```
//!AGG:G131231
//AGG:G131203
#define MOD3DPRN_FIRMWARE_VER 1.5

//*** MODELLO ***
//#define MOD3DPRN_BASIC
//#define MOD3DPRN_MONSTER
//#define MOD3DPRN_LAB
#define MOD3DPRN_BASICL
//#define MOD3DPRN_BASICLX

//*** ELETTRONICA ***
#define ELE3DPRN_MINI
//#define ELE3DPRN_MEGA

//#define ELE3DPRN_LCD



#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 scali

//===========================================================================
//============================= DELTA Printer ===============================
//===========================================================================
// For a Delta printer replace the configuration files with the files in the
// example_configurations/delta directory.
//

// User-specified version info of this build to display in [Pronterface, etc]
// startup. Implementation of an idea by Prof Braino to inform user that any c
// 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 "(Gianfranco Fazzini, 3DPRN)" // Who made the c

// SERIAL_PORT selects which serial port should be used for communication wit
// This allows the connection of wireless adapters (for instance) to non-defau
// Serial port 0 is still used by the Arduino bootloader regardless of this se
#define SERIAL_PORT 0

// This determines the communication speed of the printer
// This determines the communication speed of the printer
#define BAUDRATE 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 ch
// 10 = Gen7 custom (Alfons3 Version) "https://github.com/Alfons3/Generation_7
// 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)
// 84 = Teensy++2.0 (AT90USB1286) // CLI compile: DEFINES=AT90USBxx_TEENSYPP_A
// 9  = Gen3+
// 70 = Megatronics
// 701= Megatronics v2.0
// 702= Minitronics v1.0 / v1.1
// 703= Megatronics v3.0

// 90 = Alpha OMCA board
// 91 = Final OMCA board
// 301= Rambo
// 21 = Elefu Ra Board (v3)
// 88 = 5DPrint D8 Driver Board

//!AGG:140512
```

```
//#ifndef MOTHERBOARD
//#define MOTHERBOARD 702
//#endif
#if defined(ELE3DPRN_MINI)
  #define MOTHERBOARD 702
#endif
#if defined(ELE3DPRN_MEGA)
  #define MOTHERBOARD 703
#endif


// Define this to set a custom name for your generic Mendel,
// #define CUSTOM_MENDEL_NAME "This Mendel"
//AGG:G131231
#if defined(MOD3DPRN_BASIC)
  #define CUSTOM_MENDEL_NAME "3DPRN-BASIC"
#endif
#if defined(MOD3DPRN_BASICL)
  #define CUSTOM_MENDEL_NAME "3DPRN-BASICL"
#endif
#if defined(MOD3DPRN_BASICLX)
  #define CUSTOM_MENDEL_NAME "3DPRN-BASICL"
#endif
#if defined(MOD3DPRN_LAB)
  #define CUSTOM_MENDEL_NAME "3DPRN-LAB"
#endif
#if defined(MOD3DPRN_MONSTER)
  #define CUSTOM_MENDEL_NAME "3DPRNMONSTER"
#endif


// Define this to set a unique identifier for this printer, (Used by some prog
// You can use an online service to generate a random UUID. (eg http://www.uui
// #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 t
// 1 = ATX
// 2 = X-Box 360 203Watts (the blue wire connected to PS_ON and the red wire t

#define POWER_SUPPLY 1

// Define this to have the electronics keep the power supply off on startup. I
// #define PS_DEFAULT_OFF

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

```
//============================================================================
//
//--NORMAL IS 4.7kohm PULLUP!-- 1kohm pullup can be used on hotend sensor, usi
//
//// 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
// 5 is 100K thermistor - ATC Semitec 104GT-2 (Used in ParCan & J-Head) (4.7k
// 6 is 100k EPCOS - Not as accurate as table 1 (created using a fluke thermoc
// 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 NTCS0603E3104FXT (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 NTCS0603E3104FXT (4.7k pullup) (calibrated for N
// 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 change
//                           (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 p
//
// 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)
// 70 is 500C thermistor for Pico hot end

//!AGG:G131231
#define TEMP_SENSOR_0 5
#define TEMP_SENSOR_1 0
#define TEMP_SENSOR_2 0
#define TEMP_SENSOR_BED 1

// This makes temp sensor 1 a redundant sensor for sensor 0. If the temperatur
//#defineTEMP_SENSOR_1_AS_REDUNDANT
#define MAX_REDUNDANT_TEMP_SENSOR_DIFF 10

