ACEBOTT

Smart Car Starter Kit for ESP32

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 course designed specifically for beginners, aimed at introducing students to the world of programming, electronics, and robotics through a smart car project based on the ESP32. In this course, students will learn how to control a smart car using the ESP32 board, completing a series of challenging tasks from basic LED control to complex wireless remote operations.

With this kit, you can:

1.Learn how to effectively use the ESP32 development board, including uploading code, understanding its features, and coding in the MicroPython.
2.Build a solid foundation in the Python language, as the ESP32 utilizes a simplified version of the Python programming language for controlling circuits and sensors.
3.Explore various electronic components, such as LEDs, sensors, and motors, and understand how they work together in a smart car project.
4.Enhance your maker skills by building your own smart car using the ACEBOTT kit with step-by-step tutorials.
5.Design features for your smart car, such as breathing lights, automatic obstacle avoidance, smart line following, infrared remote control, and WiFi remote control.
6.Develop a comprehensive understanding of smart driving technologies, preparing for more advanced studies in the future.

Overall, the ACEBOTT smart car learning kit is specifically designed for beginners to learn about smart driving technologies based on the ESP32. With this kit, students will be able to independently design and program a multifunctional smart car, understand the basic principles of smart hardware, and have the ability to apply the knowledge learned to solve practical problems.

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

Adventure with Smart Car

In the vastness of the universe, there is a highly developed scientific and technological civilization of the planet, named Planet X. The economy, technology and culture of this planet have reached an extremely advanced level, leading the development of the entire universe. Among them, the most core and developed City is Dream City.In the middle of the city, there is a high-rise building as high as 1000km, towering and spectacular, straight into the sky, is the landmark of the city. This building not only reflects human architectural wisdom, but also represents the peak of Planet X technology. At the top of the building is an Institute dedicated to future technology called the Human Future Institute, or HFI. The HFI brings together the best scientists and researchers in the universe. They come from different planets and civilizations. Every year, the most talented scientists and researchers are selected to join in the "Universe Intelligence Challenge". These researchers will lead mankind to explore higher levels of science and technology, and promote scientific and technological progress throughout the universe. On Planet X, future researchers enjoy the highest honor and power, and their research results and contributions are widely recognized and praised.

One day, I was walking on the way to school, suddenly appeared in front of a mysterious black space tunnel. A strong suction sucked me in, and I felt dizzy and then fell asleep. When I woke up, I found myself in a strange room. I looked around the room. There was a window next to the bed. I went to the window, opened the window, ushered in a dazzling city scene. Towering skyscrapers seem to stretch to the extreme to touch the starry sky, and the exterior walls of the buildings are full of light, as if they are maintained by an unknown energy field. The vehicles on the city streets are suspended in the air, soaring freely, forming a colorful flowing light belt.

I was shocked when a voice suddenly said, "Hello, welcome to Planet X, I'm Ken, the director of the HFI" Then he explained to me, "After our research, you are eligible to be a contestant in the Cosmic Intelligence Challenge. The winner of the competition will be a member of the HFI and can go back to his own planet. Those who fail need to stay on Planet X and learn."

Excited and worried, I asked, "How can I win the competition?"

Ken replied, "When you get the master medal, you can challenge successfully and win the competition.

I asked, "How do I get the master medal?"

Ken replied, "We will have a quest level in the competition, and you will get medals for completing the quest. Four bronze medals can be combined into a silver medal, four silver medals can be combined into a gold medal，and six gold medals can be combined into a master medal."

I asked, "What's the competition about?"

Ken replied, "Every year the competition changes. This year the theme is' Smart Car '. You need to build a fully smart car and make it do the job.

When I heard this, I was a little flustered and upset，thinking:"Is this too hard for me?" When Ken saw your upset look, he smiled and said, "Don't worry. I'll get you an assistant. It's called Lumi, and it's an artificial intelligence robot. If you don't understand anything, you can ask it, and it can help you out." After saying that, the room space suddenly appeared a lovely humanoid robot, it greeted me warmly: "Hello, I'm Lumi and I'm here to help!" Seeing Lumi made my mood a little lighter. I greet Lumi: "Hello, Lumi, nice to meet you." And looking forward to the next challenge.Then Ken smiled and said, "Okay! Little one! I gotta go! The quest will be sent to you via Lumi! It's getting late. Get some rest early and prepare for tomorrow's competition. Good luck with the rest of the competition!" With that Ken and Lumi leave the room.

At this moment, I was alone in the room. I sat quietly on the edge of the bed, gazing out the window at the shuttles and the colorful city that looked like a science fiction movie. My heart is brimming with anxiety about the future. I don't know what the future will bring me. I hope I can return to the earth as soon as possible, and return to the home I know.

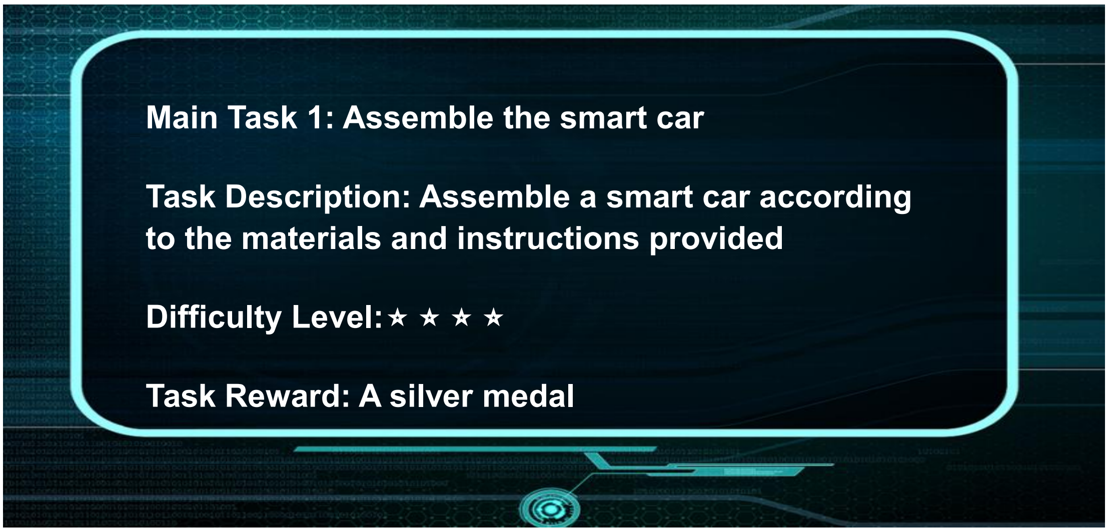
In this strange world, I felt vulnerable and small. I think of those happy days on earth, those warm laughter, they are my innermost strength. I know that only by continuing to move forward and continue to explore, can I return to Earth as soon as possible. My mind swims through a mixture of emotions and eventually sinks into sleep.

Scene 1：Build your own smart car

The next morning, Lumi used brainwaves to enter my consciousness and wake me from sleep, saying, "Don't sleep late, the competition is on."

With that, Lumi emits a virtual screen from overhead, which displays the details of the first task.

Main Task 1: Assemble the smart car



Main Task 1: Assemble the smart car

Task Description: Assemble a smart car according to the materials and instructions provided

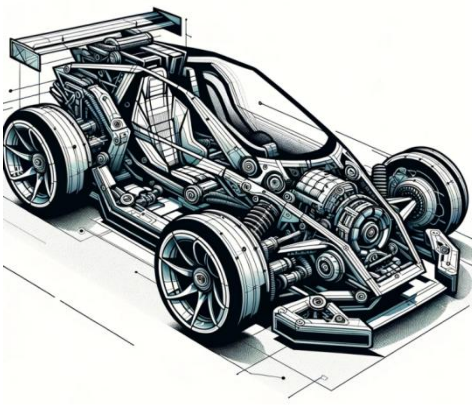
Difficulty Level:⋆ ⋆ ⋆ ⋆

Task Reward: A silver medal

Seeing this task, I realized that I was not on Earth, but on another planet, and I strongly wanted to finish the race and win the prize of returning to Earth. Anxiously, I asked Lumi, "Lumi, how can we assemble a smart car next?"

"Don't worry," Lumi said reassuringly. "We need to understand the structure of the car first, and then find the tools and materials."

Remembering the cars I usually see, I immediately replied, "I know, a car must have wheels, an engine, a chassis, and a body."



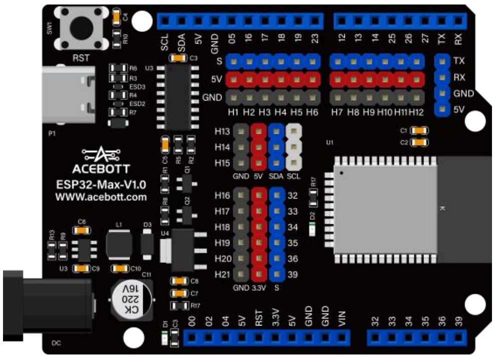
Lumi praised: "It's great, you look at life very carefully, but you still missed an important part?"

"Which part is important?" I wondered.

"To make a smart car, we need it to be as smart as a human being, so it has to have a brain like a human being," Lumi said.

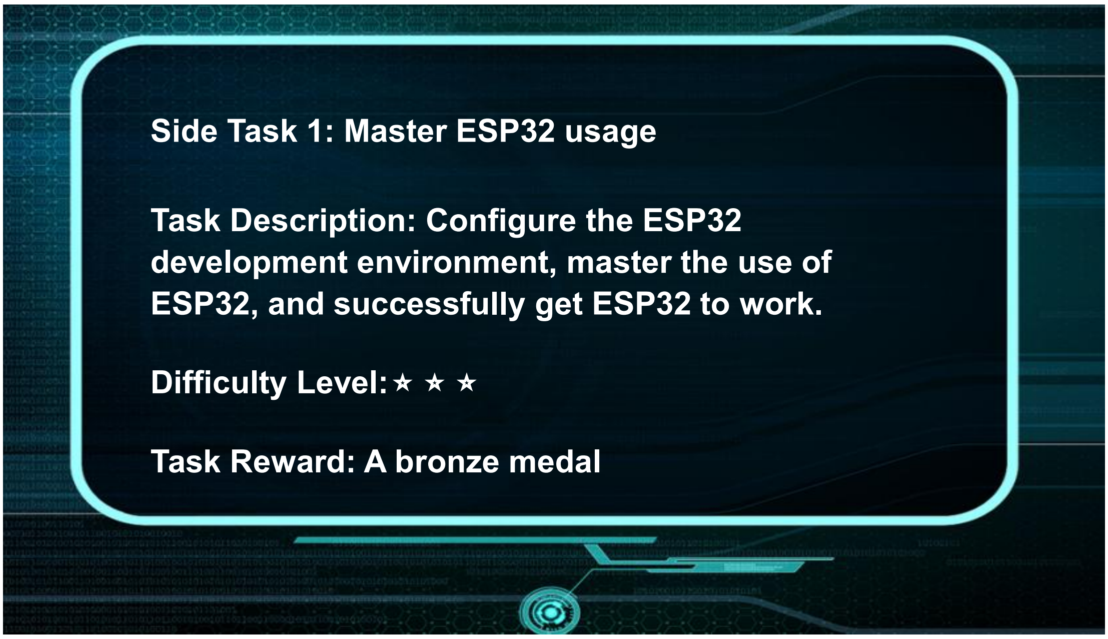
I was shocked and said, "What's the brain of the car?"

Lumi radiates a virtual screen, showing the ESP32 controller board, and says: "In fact, the brain of the smart car is such a controller board, its model is ESP32."



When Lumi finished speaking, there suddenly appeared a sound around "You have triggered the side quest, please complete the side quest", and then Lumi played the side quest information.

Side Task 1: Master ESP32 usage



Side Task 1: Master ESP32 usage

Task Description: Configure the ESP32 development environment, master the use of ESP32, and successfully get ESP32 to work.

Difficulty Level:⋆ ⋆ ⋆

Task Reward: A bronze medal

Nervous and excited, I said, "Another assignment? If it's done, it's one step closer to returning to Earth."

"This controller board is a shell right now," Lumi said, pointing to the ESP32. "To make it work like a brain, we need to program it to tell the ESP32 what to do."

"How do you program it?" I asked, puzzled.

Just after that, Lumi puts his hand on your shoulder and says: "Teleportation". In an instant, you are in another room. Lumi sees you are curious and explains: "Each contestant in the competition has his own laboratory, where you can do research and invention."

Lumi then pointed to the ESP32 on the desk in front of him and said, "To program the ESP32, we need to install software that can edit the program on the computer and upload it to the ESP32."

Lumi continued, "There are a lot of software that can be programmed, and here I recommend using MicroPython to program the ESP32."

"It seems that there is no MicroPython in this computer," you asked.

Lumi replied, "I will give you the instructions to install the MicroPython. You can follow the steps to install the MicroPython on your computer."

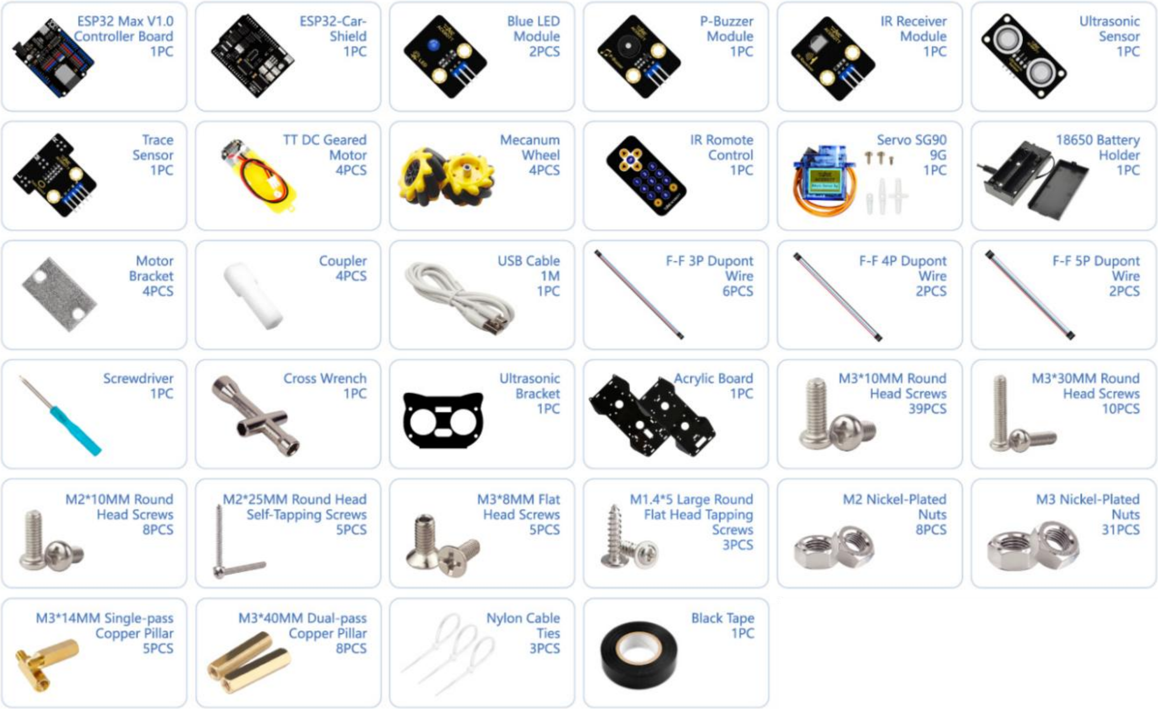
【Click to get the MicroPython installation method】

When I saw the program uploaded to the ESP32 controller board, I was excited and said, "yes, it worked." At the same time, there was a sudden voice around me: "Congratulations, you completed the side quest and earned a bronze medal."

Lumi also congratulated: "Congratulations, you got your first medal, but don't slack off! We still have our main task to do."

Hear this, I accept proud and happy mood, embarrassed to say: "You say right, that we continue to complete the main task!"

Lumi points to the kit on the front table and says, "Before you start the assembly, you need to check whether the kit is ready according to the table below. This is the first step to successfully assemble the smart car."



ESP32 Max V1.0 Controller Board 1PC	ESP32-Car-Shield 1PC	Blue LED Module 2PCS	P-Buzzer Module 1PC	IR Receiver Module 1PC	Ultrasonic Sensor 1PC
Trace Sensor 1PC	TT DC Geared Motor 4PCS	Mecanum Wheel 4PCS	IR Romote Control 1PC	Servo SG90 9G 1PC	18650 Battery Holder 1PC
Motor Bracket 4PCS	Coupler 4PCS	USB Cable 1M 1PC	F-F 3P Dupont Wire 6PCS	F-F 4P Dupont Wire 2PCS	F-F 5P Dupont Wire 2PCS
Screwdriver 1PC	Cross Wrench 1PC	Ultrasonic Bracket 1PC	Acrylic Board 1PC	M3*10MM Round Head Screws 39PCS	M3*30MM Round Head Screws 10PCS
M2*10MM Round Head Screws 8PCS	M2*25MM Round Head Self-Tapping Screws 5PCS	M3*8MM Flat Head Screws 5PCS	M1.4*5 Large Round Flat Head Tapping Screws 3PCS	M2 Nickel-Plated Nuts 8PCS	M3 Nickel-Plated Nuts 31PCS
M3*14MM Single-pass Copper Pillar 5PCS	M3*40MM Dual-pass Copper Pillar 8PCS	Nylon Cable Ties 3PCS	Black Tape 1PC		

Lumi continued: "With all the equipment ready, it's time to start assembling our smart car. Are you ready?"

I excitedly responded, "Yes!"

"You can now follow the steps to assemble the smart car," says Lumi, gesturing across the screen.

【Click to get construction drawings.】

Attention: If you want to watch the assembly video, please click the link below.

https://www.youtube.com/playlist?list=PLkW5fEtHNu6JbnSmm2qSQQ3nkMr6SF59Y

Or scan the QR code below.

I completed the assembly of the smart car according to the tips in the picture and Lumi. While completing the assembly of the smart car, a voice suddenly appeared around, "Congratulations, you have completed the main task, and received a silver medal. The current number of medals is 1 silver and 1 bronze."

Lumi congratulated: "Congratulations on your success in assembling the smart car, next, it will become your most loyal partner and helper in this adventure challenge!"

I stand in the laboratory full of science and technology, looking at the smart car in front of me, in addition to worry, I have more expectations.

