**ESP32-CAM uses the Arduino IDE for video streaming and photography**

directory

[**ESP32-CAM uses Arduino IDE for video streaming and**  **photography1**](#_Toc61701337)

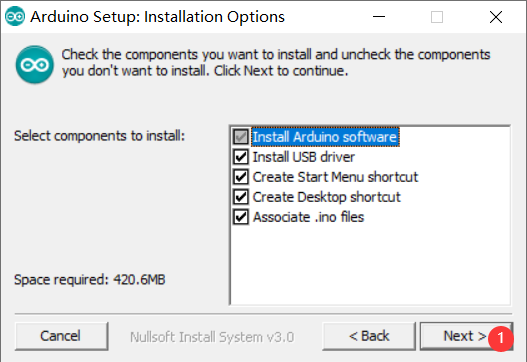
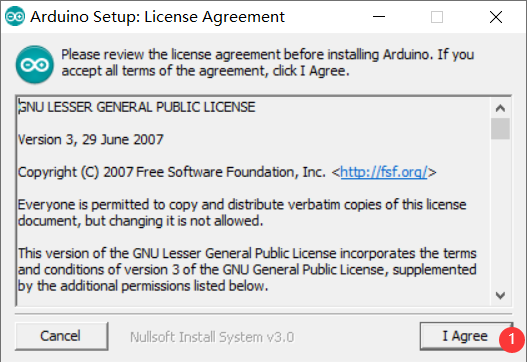
[First, Arduino development environment construction 2](#_Toc61701338)

[ESP-CAM video streaming and photo example 7](#_Toc61701339)

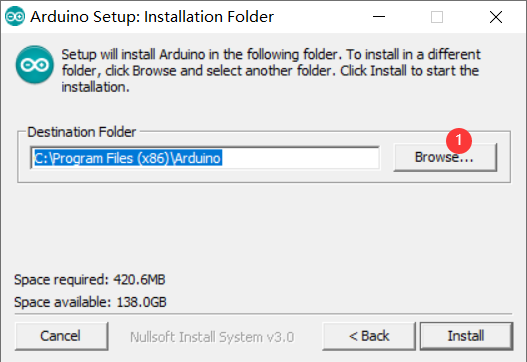
# First, Arduino development environment construction

