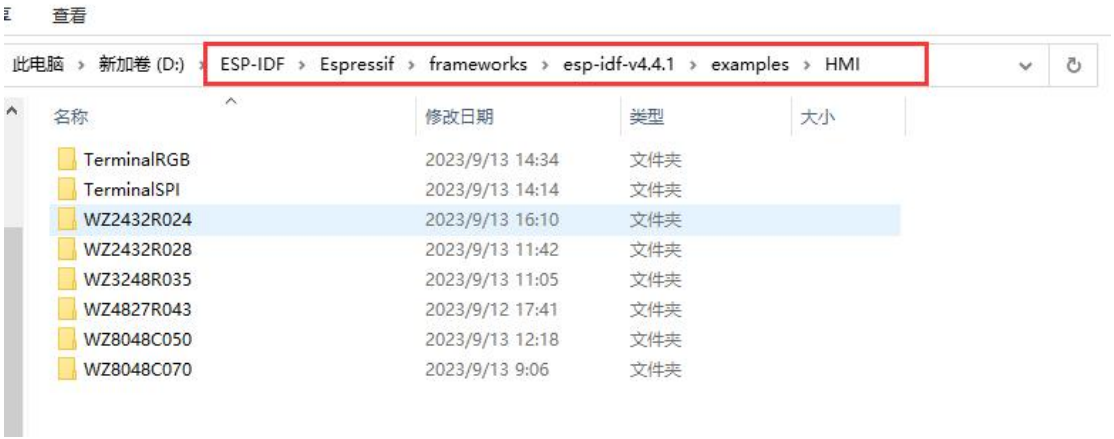
WZ2432R028 Use a tutorial

Place the downloaded project under the IDF directory (as shown below):



Let's first learn at the use of commands：

cd xxx---Moving to the xxx directory, xxx represents the name of the directory, for example: cd example

idf.py set-target esp32s3---Set the target chip for example: esp32s3

idf.py fullclean---Delete the entire build directory, including all the CMake configuration output files.

idf.py clean---It removes the building output files from the building directory and cleans up the entire project..

idf.py menuconfig---Configure the target chip

idf.py build---Compile a private code base

idf.py -p com3 flash---Download the program to the target chip

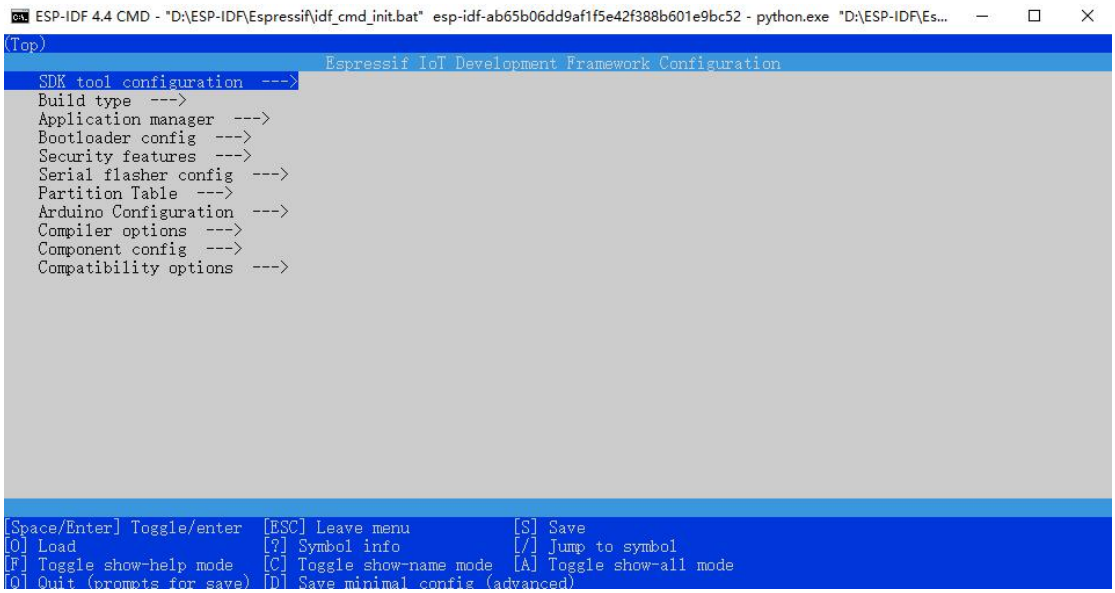
idf.py -p com3 flash monitor---Once compile burn and open monitoring

