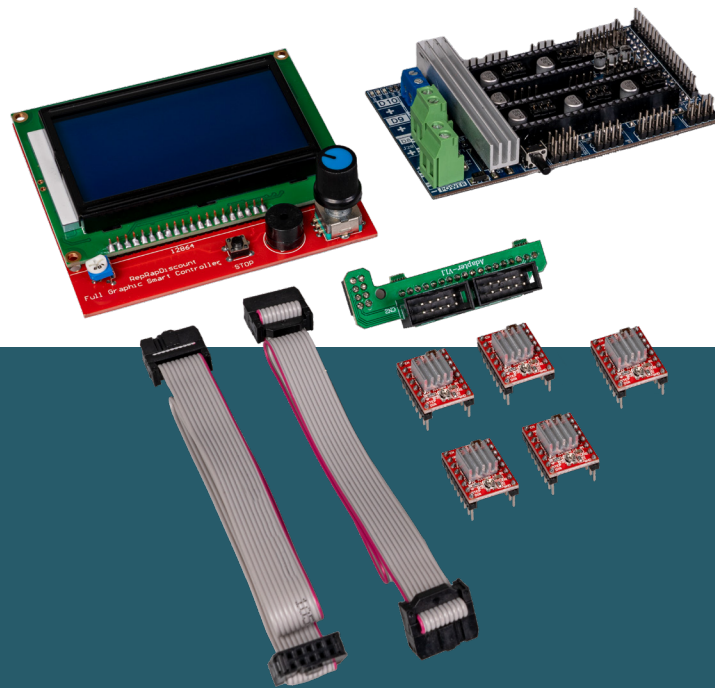




RAMPS 1.6 SET

RAMPS 1.6 MIT DISPLAY & 5X A4988 MOTOR DRIVER

1. GENERAL INFORMATION

Dear Customer,

Thank you for choosing our product. In the following, we will show you what to consider during commissioning and use.

Should you encounter any unexpected problems during use, please feel free to contact us.

With this complete set you lay the foundation for installing your own Arduino-based 3D printer.

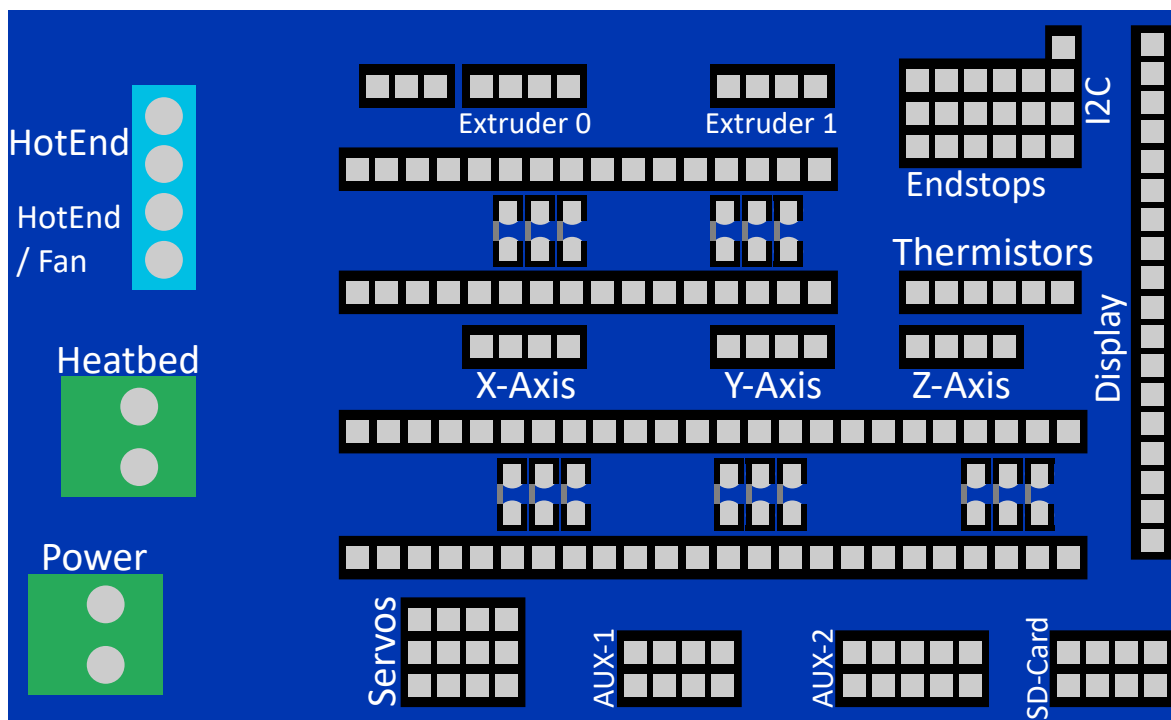
In addition to the required stepper motor drivers, the board also provides connections for up to three heating units (print bed heater, extruder 1, extruder 2). Each heating unit offers three inputs for connecting temperature sensors.

