

Welcome! [Log In](#) [Create A New Profile](#)



[Home](#) > [Electronics Folder](#) > [RAMPS Electronics](#) > Topic



[Advanced](#)

# Help with Marlin Config for MakerGear Prusa

Posted by [CSN](#)

<a href="#">CSN</a>	Registered: 1 year ago
<b><a href="#">Help with Marlin Config for MakerGear Prusa</a></b>	Posts: 21
April 12, 2012 05:07AM	

Need help with Marlin v1 configuration. Am a newbie struggling through. Bought a MakerGear Prusa Mendel kit in January with Stepper Plastruder, heated bed and nichrome/ceramic hotend. Decided to use Marlin, but have too many questions regarding configuration.h Any help would be appreciated. Ideally, I would love a copy of a configuration.h file from someone who has a MakerGear Prusa Mendel operating with it.

Am using a RAMPS 1.4 Assembled by Ultimachine( because I was not positive my kit one would work with my soldering skills so rusty) and assume I can use `#define MOTHERBOARD 33`  
 Looks like the thermistor they shipped with the Plastruder kit and heated bed are 100k (type 1?)  
 Using mechanical endstops, so I use true for `ENDSTOPS_INVERTING???`  
 do I use 0 or 1 for Inverting Stepper enable pins???  
 Lastly default settings  
`DEFAULT_AXIS_STEPS_PER_UNIT`, down through `DEFAULT_EJERK ????`  
 Thanks for any help

<a href="#">robertk925</a>	Registered: 1 year ago
<b><a href="#">Re: Help with Marlin Config for MakerGear Prusa</a></b>	Posts: 3
April 12, 2012 09:16PM	

Hey:

I also bought a MakerGear Prusa at the end of January and have really been enjoying it. I am currently using RC2 of Marlin firmware, one of the latest branches, so my configuration.h may have some additional options depending on which version you are using. Also, a few notes for things that may not match what you need:

1. I fine-tuned my hot-end heating using the M303 command, so I have added these lines to replace the standard MakerGear settings:  
`//Measured using M303 (Some Overshoot)`  
`#define DEFAULT_Kp 27.85`  
`#define DEFAULT_Ki 0.97`  
`#define DEFAULT_Kd 267.71`
2. I sped up the time required to wait and the temperature range for the hot-end temperature to stabilize because I am impatient:  
`// Actual temperature must be close to target for this long before M109 returns success`  
`#define TEMP_RESIDENCY_TIME 3 // (seconds)`

#define TEMP\_HYSTERESIS 5 // (degC) range of +/- temperatures considered "close" to the target one  
3. I have the X axis endstop in the maximum position (due to some wiring issues), but I have the Y and Z at the minimum positions:

```
// ENDSTOP SETTINGS:  
// Sets direction of endstops when homing; 1=MAX, -1=MIN  
#define X_HOME_DIR 1  
#define Y_HOME_DIR -1  
#define Z_HOME_DIR -1
```

