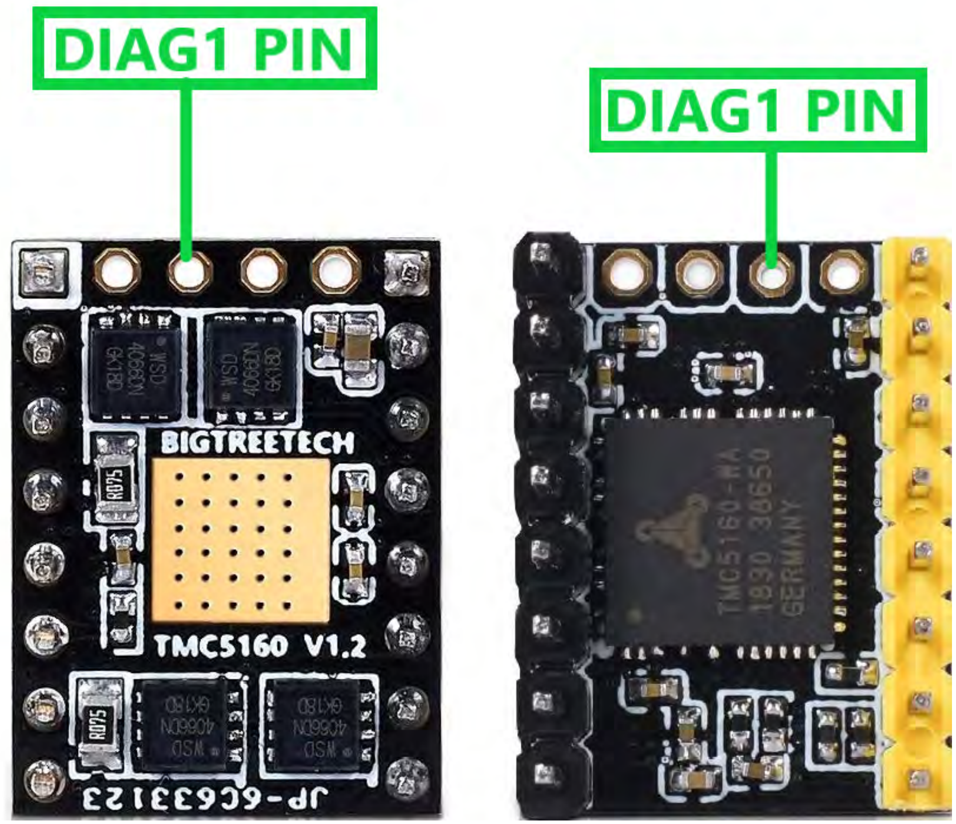


- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC5160 V1.2 Drivers in SPI Mode

- This next section covers sensor-less homing which is available for the TMC5160 in SPI mode. I want to enable it BUT for the TMC5160 I first have to solder on the DIAG1 pin onto each TMC5160 driver that will be on an axis with sensor-less homing enabled. Therefore, I want sensor-less homing for X and Y axes only. So I need to solder in a DIAG1 pin for two TMC5160 drivers. Here is a picture of the TMC5160 V1.2 pin-out.



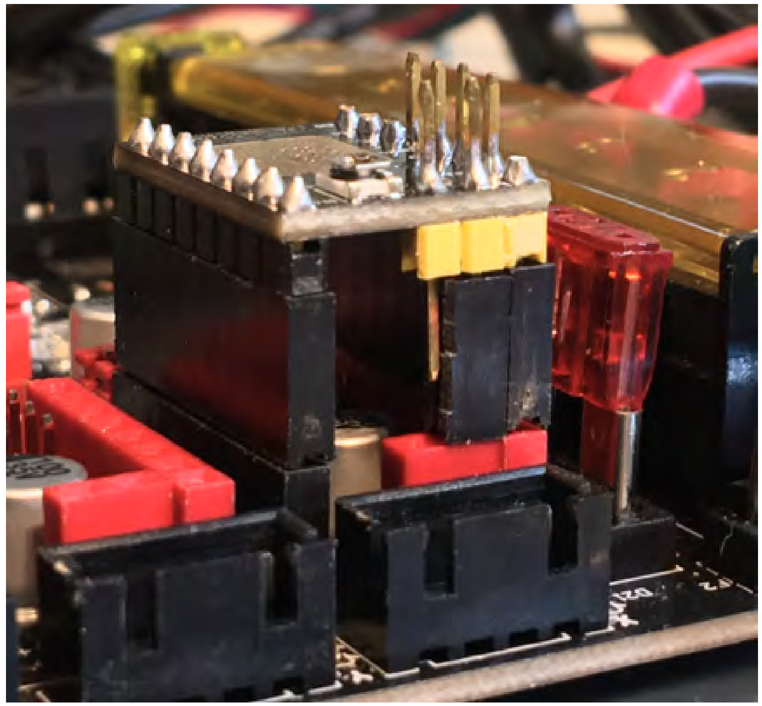


- The third pin position starting from the left on the top of the chip is where I need to solder in a header pin. I need it to face down so when I plug in the TMC5160 into the SKR PRO V1.1 board the DIAG1 pin will be seated in the SKR PRO V1.1 board.

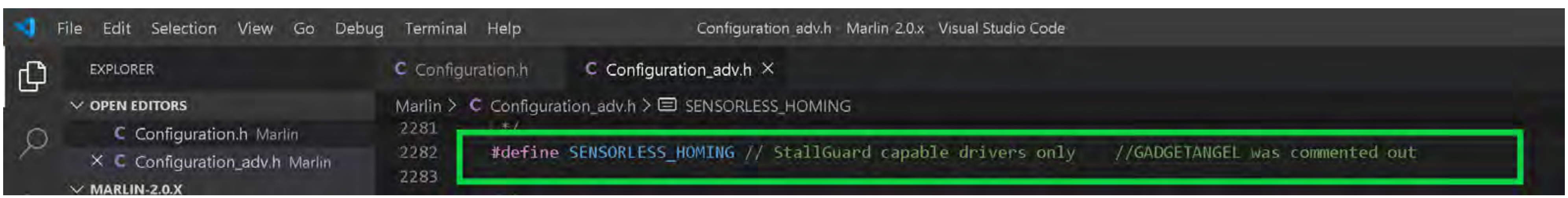
- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC5160 V1.2 Drivers in SPI Mode

- I will be covering sensor-less homing for the X and Y axis only. I will not be using sensor-less homing on my Z axis on my Ender 3 printer. For sensor-less homing to work the DIAG1 pin on the TMC5160 driver has to be plugged into the SKR PRO V1.1 board. Since I am not using sensor-less homing on my Z axis I will need to ensure that my DIAG1 pin on the Z axis TMC516 is NOT connected to the board. I plan to plug my Z axis' TMC5160 by using long stackable header pin risers, as seen in the picture below.



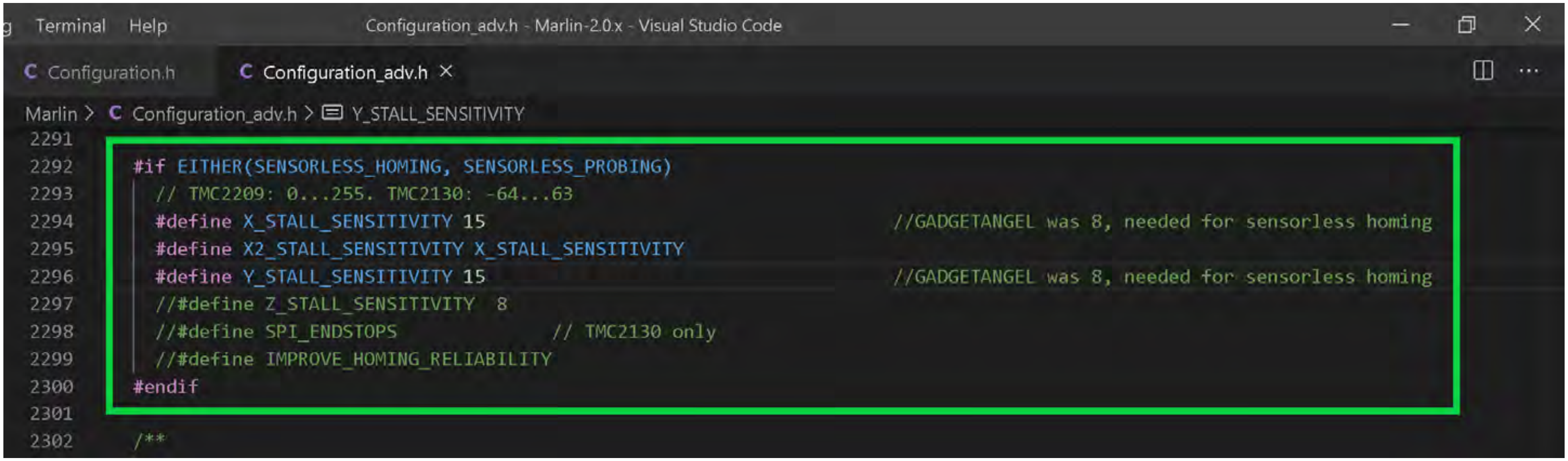
- Sensor-less homing is commented out by default. So I remove the two leading "//" to un-comment "SENSORLESS_HOMING"



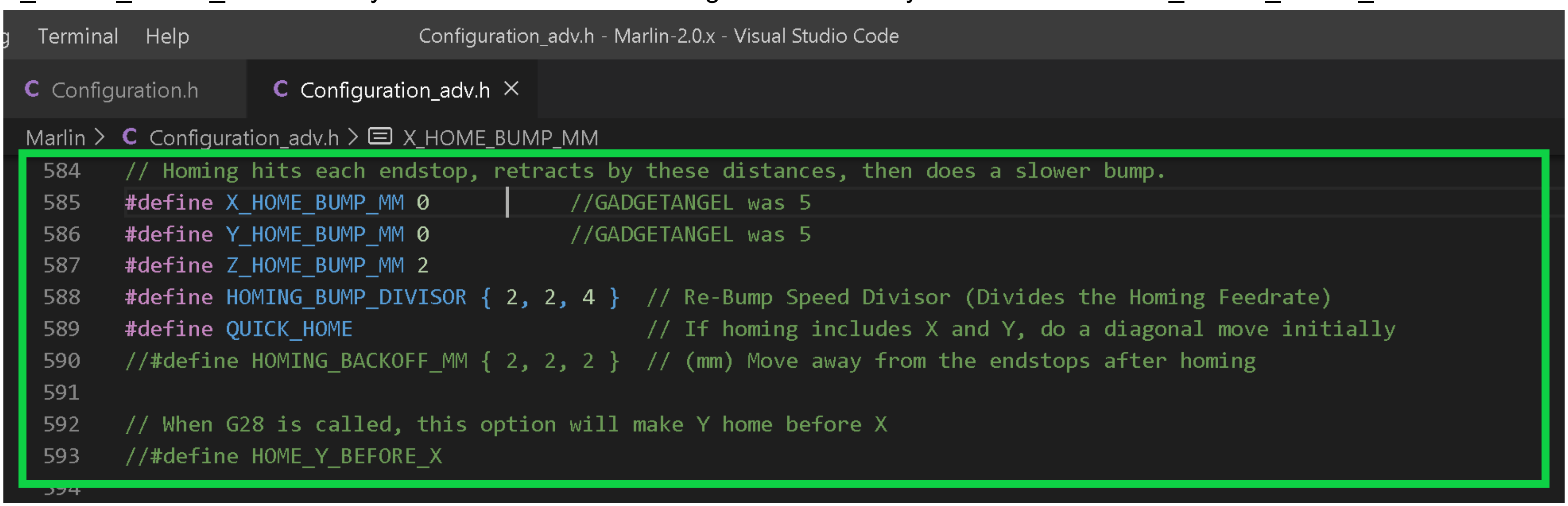
- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC5160 V1.2 Drivers in SPI Mode

- Next we set the "starting" stall sensitivity for sensor-less homing. I choose to make it 15. If the stall sensitivity is too high your motor will grind and not stop when it hits the end of travel on the axis. If the stall sensitivity is too low then the motor will barely move because it thinks it has hit the end of travel for the axis. Notice I only uncommented the "X_STALL_SENSITIVITY" and the "Y_STALL_SENSITIVITY". If you want sensor-less homing on the Z axis, then you will have to uncomment "Z_STALL_SENSITIVITY".



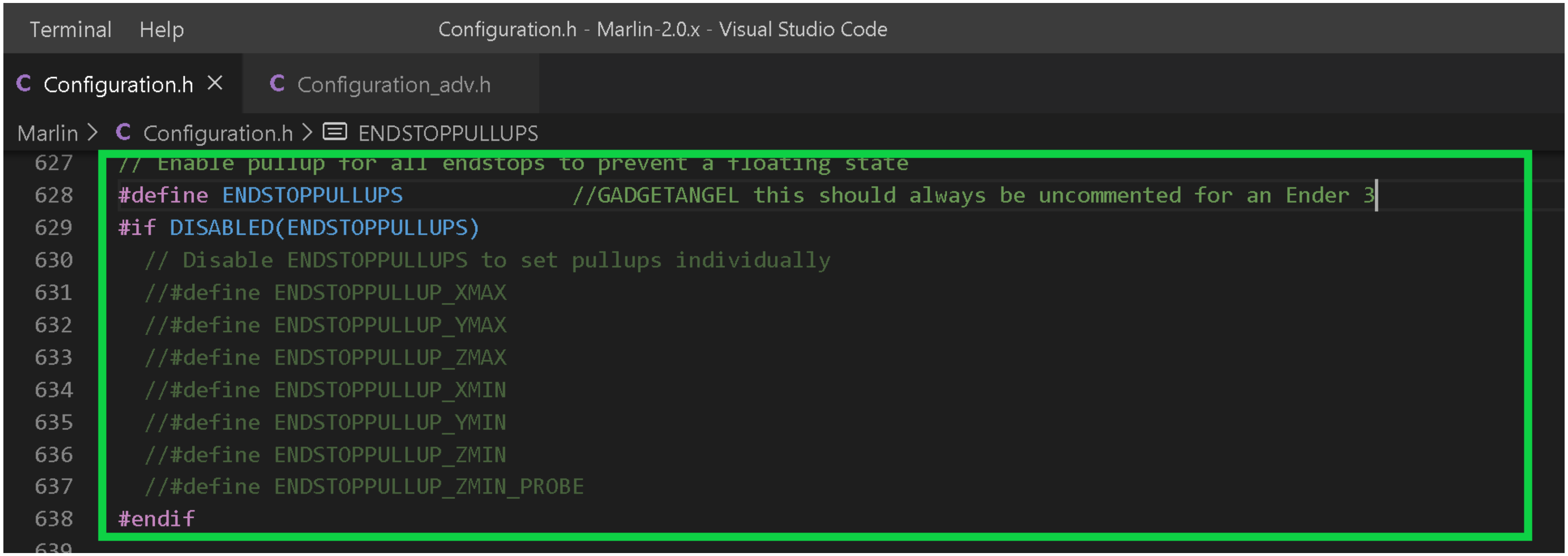
- We now have to set our home bump to 0 for each axis with sensor-less homing enabled. So I will set "X_HOME_BUMP_MM" to 0 and "Y_HOME_BUMP_MM" to 0. If you want sensor-less homing on Z axis then you will need to set "Z_HOME_BUMP_MM" to 0.



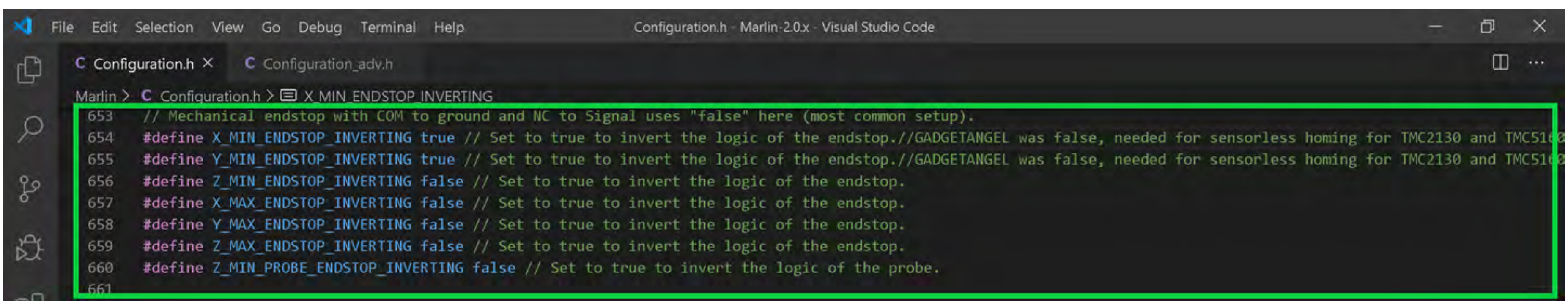
- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC5160 V1.2 Drivers in SPI Mode

- Let's check the firmware to ensure that "ENDSTOPPULLUPS" is enabled. It is by default.



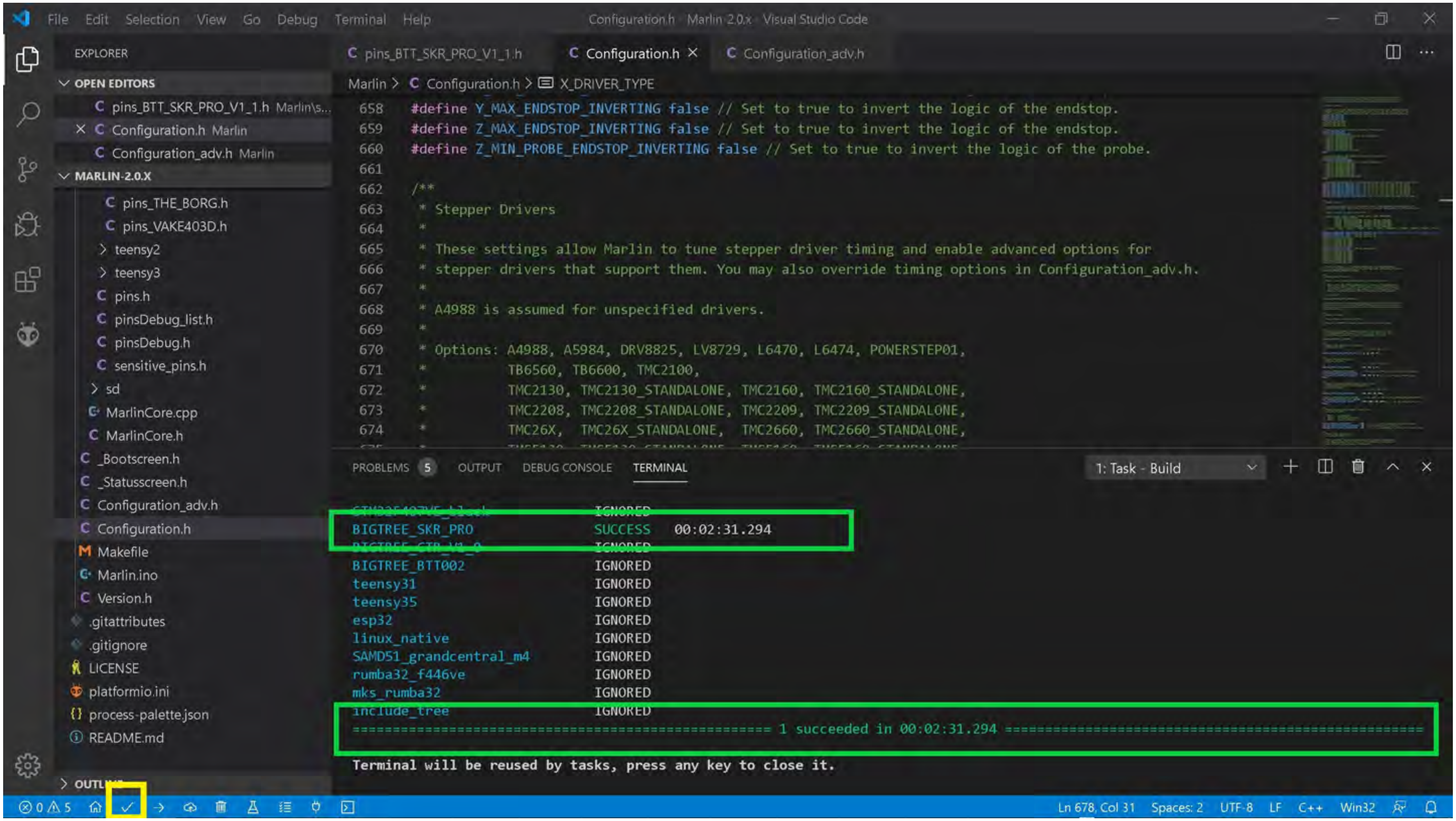
- Next to allow sensor-less homing to work (while using the BIQU TMC5160) we need to change our end stop logic. Therefore I set "X_MIN_ENDSTOP_INVERTING" to true and "Y_MIN_ENSTOP_INVERTING" to true. If you want sensor-less homing on the Z axis, you will need to set "Z_MIN_ENDSTOP_INVERTING" to true. But since I do not want sensor-less homing on the Z axis I will leave "Z_MIN_ENDSTOP_INVERTING" set to false.



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC5160 V1.2 Drivers in SPI Mode

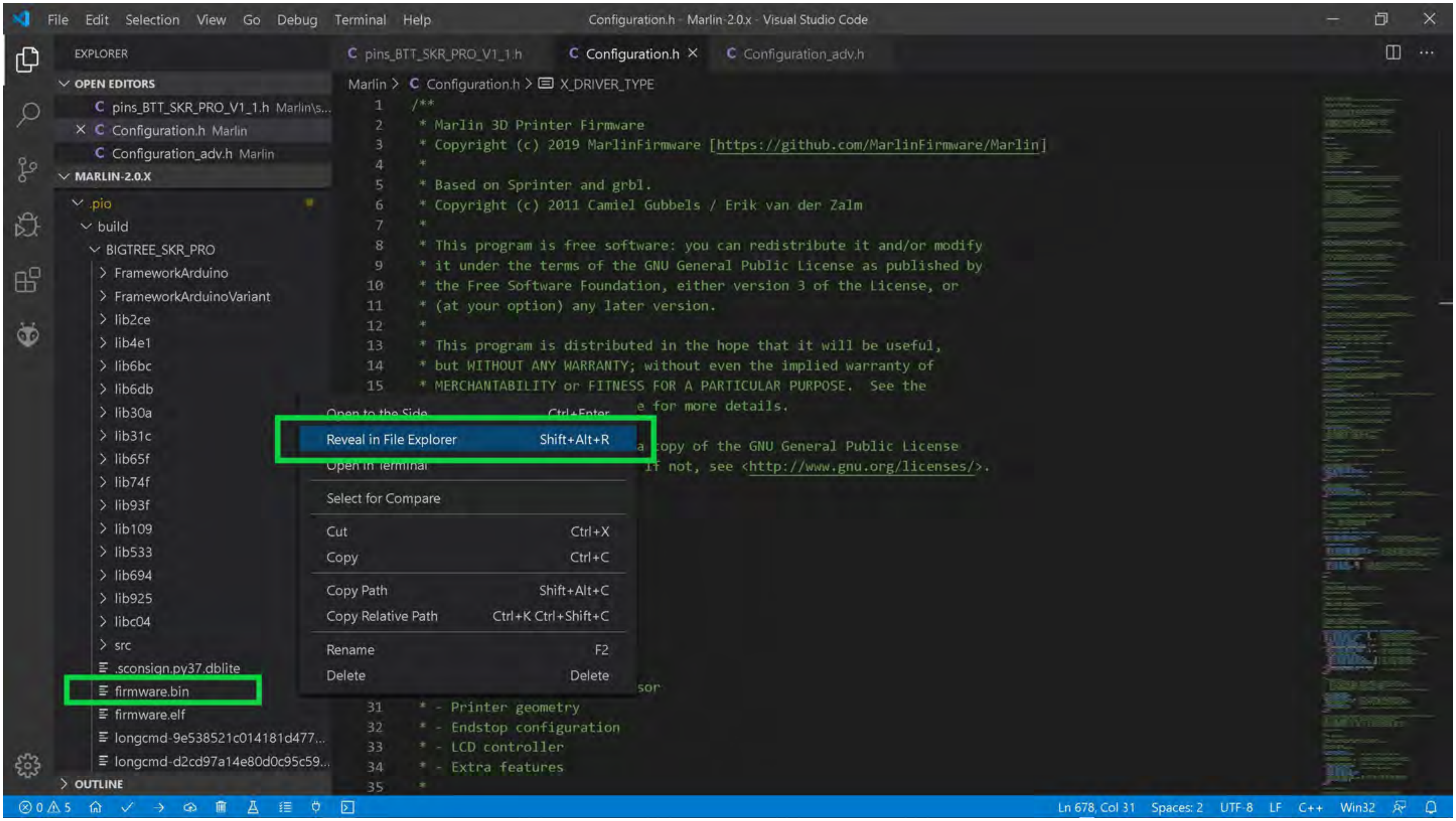
- The end of Marlin setup for BIQU TMC5160 drivers in SPI mode. Now, compile your code by clicking on the check mark, as seen in the **YELLOW** box below. Once the firmware has compiled successfully you will see the following messages, as shown in the **GREEN** boxes below.