Scene 2：Leave for Celestia Canyons

Just as Lumi and I were about to leave the lab, Lumi received a video call from Ken, and Lumi launched the video into the air. Ken saw me and said, "Hello, we meet again."

I politely responded, "Hi, ken."

Ken replied, "Yesterday, our research institute detected signs that the Seth Behemoth under Celestia Canyons is waking up, and if it wakes up, it will bring immeasurable disaster to our planet. Now, we decided to send the participants of the Intelligence Challenge to scout the situation first, and this reconnaissance mission will also become the task of the Intelligence Challenge. Any questions can be directed to Lumi. I hope you will use your courage and wisdom to complete the task. Good luck!"

Ken says this, and the video cuts out. I looked at Lumi in disbelief and asked, "Seth Behemoth?"

Lumi explains, "Seth Behemoth is a giant creature that has been around since the beginning of this planet, and is currently resting in the depths of Celestia Canyons.It's 900 meters long, and it can defend itself against all kinds of weapons. Once it wakes up, it can cause great damage to humans."

"Will we be in danger?" I worried.

"Don't worry," said Lumi. "We are only detecting a weak electromagnetic wave. The monster is still far from waking up, but we need to study and detect the situation to prevent things from becoming more serious."

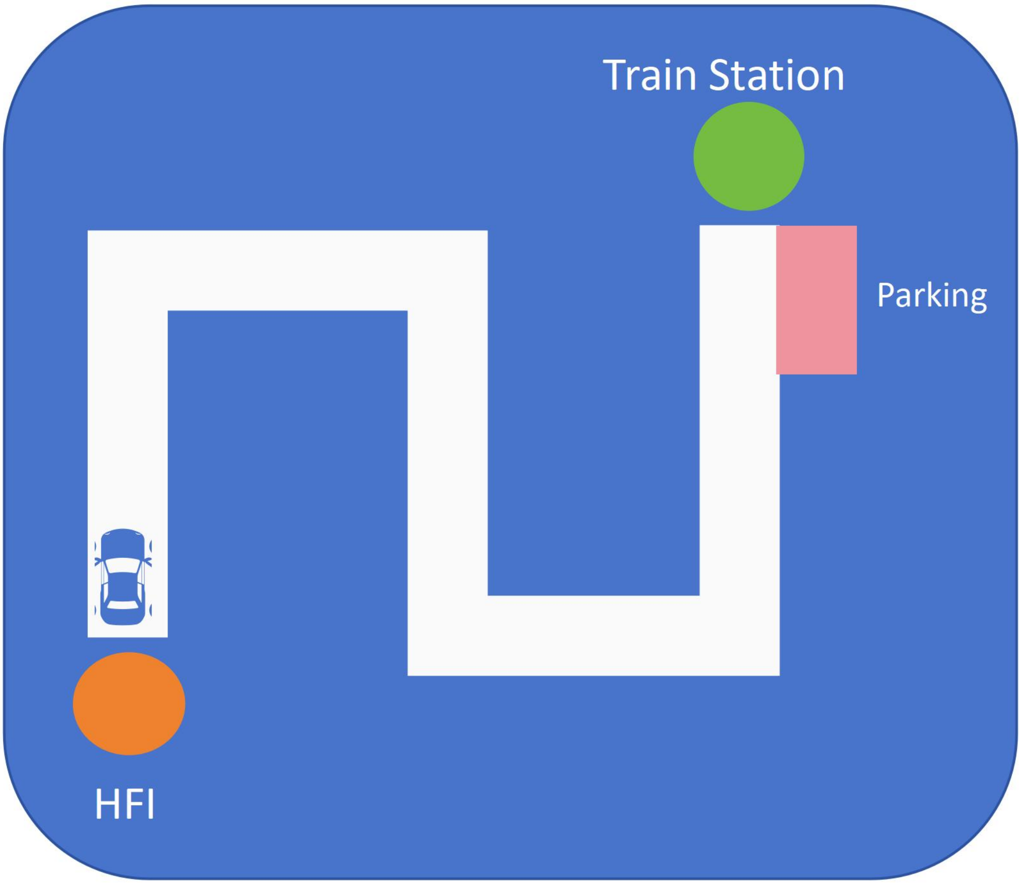
I asked, "Where is the Celestia Canyons?"

Lumi replied, "Celestia Canyons is 30,000 kilometers away from Dream City. We need to take the space train to get there. It only takes about two hours."

I said, "Where is the space train station? How are we supposed to get there?"

I said, "Where is the space train station? How are we supposed to get there?"

Lumi points to the smart car and says, "We can go in the smart car. The space train station is not far from us. Here is the navigation map." Lumi then dropped a map of the space train station in the air.

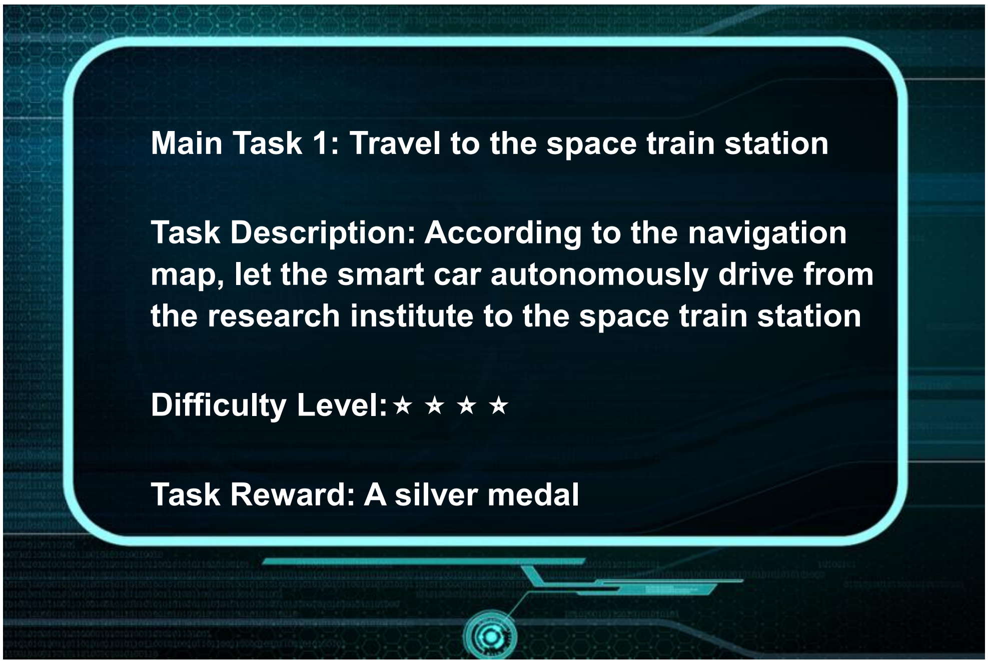


Then I asked, "The smart car can't move yet. How can we make it move and take us to the space train station?"

Lumi replied, "getting the smart car moving is the next thing we need to do."

With that, Lumi released the mission text, which showed the journey to the Space train station.

Main Task 1: Travel to the space train station



Main Task 1: Travel to the space train station

Task Description: According to the navigation map, let the smart car autonomously drive from the research institute to the space train station

Difficulty Level:★ ★ ★ ★

Task Reward: A silver medal

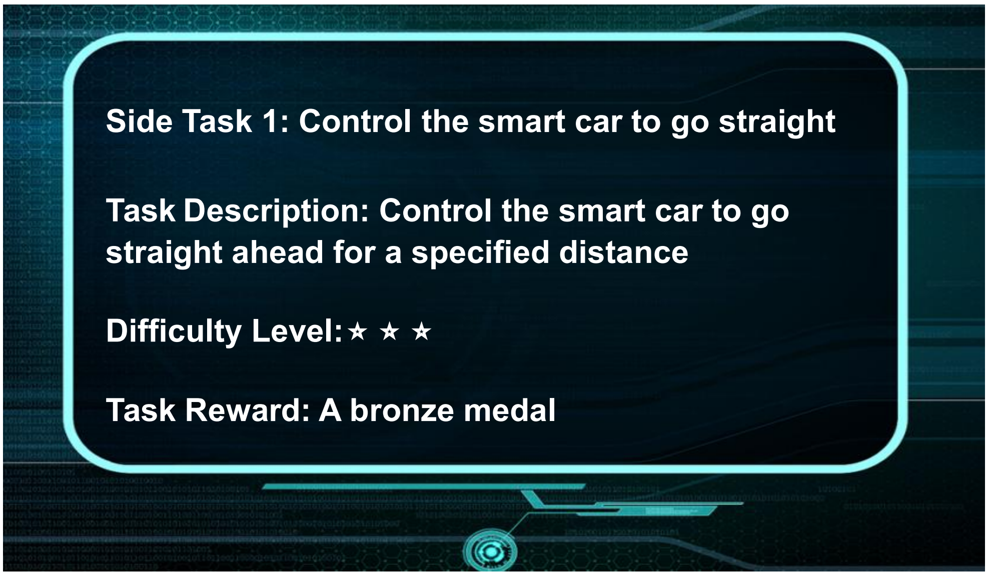
Lumi then asked, "What do you think the smart car needs to do to get to the Space train station?"

Lumi then asked, "What do you think the smart car needs to do to get to the Space train station?"

I looked at the map, thought about it, and said seriously, "To get from here to the space train station, the smart car needs to move forward, turn, and stop."

"You're looking very closely," Lumi said. "The next thing we're going to do is program the smart car to have these capabilities. These three actions correspond to three side tasks." Lumi then showed information about the first side task.

Side Task 1: Control the smart car to go straight



Side Task 1: Control the smart car to go straight

Task Description: Control the smart car to go straight ahead for a specified distance

Difficulty Level:★ ★ ★

Task Reward: A bronze medal

"The power of a smart car comes from its motors," Lumi explains. "To control the car to move is to control the motors to work. There are many types of motors, including servo, DC motor, stepper motor and so on. Different motors are suitable for different scenarios.

I wondered, "What kind of motor does our smart car use? How can we use it to control the movement of the smart car?"

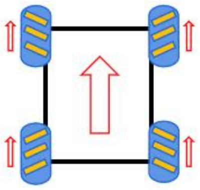
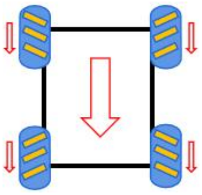
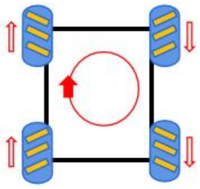
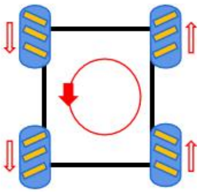
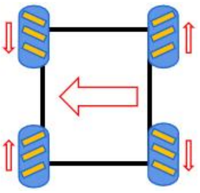
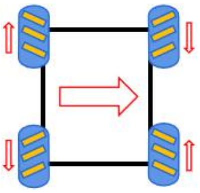
"We use DC motors," Lumi replied. You just need to supply enough voltage to the motor to turn. The motion of the smart car is not only related to the motor, but also related to its wheels. Our smart car uses McNum wheels, which allows the smart car to achieve multiple motions."

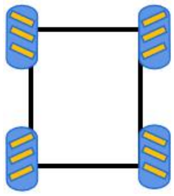
Lumi continues, "Smart cars can be difficult to control due to the variety of motion modes. In order to make it easier for you to control the motion of the smart car, we have prepared a powerful library for you, which provides a lot of command tools to control the motion of the smart car. You just need to install this library into the MicroPython development environment and you can easily control the movement of your smart car."

"Here are the library resources and instructions for installing the library," Lumi replied, gesturing to the screen.

I successfully installed the library resources provided by Lumi into MicroPython by following the installation instructions.

Lumi saw that I had installed the library and said, "OK, let's see what commands we have for motion control of the smart car! You just need to give different parameters, and the smart car can make many different motion states." After that, Lumi gave the control diagram and function of McNum wheel, in which the speed of the smart car can be selected from 0 to 255, and 255 is the fastest speed.

Motion states	Action diagram	Control command
Forward		car.move(car.Forward,speed)
Backward		car.move(car.Backward,speed)
Clockwise		car.move(car.Clockwise,speed)
Contrarotate		car.move(car.Contrarotate,speed)
Move_Left		car.move(car.Move_Left,speed)
Move_Right		car.move(car.Move_Right,speed)

Stop		car.move(car.Stop,speed)

Lumi continued, "From the graph, we can see that if the smart car is going to go straight, the first parameter of the Move function needs to be 'car.forward' so that all four wheels of the smart car are in the same direction. Here is the smart car linear movement program, you upload it to the smart car to try."

Click to get Go_straight program.

```
import libs.vehicle #Import car library
import time

car = libs.vehicle.ACB_Vehicle() #Creating a car object

car.move(car.Forward, 255) #Control car forward moving
time.sleep(3) #Control the movement time of the car
car.move(car.Stop, 0) #Control car stop
```

I couldn't wait to upload the code, and saw the smart car start to move forward, but after a while it stopped, and there was still a long way to go before the corner needed to be turned. So I asked Lumi, "How do I change the distance the car travels?"

Lumi explains, "Distance is speed times time! When the speed is determined, you try changing the time parameter to see what happens."

```
import libs.vehicle #Import car library
import time

car = libs.vehicle.ACB_Vehicle() #Creating a car object

car.move(car.Forward, 255) #Control car forward moving
time.sleep(3) #Control the movement time of the car
car.move(car.Stop, 0) #Control car stop
```

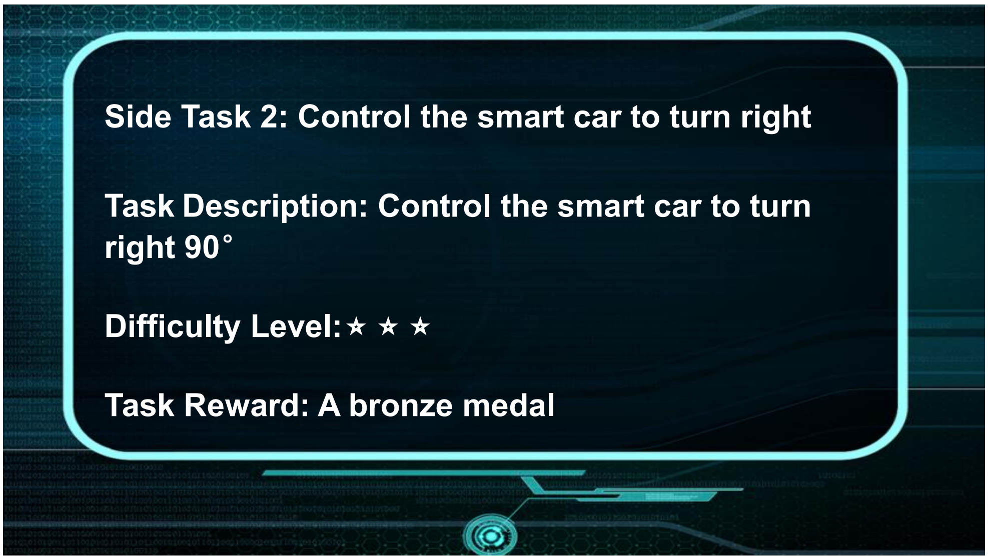
I modified the time parameters according to the tips given by Lumi, and the smart car finally realized the function of go straight. I happily said to Lumi: "Lumi, I have succeeded, I have mastered the way to control the smart car forward!"

"Yes, congratulations, you have won another bronze medal," Lumi said with a smile. "Then there is the task of turning and stopping!"

I confidently said, "Lumi, let me do the turning and stopping of the smart car!"

Lumi nodded happily: "OK, come on! I'll show you the second side quest."

Side Task 2: Control the smart car to turn right



Side Task 2: Control the smart car to turn right

Task Description: Control the smart car to turn right 90°

Difficulty Level:★ ★ ★

Task Reward: A bronze medal

I looked at the instructions, combined with the McNumm wheel motion diagram from earlier, and said, "If the smart car is going to turn right , it is Clockwise, so use myCar.move(Clockwise,speed)to control the smart car."

"That's great," Lumi said in praise. "How do you turn 90 degrees?"

I immediately replied, "Just like going straight, you can control the Angle of rotation by changing the rotation time."

I immediately replied, "Just like going straight, you can control the Angle of rotation by changing the rotation time."

Lumi responded: "Good, you seem to understand the principle of smart car motion control, this is the smart car turning program, let's make the car turn ."

Click to get Turn_right program.

```
import libs.vehicle
import time

car = libs.vehicle.ACB_Vehicle()

car.move(car.Clockwise, 255) #The car rotates clockwise
time.sleep(0.75)
car.move(car.Stop, 0)
```

I uploaded the program, and modified the time, constantly debugging the program, finally let the car to achieve turn right 90° .

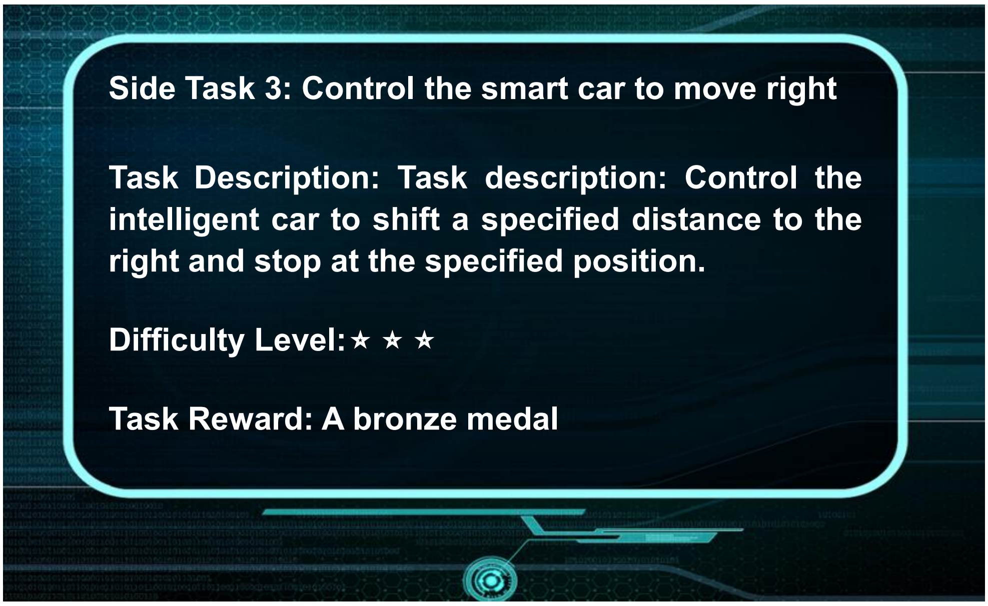
I was overjoyed. "It's finally done," I said. "We just have to stop the car, and we can go to the space train station."

