ESP32-Camera Car Tutorial

Perface

Our Company

ACEBOTT STEM Education Tech Co.,Ltd
Founded in China's Silicon Valley in 2013, ACEBOTT is a STEM education solution leader We have a team of 150 individuals, including members from research and development, sales, and logistics. Our goal is to provide high-quality STEM education products and services to our customers. We are working together with STEM education experts and our business partners to produce successful STE products together Our self-own factory also provides CEM services for our clients including logo customization on product packaging and PCB.

Our Tutorial

This is a hands-on practical course designed specifically for beginners, aimed at introducing students to the world of programming, electronics, and robotics through an ESP32-based camera module. In this course, students will learn both the theoretical knowledge and practical applications of imaging with the ESP32 camera module, covering operations from web imaging to app imaging.

Through this kit, you can:

1. Explore the component structure of the camera module and understand how they work together in the imaging process.
2. Learn about the imaging process of the camera, including image capture, generation of image signals, image signal processing, image compression, image transmission, image decoding, and image formation.
3. Study the concept of wireless communication, including WiFi communication technology, IP addresses, TCP protocol, ESP32 communication methods, etc.
4. Design the functionality for displaying images in an app for the camera module, and learn the process and methods of app development.
5. Enhance your maker skills by building a wireless transmission car project using the ACEBOTT kit, following step-by-step tutorials.

In summary, the ACEBOTT camera module learning kit is specifically designed for beginners to learn about wireless image transmission based on ESP32. With this kit, students will be able to independently design and program a smart car with image transmission capabilities, understand the basic principles of smart hardware operation, and have the ability to apply the knowledge learned to solve real-world problems.

Customer service

ACEBOTT is a dynamic and fast-growing STEM education technology company that strives to offer excellent products and quality services that meet your expectations. We value your feedback and encourage you to drop us a line at support@acebott.com with any comments or suggestions you may have.

Our experienced engineers are dedicated to promptly addressing any problems or questions you may have about our products. We guarantee a response within 24 hours during business days.

Follow Us

Scan the QR codes to Follow Us for troubleshooting & the latest news.
We have a very large community that is very helpful for troubleshooting and we also have a support team at the ready to answer any questions.

ACEBOTT FB QR Code

YouTube QR Code

Contents

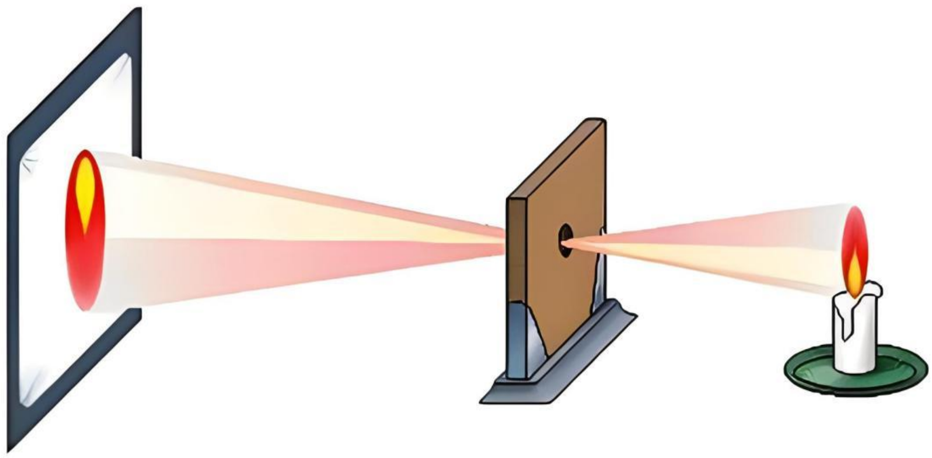
Lesson 1: Development and Components of Photography

Ⅰ.The development history of photography technology

The history of photography is long and has mainly undergone five stages: pinhole imaging, camera obscura drawing, daguerreotype photography, film cameras, and digital cameras.

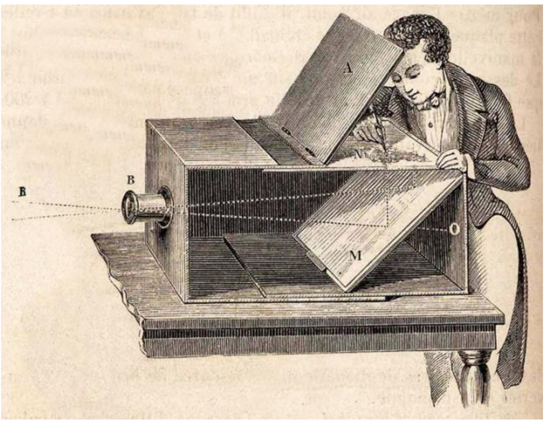
1.The first stage: Pinhole Imaging

Using a board with a small hole placed between a screen and an object will result in an inverted image of the object appearing on the screen. This phenomenon is known as pinhole imaging. Moving the board back and forth will change the size of the image, reflecting the nature of light propagation in straight lines, which laid the scientific foundation for the birth of subsequent cameras.



2.The second stage: Camera Obscura drawing

In the late fifteenth century, people constructed camera obscuras based on the principles of pinhole imaging, which served as the precursor to cameras. At that time, individuals utilized these camera obscuras to reflect images onto drawing paper using pencils, outlining contours, and then coloring to create lifelike and proportionate drawings.



3.The third stage: daguerreotype photography

In 1838, French physicist Louis Daguerre invented the Daguerreotype, a photographic process that involved exposing a steel plate coated with silver iodide in a camera obscura. The plate was then developed using mercury vapor and fixed with common salt. This method produced highly detailed images that could be permanently preserved. Subsequently, based on this method, Daguerre constructed the world's first camera.

4.The fourth stage: Film Cameras

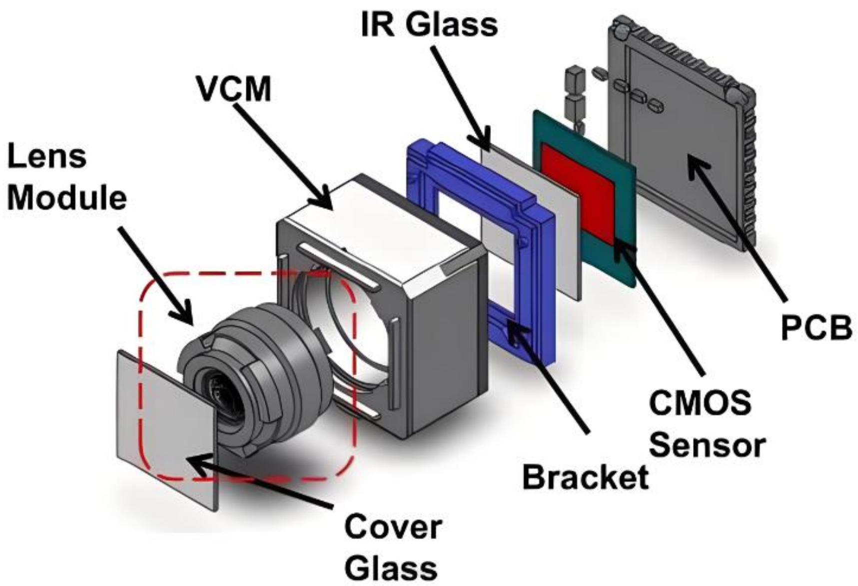
In 1888, the Eastman Dry Plate Company in the United States produced a new type of photosensitive material – flexible and rollable "film." The increased portability of the film marked a new era for photosensitive materials. In 1900, Kodak introduced the portable camera, the Brownie, which seamlessly combined film and camera, leading cameras to evolve towards smaller and lighter designs.

5.The fifth stage: Digital Cameras

Digital cameras are a type of camera that utilizes electronic sensors to convert optical images into electronic data. When light enters the camera through the lens, it is converted into digital signals by the imaging sensor in the digital camera. These digital signals are then processed and stored by computational chips. The imaging sensor in digital cameras is typically either a CCD (Charge-Coupled Device) or a CMOS (Complementary Metal-Oxide-Semiconductor). These sensors have the characteristic of converting light into electronic signals based on the intensity of the light.

Ⅱ.Camera structure

The structure of a camera mainly includes: lens, voice coil motor, filter, image sensor, and PCB circuit board.

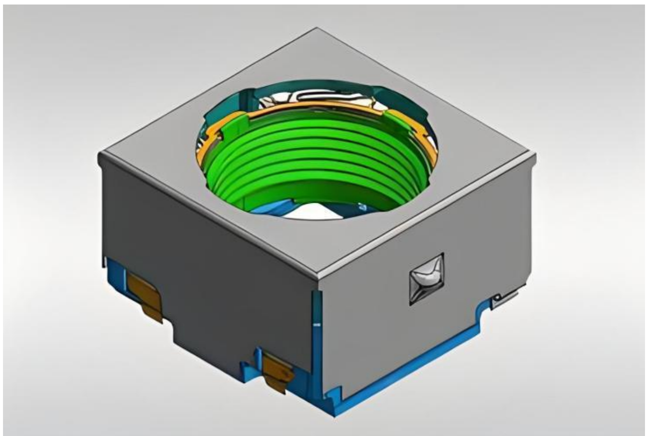


1.Lens

A lens is a transparent optical component composed of one or multiple curved optical glass pieces. The material is typically plastic or glass. Its overall effect is that of a convex lens, mainly used to converge light rays so that the light from the scene can be focused onto the photosensitive element.

2.Voice Coil Motor

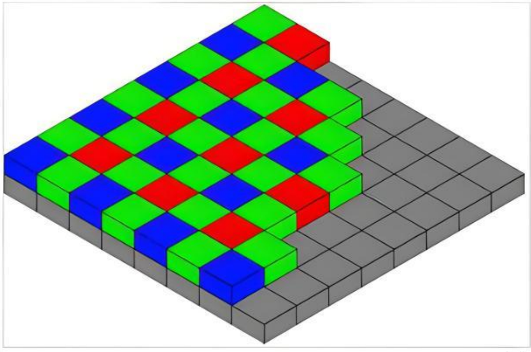
In optical instruments, a voice coil motor is commonly used for functions such as lens positioning and autofocus. Its operating principle involves controlling electrical currents to drive spring movement, adjusting the position of the lens along the XYZ axes to achieve the clearest focus on the subject.



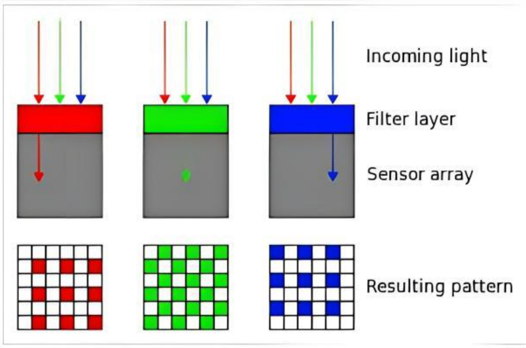
3.Filter

The infrared cutoff filter is placed between the lens and the image sensor, primarily used for filtering light. Because the range of wavelengths of visible light perceived by the human eye is smaller than that recognized by the image sensor, the image sensor may receive infrared light, resulting in images that differ from what the human eye sees. Therefore, adding an infrared cutoff filter helps to make the images more similar to what the human eye perceives.

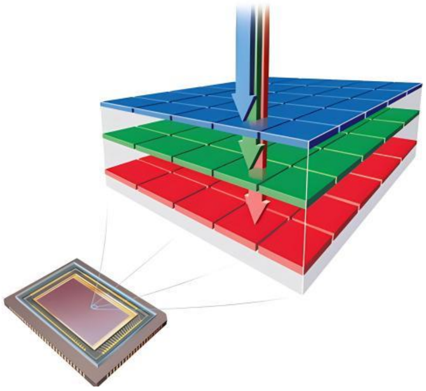
Because image sensors can only detect the intensity of light and cannot perceive its wavelength, image sensors can only capture black and white photos. Later, Kodak engineer Bryce Bayer designed the Bayer filter, in which he placed a layer of color filter array above the image sensor. Each pixel can only capture information about one color component. Then, based on this color component information, a full-color image is obtained through interpolation algorithms.



The Bayer arrangement of color filters on the pixel array of an image sensor



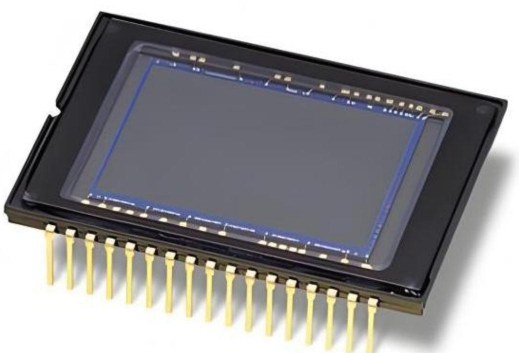
Profile/cross-section of sensor



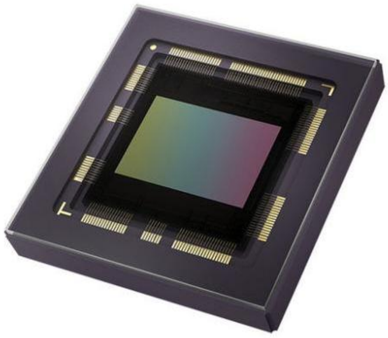
4.Image Sensor

Its main function is to convert light signals into electrical signals, and there are currently two widely used types: one is CCD (Charge-Coupled Device) components; the other is CMOS (Complementary Metal-Oxide-Semiconductor) devices.

CCD is made of a highly photosensitive semiconductor material and contains many photosensitive elements, usually in units of millions of pixels. When the surface of the CCD is illuminated by light, each photosensitive unit reflects the charge onto the component. The signals generated by all the photosensitive units are combined to form a complete image.



CMOS mainly utilizes semiconductors made of silicon and germanium, allowing for the coexistence of N-type (electron-rich) and P-type (hole-rich) semiconductors on the CMOS chip. The currents generated by them can be recorded and interpreted as images by the processing chip. Therefore, it is used as an image sensor in digital photography.

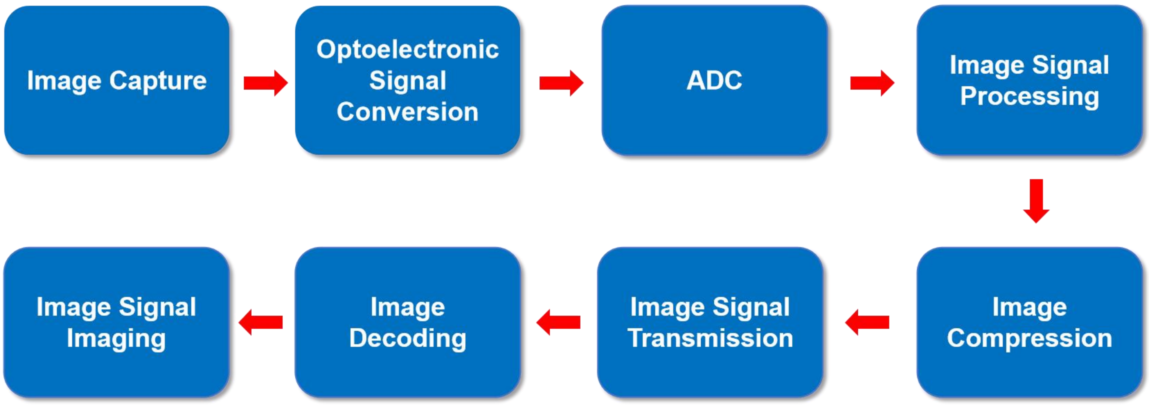


5.FPCB

The FPCB (Flexible Printed Circuit Board) serves to transmit the photoelectric signals from the sensor, connecting the camera module with the main processor and facilitating data transfer. It effectively reduces signal noise issues and improves the efficiency of image transmission.

Lesson 2: The generation of image signals

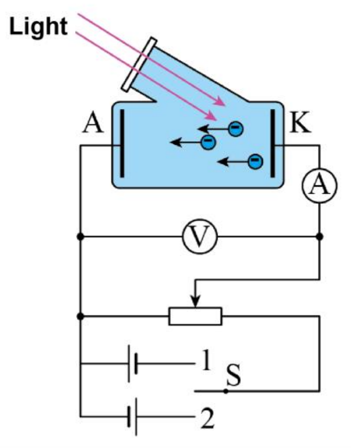
The imaging process of a camera is a complex procedure. To display external scenes on a screen, it involves several steps including image capture, photoelectric signal conversion, analog-to-digital conversion, image signal processing, image compression, image signal transmission, image decoding, and image rendering.



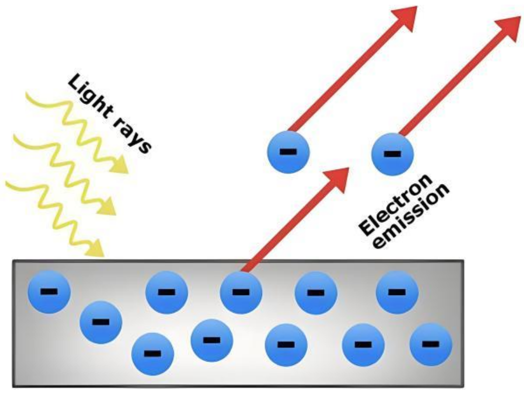
Ⅰ.The principle of photoelectric signal conversion.

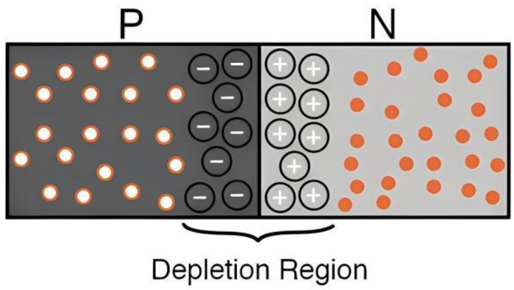
The reason why cameras can reproduce real images mainly relies on the photoelectric effect. This means that the camera can convert the light reflected from objects within the lens's field of view into electrical signals through the recognition of the image sensor.

The principle of the photoelectric effect is that certain materials, when exposed to electromagnetic waves above a specific frequency, exhibit the phenomenon where internal electrons absorb energy and are emitted, forming an electric current.



The majority of photoelectric detectors are made of silicon. When photons strike the surface of silicon, they dislodge electrons. These electrons move towards the positive terminal of the power source, generating a current and producing an electrical signal.