- For the SKR PRO V1.1 board you must save the "firmware.bin" file to the micro SD card then place the micro SD card into the micro SD card reader of the SKR PRO V1.1 board and turn on the power to your printer for the NEW firmware to be automatically applied to the SKR PRO MCU. To locate the "firmware.bin" file, and learn how to transfer it to your micro SD card go to the next page.

The (latest release of) Marlin Setup for BIQU TMC5160 V1.2 Drivers in SPI Mode

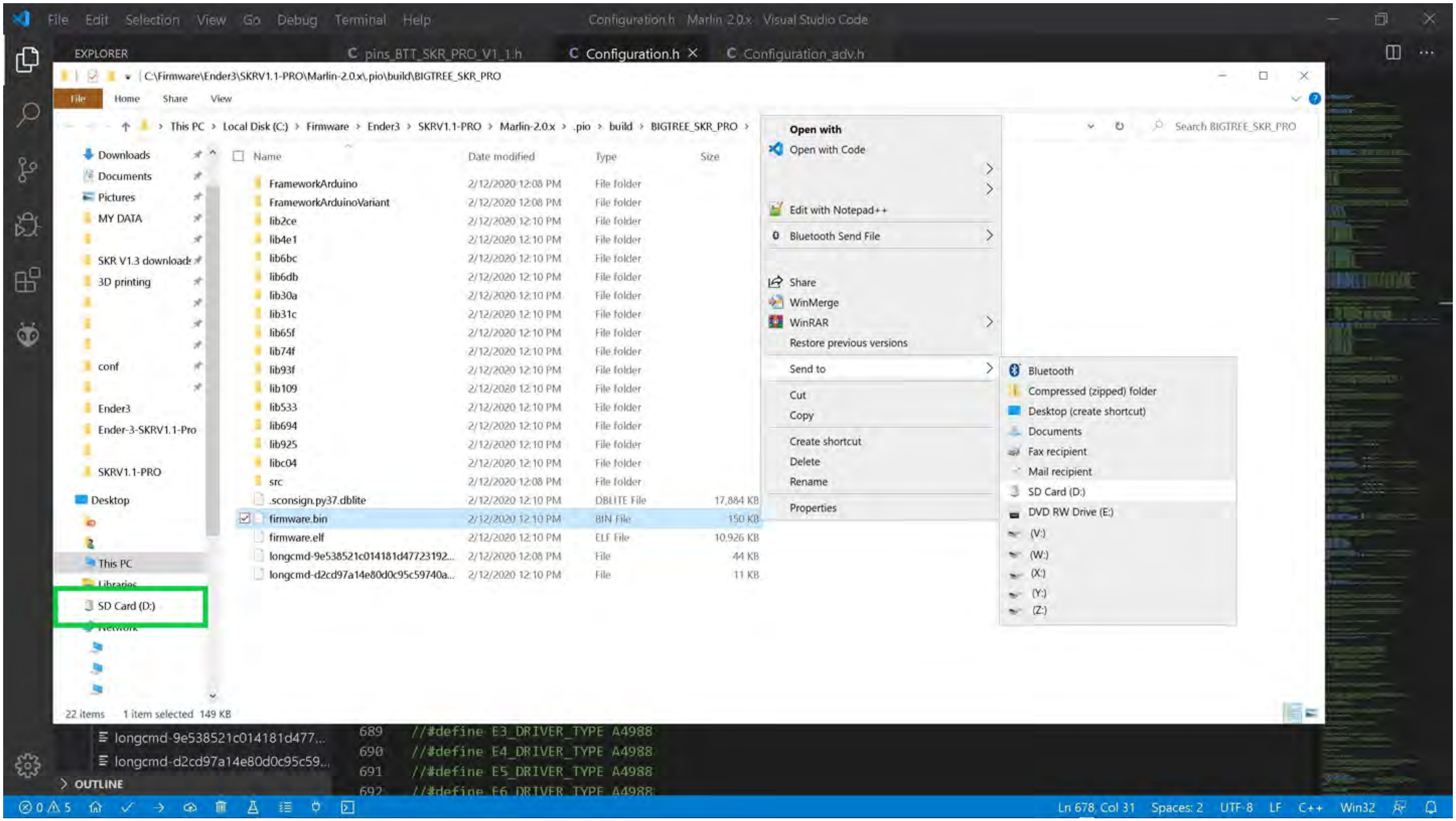
- The "firmware.bin" file can be found by using the left side of the VS code window and going to folder Marlin-2.0.X/.pio/build/ BIGTREE_SKR_PRO/ and right clicking on the "firmware.bin" file name to bring up the content menu. Select "Reveal in File Explorer" to have your Window's machine open a file explorer window.



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC5160 V1.2 Drivers in SPI Mode

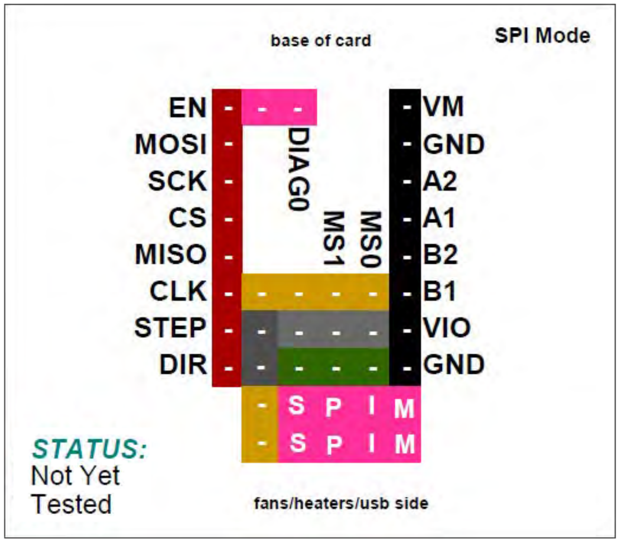
- Ensure your micro SD card, which came with your SKR PRO V1.1 board, with your SKR PRO V1.1 board is in a micro-to-SD-card adapter and plugged into your Window's SD card reader.
    - From the file explorer window right click on the "firmware.bin" file name and select "Send to" from the content menu. For my computer "D:" is my SD card reader. For your computer it may be a different drive letter. The "firmware.bin" file will be written to the SD card and renamed to "FIRMWARE.CUR".



- Right click on the SD card reader, as seen in the **GREEN** box above, and select "Eject" from the context menu. Take the micro SD card out of the adapter and place it into the micro SD card reader of the SKR PRO V1.1 board. Turn on the power to the printer (i.e. the SKR PRO V1.1 board) and the "FIRMWARE.CUR" file will be uploaded to the board. After the upload the firmware file will be renamed to "firmware.bin" and stored on the micro SD card.

BIQU TMC5161 V1.0

SPI Mode



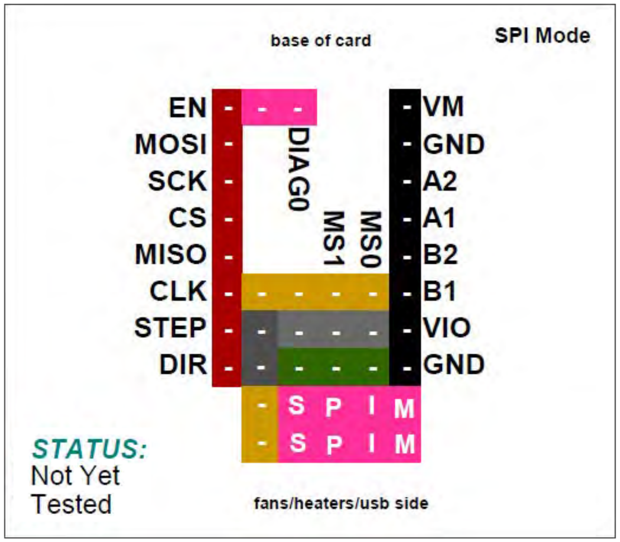
Note: You can use 50% to 90% of the calculated I_{RMS} (I_{MAX}/1.414) when tuning ("X_CURRENT", "Y_CURRENT", etc. the stepper motor driver in the firmware.

See the next page for further information.

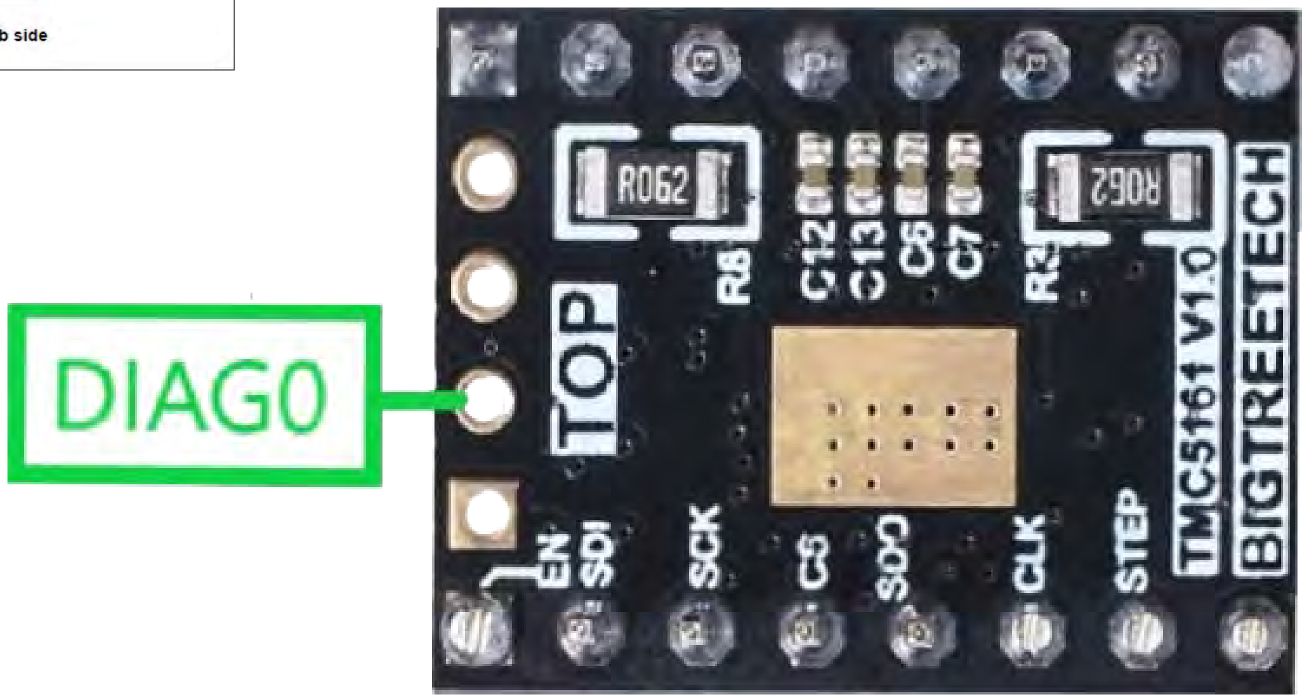
Driver Chip	Steps are set inside of your Firmware	
BIQU® TMC5161 SPI Mode Maximum 256 Subdivision 40V DC 5.2A (peak)		
Driving Current Calculation Formula R_S(Typical Sense Resistor)=0.062Ω	I_{MAX}=5.2419 See Appendix B #9. Use 50% to 90% as shown below: $I_{MAX}=I_{MAX}$ * 0.90=4.718	Current Limit is set by the current sense resistors (R_s). Use 50% - 90% of I_{MAX}.

BIQU TMC5161 V1.0

SPI Mode



Note: The TMC5161 V1.0 by default comes in SPI mode. The BIQU TMC5161 does NOT come with a POT or "V_{ref} Test point" location because the IRMS is set inside of the Firmware.



NOTE: BIQU TMC5161 has the ability to do sensor-less homing. By default the DIAG0 pin is NOT soldered onto the driver board. Therefore, for any axis you want sensor-less homing enabled, YOU WILL HAVE to solder on the DIAG0 pin.

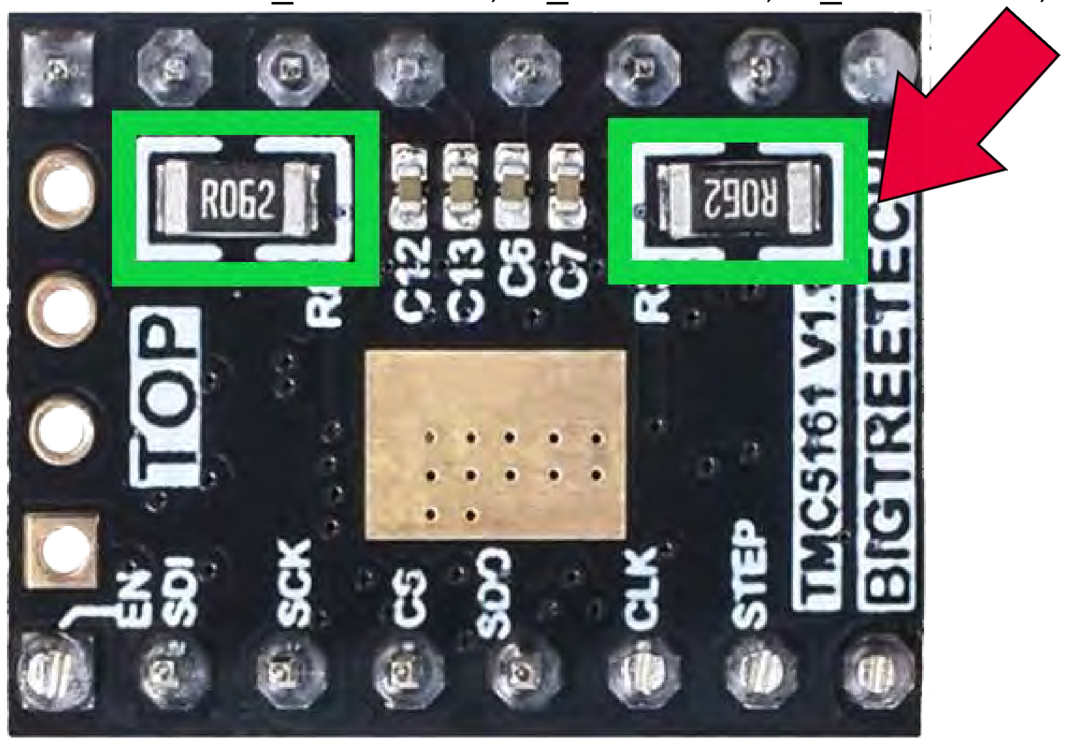
BIQU TMC5161 V1.0

SPI Mode

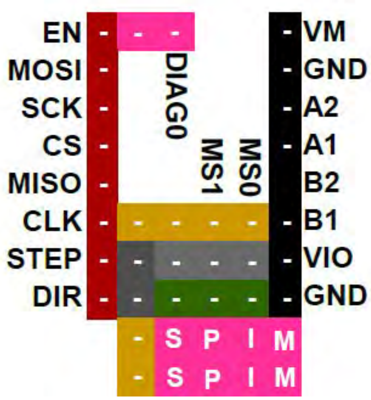
SPI Mode

Note: The location of the current sense resistors are shown in **GREEN**. Use the current sense resistors' value in the Marlin Firmware ("X_RSENSE", "Y_RSENSE", "Z_RSENSE" and/or "E0_RSENSE") so that the appropriate current limit can be sent to the driver board. If you do not want to use V_{ref} as the value for "X_CURRENT", "Y_CURRENT", "Z_CURRENT" and/or "E0_CURRENT", you should use I_{RMS} instead. You find I_{RMS} by taking I_{MAX} and dividing it by 1.414 ($I_{RMS}=I_{MAX}/1.414$). You use 50% to 90% of the calculated I_{RMS} as the value for "X_CURRENT", "Y_CURRENT", "Z_CURRENT", and/or "E0_CURRENT".

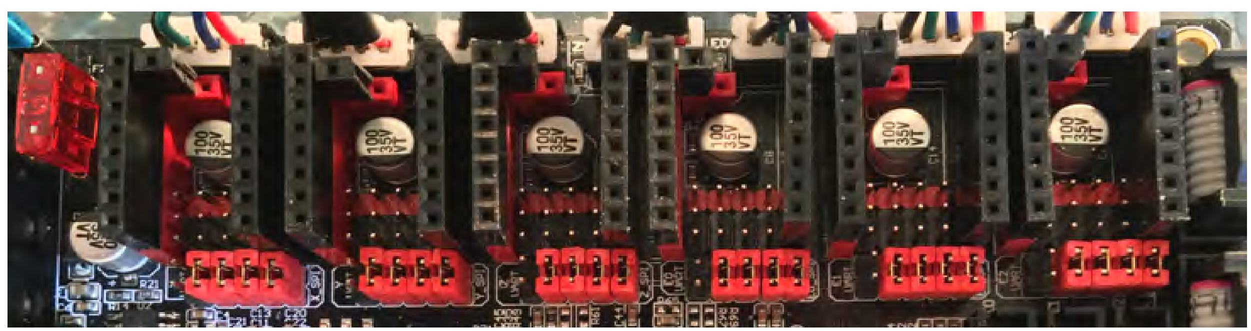
R_s = R050 is 0.05 Ohms
R_s = R062 is 0.062 Ohms
R_s = R068 is 0.068 Ohms
R_s = R075 is 0.075 Ohms
R_s = R100 is 0.1 Ohms
R_s = R110 is 0.11 Ohms
R_s = R150 is 0.15 Ohms
R_s = R200 is 0.2 Ohms
R_s = R220 is 0.22 Ohms



SPI



Note: Set JUMPERS "S","P","I","M" on the board!!

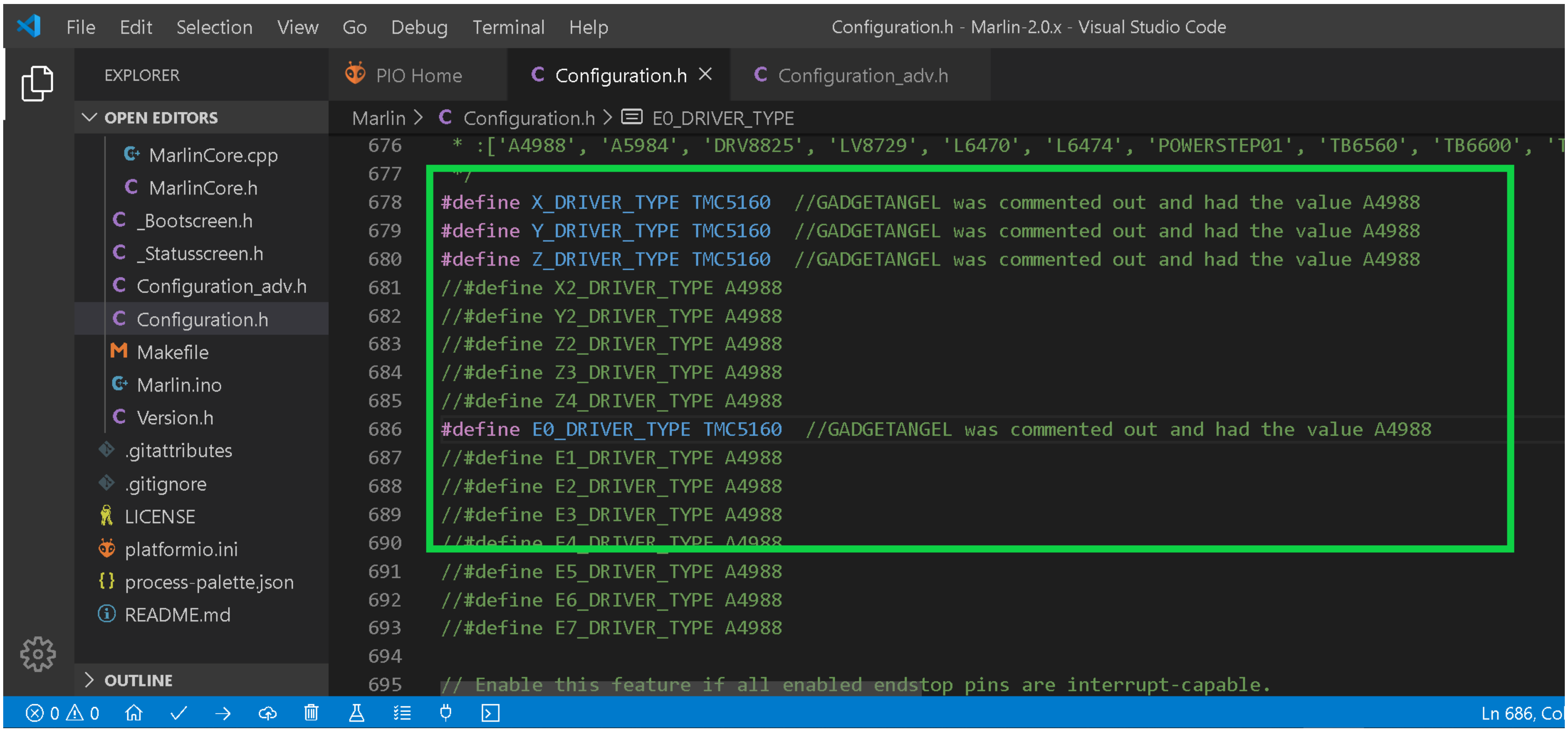


NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

The (latest release of) Marlin Setup for BIQU TMC5161 V1.0 Drivers in SPI Mode

NOTE: **Go to Appendix C**, and then come back here for the changes to Marlin for BIQU TMC5161 V1.0 stepper motor drivers in SPI mode.