Now we open the terminal and go to the WZ2432R028 project catalog

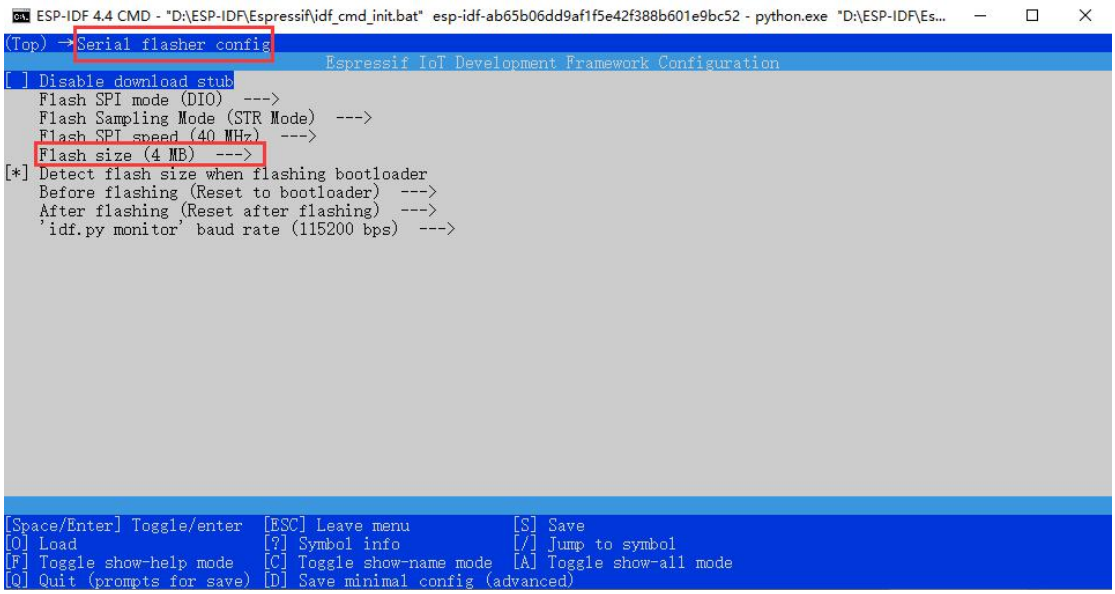
```
D:\ESP-IDF\Espressif\frameworks\esp-idf-v4.4.1\examples\HMI>cd WZ2432R028

D:\ESP-IDF\Espressif\frameworks\esp-idf-v4.4.1\examples\HMI\WZ2432R028>
```

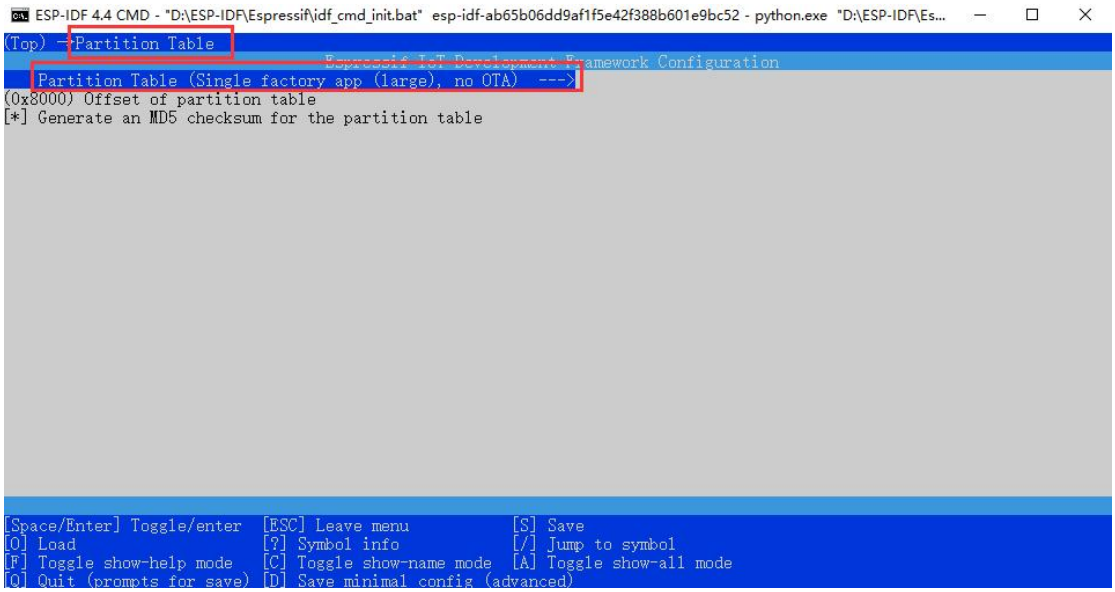
Now we have to empty the project idf.py fullclean once first, and then go into the configuration