Ⅱ.The working principle of an image sensor

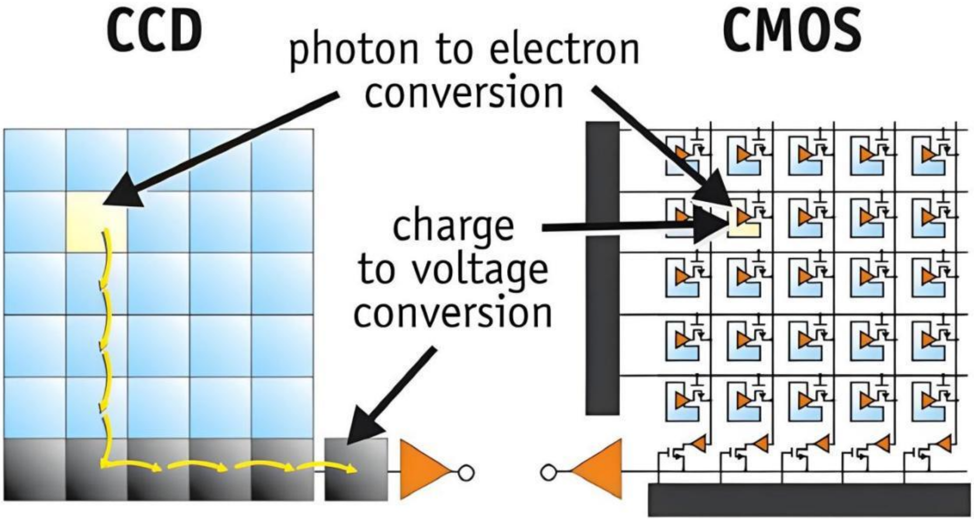
An image sensor is a semiconductor chip with several million to tens of millions of photodiodes on its surface. The function of the photodiode elements is to sense light signals and convert them into electrical signals. The working principle of an image sensor varies depending on its material.

1.The working principle of CCD

When a CCD is exposed to light, the photodiode capacitor of each pixel accumulates a certain amount of charge. After the exposure ends, the control circuitry surrounding the CCD transfers the charge of each capacitor in each column to the adjacent capacitor of the next pixel. Finally, through a charge amplifier at the end of the sensor, it is converted into a voltage signal.

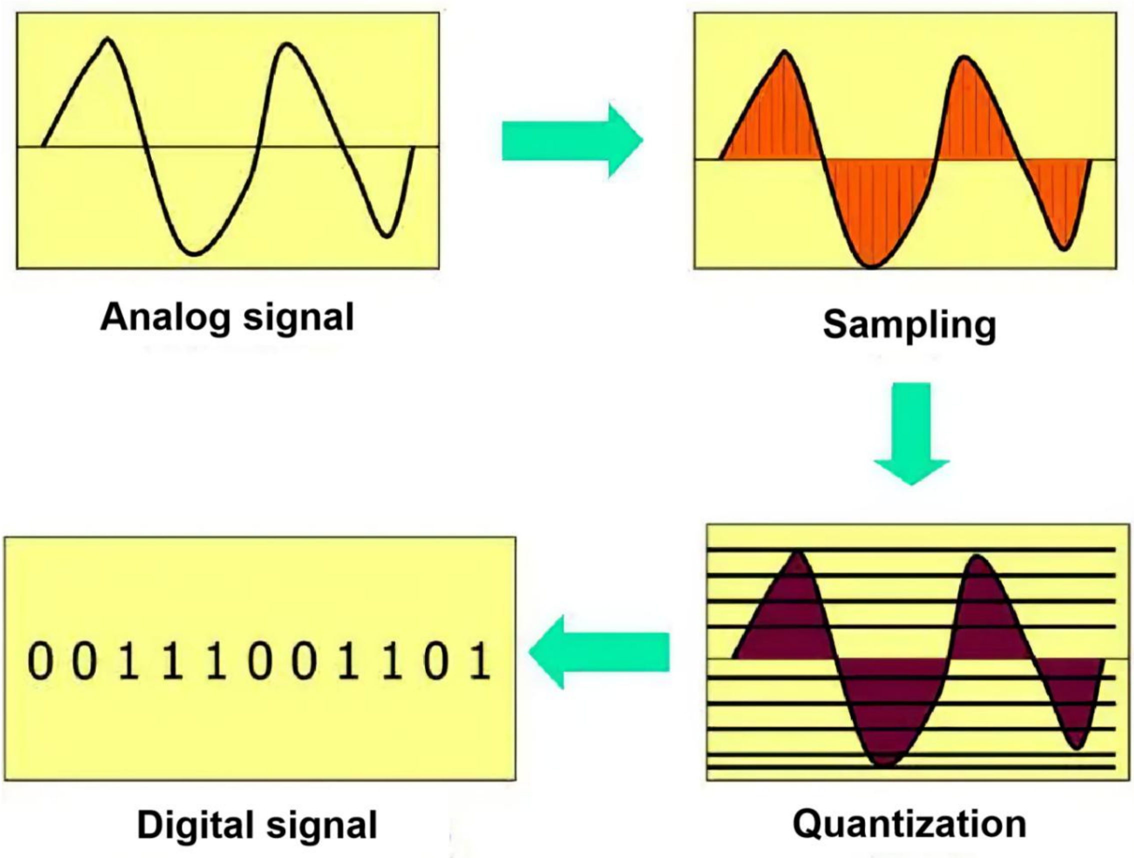
2.The working principle of CMOS

When a CMOS sensor is exposed to light, each pixel's photodiode is connected to an amplifier and ADC (Analog-to-Digital Converter) circuit. The voltage generated by the photodiode of pixels in the same row is simultaneously read out by the source follower connected to the photodiode, and then processed in parallel. The processed signals are stored in a row storage buffer and read out sequentially.



Ⅲ.Analog-to-Digital Conversion of Image Data

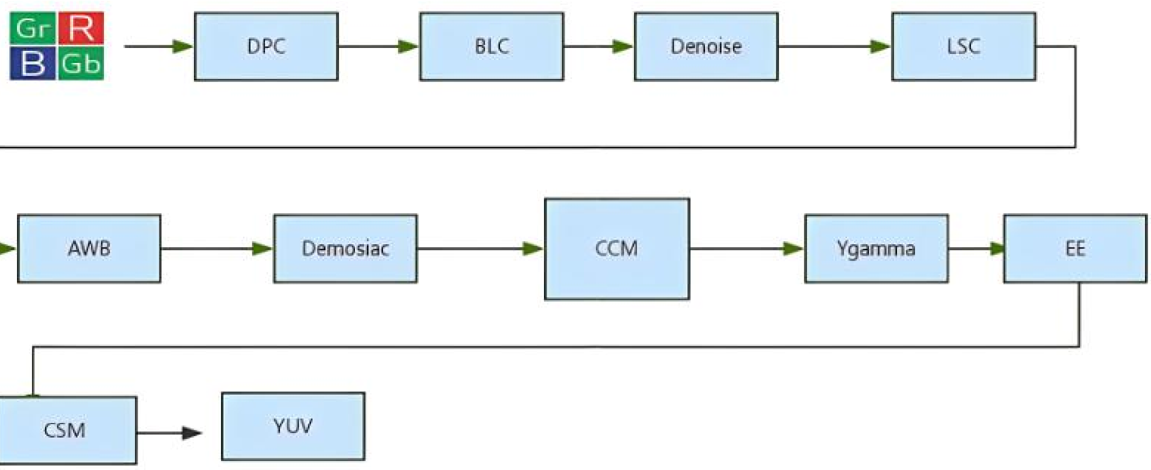
Analog-to-Digital Conversion (A/D) refers to the conversion of analog signals to digital signals. It is the process of converting continuous analog quantities into discrete sets of digital points. Through analog-to-digital conversion, data can be easily stored and processed.



Lesson 3: Image Signal Processing

Ⅰ.ISP Module

ISP stands for Image Signal Processing module, which is used to process the raw image data captured by the image sensor to output the final image. After the light signals are converted into electrical signals on the image sensor, the ISP module processes the received electrical signals by performing noise reduction, defective pixel removal, white balance adjustment, color correction, etc. The processed image is then stored and displayed.



Ⅱ.Defective Pixel Correction（DPC）

1.Bad pixel classification: The first type is dead pixels, which consistently display the darkest value; the second type is Bright pixels, which consistently display the brightest value; the third type is Drift pixels, whose behavior differs significantly from surrounding pixels.

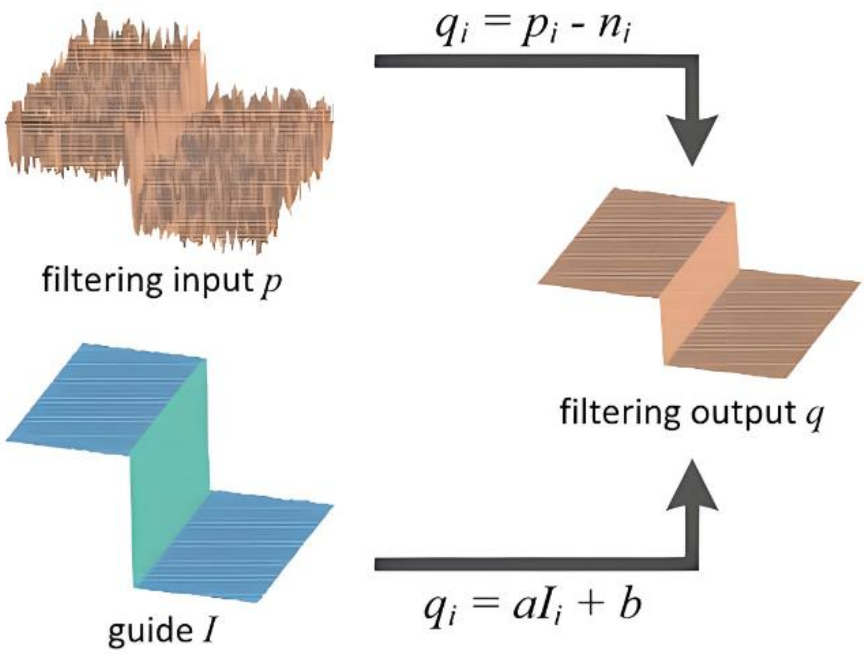
2.Correcting bad pixels: One method is automatic detection of bad pixels and automatic repair, while another method involves establishing a list of bad pixel coordinates for fixed-position repair.

Ⅲ.Black Level Calibration（BLC）

Dark current is the current observed when there is no light passing through the lens. It is a non-ideal factor in the sensor imaging process. Dark current is used to define the signal level corresponding to image data being 0. To reduce the impact of dark current on image signals, dark level calibration can be used to offset the overall signal output of the image sensor. The correction value is obtained by averaging the acquired image signals, and then subtracting this correction value from the pixels in the lower region. This corrects the dark level. Compared to images without dark level correction, images without it will appear brighter, affecting the image's contrast.

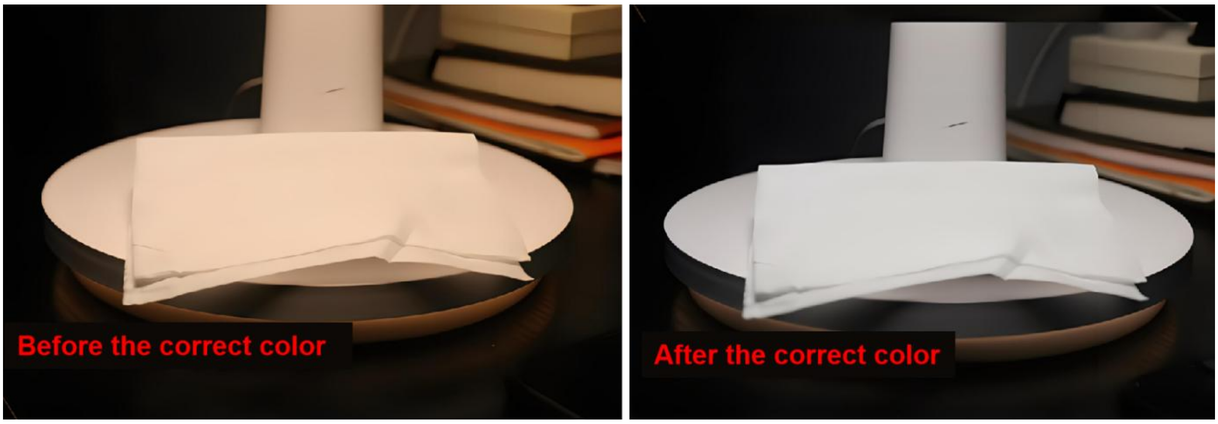
Ⅳ.Denoise Processing

The presence of ambient light and sensor issues contribute to the generation of a significant amount of noise in images. Additionally, other sources of noise arise during the image generation process, and ADC devices themselves introduce noise. When signals are amplified, noise is also amplified, resulting in colored speckles on the image, making it appear blurry. Methods for denoising images include mean filtering, Gaussian filtering, bilateral filtering, etc.



Ⅴ.Auto White Balance（AWB）

White balance is related to color temperature and is used to measure the color accuracy and fidelity of an image. The human visual system has the characteristic of color constancy, meaning that regardless of whether it's a sunny day, overcast, indoors with incandescent light, or under fluorescent light, white objects appear white to people. However, image sensors themselves do not have this characteristic. To eliminate the influence of the light source color on image sensor imaging, the automatic white balance function simulates the color constancy characteristic of the human visual system to eliminate the impact of the light source color on the image. This is achieved by adjusting the balance of the blue, green, and red channels' levels through internal circuitry in the camera, making white objects appear white under different color temperature lighting conditions. White balance algorithms include grayscale world, perfect reflection method, dynamic threshold method, etc.



Ⅵ.Color Correction Matrix(CCM）

Due to differences between the spectral response of the human eye and that of semiconductor sensors, as well as the influence of lenses, the RGB values obtained may have deviations from the true colors. Therefore, color correction is necessary. Typically, this is achieved through a 3x3 color transformation matrix. Color correction greatly enhances the aesthetics, realism, and consistency of images, resulting in high-fidelity color reproduction and visual appeal.

Ⅶ.Digitalization of Images

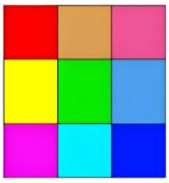
Digitalization of an image involves dividing the image into small individual units (pixels). These individual pixels can be represented individually using a quantized area. Grayscale is represented by integers, forming an array-style image where each pixel's attribute includes the position's grayscale or color. Digitalization comprises two processes: sampling and quantization.

For black and white images, each pixel can only be black or white, with no intermediate transition, hence they are also known as binary images. Binary images can only be represented using 0 and 1.

Grayscale images are represented by quantized grayscale values, without color information. They are typically represented using 8-bit bytes, allowing for the representation of 256 levels of grayscale ranging from 0 to 255.

$$I = \begin{bmatrix} 0 & 150 & 200 \\ 120 & 50 & 180 \\ 250 & 220 & 100 \end{bmatrix}$$

Color Image: A color image is described using three bytes (24 bits) per pixel, with R (Red), G (Green), and B (Blue) represented by different levels of gray.



$$R = \begin{bmatrix} 255 & 240 & 240 \\ 255 & 0 & 80 \\ 255 & 0 & 0 \end{bmatrix} \quad G = \begin{bmatrix} 0 & 160 & 80 \\ 255 & 255 & 160 \\ 0 & 255 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 80 & 160 \\ 0 & 0 & 240 \\ 255 & 255 & 255 \end{bmatrix}$$

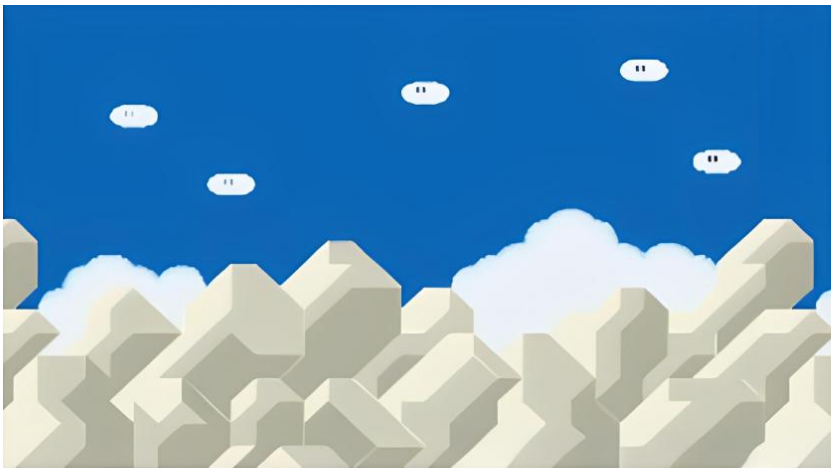
Lesson 4: Image Transmission and Imaging

Ⅰ.Image Compression

The basic principle of image compression is to remove data redundancy. Original image data contains a lot of redundant information, such as correlations between pixels, color distributions, etc. Compression techniques efficiently encode this redundant information through encoding methods, reducing file size and facilitating storage and transmission.

1.Spatial Redundancy

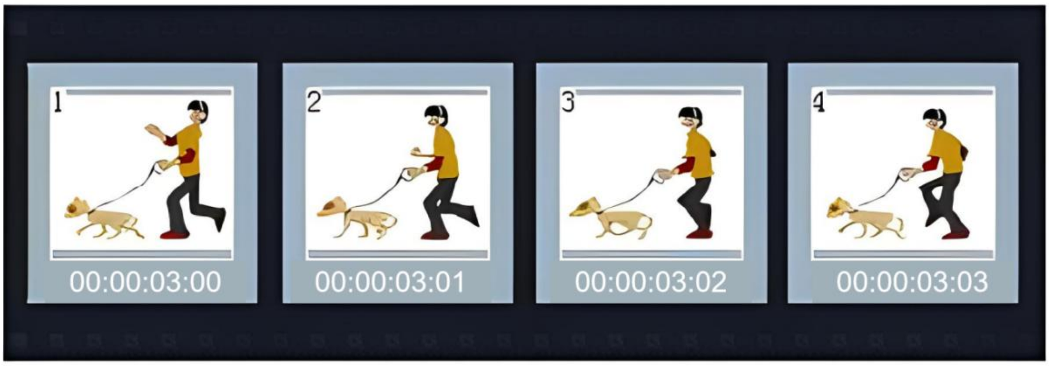
Spatial redundancy mainly exists in image data, where adjacent pixels in the image exhibit similarity or repetition in color or brightness values. For example, pixels within a contiguous region may have the same color, or the background of an image may consist of a single color, leading to the generation of spatial redundancy. Spatial redundancy can be compressed using various image processing techniques, among which Discrete Cosine Transform (DCT) encoding compression is a commonly used method. Its principle is to transform the image from the spatial domain to the frequency domain, effectively reducing data redundancy.