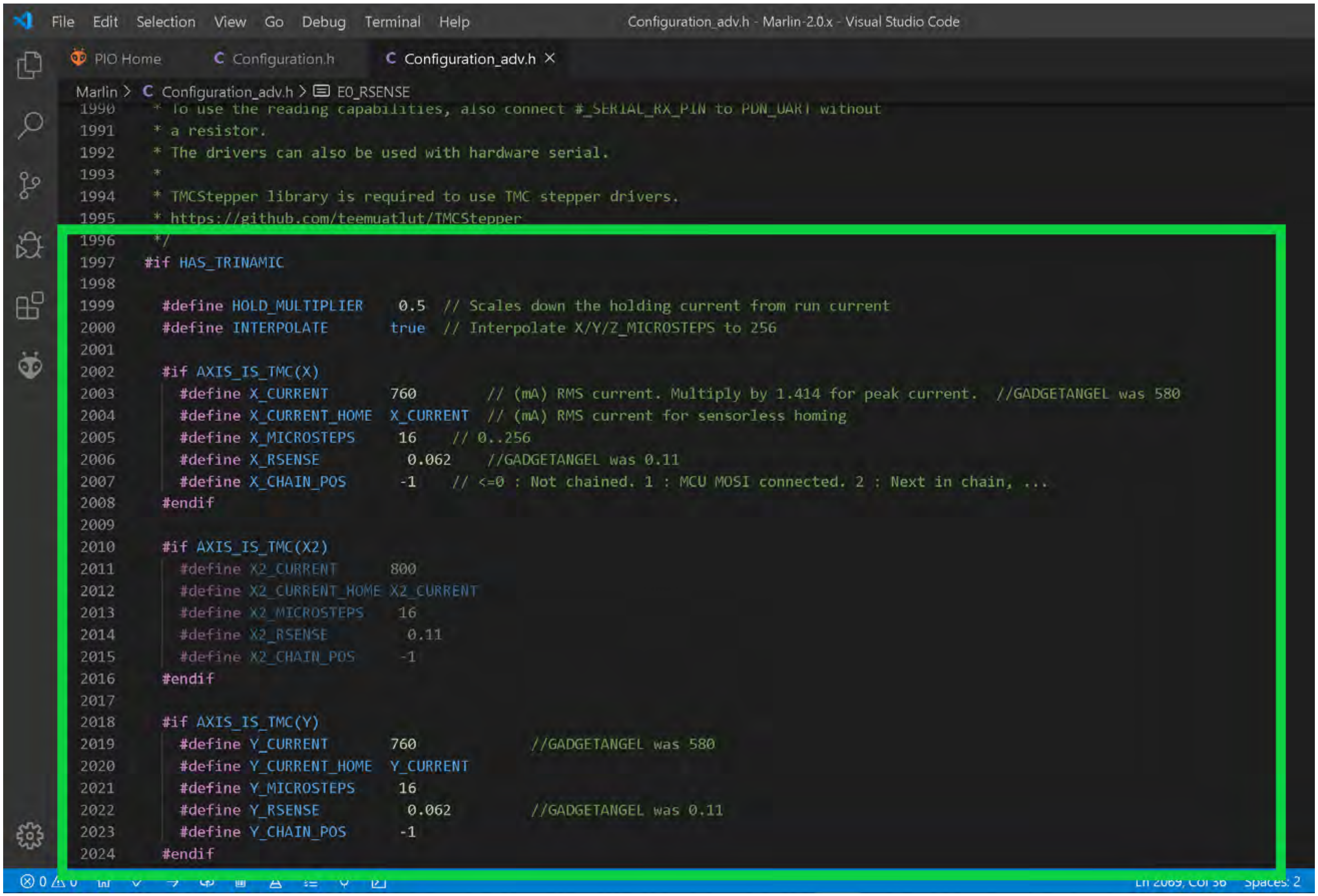
- Change the stepper motor drivers so that Marlin knows you are using TMC5161 drivers in SPI mode. Change one line for each axis and one for each extruder you will be using. See the picture below for an example of how to use TMC5161 drivers in SPI mode. When two "//" appear at the beginning of a line that means that line is commented out. To un-comment a line just remove the leading two forward slashes ("//").



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC5161 V1.0 Drivers in SPI Mode

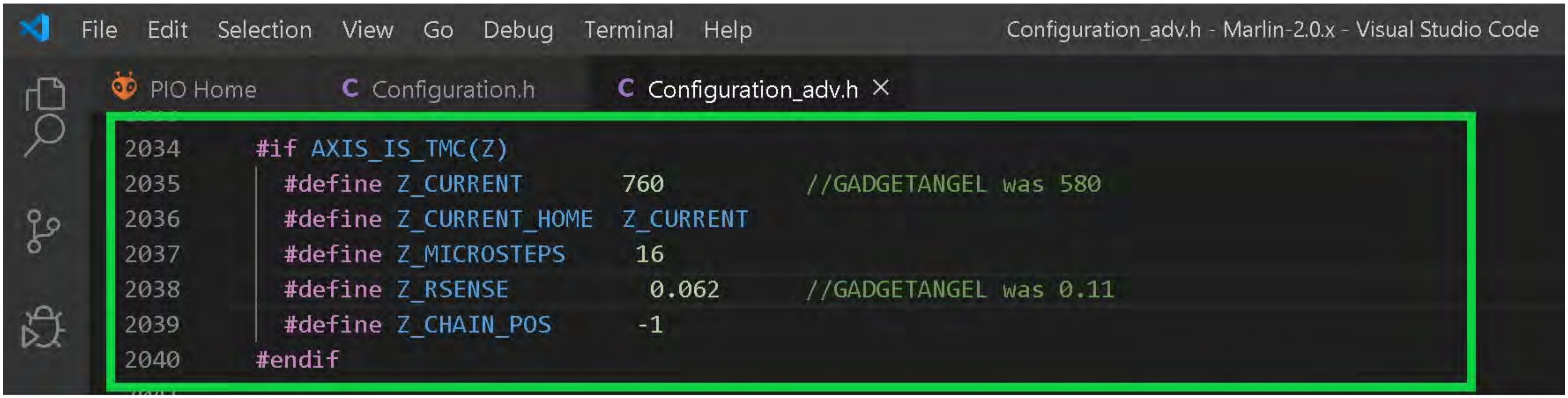
- Next you want to set your V_{ref} in the Marlin firmware for each axis that has the TMC5161 driver, as seen in the **GREEN** box below. I changed the "X_CURRENT" to be the calculated V_{ref} for my X-Axis, which is 760mV for an Ender 3. I changed the "Y_CURRENT" to be the calculated V_{ref} for my Y-Axis, which is 760mV on the Ender 3.
- Ensure "X_RSENSE" is set to 0.062. Ensure "Y_RSENSE" is set to 0.062.
- If you do not want to use V_{ref} as the value for "X_CURRENT" and/or "Y_CURRENT", you should use I_{RMS} instead. You find I_{RMS} by taking I_{MAX} and dividing it by 1.414 ($I_{RMS}=I_{MAX}/1.414$). You use 50% to 90% of the calculated I_{RMS} as the value for "X_CURRENT" and/or "Y_CURRENT".

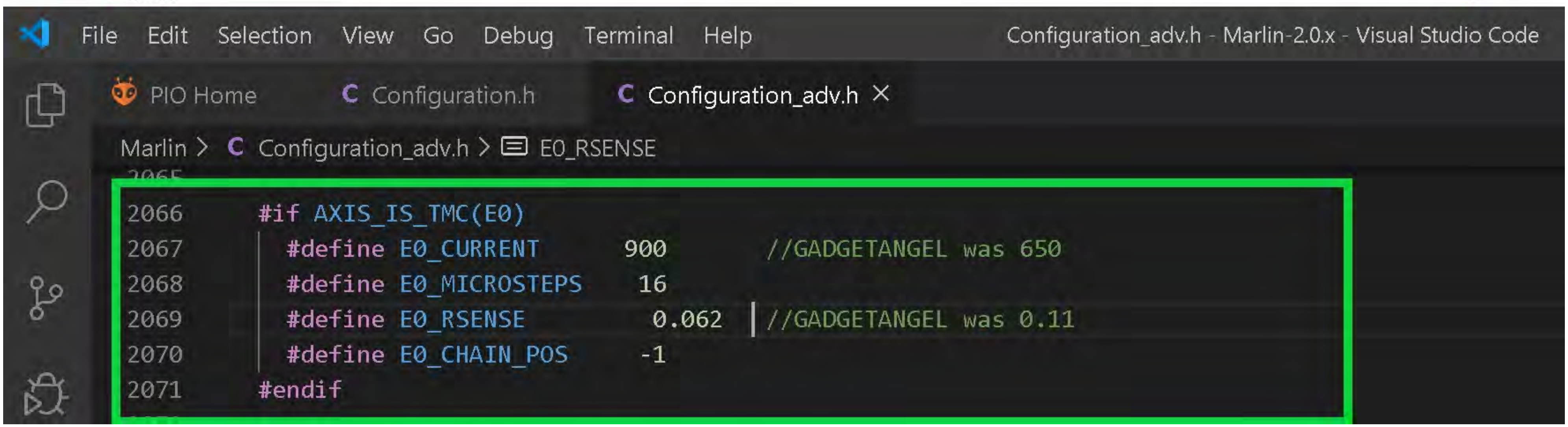


- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC5161 V1.0 Drivers in SPI Mode

- Now, I am setting the V_{ref} for Z-Axis and the extruder, as seen in the **GREEN** boxes below. I changed the "Z_CURRENT" to be the calculated V_{ref} for my Z-Axis, which is 760mV for an Ender 3. I changed the "E0_CURRENT" to be the calculated V_{ref} for my Extruder, which is 900mV on the Ender 3.
- Ensure "Z_RSENSE" is set to 0.062. Ensure "E0_RSENSE" is set to 0.062.
- If you do not want to use V_{ref} as the value for "Z_CURRENT" and/or "E0_CURRENT", you should use I_{RMS} instead. You find I_{RMS} by taking I_{MAX} and dividing it by 1.414 ($I_{RMS}=I_{MAX}/1.414$). You use 50% to 90% of the calculated I_{RMS} as the value for "Z_CURRENT" and/or "E0_CURRENT".

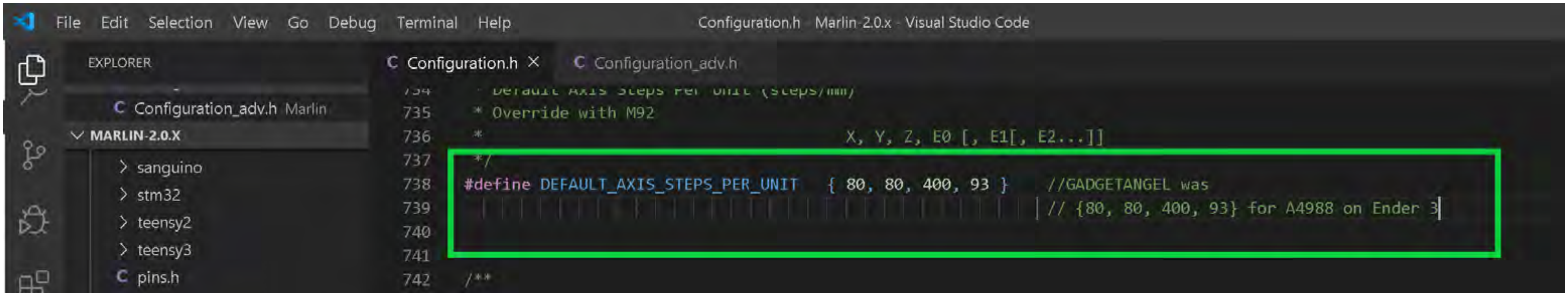




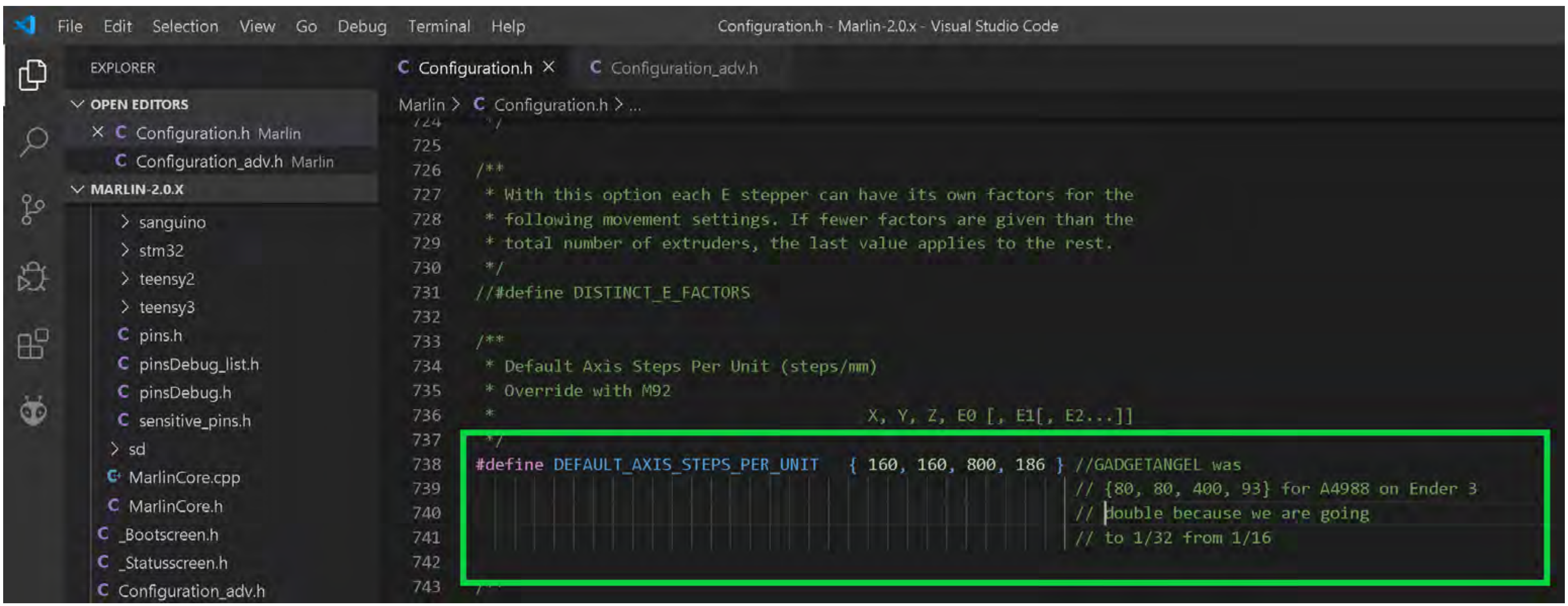
- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC5161 V1.0 Drivers in SPI Mode

- If you changed the "MICROSTEPS" for any of the axes then you will need to update "DEFAULT_AXIS_STEPS_PER_UNIT" to reflect your changes



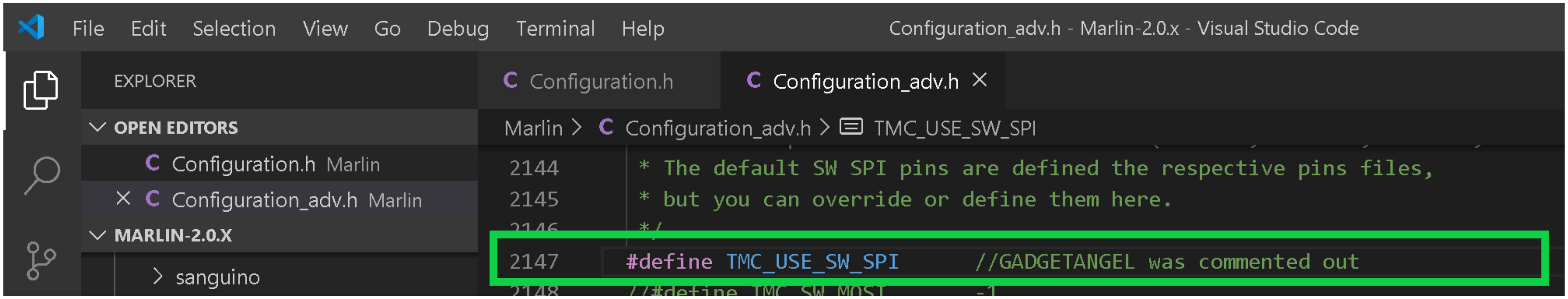
- FOR EAMPLE if you wanted to use 1/32 stepping instead of the default 1/16, you would be **doubling** your STEPS. Therefore, we must adjust our "DEFAULT_AXIS_STEPS_PER_UNIT" anytime our STEPS are NOT 1/16. So change "DEFAULT_AXIS_STEPS_PER_UNIT" to {160, 160, 800, 186}, as seen in the **GREEN** box below.



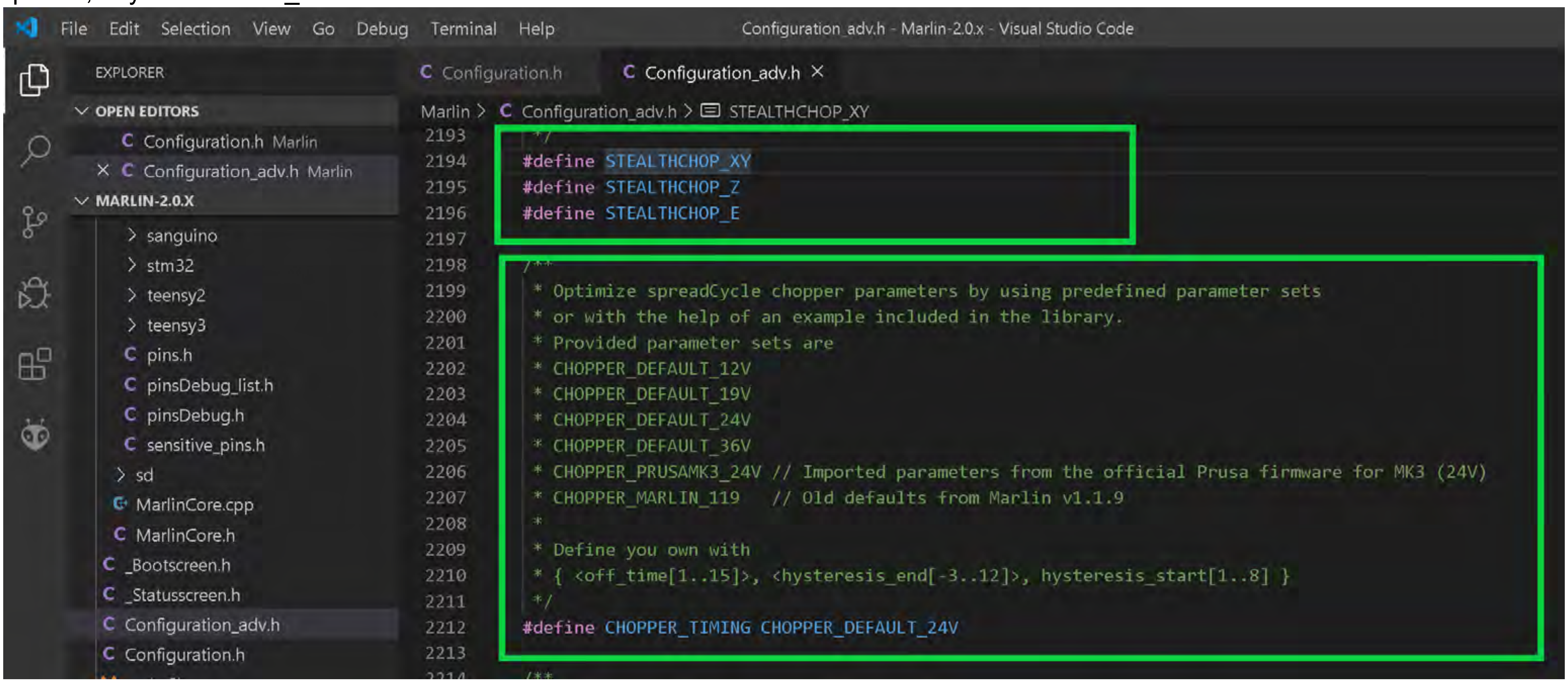
- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC5161 V1.0 Drivers in SPI Mode

- We need to uncomment out the "TMC_USE_SW_SPI" because the SKR PRO V1.1 pins file depends on this variable to define its SPI pins



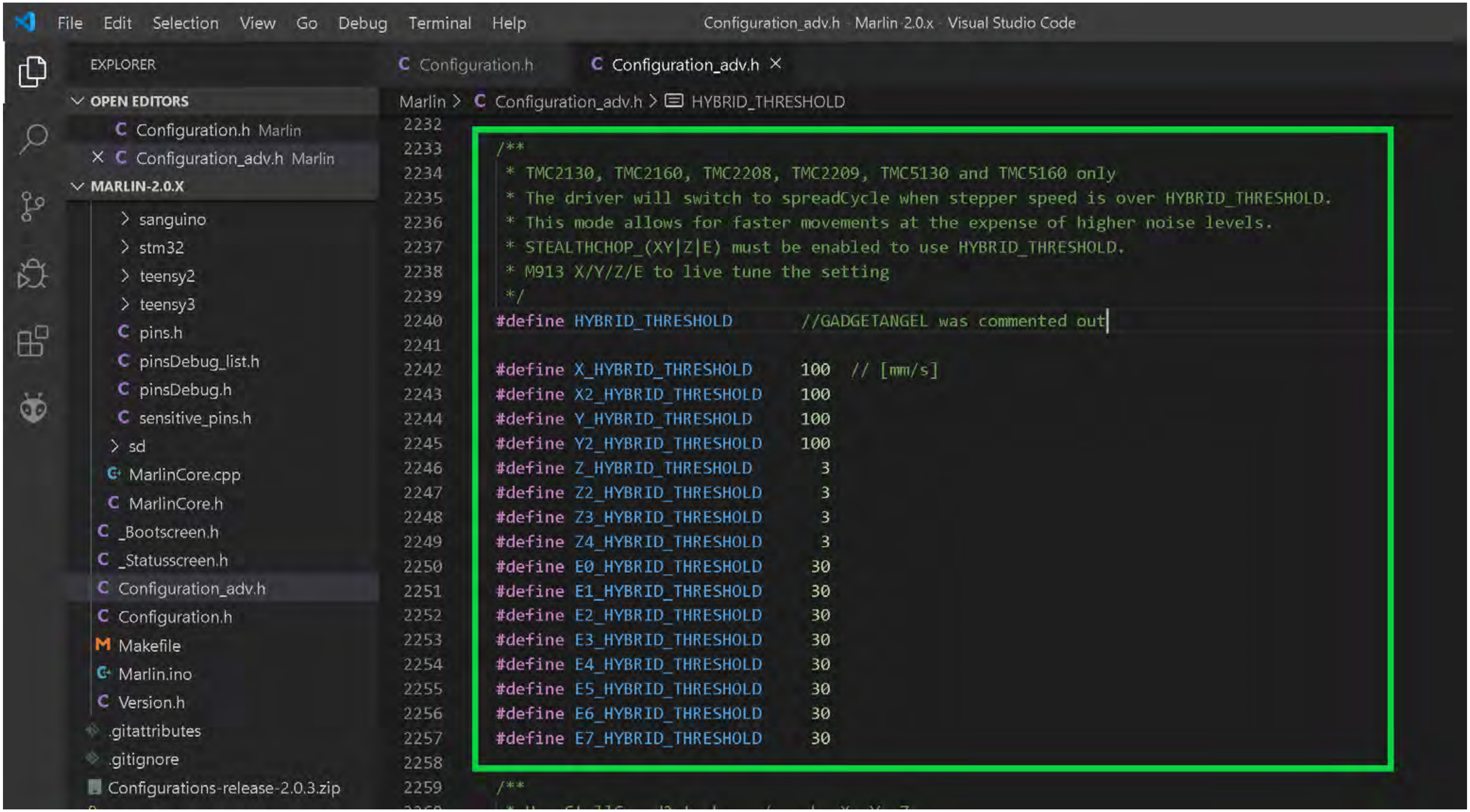
- By default stealthChop is enabled in the Marlin firmware. If you want spreadCycle ONLY then comment out the appropriate lines. I want stealthChop enabled so I want to make sure the lines are not commented out {"STEALTHCHOP_XY", "STEALTHCHOP_Z" and "STEALTHCHOP_E"}. You also want to check to see if the proper "CHOPPER_TIMING" is set for your printer. An Ender 3 is a 24VDC printer, my "CHOPPER_TIMING" is correct.