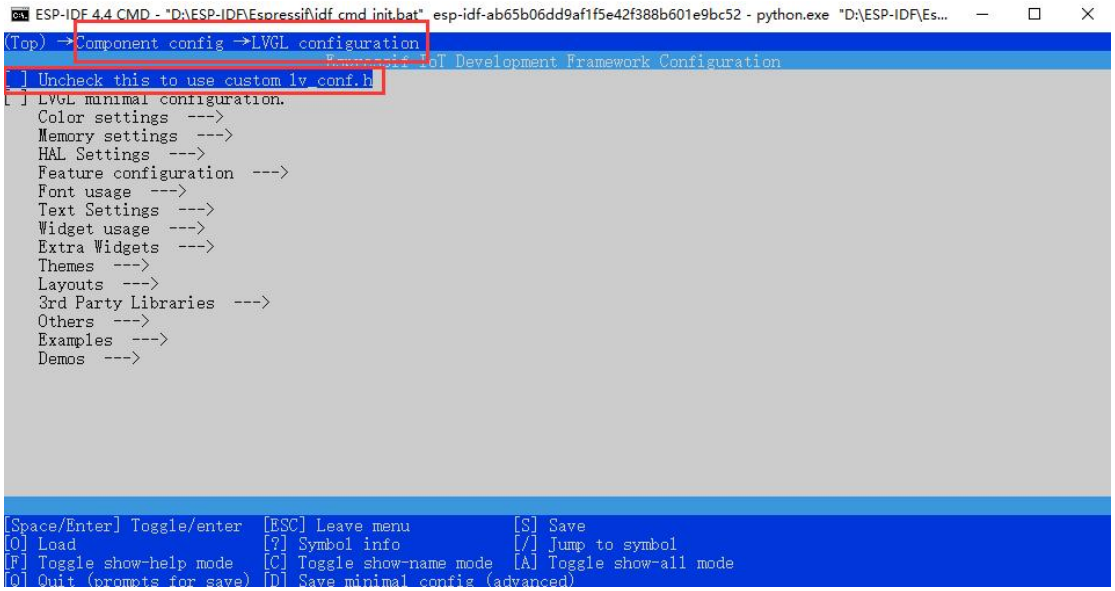
```
D:\ESP-IDF\Espressif\frameworks\esp-idf-v4.4.1\examples\HMI\WZ2432R028>idf.py fullclean
Executing action: fullclean
Done
```

