

```
#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
//////////基本设置包括： 主板类型， 温度传感器类型， 轴设置， 限位开关配置

//=====
//===== DELTA Printer =====
//=====
// For a Delta printer replace the configuration files with the files in the
// example_configurations/delta directory.
//
//////////对于三角洲并联打印机， 请找到固件里“example_configurations”文件夹下的“delta”
子文件夹， 并将其中的 Configuration.h 和 Configuration_adv.h 两个头文件拷贝到固件文件夹
下， 替换原先固件里的这两个头文件,并针对 delta 打印机作相应参数配置。

// User-specified version info of this build 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
// build by the user have been successfully uploaded into firmware.
#define STRING_VERSION_CONFIG_H __DATE__ " " __TIME__ // build date and time
#define STRING_CONFIG_H_AUTHOR "(none, default config)" // Who made the changes.

// SERIAL_PORT selects which serial port should be used for communication with the host.
// This allows the connection of wireless adapters (for instance) to non-default port pins.
// Serial port 0 is still used by the Arduino bootloader regardless of this setting.
#define SERIAL_PORT 0
//////////选择用于和上位机通讯的串口， 该值请保持 0 不动

// This determines the communication speed of the printer
// This determines the communication speed of the printer
#define BAUDRATE 250000
//////////波特率配置， 该值必须和你的上位机波特率相匹配， 否则无法通讯。常用波特率有
115200 和 250000

// This enables the serial port associated to the Bluetooth interface
// #define BTENABLED // Enable BT interface on AT90USB devices

//// The following define selects which electronics board you have. Please choose the one that
matches your setup
```

```
// 10 = Gen7 custom (Alfons3 Version) "https://github.com/Alfons3/Generation_7_Electronics"
// 11 = Gen7 v1.1, v1.2 = 11
// 12 = Gen7 v1.3
// 13 = Gen7 v1.4
// 2 = Cheaptronic v1.0
// 20 = Sethi 3D_1
// 3 = MEGA/RAMPS up to 1.2 = 3
// 33 = RAMPS 1.3 / 1.4 (Power outputs: Extruder, Fan, Bed)
// 34 = RAMPS 1.3 / 1.4 (Power outputs: Extruder0, Extruder1, Bed)
// 35 = RAMPS 1.3 / 1.4 (Power outputs: Extruder, Fan, Fan)
// 4 = Duemilanove w/ ATmega328P pin assignment
// 5 = Gen6
// 51 = Gen6 deluxe
// 6 = Sanguinololu < 1.2
// 62 = Sanguinololu 1.2 and above
// 63 = Melzi
// 64 = STB V1.1
// 65 = Azteeg X1
// 66 = Melzi with ATmega1284 (MaKr3d version)
// 67 = Azteeg X3
// 68 = Azteeg X3 Pro
// 7 = Ultimaker
// 71 = Ultimaker (Older electronics. Pre 1.5.4. This is rare)
// 72 = Ultimainboard 2.x (Uses TEMP_SENSOR 20)
// 77 = 3Drag Controller
// 8 = Teensylu
// 80 = Rumba
// 81 = Printrboard (AT90USB1286)
// 82 = Brainwave (AT90USB646)
// 83 = SAV Mk-I (AT90USB1286)
// 9 = Gen3+
// 70 = Megatronics
// 701= Megatronics v2.0
// 702= Minitronics v1.0
// 90 = Alpha OMCA board
// 91 = Final OMCA board
// 301= Rambo
// 21 = Elefu Ra Board (v3)
// 88 = 5DPrint D8 Driver Board
```

```
#ifndef MOTHERBOARD
```

```
#define MOTHERBOARD 33
```