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC5161 V1.0 Drivers in SPI Mode

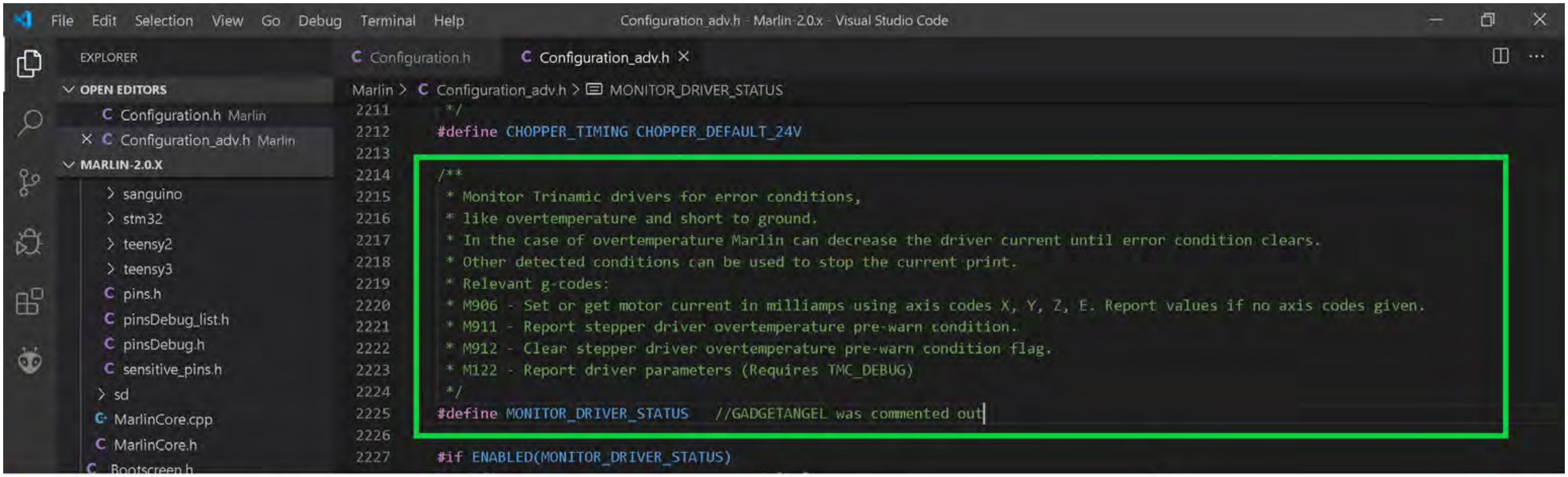
- Now you either enable "HYBRID_THRESHOLD" or disable it. By default, it is disabled. "HYBRID_THERSHOLD" allows the printer to change between stealthChop and spreadCycle dynamically depending on the print speed. I want "HYBRID_THRESHOLD" enabled so I need to remove the two leading "//", which uncomments the line in the Marlin firmware.

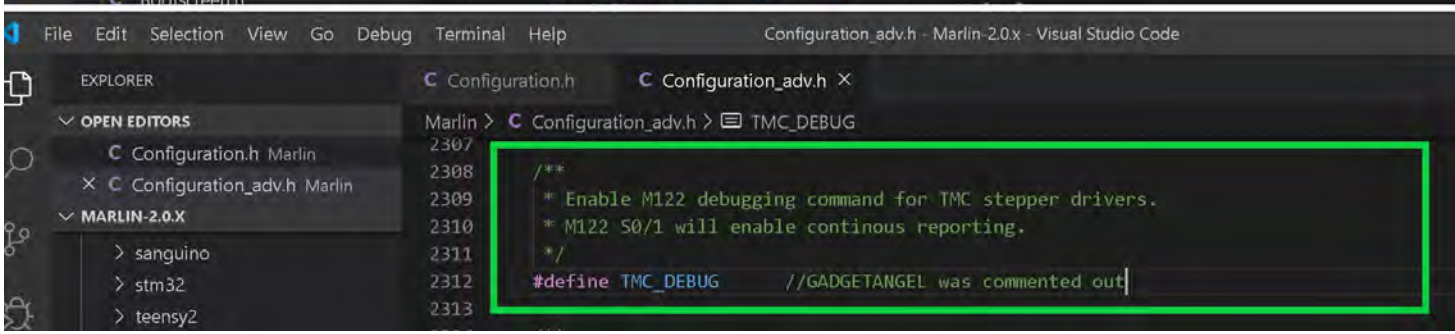


- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC5161 V1.0 Drivers in SPI Mode

- Now I want to enable some statements that allow me access to debugging the TMC drivers. I will uncomment "MONITOR_DRIVER_STATUS" and "TMC_DEBUG". "MONITOR_DRIVER_STATUS" will enable the following G-codes: M906, M911, and M912, "TMC_DEBUG" will enable the M122 G-code command. You can read about these from the comments in the firmware and in **Marlin's documentation located on-line.**

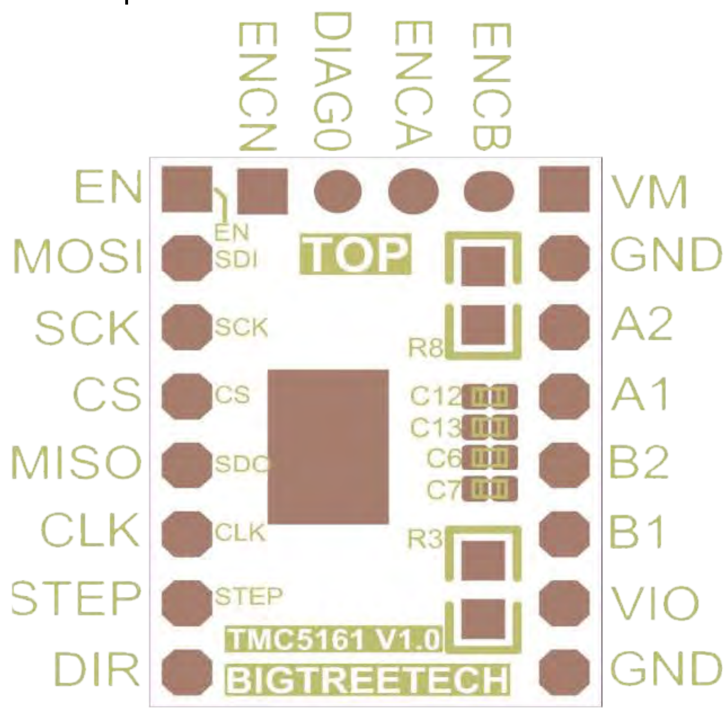


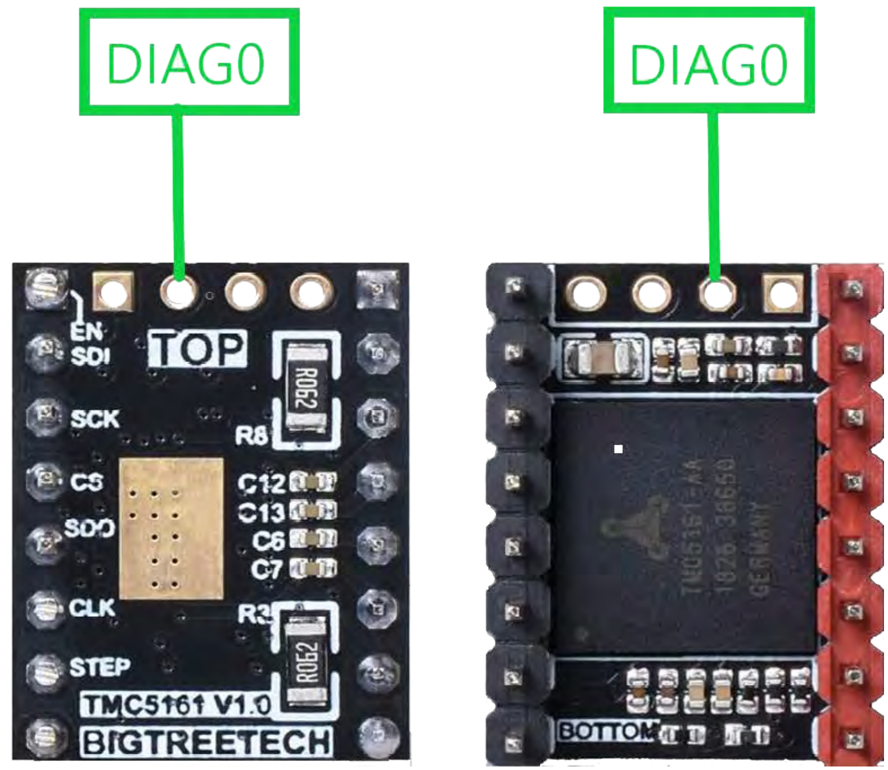


- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC5161 V1.0 Drivers in SPI Mode

- This next section covers sensor-less homing which is available for the TMC5161 in SPI mode. I want to enable it BUT for the TMC5161 I first have to solder on the DIAG0 pin onto each TMC5161 driver that will be on an axis with sensor-less homing enabled. Therefore, I want sensor-less homing for X and Y axes only. So I need to solder in a DIAG0 pin for two TMC5161 drivers. Here is a picture of the TMC5161 V1.0 pin-out.



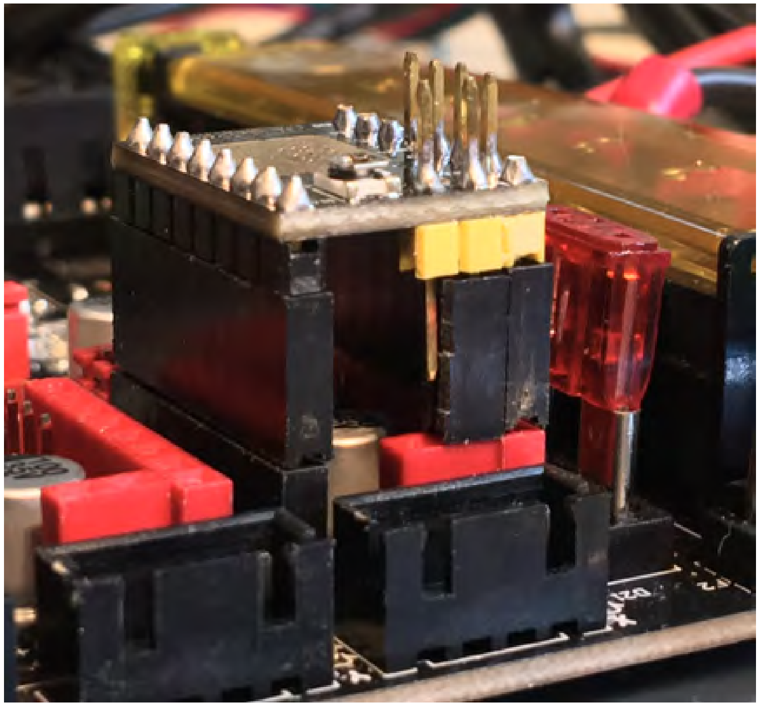


- The third pin position starting from the left on the top of the chip is where I need to solder in a header pin. I need it to face down so when I plug in the TMC5161 into the SKR PRO V1.1 board the DIAG0 pin will be seated in the SKR PRO V1.1 board.

- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC5161 V1.0 Drivers in SPI Mode

- I will be covering sensor-less homing for the X and Y axis only. I will not be using sensor-less homing on my Z axis on my Ender 3 printer. For sensor-less homing to work the DIAG0 pin on the TMC5161 driver has to be plugged into the SKR PRO V1.1 board. Since I am not using sensor-less homing on my Z axis I will need to ensure that my DIAG0 pin on the Z axis TMC5161 is NOT connected to the board. I plan to plug my Z axis' TMC5161 by using long stackable header pin risers, as seen in the picture below.

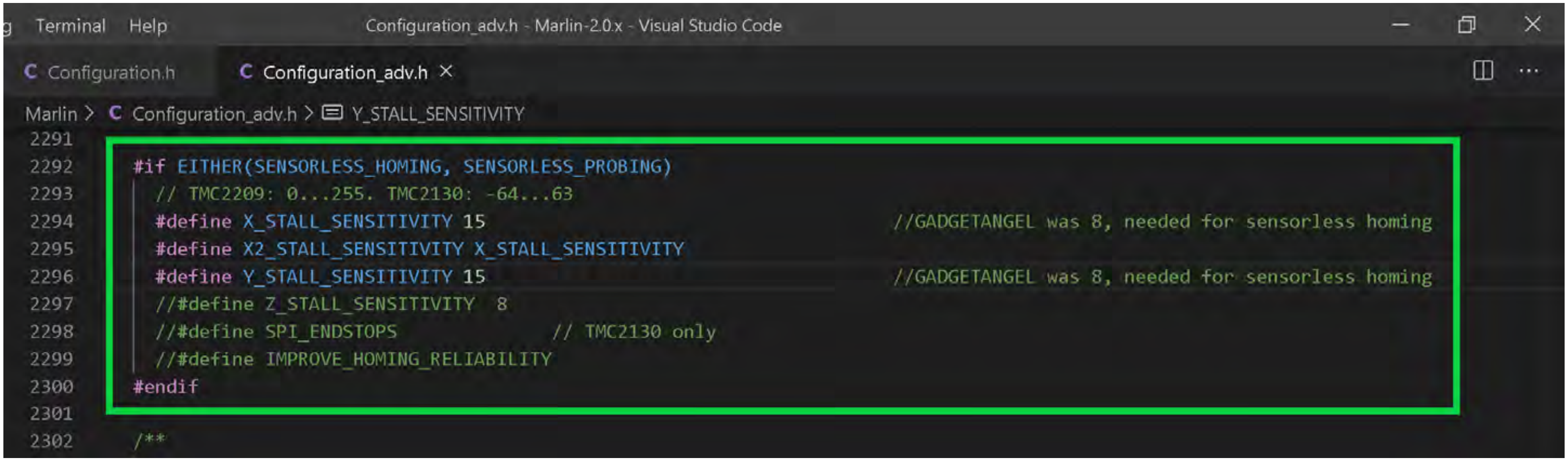


- Sensor-less homing is commented out by default. So I remove the two leading "//" to un-comment "SENSORLESS_HOMING"

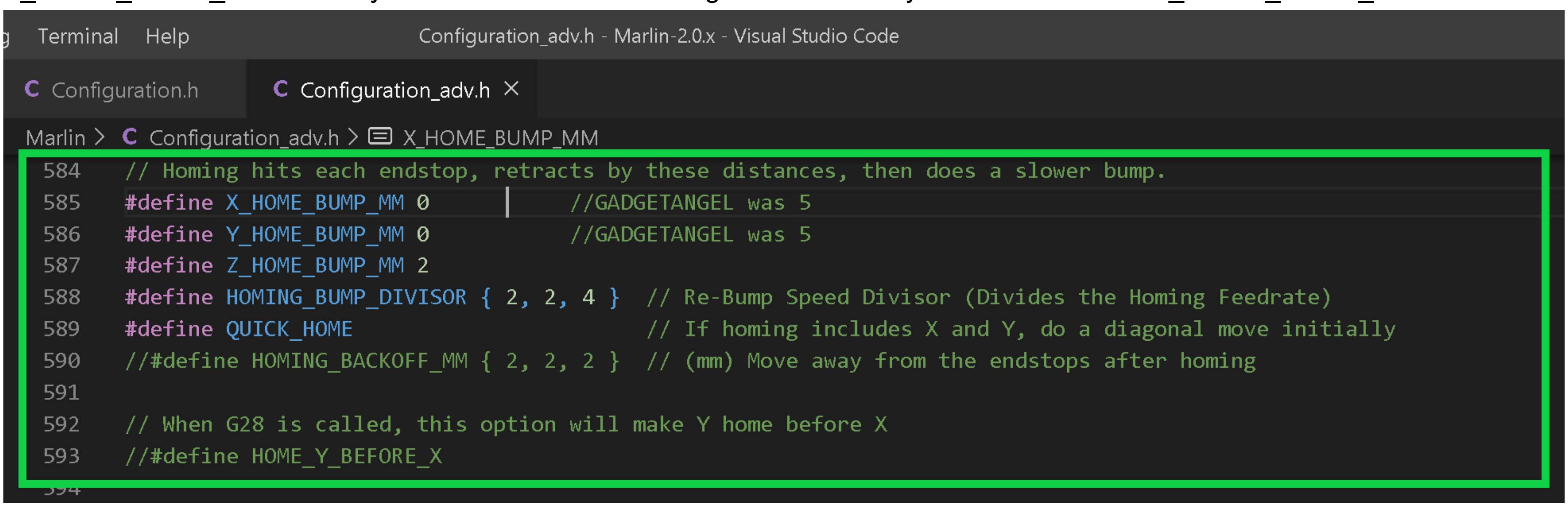
- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC5161 V1.0 Drivers in SPI Mode

- Next we set the "starting" stall sensitivity for sensor-less homing. I choose to make it 15. If the stall sensitivity is too high your motor will grind and not stop when it hits the end of travel on the axis. If the stall sensitivity is too low then the motor will barely move because it thinks it has hit the end of travel for the axis. Notice I only uncommented the "X_STALL_SENSITIVITY" and the "Y_STALL_SENSITIVITY". If you want sensor-less homing on the Z axis, then you will have to uncomment "Z_STALL_SENSITIVITY".



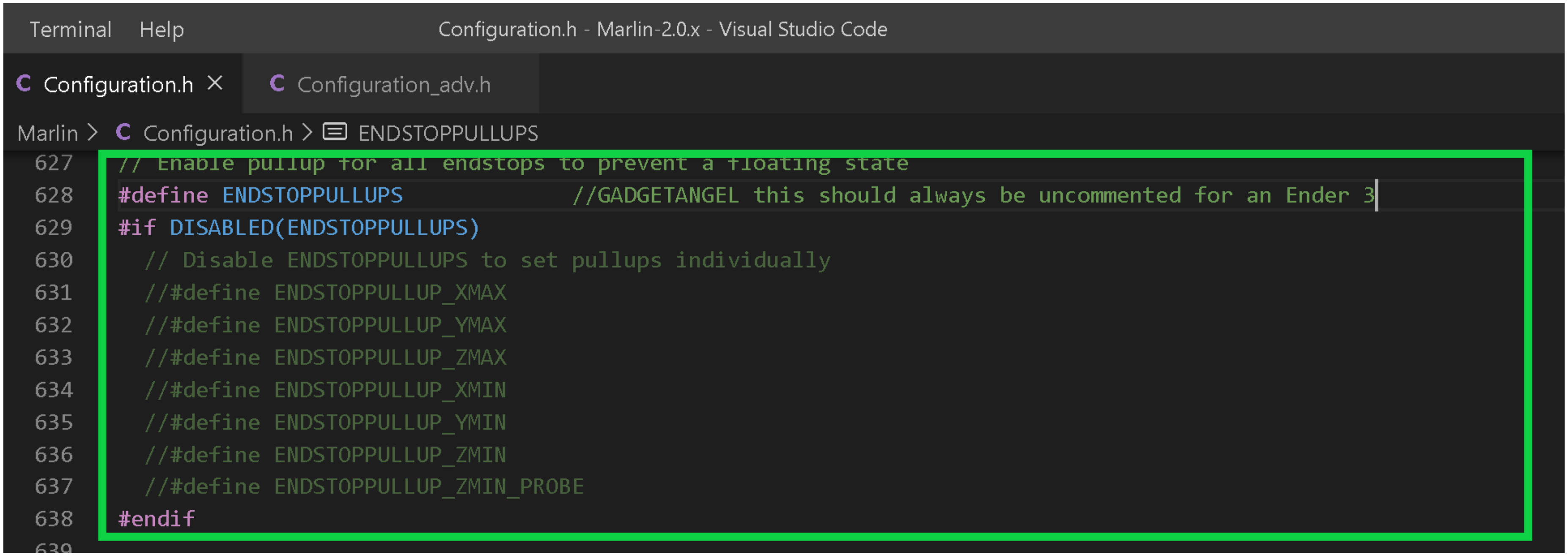
- We now have to set our home bump to 0 for each axis with sensor-less homing enabled. So I will set "X_HOME_BUMP_MM" to 0 and "Y_HOME_BUMP_MM" to 0. If you want sensor-less homing on Z axis then you will need to set "Z_HOME_BUMP_MM" to 0.



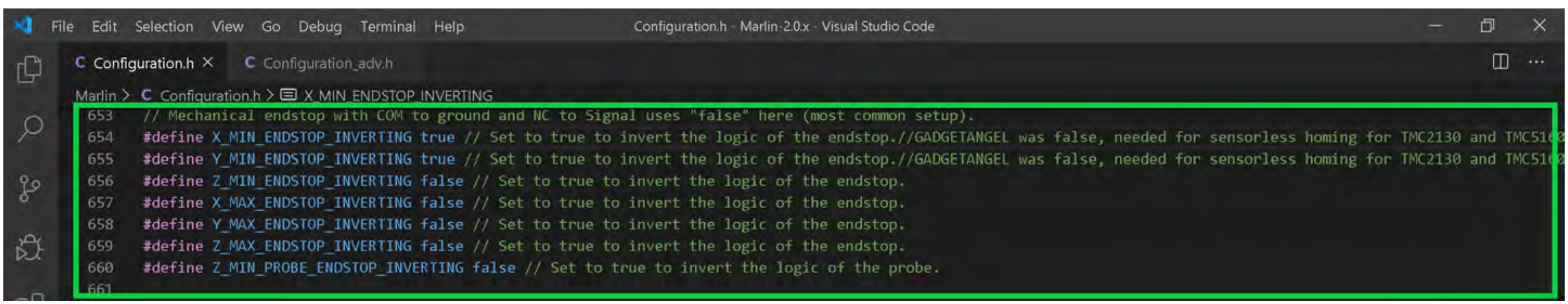
- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC5161 V1.0 Drivers in SPI Mode

- Let's check the firmware to ensure that "ENDSTOPPULLUPS" is enabled. It is by default.