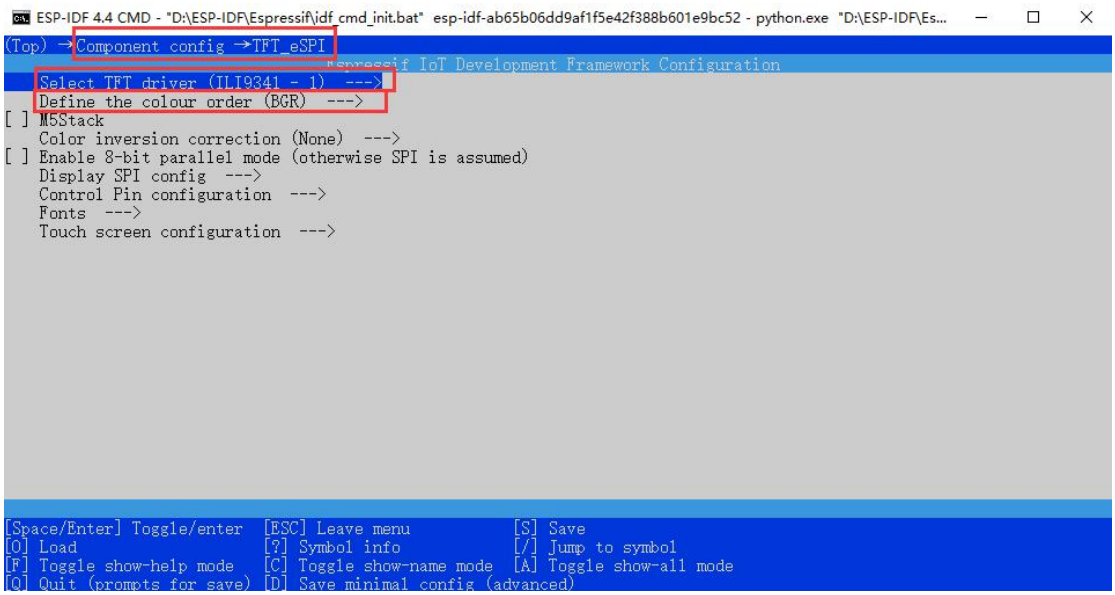


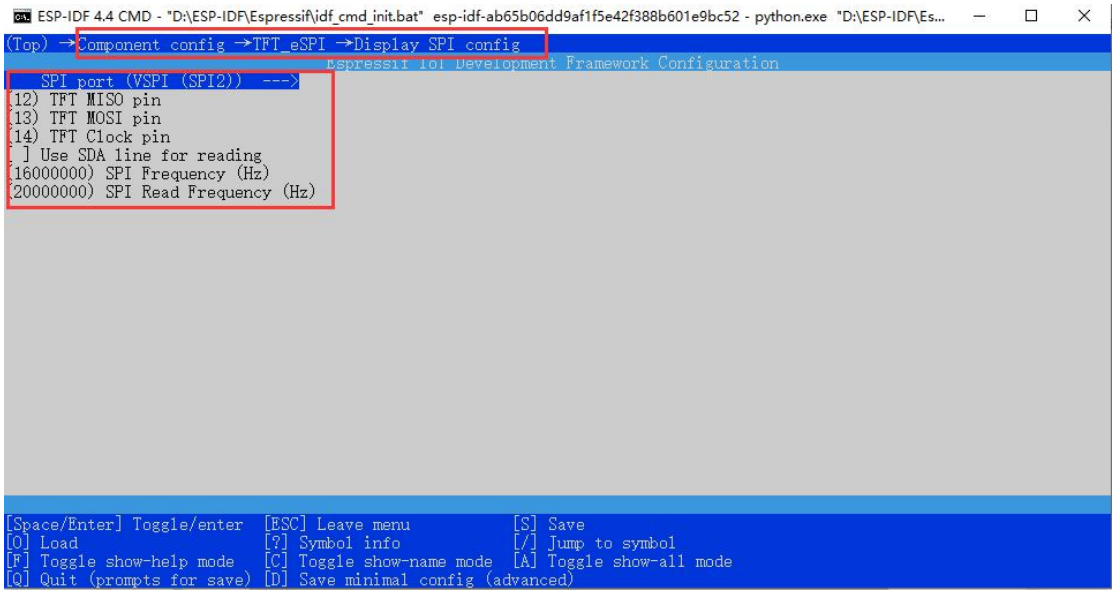
Now modify the options by following the following steps:

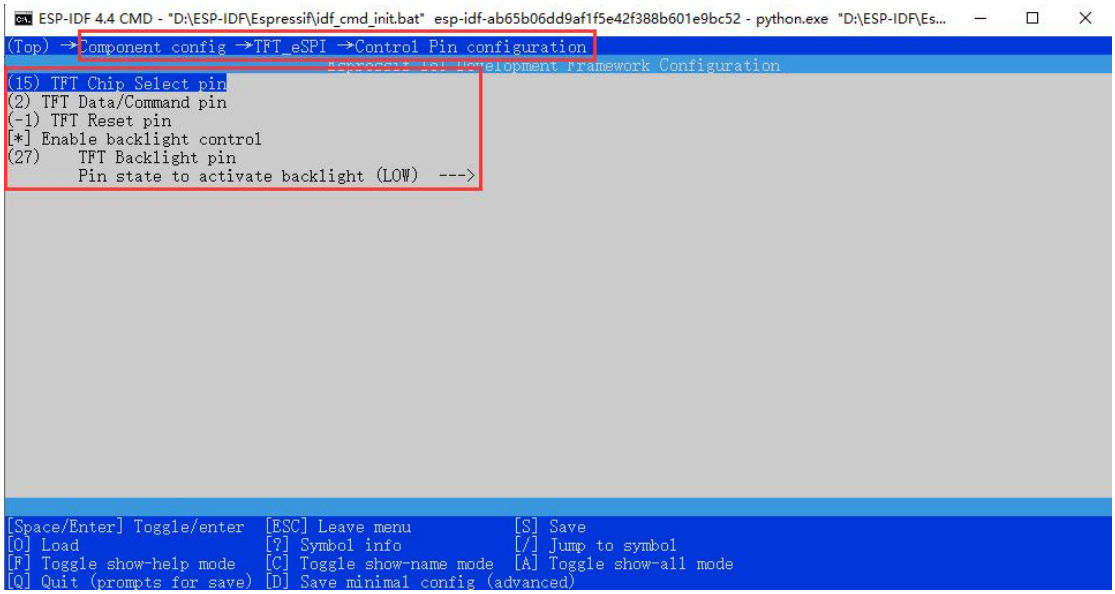


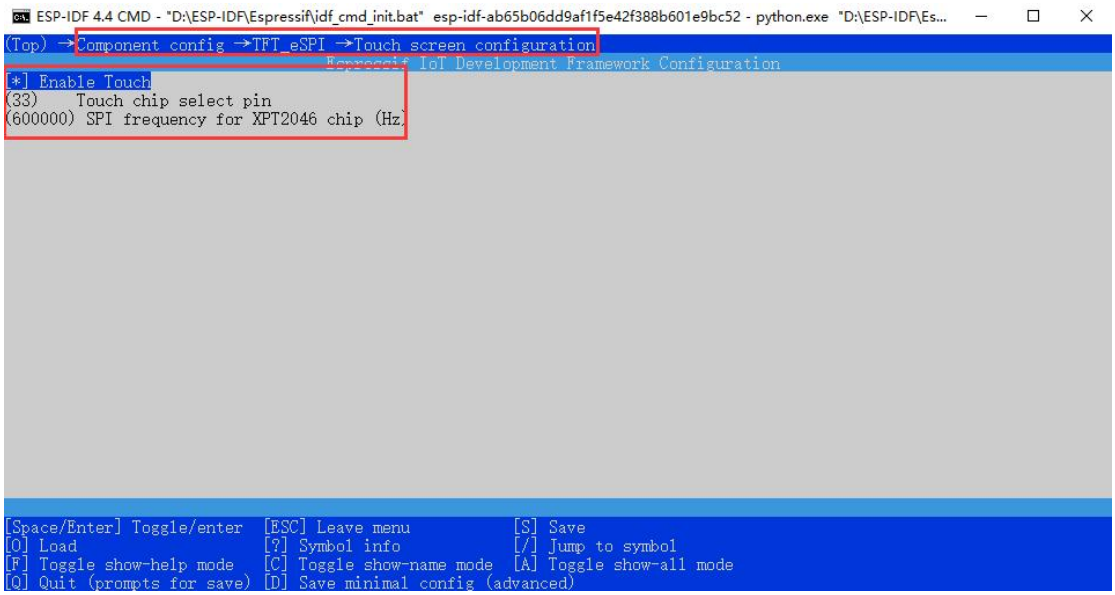

ESP-IDF 4.4 CMD - "D:\ESP-IDF\Espressif\idf_cmd_init.bat" esp-idf-ab65b06dd9af1f5e42f388b601e9bc52 - python.exe "D:\ESP-IDF\Es...
(Top) → Partition Table
Espressif IoT Development Framework Configuration
Partition Table (Single factory app (large), no OTA) --->
(0x8000) Offset of partition table
[*] Generate an MD5 checksum for the partition table
[Space/Enter] Toggle/enter [ESC] Leave menu [S] Save
[O] Load [?] Symbol info [/] Jump to symbol
[F] Toggle show-help mode [C] Toggle show-name mode [A] Toggle show-all mode
[Q] Quit (prompts for save) [D] Save minimal config (advanced)