At the same time, Lumi's voice rang out: "Very good, you now have a bronze medal, we only need the final parking function.The traffic rules in Dream City are very strict. You have to park your car in the designated parking space or you will be fined heavily." Lumi then zoomed in on the map of the parking lot at the space train station.

I looked at the parking spaces at the Space train station on the map and found that the parking spaces were on the side of the road, so I proposed a solution: "On Earth, side parking is a technical challenge. However, because of the characteristics of the McNum wheel, the smart car can realize the translation from left to right, so it can simply realize the function of side parking."

Lumi responds, "Absolutely, it seems your logic is getting clearer. Side parking is also our third side mission."

Side Task 3: Control the smart car to move right



Side Task 3: Control the smart car to move right

Task Description: Task description: Control the intelligent car to shift a specified distance to the right and stop at the specified position.

Difficulty Level:⋆ ⋆ ⋆

Task Reward: A bronze medal

Lumi continued, "Are you now confident that you can do this task on your own after practicing the last few tasks?"

I answered confidently, "No problem!" Then I opened the software and entered the following code.

Click to get Move_right program.

```
import libs.vehicle
import time

car = libs.vehicle.ACB_Vehicle()

car.move(car.Move_Right, 255) #Control car to move right
time.sleep(0.75)
car.move(car.Stop, 0)
```

I uploaded the program to the smart car, and tried to modify the time parameter, and realized the function of the smart car to shift the specified distance to the left.

As I finished the task, Lumi's voice rang out, "You have successfully implemented the car forward, turn, and side parking functions, and you have already earned three bronze medals for these three tasks. Next, we need to integrate these three functions of the smart car, navigate according to the map, and let the smart car drive to the space train car." With that, Lumi gave me the final program for the smart car to go to the space train station.

Click to get Go_to_train_station program.

```
import libs.vehicle
import time

car = libs.vehicle.ACB_Vehicle()

car.move(car.Forward, 255) #Control car forward moving
time.sleep(3)
car.move(car.Clockwise, 255) #Control car counterclockwise rotate
time.sleep(0.75) #Modify the parameters in red
car.move(car.Forward, 255) #Control car forward moving
time.sleep(1.5)
car.move(car.Clockwise, 255) #Control car counterclockwise rotate
time.sleep(0.75) #Modify the parameters in red
car.move(car.Forward, 255) #Control car forward moving
time.sleep(3)
car.move(car.Contrarotate, 255) #Control car contrarotate
time.sleep(0.75) #Modify the parameters in red
car.move(car.Forward, 255) #Control car forward moving
time.sleep(1.5)
car.move(car.Contrarotate, 255) #Control car contrarotate
time.sleep(0.75) #Modify the parameters in red
car.move(car.Forward, 255) #Control car forward moving
time.sleep(3)
car.move(car.Move_Right, 255) #Control car left moving
time.sleep(0.75) #Modify the parameters in red
car.move(car.Stop, 0) #Control car stop
```

Lumi then cautions, "Debug and change the parameters to get there!"

I uploaded the program given by Lumi, and continued to debug and modify the parameters. Finally, the smart car completed the upgrade of the automatic driving function.

Next, before we boarded the smart car to the space train station, Lumi reminded me: "Remember to bring your computer, it will be your most important tool for the rest of the journey."

After a 20-minute ride, we finally arrived at the space train station. After we parked the car, Lumi and I quickly got out of the car. Lumi congratulated me, because I had 2 silver medals and 4 bronze medals, and since the bronze medals had reached the limit, they would be automatically converted to silver medals, so I had 3 silver medals in total.

I looked up, a huge hemispherical building suspended in the air, giving people a shocking feeling. "This is the space train station," Lumi said, pointing to the building in the sky. "In front of it is the passage to the station."

Following Lumi's guidance, I found a huge light door under the space train station, through which many passengers were coming in and out. Then we entered the light door, into the lobby of the Space Train station.

In the center of the station is a huge revolving hall with unique design and a sense of technology. At the top is a translucent stargazing roof, allowing people to enjoy the beautiful view of outer space, I can't help but marvel at this extremely creative design.

We went to the ticket office and bought our tickets to Celestia Canyons. Then, through the automatic guidance system, we successfully boarded one of the trains, and the smart car was intelligently hosted so that it could follow the train.

Soon, the train started. It hurtled through the atmosphere at astonishing speed, heading for the Celestia Canyons.I sat in the train and watched the light flow through the window, looking forward to the rest of the journey.

At this point, Lumi said, "The rest of this journey is going to be a wonderful adventure. Celestia Canyons is a mysterious and beautiful place and we will have the opportunity to enjoy some of the most spectacular views in the universe."

I nodded excitedly, full of anticipation and excitement. In this day and age, I feel incredibly lucky to have the opportunity to experience such a technological marvel first-hand.

As the train continued to move forward, we began a new journey. The light outside the window changes constantly, as if telling a mysterious story. Lumi and I are ready for the next adventure and challenge.

Scene 3：Through the Dark Forest

After a long and winding two-hour journey, we finally set foot on the land of Celestia Canyons. Once out of the station, we drove the smart car to the entrance of Celestia Canyons according to the navigation instructions. However, this is only the beginning of a long journey, from here to the abyss of Celestia Canyons, we still have to experience an unknown journey.

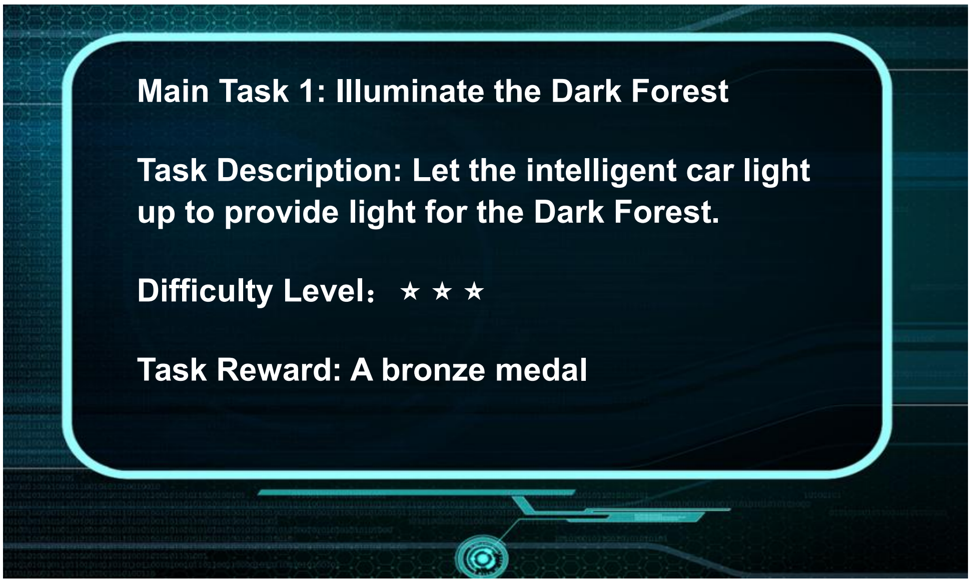
First, we need to go through the dark and endless forest. The Dark Forest, a fearsome name, is like a thick black barrier that hides the depths of Celestia Canyons in deep mystery. The scope of this forest stretches for 20 kilometers and consists of countless huge and dense trees, each of which is about 50 meters high. The canopy of the tree covers the sky, and the outside light cannot penetrate the thick leaves. Therefore, the Dark Forest is a dark environment all year round, and it is full of unknown dangers.

I looked at the Dark Forest in front of me, a little scared in my heart. I couldn't help but look at Lumi. My voice trembled and asked, "Lumi, do we have to go through this road?"

Lumi replied, "This is the only way to get to Celestia Canyons. We have no other choice. We have to go through the Dark Forest."

"Don't worry," says Lumi. "Our smart cars have lights. We can let them provide us with light.Let us better through this forest, light the Dark Forest is our first main task." With that, Lumi opened the mission screen, which said: Light the Dark Forest.

Main Task 1: Illuminate the Dark Forest



Main Task 1: Illuminate the Dark Forest

Task Description: Let the intelligent car light up to provide light for the Dark Forest.

Difficulty Level：★ ★ ★

Task Reward: A bronze medal

Lumi continues: "The headlights are the lighting system of our smart car and help us to dissipate the darkness of the Dark Forest. At present, the main types of car lights are halogen lights, xenon lights, LED lights. Our car lights are LED lights."

With an expectant look on my face, I asked, "How do you light the LED?"

"All electronic components need power and energy to work properly, and leds are no exception," Lumi explains with a smile. "If you want to light up an LED, you need to put power into it, and the smart car uses different kinds of power to make the led light on and off."

Lumi continued: "This program turns the lights on and you can try them out."

Click to get Turn_on_LED program.

```
import time
from machine import Pin #Import the pin

left_led = Pin(2, Pin.OUT) #Define the pin number of the LED
right_led = Pin(12, Pin.OUT)

while True:
    left_led.value(1) #Turn on the LED, "1 "means high level and "0" means low level
    right_led.value(1)
```

As soon as the code was uploaded, a bright light shot out of the front of the car, illuminating the road ahead. Gradually the darkness lifted, and you saw a wide, flat path that led deep into the forest.

"See, we have the light now, and we can march on bravely!" "Said Lumi encouragingly.

Lumi and I carefully drove through the forest, the smart car's LED lights illuminating the road ahead to guide us. As we reached the heart of the Dark Forest, we felt a sudden foreboding. Something seemed to be staring at me. I stopped and paused, listening carefully to the sounds around me.

"Roar ~" A roar from the depths of the forest, my heart beat fast. Worried, he asked, "Lumi, what's with the roar?"

"These are animals of the Dark Forest," Lumi explained. "Because they live in the Dark Forest all the time, their eyes are very sensitive to light. Our sudden bright light stimulated them, so they reacted with fear and anger."

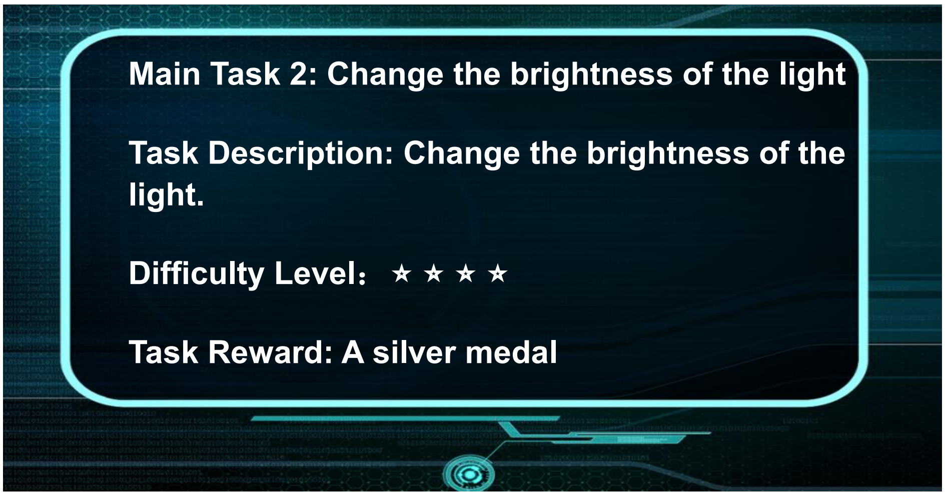
My heart tightened, and I realized that our light might also lead to danger. However, in order to move forward, we must turn on the lights to get through the Dark Forest.

"Lumi, do you have a good solution?"

"The reason they're scared is because our lights are too bright, which makes them feel uncomfortable," Lumi says with a smile. By controlling the brightness of the lights, we can make the lights adjust slowly and gradually to reduce the impact on the surrounding organisms."

I understand what Lumi means, controlling the brightness of the headlights so that it gradually increases or decreases over time, just like breathing. This ensures that we can see clearly without causing discomfort to the surrounding creatures.

Main Task 2: Change the brightness of the light



Main Task 2: Change the brightness of the light

Task Description: Change the brightness of the light.

Difficulty Level：★★★★

Task Reward: A silver medal

"We've just learned that LED need energy to work," Lumi said. Sufficient power supply can make the LED light normally. What happens if I reduce the amount of power I give to the leds?"

"Darkening?" I replied, guessing.

Lumi smiled and said, "Yes. If I reduce the amount of energy I give to the LED, its brightness will decrease accordingly. If I increase the amount of energy I put into the LED, its brightness will increase accordingly."

I immediately replied, "Oh! Then I understand, I can control the brightness of the lights by controlling the controller board to output different electrical energy."

Lumi smiled and said, "Smart! The amount of power is related to the voltage, and we can adjust the power provided by the controller board to the LED by changing the voltage output of the controller board."

I wonder: "How to change the controller board output voltage?"

Lumi replied: "ESP32 controller board can use PWM technology to change the output voltage, the following is the corresponding program, you can upload the program to see if the brightness of the lights has changed compared to before."

Click to get LED_Brightness program.

```
import time
from machine import Pin, PWM
#freq is an integer that sets the frequency (in Hz) of the PWM cycle;
left_led = PWM(Pin(2), freq=1000)
right_led = PWM(Pin(12), freq=1000)

#Change the red part of the parameter, the parameter range 0-1023
#so that the lights display different brightness
left_led.duty(500)
right_led.duty(500)
```

I uploaded the above program and found that the brightness of the light is much dimmer than before.

Lumi continued, "You can change the red parameter in the above program, which ranges from 0-255, corresponding to the output voltage range of 0-5V."

Lumi looked at me constantly modifying and debugging the voltage parameters in the code to achieve different brightness of the lights, so he said: "Congratulations, you have mastered the method of controlling the brightness, next, we need the brightness of the lights gradually from the darkest to the brightest, and then from the brightest back to the darkest? Here is the program."

Click to get Breathing_light program.

```
import time
from machine import Pin, PWM

left_led = PWM(Pin(2), freq=1000)
right_led = PWM(Pin(12), freq=1000)

while True:
    # Gradually Turn on the headlights
    for i in range(0, 1023, 1):
      left_led.duty(i)
      right_led.duty(i)
      time.sleep_ms(1) #The delay is 1 millisecond
    # Gradually extinguish the headlights
    for i in range(1023, 0, -1):
      left_led.duty(i)
      right_led.duty(i)
      time.sleep_ms(1)
```

With the upload of the program, the headlights become the effect of breathing lights. The lights change gradually from bright to dark and then from dark to bright, giving a quiet and calm feeling.

"Great! "Now we just need to keep quiet and try not to disturb the creatures, and we can get through the forest safely."

So we started driving the smart car carefully, trying not to make any noise. The brightness of the headlights is constantly adjusted over time, allowing us to move forward smoothly.

While walking through this Dark Forest, we saw many wonderful creatures and sights. Some creatures have shimmering scales, while others have strange vocal organs that make beautiful sounds. And the strange plants were of different shapes, some with huge flowers, others with glittering fruits.

"See, that's the wonder of the Dark Forest." "It's a place of uncertainty and danger, but it's also a place of incredible life," Lumi said. As long as we are vigilant and respectful, we will make it through the forest."

After 30 minutes of careful driving, we finally made it through the eery Dark Forest, and I immediately turned off my headlights so as not to draw unwanted attention. I turned to Lumi with unmasked excitement and asked excitingly, "Lumi, we have now successfully traversed the Dark Forest, and I would like to know how many medals I have now?"

Lumi responded with a warm smile. "Congratulations, so far you have completed your mission in the Dark Forest. You have earned a silver medal and a bronze medal. Since the silver medal limit has been reached, 1 gold medal is automatically synthesized. So, you now have a total of 1 gold and 1 bronze medals, to keep working hard oh!"

Scene 4：Across the Starfall

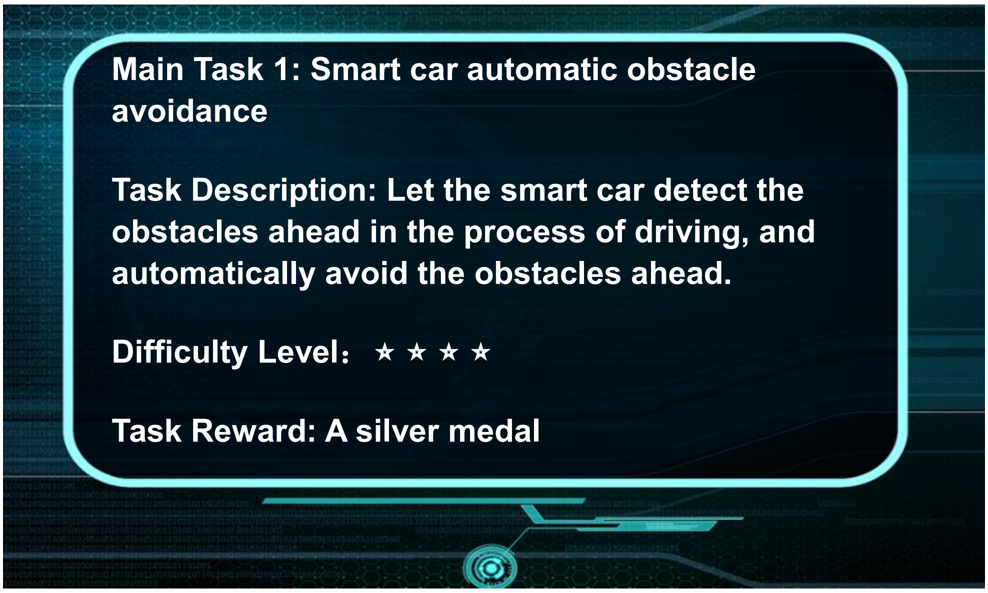
When we broke free from the Dark Forest, we found a desolate loess plain. It was in stark contrast to the Dark Forest before it, as if it were two different worlds. In this vast loess earth, countless meteorites scattered irregularly, their shapes are different, different sizes, some are square, some are round, some two or three meters high, some tens of meters high.

We need to cross this meteorite field to get to the abyss. The road situation is worrying, because these meteorites block the path of the smart car, forming countless obstacles.

I looked at the meteorite area and asked with concern: "There are so many meteorites blocking the path of the smart car, can the smart car smoothly take us through?"