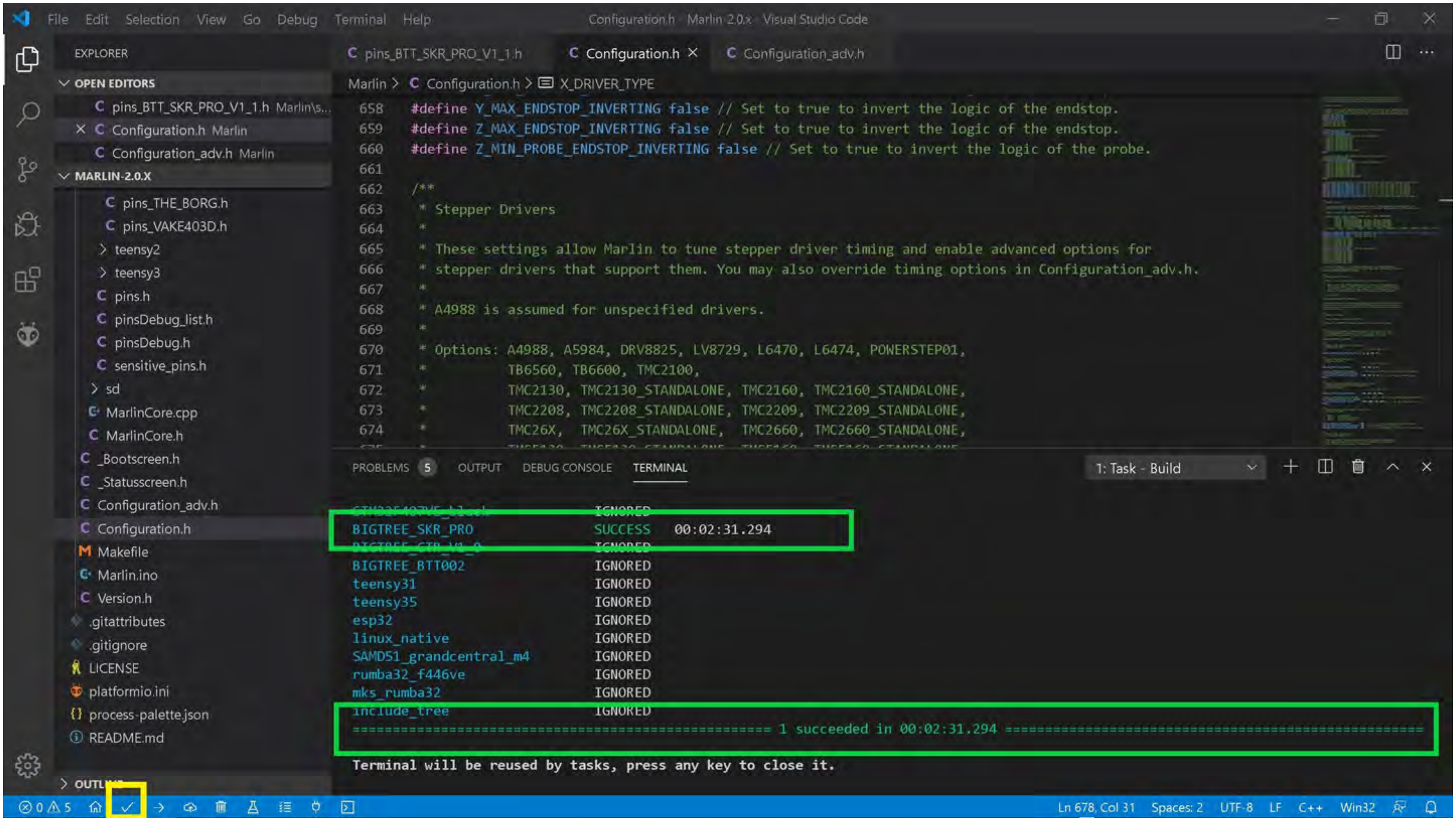
- Next to allow sensor-less homing to work (while using the BIQU TMC5161) we need to change our end stop logic. Therefore I set "X_MIN_ENDSTOP_INVERTING" to true and "Y_MIN_ENSTOP_INVERTING" to true. If you want sensor-less homing on the Z axis, you will need to set "Z_MIN_ENDSTOP_INVERTING" to true. But since I do not want sensor-less homing on the Z axis I will leave "Z_MIN_ENDSTOP_INVERTING" set to false.



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC5161 V1.0 Drivers in SPI Mode

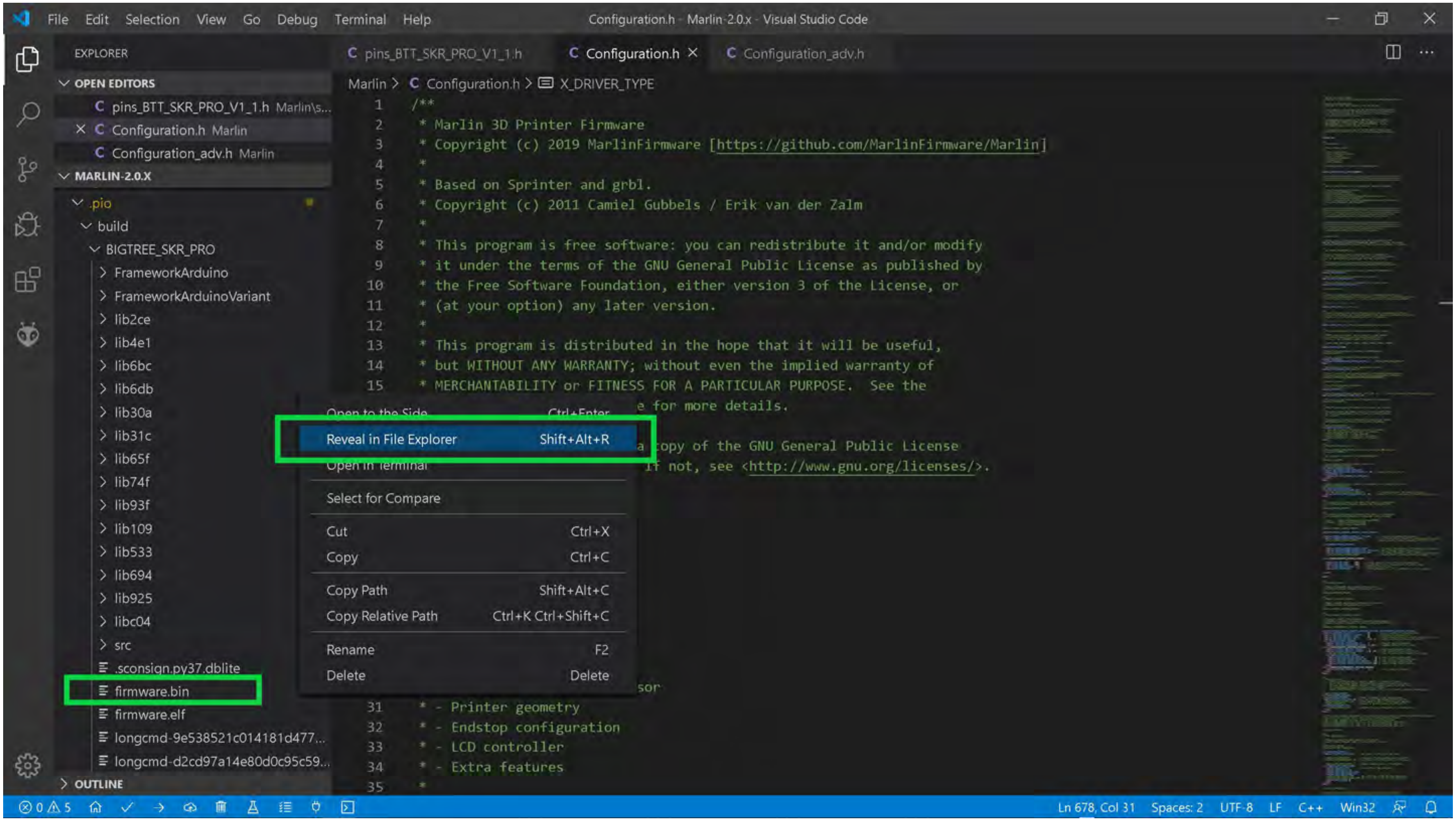
- The end of Marlin setup for BIQU TMC5161 drivers in SPI mode. Now, compile your code by clicking on the check mark, as seen in the **YELLOW** box below. Once the firmware has compiled successfully you will see the following messages, as shown in the **GREEN** boxes below.



- For the SKR PRO V1.1 board you must save the "firmware.bin" file to the micro SD card then place the micro SD card into the micro SD card reader of the SKR PRO V1.1 board and turn on the power to your printer for the NEW firmware to be automatically applied to the SKR PRO MCU. To locate the "firmware.bin" file, and learn how to transfer it to your micro SD card go to the next page.

The (latest release of) Marlin Setup for BIQU TMC5161 V1.0 Drivers in SPI Mode

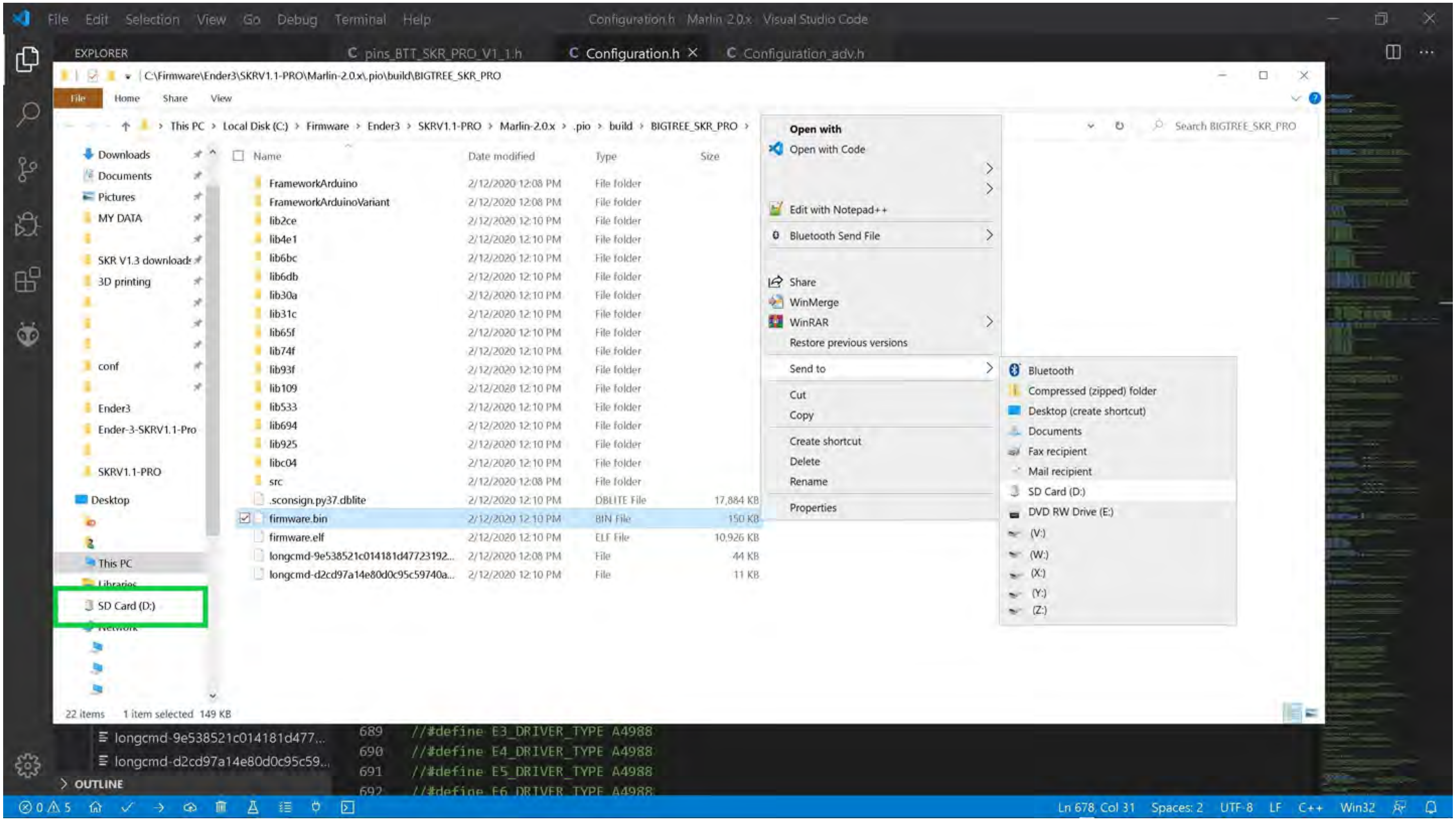
- The "firmware.bin" file can be found by using the left side of the VS code window and going to folder Marlin-2.0.X/.pio/build/BIGTREE_SKR_PRO/ and right clicking on the "firmware.bin" file name to bring up the content menu. Select "Reveal in File Explorer" to have your Window's machine open a file explorer window.



- Go to the next page.

The (latest release of) Marlin Setup for BIQU TMC5161 V1.0 Drivers in SPI Mode

- Ensure your micro SD card, which came with your SKR PRO V1.1 board, with your SKR PRO V1.1 board is in a micro-to-SD-card adapter and plugged into your Window's SD card reader.
    - From the file explorer window right click on the "firmware.bin" file name and select "Send to" from the content menu. For my computer "D:" is my SD card reader. For your computer it may be a different drive letter. The "firmware.bin" file will be written to the SD card and renamed to "FIRMWARE.CUR".



- Right click on the SD card reader, as seen in the **GREEN** box above, and select "Eject" from the context menu. Take the micro SD card out of the adapter and place it into the micro SD card reader of the SKR PRO V1.1 board. Turn on the power to the printer (i.e. the SKR PRO V1.1 board) and the "FIRMWARE.CUR" file will be uploaded to the board. After the upload the firmware file will be renamed to "firmware.bin" and stored on the micro SD card.

How to adjust the V_{ref} on a Stepper Motor Driver board using the Potentiometer[1, 2]

Modern 3D printers usually use NEMA17 motors. The first piece of information you will need is the "Rated Current" of your NEMA17 motor. So, write down your motor's part number and pay attention to the Step Angle, Holding Torque, Rated Current, Voltage, and Inductance.

Use the Step Angle to work out your printer's "DEFAULT_AXIS_STEPS_PER_UNIT" for your firmware with:

1. **http://www.prusaprinters.org/calculator/**

How to Tune Stepper Motor Drivers[2]

1. Turn power off your printer, unplug the stepper motor cables, turn power back on your printer and tune the stepper motor drivers that are already plugged into the SKR PRO V1.1 board.
2. When done, turn power off the printer, plug in the stepper motor cables turn power back on your printer and test motor movement.

NOTE: Don't tune stepper motor drivers with the motors plugged in, if you accidentally set current too high you can fry the motor or the stepper motor driver.

NOTE: Don't plug or unplug stepper motors with the power on the printer (i.e. power on the SKR PRO V1.1 board)

Measure DC voltage between the stepper motor driver's trimpot (POT) or "V_{ref} Test point" and your PSU's (12VDC/24VDC) ground. The ground at the PSU connector to the SKR PRO board is fine to use. Look up the correct current for your motor part number. If you have motors with no part number, assume they have a max of 1.00 amps (I_{MAX}) to be safe. Look up the proper formula for your stepper motor drivers (as show in this document), and find the voltage (but ONLY use 90% of the calculated V_{ref}) which corresponds with the current you want to set. Use a ceramic screw driver to adjust the POT. A ceramic screw driver is nonconductive and if you slip while making the adjustment to the POT you could short circuit the stepper motor driver board (i.e. KILL the driver board) in the process.

Time Saver tip, but more dangerous: Get slip-on alligator clips for your multimeter. Clamp ground to a 12VDC/24VDC ground (PSU Ground) wire and clamp positive to your plastic handle screwdriver. This way you'll measure the voltage as you adjust ("live adjustment") and don't need three hands.

Note: **See the next page for a diagram of the setup.**

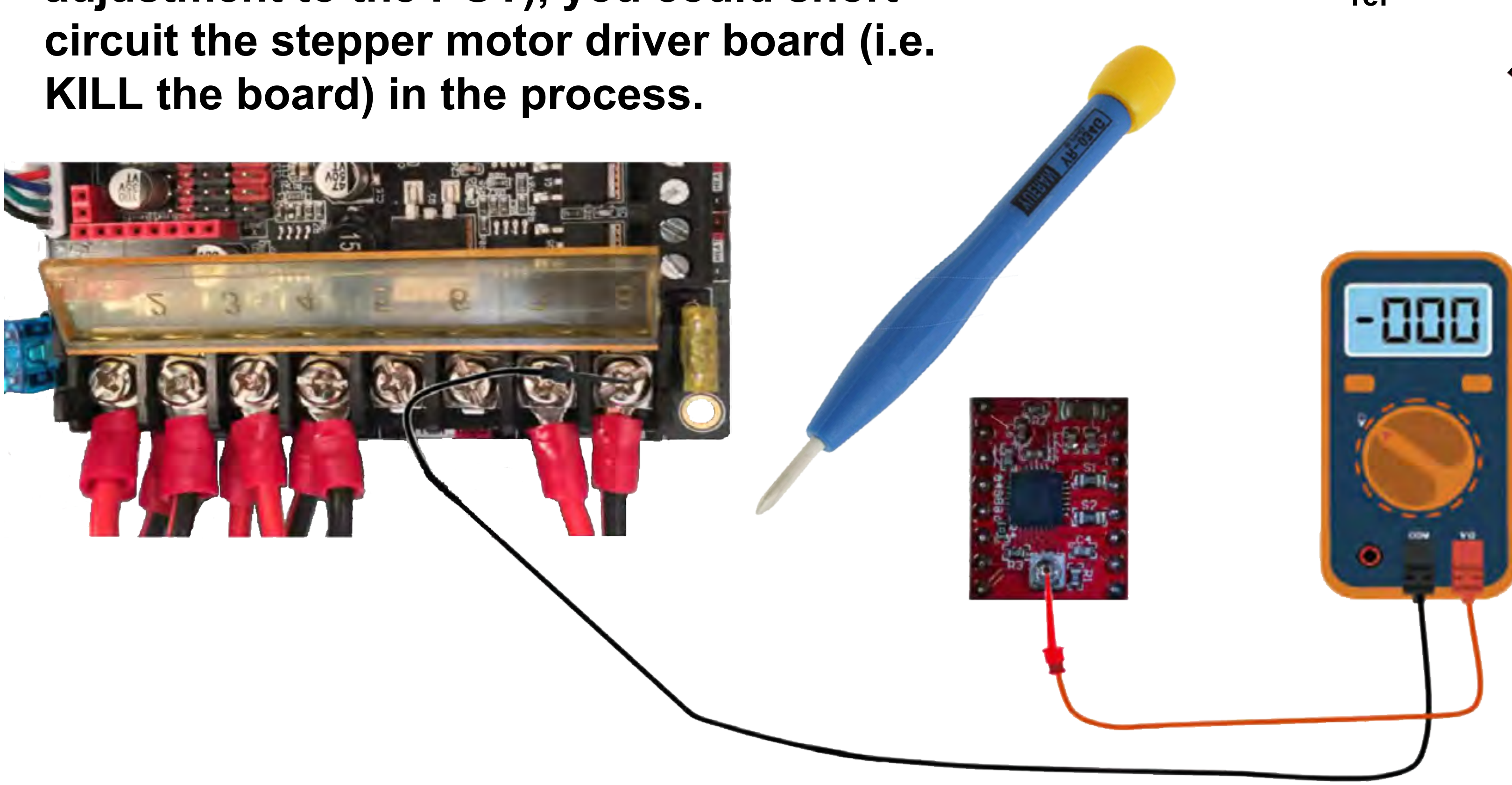
[1] from https://github.com/superjamie/lazyweb/wiki/3D-Printing-Stepper-Motors-and-Drivers and
[2] from https://github.com/superjamie/lazyweb/wiki/3D-Printing-Stepper-Motors-and-Drivers#how-to-tune-stepper-drivers

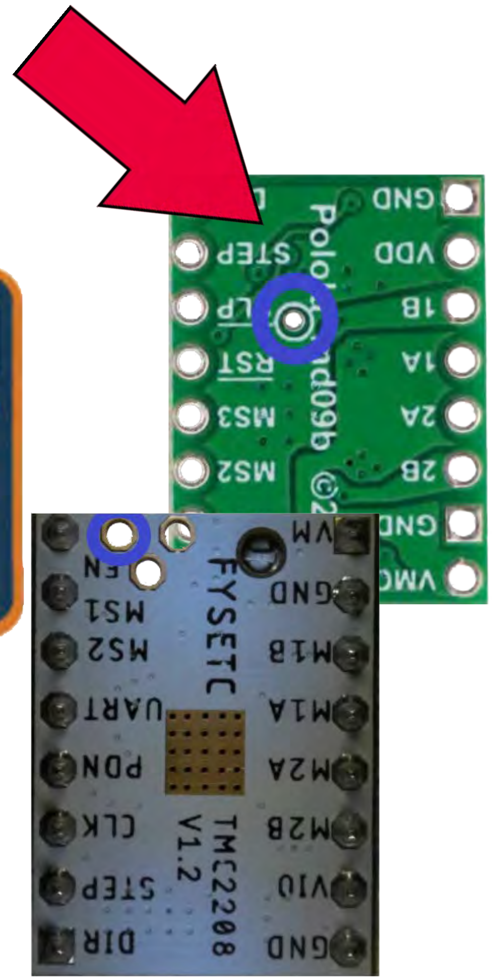
How to adjust the V_{ref} on a Stepper Motor Driver board using the Potentiometer[1, 2]

Note: A ceramic screw driver is non-conductive. If you use a plastic handle screw driver with alligator clips to your multimeter to make a "live adjustment" of V_{ref} (and you slip while making the adjustment to the POT), you could short circuit the stepper motor driver board (i.e. KILL the board) in the process.

Note: Some stepper motor driver boards have a "V_{ref} Test point" location, shown in BLUE. Check bottom or top of your board for a location.

If a "V_{ref} Test point" location is not available, use the potentiometer on the stepper driver board to do the V_{ref} measurement.





1 from https://github.com/superjamie/lazyweb/wiki/3D-Printing-Stepper-Motors-and-Drivers and
2 from https://github.com/superjamie/lazyweb/wiki/3D-Printing-Stepper-Motors-and-Drivers#how-to-tune-stepper-drivers

APPENDIX B

For the TMC drivers what's the difference between stand-alone mode and ("UART" or "SPI ") modes?

All the TMC driver chips **EXCEPT TMC5160 and TMC5161** have a stand-alone mode. **Stand-alone mode** allows you to just drop the driver into your motherboard to replace your A4988 driver. The **OTP mode** is similar, but you use some software to **PERMANENTLY** change the driver's mode of operation. The **UART or SPI modes** allow you to **dynamically** change the driver in your firmware.

How to Calculate V_{ref} for Non-TMC Stepper Motor Divers

My machine is an Ender 3, the X, Y, Z stepper motor "Rated Current" is 0.84 Amps, while E (extruder) stepper motor "Rated Current" is 1 Amps.

We use the V_{ref} formula (found on the first page of each different stepper motor driver section of this document and do the V_{ref} calculation.

Remember this V_{ref} calculation is just a suggested starting point. If your stepper motors are running too hot you will need to adjust the V_{ref} downward. If your stepper motors are skipping steps when printing then you will need to adjust your V_{ref} upwards. Our goal is to find a low enough V_{ref} where our stepper motors are cool enough without the printer missing any steps. For this example, I will use the A4988 stepper motor driver.

1. **So, A4988 V_{ref} formula is $V_{ref}=I_{MAX}*(8*R_s)$, where $R_s=0.1\Omega$.**
2. **I take each of my Axis' "Rated Current" and plug it into that equation to get X-Axis V_{ref} is equal to (0.840 * 8 * 0.100)=0.672 volts or 672mV.**
3. **Now, take 90% of that for a starting point V_{ref} value of ((0.672*0.90)=0.6048) 0.605 volts or 605mV or X-Axis V_{ref} = 0.605V. Since X, Y and Z stepper motors have the same "Rated Current" we now have the V_{ref} for X, Y and Z stepper motor drivers. Their value is 0.605 volts.**
4. **For E (extruder) V_{ref} the equation is (1.0 * 8 * 0.100)=0.800 volts. Now, take 90% of that, for a starting point, V_{ref} value of ((0.800*0.90)=0.720) 0.720 volts or 720mV. We now have the V_{ref} for E (extruder stepper motor driver) which is 0.72 volts.**
5. **We use our multimeter and turn the POT on the top of the stepper motor driver until we see the wanted V_{ref} voltage displayed.**

APPENDIX B

How to Calculate V_{ref} for TMC Stepper Motor Drivers

My machine is an Ender 3.The X, Y, and Z stepper motors have a "Rated Current" of 0.84 Amps, while E (extruder) stepper motor "Rated Current" is 1 Amp.