ESP-IDF 4.4 CMD - "D:\ESP-IDF\Espressif\idf_cmd_init.bat" esp-idf-ab65b06dd9af1f5e42f388b601e9bc52 - python.exe "D:\ESP-IDF\Es...
(Top) → Component config → LVGL configuration
Espressif IoT Development Framework Configuration
[ ] Uncheck this to use custom lv_conf.h
[ ] LVGL minimal configuration.
Color settings --->
Memory settings --->
HAL Settings --->
Feature configuration --->
Font usage --->
Text Settings --->
Widget usage --->
Extra Widgets --->
Themes --->
Layouts --->
3rd Party Libraries --->
Others --->
Examples --->
Demos --->
[Space/Enter] Toggle/enter [ESC] Leave menu [S] Save
[O] Load [?] Symbol info [/] Jump to symbol
[F] Toggle show-help mode [C] Toggle show-name mode [A] Toggle show-all mode
[Q] Quit (prompts for save) [D] Save minimal config (advanced)

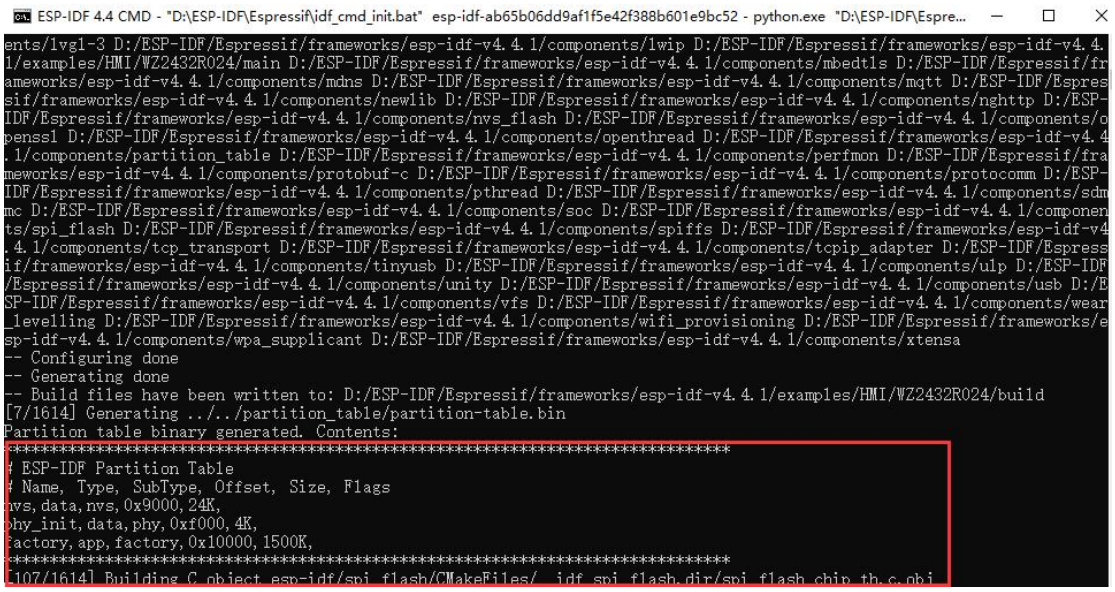

ESP-IDF 4.4 CMD - "D:\ESP-IDF\Espressif\idf_cmd_init.bat" esp-idf-ab65b06dd9af1f5e42f388b601e9bc52 - python.exe "D:\ESP-IDF\Es...
(Top) → Component config → TFT_eSPI
Espressif IoT Development Framework Configuration
Select TFT driver (ILI9341 - 1) --->
Define the colour order (BGR) --->
[ ] M5Stack
Color inversion correction (None) --->
[ ] Enable 8-bit parallel mode (otherwise SPI is assumed)
Display SPI config --->
Control Pin configuration --->
Fonts --->
Touch screen configuration --->
[Space/Enter] Toggle/enter [ESC] Leave menu [S] Save
[O] Load [?] Symbol info [/] Jump to symbol
[F] Toggle show-help mode [C] Toggle show-name mode [A] Toggle show-all mode
[Q] Quit (prompts for save) [D] Save minimal config (advanced)