"As long as we make the smart car have the ability to avoid obstacles automatically, we can safely pass through the Starfall, which is the next task we need to complete," Lumi said. With that, Lumi presented the task information.

Main Task 1: Smart car automatic obstacle avoidance



Main Task 1: Smart car automatic obstacle avoidance

Task Description: Let the smart car detect the obstacles ahead in the process of driving, and automatically avoid the obstacles ahead.

Difficulty Level：★★★★

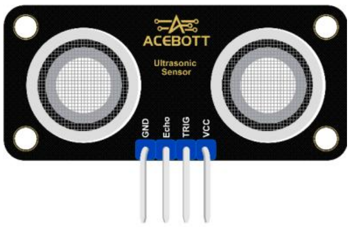
Task Reward: A silver medal

"How can a smart car detect meteorites in front of it?" I wondered.

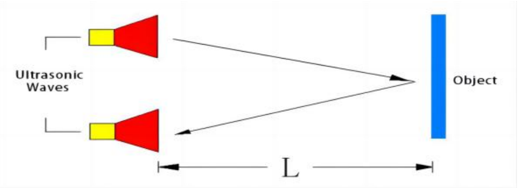
Lumi replied: "We have installed ultrasonic sensors in our smart car, which is like the eyes of the smart car, so that the smart car can sensitively detect obstacles in front of it."

I heard this, immediately came to the interest: "Ultrasonic sensor? How does it work?"

"There are two probes on the ultrasonic sensor: one acts as a transmitter to send the ultrasonic signal, and the other acts as a receiver to receive the reflected signal," Lumi explains.



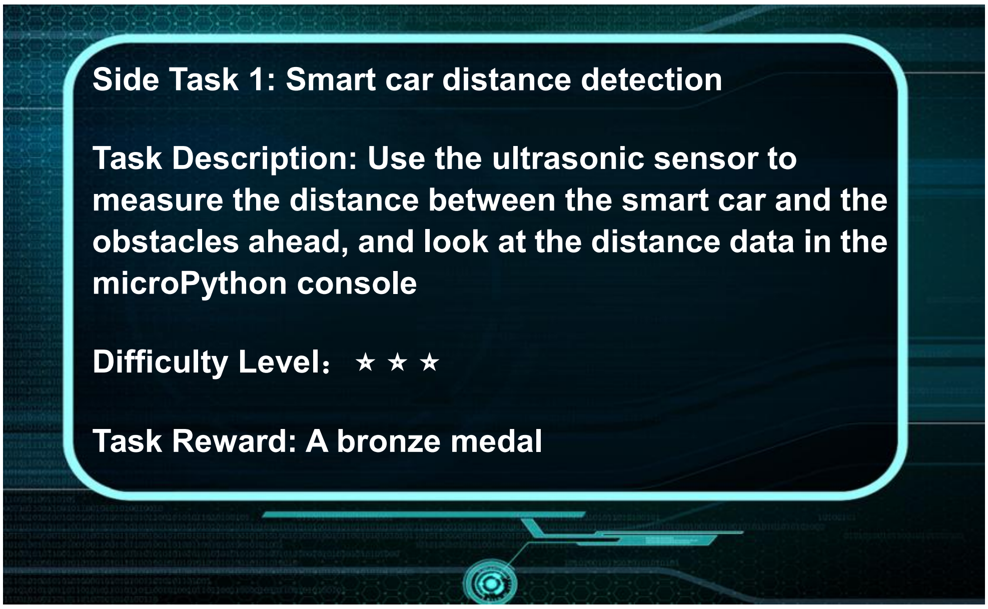
Lumi continued: "The ultrasonic transmitter sends an ultrasonic wave in a certain direction, it travels through the air, it hits an obstacle on the way, and it returns immediately, and the ultrasonic receiver receives the reflected wave. By converting the calculation, the distance L between the ultrasonic wave and the obstacle in front is obtained."



I asked, "How do we get this distance L?"

"We're going to get this data through programming," Lumi said, and then showed the distance measurement task information.

Side Task 1: Smart car distance detection



Side Task 1: Smart car distance detection

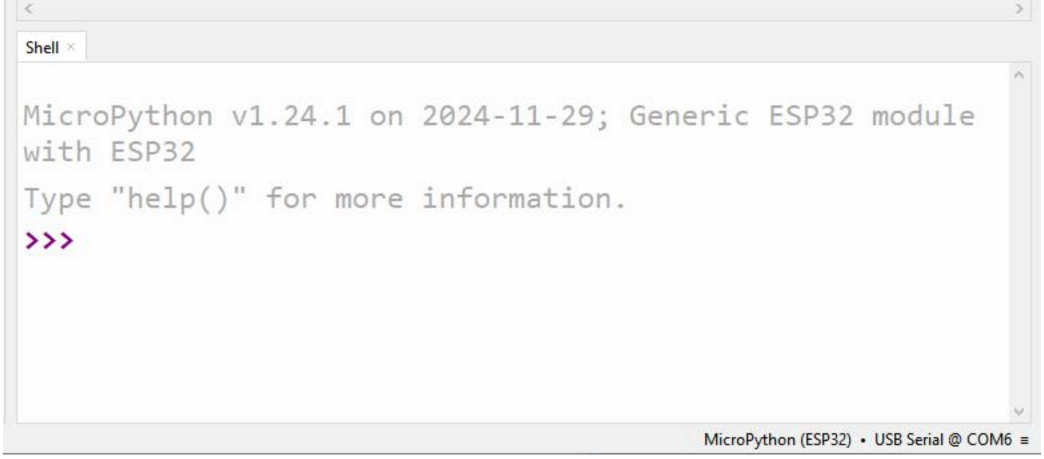
Task Description: Use the ultrasonic sensor to measure the distance between the smart car and the obstacles ahead, and look at the distance data in the microPython console

Difficulty Level：★★★

Task Reward: A bronze medal

I am confused and ask: "microPython console?"

Lumi explained: "microPython console is a built-in interactive Python interpreter environment that allows developers to direct execution in microPython Python code snippet, convenient developers write code at the same time see the results immediately."



```
MicroPython v1.24.1 on 2024-11-29; Generic ESP32 module
with ESP32
Type "help()" for more information.
>>>
```

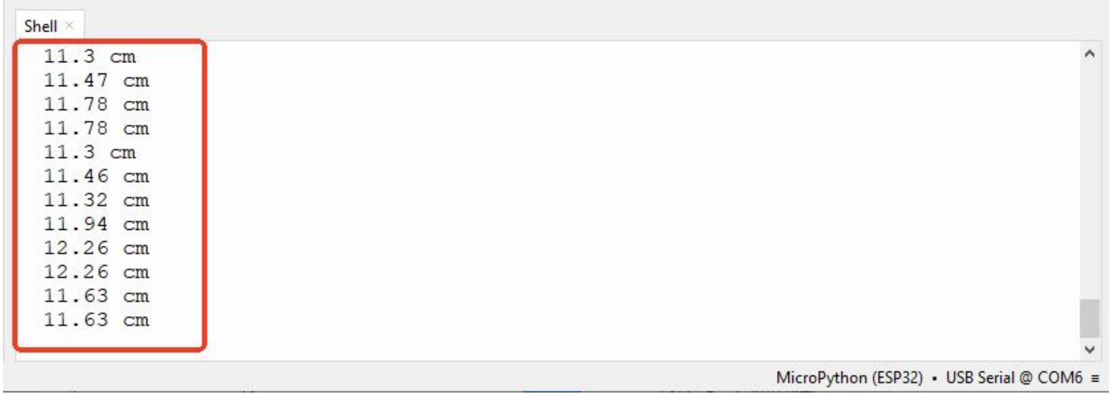
"Great, how do I program the distance to get the ultrasound?"

Lumi responded: "Next let's look at the program of ultrasonic ranging, this is the program of ultrasonic ranging, quickly let the car achieve the function of ranging."

Click to get Ultrasonic_ranging program.

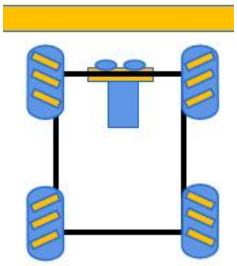
```
import libs.ultrasonic
import time

Trig_PIN = 13
Echo_PIN = 14

Ultrasonic = libs.ultrasonic.ACB_Ultrasonic(Trig_PIN,Echo_PIN)

while True:
    UT_distance = Ultrasonic.get_distance()
    #The distance of the ultrasonic detection
    print(UT_distance,"cm")
    time.sleep(1)
```

After I uploaded the program, I found the Control console continued to display the detected distance.



```
11.3 cm
11.47 cm
11.78 cm
11.78 cm
11.3 cm
11.46 cm
11.32 cm
11.94 cm
12.26 cm
12.26 cm
11.63 cm
11.63 cm
```

At the same time, Lumi's voice said, "Great, you have mastered the ultrasonic detecting function, now we can start implementing the obstacle avoidance function. Think about what the smart car needs to do if it encounters an obstacle. Lumi then presented information about side task two.



I immediately replied, "Just turn around in place."

Lumi said, "That's good. The logic is clear. Do you remember how to control the smart car turning in place and the Angle of rotation?"

I replied confidently, "Of course, you can control the rotation Angle of the car by changing the rotation time."

Lumi responded, "Great, so let's make it obstacle-avoiding."

Click to get Obstacle_avoidance program.

```
import libs.vehicle
import libs.ultrasonic
import time

Trig_PIN = 13
Echo_PIN = 14
car = libs.vehicle.ACB_Vehicle()
Ultrasonic = libs.ultrasonic.ACB_Ultrasonic(Trig_PIN,Echo_PIN)

while True:
    UT_distance = Ultrasonic.get_distance()
#The distance is less than 25cm to achieve the effect of turning
    if UT_distance <= 25:
        car.move(car.Contrarotate, 180)
        time.sleep(1.5) #The turning time is modified to realize the rotation of different angles
        car.move(car.Stop, 0)
    else:
        car.move(car.Forward, 150) #If the distance is greater than 25, move forward
```

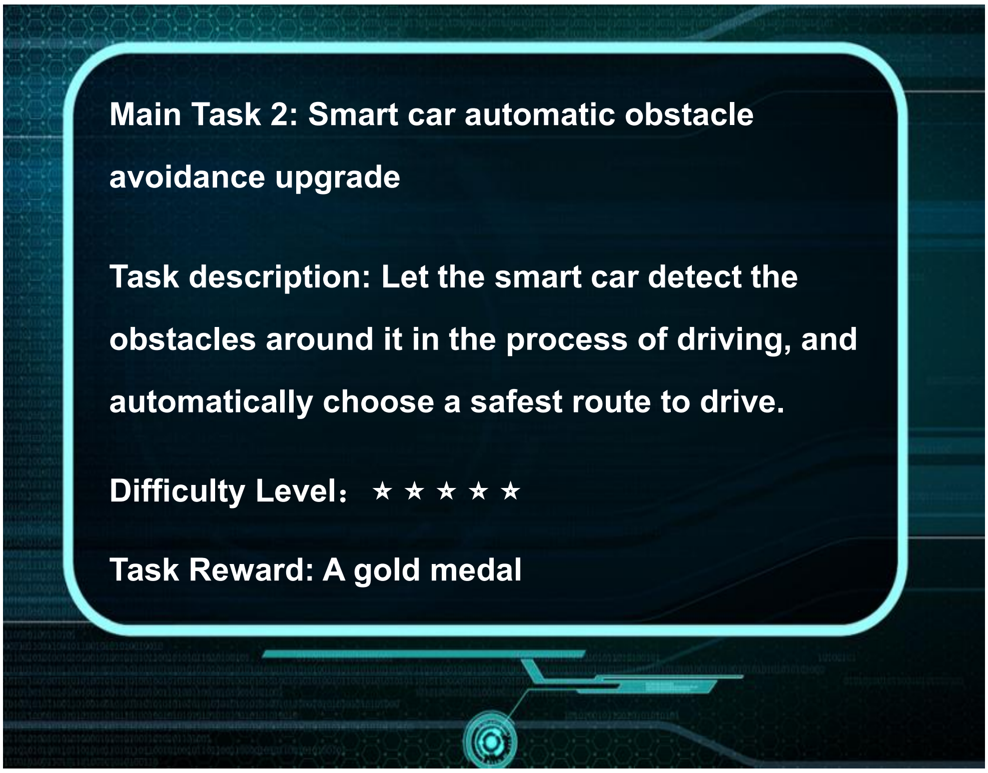
I uploaded the program, modified the turning time of the smart car (red code area), and constantly debugged the program, so that the smart car obtained the function of obstacle avoidance.

Note: It is necessary to turn the steering gear to face the ultrasonic sensor straight ahead.

We began to drive slowly into the Starfall plains. There were not so many meteorites in the Starfall plains at the beginning, so the smart car went smoothly. But as we went deeper and deeper, there were more and more meteorites on the road, and the road conditions became more and more complex, so the previous simple obstacle avoidance function could not meet the following road conditions.

Seeing that the smart car could not drive in the dense meteorite area, Lumi immediately said: "We need to upgrade the obstacle avoidance function to traverse this dense meteorite area, which is our second main task." Lumi then presented the task information.

Main Task 2: Smart car automatic obstacle avoidance upgrade



Main Task 2: Smart car automatic obstacle avoidance upgrade

Task description: Let the smart car detect the obstacles around it in the process of driving, and automatically choose a safest route to drive.

Difficulty Level：★ ★ ★ ★ ★

Task Reward: A gold medal

"How do I upgrade obstacle avoidance?" I asked.

Lumi replied, "We use the servo to help the car avoid obstacles and choose a safe road to drive."

I immediately replied: "I know the servo, you said before, it is a kind of motor."

Lumi smiled and said, "Yes! The servo is a special kind of motor. Unlike the DC motor used before, the servo helps us to control the angle of rotation of the motor precisely."

I nodded understanding.

Lumi continued, "Here is the code that controls the servo. You can modify the code in red and try to rotate the servo at different angles, but only in the range 0-180."

Click to get Servo_test program.

```
import libs.servo
import time
servo_pin = 25 #Declare the pin of the servo
servo = libs.servo.Servo() #create servo object to control a servo
servo.attach(servo_pin)
servo.write(0) #initialize servo motor

while True:
    for i in range(0, 180, 1):#Servo motor from 0 degrees to 180 degrees
        servo.write(i)
        time.sleep_ms(10)
    for i in range(180, 0, -1):#Servo motor from 180 degrees to 0 degrees
        servo.write(i)
        time.sleep_ms(10)
```

After uploading the program, I found that I could control the servo to rotate at different angles.

Lumi saw this and said, "Congratulations, you can now control the servo flexibly, and we will use the servo to upgrade the obstacle avoidance function."

"How does the servo help the smart car plan its route?" I asked.

"We can do this," Lumi replied. "When the smart car meets an obstacle, it will rotate the angle of the ultrasonic wave through the servo to detect whether there are obstacles on the left and right sides, and choose the safer road to drive."

"To use servo to control ultrasonic sensing, we need to calibrate the servo so that it rotates to 90 degrees," Lumi notes. If the ultrasonic sensor is not facing straight ahead of the smart car, then we need to remove the ultrasonic sensor and reinstall it so that the ultrasonic sensor is facing straight ahead."

Click to get Servo_calibration program.

```
import libs.servo

servo_pin = 25
servo = libs.servo.Servo()
servo.attach(servo_pin)

servo.write(90) #initialize servo motor
```

I pressed the Lumi's instructions to calibrate the servo, and after calibration, the ultrasonic sensor was installed straight ahead.

Lumi continued, "So let's upgrade our smart car to avoid obstacles and go through the Starfall."

Click to get Across_the_starland program.

```
import libs.vehicle
import libs.ultrasonic
import libs.servo
import time

Trig_PIN = 13
Echo_PIN = 14

servo_pin = 25
servo = libs.servo.Servo()
servo.attach(servo_pin)

car = libs.vehicle.ACB_Vehicle()
car.move(car.Stop, 0)


leftDistance = 0
middleDistance = 0
```

```
rightDistance = 0

Ultrasonic = libs.ultrasonic.ACB_Ultrasonic(Trig_PIN,Echo_PIN)

while True:
    middleDistance = Ultrasonic.get_distance()
    servo.write(90)
    if middleDistance <= 25:
        car.move(car.Stop, 0)
        time.sleep(0.01)
        for i in range(90, 0, -1): #Servo motor from 90 degrees to 0 degrees
            servo.write(i)
            time.sleep(0.01)

        time.sleep(0.5)
        rightDistance = Ultrasonic.get_distance()
        for i in range(0, 180, 1): #Servo motor from 0 degrees to 180 degrees
            servo.write(i)
            time.sleep(0.01)

        time.sleep(0.5)
        leftDistance = Ultrasonic.get_distance()
        if rightDistance<20 and leftDistance<20:
            car.move(car.Backward, 180)
            time.sleep(0.5)
            car.move(car.Contrarotate, 180)
            time.sleep(1)

        elif rightDistance > leftDistance:
            car.move(car.Backward, 180)
            time.sleep(0.5)
            car.move(car.Clockwise, 180)
            time.sleep(1)

        elif rightDistance < leftDistance:
            car.move(car.Backward, 180)
            time.sleep(0.5)
            car.move(car.Contrarotate, 180)
            time.sleep(1)

        else:
            car.move(car.Backward, 180)
            time.sleep(0.5)
            car.move(car.Clockwise, 180)
            time.sleep(1)

    else:
        car.move(car.Forward, 150)
```

I constantly debug and modify parameters to ensure that the smart car can accurately sense the surrounding environment and make appropriate obstacle avoidance actions.

After some efforts, the obstacle avoidance function of the smart car has finally been upgraded. Lumi and I got in the smart car again and drove through the meteor environment of Starfall. Although there are many meteorites scattered on the road, the smart car can accurately identify these obstacles, reasonably plan the route, and flexibly bypass them.

After a long drive, we finally crossed Starfall. As we walked out of this mysterious land, Lumi said to me, "Congratulations, you have successfully completed the Starfall mission and received 1 gold, 1 silver and 1 bronze. There are 2 gold, 1 silver and 2 bronze in total. Keep on refueling!"

Scene 5: Crossing the Sky-Reaching Cliff

As we crossed the Starfall, we were greeted with a magnificent sight. A steep mountain, towering into the clouds, stood before us like a huge barrier, blocking our way. The grandeur of the mountain range is awe-inspiring, as if it were an ancient and powerful being silently guarding the land.