1. Download the Arduino IDE on the official Arduino website

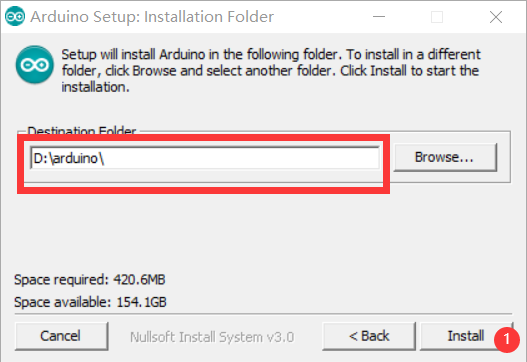
(2) Click I agree and click Next



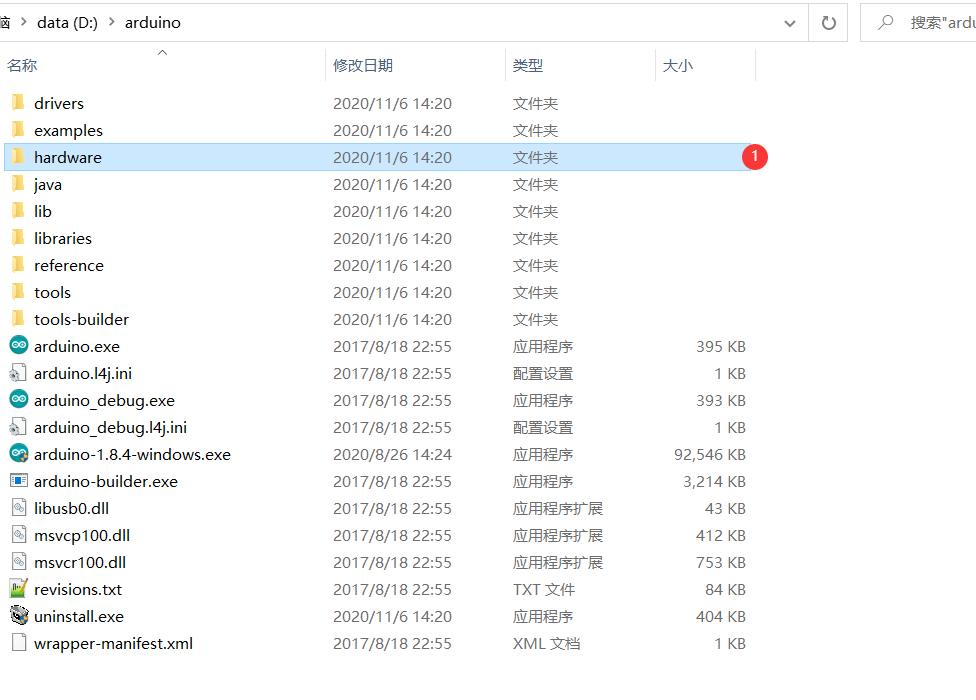
(3) The default is to install on C drive, you can click Browse to change the installation location



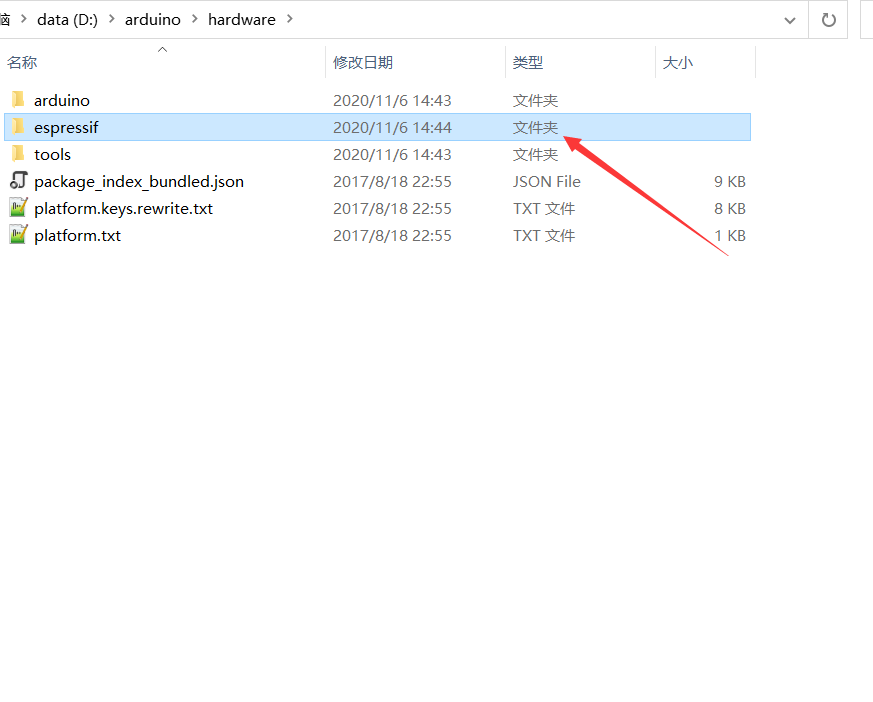
(4) Here I installed in the Arduino folder on the D drive (remember this installation path).