Anyway, here is the configuration.h file:

```
#ifndef __CONFIGURATION_H  
#define __CONFIGURATION_H  
  
// This configuration file contains the basic settings.  
// Advanced settings can be found in Configuration_adv.h  
// BASIC SETTINGS: select your board type, temperature sensor type, axis scaling, and endstop configuration  
  
//User specified version info of THIS file to display in [Pronterface, etc] terminal window during startup.  
//Implementation of an idea by Prof Braino to inform user that any changes made  
//to THIS file by the user have been successfully uploaded into firmware.  
#define STRING_VERSION_CONFIG_H "2012-02-25" //Personal revision number for changes to THIS file.  
#define STRING_CONFIG_H_AUTHOR "erik" //Who made the changes.  
  
// This determines the communication speed of the printer  
#define BAUDRATE 250000  
// #define BAUDRATE 115200  
  
//// The following define selects which electronics board you have. Please choose the one that matches your  
setup  
// Gen7 custom (Alfons3 Version) = 10 "https://github.com/Alfons3/Generation_7_Electronics"  
// Gen7 v1.1, v1.2 = 11  
// Gen7 v1.3 = 12  
// Gen7 v1.4 = 13  
// MEGA/RAMPS up to 1.2 = 3  
// RAMPS 1.3 = 33 (Power outputs: Extruder, Bed, Fan)  
// RAMPS 1.3 = 34 (Power outputs: Extruder0, Extruder1, Bed)  
// Gen6 = 5  
// Gen6 deluxe = 51  
// Sanguinololu 1.2 and above = 62  
// Ultimaker = 7  
// Teensylu = 8  
// Gen3+ = 9  
  
#ifndef MOTHERBOARD
```

```
#define MOTHERBOARD 33
#endif

//=====
//=====Thermal Settings=====
//=====

//// Temperature sensor settings:
// -2 is thermocouple with MAX6675 (only for sensor 0)
// -1 is thermocouple with AD595
// 0 is not used
// 1 is 100k thermistor
// 2 is 200k thermistor
// 3 is mende-parts thermistor
// 4 is 10k thermistor !! do not use it for a hotend. It gives bad resolution at high temp. !!
// 5 is ParCan supplied 104GT-2 100K
// 6 is EPCOS 100k
// 7 is 100k Honeywell thermistor 135-104LAG-J01

#define TEMP_SENSOR_0 1
#define TEMP_SENSOR_1 1
#define TEMP_SENSOR_2 1
#define TEMP_SENSOR_BED 1

// Actual temperature must be close to target for this long before M109 returns success
#define TEMP_RESIDENCY_TIME 3 // (seconds)
#define TEMP_HYSTERESIS 5 // (degC) range of +/- temperatures considered "close" to the target one

// The minimal temperature defines the temperature below which the heater will not be enabled It is used
// to check that the wiring to the thermistor is not broken.
// Otherwise this would lead to the heater being powered on all the time.
#define HEATER_0_MINTEMP 5
#define HEATER_1_MINTEMP 5
#define HEATER_2_MINTEMP 5
#define BED_MINTEMP 5

// When 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 use MINTEMP for thermistor short/failure protection.
#define HEATER_0_MAXTEMP 255
#define HEATER_1_MAXTEMP 255
#define HEATER_2_MAXTEMP 255
#define BED_MAXTEMP 150
```

```

// PID settings:
// Comment the following line to disable PID and enable bang-bang.
#define PIDTEMP
#define PID_MAX 255 // limits current to nozzle; 255=full current
#ifndef PIDTEMP
  //#define PID_DEBUG // Sends debug data to the serial port.
  //#define PID_OPENLOOP 1 // Puts PID in open loop. M104 sets the output power in %
  #define PID_INTEGRAL_DRIVE_MAX 255 //limit for the integral term
  #define K1 0.95 //smoothing factor withing the PID
  #define PID_dT ((16.0 * 8.0)/(F_CPU / 64.0 / 256.0)) //sampling period of the

// If you are using a preconfigured hotend then you can use one of the value sets by uncommenting it
// Ultimaker
// #define DEFAULT_Kp 22.2
// #define DEFAULT_Ki 1.08
// #define DEFAULT_Kd 114

// Makergear
// #define DEFAULT_Kp 7.0
// #define DEFAULT_Ki 0.1
// #define DEFAULT_Kd 12

//Measured using M303 (Some Overshoot)
#define DEFAULT_Kp 27.85
#define DEFAULT_Ki 0.97
#define DEFAULT_Kd 267.71

// Mendel Parts V9 on 12V
// #define DEFAULT_Kp 63.0
// #define DEFAULT_Ki 2.25
// #define DEFAULT_Kd 440
#endif // PIDTEMP

//this prevents dangerous Extruder moves, i.e. if the temperature is under the limit
//can be software-disabled for whatever purposes by
#define PREVENT_DANGEROUS_EXTRUDE
#define EXTRUDE_MINTEMP 150
#define EXTRUDE_MAXLENGTH (X_MAX_LENGTH+Y_MAX_LENGTH) //prevent extrusion of very
large distances.

//=====
//=====Mechanical Settings=====
//=====