//////////主板类型选择，请按以上各主板相应代号选择你个人的主板类型。33 是 ramps1.3/1.4（一个挤出机），34 是 ramps1.3/1.4（两个挤出机）

```
#endif

// Define this to set a custom name for your generic Mendel,
// #define CUSTOM_MENDEL_NAME "This Mendel"

// Define this to set a unique identifier for this printer, (Used by some programs to differentiate
between machines)
// You can use an online service to generate a random UUID.
(eg http://www.uuidgenerator.net/version4)
// #define MACHINE_UUID "00000000-0000-0000-0000-000000000000"

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

//// The following define selects which power supply you have. Please choose the one that
matches your setup
// 1 = ATX
// 2 = X-Box 360 203Watts (the blue wire connected to PS_ON and the red wire to VCC)

#define POWER_SUPPLY 1

// Define this to have the electronics keep the power supply off on startup. If you don't know
what this is leave it.
// #define PS_DEFAULT_OFF

//=====
//=====Thermal Settings =====
//=====
//
//--NORMAL IS 4.7kohm PULLUP!-- 1kohm pullup can be used on hotend sensor, using correct
resistor and table
//
//// 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 - best choice for EPCOS 100k (4.7k pullup)
// 2 is 200k thermistor - ATC Semitec 204GT-2 (4.7k pullup)
// 3 is Mendel-parts thermistor (4.7k pullup)
// 4 is 10k thermistor !! do not use it for a hotend. It gives bad resolution at high temp. !!
// 5 is 100K thermistor - ATC Semitec 104GT-2 (Used in ParCan & J-Head) (4.7k pullup)
// 6 is 100k EPCOS - Not as accurate as table 1 (created using a fluke thermocouple) (4.7k pullup)
// 7 is 100k Honeywell thermistor 135-104LAG-J01 (4.7k pullup)
// 71 is 100k Honeywell thermistor 135-104LAF-J01 (4.7k pullup)
```

```
// 8 is 100k 0603 SMD Vishay NTC50603E3104FXT (4.7k pullup)
// 9 is 100k GE Sensing AL03006-58.2K-97-G1 (4.7k pullup)
// 10 is 100k RS thermistor 198-961 (4.7k pullup)
// 11 is 100k beta 3950 1% thermistor (4.7k pullup)
// 12 is 100k 0603 SMD Vishay NTC50603E3104FXT (4.7k pullup) (calibrated for Makibox hot bed)
// 20 is the PT100 circuit found in the Ultimainboard V2.x
// 60 is 100k Maker's Tool Works Kapton Bed Thermistor beta=3950
//
// 1k ohm pullup tables - This is not normal, you would have to have changed out your 4.7k for
1k
//
// (but gives greater accuracy and more stable PID)
// 51 is 100k thermistor - EPCOS (1k pullup)
// 52 is 200k thermistor - ATC Semitec 204GT-2 (1k pullup)
// 55 is 100k thermistor - ATC Semitec 104GT-2 (Used in ParCan & J-Head) (1k pullup)
//
// 1047 is Pt1000 with 4k7 pullup
// 1010 is Pt1000 with 1k pullup (non standard)
// 147 is Pt100 with 4k7 pullup
// 110 is Pt100 with 1k pullup (non standard)

#define TEMP_SENSOR_0 1
#define TEMP_SENSOR_1 0
#define TEMP_SENSOR_2 0
#define TEMP_SENSOR_BED 1
//////////上面这些值是温度传感器类型配置，是能否正常读取温度的重要参数。
TEMP_SENSOR_0 是喷嘴温度传感器，TEMP_SENSOR_BED 是热床温度传感器，配置为 1 说明
两个都是 100K ntc 热敏电阻。如果你使用了其它温度传感器需要根据情况自行更改。如果没有，配置为 0

// This makes temp sensor 1 a redundant sensor for sensor 0. If the temperatures difference
between these sensors is to high the print will be aborted.
// #define TEMP_SENSOR_1_AS_REDUNDANT
#define MAX_REDUNDANT_TEMP_SENSOR_DIFF 10

// Actual temperature must be close to target for this long before M109 returns success
#define TEMP_RESIDENCY_TIME 10 // (seconds)
#define TEMP_HYSTERESIS 3 // (degC) range of +/- temperatures considered "close" to the
target one
//////////这个值决定当实际温度在正负 3 度时被认为接近设定的目标温度。适当加大该值可
以减少等待升温的时间，过大则对出丝不利，建议该值保持默认
#define TEMP_WINDOW 1 // (degC) Window around target to start the residency timer x
degC early.

// 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
```

//////////以上几个配置是喷嘴和热床在温度低于 5 度时，打印机将不能启动，表现为报错，并且加热头和热床的加热无法打开

```
// 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 275
```

```
#define HEATER_1_MAXTEMP 275
```

```
#define HEATER_2_MAXTEMP 275
```

```
#define BED_MAXTEMP 150
```

//////////以上是喷嘴和热床的最高温度配置，防止烧坏主板

```
// If your bed has low resistance e.g. .6 ohm and throws the fuse you can duty cycle it to reduce the
```

```
// average current. The value should be an integer and the heat bed will be turned on for 1 interval of
```

```
// HEATER_BED_DUTY_CYCLE_DIVIDER intervals.
```

```
//#define HEATER_BED_DUTY_CYCLE_DIVIDER 4
```

//////////该配置是为了防止热床电阻太小，长时间加热容易烧 mos 管，增加这个数字，可以防止 mos 管过热，但加热时间会增长

```
// If you want the M105 heater power reported in watts, define the BED_WATTS, and (shared for all extruders) EXTRUDER_WATTS
```

```
//#define EXTRUDER_WATTS (12.0*12.0/6.7) // P=I^2/R
```

```
//#define BED_WATTS (12.0*12.0/1.1) // P=I^2/R
```

//////////如果你希望用 M105 命令实时查看挤出机和热床功率，请将这两行取消注释(即去掉代码前的//)