// Actual temperature must be close to target for this long before M109 return
#define TEMP_RESIDENCY_TIME 1//AGG:G140212 era 10  // (seconds)
#define TEMP_HYSTERESIS 3       // (degC) range of +/- temperatures considere
```

```
#define TEMP_WINDOW     1       // (degC) Window around target to start the re

// The minimal temperature defines the temperature below which the heater wil
// 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 0//!AGG:G131231 era 5. impostato a 0 per considerare la po

// When temperature exceeds max temp, your heater will be switched off.
// This feature exists to protect your hotend from overheating accidentally, b
// You should use MINTEMP for thermistor short/failure protection.
#define HEATER_0_MAXTEMP 245//AGG:G140212 era 275
#define HEATER_1_MAXTEMP 245//AGG:G140212 era 275
#define HEATER_2_MAXTEMP 245//AGG:G140212 era 275
#define BED_MAXTEMP 120//!AGG:G131231 era 150

// If your bed has low resistance e.g. .6 ohm and throws the fuse you can duty
// average current. The value should be an integer and the heat bed will be tu
// HEATER_BED_DUTY_CYCLE_DIVIDER intervals.
//#define HEATER_BED_DUTY_CYCLE_DIVIDER 4

// If you want the M105 heater power reported in watts, define the BED_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

// PID settings:
// Comment the following line to disable PID and enable bang-bang.
#define PIDTEMP //AGG:G140219
#define BANG_MAX 255//AGG:140219 era 255 // limits current to nozzle while in
#define PID_MAX 255//AGG:G140219 era 255 // limits current to nozzle while PID


#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 outpu
  #define PID_FUNCTIONAL_RANGE 40//AGG:G140219 era 10 // If the temperature di
                                  // is more then PID_FUNCTIONAL_RANGE then t
  #define PID_INTEGRAL_DRIVE_MAX 255  //limit for the integral term
  #define K1 0.95 //smoothing factor within the PID
  #define PID_dT ((OVERSAMPLENR * 8.0)/(F_CPU / 64.0 / 256.0)) //sampling per

// If you are using a preconfigured hotend then you can use one of the value s
// 3DPRN AGG:G14022014
    #define  DEFAULT_Kp 59.33
    #define  DEFAULT_Ki 7.81
    #define  DEFAULT_Kd 112.76
/*
```

```
// 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
//
// Uncomment this to enable PID on the bed. It uses the same frequency PWM as
// If your PID_dT above is the default, and correct for your hardware/configur
// which is fine for driving a square wave into a resistive load and does not
// This also works fine on a Fotek SSR-10DA Solid State Relay into a 250W heat
// If your configuration is significantly different than this and you don't ur
// 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_D
// all forms of bed control obey this (PID, bang-bang, bang-bang with hysteres
// setting this to anything other than 255 enables a form of PWM to the bed ju
// so you shouldn't use it unless you are OK with PWM on your bed.  (see the c
#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 facto
    #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
#endif // PIDTEMPBED



//this prevents dangerous Extruder moves, i.e. if the temperature is under the
//can be software-disabled for whatever purposes by
#define PREVENT_DANGEROUS_EXTRUDE
//if PREVENT_DANGEROUS_EXTRUDE is on, you can still disable (uncomment) very l
//#define PREVENT_LENGTHY_EXTRUDE //!AGG:G131001

#define EXTRUDE_MINTEMP 170
#define EXTRUDE_MAXLENGTH (X_MAX_LENGTH+Y_MAX_LENGTH) //prevent extrusion of v

/*================== Thermal Runaway Protection ==============================
This is a feature to protect your printer from burn up in flames if it has
a thermistor coming off place (this happened to a friend of mine recently and
motivated me writing this feature).

The issue: If a thermistor come off, it will read a lower temperature than act
The system will turn the heater on forever, burning up the filament and anythi
else around.

After the temperature reaches the target for the first time, this feature will
start measuring for how long the current temperature stays below the target
minus _HYSTERESIS (set_temperature - THERMAL_RUNAWAY_PROTECTION_HYSTERESIS).

If it stays longer than _PERIOD, it means the thermistor temperature
cannot catch up with the target, so something *may be* wrong. Then, to be on t
safe side, the system will he halt.

Bear in mind the count down will just start AFTER the first time the
thermistor temperature is over the target, so you will have no problem if
your extruder heater takes 2 minutes to hit the target on heating.