2.Time Redundancy

Time redundancy mainly appears in video, animation, and audio data, and it is closely related to time. For example, in a meeting room where people are having a meeting, the background remains unchanged, and only people's movements and positions change, resulting in time redundancy.

The processing techniques for time redundancy mainly include keyframe technology and inter-frame prediction. Keyframe technology involves extracting keyframes at regular intervals and transmitting only the complete information of these keyframes. The frames between two keyframes only need to transmit the parts that are different from the keyframes, effectively reducing the amount of data that needs to be transmitted. Inter-frame prediction predicts the difference between the current frame and the previous frame and encodes only this difference, further reducing data redundancy.



3.Visual Redundancy

The human visual system does not perceive every change in an image scene. The general resolution of the human visual system is approximately 26 levels of grayscale, while typical image quantization uses 28 levels of grayscale. This type of redundancy is called visual redundancy. Typically, the human visual system is sensitive to changes in brightness but relatively insensitive to changes in chrominance. In high-brightness areas, the sensitivity of the human eye to changes in brightness decreases. It is sensitive to object edges but relatively insensitive to internal regions. It is sensitive to overall structure but relatively insensitive to internal details. Therefore, information that does not cause visual sensitivity can be appropriately discarded.

Ⅱ.Wireless Image Transmission

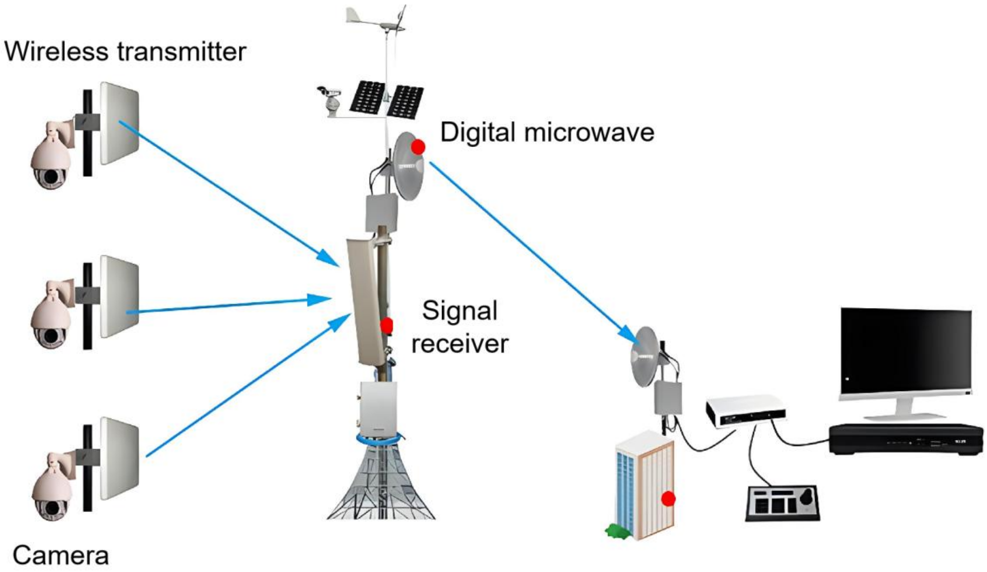
The principles of wireless image transmission mainly involve two technologies: analog microwave transmission and digital microwave transmission.

1.Analog Microwave Transmission

Analog microwave transmission involves directly modulating the video signal onto microwave frequencies, which are then transmitted via antennas. The monitoring center receives these microwave signals through antennas and uses microwave receivers to demodulate the original video signal. The advantages of this technology include clear images, no delay, no compression loss, low cost, and relatively simple installation and debugging. It is suitable for situations where there are few monitoring points and no need for relays.

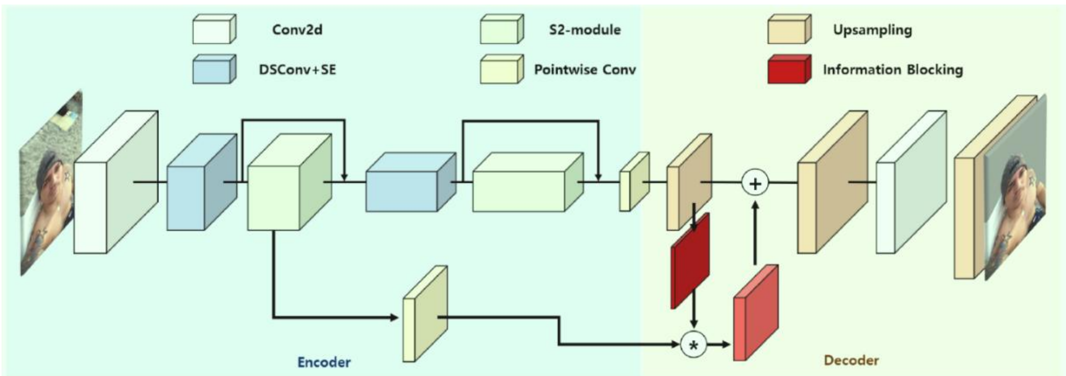
2.Digital Microwave Transmission

Digital microwave transmission involves encoding and compressing the video signal first, then modulating it through digital microwave channels before transmission via antennas. At the receiving end, the antenna receives the signal, which is then subjected to microwave demodulation and video decompression to ultimately restore it to an analog video signal. This method provides higher flexibility and control, such as video decompression, recording, playback, management, and PTZ control through computer software. It is suitable for situations where there are multiple monitoring points, complex environments, or the need for relays.



Ⅲ.Image Decoding

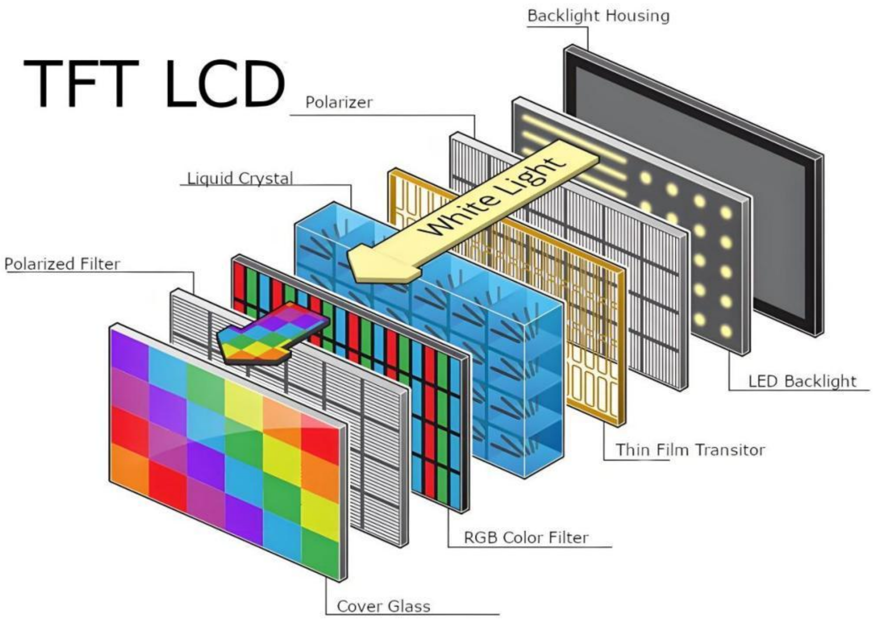
Image decoding refers to the process of restoring the original image from encoded and compressed image data through a decoding process. In lossless encoding, decoding involves reversing the encoding process to restore the image. In lossy encoding, the decoding process requires steps such as inverse quantization and inverse discrete cosine transform to recover the details of the original image.



Ⅳ.The Principle of Displaying Images on Screens

Display screens are mainly divided into two categories: CRT screens and LCD screens. Taking LCD screens as an example, LCD is a type of display that uses liquid crystal as a material. Liquid crystal is a class of organic compounds that exist between solid and liquid states. At room temperature, it exhibits both the fluidity of liquids and the optical anisotropy of crystals. When heated, it becomes a transparent liquid, and when cooled, it becomes a crystalline opaque solid.

Under the action of an electric field, the arrangement of liquid crystal molecules changes, thereby affecting the intensity of the incident light passing through the liquid crystal. This change in light intensity is further manifested as brightness and darkness through the action of polarizing films. Therefore, after the digital signal decoded from the image, it controls the brightness and darkness of the light through the control of the polarizing film in the circuit, thereby achieving the purpose of information display.



Lesson 5: Wireless Communication

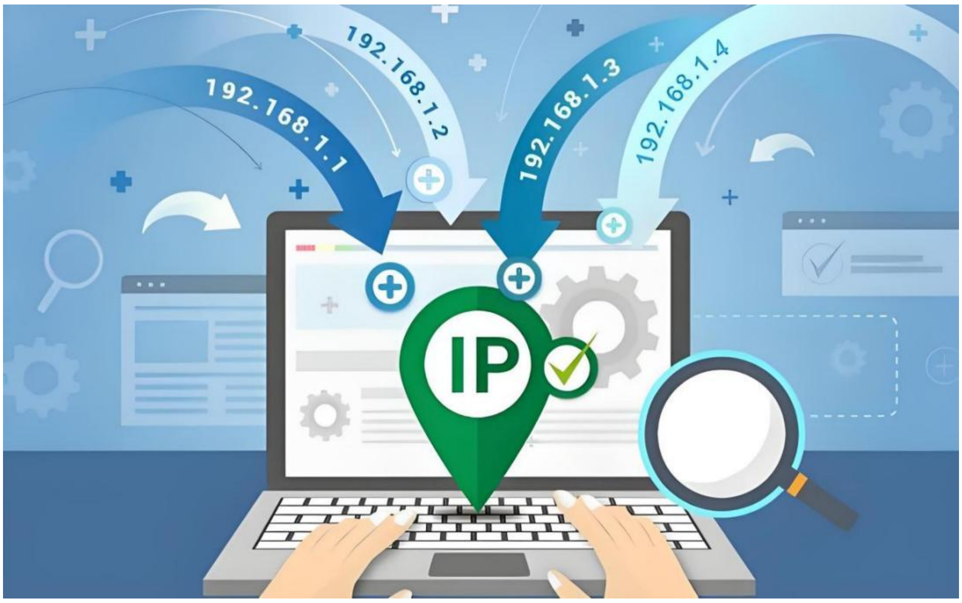
Ⅰ.WiFi Communication Technology

WiFi Communication Technology is a type of Wireless Local Area Network (WLAN) technology that allows electronic devices such as smartphones, tablets, laptops, etc., to connect to the Internet or a local area network wirelessly. WiFi stands for "Wireless Fidelity," and it is based on the IEEE 802.11 standard. It provides secure, reliable, and fast wireless connections.

Ⅱ.IP Address

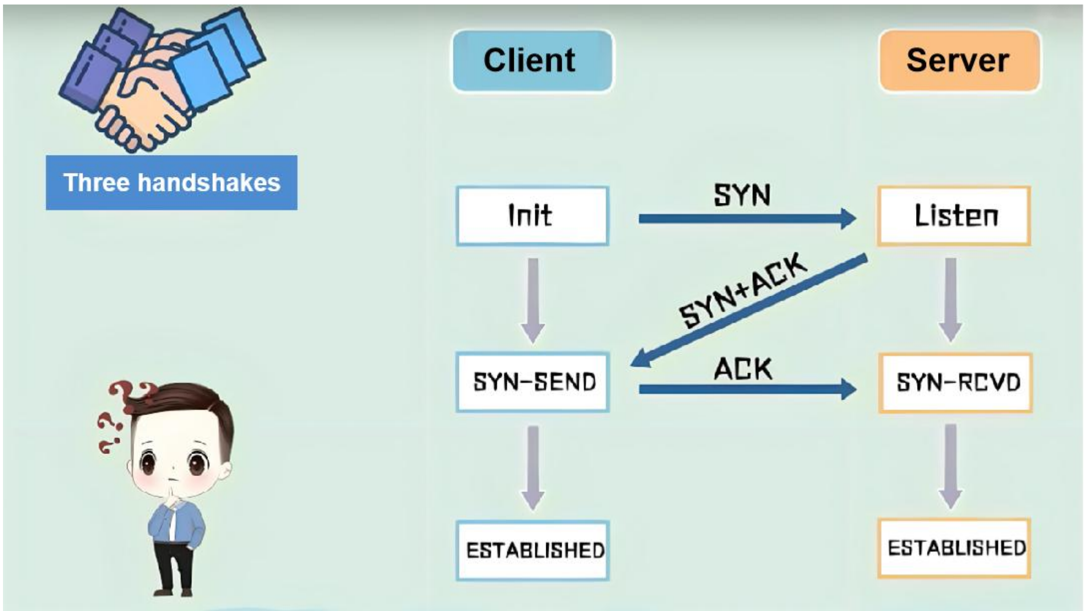
The IP address of each network device is provided by the Internet Protocol (IP), which serves as the foundation for internet communication. The IP protocol is divided into two versions: IPv4 and IPv6. IPv6 is an upgraded version of IPv4 and can provide more IP addresses, but this tutorial focuses on IPv4.

IPv4 addresses consist of four numbers separated by dots, for example, 192.168.10.136. This address is composed of two parts: the first three numbers (192.168.10) represent the address of the WiFi network to which the device is connected, while the last number (136) represents the device's address within that WiFi network.



Ⅲ.TCP Protocol

TCP (Transmission Control Protocol) is a connection-oriented and reliable protocol that ensures data packets arrive at the receiver in the correct order. If packets are lost or damaged, TCP will retransmit them. TCP is widely used in applications requiring high reliability, such as web browsing, email, and file transfer. TCP communication involves establishing a server and client, enabling data exchange between them.

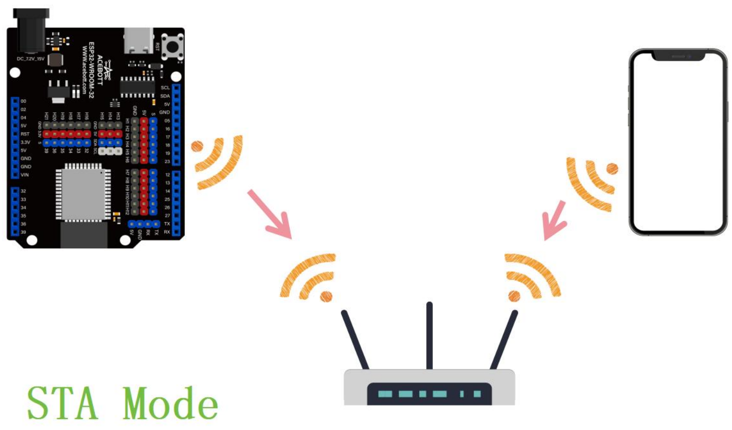


Ⅳ.STA Mode and AP Mode

WiFi communication technology connects devices to the same network through wireless routers or access points (AP), much like constructing a road between each device to provide a wireless communication channel. In a WiFi network, devices can transmit and receive data to and from each other.

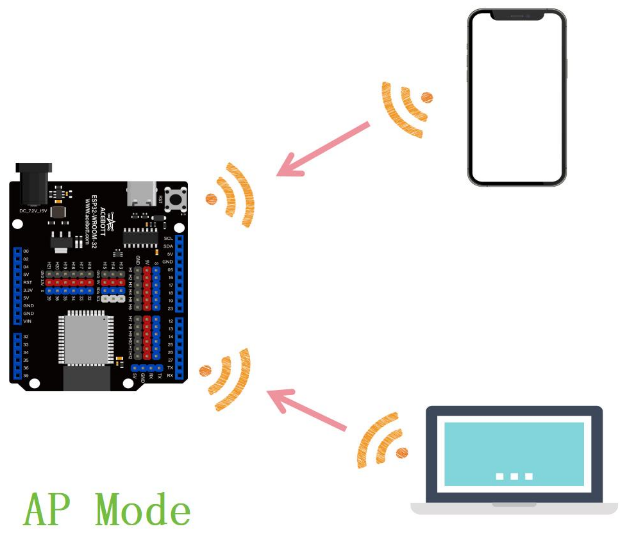
1.Wireless Terminal Mode（STA Mode）

In this mode, the ESP32 acts as a terminal device connected to a WiFi network generated by other devices.



2.Access Point mode（AP Mode）

In this mode, the ESP32 acts as a WiFi generator, creating its own WiFi network signal for other devices to connect to.



Lesson 6: Web Control

Ⅰ.To invoke the camera program

After uploading the camera program, the imaging results will be displayed on the web page. At this point, the terminal device needs to be connected to the same WiFi network as the camera module. Press the reset button on the camera, and then obtain the IP address of the camera module from the Arduino IDE. Next, enter this IP address in the terminal to access the web page displaying the camera imaging. By moving the camera direction, you can see the real-time imaging effect of the camera module on the web page.

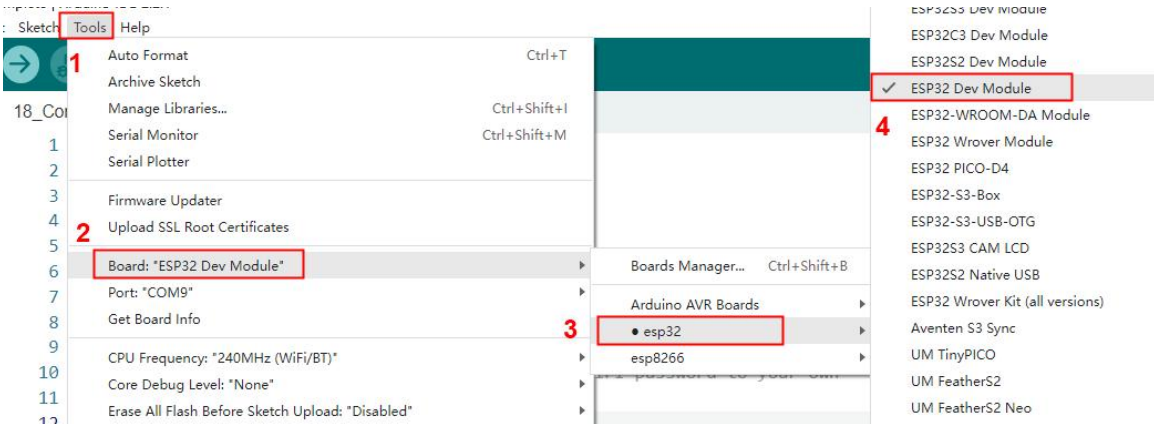
The imaging program of the camera module has two modes: STA mode and AP mode. In the basic version of the car, we have already used operations related to AP mode. This time, we will try to use STA mode to control the camera to display images on the web.