At the base of the mountain, there is a winding road winding over the mountain wall, now and then, like a slender snake through the mountain forest. The path spirals up to the top of the hill. In the middle of the road there is a black line, that is the boundary of the road, guiding the direction of the road, so that we do not get lost in the abyss of the mountains.

"Is Lumi's next mission to get over this mountain?"

Lumi replied, "Yes, Star Canyon is just behind this cliff. We need to climb over this cliff to get to Star Canyon."

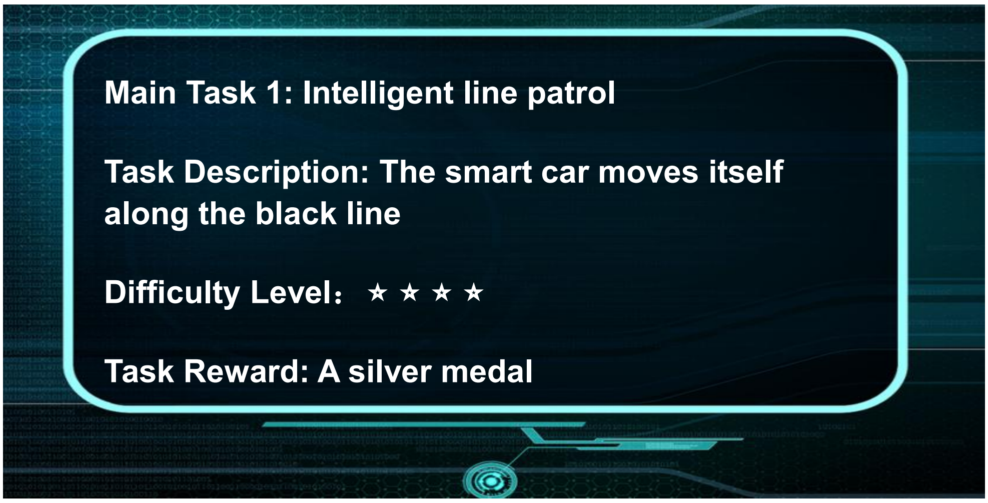
I looked at the steep mountain wall and the curved road, and said to Lumi, "Can the smart car walk on this complicated terrain? If we make a mistake, we will fall off the cliff and be crushed."

"Our smart car can deal with all kinds of complicated situations. Do you see that black line in the middle of the road?" Lumi points to the black line and says, "All we have to do is follow the black line and we will reach the top of the mountain."

I immediately replied, "I remember an amazing technology that allows the car to drive itself along the black line."

Lumi: "Yes, we call it smart line patrol. Through this technology, the car can automatically recognize and drive along the black line, and accurately reach the destination even in the face of various complex routes and environments. Our next task is to make the smart car have the function of automatic line patrol." Lumi then presented the information about the task.

Main Task 1: Intelligent line patrol



Main Task 1: Intelligent line patrol

Task Description: The smart car moves itself along the black line

Difficulty Level：★ ★ ★ ★

Task Reward: A silver medal

I asked excitedly: "Intelligent line patrol technology, very powerful technology. How can our smart car have this function?"

Lumi smiled and replied, "We have a tracking sensor on the car, which can help the car to achieve the function of line patrol."

I then asked, "How do tracking sensor work?"

"The tracking sensor is designed specifically for smart car to enable precise route tracking and navigation. The sensor is composed of three infrared sensors, each of which has a pair of infrared transmitter and receiver tubes, and the transmitter tube emits infrared rays at a specific frequency. Because objects of different colors have different absorption abilities for infrared light, the infrared signal received by the receiver tube is different. We can determine what color object it is on according to the strength of the reflected infrared signal."

Lumi suddenly asked, "Do you know where we can check the values measured by the infrared sensor?"

I immediately replied, "I know, we can use a serial monitor to check the infrared reflection intensity of objects of different colors."

Lumi looked at me with praise and said, "It looks like you have mastered the serial monitor. Can you program it yourself?"

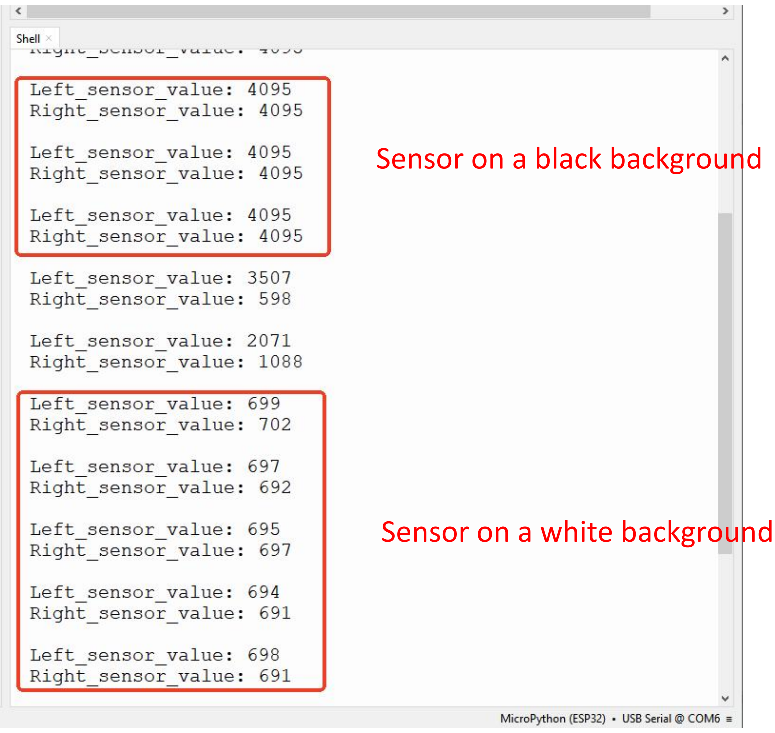
I confidently replied, "No problem."

I opened the MicroPython software and wrote the following code on the software.

Click to get Track_sensor_test program.

```
from machine import ADC
import time

Left_sensor = 35 #Declare the pin of left tracking sensor
Right_sensor = 39 #Declare the pin of right tracking sensor

adc1 = ADC(Left_sensor)
adc1.width(ADC.WIDTH_12BIT)
adc1.atten(ADC.ATTN_11DB)
adc2 = ADC(Right_sensor)
adc2.width(ADC.WIDTH_12BIT)
adc2.atten(ADC.ATTN_11DB)


while True:
    Left_Tra_Value = adc1.read() #Read the value of left tracking sensor
    Right_Tra_Value = adc2.read() #Read the value of right tracking sensor
    print("Left_sensor_value:",Left_Tra_Value)
    print("Right_sensor_value:",Right_Tra_Value)
    print(" ")
    time.sleep(1)
```

After I uploaded the program, I saw the following data when I put the smart car tracking sensor on the black line and the white line.



```
Left_sensor_value: 4095
Right_sensor_value: 4095

Left_sensor_value: 4095
Right_sensor_value: 4095

Left_sensor_value: 4095
Right_sensor_value: 4095

Left_sensor_value: 3507
Right_sensor_value: 598

Left_sensor_value: 2071
Right_sensor_value: 1088

Left_sensor_value: 699
Right_sensor_value: 702

Left_sensor_value: 697
Right_sensor_value: 692

Left_sensor_value: 695
Right_sensor_value: 697

Left_sensor_value: 694
Right_sensor_value: 691

Left_sensor_value: 698
Right_sensor_value: 691
```

Sensor on a black background

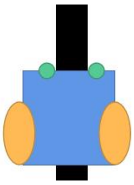
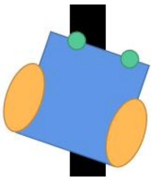
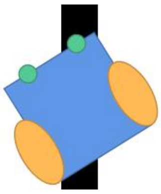
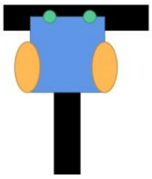
Sensor on a white background

I said excitedly, "Lumi, I get it! When the tracking sensor is on the black line, the sensor feedback data will be greater than 2000. So any time a tracking sensor detects more than 2,000, that sensor is on the black line!"

Lumi nodded and said, "That's exactly right. It's great. The black line absorbs most of the infrared light and reflects some of the light back to the receiver. The weaker the infrared light is at the receiver, the larger the sensor signal is."

"Next, we need to complete a more challenging task - the two-way patrol of smart cars," Lumi continued.

As soon as it spoke, a state diagram of a double-way line inspection program appeared in front of us. The figure depicts in detail the state changes of each sensor when the smart car is driving on the online path. I look at this picture, my heart is full of expectation and fight.

State Diagram	State Description	Sensor Status	State of Motion
	The line is in the middle of the car	Left<2000 Right<2000	Forward
	The line is on the left side of the car	Left>2000 Right<2000	Turn left
	The line is on the right side of the car	Left<2000 Right>2000	Turn right
	The car came to the end	Left>2000 Right>2000	Stop

Lumi continued: "It's not hard to see from the picture that if the black line is in the middle of the smart car and I need to go straight, which side of the sensor is pressing on the black line, the smart car turns in that direction to straighen the car. The following is the car line inspection program, you now try to run it!"

Click to get Tracking_with_two_sensor program.

```
import libs.vehicle
from machine import ADC
import time

Left_sensor = 35 #Declare the pin of left tracking sensor
Right_sensor = 39 #Declare the pin of right tracking sensor
adc1 = ADC(Left_sensor)
adc1.width(ADC.WIDTH_12BIT)
adc1.atten(ADC.ATTN_11DB)
```

```
adc2 = ADC(Right_sensor)
adc2.width(ADC.WIDTH_12BIT)
adc2.atten(ADC.ATTN_11DB)
Black_Line = 2000

car = libs.vehicle.ACB_Vehicle()
car.move(car.Stop, 0)

while True:
    Left_Tra_Value = adc1.read() #Read the value of left tracking sensor
    Right_Tra_Value = adc2.read()
    time.sleep(0.01)

    #Both sensors are on the black line
    if Left_Tra_Value < Black_Line and Right_Tra_Value < Black_Line:
        car.move(car.Forward, 150) #Smart car forward

    #Left sensor is on black line,right sensor is on white background
    elif Left_Tra_Value >= Black_Line and Right_Tra_Value < Black_Line:
        car.move(car.Contrarotate, 150) #Smart car turn left

    #Right sensor is on black line,left sensor is on the white background
    elif Left_Tra_Value < Black_Line and Right_Tra_Value >= Black_Line:
        car.move(car.Clockwise, 150) #Smart car turn right

    #Both sensors are on the white background
    elif Left_Tra_Value >= Black_Line and Right_Tra_Value >= Black_Line:
        car.move(car.Stop, 0) #Smart car stop
```

I couldn't wait to upload the code, and the smart car took us to the mountain road as expected.

Note: The test can be attached with black tape to paste out the walking route.

If the car cannot follow the line properly during the line-tracing process, it may be due to the following reasons:

1.The car's speed is too slow.

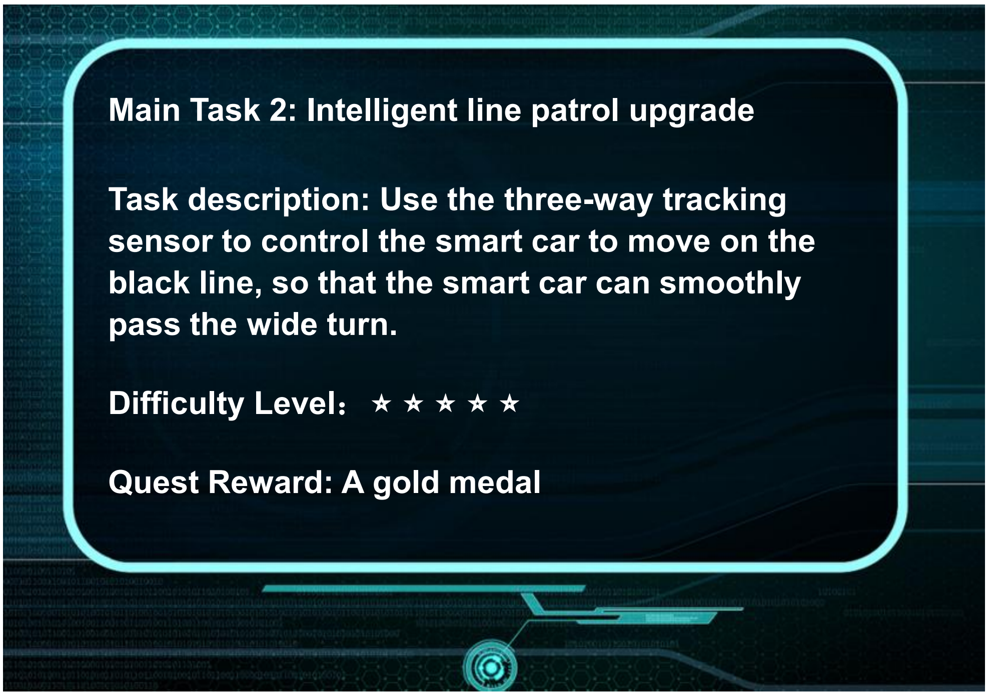
To solve this problem, you can increase the value of the Speed variable in the program to give the car enough speed.

2.Due to interference from ambient light, the ESP32 might mistakenly identify the white area as a black line.

To address this issue, you can first test the specific values of the white area and the black line using the line-tracing sensor, and then adjust the value of the Black_Line variable in the program to more accurately match the actual measured value of the black line.

As the journey went deeper, the mountain road became more rugged, and the road turned more and more sharply. There were several times when our smart car almost went off the road at a turn. Lumi then suggested, "The road is turning more and more sharply, so we need to upgrade the patrol function of our smart car to stay on the road while driving." After that, Lumi showed the task information of intelligent patrol upgrade.

Main Task 2: Intelligent line patrol upgrade



Main Task 2: Intelligent line patrol upgrade

Task description: Use the three-way tracking sensor to control the smart car to move on the black line, so that the smart car can smoothly pass the wide turn.

Difficulty Level: ★ ★ ★ ★ ★

Quest Reward: A gold medal

I quickly asked Lumi, "How do I upgrade?"

Lumi consoled: "Don't worry, let's analyze the reason why the smart car almost went off track!"

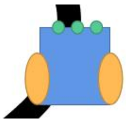
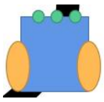
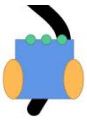
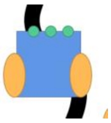
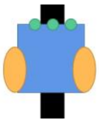
Lumi continued, "The reason why the smart car almost went off the track was because the smart car didn't turn the corner enough when it was on a big turning route. So as long as we get to a big corner and we increase the angle of the turn, we can make a safe turn."

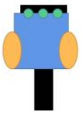
I wondered: "How can the smart car know which turns are big and which are small?"

Lumi replies, "Good question! In order for smart cars to recognize more kinds of road conditions, we need to increase the number of sensors and improve the detection accuracy of smart cars."

I immediately responded, "We have three tracking sensors, but we just used two. Can I use the rest to increase the accuracy of the smart car?"

"Yes," Lumi said. "We're going to use three-way tracking sensors to help us cross the cliff. How does the three-way tracking sensor work?" Lumi then showed the schematic diagram of the line patrol of the three-way tracking sensor.

State Diagram	State Description	Sensor Status	State of Motion
	Both sensors on the far left are on the line	Left>2000 Middle>2000 Right<2000	Forward
	Both sensors on the far right are on the line	Left<2000 Middle>2000 Right>2000	Forward
	The sensor on the far right is on the line.	Left<2000 Middle<2000 Right>2000	Turn right
	The sensor on the far left is on the line.	Left>2000 Middle<2000 Right<2000	Turn left
	The line is in the middle of the smart car	Left<2000 Middle>2000 Right<2000	Forward

	All three sensors are on the line.	Left>2000 Middle>2000 Right>2000	Depends on the actual situation

I watched Lumi show the diagram and tried to program it.

Click to get Tracking_with_three_sensor program.

```
import libs.vehicle
from machine import ADC
import time

Left_sensor = 35 #Declare the pin of left tracking sensor
Middle_sensor = 36 #Declare the pin of middle tracking sensor
Right_sensor = 39 #Declare the pin of right tracking sensor

adc1 = ADC(Left_sensor)
adc1.width(ADC.WIDTH_12BIT)
adc1.atten(ADC.ATTN_11DB)
adc2 = ADC(Right_sensor)
adc2.width(ADC.WIDTH_12BIT)
adc2.atten(ADC.ATTN_11DB)
adc3 = ADC(Middle_sensor)
adc3.width(ADC.WIDTH_12BIT)
adc3.atten(ADC.ATTN_11DB)

Black_Line = 2000
car = libs.vehicle.ACB_Vehicle()
car.move(car.Stop, 0)

while True:
    Left_Tra_Value = adc1.read()
    Right_Tra_Value = adc2.read()
    Middle_Tra_Value = adc3.read()
    time.sleep(0.05)

    if Left_Tra_Value < Black_Line and Middle_Tra_Value >= Black_Line and Right_Tra_Value < Black_Line:
        car.move(car.Forward, 150)

    if Left_Tra_Value < Black_Line and Middle_Tra_Value >= Black_Line and Right_Tra_Value >= Black_Line:
        car.move(car.Forward, 120)
    if Left_Tra_Value >= Black_Line and Middle_Tra_Value >= Black_Line and Right_Tra_Value < Black_Line:
        car.move(car.Forward, 120)

    elif Left_Tra_Value >= Black_Line and Middle_Tra_Value < Black_Line and Right_Tra_Value < Black_Line:
        car.move(car.Contrarotate, 120)

    elif Left_Tra_Value < Black_Line and Middle_Tra_Value < Black_Line and Right_Tra_Value >= Black_Line:
        car.move(car.Clockwise, 120)

    elif Left_Tra_Value >= Black_Line and Middle_Tra_Value >= Black_Line and Right_Tra_Value >= Black_Line:
        car.move(car.Forward, 120)
```

After the carefully written program is successfully uploaded to the smart car, the smart car starts to drive automatically along the black line. It drives steadily and accurately along the black line.

Therefore, we confidently started the journey over the Sky-reaching Cliff. The smart car starts to drive automatically along the black line of Sky-reaching Cliff, and slowly starts to climb Sky-reaching Cliff. It was like a dream journey, as if we were in a magnificent painting.