It is also possible to connect fans for cooling the components and extruders.

With the appropriate software installed on the Arduino compatible [**ARD-MEGA2560R3**](#), current status messages or information such as temperature can be displayed on the large LCD display included in the scope of delivery.

2. SCHEMATIC DRAWING

The following schematic diagram shows a detailed pin assignment.

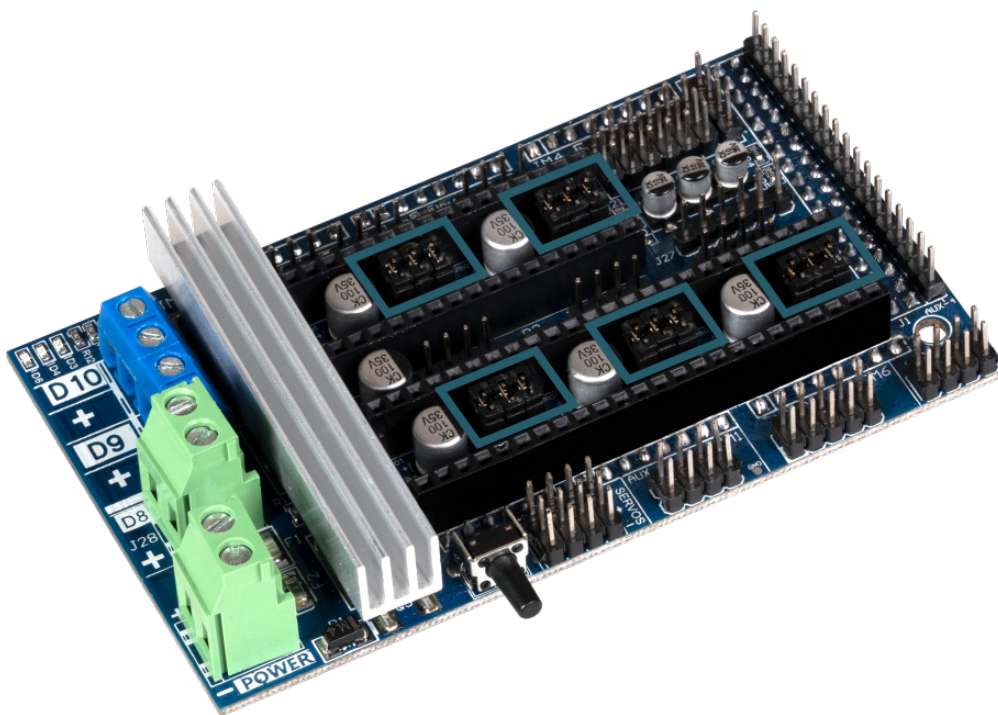


3. CONNECTION TO THE ARDUINO MEGA

First, start by connecting the Ramps 1.6 expansion board to your [ARD-MEGA2560R3](#). Simply plug the board onto your ARD-MEGA2560R3 so that the pins on the bottom of the board are routed into the corresponding connectors on the ARD-MEGA2560R3.

4. JUMPER CONFIGURATION

Before the necessary stepper motors can be plugged on the expansion board, it is necessary to set the respective jumpers. The jumpers define the number of possible microsteps for the stepper motors. Up to 1/128 steps are supported. The exact configuration depends on the used stepper motor drivers.