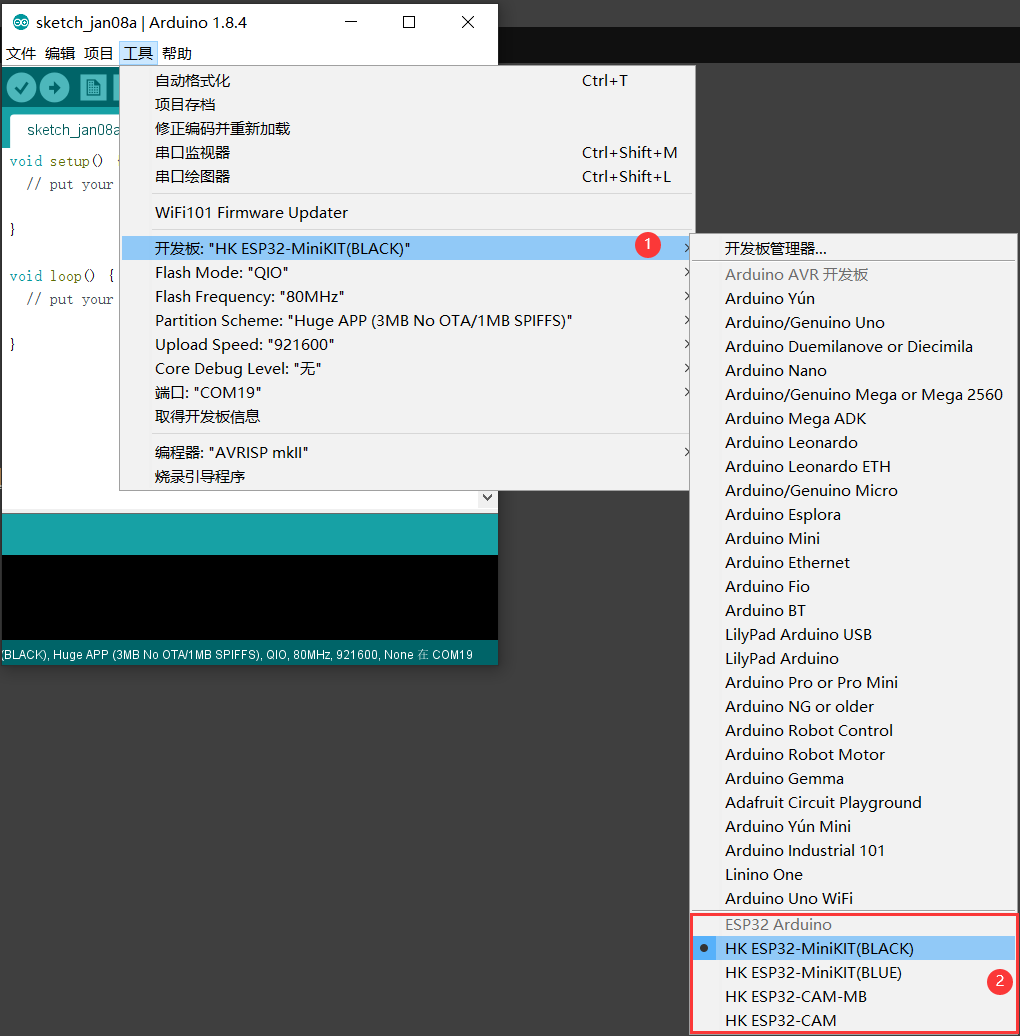
(5) Wait for the installation to complete and open the hardware under the installation path



(6) Download SDK: https://pan.baidu.com/s/1BXT0BUuBzCrGQT9FAb5qXw (extraction code: 8ita), download and extract to the hardware directory