The scenery on the way is beautiful, the trees are shaded, and the birds are singing. Sometimes there are mountain streams gurgling, and sometimes there is a colorful fall. We immersed in this picturesque scenery, marveling at the magic and beauty of nature. At every corner there is a new landscape to discover, and every cliff tells us the power and charm of nature.

After the successful descent, Lumi immediately announced: "This mission you have 1 gold medal and 1 silver medal, cumulative medals for 3 gold, 2 silver and 2 bronze."

Note:QD001 car may use the line map in the course, the map has two shapes, one is oval, the other is "figure 8". A PDF file of the trail map is available here, and you can print your own if necessary.

【Please click to get the car patrol map file】

Scene 6: Pacify the monster

After successfully crossing the formidable Sky-reaching Cliff, we came to a quiet valley, which seemed to be another world, so quiet that we could hear the sound of our own heartbeat. In the middle of the valley, there is a huge abyss, it is like a bottomless black eye, straight into the ground, unfathomable.

I felt a sense of uneasiness. The sound of faint, heavy breathing from the abyss spoke of some unknown danger. Although the voice is weak, but like a heavy hammer hit my nerve, I can not ignore. Scared, I asked Lumi, "Lumi, do you hear something?"

Lumi reassured me in his steady, calm voice. "Don't be afraid. It's the breath of Seth Monster. The louder its breathing, the sooner it wakes up. The sound of his breathing is still weak, which means he is still sleeping and not in danger."

At this point, Lumi received a message from the institute, and he immediately told me the news: "According to the latest research, music can effectively calm the giant monster. We can play music to keep the giant monster asleep. At the same time, the institute also set this task as the final task of the competition."

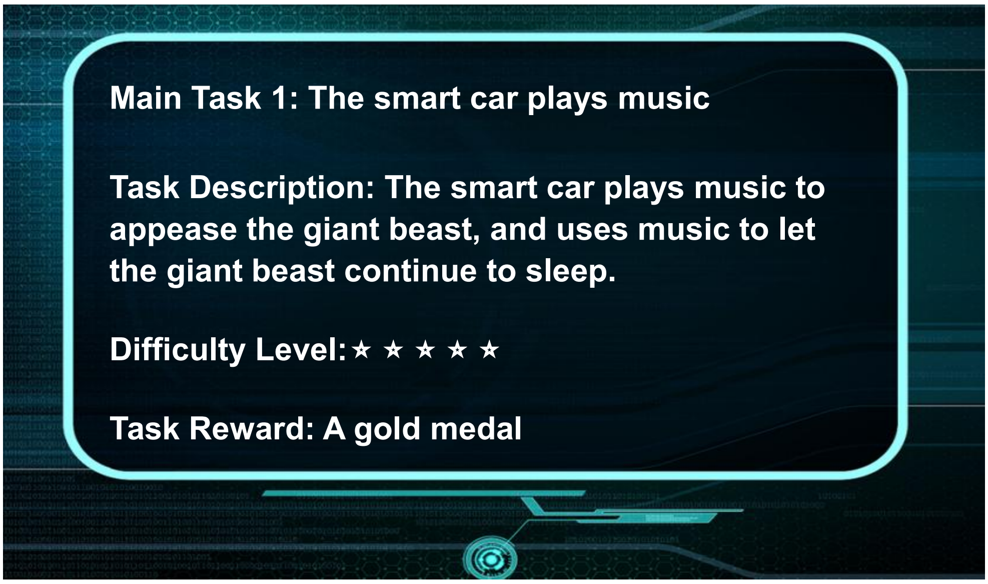
"The final mission?" I asked excitedly. That finish this task I can go to Earth?"

Lumi replied: "The completion of the final mission is only the end of the Universe Intelligence Challenge, after the end of the medal settlement, only the synthetic master medal challengers are considered successful. Come on!"

I calmed down my excitement and asked, "What can we do to make the smart car play music?"

"That's what we're going to do next," Lumi said. "Make the smart car play music." Next, Lumi showed the relevant information about the task.

Main task 1: The smart car plays music



Main Task 1: The smart car plays music

Task Description: The smart car plays music to appease the giant beast, and uses music to let the giant beast continue to sleep.

Difficulty Level:★ ★ ★ ★ ★

Task Reward: A gold medal

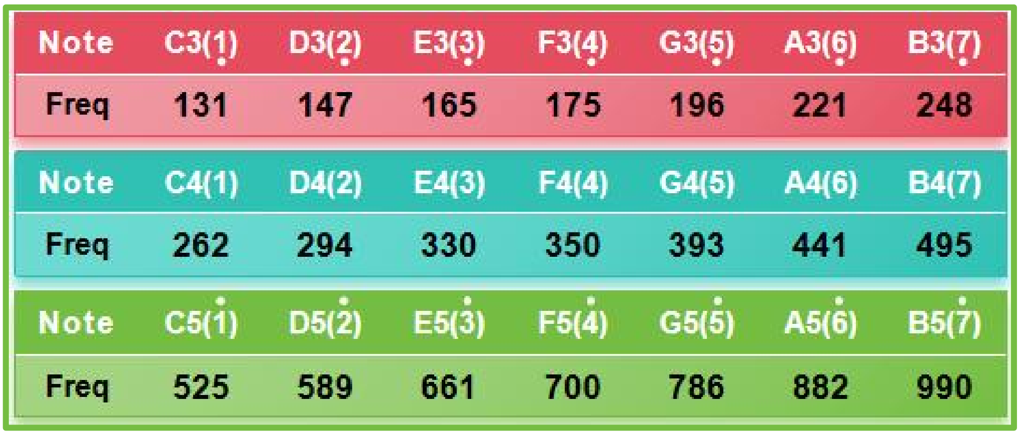
Lumi continued: "The car has a buzzer, and we can play music through the buzzer. In fact, buzzers are divided into passive buzzers and active buzzers. There is a vibration source inside the active buzzer, which can make the vibrator vibrate according to a fixed frequency, so the sound emitted is relatively single and fixed. A passive buzzer, on the other hand, has a vibrator inside, and when the current passes through the vibrator, it creates a magnetic field, and the vibrator vibrates under the action of the magnetic field, which creates a sound, just like when we hit the drum surface, it makes a sound."

I quickly replied, "I see! That means we need a passive buzzer to play music because it can make different sounds."

Lumi complimented, "That's great, you understand that very well! Our smart cars also use passive buzzers."

"How do we make the buzzer play different notes?" I wondered.

"Each note is a sound at a certain frequency," Lumi replied. "We just have to vibrate the buzzer at a certain frequency to make it emit the corresponding note. Here is the frequency of the sound corresponding to each note." Lumi then displays a graph of the frequencies of the notes.

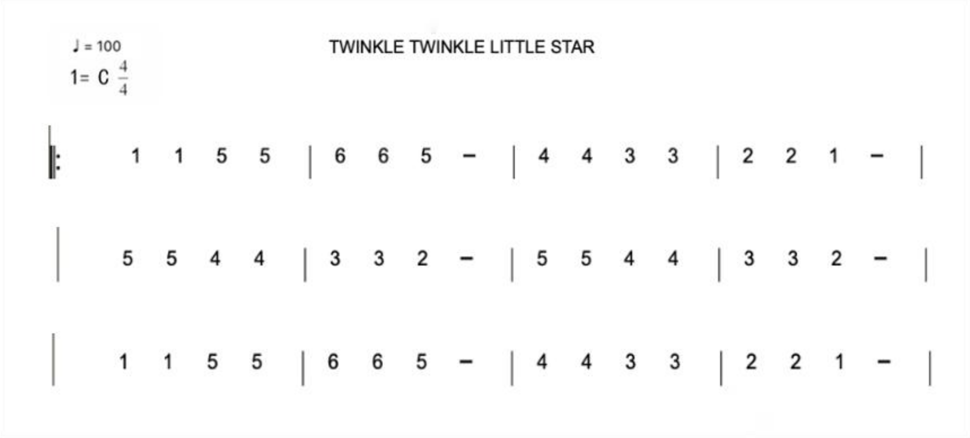


Note	C3($\underset{\cdot}{1}$)	D3($\underset{\cdot}{2}$)	E3($\underset{\cdot}{3}$)	F3($\underset{\cdot}{4}$)	G3($\underset{\cdot}{5}$)	A3($\underset{\cdot}{6}$)	B3($\underset{\cdot}{7}$)
Freq	131	147	165	175	196	221	248
Note	C4(1)	D4(2)	E4(3)	F4(4)	G4(5)	A4(6)	B4(7)
Freq	262	294	330	350	393	441	495
Note	C5($\dot{1}$)	D5($\dot{2}$)	E5($\dot{3}$)	F5($\dot{4}$)	G5($\dot{5}$)	A5($\dot{6}$)	B5($\dot{7}$)
Freq	525	589	661	700	786	882	990

I replied, "With frequency maps, we can play music!"

Lumi replied, "Don't worry, music doesn't just have notes, it has rhythms. Write it down so we know in detail what a piece of music is made of."

Lumi then showed the musical notation of Little Star and introduced the meaning of the notes and beats in it.



1. Common notes, such as the first note 1, account for one beat.

2. An underlined note indicates 0.5 beat.

3. Some notes are followed by a dot to indicate an additional 0.5 beat, that is, a note is 1+0.5 = 1.5 beats.

4. Some notes are followed by -, indicating that one more beat is added, that is, a note is 1+1 = 2 beats.

Symbol	Beat
1	1 beat
1	0.5 beat
1.	1.5 beat
1-	2 beat

"The second thing you need to know is the beat," Lumi continued. "The beat is the time between notes, and each song has a different delay time . We can see♪=100，on the notation, which means 100 beats per minute, so we can know that the duration of a beat is 60000/100=600ms."

I nodded to indicate understanding.

Lumi then showed off a program for Little Star and said, "Here's a program you can try to make your smart car play Little Star ."

Click to get Buzzer_music program.

```
import time
from machine import Pin, PWM

buzzer = 33
```

```
# Define a list of tone frequencies
C4 = 262
D4 = 294
E4 = 330
F4 = 350
G4 = 393
A4 = 441
B4 = 495
N = 0

# little star
tune0 = [C4, N, C4, G4, N, G4, A4, N, A4, G4, N, F4, N, F4, E4, N, E4, D4, N, D4, C4]
durt0 = [0.99, 0.01, 1, 0.99, 0.01, 1, 0.99, 0.01, 1, 1.95, 0.05, 0.99, 0.01, 1, 0.99, 0.01, 1, 0.99, 
0.01, 1, 2]

buzzer_pin = PWM(Pin(buzzer), freq=1000, duty=0) #Buzzer initialization

# Define the function to play the music
def play_music(tune, durt, delay_time):
    for i in range(len(tune)):
        if tune[i] is not None:
            freq = int(tune[i])
            if freq < 1:
                freq = 1
            elif freq > 40000000:
                freq = 40000000
            buzzer_pin.freq(freq)
            buzzer_pin.duty_u16(32767)    # 50% duty cycle
        time.sleep(delay_time * durt[i])
        buzzer_pin.duty_u16(0)    # Turn off the buzzer
        time.sleep(delay_time * durt[i] * 0.1)    # Add a small delay between notes

play_music(tune0, durt0, 0.5)
```

After the program was uploaded to the smart car, it successfully played the music of Little Star. I felt very excited and proud to hear the beautiful melody coming from the smart car.

I asked Lumi excitedly, "How much is my medal now?"

Lumi replied, "With one gold medal from this mission, we now have 4 gold, 2 silver and 2 bronze medals. We are not far away from victory. Let's keep going."

Scene 7: To the Abyss

We succeeded in making the smart car play music. However, due to the distance, the giant beast does not seem to hear the beautiful melody. The sound of its heavy breathing reverberated through the air, undiminished.

Lumi and I decided, in order to make this behemath feel our efforts, to bring the smart car closer to the Seth Monster abyss, a little closer.

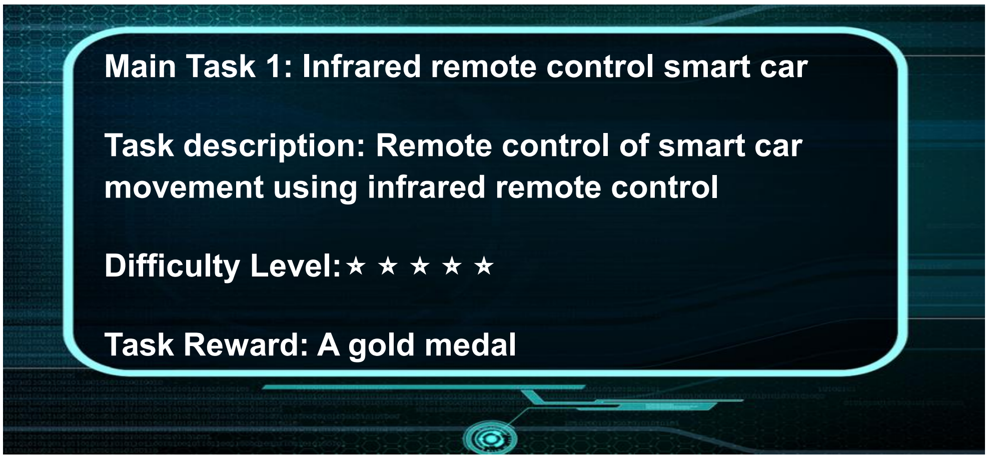
"The abyss is miasmatic and dangerous," Lumi reminds me. "We were too dangerous." Facing the temptation of imminent success, I asked impatiently: "What should I do?" Lumi calmly replied, "Don't worry, we can't go to the abyss in person, but we can control the smart car remotely."

Curious, I pressed, "Remote control? How do we do that?"

"We can use infrared remote control," Lumi patiently explains. Infrared is a kind of electromagnetic wave with a longer wavelength and better penetration, which can pass through miasma and transmit signals to smart cars."

"Our next task is to use infrared to remotely control smart cars," Lumi said. Lumi then presented the information about the task.

Main Task 1: Infrared remote control smart car



Main Task 1: Infrared remote control smart car

Task description: Remote control of smart car movement using infrared remote control

Difficulty Level:★ ★ ★ ★ ★

Task Reward: A gold medal

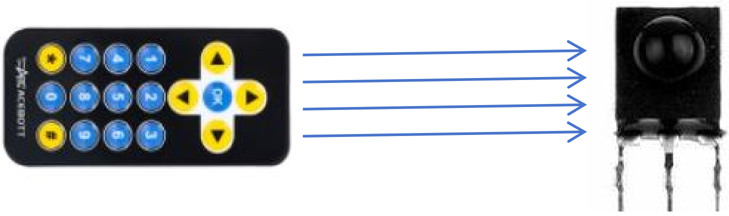
"So," I asked, "how do we achieve infrared remote control?"

Lumi replied, "To achieve infrared remote control, you need two devices: one is an infrared transmitter and the other is an infrared receiver. The transmitter is used to send infrared signals, and the receiver is used to receive signals and pass them to the smart car. The infrared remote control is a transmitter, and when we press the button on the remote control, the remote control emits infrared light."

"There are so many buttons on the remote control and only one receiver," I wondered. "How does it know which button you pressed?"

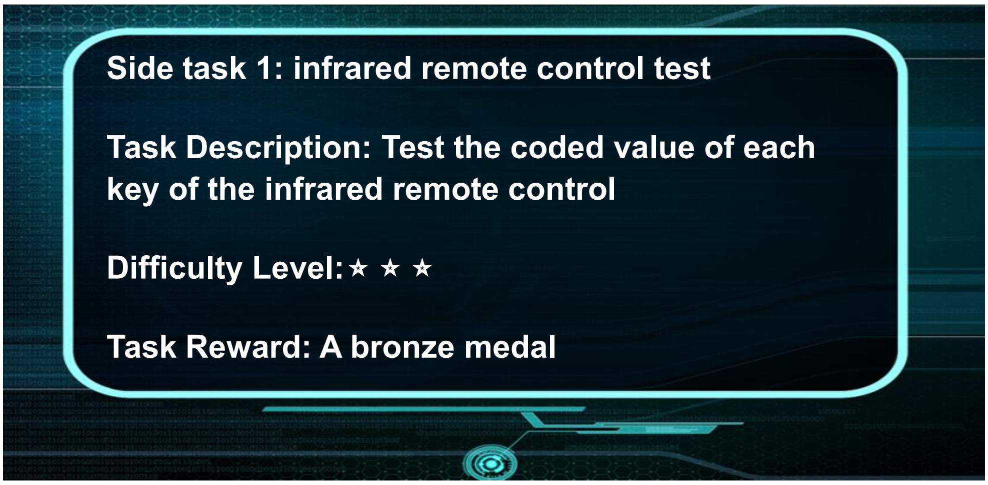
"Good question!" praises Lumi. The infrared receiver can detect which button is pressed on the remote control because there is some kind of communication protocol between them, and each button corresponds to a different infrared code, and the receiver knows which button is pressed when it receives that code." Lumi then showed the code corresponding to each key on the infrared remote control, which is in the form of value.

Button	Value	Button	Value	Button	Value
▲	Up	▼	Down	OK	OK
◀	Left	▶	Right	1	1
2	2	3	3	4	4
5	5	6	6	7	7
8	8	9	9	0	0
*	*	#	#		



"The next side task was to test whether the encoding of each key corresponded to the type of table," Lumi said. After that, Lumi showed off the side task information.

Side Task 1: Infrared remote control test



Side task 1: infrared remote control test

Task Description: Test the coded value of each key of the infrared remote control

Difficulty Level:★ ★ ★

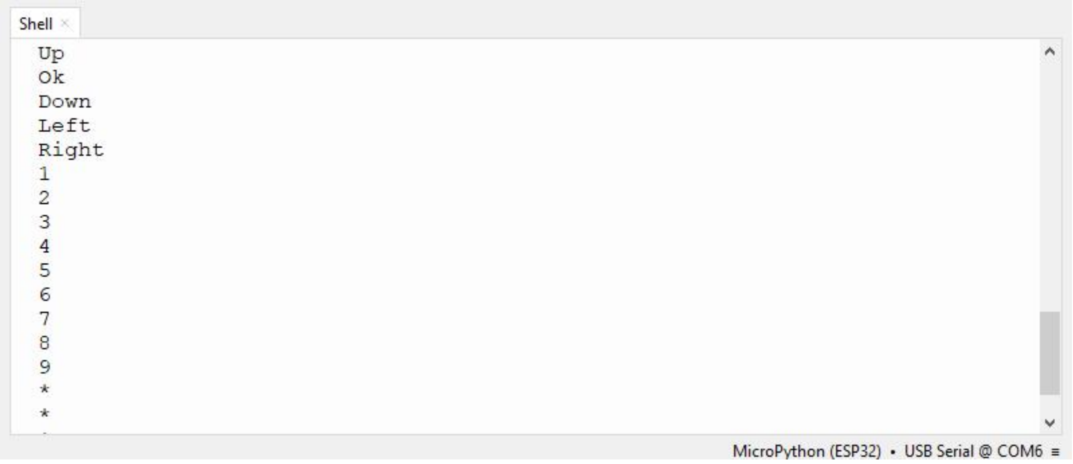
Task Reward: A bronze medal

Lumi continued, "We can use a serial monitor to see the encoding value of each key. It should be noted that in our infrared remote control to the infrared transmitter to send infrared signal, infrared receiver can receive the corresponding signal, otherwise the infrared receiver will not be able to receive the infrared command sent by the remote control.

