

# RAMPS 1.4 KIT

Ramps 1.4 with display & 5x A4988 motor driver



## 1. GENERAL INFORMATION

Dear customer,

thank you for choosing our product. In the following, we will show you how to use your product. Should you encounter any unexpected problems during use, please do not hesitate to contact us.

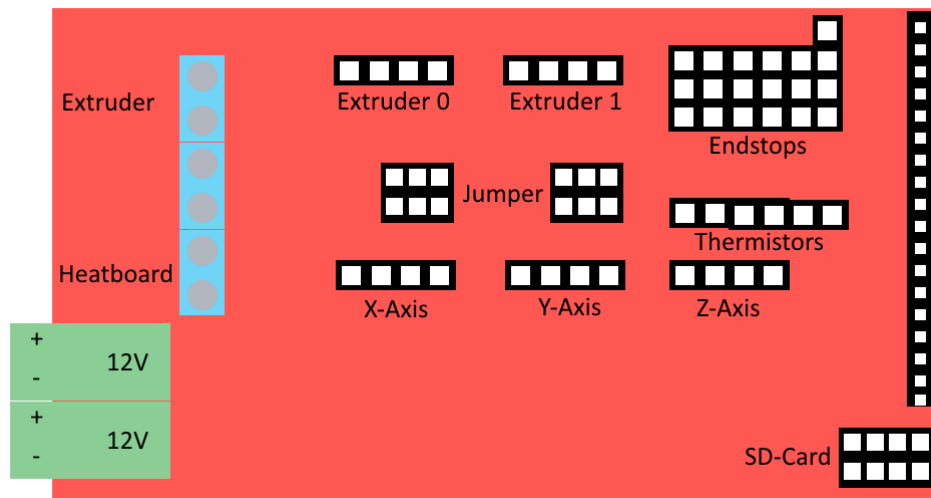
With this set you lay the foundation for the installation of your own 3D printer on the basis of Arduino.

Not only the required stepper motor output stages are included, but the board also offers connections for up to 3 heating units (printing bed heating, extruder 1, extruder 2) with each 3 possible inputs for connecting the required temperature sensors. In addition, fans for component and extruder cooling can also be connected.

With the matching, installed software on the Arduino MEGA, it can show you the current status or rather information like temperature on the included LCD-display.

## 2. SCHEMATIC DRAWING

The following schematic drawing shows the detailed pin assignment.