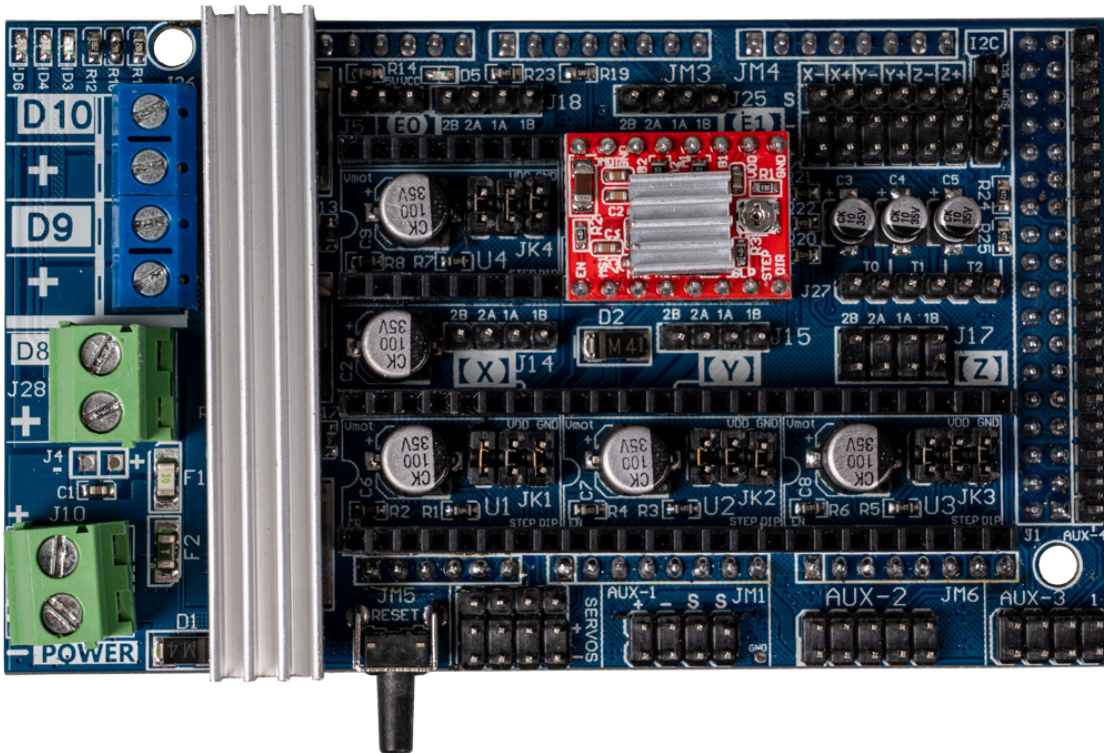
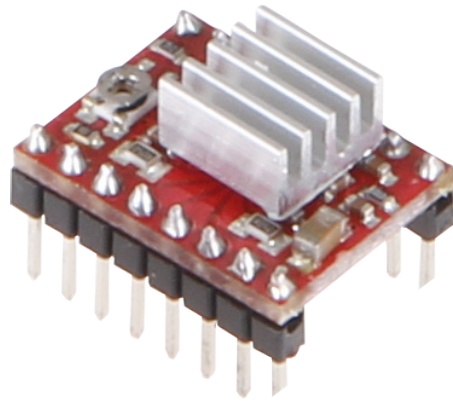
To determine the step sizes with which the motors should move, you can set jumpers on the main board. The pins used for this are marked in the picture above.

JUMPER 1	JUMPER 2	JUMPER 3	STEPS
No	No	No	Whole step
Yes	No	No	Half step
No	Yes	No	1/4 step
Yes	Yes	No	1/8 step
No	No	Yes	1/16 step
Yes	No	Yes	1/32 step
Yes	Ja	Yes	1/128 step

5. STEPPER MOTOR DRIVER

After the jumpers have been set, the stepper motor drivers can be inserted into the black slots, which are located directly at the jumper slots. Please take care to insert the stepper motor drivers correctly.

In the following picture you can see how to insert a motor driver on the board. After the successful insertion of the motor drivers, the corresponding stepper motors can now be connected.



6. END STOPS AND TEMPERATURE SENSORS

Now the end stops and the temperature sensors can be connected. Up to 6 end stops and 3 temperature sensors can be connected. The exact connection of the end stops can also be seen [here](#).