Click to get IRremote_test program.

```
import libs.ir #Import infrared remote control library
import time

IR_PIN = 4;
irm = libs.ir.ACB_IR(IR_PIN)

while True:
        ir_cmd = irm.read_IR() #Get the value of the infrared remote control
        if ir_cmd != None:
            print(ir_cmd)
        time.sleep(0.1)
```

After uploading the program: I press any key, the serial port will display the corresponding coding instructions.



```
Up
Ok
Down
Left
Right
1
2
3
4
5
6
7
8
9
*
*
```

"Congratulations," Lumi said. "You're more closer to accomplishment.You've now decoded the transmitter code. Next, we need to control the movement of our car according to different key commands. How many movements does the smart car need to control the car into the abyss?"

I thought for a moment and replied, "It requires seven different movements: forward, backward, turn left, turn right, translate left, translate right, and stop."

Lumi then asked, "The first six movements can be controlled by a button on the remote, but should you also use a button to control the stop?"

I shook my head and replied, "No, we can't stop the smart car by pressing a button. This is because if we don't react in time and press the stop button in time, the smart car may not stop in time and cause an unexpected situation." Thinking about the remote control car in our life, he added, "We should make the smart car stop automatically without receiving any signal. Even if we forget to press the stop button, the smart car will stop automatically to avoid accidents."

"You're right," says Lumi. We're going to program the car to do that."

Click to get IRremote_car program.

```
import libs.vehicle
from machine import Pin, PWM, Timer, ADC
import libs.ir
import time

IR_PIN = 4;
irm = libs.ir.ACB_IR(IR_PIN)
```

```
speeds = 220

car = libs.vehicle.ACB_Vehicle()
car.move(car.Stop, 0)

while True:
    ir_cmd = irm.read_IR()
    if ir_cmd != None:
        print(ir_cmd)
    if ir_cmd=="Up": #Press "up" button to move forward
        car.move(car.Forward, speeds)
    elif ir_cmd=="Down":#Press "down" button to move backward
        car.move(car.Backward, speeds)
    elif ir_cmd=="Left": #Press "left" button to turn left
        car.move(car.Contrarotate, speeds)
    elif ir_cmd=="Right": #Press "right" button to turn right
        car.move(car.Clockwise, speeds)
    elif ir_cmd=="1": #Press button "1" to move left
        car.move(car.Move_Left, speeds)
    elif ir_cmd=="3": #Press button "3" to move right
        car.move(car.Move_Right, speeds)
    else:
        car.move(car.Stop, 0)
time.sleep(0.2)
```

I can't wait to upload the code to realize the remote control of the smart car. However, when the smart car went a certain distance, I found that I lost control of it and could no longer move it forward. So I turned to Lumi and said, "What's wrong, Lumi, why can't we control the car?"

"The road to the abyss is bumpy and there are many obstacles," explains Lumi. The infrared ray propagation can only be carried out in a straight line without penetration. Once there is an obstacle between the remote control and the receiver, the infrared remote control cannot be carried out. And the effective distance of infrared communication is relatively close, when the car can only run to a certain distance, we can not continue to control it?"

I anxiously asked: "How to do that, there is a distance to come, is there a remote control technology for a longer range of control?"

Lumi replied, "Of course, we can also use WiFi to control our smart car."

"Remote control with WiFi, sounds great! But what are we to do?" "You ask excitedly.

"The ESP32 controller board of the smart car comes with WiFi," Lumi explained. It can receive external WiFi signal, this mode is called Sta mode; At the same time, it can also send WiFi signals to the outside, this function is called AP mode. The ESP32 controller uses these two ways to communicate with external devices that also have WiFi capabilities."

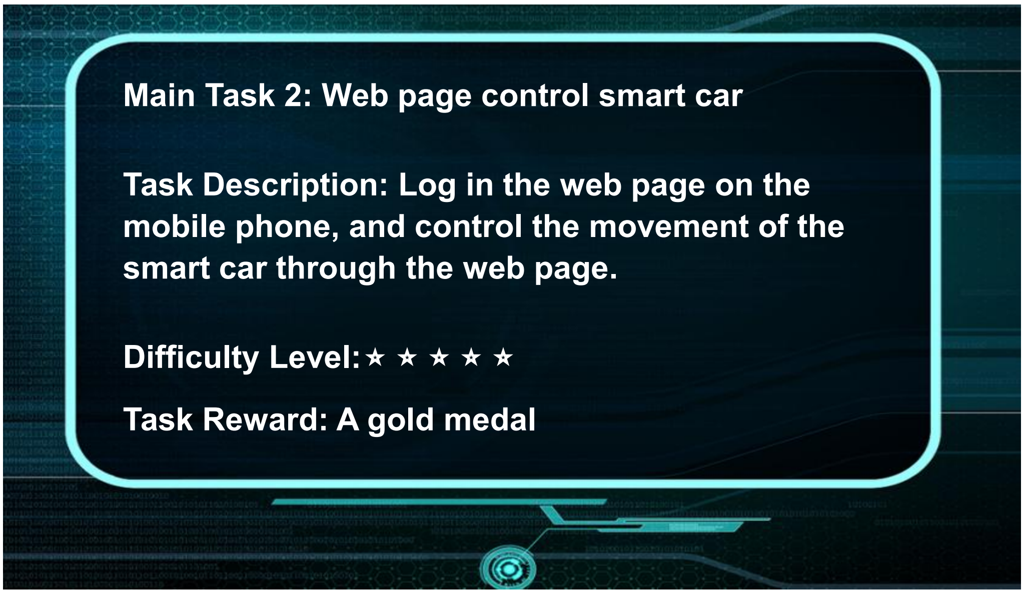
"How do we communicate here?" I asked.

Lumi then replied: "We are in the middle of nowhere and there is no other network signal, so we can use the AP mode of the WiFi module and let the ESP32 generate a WiFi hotspot, and then we can connect the hotspot with the phone, so that the phone can communicate with the smart car."

I immediately asked, "How can we use the phone to send control commands to the smart car once the phone is connected to the hotspot?"

Lumi patiently replied, "Here, we can send instructions to control the smart car through the web and the APP. First, let's see how we can control a smart car from a web page." After that, Lumi showed the web page to control the task information of the smart car.

Main Task 2: Web page control smart car



Main Task 2: Web page control smart car

Task Description: Log in the web page on the mobile phone, and control the movement of the smart car through the web page.

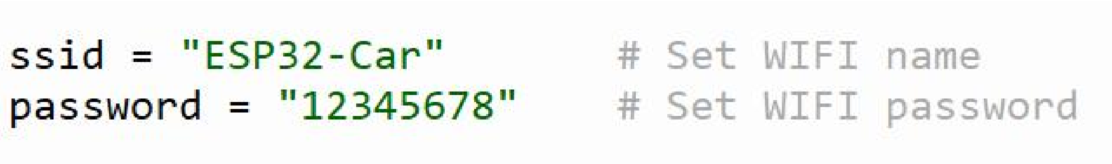
Difficulty Level:★ ★ ★ ★ ★

Task Reward: A gold medal

Lumi continued, "When we use a web page to control the smart car, we need to set up a web server and create a web interface to send control instructions to the smart car. This is a program for the web to control the smart car."

Click to get Web_control_car program.

After I uploaded the program, Lumi reminded me, "First, we need to connect our phone to the WIFI sent by our smart car. The WIFI name and password are already defined in the program."



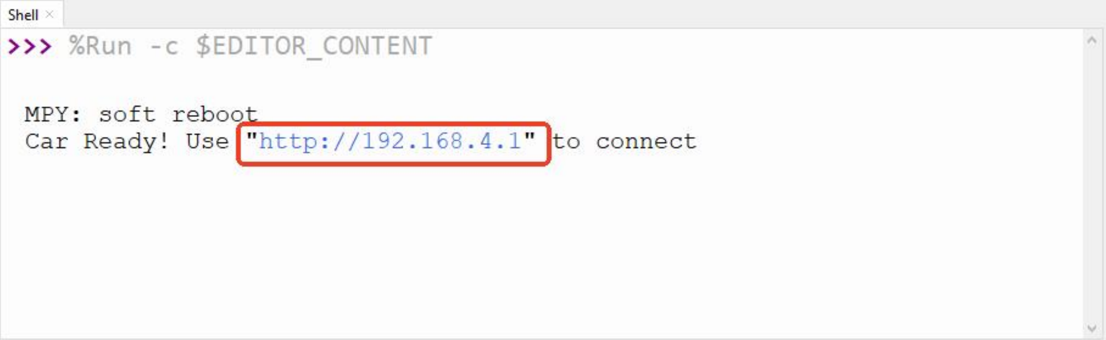
```
ssid = "ESP32-Car"           # Set WIFI name
password = "12345678"        # Set WIFI password
```

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

Lumi continues: "Once the phone is connected to the WIFI, we need to open the browser and enter the IP address of the web server in the URL bar to access the web page."

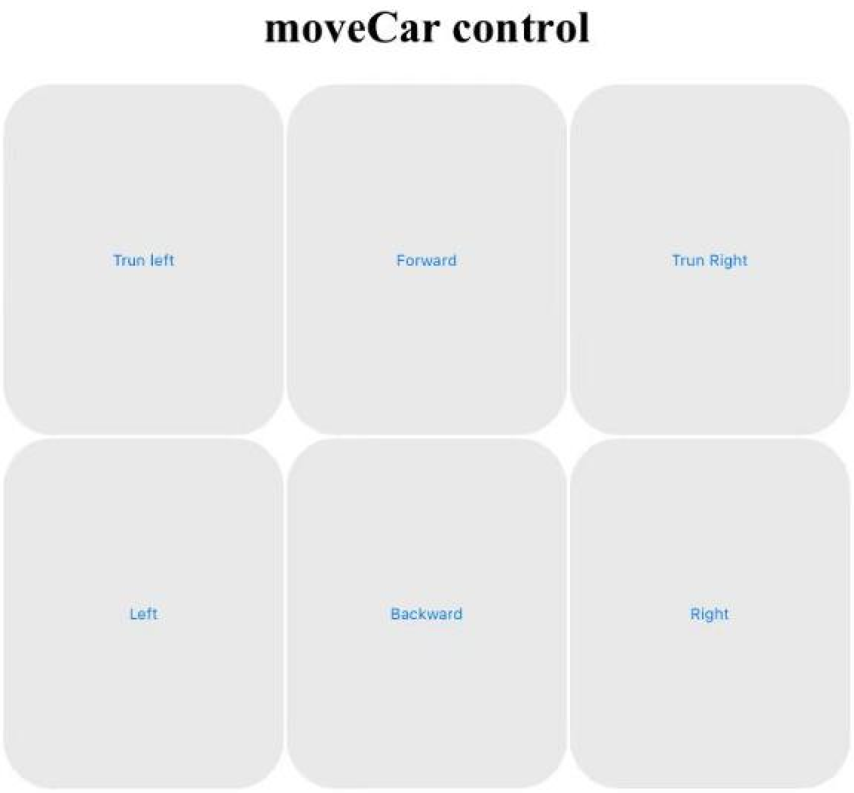
"Where can we check the IP address of the smart car's web server?" I asked.

Lumi replied, "We can look at the serial monitor, turn on the serial monitor and press the reset button on the ESP32 controller board, and we can see the IP address."



```
>>> %Run -c $EDITOR_CONTENT

 MPY: soft reboot
 Car Ready! Use "http://192.168.4.1" to connect
```

Following Lumi's instructions, I connected my phone to WIFI and opened a browser. I entered 192.168.4.1 in the browser's web search bar, and then the browser entered the following web page.



moveCar control

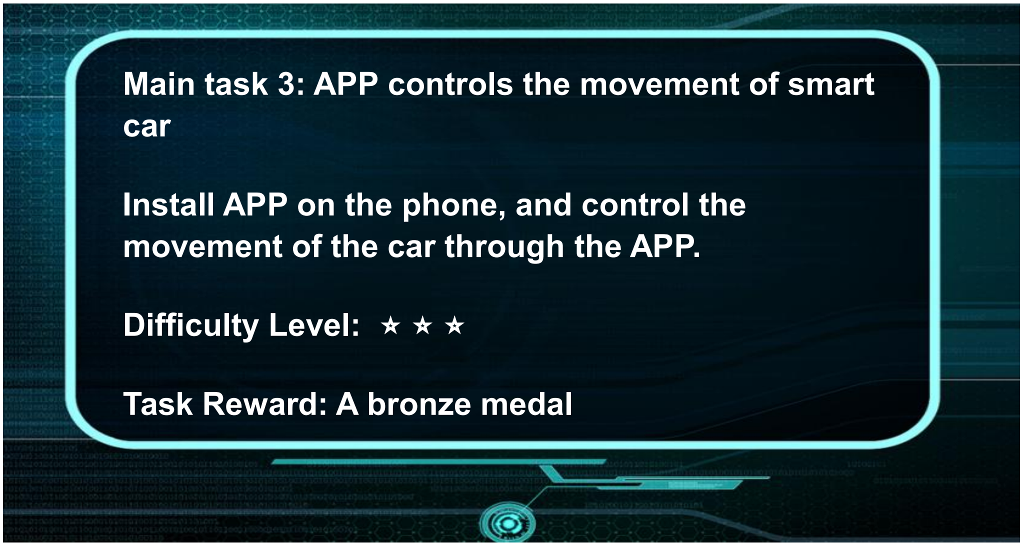
I clicked on different square instructions on the page and found that the smart car moved according to the corresponding instructions.

Lumi is pleased to see that, "Very good, you have mastered the web to control the movement of the smart car."

I was excited and said, "That's fun. How does the APP control the smart car? I can't wait to use the APP to control the smart car."

"That's what we're going to do next," Lumi says with a smile. I have prepared an APP for you, you just need to install it on your phone, you can control your smart car more easily." Then Lumi showed the APP to control the smart car and ground task information.

Main Task 3: APP controls the movement of smart car



Main task 3: APP controls the movement of smart car

Install APP on the phone, and control the movement of the car through the APP.

Difficulty Level: ★ ★ ★

Task Reward: A bronze medal

I felt very happy because Lumi's APP sounded very interesting. So I asked, "So, how do I install the APP?"

"If it's an IOS phone, search the APP Store for ACEBOTT and download it," Lumi says.

"How do I download it for Android?"

"If you have an Android phone, you need to search the Google Play Store for ACEBOTT and download it."

I followed this method to download and install the program.

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.

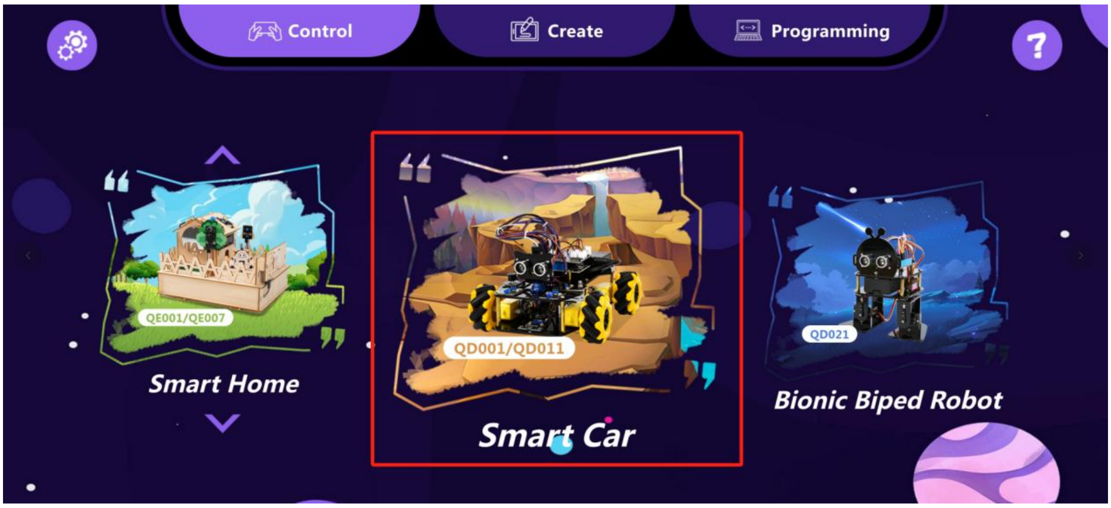
"ACEBOTT app is installed, can you directly control the smart car?"

Lumi shook his head and said, "Not yet. Just like the web page to control the smart car, the APP to control the smart car also requires the smart car to send WIFI, and the phone to connect to the WIFI of the smart car. So we also need to download the corresponding control program to the smart car, so that we can use the APP to control the smart car." Lumi shook his head and said, "Not yet. Just like the web page to control the smart car, the APP to control the smart car also requires the smart car to send the hot spot, and the phone to connect the hot spot of the smart car. So we also need to download the corresponding control program to the smart car, so that we can use the APP to control the smart car."