*/
// If you want to enable this feature for all your extruder heaters,
// uncomment the 2 defines below:

// Parameters for all extruder heaters
//AGG:G140721
#define THERMAL_RUNAWAY_PROTECTION_PERIOD 40 //in seconds
#define THERMAL_RUNAWAY_PROTECTION_HYSTERESIS 4 // in degree Celsius

// If you want to enable this feature for your bed heater,
// uncomment the 2 defines below:

// Parameters for the bed heater
//AGG:G140721
```

```
#define THERMAL_RUNAWAY_PROTECTION_BED_PERIOD 20 //in seconds
#define THERMAL_RUNAWAY_PROTECTION_BED_HYSTERESIS 2 // in degree Celsius
//===========================================================================



//===========================================================================
//=============================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)

#ifndef ENDSTOPPULLUPS
  // fine endstop settings: Individual pullups. will be ignored if ENDSTOPPULL
  // #define ENDSTOPPULLUP_XMAX
  // #define ENDSTOPPULLUP_YMAX
  // #define ENDSTOPPULLUP_ZMAX
  // #define ENDSTOPPULLUP_XMIN
  // #define ENDSTOPPULLUP_YMIN
  // #define ENDSTOPPULLUP_ZMIN
#endif

#ifdef ENDSTOPPULLUPS
  #define ENDSTOPPULLUP_XMAX
  #define ENDSTOPPULLUP_YMAX
  #define ENDSTOPPULLUP_ZMAX
  #define ENDSTOPPULLUP_XMIN
  #define ENDSTOPPULLUP_YMIN
  #define ENDSTOPPULLUP_ZMIN
#endif

//The pullups are needed if you directly connect a mechanical endswitch betwe
#ifdef ELE3DPRN_MINI
  const bool X_MIN_ENDSTOP_INVERTING = false; // set to true to invert the log
  const bool Y_MIN_ENDSTOP_INVERTING = false; // set to true to invert the log
  const bool Z_MIN_ENDSTOP_INVERTING = false; // set to true to invert the log
  const bool X_MAX_ENDSTOP_INVERTING = false; // set to true to invert the log
  const bool Y_MAX_ENDSTOP_INVERTING = false; // set to true to invert the log
  const bool Z_MAX_ENDSTOP_INVERTING = false; // set to true to invert the log
#endif
#ifdef ELE3DPRN_MEGA
  const bool X_MIN_ENDSTOP_INVERTING = false; // set to true to invert the log
  const bool Y_MIN_ENDSTOP_INVERTING = false; // set to true to invert the log
  const bool Z_MIN_ENDSTOP_INVERTING = true; // set to true to invert the logi
  const bool X_MAX_ENDSTOP_INVERTING = true; // set to true to invert the logi
  const bool Y_MAX_ENDSTOP_INVERTING = true; // set to true to invert the logi
```

```
  const bool Z_MAX_ENDSTOP_INVERTING = false; // set to true to invert the log
#endif
//#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 (Activ
#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 DISABLE_INACTIVE_EXTRUDER true //disable only inactive extruders and l

#define INVERT_X_DIR false    // 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 true   // for direct drive extruder v9 set to true, for
#define INVERT_E1_DIR true    // for direct drive extruder v9 set to true, for
#define INVERT_E2_DIR true   // for direct drive extruder v9 set to true, for
//!AGG:G140514
#if defined(ELE3DPRN_MEGA)
  #define INVERT_E0_DIR false   // for direct drive extruder v9 set to true, f
  #define INVERT_E1_DIR false    // for direct drive extruder v9 set to true,
  #define INVERT_E2_DIR false   // for direct drive extruder v9 set to true, f
#endif

// 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
//AGG:G131231
#if defined(MOD3DPRN_LAB)
  #define min_software_endstops false //If true, axis won't move to coordinate
#endif
#define max_software_endstops true  // If true, axis won't move to coordinates
```

```
// 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

//AGG:G131203
#if defined(MOD3DPRN_MONSTER)
  #define X_MAX_POS 650
  #define Y_MAX_POS 670
  #define Z_MAX_POS 515
#endif

#if defined(MOD3DPRN_LAB)
  #define X_MAX_POS 320
  #define Y_MAX_POS 450
  #define Z_MAX_POS 300
#endif

#if defined(MOD3DPRN_BASICL) || defined(MOD3DPRN_BASICLX)
  #define X_MAX_POS 200
  #define Y_MAX_POS 200
  #define Z_MAX_POS 200
#endif

#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)
//============================ Bed Auto Leveling ============================