For the temperature sensors, the T0 connection is used for the HotEnd and the T1 connection for the heating bed. The polarity of the temperature sensors is not relevant.

7. SETTING UP THE MOTOR DRIVER

7.1 Different resistors

Please note that the motor drivers of the CNC kit must first be adapted to the motors to be operated before they can be used. The adaptation must be carried out in order not to cause any damage to the motors or the motor drivers themselves in the long run.

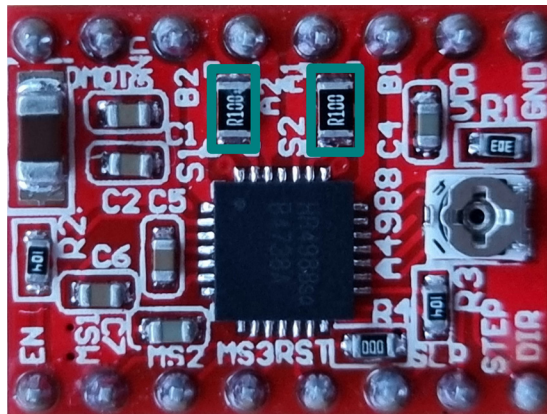
During this step, as shown in the following figure, note the resistors on the motor driver boards. These are necessary to set the correct value for the rotary potentiometer.

There are different variants of the A4988 motor drivers, so there are also different resistors with different values. The most common variants use the resistors **R050** (0.05 Ohm), **R100** (0.1 Ohm) and **R200** (0.2 Ohm).

7.2 Calculation of the value to be set

In order to be able to calculate to which mV value the rotary potentiometer must be set, we calculate:

$$V_{REF} = 8 * I_{MAX} * R_{CS}$$



V_{REF} (reference voltage) is the value that must be set on the rotary potentiometer.

rotary potentiometer.

I_{MAX} (rated current) can be found in the data sheets of the motors you want to use.

R_{CS} (Current Sense Resistor) is the value of the resistors marked in the picture above.

7.3 Measuring the value to be set

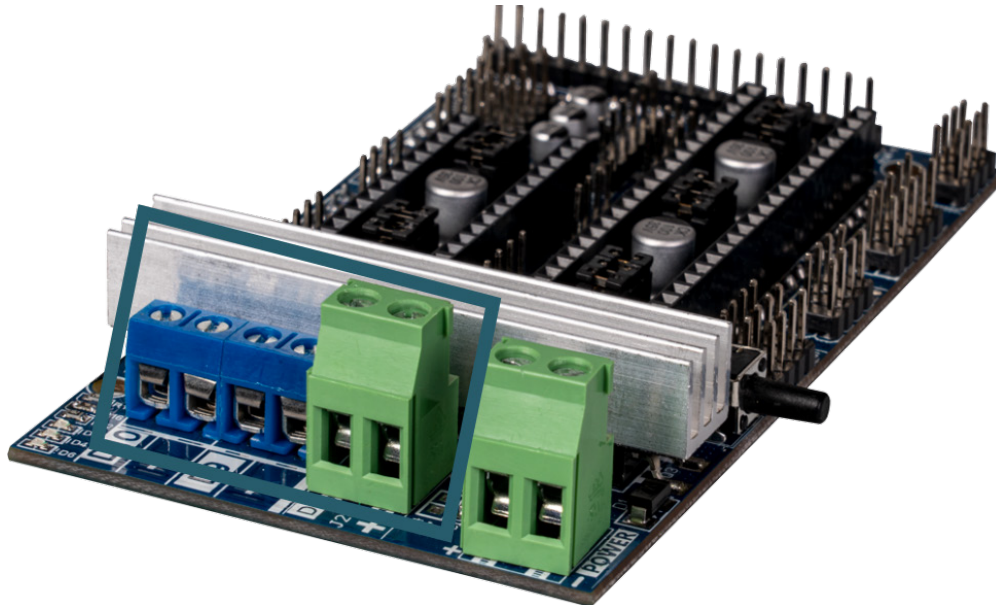
To now be able to set the rotary potentiometer correctly, you will need a multimeter, a cable with crocodile clips and a conductive screwdriver.