ESP-IDF 4.4 CMD - "D:\ESP-IDF\Espressif\idf_cmd_init.bat" esp-idf-ab65b06dd9af1f5e42f388b601e9bc52 - python.exe "D:\ESP-IDF\Es...
(Top) → Component config → TFT_eSPI → Display SPI config
Espressif IoT Development Framework Configuration
SPI port (VSPI (SPI2)) --->
(12) TFT MISO pin
(13) TFT MOSI pin
(14) TFT Clock pin
[ ] Use SDA line for reading
(16000000) SPI Frequency (Hz)
(20000000) SPI Read Frequency (Hz)
[Space/Enter] Toggle/enter [ESC] Leave menu [S] Save
[O] Load [?] Symbol info [/] Jump to symbol
[F] Toggle show-help mode [C] Toggle show-name mode [A] Toggle show-all mode
[Q] Quit (prompts for save) [D] Save minimal config (advanced)

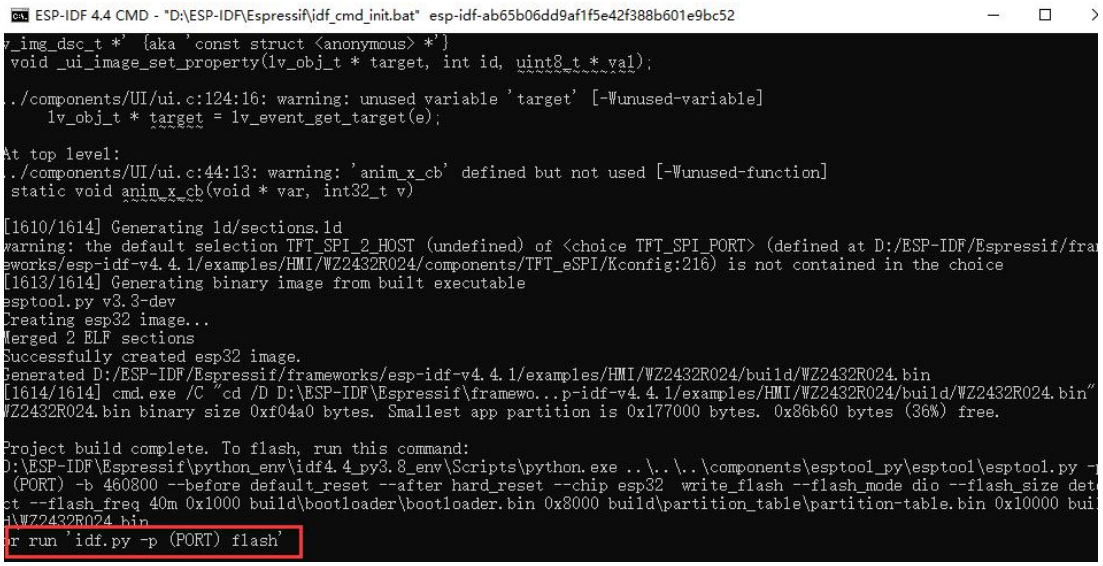

ESP-IDF 4.4 CMD - "D:\ESP-IDF\Espressif\idf_cmd_init.bat" esp-idf-ab65b06dd9af1f5e42f388b601e9bc52 - python.exe "D:\ESP-IDF\Es...
(Top) → Component config → TFT_eSPI → Control Pin configuration
(15) TFT Chip Select pin
(2) TFT Data/Command pin
(-1) TFT Reset pin
[*] Enable backlight control
(27) TFT Backlight pin
Pin state to activate backlight (LOW) --->
[Space/Enter] Toggle/enter [ESC] Leave menu [S] Save
[O] Load [?] Symbol info [/] Jump to symbol
[F] Toggle show-help mode [C] Toggle show-name mode [A] Toggle show-all mode
[Q] Quit (prompts for save) [D] Save minimal config (advanced)