【Click to get the CAM-STA-MODLE program】

```
const char* ssid = "ACEBOTT";//your wifi name
const char* password = "12345678";//your wifi password
```

Note: You need to fill in your own WiFi account and password in the program.

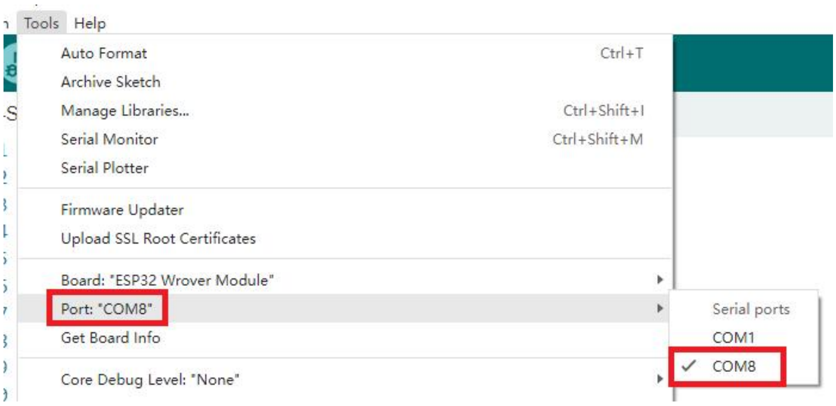
1. Select the controller board type of the camera module. Click Tools>Board>esp32>ESP32 Dev Module.



2.Connect the serial port.

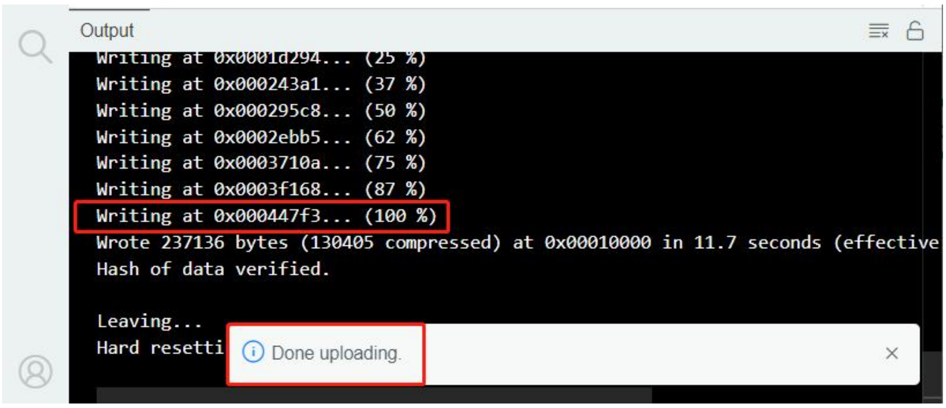
Click Tools>Port>COM8(Select the COM port corresponding to your computer)

Note: The COM port to connect is usually not COM1. Click on another COM port.



3.Click on "Upload" to burn the program onto the board.

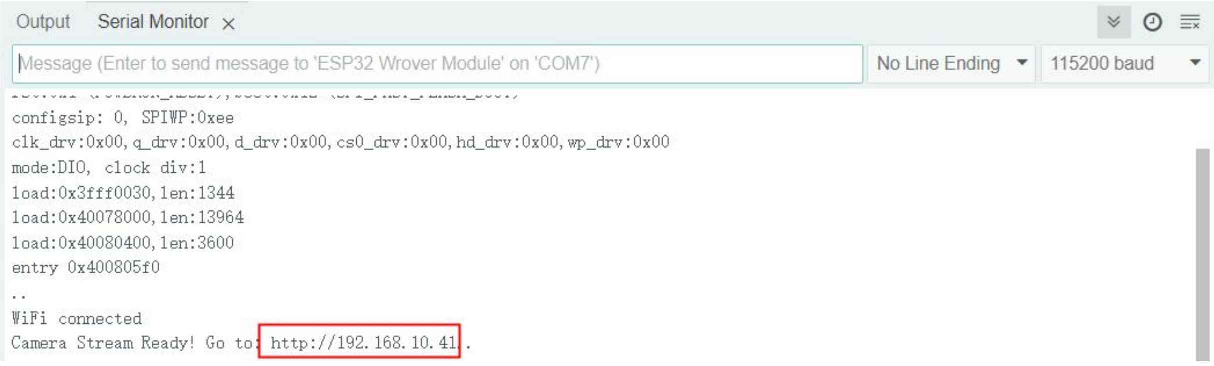
4.When the progress shows 100%, the program upload is complete.



Ⅱ.Displaying images on the web

1.Pressing the reset button on the camera module will display the connected IP address on the Arduino IDE serial port.

Note: The IP address will be different under different Wi-Fi networks.



2.The terminal device needs to be on the same Wi-Fi network as the camera module, and then enter the IP address displayed on the serial port in the browser.

Ⅲ.Modify image sharpness parameters

By selecting the byte length sample value, the image clarity can be adjusted. There are a total of three clarity levels to choose from, but we can only select one. You need to comment out the other two lines. Generally, the higher the image quality, the more likely it is that image delay will occur.

```
//s->set_framesize(s, FRAMESIZE_SXGA); //Byte length Sample value :60000
s->set_framesize(s, FRAMESIZE_SVGA); //Byte length Sample value :40000
//s->set_framesize(s, FRAMESIZE_QVGA); //Byte length Sample value :10000
```

Ⅳ.Modify the image display direction

If your camera module is "ESP32-CAM-V2.0", then in order to display the picture normally, you need to modify the instructions in the CameraWebServer.cpp program file, the instructions for the left and right direction of the picture are set by the original s->set_hmirror(s, 1); Change to: s->set_hmirror(s, 0);

```
s->set_vflip(s, 1);   //Picture orientation Settings (up and down)
s->set_hmirror(s, 0); //Picture orientation Settings (left and right)
```

Note:In the APP control program is also the same.

Ⅴ.Web Controlling the Camera Car

After completing the web display for the camera, we can further integrate it with the smart car to achieve the method of controlling the camera car's movement via a webpage. For this functionality, different programs need to be uploaded to the CAM module and the smart car, respectively.

1.Web Control Program for the CAM Module:

Connect the CAM module to the computer using a USB cable. Then, click the link below to get the program for the CAM module. In the Arduino IDE, go to the toolbar, click on Tools, and select the board type as "ESP32 Dev Module". Finally, connect to the corresponding serial port and upload the program.

【Click to get the acebott-esp32-car-camera program】

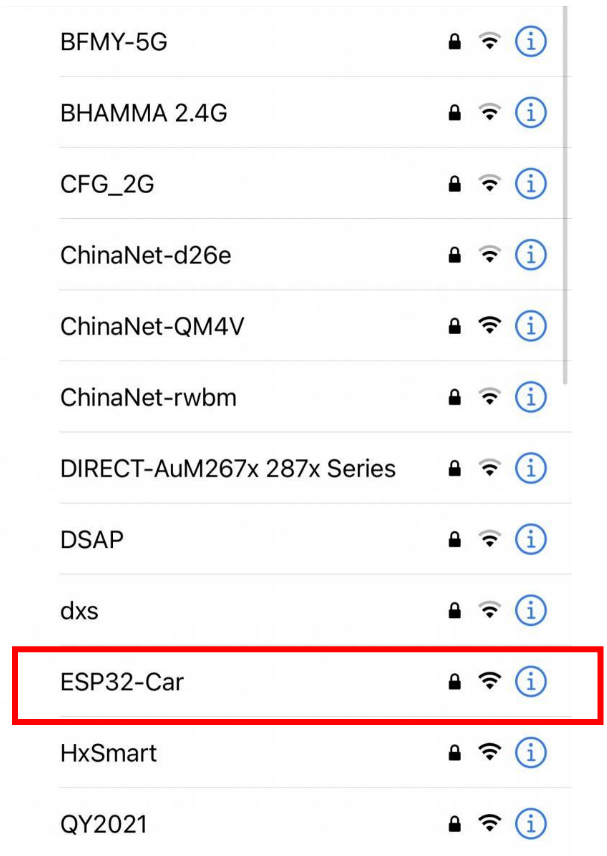
2.Program for the Smart Car

Turn on the battery box and connect the ESP32 controller board of the car to the computer using a USB cable. Then, click the link below to get the program for the smart car. In the Arduino IDE, go to the toolbar, click on Tools, and select the board type as "ESP32 Dev Module". Finally, connect to the corresponding serial port and upload the program.

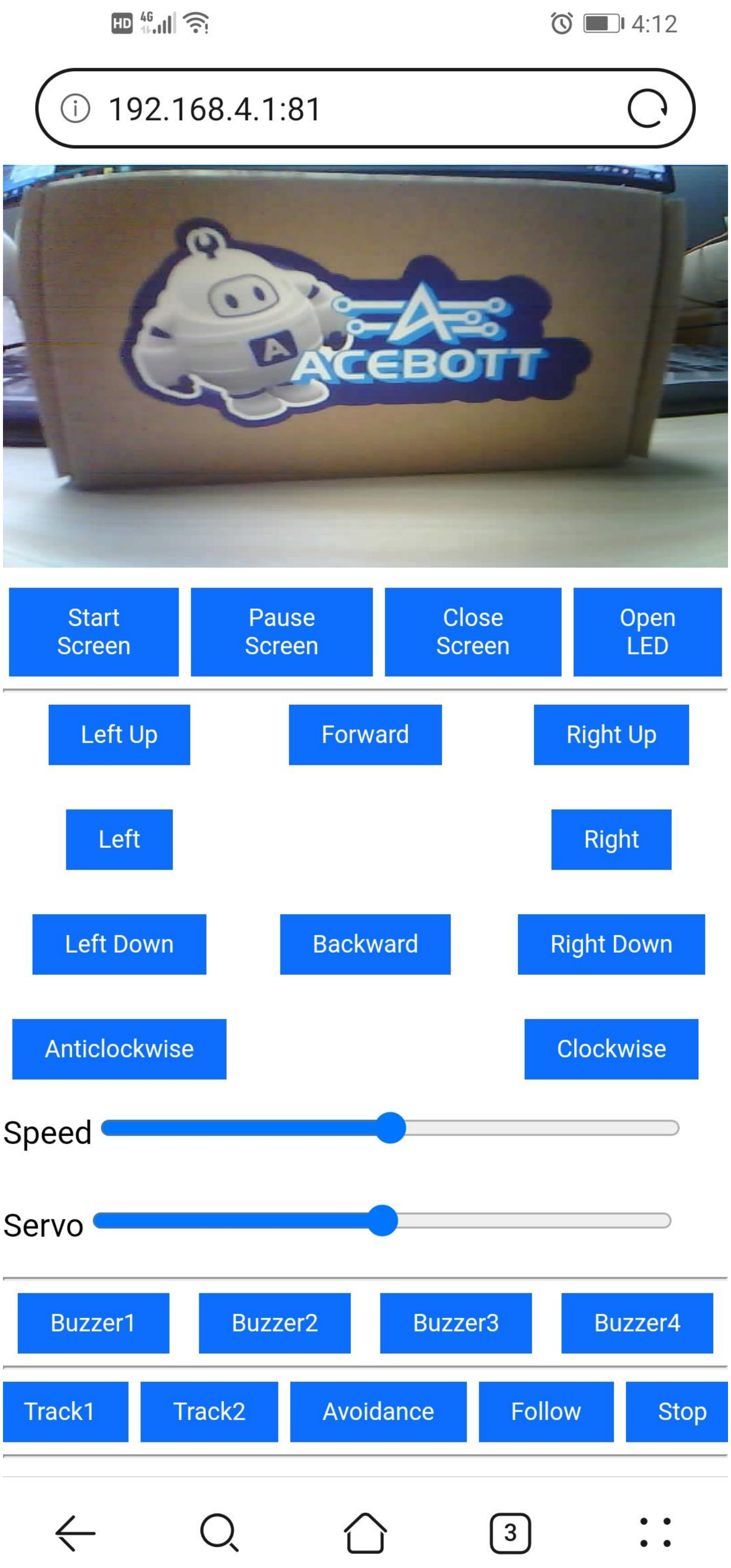
【Click to get the acebott-esp32-car-body program】

Note: When programming the smart car's controller board, you need to disconnect the TX and RX wires of the camera. Once the program upload is complete, reconnect the TX and RX wires.

After uploading the program, open your phone's WiFi settings, and connect to the WiFi hotspot named "ESP32-Car" with the password "12345678" as shown below.



We need to open the browser on the phone and enter the IP address: 192.168.4.1 in the address bar. This will take you to the Web control page, where you can proceed with the control.



Lesson 7: APP Design and Debugging

Ⅰ.App Development Process

To facilitate convenient viewing of the camera's imaging effect, we can choose to use a mobile app as the client to achieve this effect. The process of app development involves the following steps:

1. Determine the development platform for the app.
2. Design the interface of the app.
3. Integrate the WiFi module.
4. Write the control logic.
5. Perform testing and optimization.
6. Release and promote the app.

Ⅱ.APP Inventor

APP Inventor is an online, visual programming platform for developing Android mobile apps, originally developed by Google Labs. In 2012, Google transferred the project to MIT (Massachusetts Institute of Technology) for management and development, renaming it MIT APP INVENTOR. MIT APP INVENTOR has four main features: online programming, rich components, visual interface, and real-time testing.

MIT
APP INVENTOR

Note: APP INVENTOR currently only supports Android systems.

Ⅲ.App Development

If you'd like to experience it directly, you can bypass the process of making the app and simply click the link below to download the installation package for an app named "CAM_1".

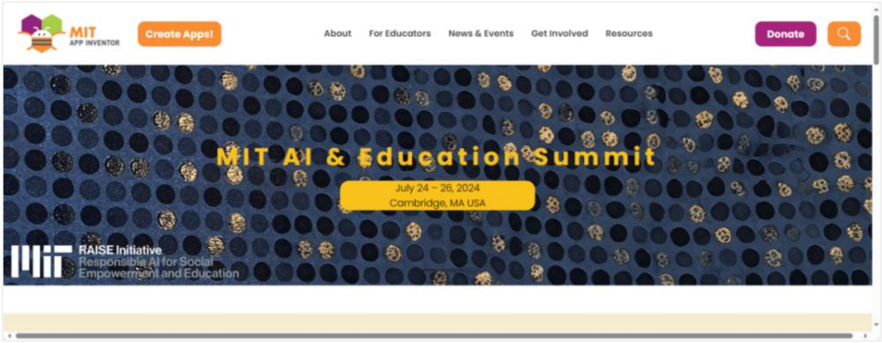
Note: The app installation package is only compatible with Android systems.

【Please click here to obtain the app installation package】

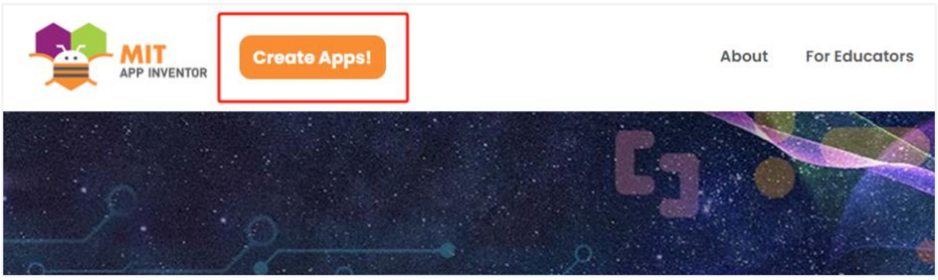
1.Log in to APP INVENTOR

(1)Log in to the official MIT APP INVENTOR website in your browser：

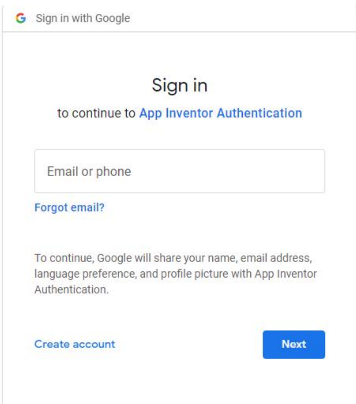
https://appinventor.mit.edu/



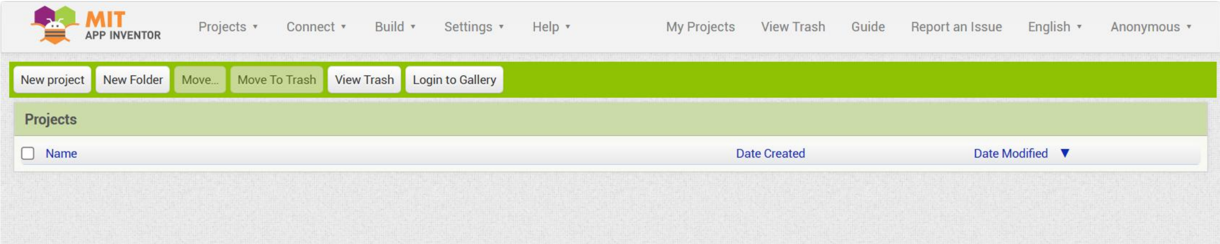
(2)Click the "Create AppS" button



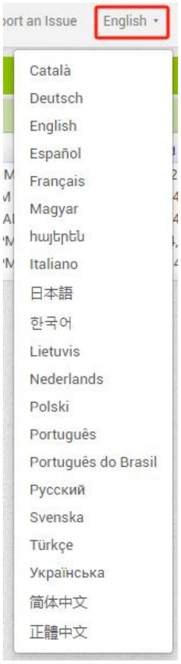
(3)Log in to APP INVENTOR with your Google account