```

```
// Endstop Settings
#define ENDSTOPPULLUPS // Comment this out (using // at the start of the line) to disable the endstop pullup
resistors

// The pullups are needed if you directly connect a mechanical endswitch between the signal and ground pins.
const bool X_ENDSTOPS_INVERTING = false; // set to true to invert the logic of the endstops.
const bool Y_ENDSTOPS_INVERTING = false; // set to true to invert the logic of the endstops.
const bool Z_ENDSTOPS_INVERTING = false; // set to true to invert the logic of the endstops.
#define DISABLE_MAX_ENDSTOPS

// For Inverting Stepper Enable Pins (Active Low) use 0, Non Inverting (Active High) use 1
#define X_ENABLE_ON 0
#define Y_ENABLE_ON 0
#define Z_ENABLE_ON 0
#define E_ENABLE_ON 0 // For all extruders

// Disables axis when it's not being used.
#define DISABLE_X false
#define DISABLE_Y false
#define DISABLE_Z false
#define DISABLE_E false // For all extruders

#define INVERT_X_DIR true // for Mendel set to false, for Orca set to true
#define INVERT_Y_DIR true // for Mendel set to true, for Orca set to false
#define INVERT_Z_DIR true // for Mendel set to false, for Orca set to true
#define INVERT_E0_DIR false // for direct drive extruder v9 set to true, for geared extruder set to false
#define INVERT_E1_DIR false // for direct drive extruder v9 set to true, for geared extruder set to false
#define INVERT_E2_DIR false // for direct drive extruder v9 set to true, for geared extruder set to false

// ENDSTOP SETTINGS:
// Sets direction of endstops when homing; 1=MAX, -1=MIN
#define X_HOME_DIR 1
#define Y_HOME_DIR -1
#define Z_HOME_DIR -1

#define min_software_endstops true //If true, axis won't move to coordinates less than HOME_POS.
#define max_software_endstops true //If true, axis won't move to coordinates greater than the defined lengths
below.
#define X_MAX_LENGTH 180
#define Y_MAX_LENGTH 180
#define Z_MAX_LENGTH 100

// The position of the homing switches. Use MAX_LENGTH * -0.5 if the center should be 0, 0, 0
#define X_HOME_POS 0
#define Y_HOME_POS 0
```

```

#define Z_HOME_POS 0

//// MOVEMENT SETTINGS
#define NUM_AXIS 4 // The axis order in all axis related arrays is X, Y, Z, E
#define HOMING_FEEDRATE {50*60, 50*60, 4*60, 0} // set the homing speeds (mm/min)

// default settings

//#define DEFAULT_AXIS_STEPS_PER_UNIT {44.444, 44.444, 1814.173, 1487/1.1} // For GT2-
XY/Acme-Z/MakerGear Extruder
#define DEFAULT_AXIS_STEPS_PER_UNIT {52.4934, 52.4934, 3200/1.25, 1487/1.1} // For MakerGear
#define DEFAULT_MAX_FEEDRATE {500, 500, 5, 45} // (mm/sec)
#define DEFAULT_MAX_ACCELERATION {900,900,80,10000} // X, Y, Z, E maximum start speed for
accelerated moves. E default values are good for skeinforge 40+, for older versions raise them a lot.

#define DEFAULT_ACCELERATION 3000 // X, Y, Z and E max acceleration in mm/s^2 for printing moves
#define DEFAULT_RETRACT_ACCELERATION 3000 // X, Y, Z and E max acceleration in mm/s^2 for r
retracts

//
#define DEFAULT_XYJERK 20.0 // (mm/sec)
#define DEFAULT_ZJERK 0.4 // (mm/sec)
#define DEFAULT_EJERK 5.0 // (mm/sec)

//=====
//=====Additional Features=====
//=====

// EEPROM
// the microcontroller can store settings in the EEPROM, e.g. max velocity...
// M500 - stores paramters in EEPROM
// M501 - reads parameters from EEPROM (if you need reset them after you changed them temporarily).
// M502 - reverts to the default "factory settings". You still need to store them in EEPROM afterwards if you
want to.
//define this to enable eeprom support
//#define EEPROM_SETTINGS
//to disable EEPROM Serial responses and decrease program space by ~1700 byte: comment this out:
// please keep turned on if you can.
//#define EEPROM_CHITCHAT

//LCD and SD support
//#define ULTRA_LCD //general lcd support, also 16x2
//#define SDSUPPORT // Enable SD Card Support in Hardware Console

//#define ULTIPANEL