We use the I_{MAX} formula and use Algebra to find the V_{ref} formula. The I_{MAX} formula for each TMC Driver is listed on the following pages. But we will use TMC2100 drivers so we will use the below equation:

TMC2100 Stand-alone Mode, (with R_s=110mΩ):

$I_{RMS}=((V_{ref} / 2.5) * (1 / 1.41) * ((320mV / (R_s + 20m\Omega)))$

Since $I_{MAX}=(I_{RMS} * 1.41)$ is a known value then the above equation can be written as follows:

$(I_{MAX} * (1 / 1.41))= ((1 / 1.41) * (V_{ref} / 2.5) * ((320 / (110+20)))$

Since (1 / 1.41) is on both sides of the Algebra equation they cancel each other out leaving the equation as follows:

$I_{MAX} = (V_{ref} / 2.5) * (2.46 \cong 2.5)$,

Therefore $I_{MAX} = V_{ref}$, and $V_{ref} = I_{MAX}$.

We use 50% to 90% of V_{ref} (i.e. I_{MAX}) to set the current limit for TMC stepper motor

driver. To take 90% we do the following:

Since $V_{ref}=I_{MAX}$ we will use I_{MAX} instead of V_{ref}.

90% of V_{ref} for Ender 3's X, Y or Z =$(I_{MAX} * 0.90)=(0.84 * 0.90)=0.756$ or 756mA. Since $I_{MAX}=V_{ref}$, then it's also equal to 756mV.

Remember this V_{ref} calculation is just a suggested starting point. If your stepper motors are running too hot you will need to adjust the V_{ref} downward. If your stepper motors are skipping steps when printing then you will need to adjust your V_{ref} upwards. Our goal is to find a low enough V_{ref} where our stepper motors are cool enough without the printer missing any steps.

APPENDIX B

Driving Current Calculation Formulas for TMC Stepper Motor Drivers

1. **TMC2100 with R_s=0.110Ω (110mΩ) :**

I_{RMS}=((1 / 1.41)*(V_{ref} / 2.5)*((**320mV** / (**R_s + 20mΩ**)))
=((1 / 1.41)*(V_{ref} / 2.5)*((320/110+20)))
=((1/1.41)*(V_{ref}/2.5)*(2.46))
=(1/1.41)*V_{ref}*0.99
=(1/1.41)*V_{ref}*1
Since I_{RMS}=I_{MAX}*(1/1.41)
Therefore,
I_{MAX}*(1/1.41)=(1/1.41)*V_{ref}
I_{MAX}=V_{ref}
V_{ref}=I_{MAX}

2. **TMC2130 with R_s=0.110Ω (110mΩ) :**

I_{RMS}=((1 / 1.41)*(V_{ref} / 2.5)*((**325mV** / (**R_s + 20mΩ**)))
=((1 / 1.41)*(V_{ref} / 2.5)*((325/110+20)))
=((1/1.41)*(V_{ref}/2.5)*(2.5))
=(1/1.41)*V_{ref}
Since I_{RMS}=I_{MAX}*(1/1.41)
Therefore,
I_{MAX}*(1/1.41)=(1/1.41)*V_{ref}
I_{MAX}=V_{ref}
V_{ref}=I_{MAX}

3. **TMC2208 with R_s=0.110Ω (110mΩ) for Stand-alone Mode:**

I_{RMS}=((1 / 1.41)*(V_{ref} / 2.5)*((**325mV** / (**R_s + 30mΩ**)))
=((1 / 1.41)*(V_{ref} / 2.5)*((325/110+30)))
=((1/1.41)*(V_{ref}/2.50)*(2.32143))
=(1/1.41)*V_{ref}*0.928572
Since I_{RMS}=I_{MAX}*(1/1.41)
Therefore,
I_{MAX}*(1/1.41)=(1/1.41)*V_{ref}*0.928572
I_{MAX}=V_{ref}*0.9286
V_{ref}=I_{MAX} *1.0769

See next page for other TMC stepper motor drivers

APPENDIX B

Driving Current Calculation Formulas for TMC Stepper Motor Drivers

4. TMC2208 with R_s=0.110Ω (110mΩ) for UART Mode:

I_{RMS}=((1 / 1.41)*(V_{ref} / 2.5)*((**325mV** / (**R_s + 20**mΩ)))
=((1 / 1.41)*(V_{ref}/2.5)*((325/110+20)))
=((1/1.41)*(V_{ref}/2.5)*(2.5))
=(1/1.41)*V_{ref}*1
Since I_{RMS}=I_{MAX}*(1/1.41)
Therefore,
I_{MAX}*(1/1.41)=(1/1.41)*V_{ref}
I_{MAX}=V_{ref}
V_{ref}=I_{MAX}

5. TMC2209 with R_s=0.110Ω (110mΩ) for Stand-alone Mode:

I_{RMS}=((1 / 1.41)*(V_{ref} / 2.5)*((**325mV** / (**R_s + 20**mΩ)))
=((1/1.41)*(V_{ref}/2.5)*((325/(110+20)))
=((1/1.41)*(V_{ref}/2.5)*(2.5))
=(1/1.41)*V_{ref}*1
Since I_{RMS}=I_{MAX}*(1/1.41)
Therefore,
I_{MAX}*(1/1.41)=(1/1.41)*V_{ref}
I_{MAX}=V_{ref}
V_{ref}=I_{MAX}

See next page for other TMC stepper motor drivers

APPENDIX B

Driving Current Calculation Formulas for TMC Stepper Motor Drivers

6. TMC2209 with R_s=0.110Ω (110mΩ) for UART Mode:

I_{RMS}=((1 / 1.41)*(V_{ref} / 2.5)*((**325mV** / (**R_s + 20mΩ**)))
=((1 / 1.41)*(V_{ref} / 2.5)*((325/110+20)))
=((1/1.41)*(V_{ref}/2.5)*(2.5))
=(1/1.41)*V_{ref}*1
Since I_{RMS}=I_{MAX}*(1/1.41)
Therefore,
I_{MAX}*(1/1.41)=(1/1.41)*V_{ref}
I_{MAX}=V_{ref}
V_{ref}=I_{MAX}

7. TMC2225 with R_s=0.150Ω (150mΩ) for UART Mode:

I_{RMS}=((1 / 1.41)*(V_{ref} / 2.5)*((**325mV** / (**R_s + 20mΩ**)))
=((1/1.41)*(V_{ref}/2.5)*((325/(150+20)))
=((1/1.41)*(V_{ref}/2.5)*(1.9118)
=(1/1.41)*V_{ref}*0.7647
Since I_{RMS}=I_{MAX}*(1/1.41)
Therefore,
I_{MAX}*(1/1.41)=(1/1.41)*V_{ref}*0.7647
I_{MAX}=V_{ref}*0.7647
V_{ref}=I_{MAX}*1.3077

8. TMC5160 with R_s=0.075Ω (75mΩ) for SPI Mode:

I_{RMS}=((1 / 1.41)*(**325mV** / **R_s**))
=((1/1.41)*(325/75))
=(1/1.41)*4.33

Since I_{RMS}=I_{MAX}*(1/1.41) Therefore,
I_{MAX}*(1/1.41)=(1/1.41)*4.333
I_{MAX}=4.333 Amps

You will use **50% to 90%** of I_{MAX} (4.333*.50 or 4.333*.90) which is **2.1665 Amps (2167 mA)** to **3.8997 Amps (3900 mA)** for the Marlin Firmware.

See next page for other TMC stepper motor drivers

APPENDIX B

Driving Current Calculation Formulas for TMC Stepper Motor Drivers

9. TMC5161 with R_s=0.062Ω (62mΩ) for SPI Mode:

I_{RMS}=((1 / 1.41)*(**325mV** / R_s))
=((1/1.41)*(325/62))
=(1/1.41)*5.24194

Since I_{RMS}=I_{MAX}*(1/1.41) Therefore,
I_{MAX}*(1/1.41)=(1/1.41)*5.24194
I_{MAX}=5.24194 or 5.2419

You will use **50% to 90%** of I_{MAX} (5.2419*.50 or 5.2419*.90) which is **2.621 Amps (2621 mA)** to **4.7177 Amps (4718 mA)** for the Marlin Firmware.

10. TMC2225 with R_s=0.150Ω (150mΩ) for Stand-alone Mode:

I_{RMS}=((1 / 1.41)*(V_{ref} / 2.5)*((**325mV** / (R_s + **30**mΩ)))
=((1/1.41)*(V_{ref}/2.5)*((325/(150+30)))
=((1/1.41)*(V_{ref}/2.5)*(1.8056)
=(1/1.41)*V_{ref}*0.7222
Since I_{RMS}=I_{MAX}*(1/1.41)
Therefore,
I_{MAX}*(1/1.41)=(1/1.41)*V_{ref}*0.7222
I_{MAX}=V_{ref}*0.7222
V_{ref}=I_{MAX}*1.3846

APPENDIX C

The (Latest Release of) Marlin Setup That Is Common To ALL Stepper Motor Drivers

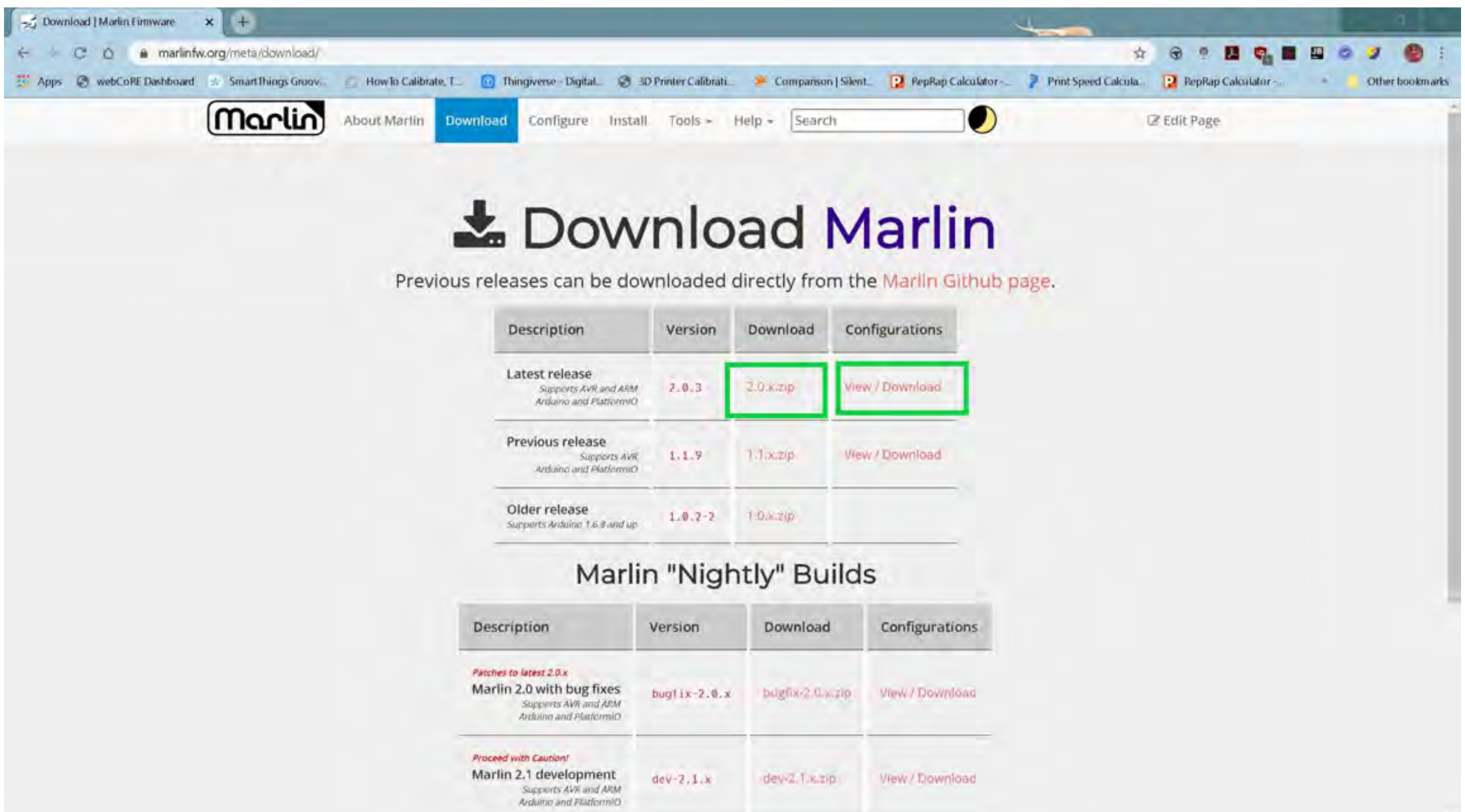
Please follow this guide to install Visual Studio Code with Platformio onto your computer. The link to the instruction are located at: https://marlinfw.org/docs/basics/install_platformio.html

Please refer to the following documents:

- **BIGTREETECH SKR-PRO-V1.1 User Manual.pdf**
- **BIGTREETECH SKR-PRO-V1.1 Guide.pdf**

This example will use the Creality Ender 3 printer. Select the appropriate default configuration files for your specific printer!

- Download the latest release of Marlin from here: **https://marlinfw.org/meta/download/**
- Unzip the latest release of Marlin onto your hard drive
- Also, download the latest release of the Marlin Configuration files and then unzip the Marlin Configuration files so they reside in the same subdirectory as the Marlin files, see the pictures below for how I organize my file structure for Marlin

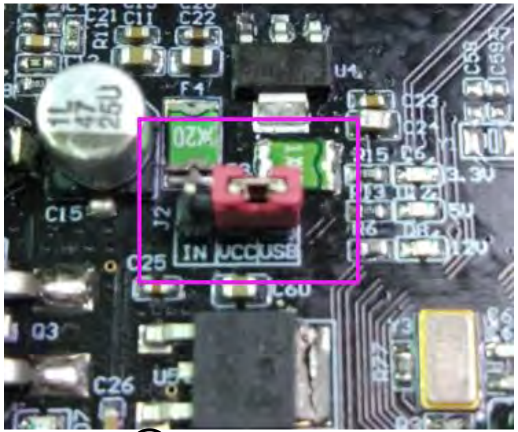


- Go to the next page.

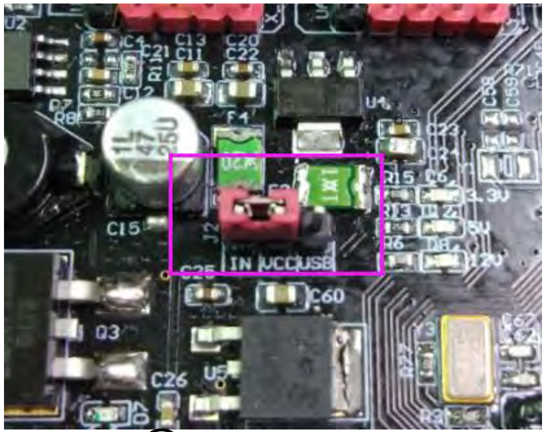
The (Latest Release of) Marlin Setup That Is Common To ALL Stepper Motor Drivers

To ensure that any changes I make to my Marlin Firmware are permanent, I do a couple of things. I will send the G-code M503 by using the **Pronterface software** **BEFORE I change the Marlin Firmware** so I can **write down all the calibrations that I have previously set**.

I ensure that the "Power selection", as defined in "SKR PRO V1.1 user manual v1.1.pdf" is set to the correct power input source ②. Even though you can use the USB as the source of power, I prefer to use the 12V/24V DC power source, as shown below.



①USB power



②12/24V power

Marlin2.0 Firmware Update Method:

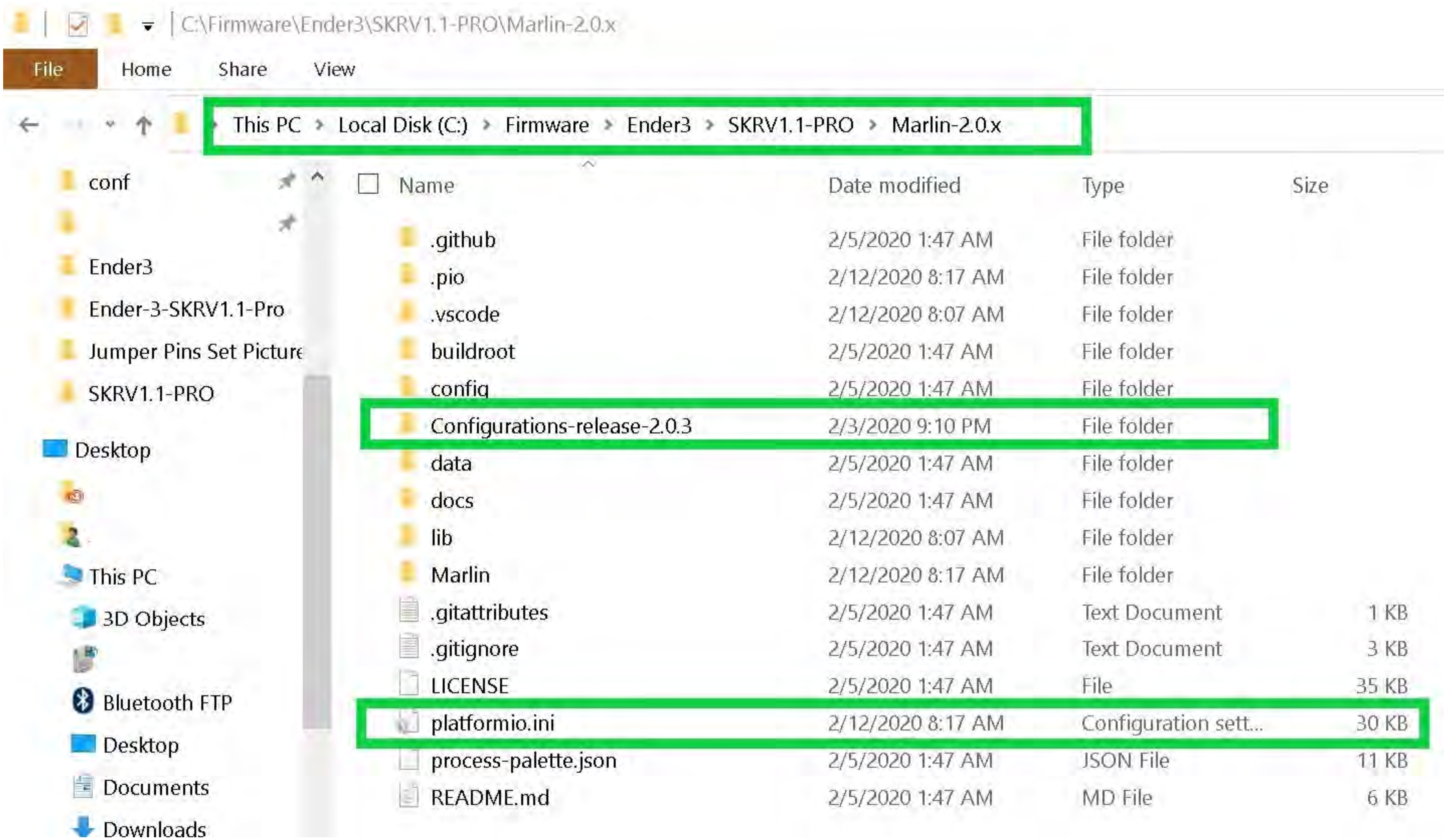
After downloading the files, use Visual Studio Code (VScode) to open the project for compilation. Customize the firmware and compile it. Check for errors. If there are no errors, find the "firmware.bin" file. Copy it to the SD card and plug the micro SD card into the board. Reboot the board; wait for about 10 seconds before doing anything else with the board. I then use the **Pronterface software**, to send the following "G-code" commands:

- M502
- M500
- M504

M502 reset all configurable settings to their factory defaults. When you follow a M502 by a M500 the M500 will also reset settings in EEPROM. The M504 command will validate the contents of the EEPROM to ensure that the EEPROM settings have been changed to the factory defaults. If the reset does not show the correct settings, find your compiled "firmware.bin" file and copy it again to the SD card, then plug the micro SD card into the SKR PRO board. Reboot the board, wait for about 10 seconds and check the settings again.

After uploading new firmware, you will need to calibrate your 3D printer again. Please see the following document for instructions on how to calibrate your 3D printer: https://drive.google.com/open?id=19HVpv2jNMkPamlhMGiEaGEmsnmzsoKVe

The (Latest Release of) Marlin Setup That Is Common To ALL Stepper Motor Drivers



- Go to the next page.

The (Latest Release of) Marlin Setup That Is Common To ALL Stepper Motor Drivers

NOTE: This example will use the Creality Ender 3 printer and using the latest release of Marlin firmware which is version 2.0.3. Select the appropriate default configuration files for your specific printer!

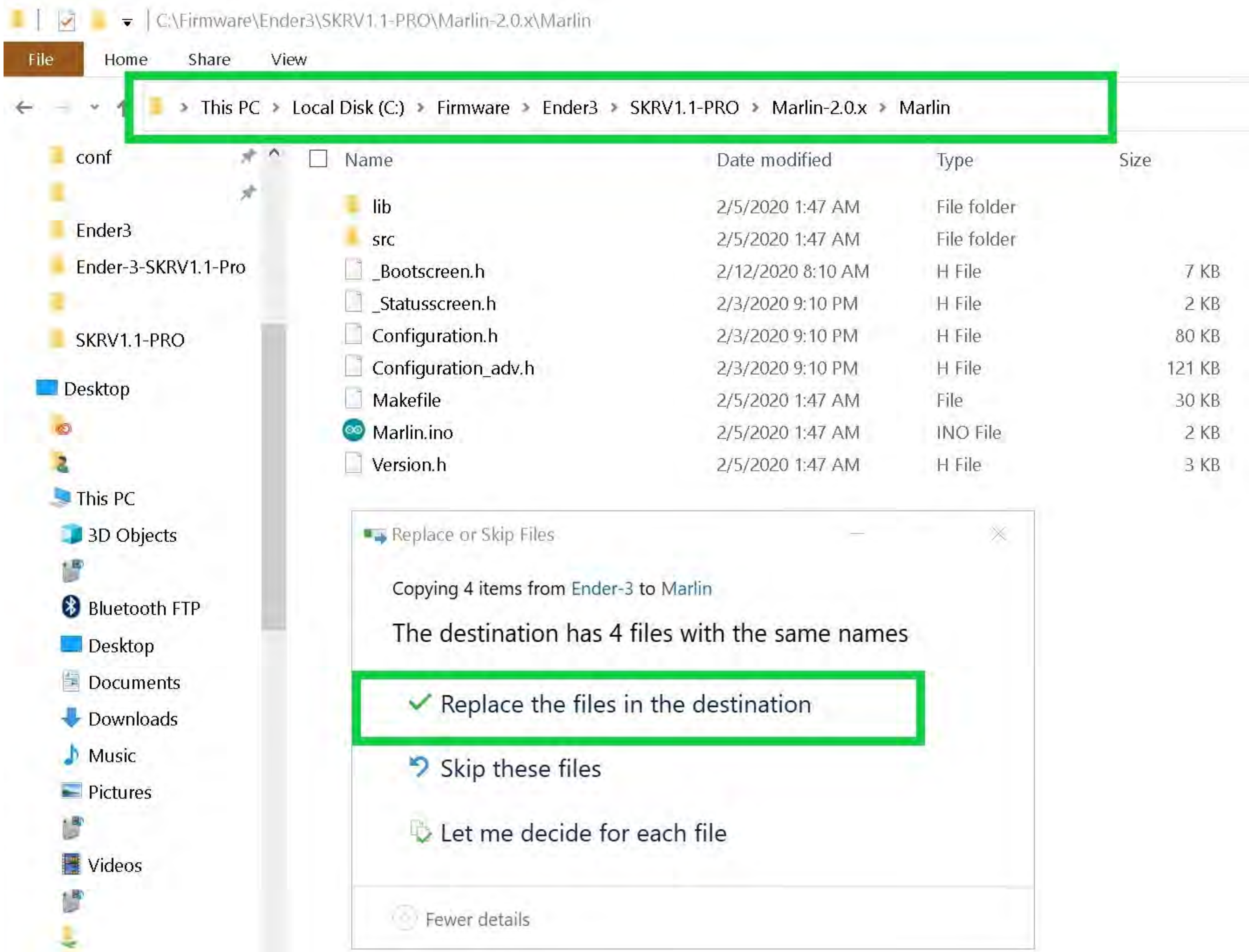
- Copy the below default configuration files (for me Ender 3 files, for you it could be another printer) to the directory where Marlin's Configuration.h and Configuration_adv.h reside. See picture below for which files you will copy. See the next page to see where to place the files.