(4)After login, the APP project management page will appear

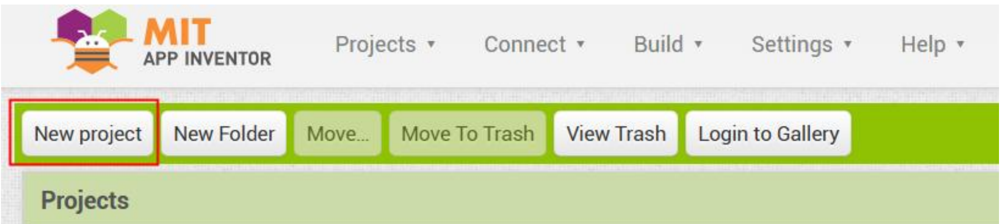


(5)Set Language

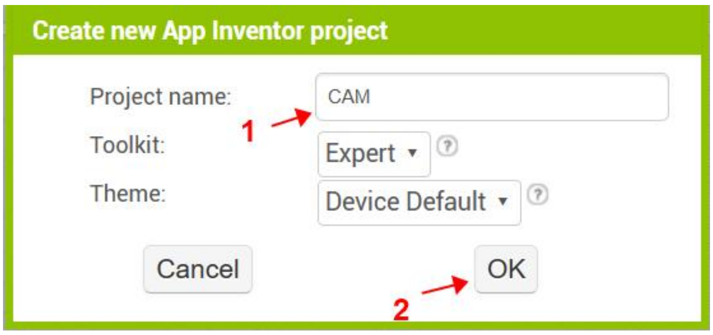


2.Creating a new APP project

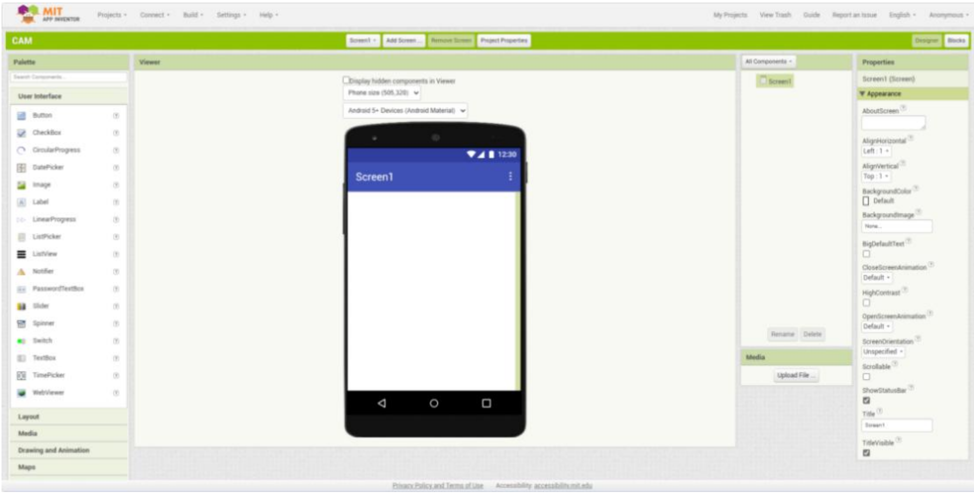
(1)Click “New project”



(2)Set the project name



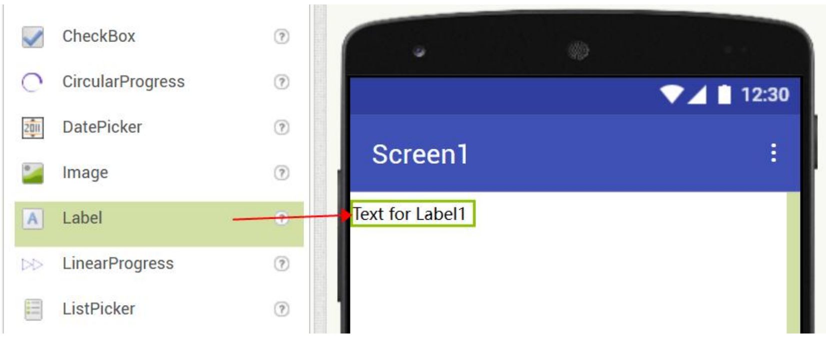
(3)Pop up the interface made by the APP



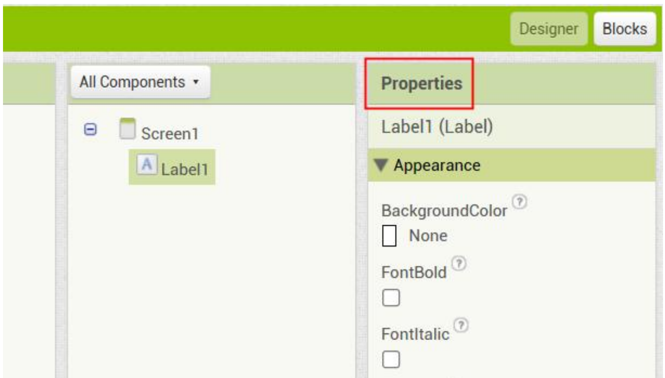
3.APP interface production

(1)Make a title

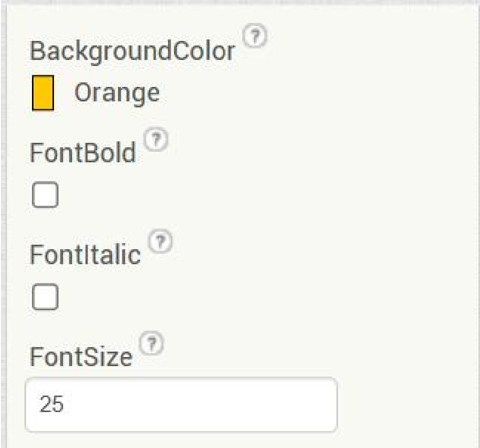
①Find the Label component from the user interface and drag it into the phone screen.



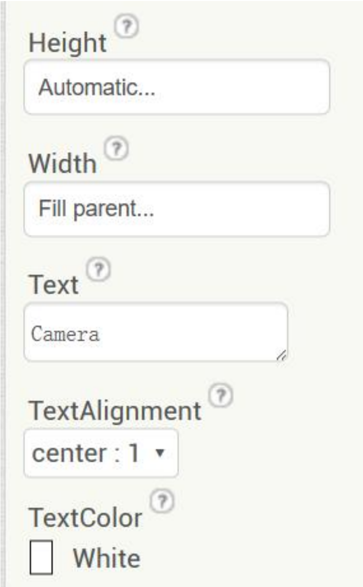
②Find the component properties on the right side of the page, and modify the properties of the corresponding component as needed.



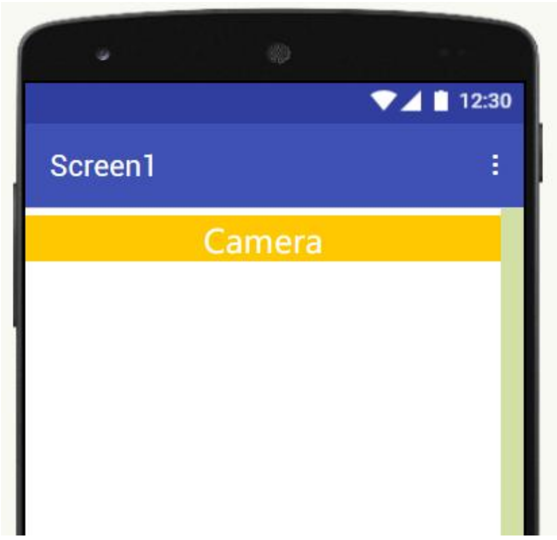
③Change the background color of label 1 to orange and the font size to 25.



④Change the width of label 1 to fill the screen, change the text to "Camera", change the text alignment to center, and change the text color to white.

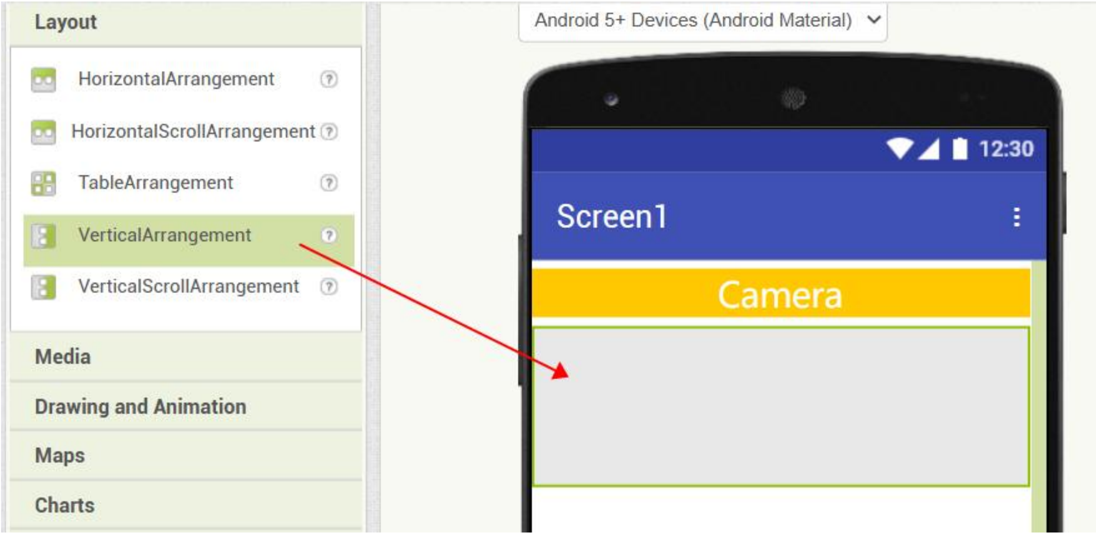


⑤After the modification, the page title is finished.

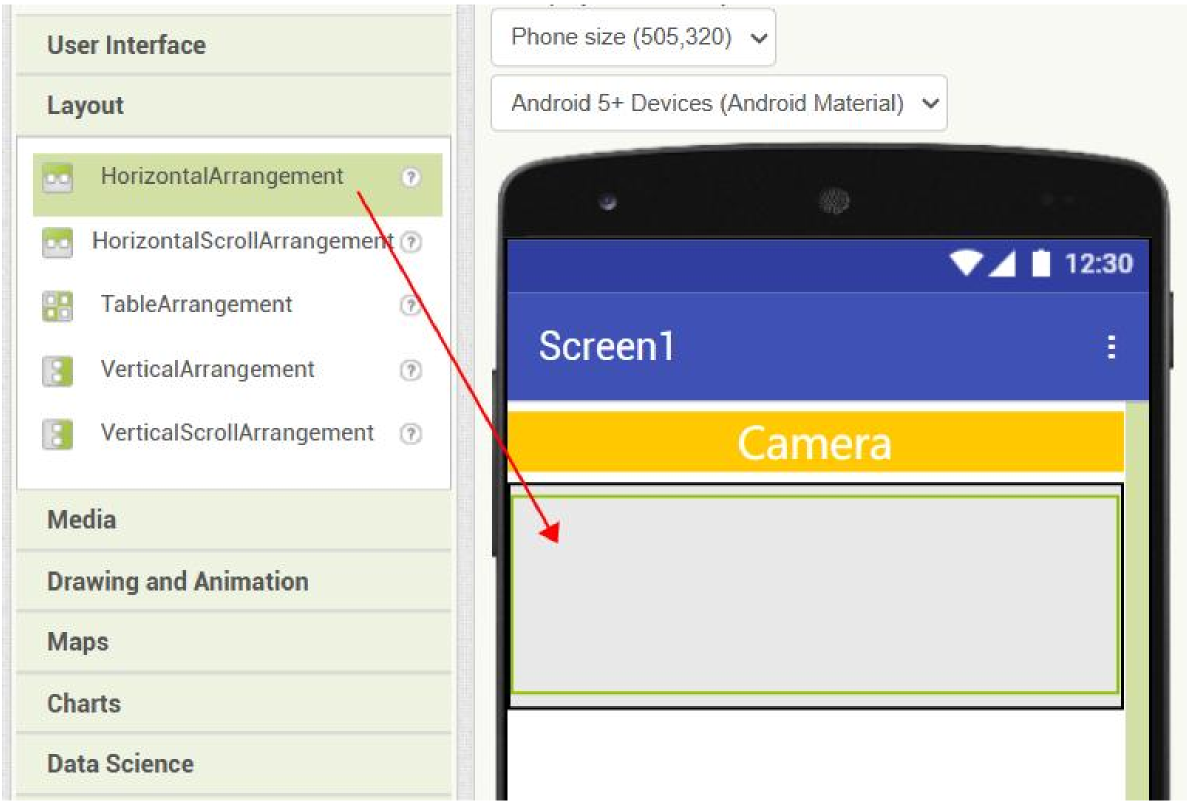


(2)Make IP input and port input

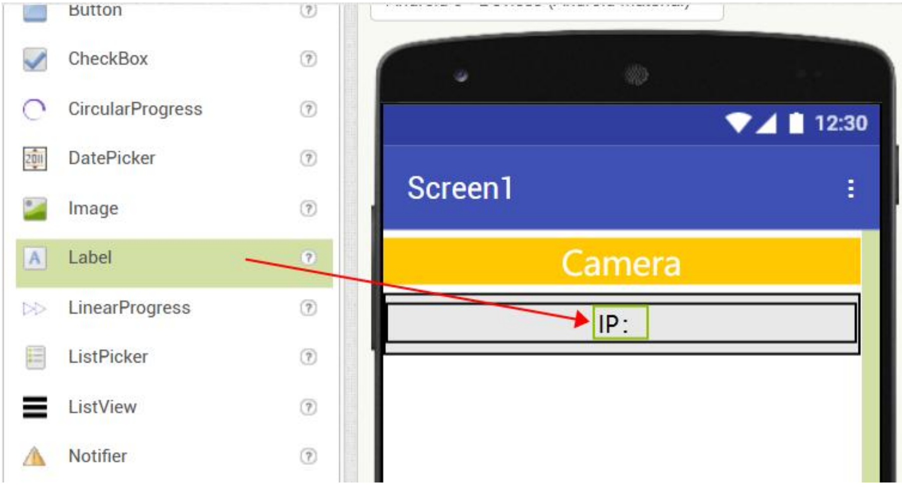
①Drag a Vertical Layout 1 component into the mobile interface, and change the background color to light gray. Set both horizontal and vertical alignment properties to center, and set the width to fill the screen.



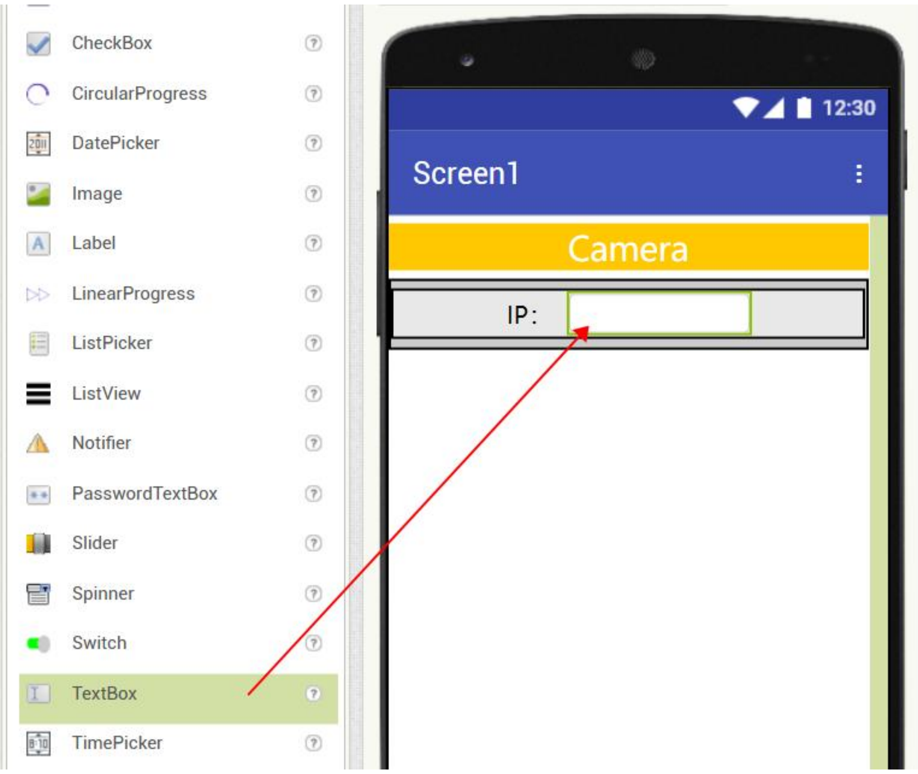
②Drag a Horizontal Layout 1 component into the Vertical Layout 1 component. Set both horizontal and vertical alignment properties to center, and adjust the width to fill the screen.



③Drag a Label 2 component into the Horizontal Layout 1 component. Change the font size of Label 2 to 18 and modify the text to IP:

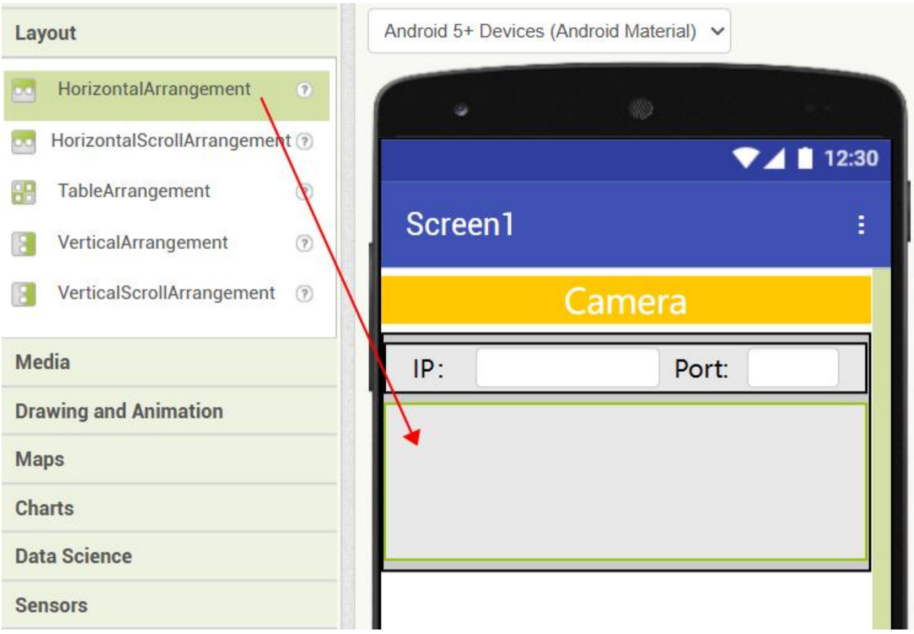


④Within the Horizontal Layout 1 component, drag a Text Input 1 component after Label 2 and set its width to 120 pixels.