(7) Open ArduinoIDE and select the corresponding development board

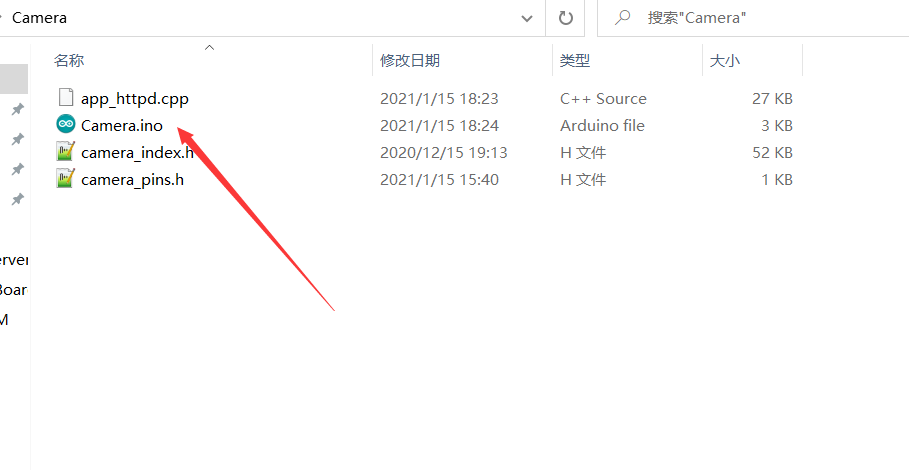


At this point, the Arduino IDE development platform has been built, and the following describes how to use the example

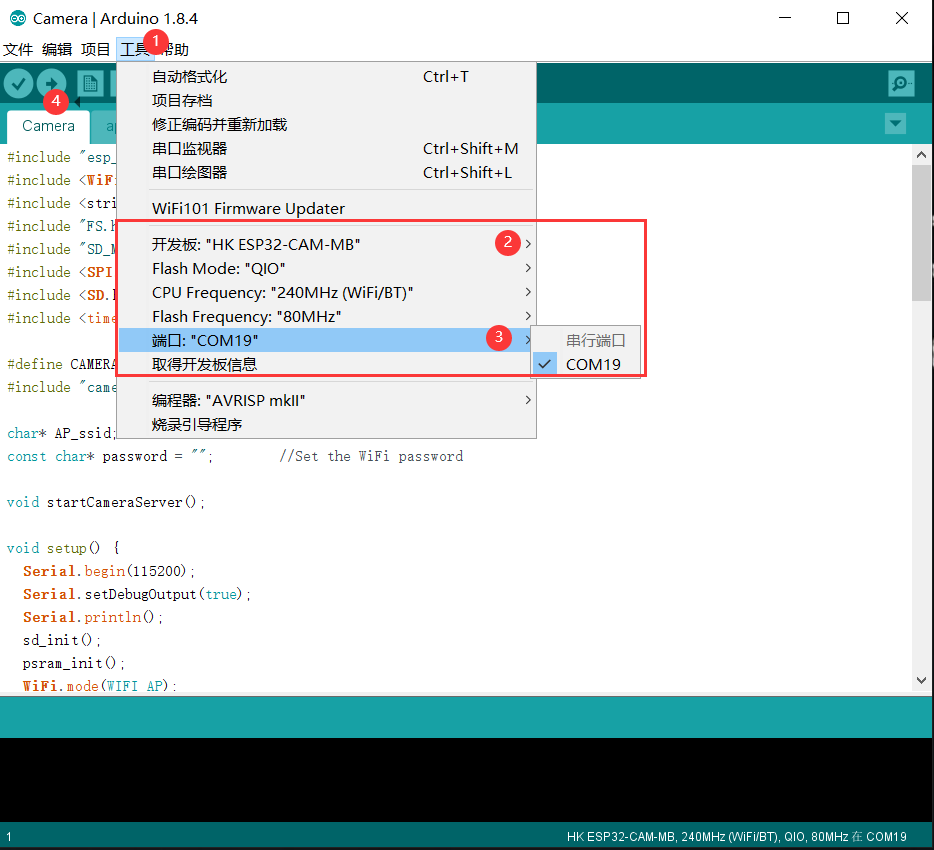
# 2. ESP-CAM video streaming and photo example

(1) Download the Camera example: <https://pan.baidu.com/s/1VSRq1r4uuFC9FuJgUIHqwg> (extraction code: 7cld).