//////////PID 温控配置，PID 参数设置，需要根据系统情况设置，可以通过 M303 代码调用 PID_autotune 函数获得基本 PID 参数，然后参照修改下面的 DEFAULT_Kp、DEFAULT_Ki、DEFAULT_Kd，例如：M303 E0 C8 S190，表示获取打印头 0（第一个打印头）、目标温度 190 度、循环调用 PID_autotune8 次后的相应的 PID 参数。系统自动生成的 PID 参数相当不错，

可保持默认。当然机器不同，也不一定都有效

```
// PID settings:
// Comment the following line to disable PID and enable bang-bang.
#define PIDTEMP
#define BANG_MAX 255 // limits current to nozzle while in bang-bang mode; 255=full current
#define PID_MAX 255 // limits current to nozzle while PID is active (see
PID_FUNCTIONAL_RANGE below); 255=full current
#ifdef PIDTEMP
  // #define PID_DEBUG // Sends debug data to the serial port.
  // #define PID_OPENLOOP 1 // Puts PID in open loop. M104/M140 sets the output power from 0
  to PID_MAX
  #define PID_FUNCTIONAL_RANGE 10 // If the temperature difference between the target
  temperature and the actual temperature
                                // is more then PID_FUNCTIONAL_RANGE then the PID will be shut off
  and the heater will be set to min/max.
  #define PID_INTEGRAL_DRIVE_MAX 255 //limit for the integral term
  #define K1 0.95 //smoothing factor within the PID
  #define PID_dT ((OVERSAMPLING * 8.0)/(F_CPU / 64.0 / 256.0)) //sampling period of the
  temperature routine

// If you are using a pre-configured 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

// Bed Temperature Control
// Select PID or bang-bang with PIDTEMPBED. If bang-bang, BED_LIMIT_SWITCHING will enable
  hysteresis
//
// Uncomment this to enable PID on the bed. It uses the same frequency PWM as the extruder.
// If your PID_dT above is the default, and correct for your hardware/configuration, that means
```

```
7.689Hz,
// which is fine for driving a square wave into a resistive load and does not significantly impact
you FET heating.
// This also works fine on a Fotek SSR-10DA Solid State Relay into a 250W heater.
// If your configuration is significantly different than this and you don't understand the issues
involved, you probably
// shouldn't use bed PID until someone else verifies your hardware works.
// If this is enabled, find your own PID constants below.
// #define PIDTEMPBED
//
// #define BED_LIMIT_SWITCHING

// This sets the max power delivered to the bed, and replaces the
HEATER_BED_DUTY_CYCLE_DIVIDER option.
// all forms of bed control obey this (PID, bang-bang, bang-bang with hysteresis)
// setting this to anything other than 255 enables a form of PWM to the bed just like
HEATER_BED_DUTY_CYCLE_DIVIDER did,
// so you shouldn't use it unless you are OK with PWM on your bed. (see the comment on
enabling PIDTEMPBED)
#define MAX_BED_POWER 255 // limits duty cycle to bed; 255=full current

#ifdef PIDTEMPBED
//120v 250W silicone heater into 4mm borosilicate (MendelMax 1.5+)
//from FOPDT model - kp=.39 Tp=405 Tdead=66, Tc set to 79.2, aggressive factor of .15 (vs .1, 1,
10)
    #define DEFAULT_bedKp 10.00
    #define DEFAULT_bedKi .023
    #define DEFAULT_bedKd 305.4

//120v 250W silicone heater into 4mm borosilicate (MendelMax 1.5+)
//from pidautotune
//    #define DEFAULT_bedKp 97.1
//    #define DEFAULT_bedKi 1.41
//    #define DEFAULT_bedKd 1675.16

// FIND YOUR OWN: "M303 E-1 C8 S90" to run autotune on the bed at 90 degreesC for 8 cycles.
#endif // PIDTEMPBED

//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
//if PREVENT_DANGEROUS_EXTRUDE is on, you can still disable (uncomment) very long bits of
```

```
extrusion separately.
#define PREVENT_LENGTHY_EXTRUDE

#define EXTRUDE_MINTEMP 170
//////////该值防止挤出头温度未达到设定目标温度而进行挤出操作时的潜在风险

#define EXTRUDE_MAXLENGTH (X_MAX_LENGTH+Y_MAX_LENGTH) //prevent extrusion of very
large distances.
//////////该值限制挤出的最大长度，超过该长度，挤出机不动作

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

// Uncomment the following line to enable CoreXY kinematics
// #define COREXY

// coarse Endstop Settings
#define ENDSTOPPULLUPS // Comment this out (using // at the start of the line) to disable the
endstop pullup resistors
//////////限位开关上拉电阻配置，如果你使用的是机械式的限位开关，请保留此部分

//////////机械式限位开关请保持该处不动
#ifndef ENDSTOPPULLUPS ////////////没有配置限位开关上拉电阻时，限位开关上拉电阻细分
控制
    // fine endstop settings: Individual pullups. will be ignored if ENDSTOPPULLUPS is defined
    // #define ENDSTOPPULLUP_XMAX
    // #define ENDSTOPPULLUP_YMAX
    // #define ENDSTOPPULLUP_ZMAX
    // #define ENDSTOPPULLUP_XMIN
    // #define ENDSTOPPULLUP_YMIN
    // #define ENDSTOPPULLUP_ZMIN
#endif
#define ENDSTOPPULLUPS ////////////配置限位开关上拉电阻时，限位开关上拉电阻细分控制
#define ENDSTOPPULLUP_XMAX
#define ENDSTOPPULLUP_YMAX
#define ENDSTOPPULLUP_ZMAX
#define ENDSTOPPULLUP_XMIN
#define ENDSTOPPULLUP_YMIN
#define ENDSTOPPULLUP_ZMIN
#endif
```