- Go to the next page.

APPENDIX C

The (Latest Release of) Marlin Setup That Is Common To ALL Stepper Motor Drivers

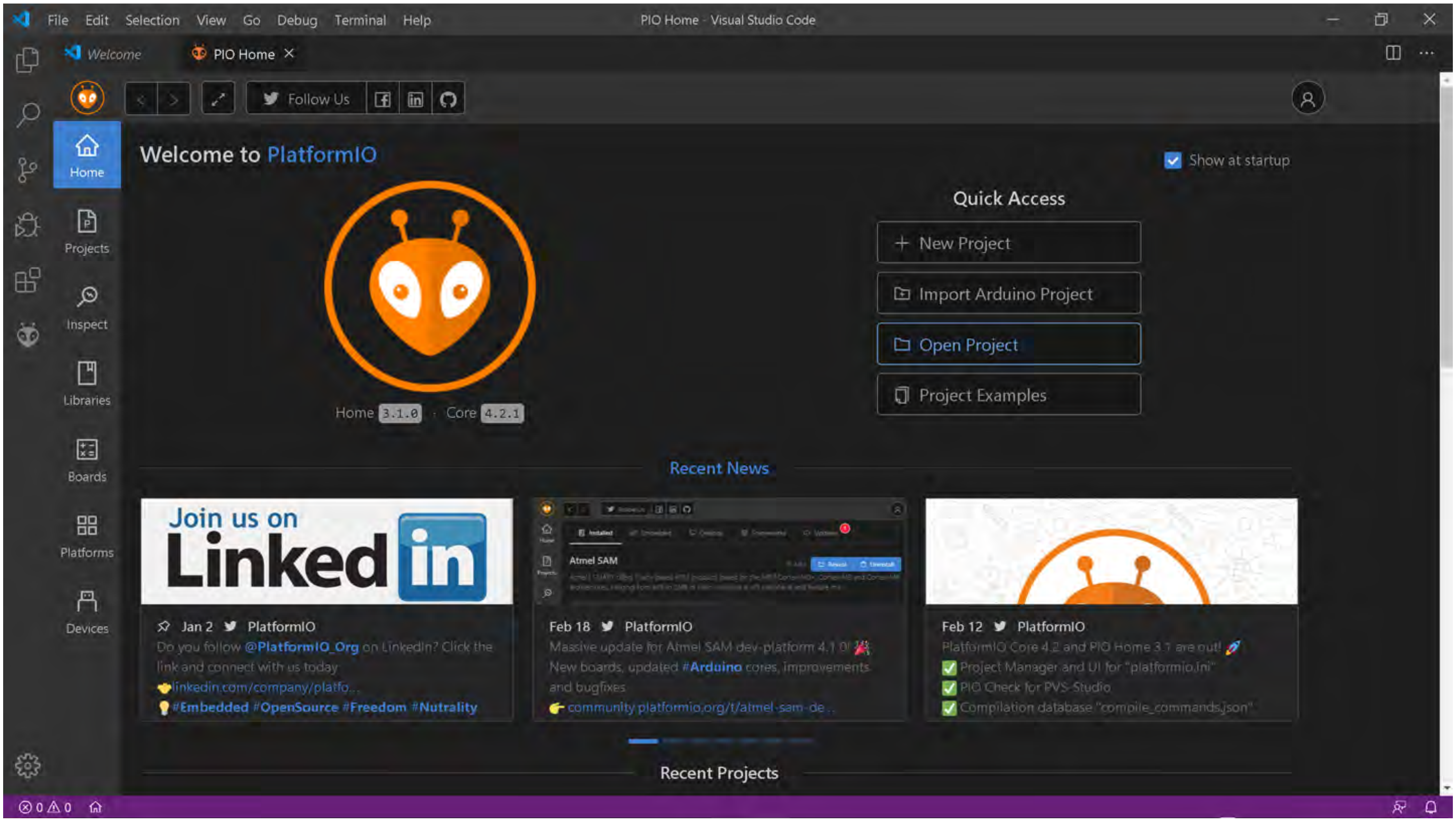
- Copy TO THIS directory so the above files reside in the same directory as Configuation.h and Configuration_adv.h. When prompted allow the files to be overwritten!



- Go to the next page.

The (Latest Release of) Marlin Setup That Is Common To ALL Stepper Motor Drivers

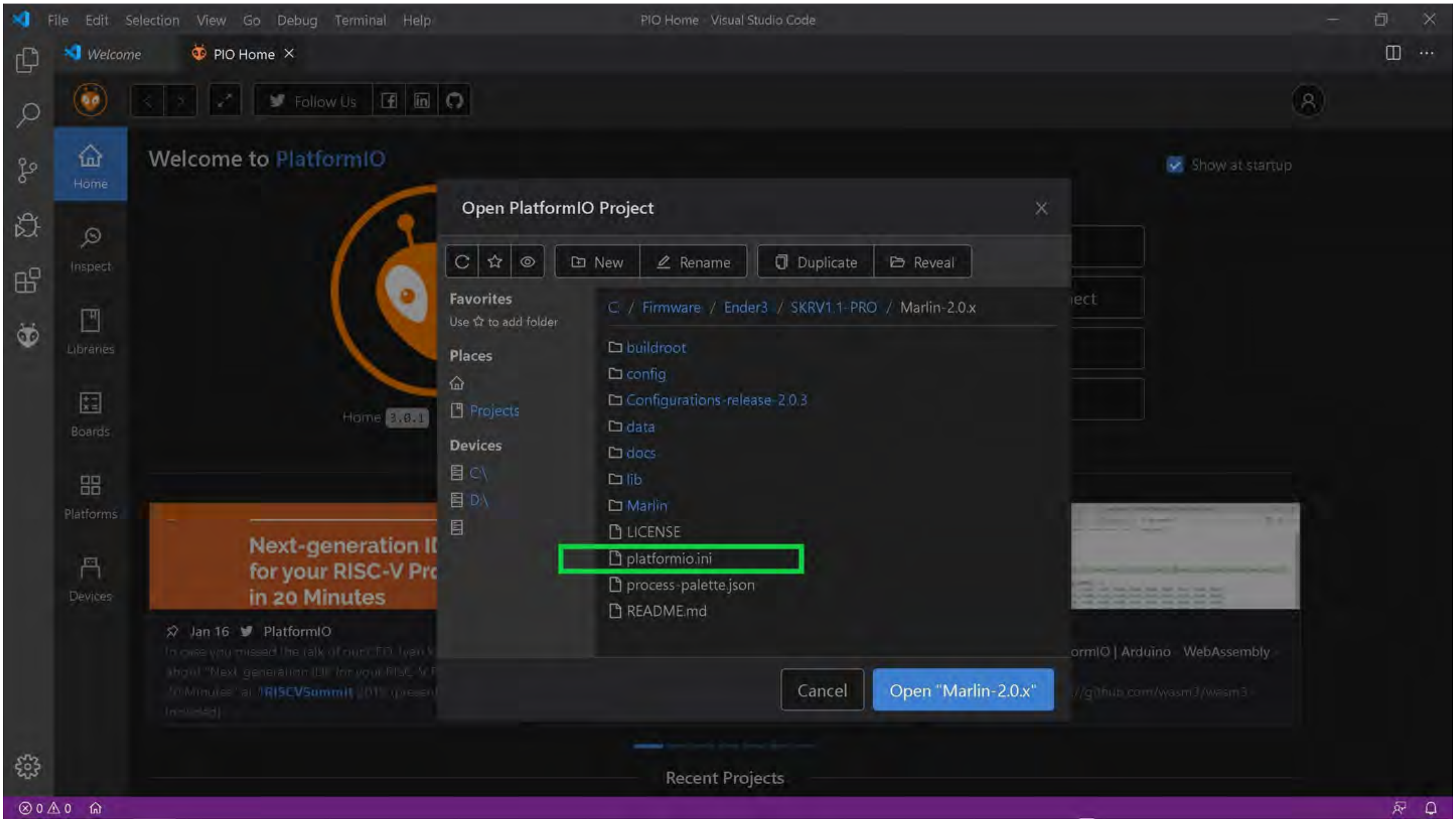
- Open VS code (see picture below) and then select "Open Project".



- Go to the next page.

The (Latest Release of) Marlin Setup That Is Common To ALL Stepper Motor Drivers

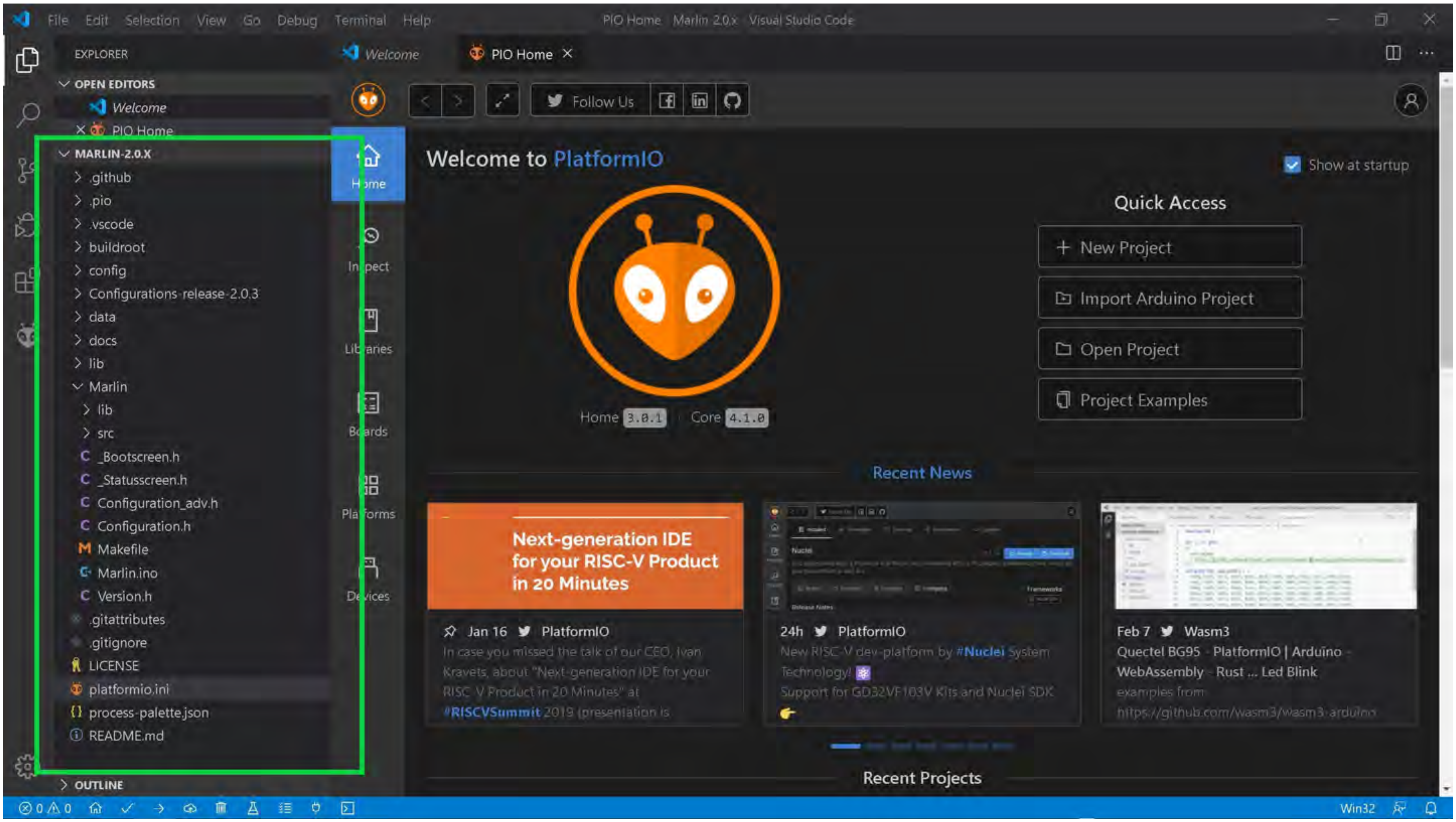
- Go to the directory where the platformio.ini file resides and open that folder (see picture below)



- Go to the next page.

The (Latest Release of) Marlin Setup That Is Common To ALL Stepper Motor Drivers

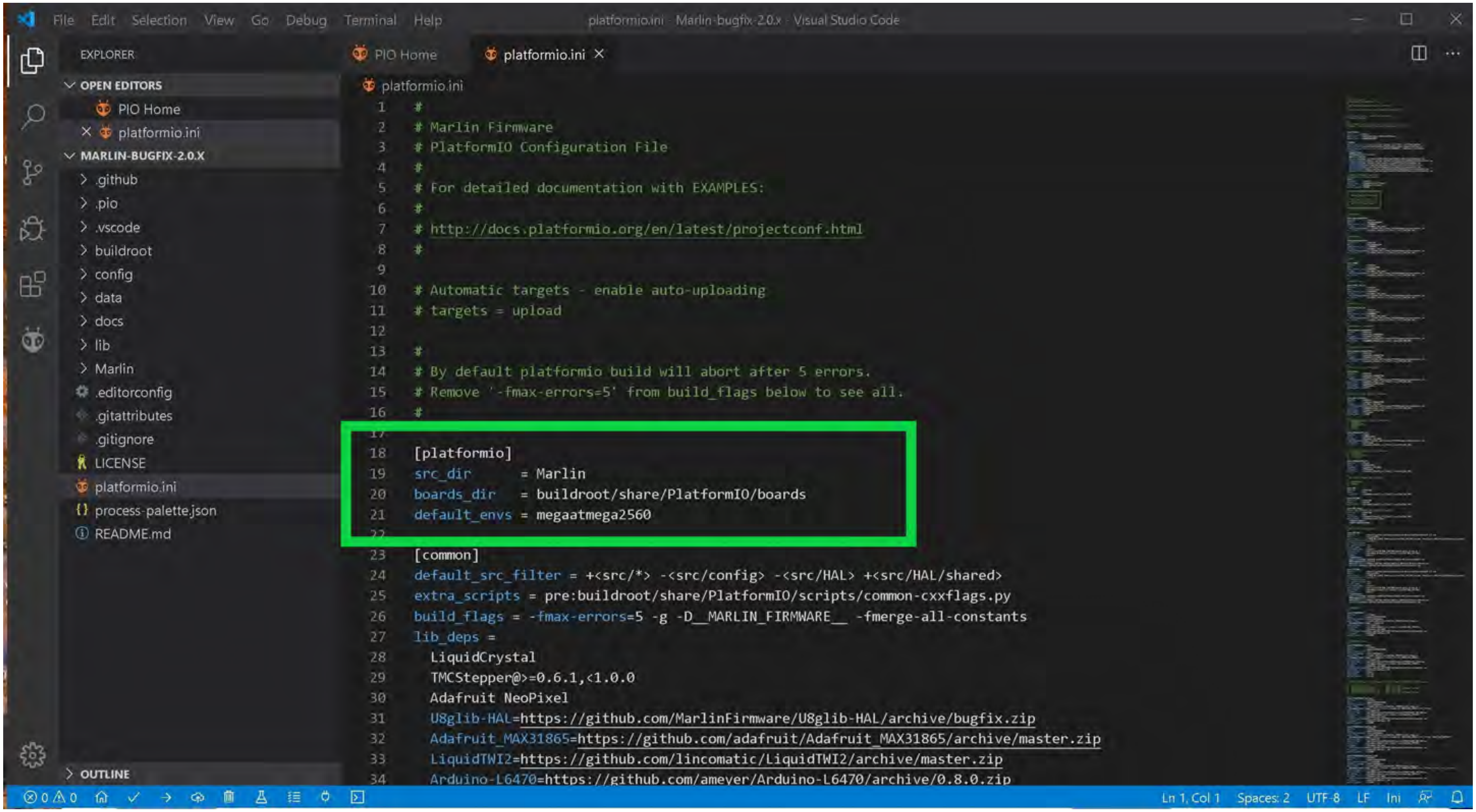
- On the left side you will see the file structure, double click on the "platformio.ini" file to open it up in the editor window.



- Go to the next page.

The (Latest Release of) Marlin Setup That Is Common To ALL Stepper Motor Drivers

- What you will see when "platformio.ini" is opened up in the VS code editor window

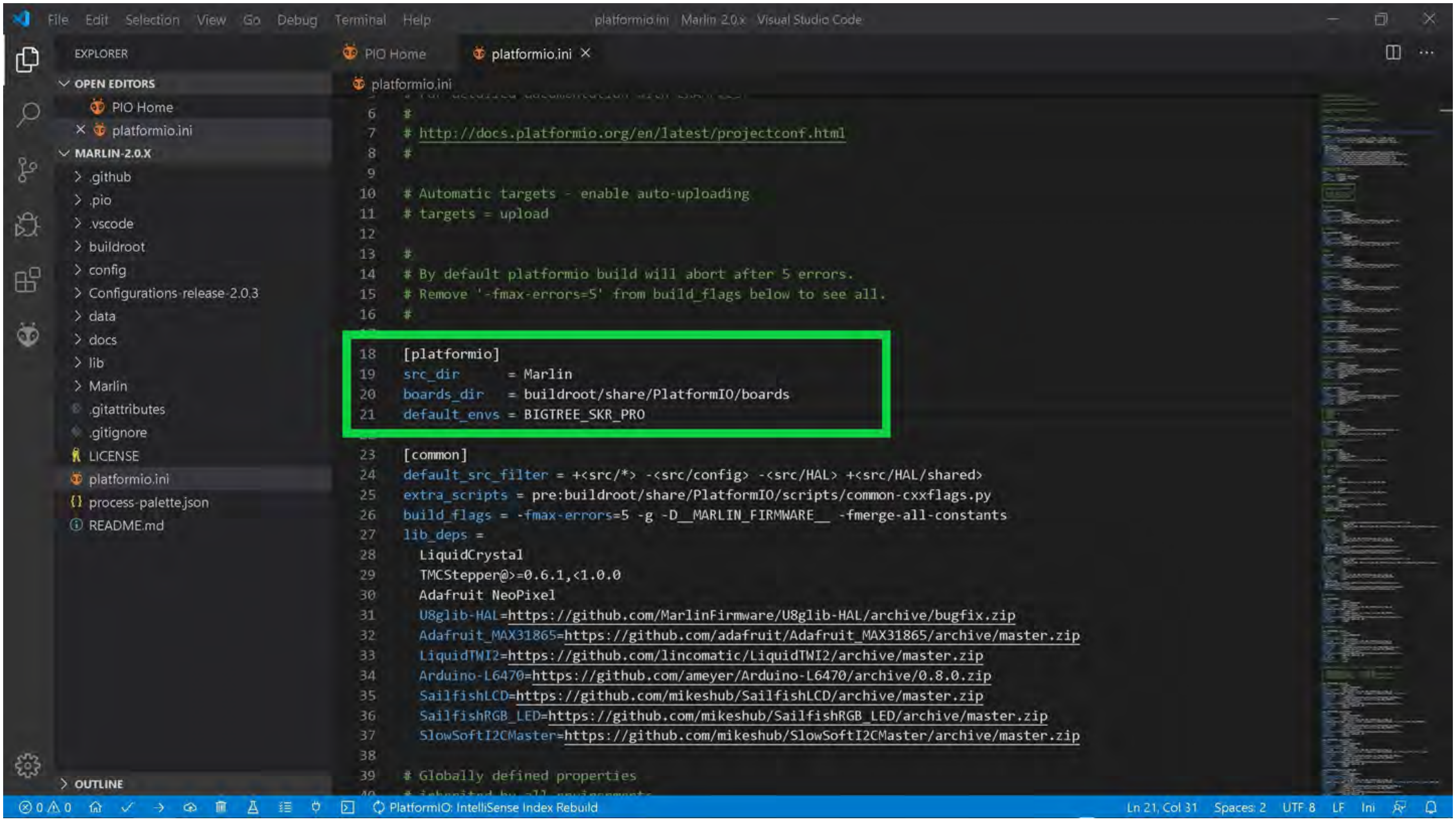


- Go to the next page.

