

MKS MINI LCD12864 firmware modification method:

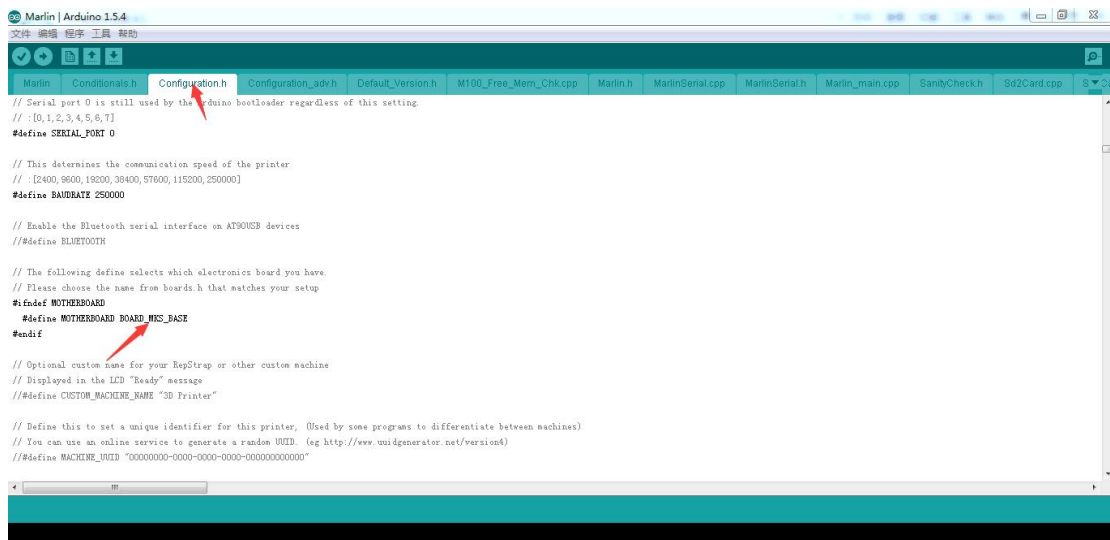
Some older firmware versions may not be available. Please download new firmware from GitHub <https://github.com/thingsmart/Marlin-RC>

On the basis of the Marlin-RC.zip modification:

In this firmware, marlin folder file by default is an XYZ machine, if is the delta machine, can copy the firmware "marlin / example_configurations / delta / generic" folder "Configuration.h" and "Configuration_adv.h" the two files to the Marlin folder, replace the original two files, and then modify as described below method

1. Find the motherboard type in configuration.h

Board Type Select "BOARD_MKS_BASE"



```
Marlin | Arduino 1.5.4
文件 编辑 程序 工具 帮助
Marlin Conditional.h Configuration.h Configuration_adv.h Default_Version.h M100_Free_Mem_Chk.cpp Marlin.h MarlinSerial.cpp MarlinSerial.h Marlin_main.cpp SanityCheck.h Sd2Card.cpp Sd2Spi.cpp
// Serial port 0 is still used by the Arduino bootloader regardless of this setting
// :0,1,2,3,4,5,6,7]
#define SERIAL_PORT 0

// This determines the communication speed of the printer
// :[2400,9600,19200,38400,57600,115200,250000]
#define BAUDRATE 250000

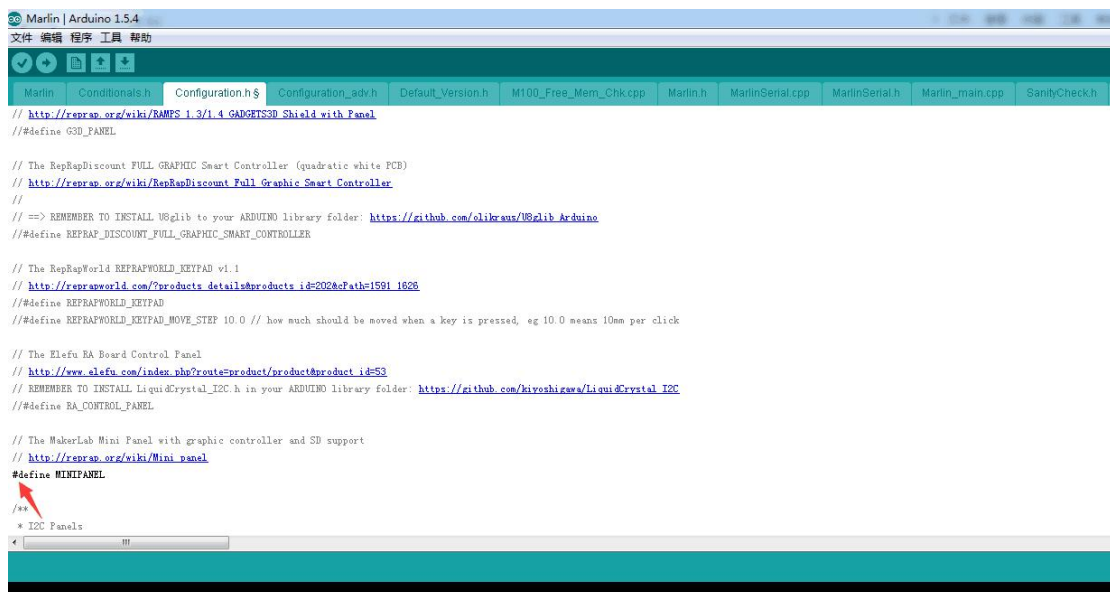
// Enable the Bluetooth serial interface on AT90USB devices
// #define BLUETOOTH

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

// Optional custom name for your RepRap or other custom machine
// Displayed in the LCD "Ready" message
// #define CUSTOM_MACHINE_NAME "3D Printer"

// 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"
```