//////////机械限位开关通常会把连线连在 NC 端,这就会有有一个数字信号 1 输入到主板中。如果你的机械限位开关的连线是连在 NO 端,这就需要 ENDSTOP_INVERTING 采用 true 从而使数字信号 0 反向变为 1。对于光学开关来说,则一般不用进行调整。

你应该在电脑控制软件中实际试验一下限位开关的方向是否与你的预期相符,如果轴运动时触动限位开关就停止则说明,该项设置正常。如果没有停止,则应相应修改该值。注意!!!在做此测试时,应该给各个轴留出足够的运动反应空间。以免损坏机器

// The pullups are needed if you directly connect a mechanical endswitch between the signal and ground pins.

```
const bool X_MIN_ENDSTOP_INVERTING = true; // set to true to invert the logic of the endstop.
const bool Y_MIN_ENDSTOP_INVERTING = true; // set to true to invert the logic of the endstop.
const bool Z_MIN_ENDSTOP_INVERTING = true; // set to true to invert the logic of the endstop.
const bool X_MAX_ENDSTOP_INVERTING = true; // set to true to invert the logic of the endstop.
const bool Y_MAX_ENDSTOP_INVERTING = true; // set to true to invert the logic of the endstop.
const bool Z_MAX_ENDSTOP_INVERTING = true; // set to true to invert the logic of the endstop.
// #define DISABLE_MAX_ENDSTOPS
// #define DISABLE_MIN_ENDSTOPS
```

```
// Disable max endstops for compatibility with endstop checking routine
#if defined(COREXY) && !defined(DISABLE_MAX_ENDSTOPS)
  #define DISABLE_MAX_ENDSTOPS
#endif
```

```
// 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
```

//////////通常情况下上面的代码是不改动的,所有轴都是选择 false 的。然而,如果你的 3d 打印机 z 轴有手动调整的部件,可以在#define DISABLE_Z 行改为 true,这样在打印机打印时,可以手动调整 z 轴

```
// 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
```

//////////这几个参数是比较容易错的。根据自己机械的类型不同,两个的配置不尽相同。但是原则就是要保证原点应该在打印平台的左下角(原点位置为[0,0]),或右上角(原点位置为[max,max])。只有这样打印出来的模型才是正确的,否则会是某个轴的镜像而造成模型方位不对。

修改对应某个轴的配置(true 或 false)后,电机可能会反向

```
#define INVERT_X_DIR true // for Mendel set to false, for Orca set to true
#define INVERT_Y_DIR false // 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
```

//////////回原点方向配置。如果原点位置为最小值配置为-1，如果原点位置为最大值配置为1

// 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.
```

//////////这几个参数是配置打印尺寸的重要参数。这里需要说明的是坐标原点并不是打印中心，真正的打印中心一般在 $[(x.max-x.min)/2, (y.max-y.min)/2]$ 的位置。中心位置的坐标需要在后面的切片工具中使用到，打印中心坐标应该与这里的参数配置匹配，否则很可能会打印到平台以外。

// Travel limits after homing

```
#define X_MAX_POS 205
```

```
#define X_MIN_POS 0
```

```
#define Y_MAX_POS 205
```

```
#define Y_MIN_POS 0
```

```
#define Z_MAX_POS 200
```

```
#define Z_MIN_POS 0
```

```
#define X_MAX_LENGTH (X_MAX_POS - X_MIN_POS)
```

```
#define Y_MAX_LENGTH (Y_MAX_POS - Y_MIN_POS)
```

```
#define Z_MAX_LENGTH (Z_MAX_POS - Z_MIN_POS)
```

//////////热床自动调平配置（这个功能我没有用到，所以不是很清楚，请各位自行查找相关资料）

```
//===== Bed Auto Leveling =====
```

```
//#define ENABLE_AUTO_BED_LEVELING // Delete the comment to enable (remove // at the
start of the line)
```

```
#ifndef ENABLE_AUTO_BED_LEVELING

// There are 2 different ways to pick the X and Y locations to probe:

// - "grid" mode
//   Probe every point in a rectangular grid
//   You must specify the rectangle, and the density of sample points
//   This mode is preferred because there are more measurements.
//   It used to be called ACCURATE_BED_LEVELING but "grid" is more descriptive

// - "3-point" mode
//   Probe 3 arbitrary points on the bed (that aren't colinear)
//   You must specify the X & Y coordinates of all 3 points

#define AUTO_BED_LEVELING_GRID
// with AUTO_BED_LEVELING_GRID, the bed is sampled in a
// AUTO_BED_LEVELING_GRID_POINTSxAUTO_BED_LEVELING_GRID_POINTS grid
// and least squares solution is calculated
// Note: this feature occupies 10'206 byte
#ifdef AUTO_BED_LEVELING_GRID

    // set the rectangle in which to probe
    #define LEFT_PROBE_BED_POSITION 15
    #define RIGHT_PROBE_BED_POSITION 170
    #define BACK_PROBE_BED_POSITION 180
    #define FRONT_PROBE_BED_POSITION 20

    // set the number of grid points per dimension
    // I wouldn't see a reason to go above 3 (=9 probing points on the bed)
    #define AUTO_BED_LEVELING_GRID_POINTS 2

#else // not AUTO_BED_LEVELING_GRID
    // with no grid, just probe 3 arbitrary points. A simple cross-product
    // is used to estimate the plane of the print bed

    #define ABL_PROBE_PT_1_X 15
    #define ABL_PROBE_PT_1_Y 180
    #define ABL_PROBE_PT_2_X 15
    #define ABL_PROBE_PT_2_Y 20
    #define ABL_PROBE_PT_3_X 170
    #define ABL_PROBE_PT_3_Y 20

#endif // AUTO_BED_LEVELING_GRID
```

```
// these are the offsets to the probe relative to the extruder tip (Hotend - Probe)
#define X_PROBE_OFFSET_FROM_EXTRUDER -25
#define Y_PROBE_OFFSET_FROM_EXTRUDER -29
#define Z_PROBE_OFFSET_FROM_EXTRUDER -12.35

#define Z_RAISE_BEFORE_HOMING 4      // (in mm) Raise Z before homing (G28) for Probe
Clearance.

                                // Be sure you have this distance over your Z_MAX_POS in case

#define XY_TRAVEL_SPEED 8000        // X and Y axis travel speed between probes, in mm/min

#define Z_RAISE_BEFORE_PROBING 15    //How much the extruder will be raised before
traveling to the first probing point.
#define Z_RAISE_BETWEEN_PROBINGS 5 //How much the extruder will be raised when
traveling from between next probing points

//If defined, the Probe servo will be turned on only during movement and then turned off to
avoid jerk
//The value is the delay to turn the servo off after powered on - depends on the servo speed;
300ms is good value, but you can try lower it.
// You MUST HAVE the SERVO_ENDSTOPS defined to use here a value higher than zero
otherwise your code will not compile.

// #define PROBE_SERVO_DEACTIVATION_DELAY 300

//If you have enabled the Bed Auto Leveling and are using the same Z Probe for Z Homing,
//it is highly recommended you let this Z_SAFE_HOMING enabled!!!

#define Z_SAFE_HOMING // This feature is meant to avoid Z homing with probe outside the
bed area.

                                // When defined, it will:
                                // - Allow Z homing only after X and Y homing AND stepper drivers still
enabled
                                // - If stepper drivers timeout, it will need X and Y homing again before Z
homing
                                // - Position the probe in a defined XY point before Z Homing when homing
all axis (G28)
                                // - Block Z homing only when the probe is outside bed area.

#ifndef Z_SAFE_HOMING
```

```
#define Z_SAFE_HOMING_X_POINT (X_MAX_LENGTH/2) // X point for Z homing when
homing all axis (G28)
#define Z_SAFE_HOMING_Y_POINT (Y_MAX_LENGTH/2) // Y point for Z homing when
homing all axis (G28)

#endif

#endif // ENABLE_AUTO_BED_LEVELING

// The position of the homing switches
// #define MANUAL_HOME_POSITIONS // If defined, MANUAL_*_HOME_POS below will be used
// #define BED_CENTER_AT_0_0 // If defined, the center of the bed is at (X=0, Y=0)
//////////取消该行注释，可将热床中心定义为 X=0, Y=0

//////////手动设置回原点位置，如果要使用该功能，请将上面#define
MANUAL_HOME_POSITIONS 前的“//”删除。使用该功能后，默认回原点位置将是你所设定的
以下三个值的位置
//Manual homing switch locations:
// For deltabots this means top and center of the Cartesian print volume.
#define MANUAL_X_HOME_POS 0
#define MANUAL_Y_HOME_POS 0
#define MANUAL_Z_HOME_POS 0
// #define MANUAL_Z_HOME_POS 402 // For delta: Distance between nozzle and print surface
after homing.