You must set the multimeter to the setting (**V_{DC} voltage**). Next, you need to attach the alligator clip cable with one side to the positive measuring tip of the multimeter and with the other side to the metal of the screwdriver. With the negative measuring tip of the multimeter, go to the nearest GND of the PCB. Now you can read on your multimeter how much voltage is coming to your motor driver.

To increase the voltage you must turn the screwdriver clockwise and to decrease the voltage you need to turn it counterclockwise.

8. POWER SUPPLY AND DISPLAY

In addition, the heating bed is now connected to port **D8** and the HotEnd to port **D10**. **D9** can be used either for a fan or for an additional HotEnd. The display can now also be connected to the slot provided for it.



Finally, the expansion board must now be supplied with power. Please note that your expansion board has two connectors. One for 11 amps and one for 5 amps. Both connections can be supplied with a maximum of 12 V.

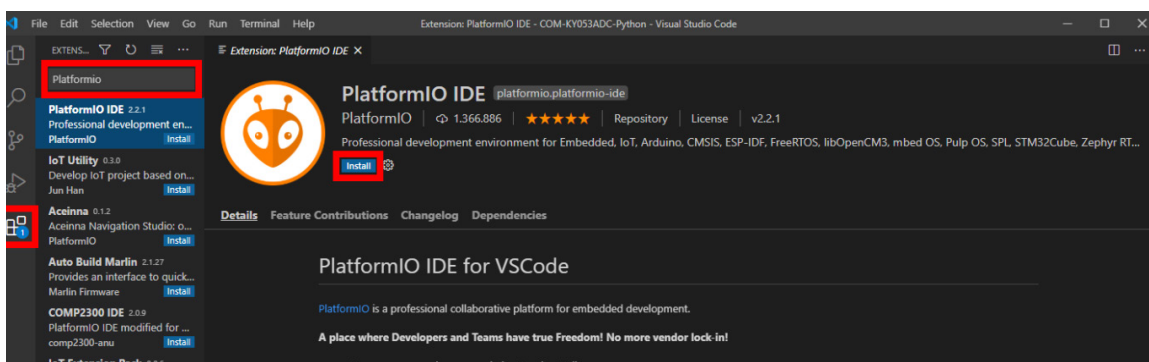
9. SETTING UP THE DEVELOPMENT ENVIRONMENT

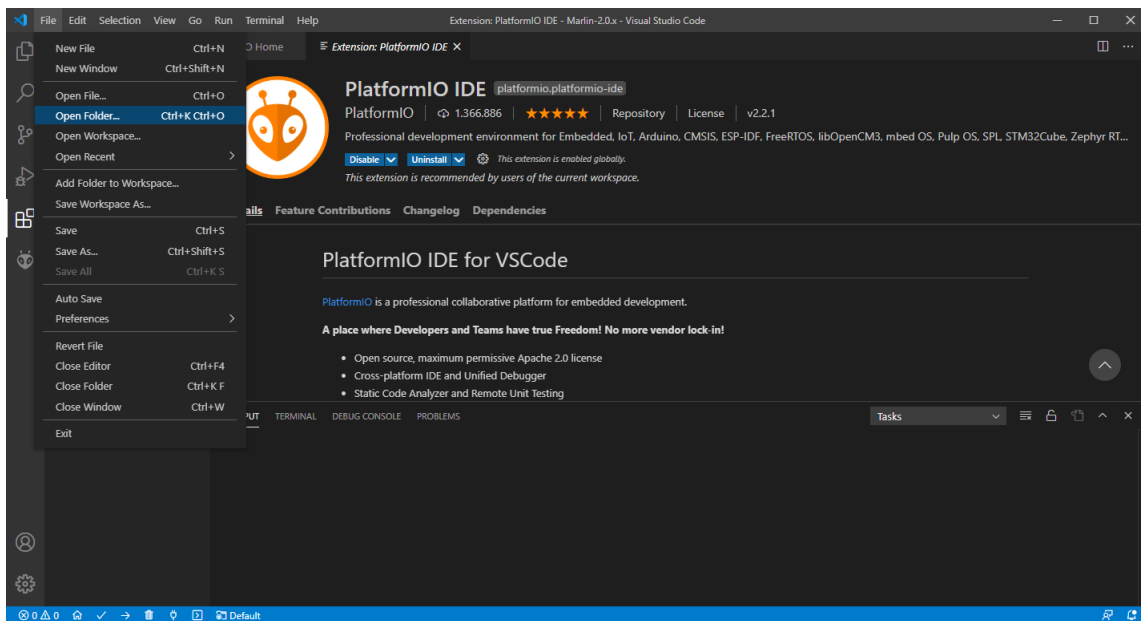
For use you need Visual Studio Code as well as the Marlin 3D Printer firmware.

