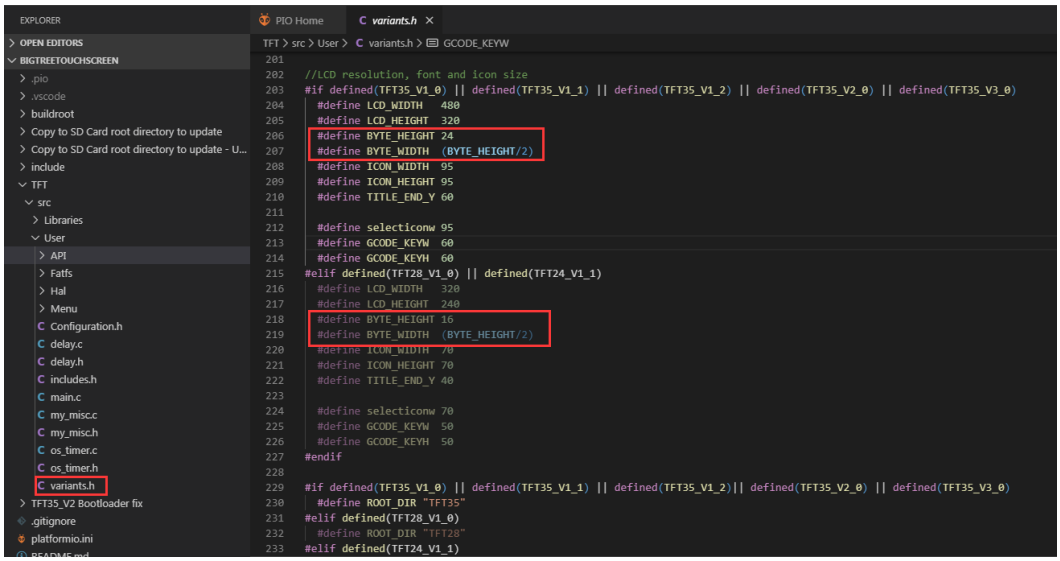
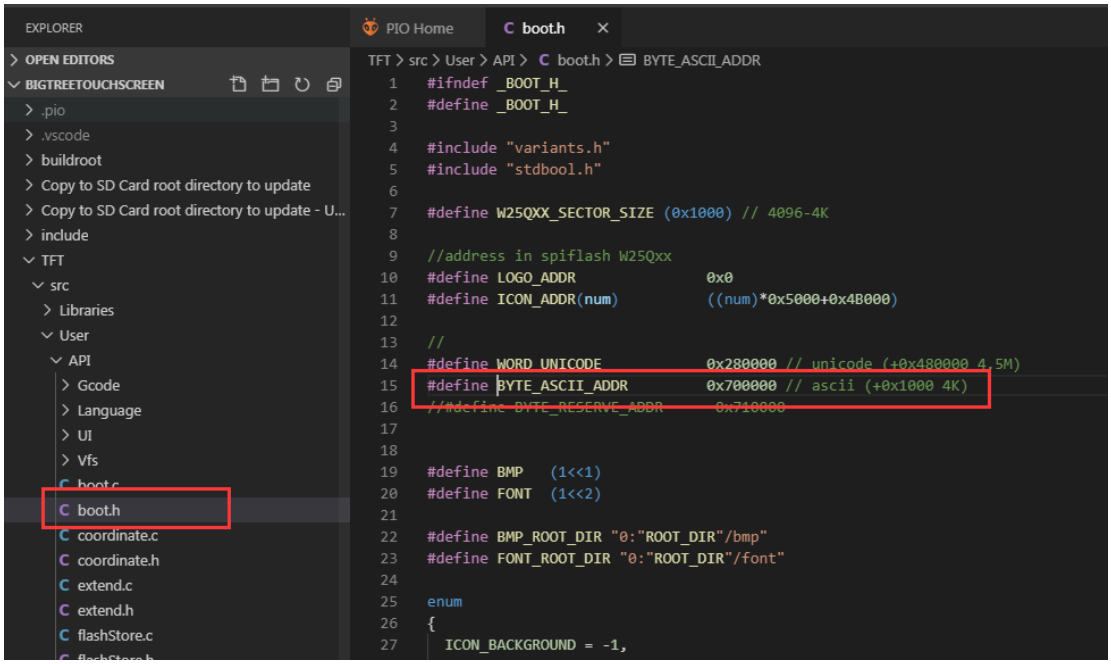
1. First, find the encoding range of the character you want to display in unicode
https://www.unicode.org/charts/