//// 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)
//////////上面配置为回原点的速度，可根据实际情况做相应调整。单位是 mm/min

// default settings

#define DEFAULT_AXIS_STEPS_PER_UNIT {78.7402,78.7402,200.0*8/3,760*1.1} // default
steps per unit for Ultimaker
//////////这个参数是打印机打印尺寸是否正确的重要参数，参数含义为各轴运行 1mm 所需
要的脉冲数，分别对应 x,y,z,e 四轴。多数情况下这个数字都需要自己计算才可以。（可参考
http://calculator.josefprusa.cz/）
同步带传动时的计算公式（X/Y 轴）：步进电机每转步数(1.8 度步距角的电机为 200,0.9 度步
距角的电机为 400)*步进电机驱动细分配置（一般 16 细分）/同步带齿间距/同步轮齿数
丝杠传动时的计算公式（Z 轴）：步进电机每转步数*步进电机驱动细分配置/丝杠导程
挤出机计算公式（E 轴）：步进电机每转步数*步进电机驱动细分配置*挤出机齿轮传动比/
挤出轮周长
```

```
#define DEFAULT_MAX_FEEDRATE      {500, 500, 5, 25}  // (mm/sec)
/////该配置为各电机最高速度。过高的值需要更大的电流输出，这将导致电机过热，并且
有可能使电机在打印时丢步。一般可设置在 200-400
```

```
/////该配置为电机最大加速度。过高的电机加速度将导致电机在打印动作时过冲，从而丢
步，建议将 X/Y 最大加速度修改为 1000-3000
```

```
#define DEFAULT_MAX_ACCELERATION  {9000,9000,100,10000}  // 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.
```

```
/////默认打印加速度
```

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

```
/////默认回抽加速度
```

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

```
// Offset of the extruders (uncomment if using more than one and relying on firmware to
position when changing).
```

```
// The offset has to be X=0, Y=0 for the extruder 0 hotend (default extruder).
```

```
// For the other hotends it is their distance from the extruder 0 hotend.
```

```
// #define EXTRUDER_OFFSET_X {0.0, 20.00} // (in mm) for each extruder, offset of the hotend on
the X axis
```

```
// #define EXTRUDER_OFFSET_Y {0.0, 5.00} // (in mm) for each extruder, offset of the hotend on
the Y axis
```

```
/////该配置为加速度变化率。该值过大同样也会导致电机丢步。当该值处于合理范围内的
较小值时，打印动作将更平滑、打印机机械应力将更小、材料在换向时将有更好的附着力、
打印噪声也将降低；当该值处于合理范围内的较大值时，打印时间将缩短。建议谨慎修改
该值，最好不动
```

```
// The speed change that does not require acceleration (i.e. the software might assume it can be
done instantaneously)
```

```
#define DEFAULT_XYJERK            20.0  // (mm/sec)
```

```
#define DEFAULT_ZJERK             0.4   // (mm/sec)
```

```
#define DEFAULT_EJERK             5.0   // (mm/sec)
```

```
//=====
```

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

```
//=====
```

```
// Custom M code points
```

```
#define CUSTOM_M_CODES
#ifdef CUSTOM_M_CODES
    #define CUSTOM_M_CODE_SET_Z_PROBE_OFFSET 851
    #define Z_PROBE_OFFSET_RANGE_MIN -15
    #define Z_PROBE_OFFSET_RANGE_MAX -5
#endif

// EEPROM
// The microcontroller can store settings in the EEPROM, e.g. max velocity...
// M500 - stores parameters 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

//////////预加热配置，无需作修改
// 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

//////////LCD 和 SD 卡配置，请根据你的 LCD 板子做相应配置，在此不作说明
//LCD and SD support
//#define ULTRA_LCD //general LCD support, also 16x2
//#define DOGLCD // Support for SPI LCD 128x64 (Controller ST7565R graphic Display Family)
//#define SDSUPPORT // Enable SD Card Support in Hardware Console
//#define SDSLOW // Use slower SD transfer mode (not normally needed - uncomment if you're
getting volume init error)
//#define SD_CHECK_AND_RETRY // Use CRC checks and retries on the SD communication
//#define ENCODER_PULSES_PER_STEP 1 // Increase if you have a high resolution encoder
//#define ENCODER_STEPS_PER_MENU_ITEM 5 // Set according to ENCODER_PULSES_PER_STEP
or your liking
//#define ULTIMAKERCONTROLLER //as available from the Ultimaker online store.
```

```
// #define ULTIPANEL //the UltiPanel as on Thingiverse
// #define LCD_FEEDBACK_FREQUENCY_HZ 1000 // this is the tone frequency the buzzer
plays when on UI feedback. ie Screen Click
// #define LCD_FEEDBACK_FREQUENCY_DURATION_MS 100 // the duration the buzzer plays the
UI feedback sound. ie Screen Click