//#define ENABLE_AUTO_BED_LEVELING // Delete the comment to enable (remove //

#ifdefENABLE_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 descriptiv

//  - "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_POINTSgrid
// 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 esimate 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 -
#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)
                                      // Be sure you have this distance over

#define XY_TRAVEL_SPEED 8000         // X and Y axis travel speed between pr

#define Z_RAISE_BEFORE_PROBING 15    //How much the extruder will be raised
#define Z_RAISE_BETWEEN_PROBINGS 5  //How much the extruder will be raised w


//If defined, the Probe servo will be turned on only during movement and the
//The value is the delay to turn the servo off after powered on - depends on
// You MUST HAVE the SERVO_ENDSTOPS defined to use here a value higher than
```

```
//  #define PROBE_SERVO_DEACTIVATION_DELAY 300


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

  #define Z_SAFE_HOMING   // This feature is meant to avoid Z homing with prob
                          // When defined, it will:
                          // - Allow Z homing only after X and Y homing AND st
                          // - If stepper drivers timeout, it will need X and
                          // - Position the probe in a defined XY point before
                          // - Block Z homing only when the probe is outside b

  #ifdef Z_SAFE_HOMING

    #define Z_SAFE_HOMING_X_POINT (X_MAX_LENGTH/2)    // X point for Z homing
    #define Z_SAFE_HOMING_Y_POINT (Y_MAX_LENGTH/2)    // Y point for Z homing

  #endif

#endif //ENABLE_AUTO_BED_LEVELING


// The position of the homing switches
//#define MANUAL_HOME_POSITIONS  // If defined, MANUAL_*_HOME_POS below will b
//#define BED_CENTER_AT_0_0  // If defined, the center of the bed is at (X=0,

//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 prin

//// MOVEMENT SETTINGS
#define NUM_AXIS 4 // The axis order in all axis related arrays is X, Y, Z, E
//AGG:G140901 #define HOMING_FEEDRATE {50*60, 50*60, 10*60, 0}  // set the hom
#define HOMING_FEEDRATE {40*60, 40*60, 30*60, 0}  // set the homing speeds (mm

// default settings

#define DEFAULT_AXIS_STEPS_PER_UNIT  {160,160,160,810.7}  // default steps pe
#define DEFAULT_MAX_FEEDRATE         {500, 500, 500, 500}    // (mm/sec)
#define DEFAULT_MAX_ACCELERATION     {4500,4500,1000,10000}    // X, Y, Z, E

#define DEFAULT_ACCELERATION         1000    // X, Y, Z and E max acceleratio
#define DEFAULT_RETRACT_ACCELERATION 8000   // X, Y, Z and E max acceleration

// Offset of the extruders (uncomment if using more than one and relying on fi
```

```
// 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, offse
// #define EXTRUDER_OFFSET_Y {0.0, 5.00}  // (in mm) for each extruder, offse

// The speed change that does not require acceleration (i.e. the software migh
#define DEFAULT_XYJERK                10.0    // (mm/sec)ORIGINAL 20.0
#define DEFAULT_ZJERK                 10     // (mm/sec)
#define DEFAULT_EJERK                 50.0    // (mm/sec)//AGG:G2014-02-05 er

//!AGG:G131220
#if defined(MOD3DPRN_LAB)
  #define DEFAULT_MAX_ACCELERATION     {1500,1500,1500,20000}    // X, Y, Z,
  #define DEFAULT_ACCELERATION          800    // X, Y, Z and E max accelerati
  #define DEFAULT_XYJERK                10.0    // (mm/sec)
#endif
#if defined(MOD3DPRN_MONSTER)
  #define DEFAULT_MAX_ACCELERATION     {1000,1000,1000,8000}    // X, Y, Z, E
  #define DEFAULT_ACCELERATION          200    // X, Y, Z and E max accelerati
  #define DEFAULT_XYJERK                10.0    // (mm/sec)
#endif
#if defined(MOD3DPRN_BASICL) || defined(MOD3DPRN_BASICLX)
  #define DEFAULT_MAX_FEEDRATE          {200, 200, 80, 500}    // (mm/sec)
  #define DEFAULT_MAX_ACCELERATION     {1000,1000,1000,8000}    // X, Y, Z, E
  #define DEFAULT_ACCELERATION          1000    // X, Y, Z and E max accelerat
  #define DEFAULT_XYJERK                10.0    // (mm/sec) ORIGINAL 20.0
#endif

//===========================================================================
//=============================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 chang
// M502 - reverts to the default "factory settings".  You still need to store
//define this to enable EEPROM support
#define EEPROM_SETTINGS //AGG:G131231
//to disable EEPROM Serial responses and decrease program space by ~1700 byte
```