For example
Basic Latin(ASCII) is 0x00~0x7F https://www.unicode.org/charts/PDF/U0000.pdf
Cyrillic(Russia) is 0x400~0x4FF https://www.unicode.org/charts/PDF/U0400.pdf
Armenian is 0x530~0x58F https://www.unicode.org/charts/PDF/U0530.pdf
Latin extended (Czech/French/etc…)

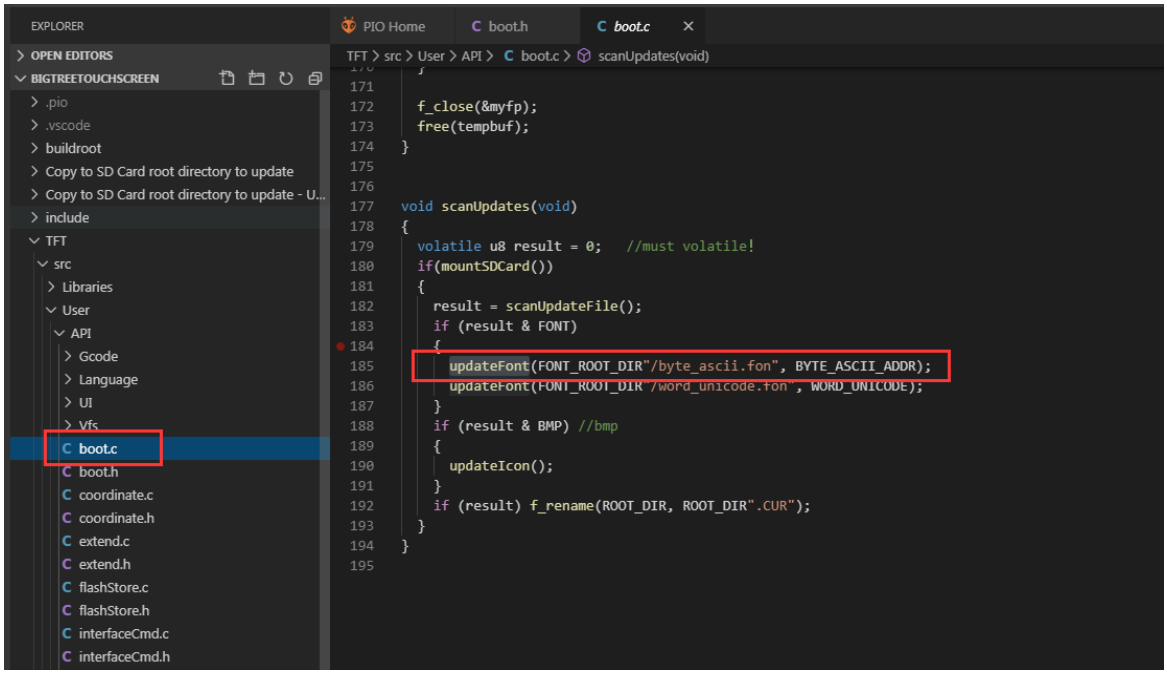
2. Let's take ASCII as an example. You need to create a dot matrix font with all characters in ASCII. The scanning method of dot matrix fonts requires first from top to bottom, then from left to right, and the high position is first. The default size of the font is high*width 24*12 (TFT35), 16*8 (TFT24/TFT28), and the font size can be modified in variants.h



