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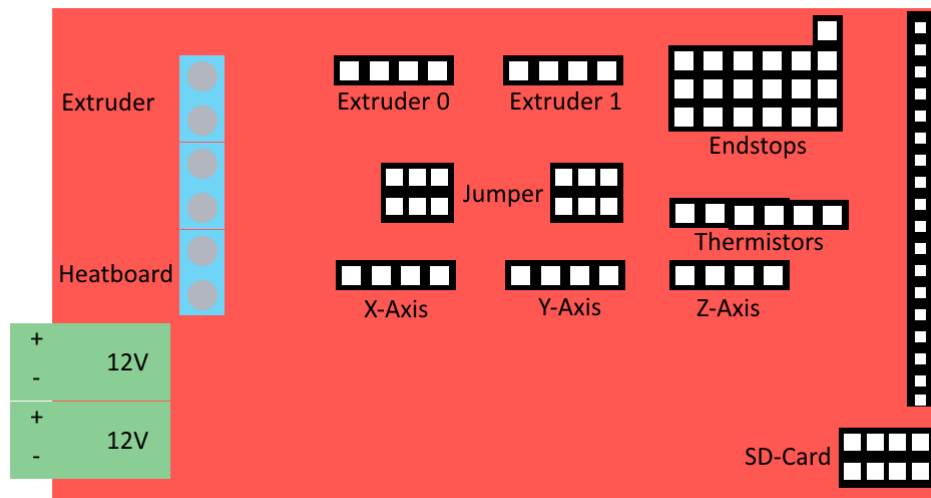
With this set you lay the foundation for the installation of your own 3D printer on the basis of Arduino.

Not only the required stepper motor output stages are included, but also the board 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 fitting 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

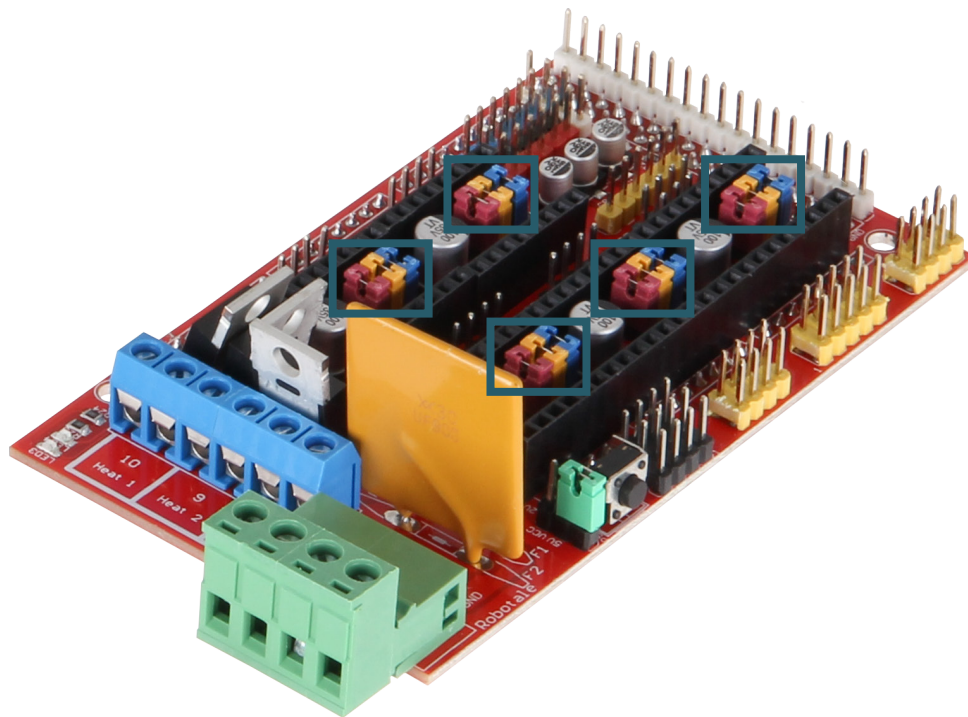The following schematic drawing shows the detailed pin assignment.