```

```

#ifndef ULTIPANEL
// #define NEWPANEL //enable this if you have a click-encoder panel
#define SDSUPPORT
#define ULTRA_LCD
#define LCD_WIDTH 20
#define LCD_HEIGHT 4

// Preheat Constants
#define PLA_PREHEAT_HOTEND_TEMP 180
#define PLA_PREHEAT_HPB_TEMP 70
#define PLA_PREHEAT_FAN_SPEED 255 // Insert Value between 0 and 255

#define ABS_PREHEAT_HOTEND_TEMP 240
#define ABS_PREHEAT_HPB_TEMP 100
#define ABS_PREHEAT_FAN_SPEED 255 // Insert Value between 0 and 255

#else //no panel but just lcd
#ifndef ULTRA_LCD
#define LCD_WIDTH 16
#define LCD_HEIGHT 2
#endif
#endif

// M240 Triggers a camera by emulating a Canon RC-1 Remote
// Data from: [www.doc-diy.net]
// #define PHOTOGRAPH_PIN 23

#include "Configuration_adv.h"
#include "thermistortables.h"

#endif // __CONFIGURATION_H

```

CSN

**Re: Help with Marlin Config for Makergear Prusa**

April 16, 2012 05:43PM

Registered: 1 year ago

Posts: 21

Many thanks robertk925. Making progress. Modified your code slightly, compiled, uploaded and started Pronterface for Mac. Downloaded the precompiled Mac version of printrun at [kotikapsi.fi]. I assume that means I do not have to install python at all??? Screen shot shows at least one problem. I cannot seem to get the hot end to heat or the bed to heat. Thermistors are connected as are heaters, but the heaters both show off. I thought setting the temperatures would turn them on, but they do not get warm. Any suggestions?

My modified configuration.h is:

```

#ifndef __CONFIGURATION_H
#define __CONFIGURATION_H