APPENDIX C

The (Latest Release of) Marlin Setup That Is Common To ALL Stepper Motor Drivers

- Edit the "default_envs = megaatmega2560" line and make "default_envs = BIGTREE_SKR_PRO", as shown below in **GREEN**

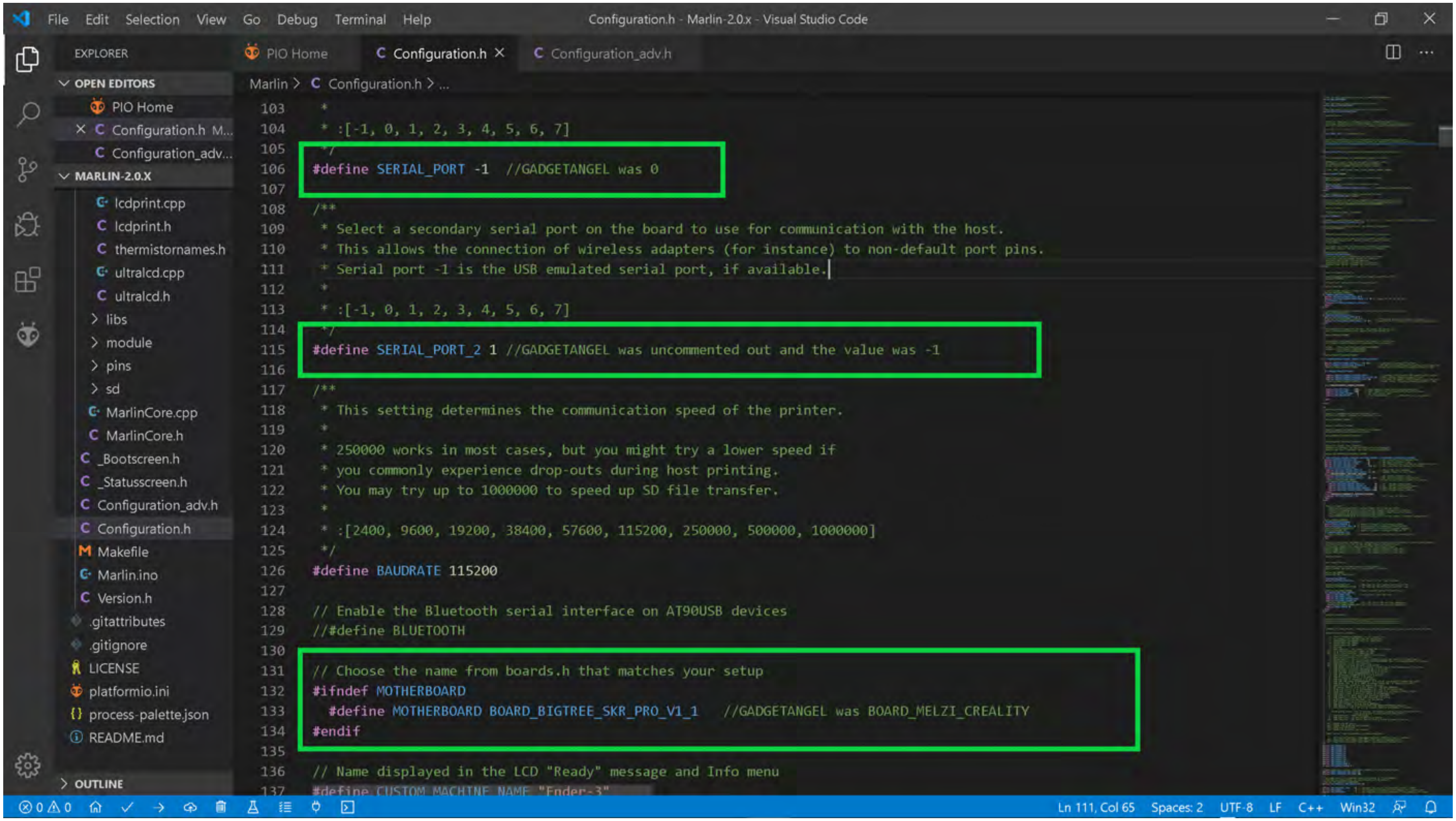


- Go to the next page.

NOTICE: pin labels are relative to stepper driver board; double check compatibility before use. Numbered pairs denote required jumpers.

The (Latest Release of) Marlin Setup That Is Common To ALL Stepper Motor Drivers

- Go to the Configuration.h file and change the following three items, as seen in the **3 GREEN** boxes below.



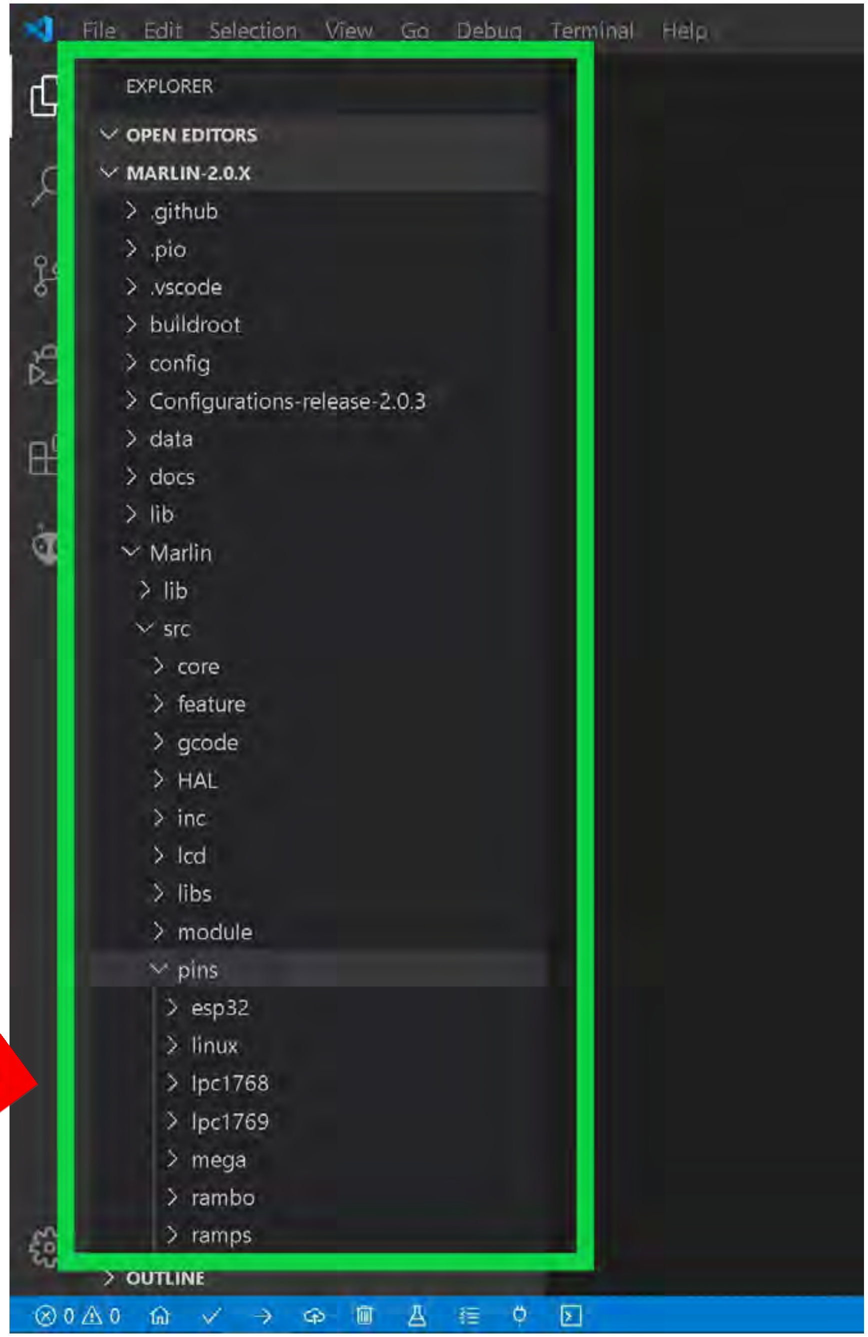
- You can set "BAUDRATE" to "115200" or "250000 ". Either setting will work but I have found that "115200" option works with any LCD that you choose to use.

- Go to the next page.

APPENDIX C

The (Latest Release of) Marlin Setup That Is Common To ALL Stepper Motor Drivers

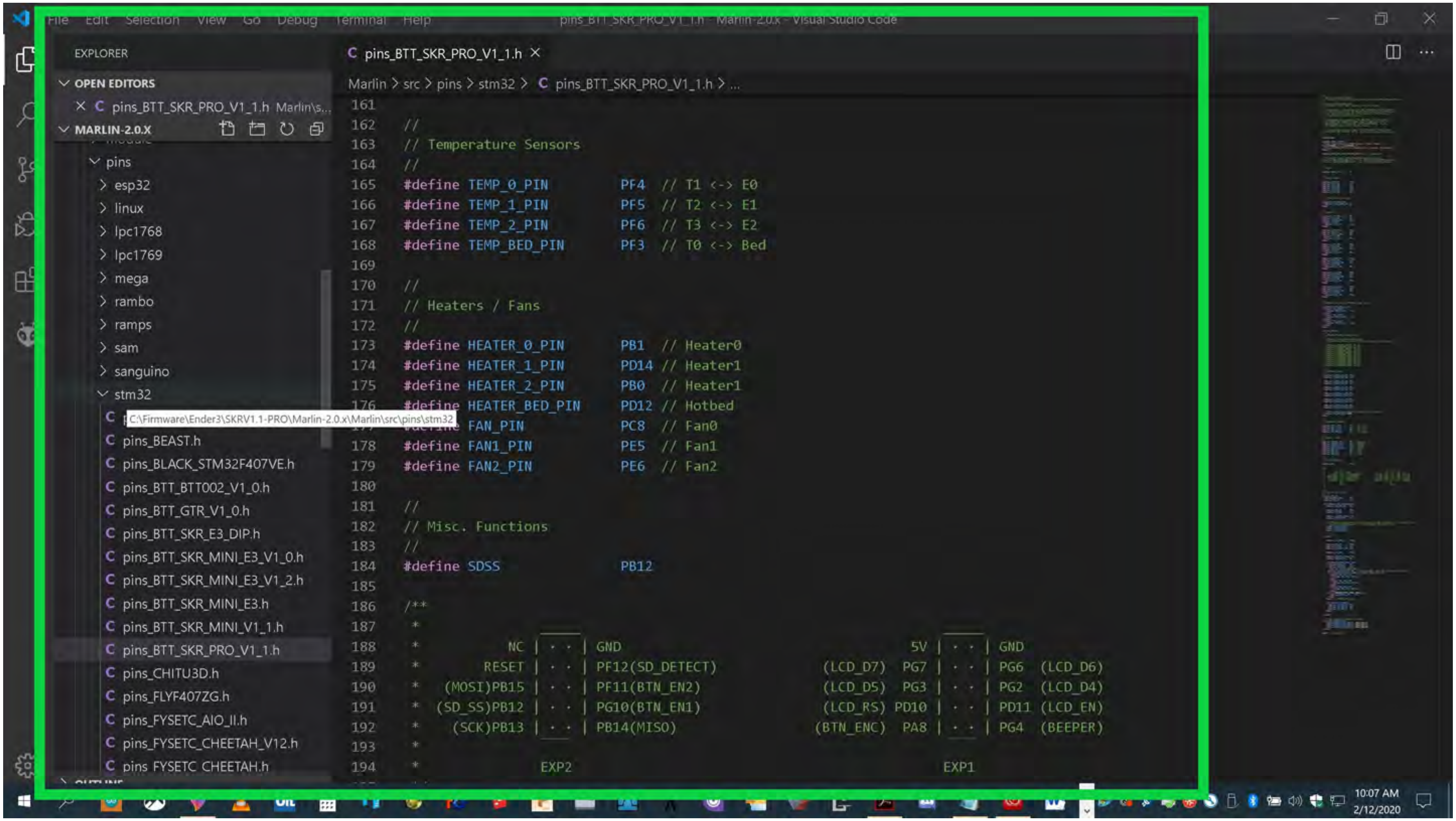
- Time to learn where the SKR PRO V1.1 board's pins file is located: look at the left side and find the Marlin-2.0.X/Marlin/src/pins/stm32 subdirectory, as seen in the pictures below. Open the file, pins_BTT_SKR_PRO_V1_1.h, by double clicking on it.



- Go to the next page.

APPENDIX C

The (Latest Release of) Marlin Setup That Is Common To ALL Stepper Motor Drivers

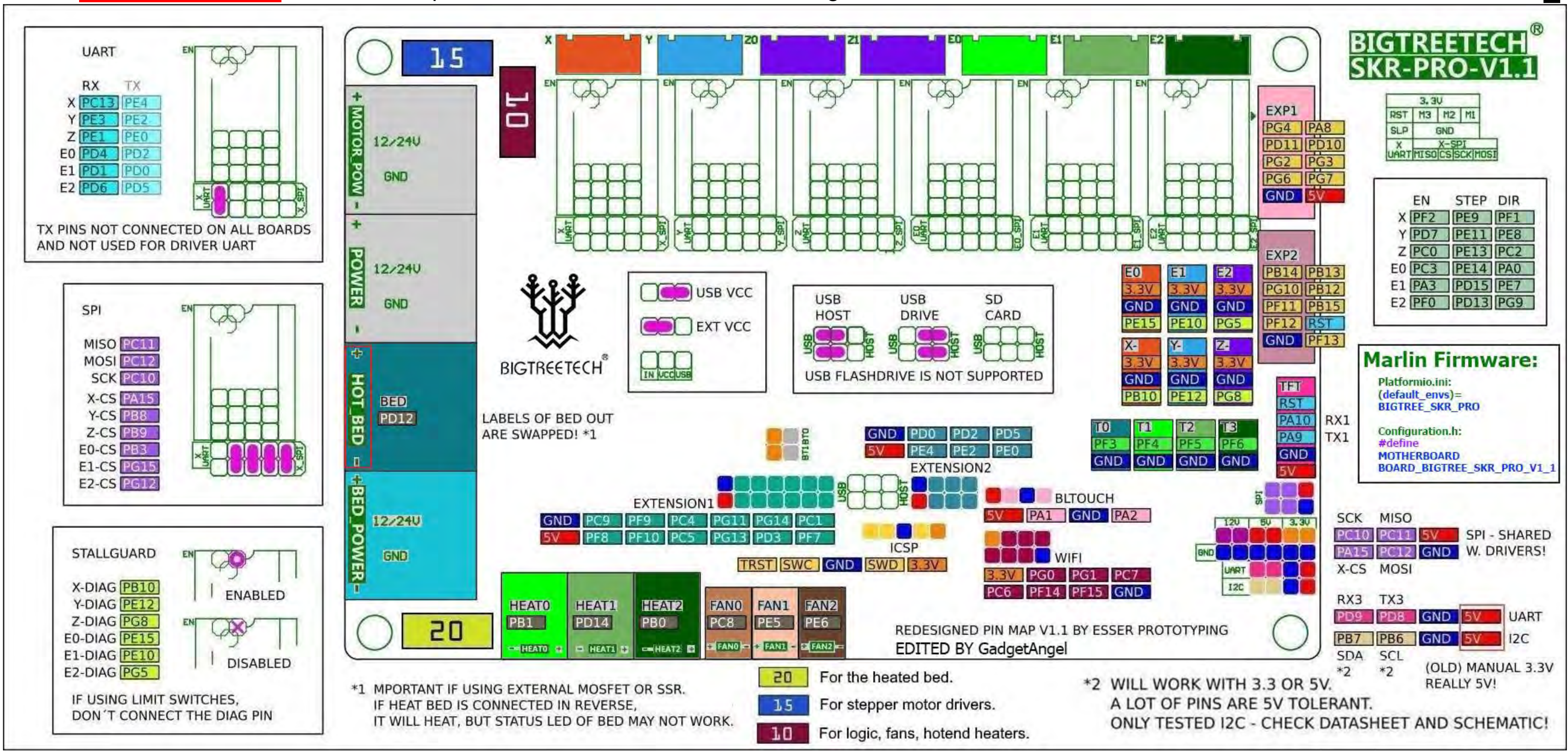


- Go to the next page.

APPENDIX C

The (Latest Release of) Marlin Setup That Is Common To ALL Stepper Motor Drivers

- We want to set the fan on the hot end to automatically turn on when the hot end starts to get hot. In the pins_BTT_SKR_PRO_V1_1.h file you will find all the pins that are defined for the board. They should all conform to the SKR PRO V1.1 Pin diagram shown below. See the next page for a function picture of the SKR PRO V1.1 wire diagram.[3]

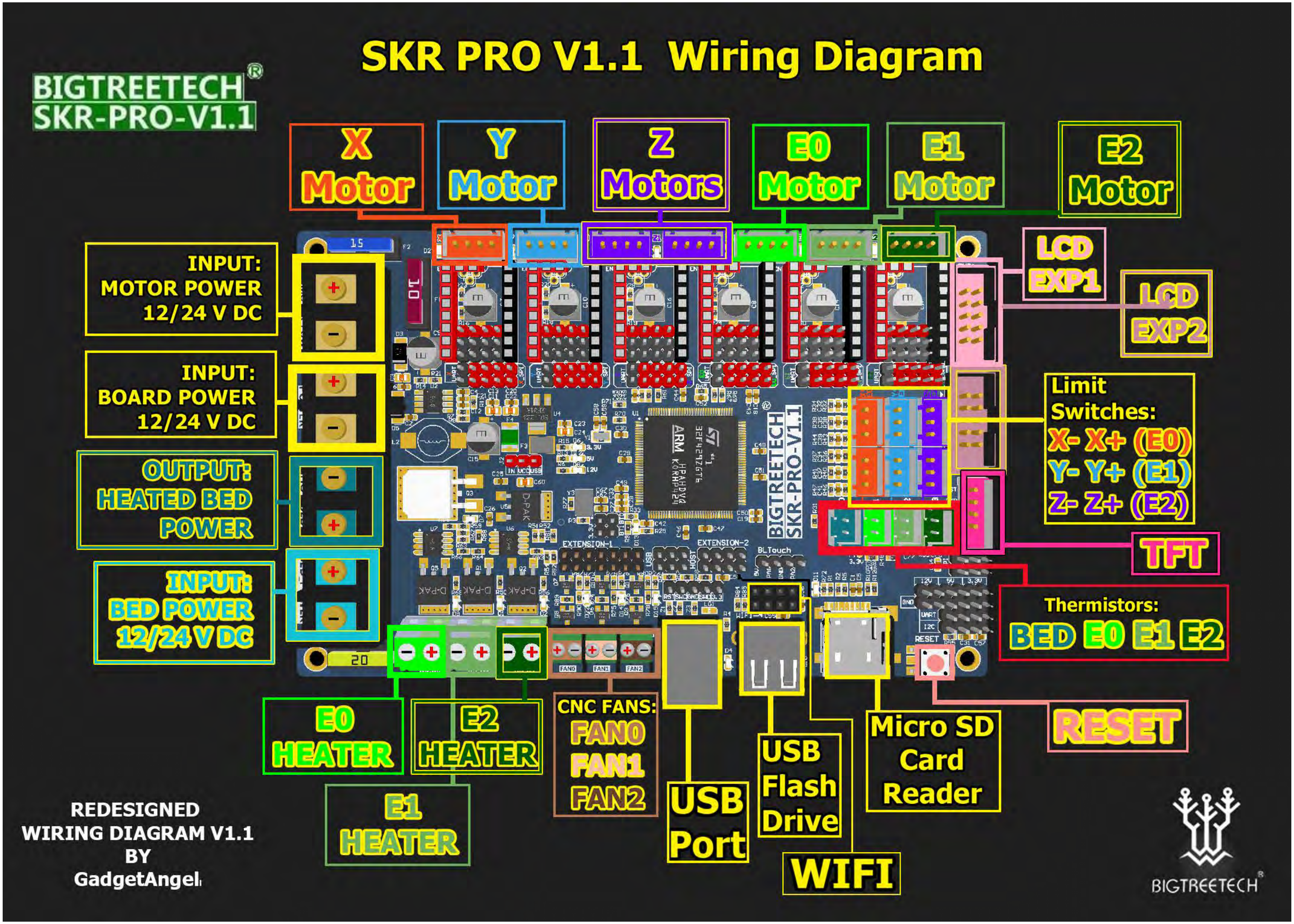


- We want to copy and paste the Marlin name or the actual pin number of where you hooked up the fan for your hot end and use that to set "E0_AUTO_FAN_PIN" in the Configuration_adv.h file. I am connecting FAN1 port to my electronics case fan. I am connecting FAN0 port to my part or print cooling fan and connecting FAN2 port to my dual 5015 hot end cooling fans. But we are only interested in how to set the hot end cooling fan up. So, in this example I will use FAN2 port for my hot end cooling fan. In pins_BTT_SKR_PRO_V1_1.h file we see FAN2 port is defined as PE6. So you can choose to copy "PE6" or "FAN2_PIN". Go to next page.

[3] Pin Diagram is done by **Thomas White** and updated by **GadgetAngel**

APPENDIX C

The (Latest Release of) Marlin Setup That Is Common To ALL Stepper Motor Drivers

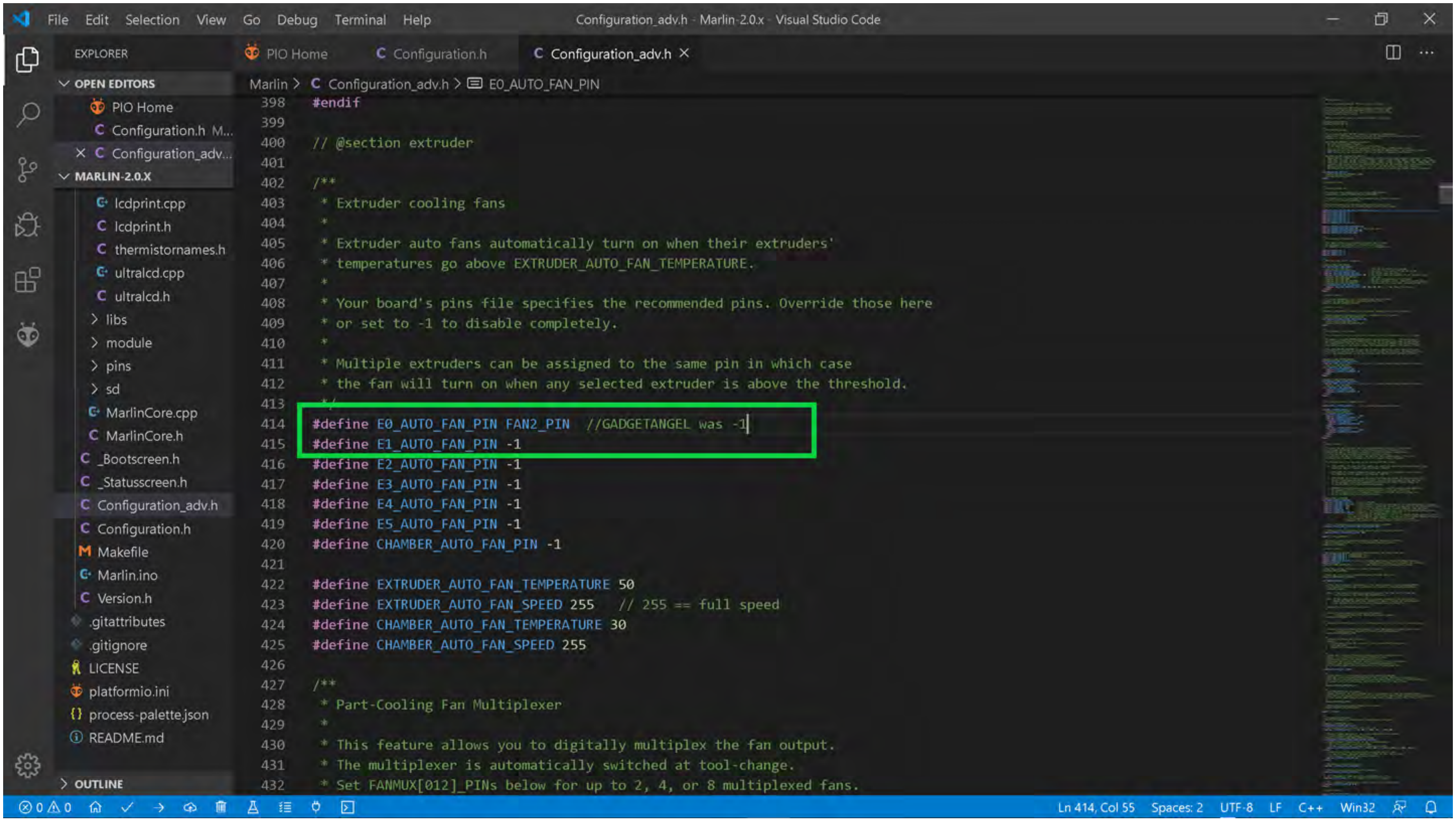


- Go to the next page.

APPENDIX C

The (Latest Release of) Marlin Setup That Is Common To ALL Stepper Motor Drivers

- To set the hot end fan, I will use FAN2_PIN and set "E0_AUTO_FAN_PIN" in the Configuration_adv.h file to FAN2_PIN, as seen in the picture below



- To see more Marlin setup for the latest release, **please refer to the stepper motor driver section of this document for the stepper motor driver of your choice.**