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 that way that the pins on the bottomside of the boards are connected in accordance with the ports of the Arduino.
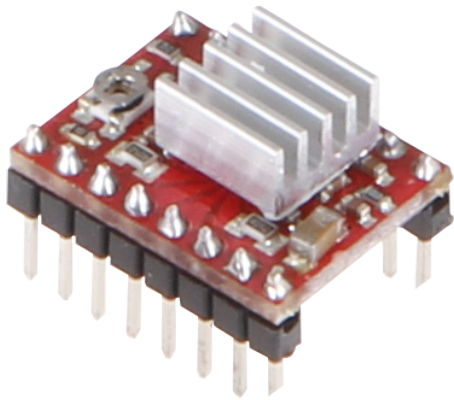
4. JUMPER-CONFIGURATION

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

The different jumper slots are shown in the picture above and in the schematic drawing before. The various step configuartion 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. Note that the stepper motor driver are rightfully plugged in. You can gather the information from the following picture how to plug in the motor drivers accurately.

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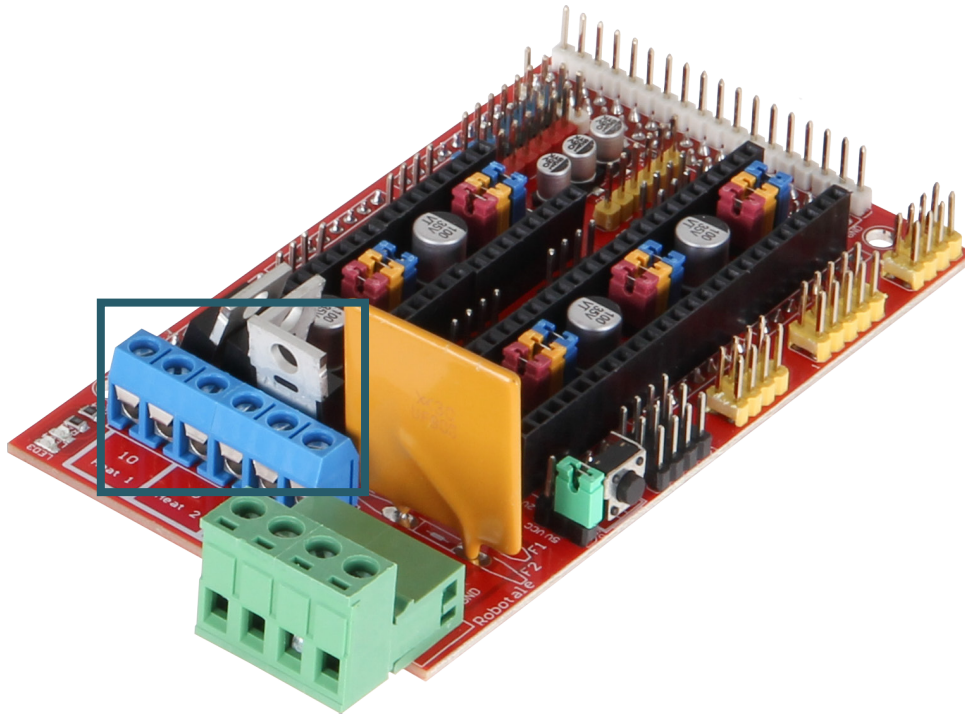

6. ENDSTOPS AND TEMPERATURE SENSORS

Now the endstops and the temperature sensores can be connected. There is a slot where up to 6 endstops and 3 temperature sensors are able to find their place. 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.