```
// please keep turned on if you can.
#define EEPROM_CHITCHAT //AGG:G131231

// Preheat Constants
#define PLA_PREHEAT_HOTEND_TEMP 190
#define PLA_PREHEAT_HPB_TEMP 70
#define PLA_PREHEAT_FAN_SPEED 255   // Insert Value between 0 and 255

#define ABS_PREHEAT_HOTEND_TEMP 230
#define ABS_PREHEAT_HPB_TEMP 100
#define ABS_PREHEAT_FAN_SPEED 255   // Insert Value between 0 and 255

//LCD and SD support
//#define ULTRA_LCD  //general LCD support, also 16x2
//#define DOGLCD  // Support for SPI LCD 128x64 (Controller ST7565R graphic Di
//#define SDSUPPORT // Enable SD Card Support in Hardware Console
//#define SDSLOW // Use slower SD transfer mode (not normally needed - uncomme
//#define SD_CHECK_AND_RETRY // Use CRC checks and retries on the SD communica
//#define ENCODER_PULSES_PER_STEP 1 // Increase if you have a high resolution
//#define ENCODER_STEPS_PER_MENU_ITEM 5 // Set according to ENCODER_PULSES_PE
//#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 buz
//#define LCD_FEEDBACK_FREQUENCY_DURATION_MS 100 // the duration the buzzer p
//!AGG:140512
#if defined(ELE3DPRN_LCD)
  #define ULTIPANEL  //the ultipanel as on thingiverse
  #define NEWPANEL
#endif

// 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
//#defineREPRAP_DISCOUNT_SMART_CONTROLLER

// 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
//#defineREPRAP_DISCOUNT_FULL_GRAPHIC_SMART_CONTROLLER

// The RepRapWorld REPRAPWORLD_KEYPAD v1.1
```

```
// http://reprapworld.com/?products_details&products_id=202&cPath=1591_1626
//#define REPRAPWORLD_KEYPAD
//#define REPRAPWORLD_KEYPAD_MOVE_STEP 10.0 // how much should be moved when

// The Elefu RA Board Control Panel
// http://www.elefu.com/index.php?route=product/product&product_id=53
// REMEMBER TO INSTALL LiquidCrystal_I2C.h in your ARUDINO library folder: ht
//#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)
 #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

//#defineLCD_I2C_SAINSMART_YWROBOT
#if defLCD_I2C_SAINSMART_YWROBOT
  // This uses the LiquidCrystal_I2C library ( https://bitbucket.org/fmalpart
  // 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
#ifdef LCD_I2C_PANELOLU2
  // This uses the LiquidTWI2 library v1.2.3 or later ( https://github.com/li
  // Make sure the LiquidTWI2 directory is placed in the Arduino or Sketchbook
  // (v1.2.3 no longer requires you to define PANELOLU in the LiquidTWI2.h lib
  // Note: The PANELOLU2 encoder click input can either be directly connected
  //       (if BTN_ENC defined to != -1) or read through I2C (when BTN_ENC ==
  #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


  #ifdef 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, sep
//#define LCD_I2C_VIKI
#ifdef LCD_I2C_VIKI
  // This uses the LiquidTWI2 library v1.2.3 or later ( https://github.com/li
  // Make sure the LiquidTWI2 directory is placed in the Arduino or Sketchbook
  // Note: The pause/stop/resume LCD button pin should be connected to the Arc
  //       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
  #define NEWPANEL
  #define ULTIPANEL
#endif

// Shift register panels
// ----------------------
```

```
// 2 wire Non-latching LCD SR from:
//https://bitbucket.org/fmalpartida/new-liquidcrystal/wiki/schematics#!shiftr
//#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 displa
    #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 heatin
//#define FAST_PWM_FAN

// Temperature status LEDs that display the hotend and bet temperature.
// If all hotends and bed temperature and temperature setpoint are < 54C then
// 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 lo
// which is not ass annoying as with the hardware PWM. On the other hand, if t
```

```
// 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
// set it manually if you have more servos than extruders and wish to manually
// 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
// Use M206 command to correct for switch height offset to actual nozzle heigh
//
//#define SERVO_ENDSTOPS {-1, -1, 0} // Servo index for X, Y, Z. Disable with
//#define SERVO_ENDSTOP_ANGLES {0,0, 0,0, 70,0} // X,Y,Z Axis Extend and Retra

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

#endif //__CONFIGURATION_H
```