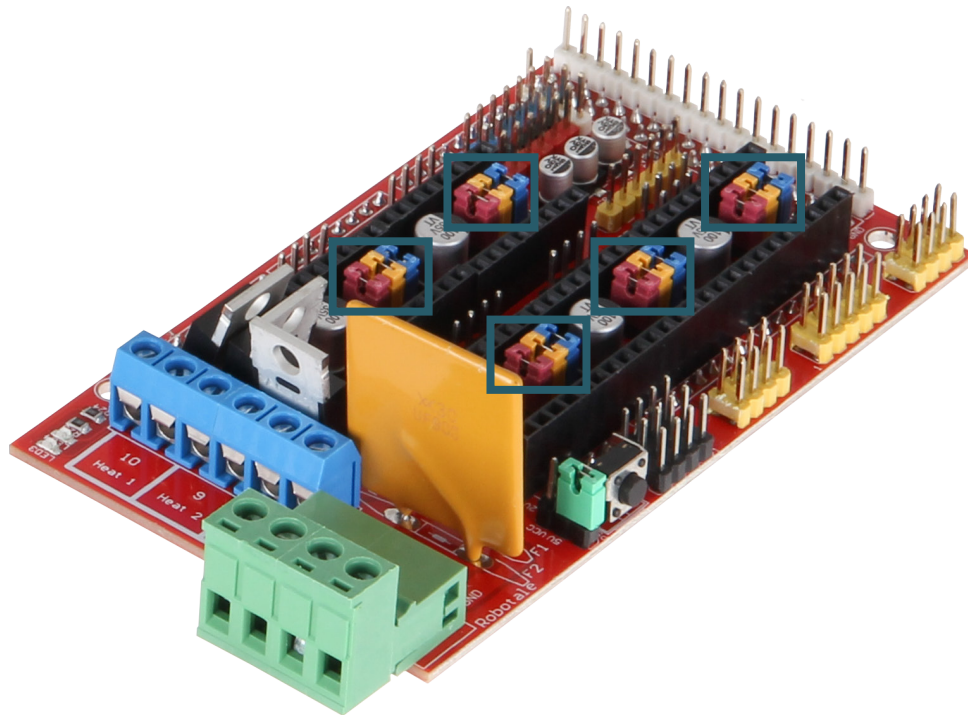


## 3. CONNECTION TO ARDUINO MEGA

First you have to connect the Arduino MEGA with the Ramps 1.4 expansion board. Plug the board on the Arduino so that the pins on the bottom of the boards are connected in accordance with the ports of the Arduino.

## 4. JUMPER-CONFIGURATION

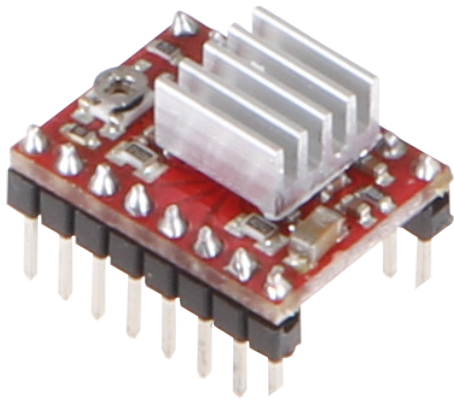
Before you can plug the necessary stepper motors on the expansion board, you must set the corresponding jumper first. The jumpers define the amount of possible microsteps for the stepper motors. It can support up to 1/128 steps. The exact configuration is additionally dependent of the utilized stepper motor drivers.



The different jumper slots are shown in the picture above and in the schematic drawing before. The various step configurations are displayed in the following chart.

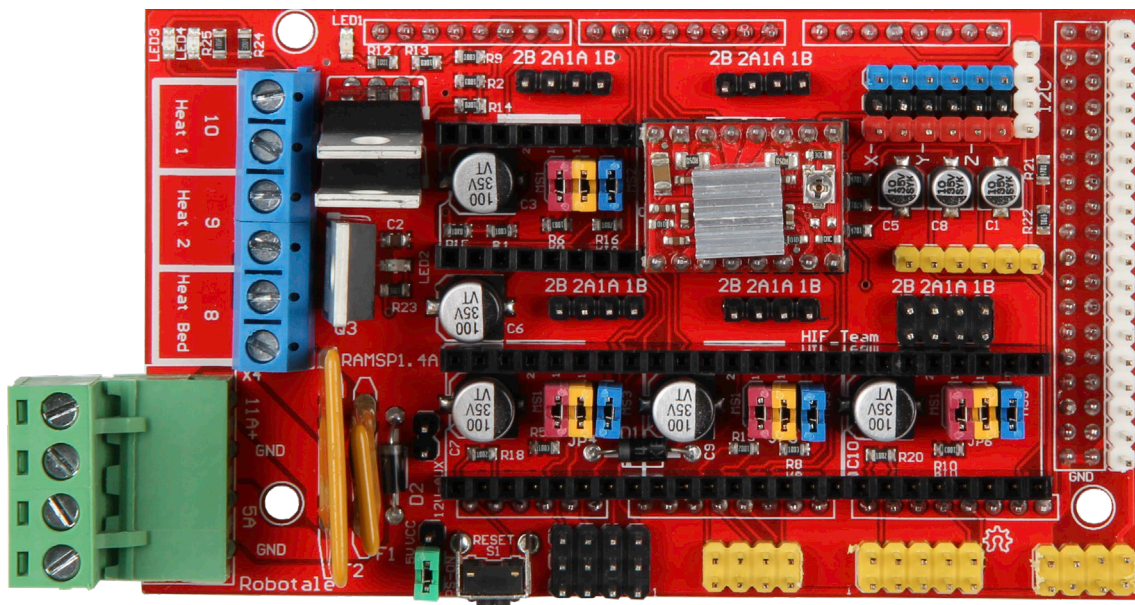
Jumper 1	Jumper 2	Jumper 3	Steps
No	No	No	full step
Yes	No	No	half step
No	Yes	Yes	1/4 step
Yes	Yes	No	1/8 step
No	No	Yes	1/16 step
Yes	No	Yes	1/32 step
Yes	Yes	Yes	1/128 step

## 5. STEPPER MOTOR DRIVER



After the jumpers are set, the stepper motor drivers have to be plugged into the black slots which are directly at the jumper slots. Make sure that the stepper motor driver are rightfully plugged in. You can gather the information from the following picture on how to plug in the motor drivers correctly.