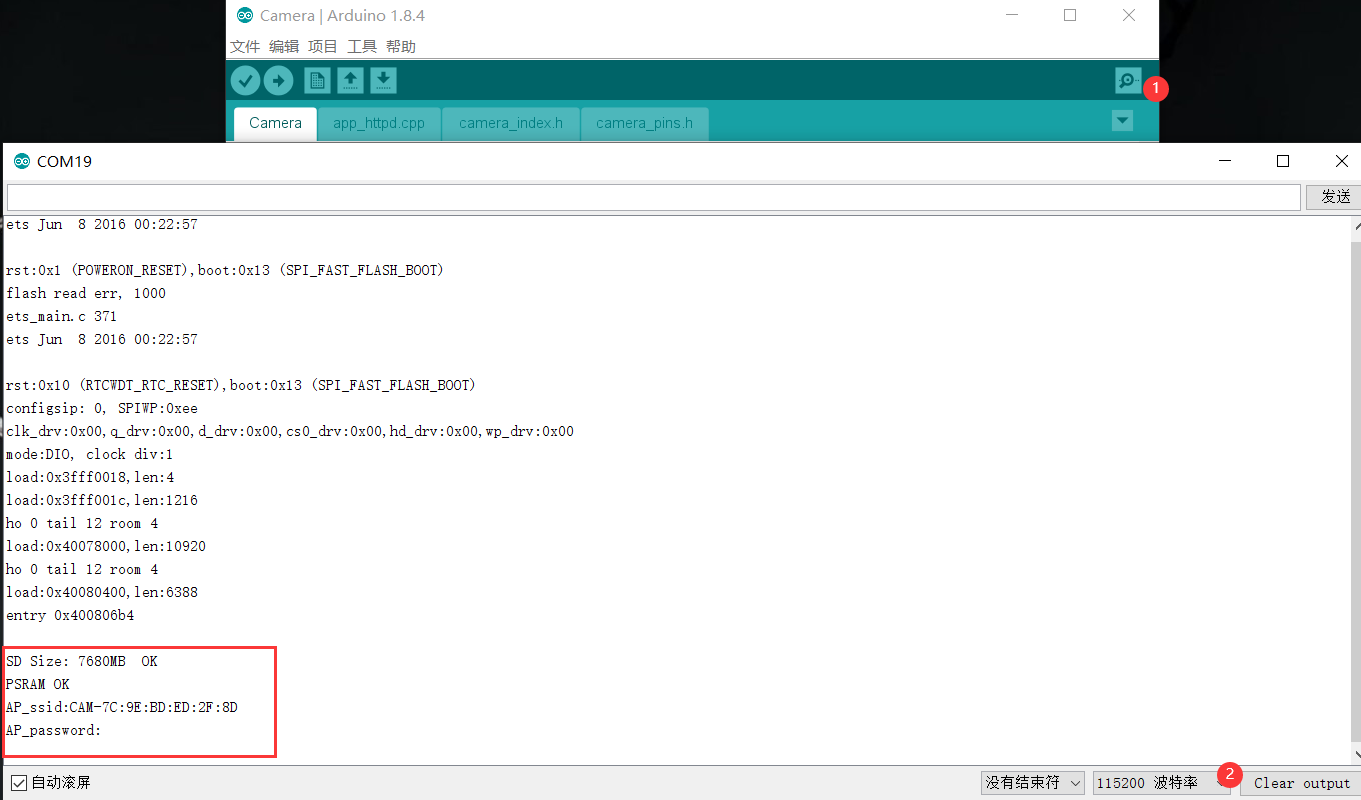
(2) After the download is complete, go to the "Camera" directory and double-click to open "Camera.ino"



(3) Select the development board, select the port, and upload the code



(4) \*\*Open the serial port, select the baud rate of 115200, reset the development board, and view the serial port output\*\*



(5) Access the video streaming server

You can turn on your mobile phone WIFI and connect "ESP-xx:xx:xx" with your mobile phone (the suffix is different for each development board). Open a browser and type the 192.168.4.1 address. Press the Start Streaming button to start the video streaming. Press Get Still to take a photo, the photo is stored in the SD card, and the photo name is randomly named