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

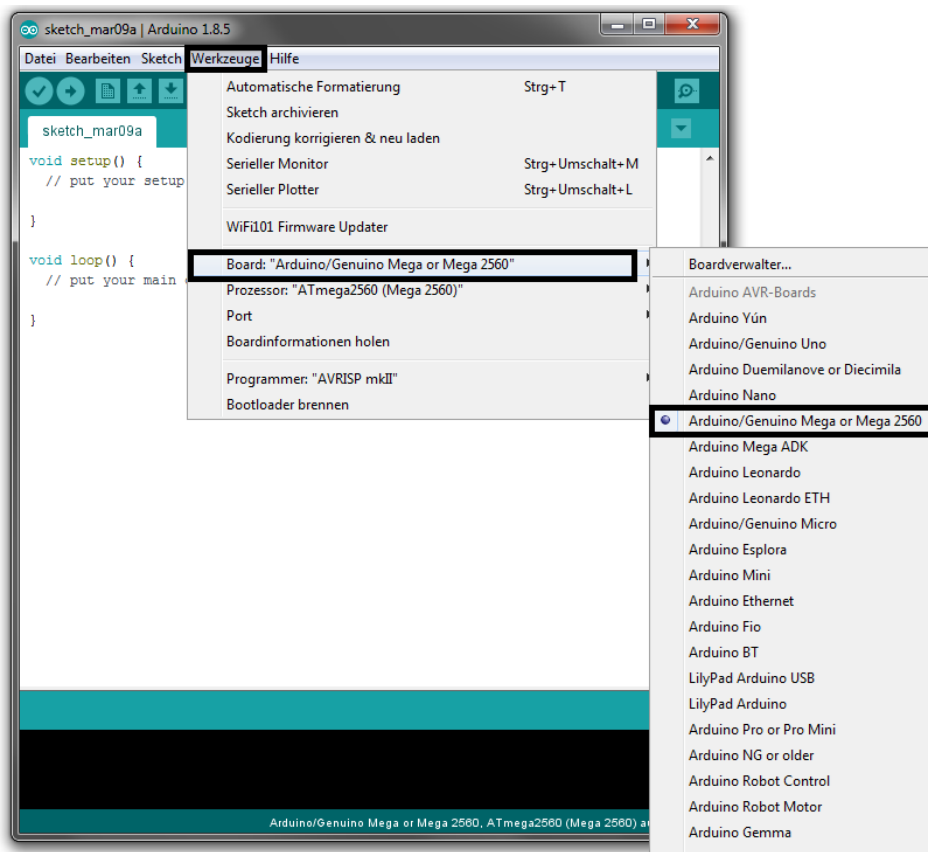For the application is a Arduino development environment as well as a Marlin 3D firmware needed.

For that you have to download first the Arduino software **from their website** and install it.
Afterwards download the Marlin firmware **from github**.
Open in the Arduino development environment the file "Marlin.ino" from the Marlin firmware folder.
The development environment will load the firmware and open all necessary files.

Change the selected board type to "Arduino Mega 2560". That way the transmisson of the firmware on the board will be enabled:

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 configuartion the motherboard in the **Configuration.h** file must be registered.

Open the **Configuration.h** file and search for the corresponding definition of the motherboard.
For the earlier mentioned configuration (HotEnd, fan, bed) should the motherboard configuartion look like the following:

```
// The following define selects which electronics board you have
// Please choose the name from boards.h that matches your setup
#ifndef MOTHERBOARD
      #define MOTHERBOARD BOARD_RAMPS_14_EFB
#endif
```

Display

For the application of the display must be an additional configuration performed. Search therefore in the **Configuration.h** file for the following section:

```
/**
 * LCD TYPE
 *
 * Enable ULTRA_LCD for a 16x2, 16x4, 20x2 or 20x4 character-based LCD.
 * Enable DOGLCD for a 128x64 (ST7565R) Full Graphical Display.
 * (These optionswill be enabled automatically for most displays.)
 *
 * IMPORTANT:The U8glib library is required for Full Graphic Display!
 *           https://github.com/olikraus/U8glib_Arduino
 */
// #define ULTRA_LCD   // Character based
// #define DOGLCD      // Full graphics display
```

To activate the display, delete the commentary lines in the last line that it looks like that:

```
#define DOGLCD    // Full graphics display
```

Moreover, for the application of the display is also the U8glib-library necessary. Download the library from **this link** and copy it into the Arduino library file
(C:\Users\\[**your user name**]\Documents\Arduino\libraries).

Should you use an extruder then you must also adjust the following line appropriatly:

```
// This defines the number of extruders
#define EXTRUDERS 1
```

Amount of extruders

Should you use more than one extruder then you must also adjust the following line in the **Configuration.h** file appropriatly:

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

temperature sensors

Even the configuartion 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 rather 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
```

10. FURTHER INFORMATION

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

Symbol on electrial 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 appli-ance 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 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
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

Published: 17.07.2019