// The MaKr3d Makr-Panel with graphic controller and SD support
// http://reprap.org/wiki/MaKr3d\_MaKrPanel
// #define MAKRPANEL

// The RepRapDiscount Smart Controller (white PCB)
// http://reprap.org/wiki/RepRapDiscount\_Smart\_Controller
// #define REPRAP_DISCOUNT_SMART_CONTROLLER (去除这行代码前的//, 即代表可使用
LCD2004 屏)

// The GADGETS3D G3D LCD/SD Controller (blue PCB)
// http://reprap.org/wiki/RAMPS\_1.3/1.4\_GADGETS3D\_Shield\_with\_Panel
// #define G3D_PANEL

// The RepRapDiscount FULL GRAPHIC Smart Controller (quadratic white PCB)
// http://reprap.org/wiki/RepRapDiscount\_FULL\_GRAPHIC\_Smart\_Controller
//
// ==> REMEMBER TO INSTALL U8glib to your ARDUINO library
folder: http://code.google.com/p/u8glib/wiki/u8glib (将\ArduinoAddons\Arduino_1 或
0.x.x\libraries 里面的 U8glib 文件夹整个拷贝到 arduino 安装目录下\libraries 的这个文件夹。
然后关闭 arduino, 重启一下, 再编译。)
// #define REPRAP_DISCOUNT_FULL_GRAPHIC_SMART_CONTROLLER(去除这行代码前的//, 即代
表可使用 LCD12864 屏)

// The RepRapWorld REPRAPWORLD_KEYPAD v1.1
// http://reprapworld.com/?products ... 202&cPath=1591\_1626
// #define REPRAPWORLD_KEYPAD
// #define REPRAPWORLD_KEYPAD_MOVE_STEP 10.0 // how much should be moved when a key
is pressed, eg 10.0 means 10mm per click

// The Elefu RA Board Control Panel
// http://www.elefu.com/index.php?route=product&product\_id=53
// REMEMBER TO INSTALL LiquidCrystal_I2C.h in your ARUDINO library
folder: https://github.com/kiyoshigawa/LiquidCrystal\_I2C
// #define RA_CONTROL_PANEL

// automatic expansion
# if defined (MAKRPANEL)
# define DOGLCD
```



```
#define SDSUPPORT
#define ULTIPANEL
#define NEWPANEL
#define DEFAULT_LCD_CONTRAST 17
#endif

#if defined (REPRAP_DISCOUNT_FULL_GRAPHIC_SMART_CONTROLLER)
#define DOGLCD
#define U8GLIB_ST7920
#define REPRAP_DISCOUNT_SMART_CONTROLLER
#endif

#if defined(ULTIMAKERCONTROLLER) || defined(REPRAP_DISCOUNT_SMART_CONTROLLER) ||
defined(G3D_PANEL)
#define ULTIPANEL
#define NEWPANEL
#endif

#if defined(REPRAPWORLD_KEYPAD)
#define NEWPANEL
#define ULTIPANEL
#endif

#if defined(RA_CONTROL_PANEL)
#define ULTIPANEL
#define NEWPANEL
#define LCD_I2C_TYPE_PCA8574
#define LCD_I2C_ADDRESS 0x27 // I2C Address of the port expander
#endif

//I2C PANELS

//#define LCD_I2C_SAINSMART_YWROBOT
#ifdef LCD_I2C_SAINSMART_YWROBOT
// This uses the LiquidCrystal_I2C library
( https://bitbucket.org/fmalpartida/new-liquidcrystal/wiki/Home )
// Make sure it is placed in the Arduino libraries directory.
#define LCD_I2C_TYPE_PCF8575
#define LCD_I2C_ADDRESS 0x27 // I2C Address of the port expander
#define NEWPANEL
#define ULTIPANEL
#endif

// PANELolu2 LCD with status LEDs, separate encoder and click inputs
//#define LCD_I2C_PANELolu2
```

```
#ifndef LCD_I2C_PANELOLU2
// This uses the LiquidTWI2 library v1.2.3 or later
( https://github.com/lincomatic/LiquidTWI2 )
// Make sure the LiquidTWI2 directory is placed in the Arduino or Sketchbook libraries
subdirectory.
// (v1.2.3 no longer requires you to define PANELOLU in the LiquidTWI2.h library header file)
// Note: The PANELOLU2 encoder click input can either be directly connected to a pin
//      (if BTN_ENC defined to != -1) or read through I2C (when BTN_ENC == -1).
#define LCD_I2C_TYPE_MCP23017
#define LCD_I2C_ADDRESS 0x20 // I2C Address of the port expander
#define LCD_USE_I2C_BUZZER //comment out to disable buzzer on LCD
#define NEWPANEL
#define ULTIPANEL

#ifndef ENCODER_PULSES_PER_STEP
    #define ENCODER_PULSES_PER_STEP 4
#endif

#ifndef ENCODER_STEPS_PER_MENU_ITEM
    #define ENCODER_STEPS_PER_MENU_ITEM 1
#endif