If you have successfully inserted the motor drivers, you can connect the stepper motors.



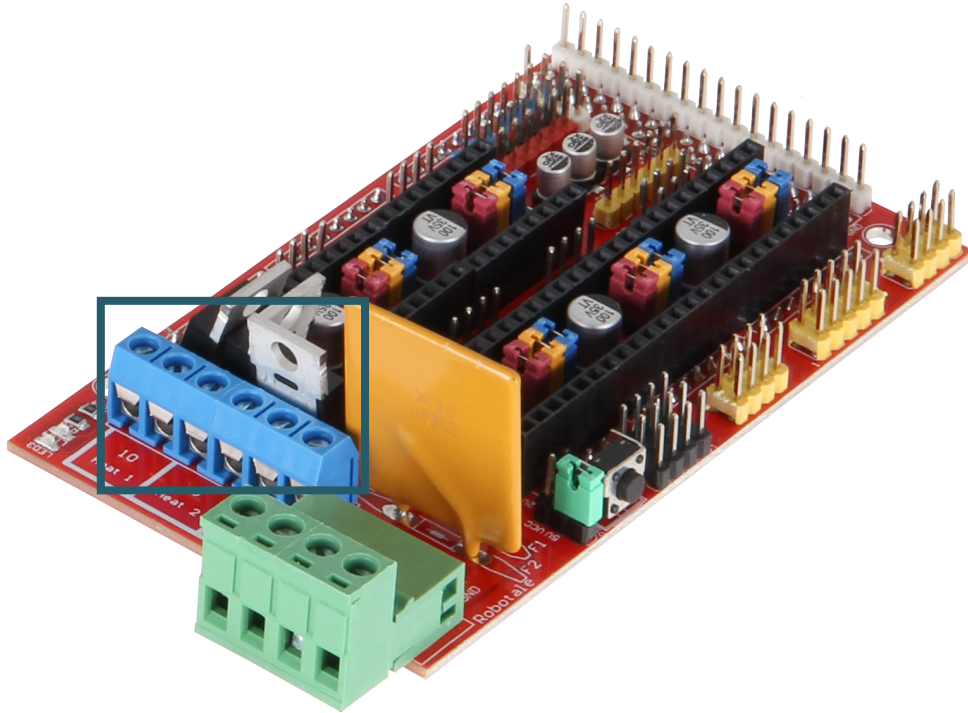
## 6. ENDSTOPS AND TEMPERATURE SENSORS

Now the endstops and the temperature sensors can be connected. Up to 6 endstops and 3 temperature sensors can be connected. You can see [here](#) the typical connection of the endstops. The temperature sensors get the connector **T0** for the HotEnd and the connector **T1** for the heat bed. The polarity of the temperature sensors are not relevant.

## 7. POWER SUPPLY AND DISPLAY

Additionally, the heatbed must be connected to the connector **D8** and the HotEnd to the connector **D10**. **D9** can be used for a fan or for an additional HotEnd. You can also now plug in the display into the provided slot.

Lastly, the expansion board only needs power to go into operation.