2. Display Type Selection:



```
Marlin | Arduino 1.5.4
文件 编辑 程序 工具 帮助
Marlin Conditional.h Configuration.h Configuration_adv.h Default_Version.h M100_Free_Mem_Chk.cpp Marlin.h MarlinSerial.cpp MarlinSerial.h Marlin_main.cpp SanityCheck.h
// 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 Usglib to your ARDUINO library folder: https://github.com/olibrus/USglib_Arduino
// #define REPRAP_DISCOUNT_FULL_GRAPHIC_SMART_CONTROLLER

// The RepRapWorld REPRAPWORLD_KEYPAD v1.1
// http://reprapworld.com/?products_details&products_id=202&Path=1591_1606
// #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&product_id=53
// REMEMBER TO INSTALL LiQui@Crystal_I2C.h in your ARDUINO library folder: https://github.com/kivoshisawa/Liqui@Crystal_I2C
// #define RA_CONTROL_PANEL

// The MakerLab Mini Panel with graphic controller and SD support
// http://reprap.org/wiki/Mini_panel
#define MIMIPANEL
// **
* I2C Panels
```

3. Find pins_RAMPS_13.h

#elif ENABLED (MINIPANEL) part is the modified as follows: (red part of the words on behalf of the need to modify)

```
#elif ENABLED(MINIPANEL)

#define BEEPER_PIN 42

// Pins for DOGM SPI LCD Support

#define DOGLCD_A0 44

#define DOGLCD_CS 66

#define LCD_PIN_BL 65 // backlight LED on A11/D65

#define SDSS 53

#define KILL_PIN 64

// GLCD features

// #define LCD_CONTRAST 190

// Uncomment screen orientation

// #define LCD_SCREEN_ROT_90

// #define LCD_SCREEN_ROT_180

// #define LCD_SCREEN_ROT_270

// The encoder and click button

#define BTN_EN1 40

#define BTN_EN2 63

#define BTN_ENC 59 // the click switch

// not connected to a pin

#define SD_DETECT_PIN 49
```