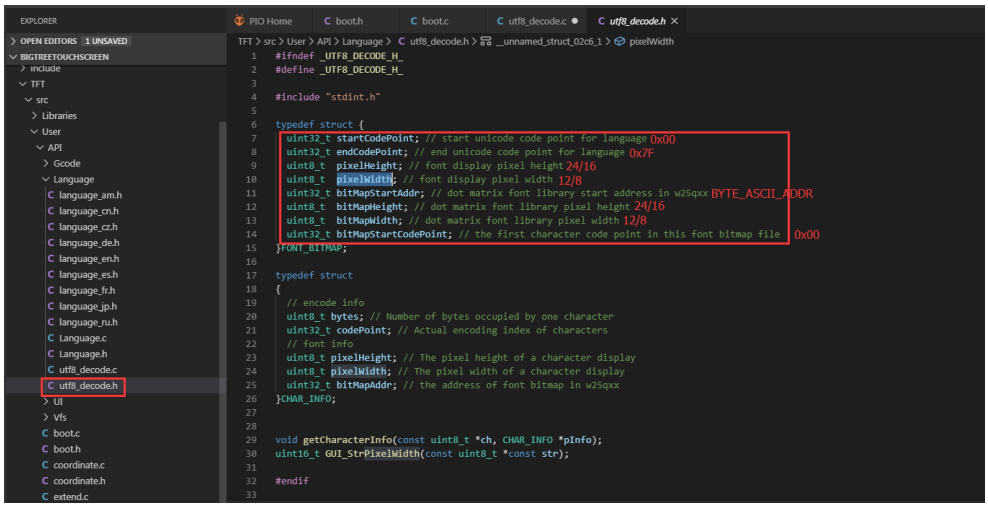
3. Set the dot matrix font in boot.h to store the starting address in SPI Flash (note the total size of the font file, do not overlap with other font addresses, and the total capacity of Flash is 8MByte, the ending address is
0x800000)

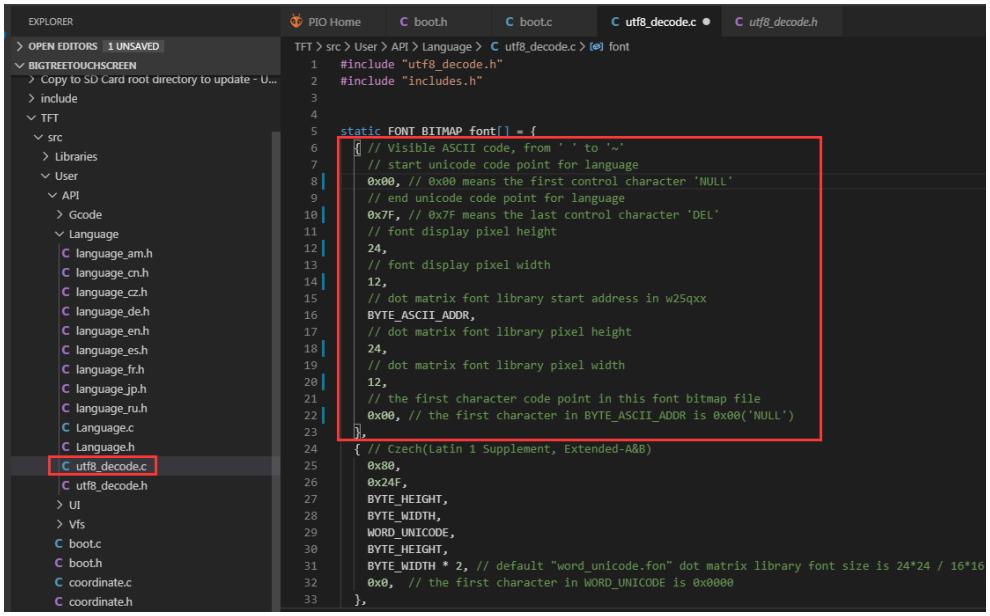


4. Add the ability to update fonts to SPI Flash in boot.c



5. In the static FONT_BITMAP font[] array of the utf8_decode.c file, add the character encoding to be parsed. The information to be added is as follows





6. Compile, generate and update new firmware, change the name of the font file to the name "byte_ascii.fon" set in the firmware boot.c, put it in the "TFT35 (TFT28, TFT24)/font" folder of the SD card, and then put the SD Insert the card into the card slot of the touch screen, reset the font file, then switch to the language you want in the settings to use your customized font.