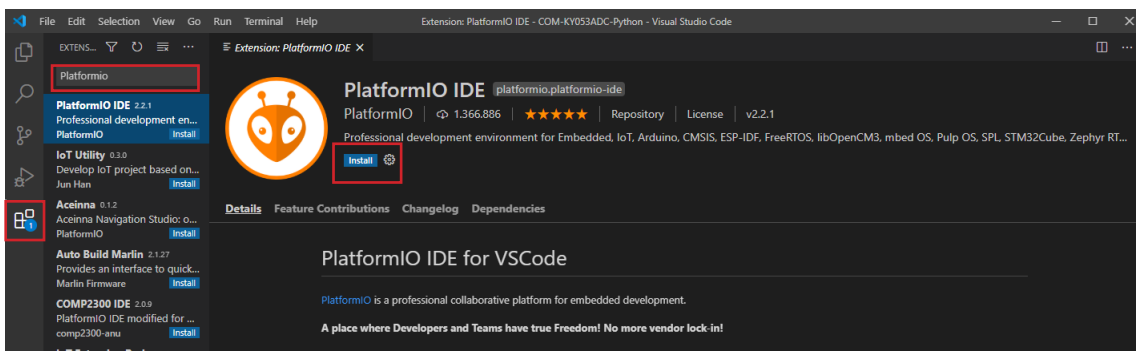
## 8. INSTALLATION OF DEVELOPMENT ENVIRONMENT

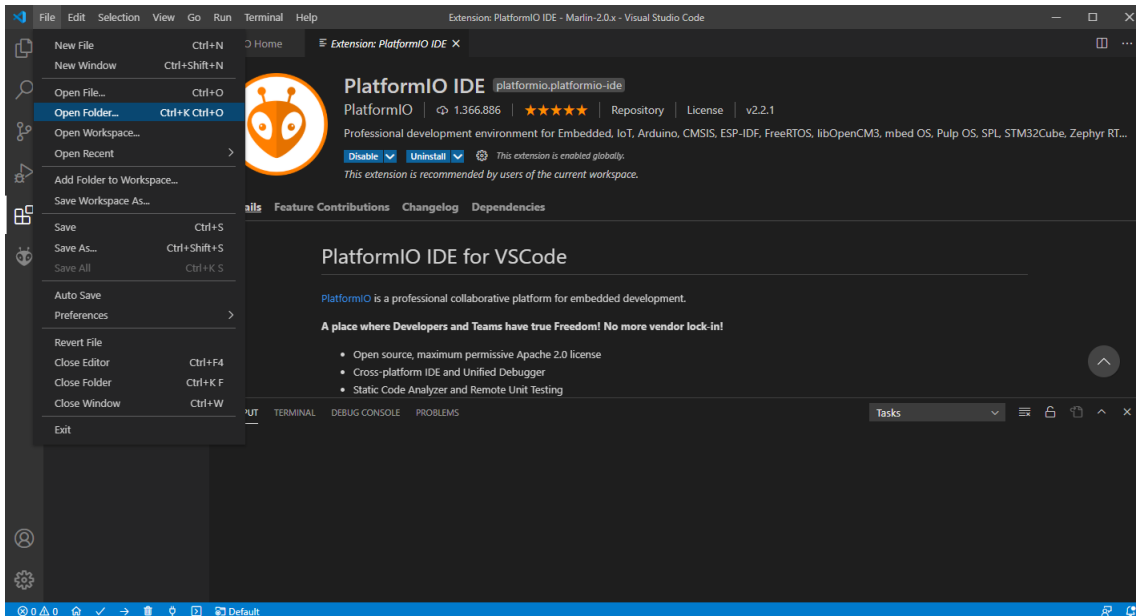
For the application is a development environment as well as the Marlin 3D firmware needed.

For that you first have to download the Visual Studio Code software [from their website](#) and install it.

Afterwards download the Marlin firmware [here](#).

Open Visual Studio Code and click **Extensions** on the left. Now type 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.

## 9. CONFIGURATION OF THE SOFTWARE

### Board selection

Before the software can be transmitted to the Arduino, the used configuration must be adjusted. For the Ramps 1.4 board is the following configuration intended:

```
#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 one HotEnd, one fan and one bed. Depending to its purpose and structure the configuration can be differed. In accordance with the configuration, the motherboard in the **Configuration.h** file must be registered.

Open the **Configuration.h** file and search for the corresponding definition of the motherboard. You can quickly search for the corresponding line by pressing **STR + F**.

For the earlier mentioned configuration (HotEnd, fan, bed), the motherboard configuration should look like the following example:

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

## **Display**

To activate the display, an additional configuration needs to be performed. Therefore, in the **Configuration.h** file, search for the following section:

```
// 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, delete the commentary lines in the last line so that it looks like that:

```
#define REPRAP_DISCOUNT_FULL_GRAPHIC_SMART_CONTROLLER
```

If you have problems with the image on your display, you can fix 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)
```

## **Amount of extruders**

If you use more than one extruder then you also need to adjust the following line in the **Configuration.h** file appropriately:

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

## **Temperature sensors**

Also the configuration of the temperature sensors must be adjusted in accordance with the used sensors.

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
```

Even the maximal permitted temperatures can deviate in accordance to the used components. Because of that, you should match the maximal temperatures with your components in the following section:

```
// If temperature exceeds max temp, your heater will be switched off.
// This feature exists to protect your hotend from overheating
// accidentally, but *NOT* from thermistor short/failure!
// You should us 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**

In accordance with the wiring or polarity of the used stepper motors, it can be necessary to invert the direction of rotation of individual motors.

This is possible in the following section:

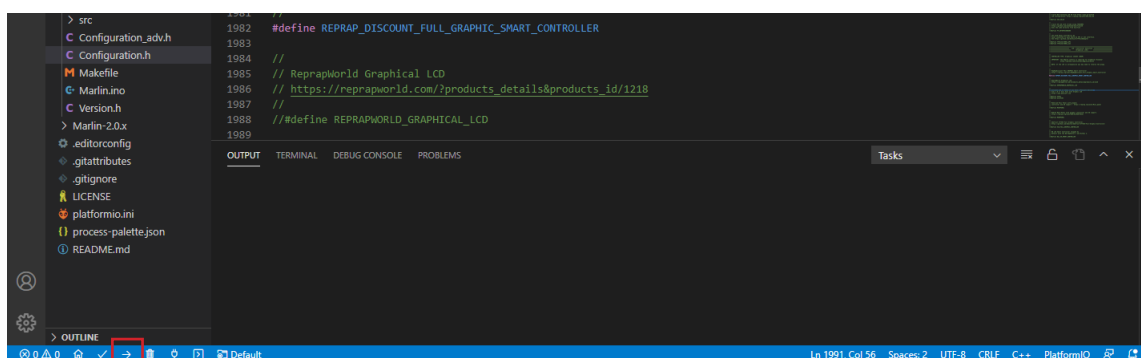
```
// Invert the stepper direction. Change (or reverse the motor connector)if
// an axis goes the wrong way.
#define INVERT_X_DIR false
#define INVERT_Y_DIR true
#define INVERT_Z_DIR false

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

// @section extruder

// For direct drive extruder v9 set to true, for geares 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.





## 10. FURTHER INFORMATION

Our information and take-back obligation according to the electrical and electronic Equipment Act (ElektroG)



### Symbol on electrical and electronic products :

This crossed-out bin means that electrical and electronic products do **not** belong into the household waste. You must hand over your old appliance to a registration office. Before you can hand over the old appliance, you must remove used batteries and accumulators which are not enclosed by the device.

### Return options :

As the end user, you can hand over with the purchase of a new device your old appliance (which has essentially the same functions as the new one) free of charge for disposal. Small devices which do not have outer dimensions greater than 25 cm can be submitted independently of the purchase of a new product in normal household quantities.

### Possibility of restitution at our company location during our opening hours :

Simac GmbH, Pascalstr. 8, D-47506 Neukirchen-Vluyn

### Possibility of restitution nearby :

We send you a parcel stamp with which you can send us your old appliance free of charge. For this possibility, you must contact us via e-mail at [service@joy-it.net](mailto:service@joy-it.net) or via telephone.

### Information about packaging:

Please package your old appliance safe during transport. Should you not have a suitable packaging material or you do not want to use your own material, you can contact us and we will send you an appropriate package.



## 11. SUPPORT

If any questions remain open or problems arise after your purchase, we are available by e-mail, telephone and with a ticket support system to answer these.

E-Mail: [service@joy-it.net](mailto:service@joy-it.net)

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

Telephone: +49 (0)2845 98469 – 66 (10 - 17 o'clock)

For further information visit our website:

[www.joy-it.net](http://www.joy-it.net)