First download and install the latest Visual Studio Code software [here](#).

Then, download the latest Marlin firmware [here](#).

Open Visual Studio Code and click on **Extensions** on the left. Now enter "**Platformio**" in the search bar and install the **PlatformIO IDE**.





Now you can open your previously downloaded Marlin firmware folder under **File -> Open Folder** and adjust the configuration.h file. To do this, click on the Explorer icon in the upper left corner and select the “configuration.h” file.

10. CONFIGURATION OF THE SOFTWARE

Board selection

Before the software can be transferred to the Arduino, it must first be adapted to the configuration used.

For the Ramps 1.6 board the following configurations are provided in the Marlin software:

```
#define BOARD_RAMPS_14_EFB 43 //RAMPS 1.4(Power outputs: Hotend,Fan,Bed)
#define BOARD_RAMPS_14_EEB 44 //RAMPS 1.4(Power outputs: Hotend0,Hotend1,Bed)
#define BOARD_RAMPS_14_EFF 45 //RAMPS 1.4(Power outputs: Hotend,Fan0,Fan1)
#define BOARD_RAMPS_14_EEF 46 //RAMPS 1.4(Power outputs: Hotend0,Hotend1,Fan)
#define BOARD_RAMPS_14_SF 48 //RAMPS 1.4(Power outputs: Spindle,Controller Fan)
```

The most common configuration consists of a HotEnd, a fan and a bed. However, this configuration may differ depending on your application and setup. The corresponding configuration must now be entered as a motherboard in the **Configuration.h** file.

To do this, open the **Configuration.h** file and search here for the corresponding motherboard definition. You can quickly search for the corresponding line by pressing **CTRL + F**. For the above mentioned configuration (HotEnd, Fan, Bed) the motherboard configuration should look like this:

```
// Choose the name from boards.h that matches your setup
#ifndef MOTHERBOARD
  #define MOTHERBOARD BOARD_RAMPS_14_EFB
#endif
```

Display

For the use of the display an additional configuration must be made. For this, also look for the following section in the Configuration.h file:

```
// RepRapDiscount FULL GRAPHIC Smart Controller
// https://reprap.org/wiki/RepRapDiscount_Full_Graphic_Smart_Controller
//
// #define REPRAP_DISCOUNT_FULL_GRAPHIC_SMART_CONTROLLER
```

To activate the display, remove the comment strokes from the last line so that the line looks like this:

```
#define REPRAP_DISCOUNT_FULL_GRAPHIC_SMART_CONTROLLER
```

If you have problems with the display on your screen, you can change this by changing the timings. To do this, insert the following lines:

```
#define ST7920_DELAY_1 DELAY_NS(0)
#define ST7920_DELAY_2 DELAY_NS(400)
#define ST7920_DELAY_3 DELAY_NS(0)
```

Number of extruders

If you use more than one extruder, the following line in the **Configuration.h** file must also be adapted accordingly:

```
// This defines the number of extruders
#define EXTRUDERS 3
```

Temperature sensors

A configuration of the temperature sensors may also be necessary, depending on the sensors used.