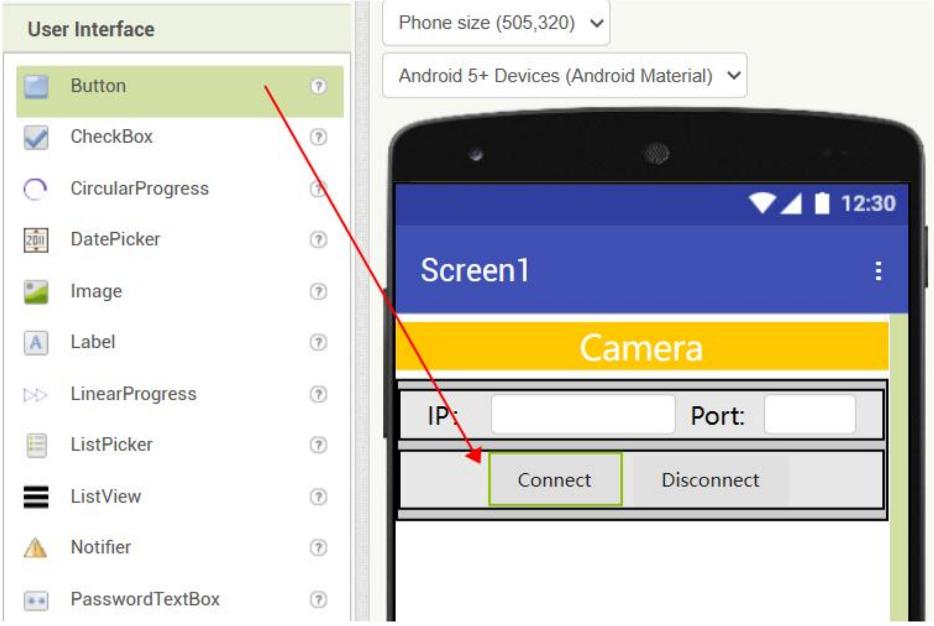


⑤Drag a Label 3 component and a Text Input 2 component after the Text Input 1 component. Change the font size of Label 3 to 18 and modify the text to 'Port:'. Adjust the width of Text Input 2 component to 60 pixels.

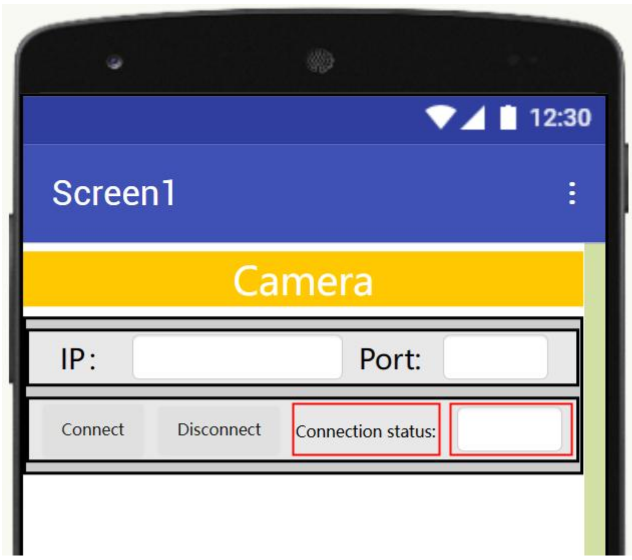
⑥Within the Vertical Layout 1 component, drag a Horizontal Layout 2 component below Horizontal Layout 1. Set both horizontal and vertical alignment properties to center, and adjust the width to fill the screen.



⑦Drag the Button 1 component and Button 2 component inside the Horizontal Layout 2 component. Change the text of the button 1 component to "connect" and the text of the button 2 component to "disconnect",Change the font size to 11.

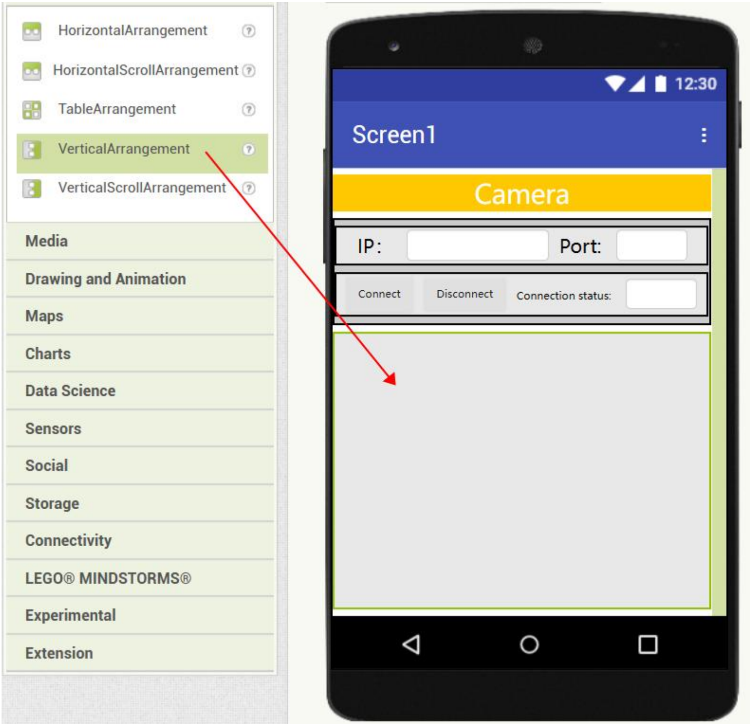


⑧Inside the Horizontal Layout 2 component, after the button 2 component, drag the label 4 component and the text input 3 component. Change the text of the label 4 component to "Connection status:" and Change the font size to 11. Change the width of the text input 3 component to 60 pixels.

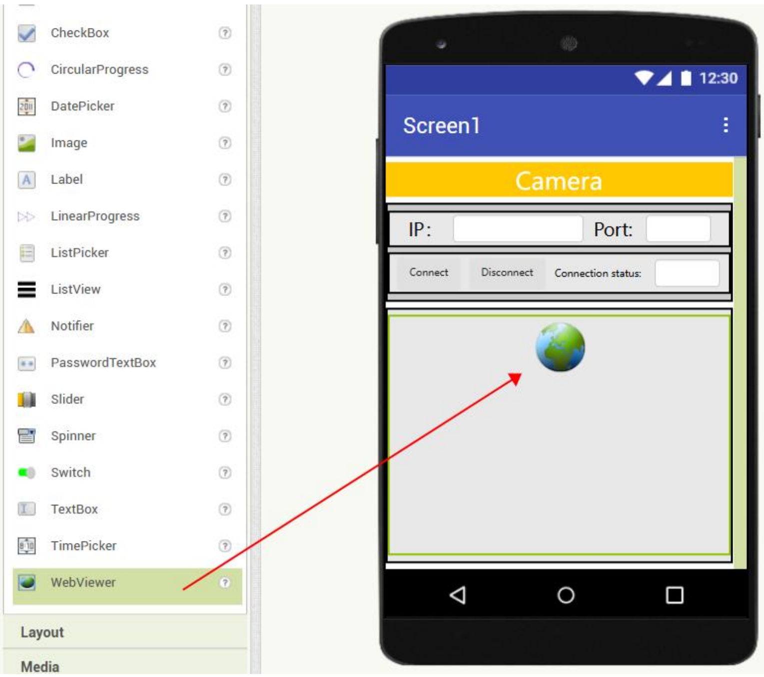


(3)Make the camera display window

①Drag a vertical Layout 2 component below the vertical Layout 1 component and change both the horizontal and vertical alignment properties to center, the height to fill screen, and the width to fill screen.



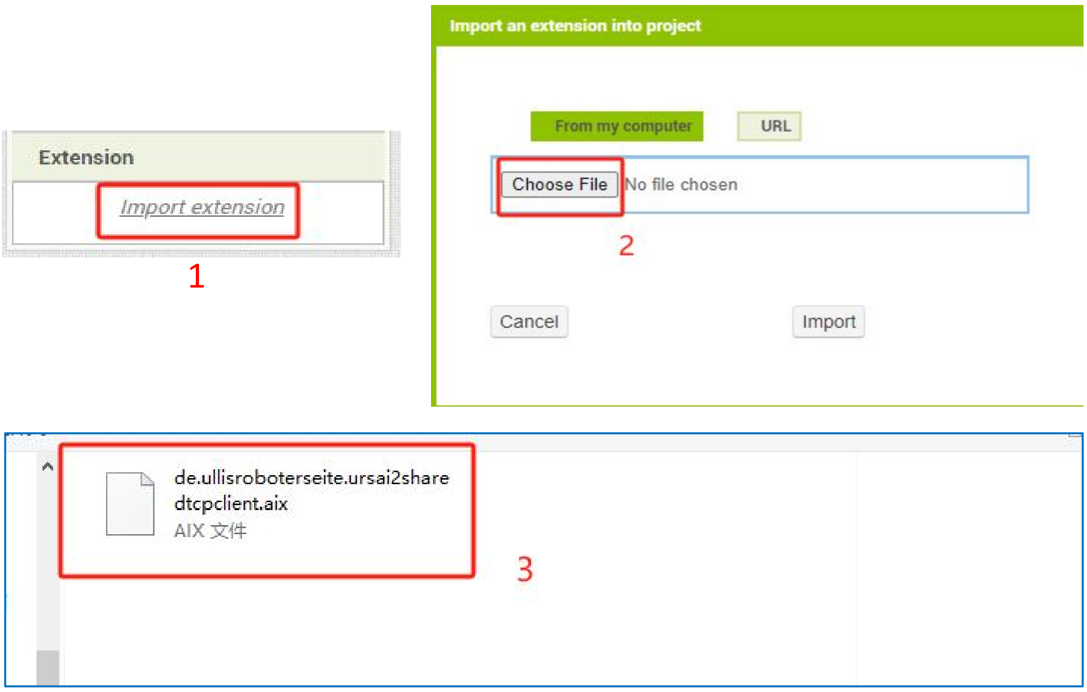
②In the Vertical Layout 2 component drag a web browser box component, change the height to fill screen, and change the width to fill screen.



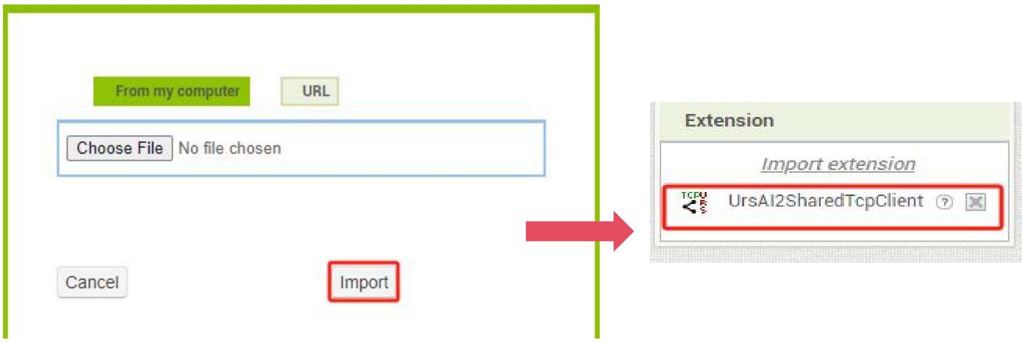
(4)Add extension component

①Go to the APP INVENTOR Component Design page and find the Extension option in the components panel.

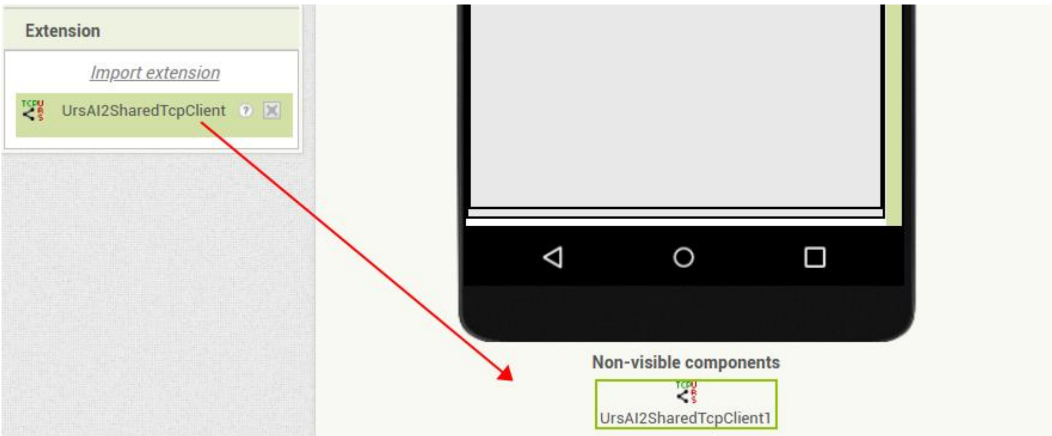
②Click "Import extension", then a window for importing projects will pop up. Click "Choose File" and browse the file path in "ACEBOTT QD002 ESP32-Camera Car\English\4.APP INVENTOR Extension Components", find the file named "de.ullisrobotseite.ursai2sharedtcpclient.aix", and then click on the component you want to add.



③After selecting the Extension, click "Import" to import the component. The Extension panel will show you the newly imported extension.

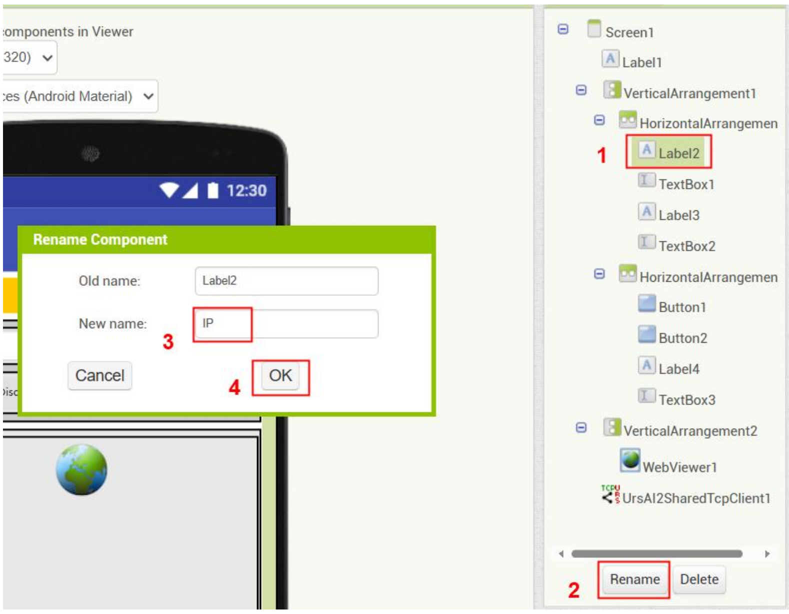


④Drag the TCP extension under the phone screen.

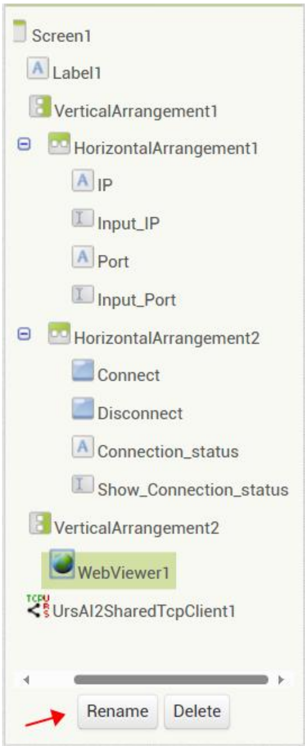


(5)Define the name of the component

①In order to distinguish each component better, rename each component by selecting the component you want to rename, selecting the "Rename" button, entering a new name in the pop-up window, and clicking "OK".

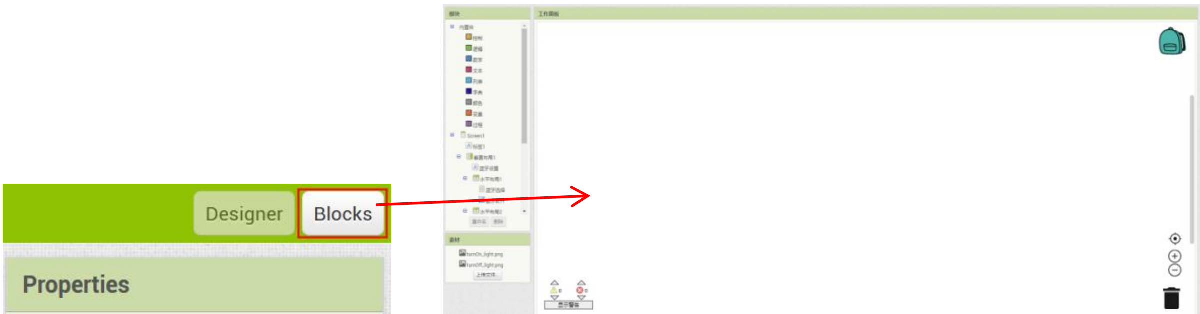


②From the previous step, change the name of the selected component as shown in the following screenshot:

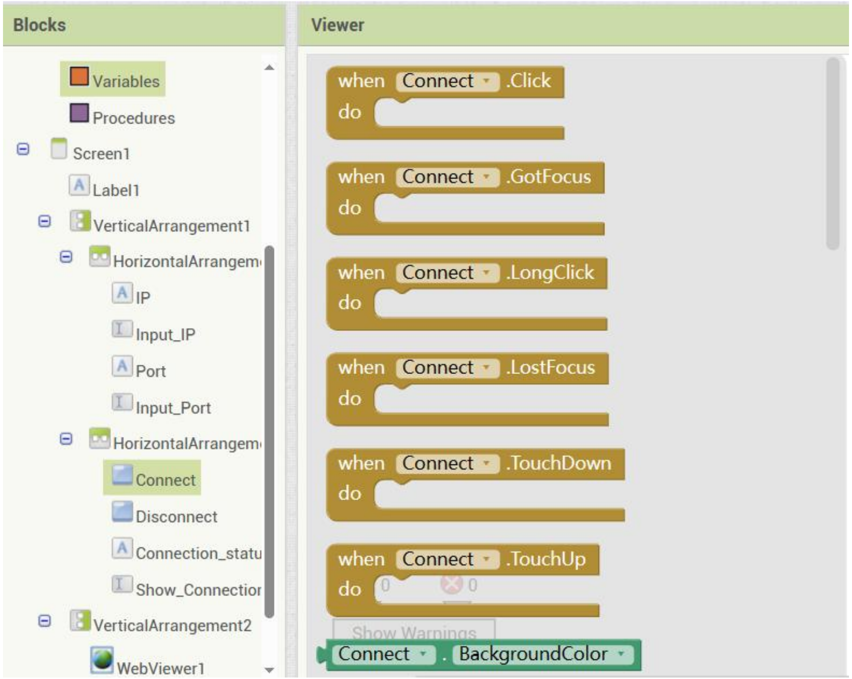


4.App Logic Design

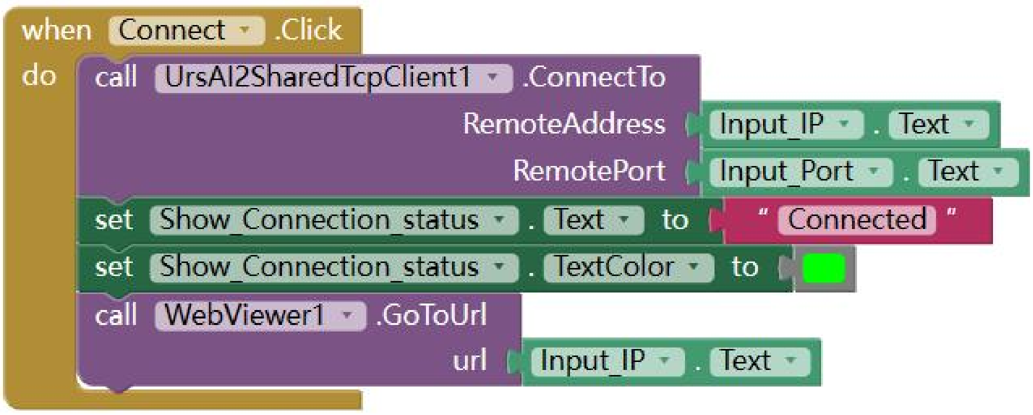
(1)Click the 'Blocks' button in the top right corner of the page to enter the logic design page.