ESP-IDF 4.4 CMD - "D:\ESP-IDF\Espressif\idf_cmd_init.bat" esp-idf-ab65b06dd9af1f5e42f388b601e9bc52 - python.exe "D:\ESP-IDF\Es...
(Top) → Component config → TFT_eSPI → Touch screen configuration
Espressif IoT Development Framework Configuration
[*] Enable Touch
(33) Touch chip select pin
(600000) SPI frequency for XPT2046 chip (Hz)
[Space/Enter] Toggle/enter [ESC] Leave menu [S] Save
[O] Load [?] Symbol info [/] Jump to symbol
[F] Toggle show-help mode [C] Toggle show-name mode [A] Toggle show-all mode
[Q] Quit (prompts for save) [D] Save minimal config (advanced)

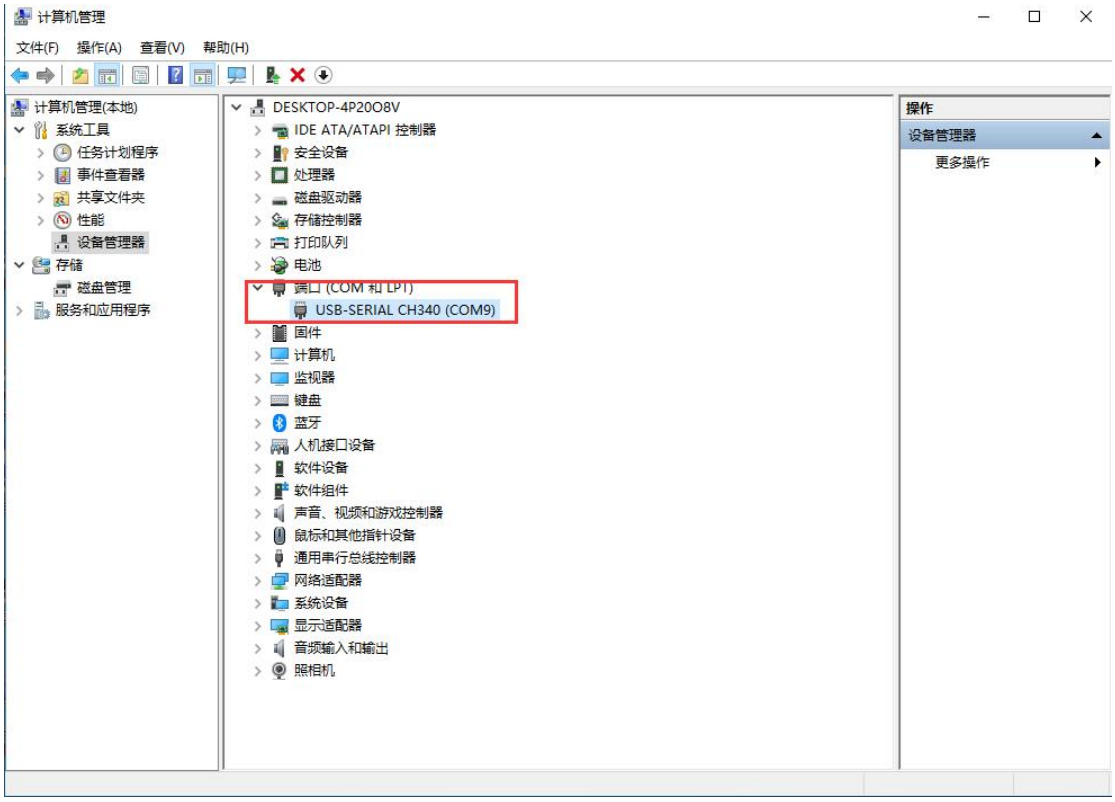
Save the exit after setup, and then execute the idf.py build



Waiting for the compilation to complete, the following figure interface appears:



Perform the idf.py-p com9 flash



success!