#ifndef LCD_USE_I2C_BUZZER
    #define LCD_FEEDBACK_FREQUENCY_HZ 1000
    #define LCD_FEEDBACK_FREQUENCY_DURATION_MS 100
#endif

#endif

// Panucatt VIKI LCD with status LEDs, integrated click & L/R/U/P buttons, separate encoder
inputs
//#define LCD_I2C_VIKI
#ifndef LCD_I2C_VIKI
// This uses the LiquidTWI2 library v1.2.3 or later
( https://github.com/lincomatic/LiquidTWI2 )
// Make sure the LiquidTWI2 directory is placed in the Arduino or Sketchbook libraries
subdirectory.
// Note: The pause/stop/resume LCD button pin should be connected to the Arduino
//      BTN_ENC pin (or set BTN_ENC to -1 if not used)
#define LCD_I2C_TYPE_MCP23017
#define LCD_I2C_ADDRESS 0x20 // I2C Address of the port expander
#define LCD_USE_I2C_BUZZER //comment out to disable buzzer on LCD (requires LiquidTWI2
v1.2.3 or later)
```

```
#define NEWPANEL
#define ULTIPANEL
#endif

// Shift register panels
// -----
// 2 wire Non-latching LCD SR from:
// https://bitbucket.org/fmalpartid ... register-connection
// #define SR_LCD
#ifdef SR_LCD
    #define SR_LCD_2W_NL    // Non latching 2 wire shift register
    // #define NEWPANEL
#endif

#ifdef ULTIPANEL
// #define NEWPANEL //enable this if you have a click-encoder panel
#define SDSUPPORT
#define ULTRA_LCD
#ifdef DOGLCD // Change number of lines to match the DOG graphic display
    #define LCD_WIDTH 20
    #define LCD_HEIGHT 5
#else
    #define LCD_WIDTH 20
    #define LCD_HEIGHT 4
#endif
#else //no panel but just LCD
#ifdef ULTRA_LCD
#ifdef DOGLCD // Change number of lines to match the 128x64 graphics display
    #define LCD_WIDTH 20
    #define LCD_HEIGHT 5
#else
    #define LCD_WIDTH 16
    #define LCD_HEIGHT 2
#endif
#endif
#endif

// default LCD contrast for dogm-like LCD displays
#ifdef DOGLCD
# ifndef DEFAULT_LCD_CONTRAST
#  define DEFAULT_LCD_CONTRAST 32
# endif
#endif
```

```
// Increase the FAN pwm frequency. Removes the PWM noise but increases heating in the
FET/Arduino
// #define FAST_PWM_FAN

// Temperature status LEDs that display the hotend and bed temperature.
// If all hotends and bed temperature and temperature setpoint are < 54C then the BLUE led is on.
// Otherwise the RED led is on. There is 1C hysteresis.
// #define TEMP_STAT_LEDS

// Use software PWM to drive the fan, as for the heaters. This uses a very low frequency
// which is not as annoying as with the hardware PWM. On the other hand, if this frequency
// is too low, you should also increment SOFT_PWM_SCALE.
// #define FAN_SOFT_PWM

// Incrementing this by 1 will double the software PWM frequency,
// affecting heaters, and the fan if FAN_SOFT_PWM is enabled.
// However, control resolution will be halved for each increment;
// at zero value, there are 128 effective control positions.
#define SOFT_PWM_SCALE 0

// M240 Triggers a camera by emulating a Canon RC-1 Remote
// Data from: http://www.doc-diy.net/photo/rc-1\_hacked/
// #define PHOTOGRAPH_PIN 23

// SF send wrong arc g-codes when using Arc Point as fillet procedure
// #define SF_ARC_FIX

// Support for the BariCUDA Paste Extruder.
// #define BARICUDA

// define BlinkM/CyzRgb Support
// #define BLINKM

/*****\
* R/C SERVO support
* Sponsored by TrinityLabs, Reworked by codexmas
*****/

// Number of servos
//
// If you select a configuration below, this will receive a default value and does not need to be set
manually
// set it manually if you have more servos than extruders and wish to manually control some
```

```
// leaving it undefined or defining as 0 will disable the servo subsystem
// If unsure, leave commented / disabled
//
// #define NUM_SERVOS 3 // Servo index starts with 0 for M280 command

// Servo Endstops
//
// This allows for servo actuated endstops, primary usage is for the Z Axis to eliminate calibration
// or bed height changes.
// Use M206 command to correct for switch height offset to actual nozzle height. Store that
// setting with M500.
//
// #define SERVO_ENDSTOPS {-1, -1, 0} // Servo index for X, Y, Z. Disable with -1
// #define SERVO_ENDSTOP_ANGLES {0,0, 0,0, 70,0} // X,Y,Z Axis Extend and Retract angles

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

#endif // __CONFIGURATION_H
```