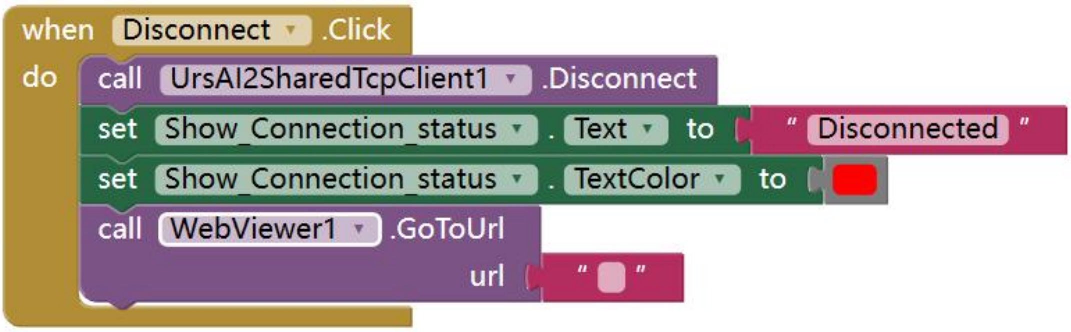
(2)Clicking on the module on the left will bring up the corresponding built-in programming instructions and component-related control instructions.



(3)Implement communication connections. Click the "Connect" button, the mobile phone sends a connection request to the TCP server, and sets the text of the "connection status display" text input box to "connected", and the text color is set to green.

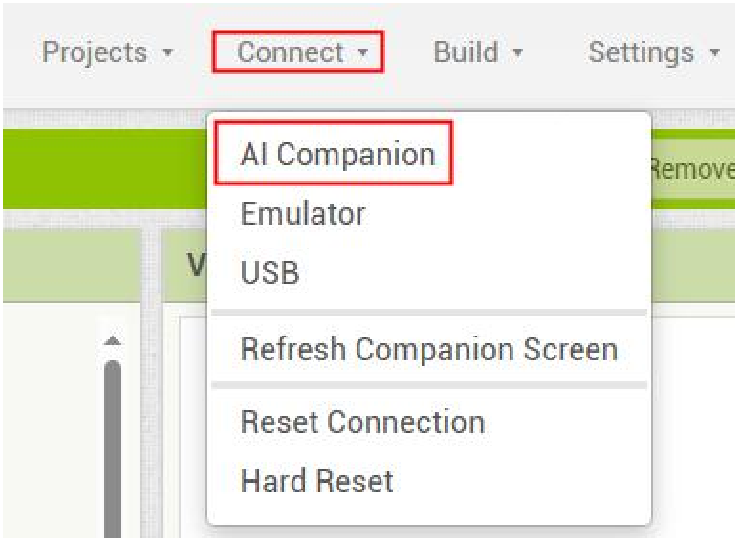


(4)Disconnect the communication connection. Click the "Disconnect" button, the phone sends a connection request to the TCP server, and sets the text of the "connection status display" text input box to "Disconnected", and the text color is set to red.

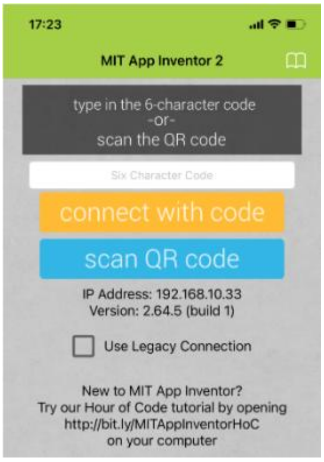


5.APP Testing and Debugging

APP INVENTOR programs can be debugged online in real time with an AI companion.

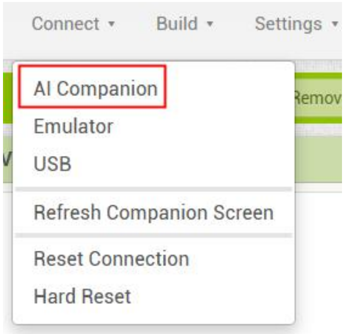


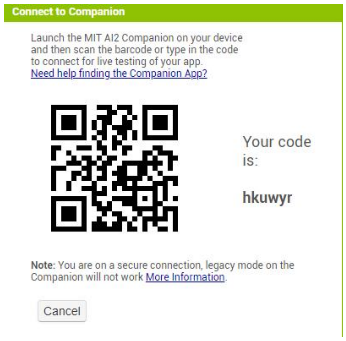
To debug the AI companion, you'll need to download an AI companion app on your mobile device. Navigate to "ACEBOTT QD002 ESP32-Camera Car\English\5.Installation package" and locate the file named "MITAI2Companion.apk". Then, install the AI companion app on your mobile device.



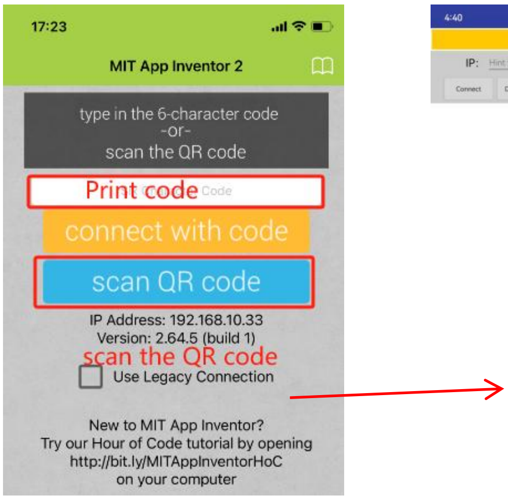
Note: AI companion debugging requires your phone and computer to be connected to the same WiFi.

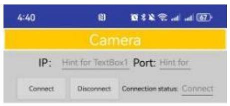
On the APP INVENTOR page, select the connection method as AI partner, and a QR code and code will pop up.





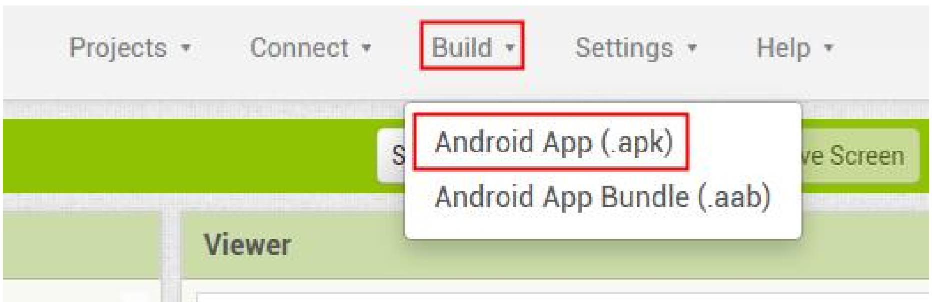
Then open the AI companion app on your mobile device and scan the QR code on the computer for connection, or manually enter the code for connection. After successful connection, the AI companion interface will switch to the program debugging interface.



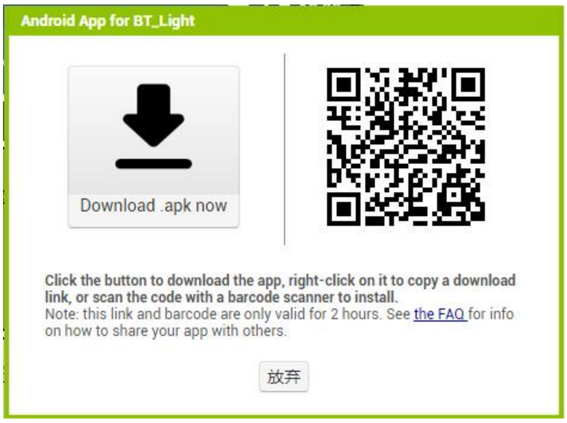


6. APP package installation

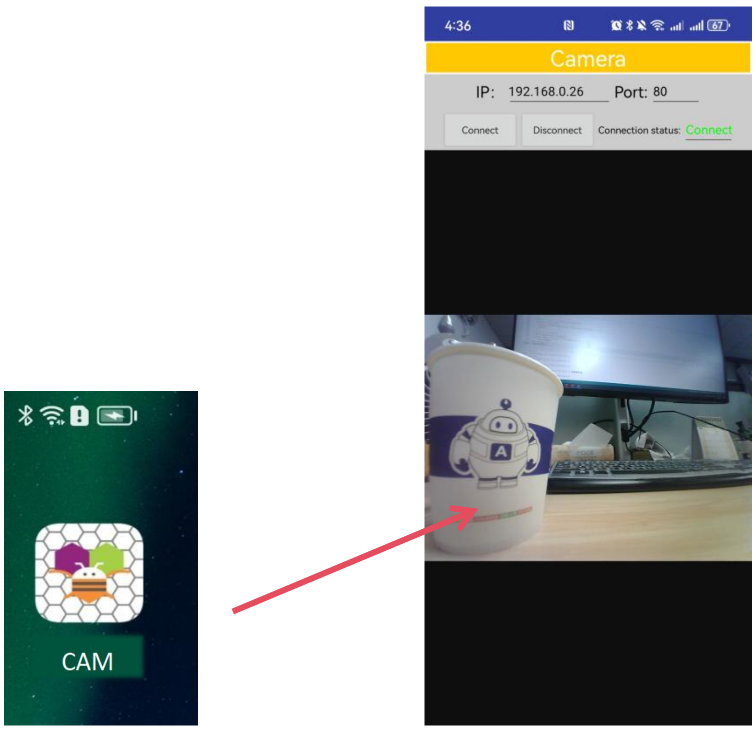
(1)After debugging, the APP will be packaged in.apk format.



(2)After the package is completed, a download window will pop up, and you can download the.apk file to your computer through the download button on the left, or you can download it by scanning the QR code on the right.



(3)Upload the downloaded .apk file to your mobile device. Currently, APP INVENTOR only supports the Android system. Tap the .apk file on your device to install it. Once the installation is successful, an app icon will appear on your home screen. Tap this icon to open the app. Then, ensure your phone is connected to the same WiFi as the camera module. Enter the camera's IP address in the IP input field and enter 80 in the port input field. Finally, click the "Connect" button. The live feed from the camera will appear below the button.



Lesson 8: Control with ACEBOTT APP

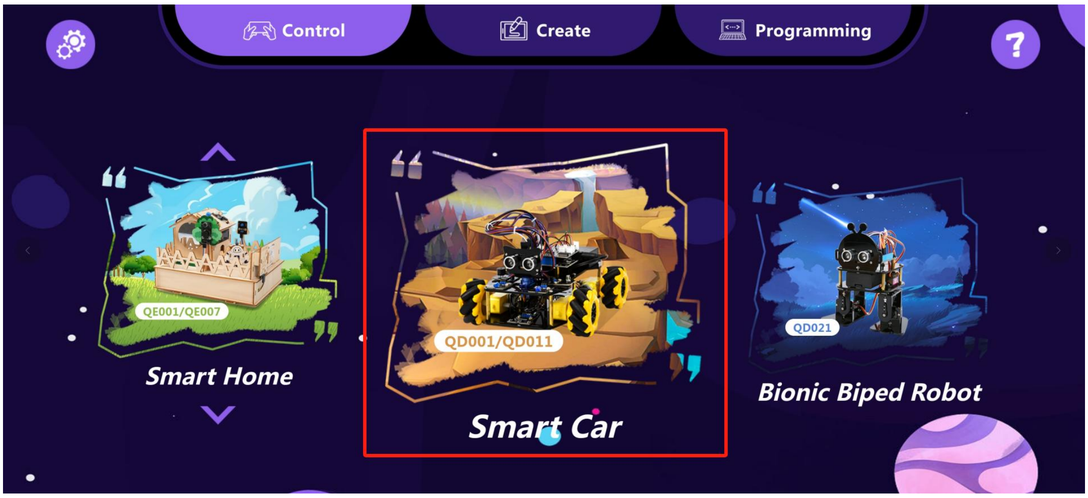
Ⅰ.APP download

(1)If you are using an iOS device, search for the keyword "ACEBOTT" in the App Store and download the app. If you are using an Android device, search for the keyword "ACEBOTT" in the Google Play Store and download the app. The icon looks as shown below.

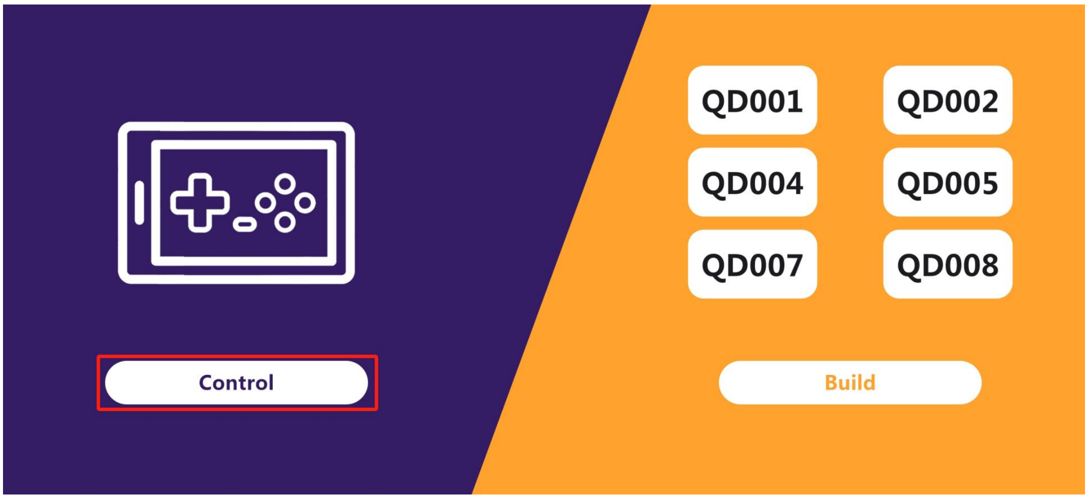
Note: 1. This tutorial is applicable to ACEBOTT APP version 2.0 and above. You can click the settings button in the upper left corner of the APP to view the software version number. Please make sure that the software version you are using meets the requirements; 2. If you need to update the ACEBOTT software version, you can refer to the method prompted in this tutorial to download the latest APP version.

(2)After opening the app, you will enter the splash screen.

(3)Enter the selection screen and choose the Smart Car as shown below.

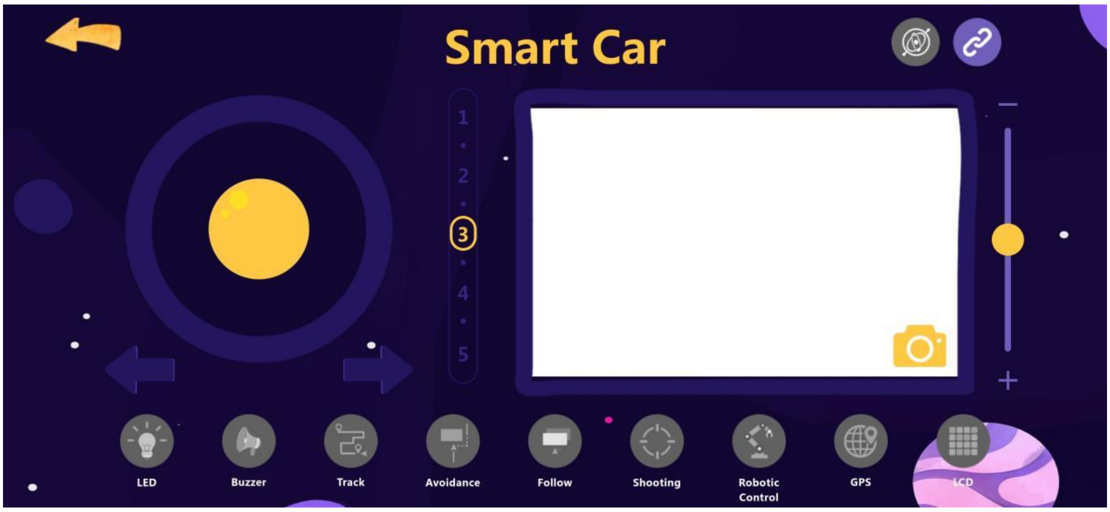


After clicking to enter, select "Control" to access the car control page.



Note: You can click the build button on the right to view the assembly video of this project.

(4) Enter the car control interface (you cannot control it yet; the program needs to be flashed first).



Ⅱ.Program download

1.Robot APP control program

Currently, you cannot directly control the robot with the app because you need to flash the app's control program onto the ESP32-Camera car first. (The app program is shared with the web-controlled car program. If you have already flashed it, you can skip this step.)