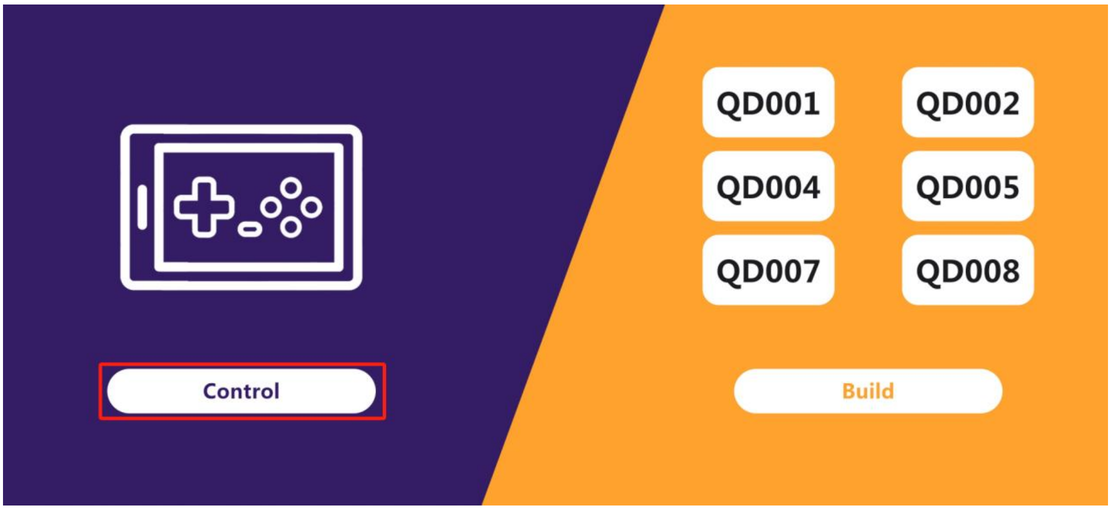
Click to get APP_Control_Car program.

After uploads the APP, Lumi continues, "You can now open our app. When you open it, you will see this screen."

"You will then be brought to the product selection screen and select the 'Smart Car' option."

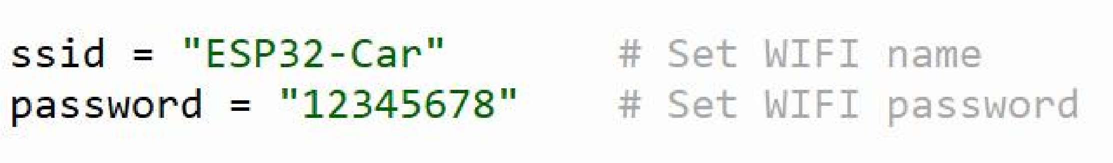


After clicking to enter, select "Control" to access the car control page.



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

Mobile phone wireless network scan WIFI , connect to the name of "ESP32-Car" wifi, the password is 12345678, as shown below.



```
ssid = "ESP32-Car"           # Set WIFI name
password = "12345678"        # Set WIFI password
```

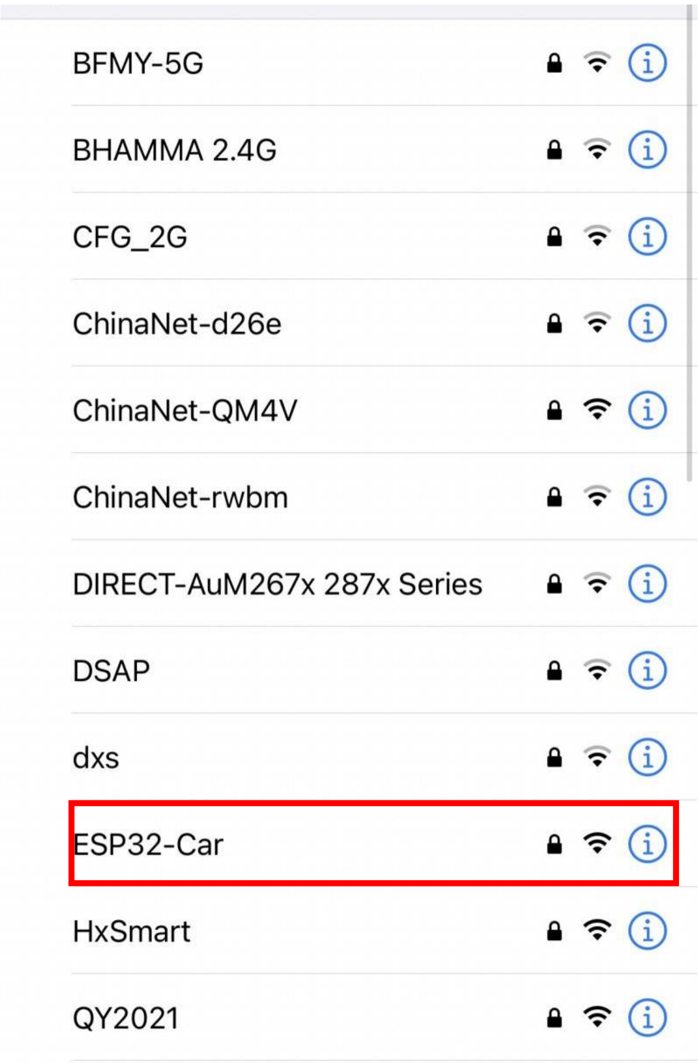
Note:

1. When controlling the smart car with a mobile app, please ensure that the distance between the phone and the smart car is within 5 meters, as the WiFi signal can be affected by distance. If it is too far, it may cause the WiFi disconnection of the car.

2. Try to avoid using the app to control the smart car in areas with strong WiFi signals to prevent interference with the car's WiFi signal, which could lead to WiFi connection failure.

3. When using the app to control the smart car at home, it is recommended to "forget" the connected WiFi on your phone to reduce the frequency of the phone automatically disconnecting from the car's WiFi.

4. If you use the APP to control the smart car at home, try to set the connected WiFi on the phone "forget to connect" to reduce the frequency of the phone automatically disconnecting the car WiFi.



BFMY-5G

BHAMMA 2.4G

CFG_2G

ChinaNet-d26e

ChinaNet-QM4V

ChinaNet-rwbm

DIRECT-AuM267x 287x Series

DSAP

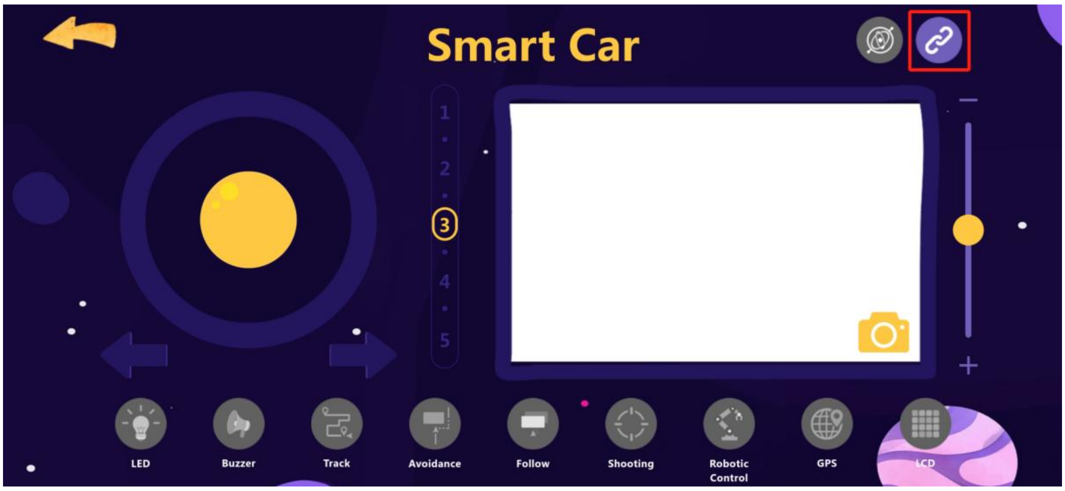
dxs

ESP32-Car

HxSmart

QY2021

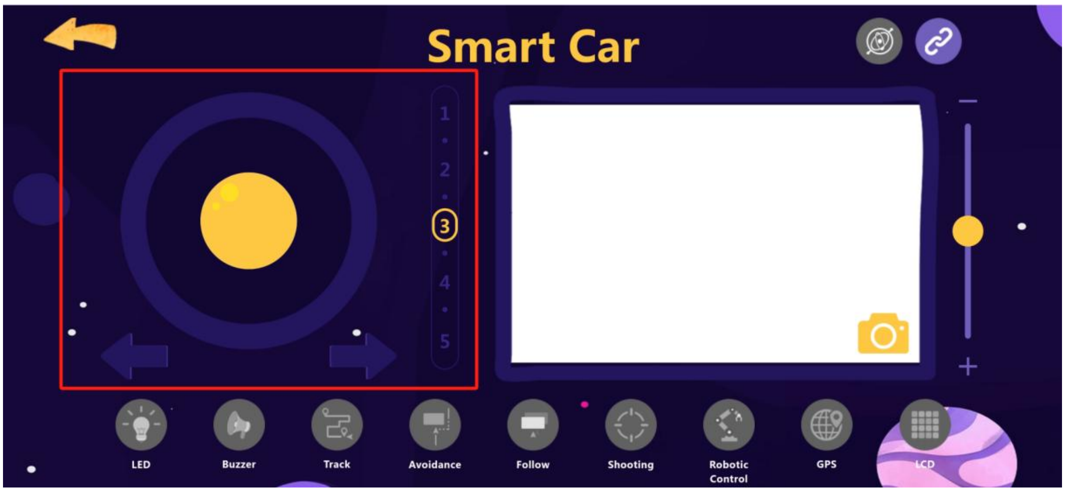
After connecting the WiFi, click the connection icon in the upper right corner of the APP to complete the connection.



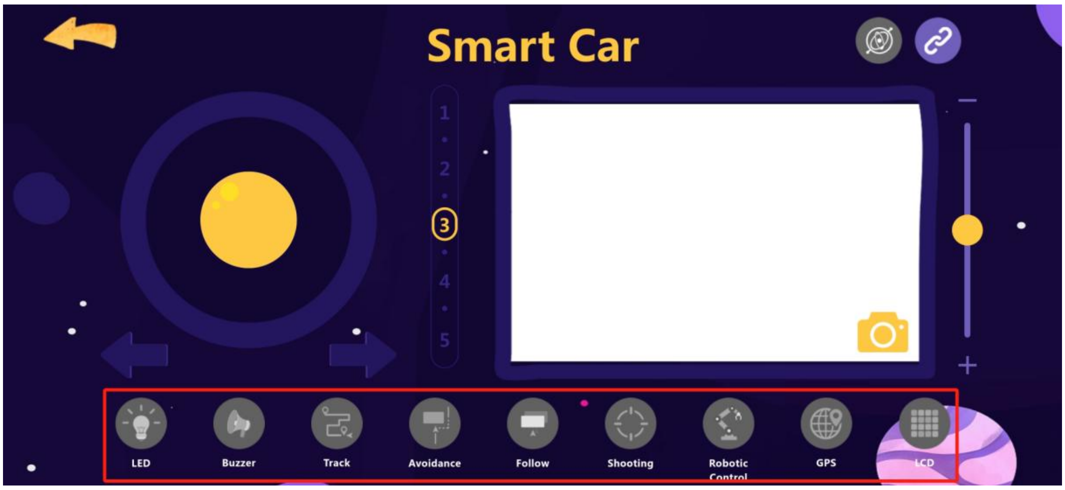
After successful connection, you can control the motion of the car through the APP. The specific operation is as follows.

The 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.



Below 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.



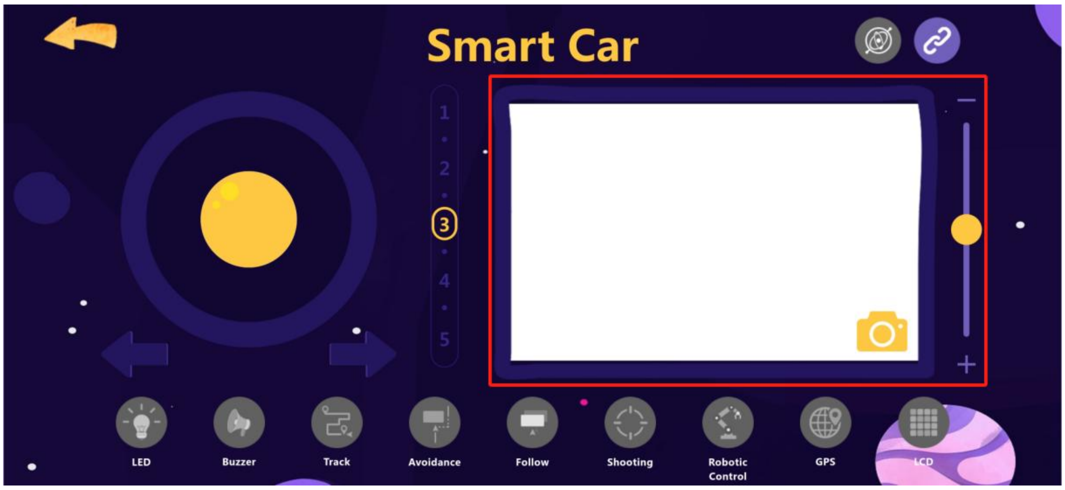
Note:

1. After clicking on the track mode, avoidance mode, or Follow mode, the

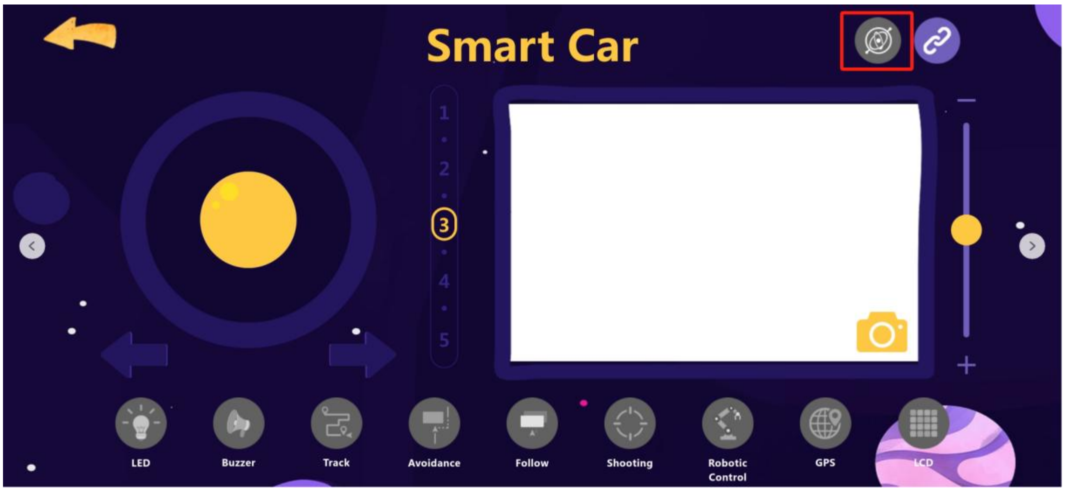
car will enter a continuous tracking, avoidance, or follow state, respectively. To exit this function, you need to click the corresponding function button again to exit the current state. For example, to turn off track mode, you need to click the track button again to exit the track state.

2. Shooting, robotic control, GPS, and LCD are extended functions of the car, which require the corresponding expansion packs to be added for use.

The central control panel is for the car's camera function, and the slider on the right is for servo control. Both of these features are extended functions of the car and require the corresponding expansion packs to be used.

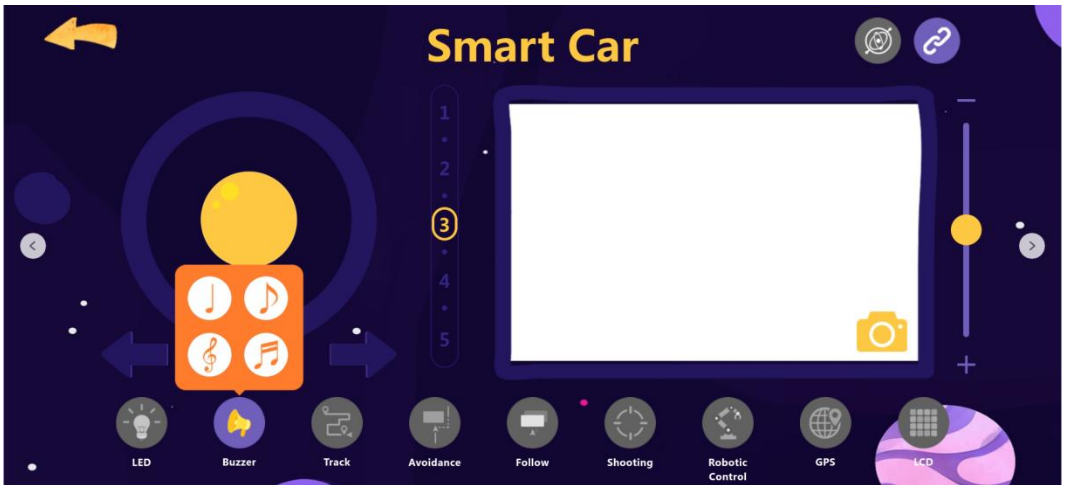


In the upper right corner of the smart car's operation interface, there is a gyroscope control provided. After clicking this button, you can control the movement of the car through the gyroscope of your mobile phone. If your phone does not have a built-in gyroscope, you can ignore this feature.



Following Lumi's instructions, I carefully operated the smart car to approach the abyss once again. I watched it nervously and excitedly, while silently praying in my heart that it could smoothly reach the vicinity of the Sis monster.

Fortunately, the smart car reached the destination very smoothly and started playing a pleasant music when I pressed the play music button. The music seemed to have an effect on Seth. His breathing slowly weakened and he slowly fell into a deep sleep. Seeing this scene, I cried out with excitement: "Lumi, I have successfully completed the task."



Lumi also congratulated: "Congratulations, you successfully completed the challenge, received this unique master medal. From now on, you will be one of our future researchers, and you can go back to your planet to create a better future!"

End

In the vast universe, I embarked on the final journey, completed the task of the competition, and was honored to become a member of the Future Institute. This experience gave me a deep understanding of the power of technology and made me full of expectations for the future.

During the course of the competition, I experienced many challenges and difficulties, but these experiences made me more determined to love technology. I kept exploring, trying, failing, trying again, and each step brought me closer to my dream. In this process, I not only learned how to face challenges, but also how to cooperate with others to solve problems together. This experience has made me grow a lot and make me cherish my harvest more.

After returning to Earth, I deeply felt the importance of technology. Technology has not only changed the way we live, but also allowed us to understand the universe more deeply. I have developed a strong interest in intelligent technology, and hope to contribute to the development of mankind through my own efforts. So, I went back to Planet X and started my smart technology research journey.

I will continue to work hard, continue to move forward, let their dreams illuminate the whole universe.

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