```

```
// This configuration file contains the basic settings.
// Advanced settings can be found in Configuration_adv.h
// BASIC SETTINGS: select your board type, temperature sensor type, axis scaling, and endstop configuration

//User specified version info of THIS file to display in [Pronterface, etc] terminal window during startup.
//Implementation of an idea by Prof Braino to inform user that any changes made
//to THIS file by the user have been successfully uploaded into firmware.
#define STRING_VERSION_CONFIG_H "2012-02-25" //Personal revision number for changes to THIS file.
#define STRING_CONFIG_H_AUTHOR "erik" //Who made the changes.

// This determines the communication speed of the printer
// #define BAUDRATE 250000
#define BAUDRATE 115200

//// The following define selects which electronics board you have. Please choose the one that matches your
setup
// Gen7 custom (Alfons3 Version) = 10 "https://github.com/Alfons3/Generation_7_Electronics"
// Gen7 v1.1, v1.2 = 11
// Gen7 v1.3 = 12
// Gen7 v1.4 = 13
// MEGA/RAMPS up to 1.2 = 3
// RAMPS 1.3 = 33 (Power outputs: Extruder, Bed, Fan)
// RAMPS 1.3 = 34 (Power outputs: Extruder0, Extruder1, Bed)
// Gen6 = 5
// Gen6 deluxe = 51
// Sanguinololu 1.2 and above = 62
// Ultimaker = 7
// Teensylu = 8
// Gen3+ = 9

#ifndef MOTHERBOARD
#define MOTHERBOARD 33
#endif

//=====
//=====Thermal Settings=====
//=====

//// Temperature sensor settings:
// -2 is thermocouple with MAX6675 (only for sensor 0)
// -1 is thermocouple with AD595
// 0 is not used
// 1 is 100k thermistor
// 2 is 200k thermistor
// 3 is mendel-parts thermistor
```



```
// 4 is 10k thermistor !! do not use it for a hotend. It gives bad resolution at high temp. !!
// 5 is ParCan supplied 104GT-2 100K
// 6 is EPCOS 100k
// 7 is 100k Honeywell thermistor 135-104LAG-J01

#define TEMP_SENSOR_0 1
#define TEMP_SENSOR_1 0
#define TEMP_SENSOR_2 0
#define TEMP_SENSOR_BED 1

// Actual temperature must be close to target for this long before M109 returns success
#define TEMP_RESIDENCY_TIME 10 // (seconds)
#define TEMP_HYSTERESIS 5 // (degC) range of +/- temperatures considered "close" to the target one

// The minimal temperature defines the temperature below which the heater will not be enabled It is used
// to check that the wiring to the thermistor is not broken.
// Otherwise this would lead to the heater being powered on all the time.
#define HEATER_0_MINTEMP 5
#define HEATER_1_MINTEMP 5
#define HEATER_2_MINTEMP 5
#define BED_MINTEMP 5

// When 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 use MINTEMP for thermistor short/failure protection.
#define HEATER_0_MAXTEMP 255
#define HEATER_1_MAXTEMP 255
#define HEATER_2_MAXTEMP 255
#define BED_MAXTEMP 150

// PID settings:
// Comment the following line to disable PID and enable bang-bang.
#define PIDTEMP
#define PID_MAX 255 // limits current to nozzle; 255=full current
#ifndef PIDTEMP
  // #define PID_DEBUG // Sends debug data to the serial port.
  // #define PID_OPENLOOP 1 // Puts PID in open loop. M104 sets the output power in %
  #define PID_INTEGRAL_DRIVE_MAX 255 //limit for the integral term
  #define K1 0.95 //smoothing factor withing the PID
  #define PID_dT ((16.0 * 8.0)/(F_CPU / 64.0 / 256.0)) //sampling period of the

// If you are using a preconfigured hotend then you can use one of the value sets by uncommenting it
// Ultimaker
```

```
// #define DEFAULT_Kp 22.2
// #define DEFAULT_Ki 1.08
// #define DEFAULT_Kd 114

// Makergear
#define DEFAULT_Kp 7.0
#define DEFAULT_Ki 0.1
#define DEFAULT_Kd 12

// Mendel Parts V9 on 12V
// #define DEFAULT_Kp 63.0
// #define DEFAULT_Ki 2.25
// #define DEFAULT_Kd 440
#endif // PIDTEMP

//this prevents dangerous Extruder moves, i.e. if the temperature is under the limit
//can be software-disabled for whatever purposes by
#define PREVENT_DANGEROUS_EXTRUDE
#define EXTRUDE_MINTEMP 150
#define EXTRUDE_MAXLENGTH (X_MAX_LENGTH+Y_MAX_LENGTH) //prevent extrusion of very
large distances.

//=====
//=====Mechanical Settings=====
//=====

// Endstop Settings
#define ENDSTOPPULLUPS // Comment this out (using // at the start of the line) to disable the endstop pullup
resistors

// The pullups are needed if you directly connect a mechanical endswitch between the signal and ground pins.
const bool X_ENDSTOPS_INVERTING = false; // set to true to invert the logic of the endstops.
const bool Y_ENDSTOPS_INVERTING = false; // set to true to invert the logic of the endstops.
const bool Z_ENDSTOPS_INVERTING = false; // set to true to invert the logic of the endstops.
#define DISABLE_MAX_ENDSTOPS

// For Inverting Stepper Enable Pins (Active Low) use 0, Non Inverting (Active High) use 1
#define X_ENABLE_ON 0
#define Y_ENABLE_ON 0
#define Z_ENABLE_ON 0
#define E_ENABLE_ON 0 // For all extruders