The ESP32-Camera car's program includes both the smart car's program and the Cam module's program, so you need to upload different programs to the respective boards. Please turn on the car's power before uploading.

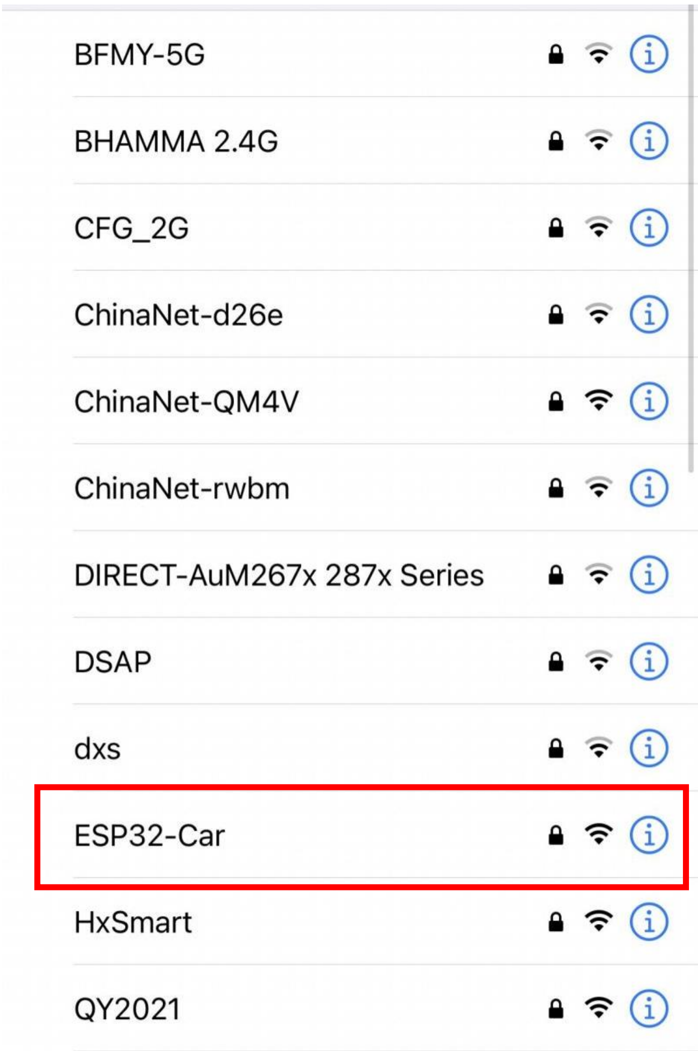
(1)Smart Car Program：【Click to get the acebott-esp32-car-body program】

Note: When flashing the program to the smart car's controller board, you need to disconnect the TX and RX lines of the camera first. Reconnect the TX and RX lines after the program upload is complete.

(2)Cam Module Program：【Click to get the acebott-esp32-car-camera program】

2.Connect to WiFi

Scan for WiFi networks on your phone (disable GPRS and other shared networks to ensure WiFi is the only network being used). This can be done in the phone's "Settings" → "WLAN". Connect to the WiFi hotspot named "ESP32-Car" with the password 12345678, as shown in the image below.

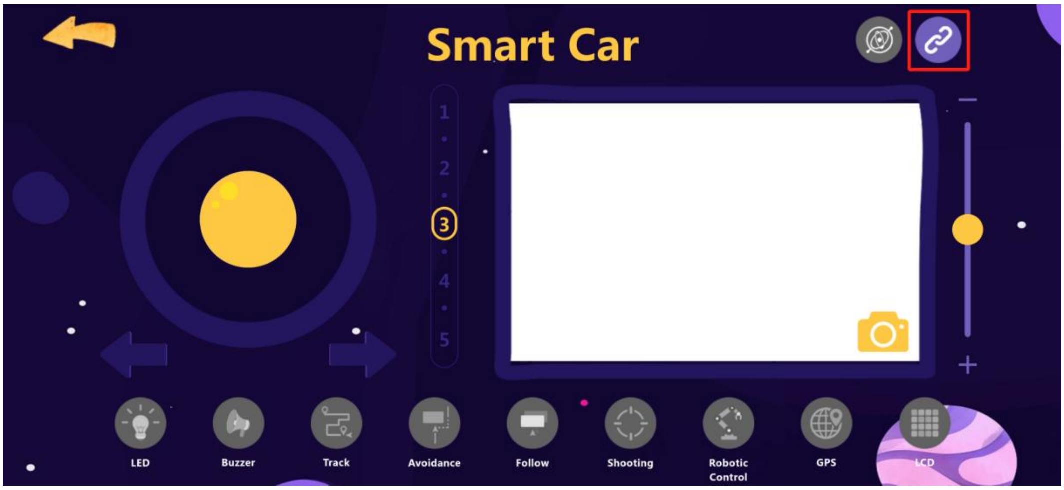


Note: The WiFi name and password can be customized. When we have multiple smart cars, we can distinguish each car by using different WiFi names.

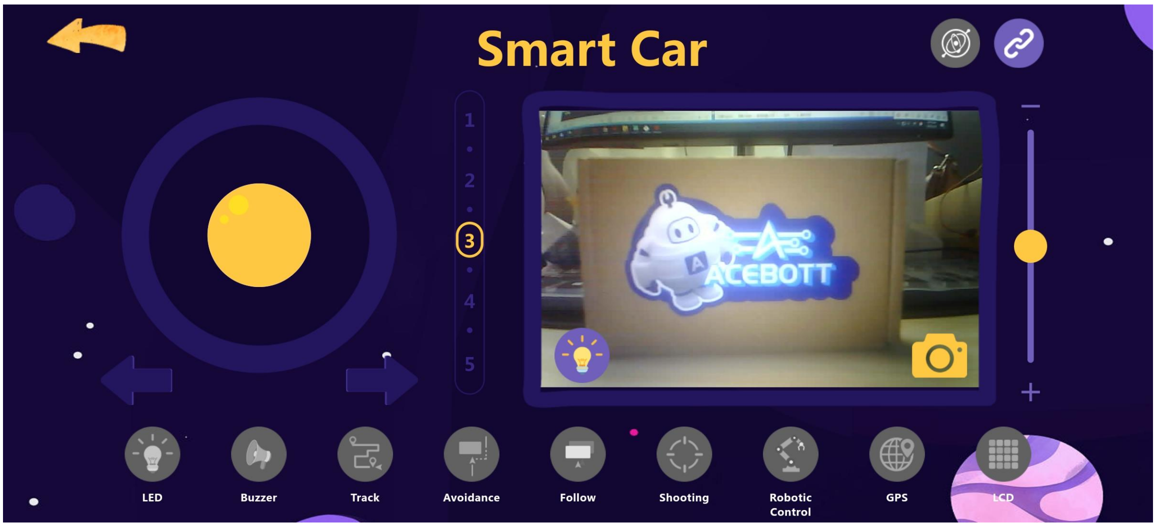
```
const char *ssid = "ESP32-Car";
const char *password = "12345678";
```

3.Use APP control

(1)After connecting to WiFi, click the connection icon in the upper right corner of the APP to complete the connection.

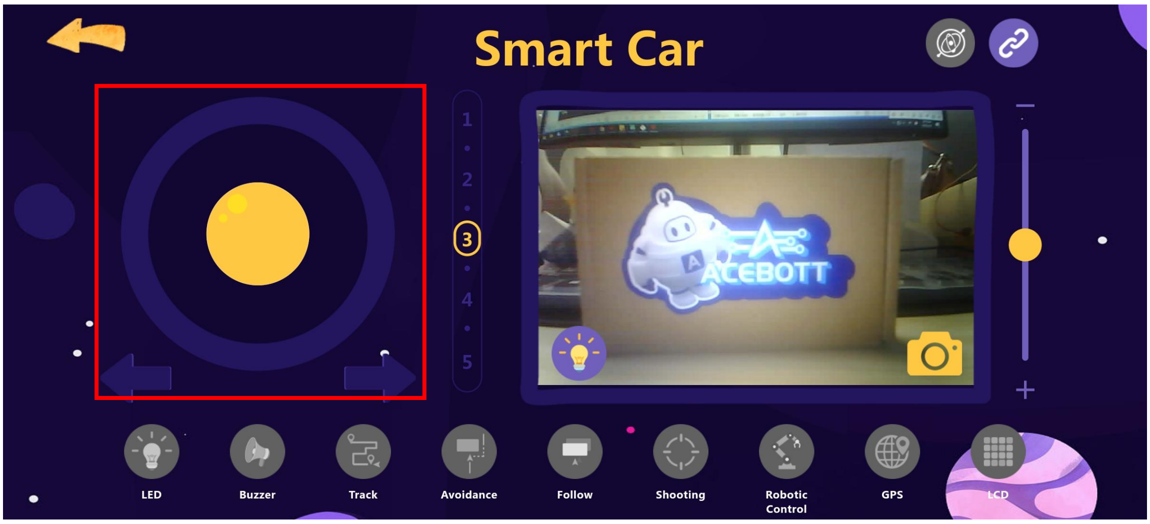


(2)After the connection is successful, the real-time imaging effect of the camera will appear, and then you can control the ESP32-Camera car according to the button prompts.

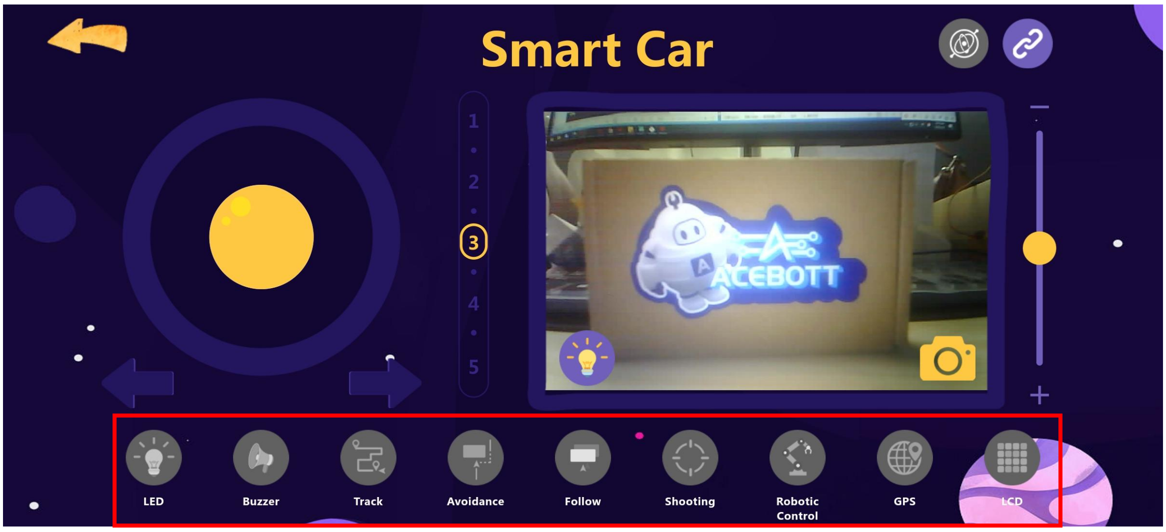


①The left panel controls the basic movements of the car: forward, backward, left move, right move, turn left, turn right; the middle numbers are used for speed adjustment of the car, the larger the number, the faster the speed.

Note: To avoid the car's speed being too low, which affects the normal movement of the car, it is recommended to select a gear of 3 or above for the car's speed.



②Below is a row of buttons for controlling the functions of the car, from left to right, they control the following functions of the car: LED function, music playback (after clicking, there are four types of music to choose from), track mode (after clicking, there are two track modes to choose from), avoidance mode, follow mode, shooting, robotic arm, GPS, LCD function.



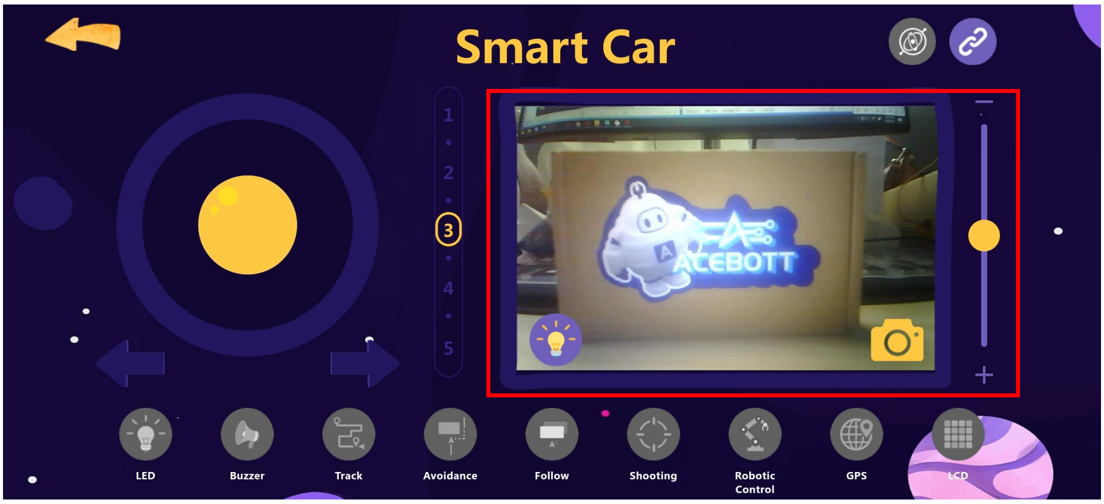
Note:

1. After clicking on the track mode, avoidance mode, or Follow mode, the car will enter a continuous tracking, avoidance, or follow state, respectively. To exit this function, you need to click the corresponding function button again to exit the current state. For example, to turn off track mode, you need to click the track button again to exit the track state.

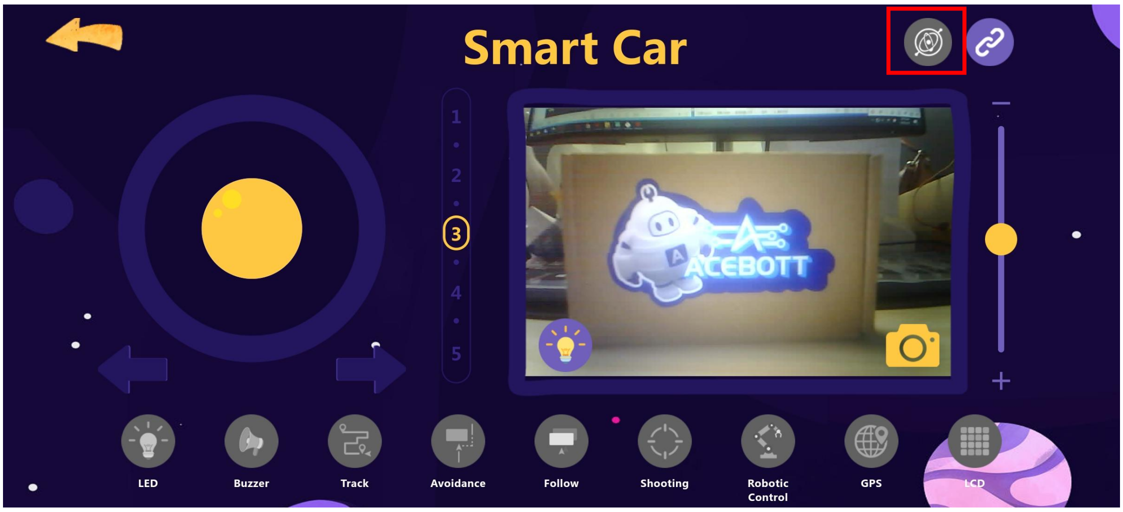
2. Shooting, robotic control, GPS, and LCD are extended functions of the car, which

require the corresponding expansion packs to be added for use.

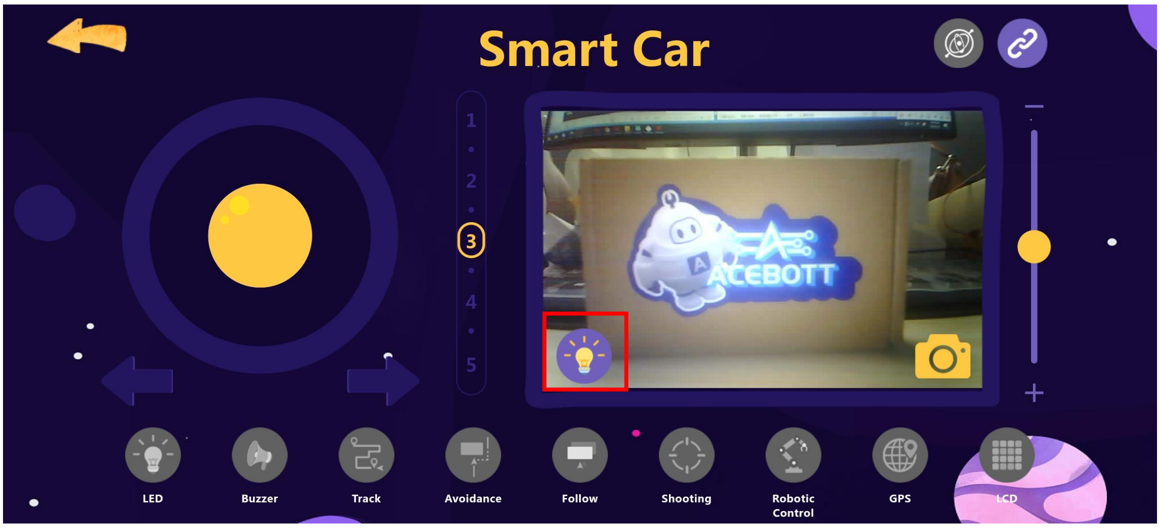
③The central control panel is for the car's camera function. The video frame can display real-time images. After clicking the camera icon in the bottom right corner of the frame, you can save the current image displayed by the camera to the phone's album. The slider on the right is for servo control, which can adjust the pitch angle of the camera.



④In the upper right corner of the smart car's operation interface, there is a gyroscope control provided. After clicking this button, you can control the movement of the car through the gyroscope of your mobile phone. If your phone does not have a built-in gyroscope, you can ignore this feature.



⑤In the bottom left corner of the camera box is the LED light switch button on the camera module. After clicking this button, you can turn on and off the LED light on the camera module.



Follow Us

Scan the QR codes to Follow Us for troubleshooting & the latest news.

We have a very large community that is very helpful for troubleshooting and we also have a support team at the ready to answer any questions.

ACEBOTT FB QR Code

YouTube QR Code