Please note the configuration notes in the following section of the **Configuration.h** file:

```
#define TEMP_SENSOR_0 1
#define TEMP_SENSOR_1 0
#define TEMP_SENSOR_2 0
#define TEMP_SENSOR_3 0
#define TEMP_SENSOR_4 0
#define TEMP_SENSOR_BED 0
```

The maximum permissible temperatures may also differ depending on the components used. Therefore, match the maximum temperatures in the following range to your components:

```
// Above this temperature the heater will be switched off.
// This can protect components from overheating, but NOT from shorts
// and failures.
// (Use MINTEMP for thermistor short/failure protection.)
#define HEATER_0_MAXTEMP 275
```

```
#define HEATER_1_MAXTEMP 275
#define HEATER_2_MAXTEMP 275
#define HEATER_3_MAXTEMP 275
#define HEATER_4_MAXTEMP 275
#define BED_MAXTEMP 150
```

Direction of rotation of the stepper motors

Depending on the wiring or polarity of the stepper motors used, it may be necessary to invert the direction of rotation of individual motors.

This is possible in the following section:

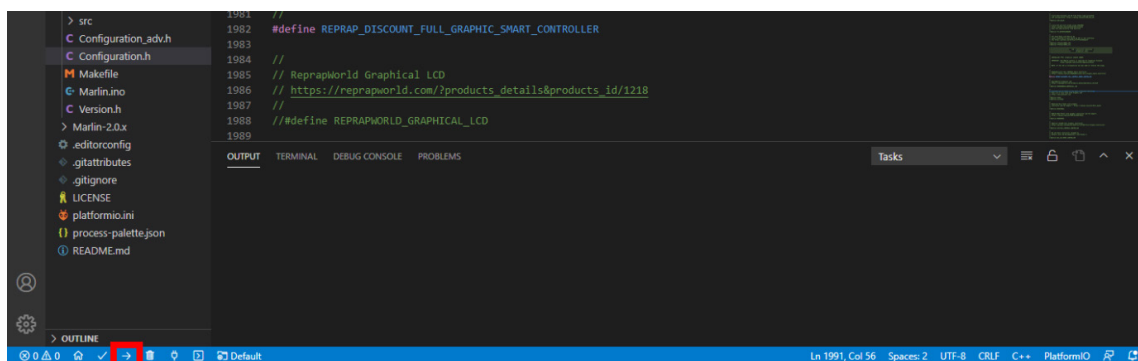
```
// Invert the stepper direction. Change (or reverse the motor connector) if
  an axis goes the wrong way.
#define INVERT_X_DIR false
#define INVERT_Y_DIR true
#define INVERT_Z_DIR false

// Enable this option for Toshiba stepper drivers
// #define CONFIG_STEPPERS_TOSHIBA

// @section extruder

// For direct drive extruder v9 set to true, for geared extruder set to false.
#define INVERT_E0_DIR false
#define INVERT_E1_DIR false
#define INVERT_E2_DIR false
#define INVERT_E3_DIR false
#define INVERT_E4_DIR false
```

When you have made all settings, you can transfer the program to your Arduino by clicking on PlatformIO: Upload in the bottom left corner.



11. INFORMATION & TAKE-BACK OBLIGATIONS

Our information and take-back obligations under the Electrical and Electronic Equipment Act (ElektroG).



Symbol on electrical and electronic equipment:

This crossed-out trash can means that electrical and electronic equipment does not belong in household trash. You must hand in the waste equipment at a collection point. Before dropping off, you must separate used batteries and accumulators that are not enclosed by the waste equipment from it.

Return options:

As an end user, when you purchase a new device, you may return your old device (which performs essentially the same function as the new device you purchased from us) for disposal at no charge. Small appliances with no external dimensions larger than 25 cm can be returned in household quantities regardless of the purchase of a new appliance.

Possibility of return at our company location during opening hours:

SIMAC Electronics GmbH, Pascalstr. 8, D-47506 Neukirchen-Vluyn, Germany.

Possibility of return in your area:

We will send you a parcel stamp with which you can return the device to us free of charge. To do so, please contact us by e-mail at Service@joy-it.net or by phone.

Packaging information:

Please pack your old device securely for transport, if you do not have suitable packaging material or do not wish to use your own contact us and we will send you suitable packaging.

12. SUPPORT

We are also there for you after the purchase. If any questions remain or problems arise, we are also available to assist you via email, phone and ticket support system.

E-Mail: service@joy-it.net

Ticket-System: <http://support.joy-it.net>

Phone: +49 (0)2845 9360 - 50 (10 am - 5 pm)

For more information, please visit our website:

www.joy-it.net