// Disables axis when it's not being used.
#define DISABLE_X false
#define DISABLE_Y false
```

```
#define DISABLE_Z false
#define DISABLE_E false // For all extruders

#define INVERT_X_DIR true // for Mendel set to false, for Orca set to true
#define INVERT_Y_DIR true // for Mendel set to true, for Orca set to false
#define INVERT_Z_DIR true // for Mendel set to false, for Orca set to true
#define INVERT_E0_DIR false // for direct drive extruder v9 set to true, for geared extruder set to false
#define INVERT_E1_DIR false // for direct drive extruder v9 set to true, for geared extruder set to false
#define INVERT_E2_DIR false // for direct drive extruder v9 set to true, for geared extruder set to false

// ENDSTOP SETTINGS:
// Sets direction of endstops when homing; 1=MAX, -1=MIN
#define X_HOME_DIR -1
#define Y_HOME_DIR -1
#define Z_HOME_DIR -1

#define min_software_endstops true //If true, axis won't move to coordinates less than HOME_POS.
#define max_software_endstops true //If true, axis won't move to coordinates greater than the defined lengths
below.
#define X_MAX_LENGTH 200
#define Y_MAX_LENGTH 200
#define Z_MAX_LENGTH 180

// The position of the homing switches. Use MAX_LENGTH * -0.5 if the center should be 0, 0, 0
#define X_HOME_POS 0
#define Y_HOME_POS 0
#define Z_HOME_POS 0

//// MOVEMENT SETTINGS
#define NUM_AXIS 4 // The axis order in all axis related arrays is X, Y, Z, E
#define HOMING_FEEDRATE {50*60, 50*60, 4*60, 0} // set the homing speeds (mm/min)

// default settings

#define DEFAULT_AXIS_STEPS_PER_UNIT {52.4934,52.4934,1814.173, 1487/1.1} // default steps per
unit for ultimaker
#define DEFAULT_MAX_FEEDRATE {500, 500, 5, 45} // (mm/sec)
#define DEFAULT_MAX_ACCELERATION {900,900,80,10000} // X, Y, Z, E maximum start speed for
accelerated moves. E default values are good for skeinforge 40+, for older versions raise them a lot.

#define DEFAULT_ACCELERATION 3000 // X, Y, Z and E max acceleration in mm/s^2 for printing moves
#define DEFAULT_RETRACT_ACCELERATION 3000 // X, Y, Z and E max acceleration in mm/s^2 for r
etracts

//
```

```
#define DEFAULT_XYJERK 20.0 // (mm/sec)
#define DEFAULT_ZJERK 0.4 // (mm/sec)
#define DEFAULT_EJERK 5.0 // (mm/sec)

//=====
//=====Additional Features=====
//=====

// EEPROM
// the microcontroller can store settings in the EEPROM, e.g. max velocity...
// M500 - stores paramters in EEPROM
// M501 - reads parameters from EEPROM (if you need reset them after you changed them temporarily).
// M502 - reverts to the default "factory settings". You still need to store them in EEPROM afterwards if you
want to.
//define this to enable eeprom support
//#define EEPROM_SETTINGS
//to disable EEPROM Serial responses and decrease program space by ~1700 byte: comment this out:
// please keep turned on if you can.
//#define EEPROM_CHITCHAT

//LCD and SD support
//#define ULTRA_LCD //general lcd support, also 16x2
//#define SDSUPPORT // Enable SD Card Support in Hardware Console

//#define ULTIPANEL
#ifndef ULTIPANEL
// #define NEWPANEL //enable this if you have a click-encoder panel
#define SDSUPPORT
#define ULTRA_LCD
#define LCD_WIDTH 20
#define LCD_HEIGHT 4

// Preheat Constants
#define PLA_PREHEAT_HOTEND_TEMP 180
#define PLA_PREHEAT_HPB_TEMP 70
#define PLA_PREHEAT_FAN_SPEED 255 // Insert Value between 0 and 255

#define ABS_PREHEAT_HOTEND_TEMP 240
#define ABS_PREHEAT_HPB_TEMP 100
#define ABS_PREHEAT_FAN_SPEED 255 // Insert Value between 0 and 255

#else //no panel but just lcd
#ifndef ULTRA_LCD
#define LCD_WIDTH 16
#define LCD_HEIGHT 2
```

```
#endif
#endif

// M240 Triggers a camera by emulating a Canon RC-1 Remote
// Data from: [www.doc-diy.net]
// #define PHOTOGRAPH_PIN 23

#include "Configuration_adv.h"
#include "thermistortables.h"

#endif//__CONFIGURATION_H
```

Edited 1 time(s). Last edit at 04/16/2012 06:13PM by CSN.

**robertk925**

Registered: 1 year ago

**Re: Help with Marlin Config for Makergear Prusa**

Posts: 3

April 16, 2012 09:43PM

I think the precompiled versions of Pronterface already have python included.

Looking at your screenshot the temperatures shown looks like the thermistor(s) are not plugged in correctly. They should show at least the ambient temperature of the room, not 0 degrees (unless your room is really cold!) Are they plugged in to the correct pins on the Ramps PCB?

**CSN**

Registered: 1 year ago

**Re: Help with Marlin Config for Makergear Prusa**

Posts: 21

April 17, 2012 01:06PM

robertl925,

They were plugged in correctly, but found a bad connector (molex?) on the hotend thermistor. Took out the connector entirely and that was it. Thanks for helping locate the problem. I can now control all 4 motors, two heaters and extrude. Many thanks.

Robert

This [forum](#) is powered by [Phorum](#).