Modified to: (blue word means the modified part)

```
#elif ENABLED(MINIPANEL)

#define BEEPER_PIN 37

// Pins for DOGM SPI LCD Support

#define DOGLCD_A0 27

#define DOGLCD_CS 25

#define LCD_PIN_BL 65 // backlight LED on A11/D65

#define SDSS 53

#define KILL_PIN 64

// GLCD features

// #define LCD_CONTRAST 190

// Uncomment screen orientation

// #define LCD_SCREEN_ROT_90

// #define LCD_SCREEN_ROT_180

// #define LCD_SCREEN_ROT_270

// The encoder and click button

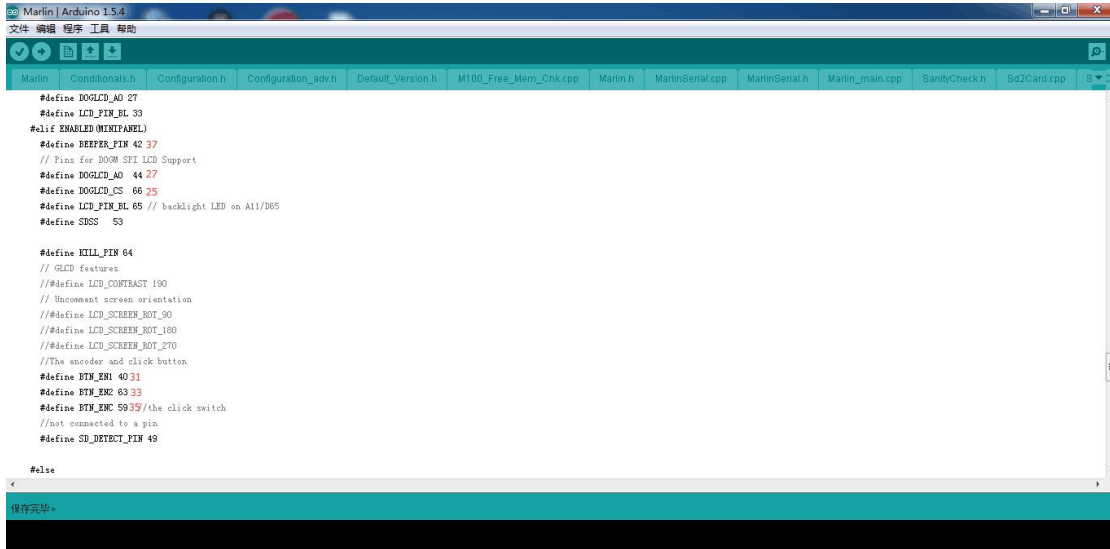
#define BTN_EN1 31

#define BTN_EN2 33

#define BTN_ENC 35 // the click switch

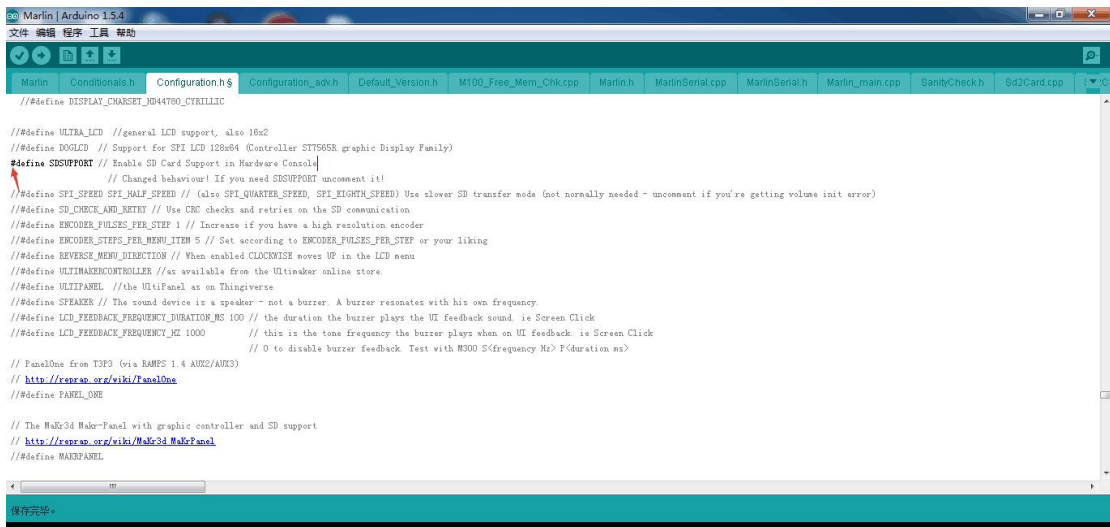
// not connected to a pin

#define SD_DETECT_PIN 49
```



Also need to pay attention to the place:

SD card in the configuration.h need to be enabled, otherwise can not read the card. As shown below, remove the double slash .



In the configuration.h, the arrow inside the brackets to "en" is English, "cn" is Chinese, but also support other languages, can be changed then need to re-compile the programming process!!

```
Marlin | Arduino 1.5.4
文件 编辑 程序 工具 帮助

Marlin  Conditionals.h  Configuration.h$  Configuration_adv.h  Default_Version.h  M100_Freq_Mem_Cbk.cpp  Marlin.h  MarlinSerial.cpp  MarlinSerial.h  Marlin_main.cpp  SanityCheck.h  Sd2Card.cpp

#define ABS_PREHEAT_HOTEND_TEMP 240
#define ABS_PREHEAT_HIP_TEMP 110
#define ABS_PREHEAT_FAN_SPEED 0 // Insert Value between 0 and 255

//=====LCD and SD support=====
// @section lcd

// Define your display language below. Replace (en) with your language code and uncomment.
// en, pl, fr, de, es, ru, bg, it, pt, pt_utf8, pt-br, pt-br_utf8, fi, sh, nl, ca, eu, kana, kana_utf8, cn, cz, test
// See also language.h
#define LANGUAGE_INCLUDE GENERATE_LANGUAGE_INCLUDE(cn)

// Choose ONE of these 3 charsets. This has to match your hardware. Ignored for full graphic display.
// To find out what type you have - compile with (test) - upload - click to get the menu. You'll see two typical lines from the upper half of the charset.
// See also https://github.com/MarlinFirmware/Marlin/wiki/LCD-Language
#define DISPLAY_CHARSET_ID44780_JAPAN // this is the most common hardware
// #define DISPLAY_CHARSET_ID44780_WESTERN
// #define DISPLAY_CHARSET_ID44780_CTRILLIC

// #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
// Changed behaviour! If you need SDSUPPORT uncomment it!
// #define SPI_SPEED SPI_HALF_SPEED // (also SPI_QUARTER_SPEED, SPI_EIGHTH_SPEED) Use slower SD transfer mode (not normally needed - uncomment if you're getting volume init error)

保存完毕
```

Other configuration parameters with the old version of the firmware is the same, no longer described!

Dimensions:

