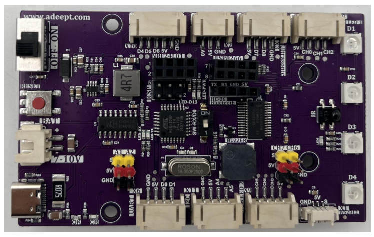
Lesson 12 Introduction to the Light-Tracking Module

12.1 Overview

This course mainly introduces the working principle and application scenarios of the light tracking module, as well as how to combine it with the Adeept Robot Control Board to achieve light tracking function through Arduino programming. Learners will master the complete process from hardware connection to code writing and debugging, and understand the application of optical tracking technology in smart device control.

12.2 Required Components

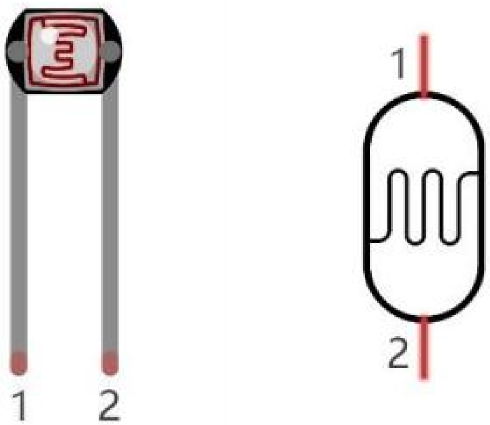
Components	Quantity	Picture
Adeept Robot Control Board	1	
Type-C USB Cable	1	
Light Tracking Module	1	

12.3 Principle Introduction

Light Tracking Module is composed of two photoresistors. The photoresistor is very sensitive to the amount of light present. We can use this feature to make a light tracing car. The car is controlled to turn toward the light source by reading the ADC values of the two photoresistors at the head of the car.

ADC is an electronic integrated circuit used to convert analog signals such as voltages to digital or binary form consisting of 1s and 0s. The range of our ADC on Arduino Uno is 10 bits, that means the resolution is 2^10=1024, and it represents a range (at 5V) will be divided equally to 1024 parts. The rage of analog values corresponds to ADC values. So the more bits the ADC has, the denser the partition of analog will be and the greater the precision of the resulting conversion.

A photoresistor is simply a light sensitive resistor. It is an active component that decreases resistance with respect to receiving luminosity (light) on the component's light sensitive surface. A photoresistor’ s resistance value will change in proportion to the ambient light detected. With this characteristic, we can use a photoresistor to detect light intensity. The photoresistor and its electronic symbol are as follows.



When a photoresistor’ s resistance vale changes due to a change in light intensity, the voltage between the photoresistor and resistor R1 will also change. Therefore, the intensity of the light can be obtained by measuring this voltage.

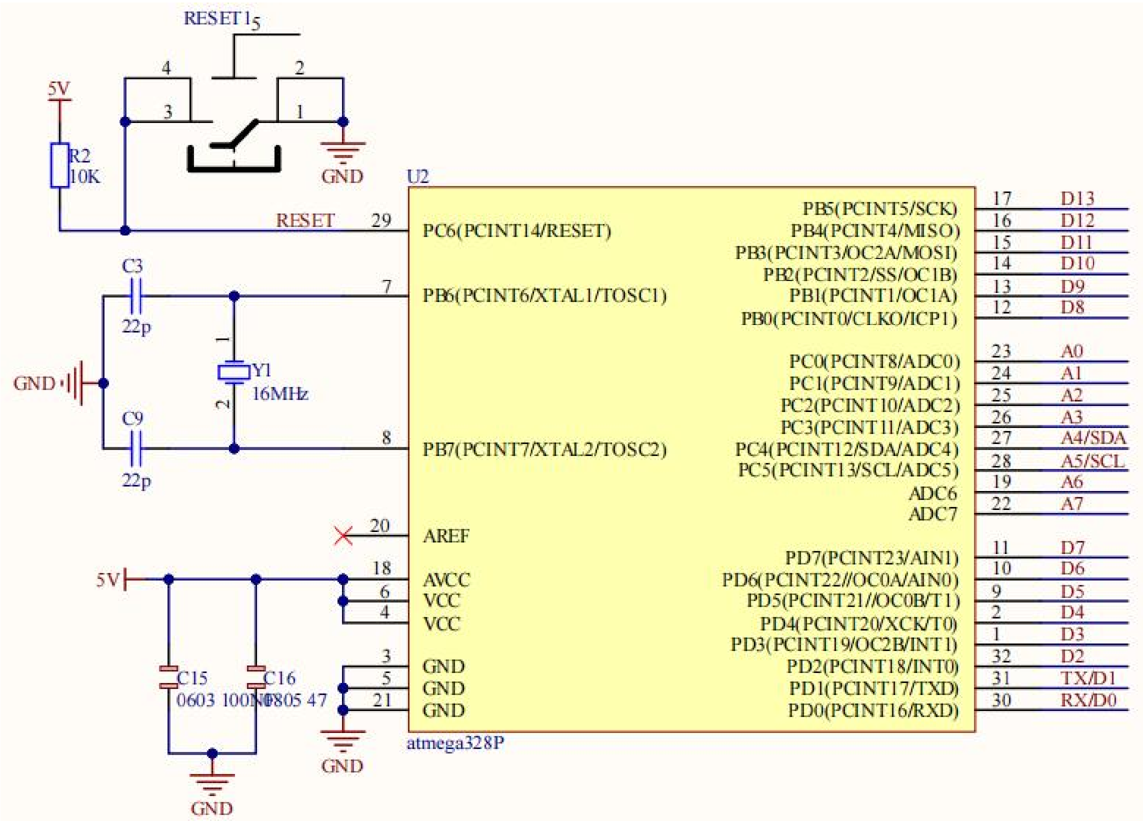
When the brightness of the light received by the photoresistor R1 and R2 is the same, the voltage at SIG is 5/2, which is 2.5V. Therefore, when the brightness of the light received by the two photoresistors is different, the voltage at SIG will be greater or less than 2.5V, and the range is 0-5V. The ADC value range corresponding to the SIG voltage range is 0-1024.

Due to the problem of resistance accuracy, when the module is under the same light intensity, the detected value may deviate, but it does not affect the basic functions of the module.

PINS of Adeept Robot Control Board	Arduino(X2)

SIG	A0
VCC	5V
GND	GND

12.4 Wiring Diagram



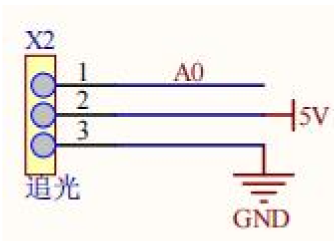
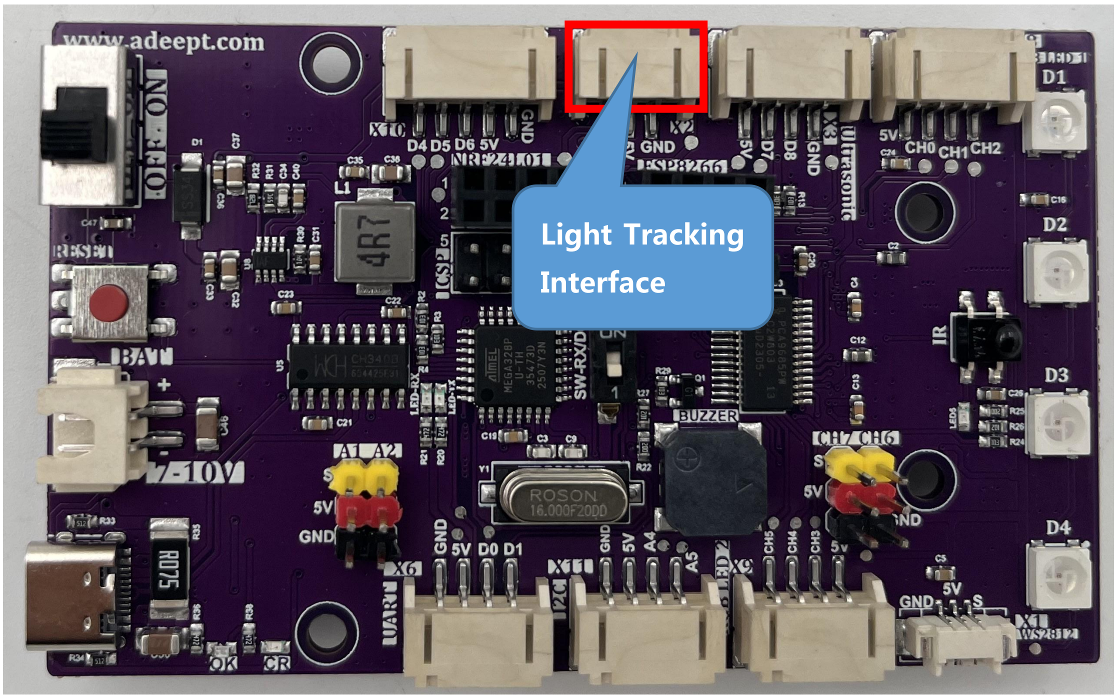
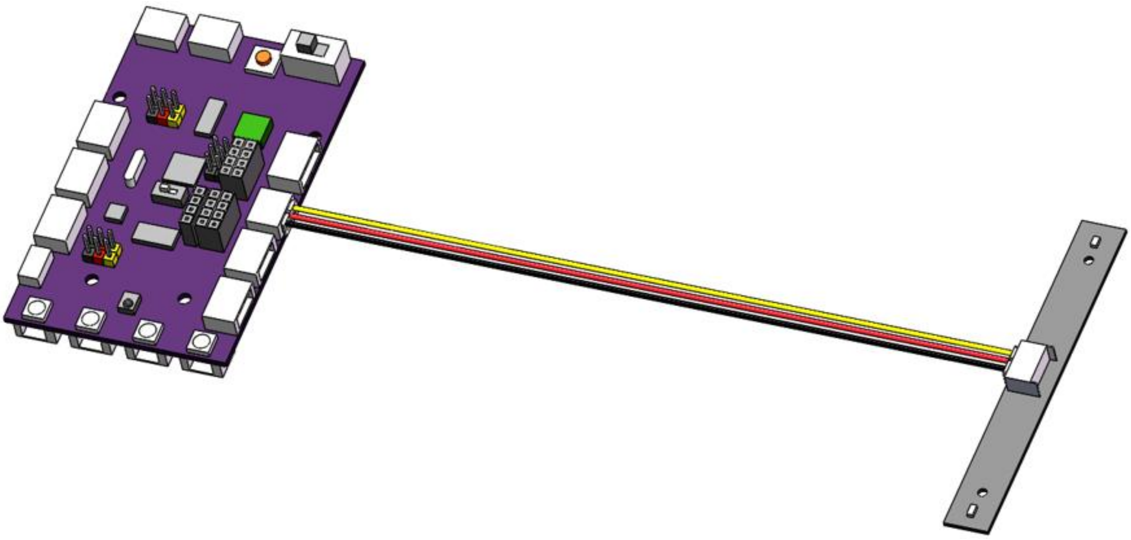


Figure as below：

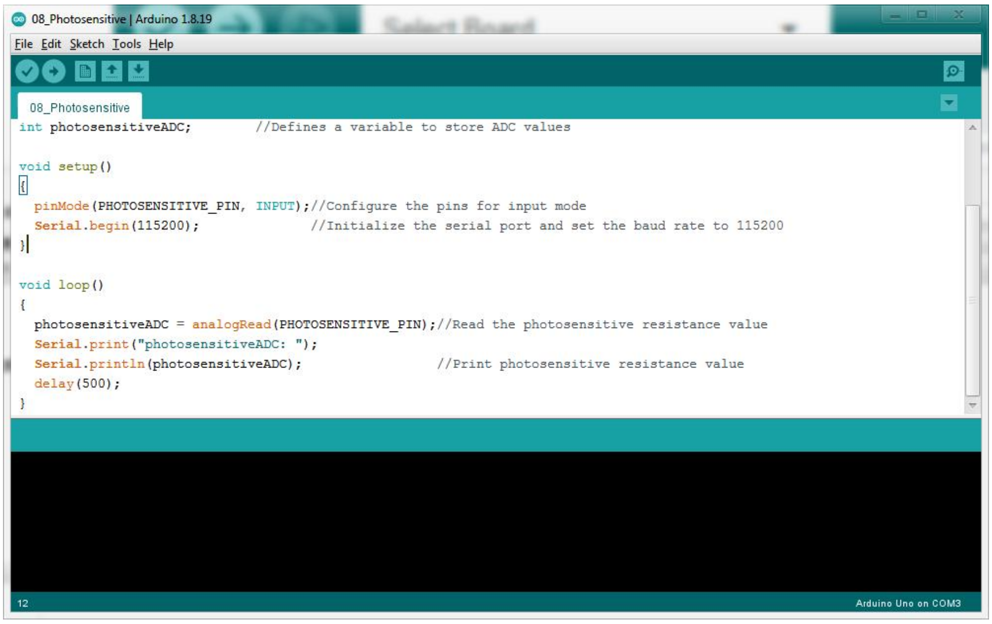




12.5 Demonstration

1. Connect your computer and Adeept Robot Control Board (Arduino Board) with a USB cable.

2. Open "**08_Photosensitive**" folder in "/Code", double-click "08_Photosensitive.ino".

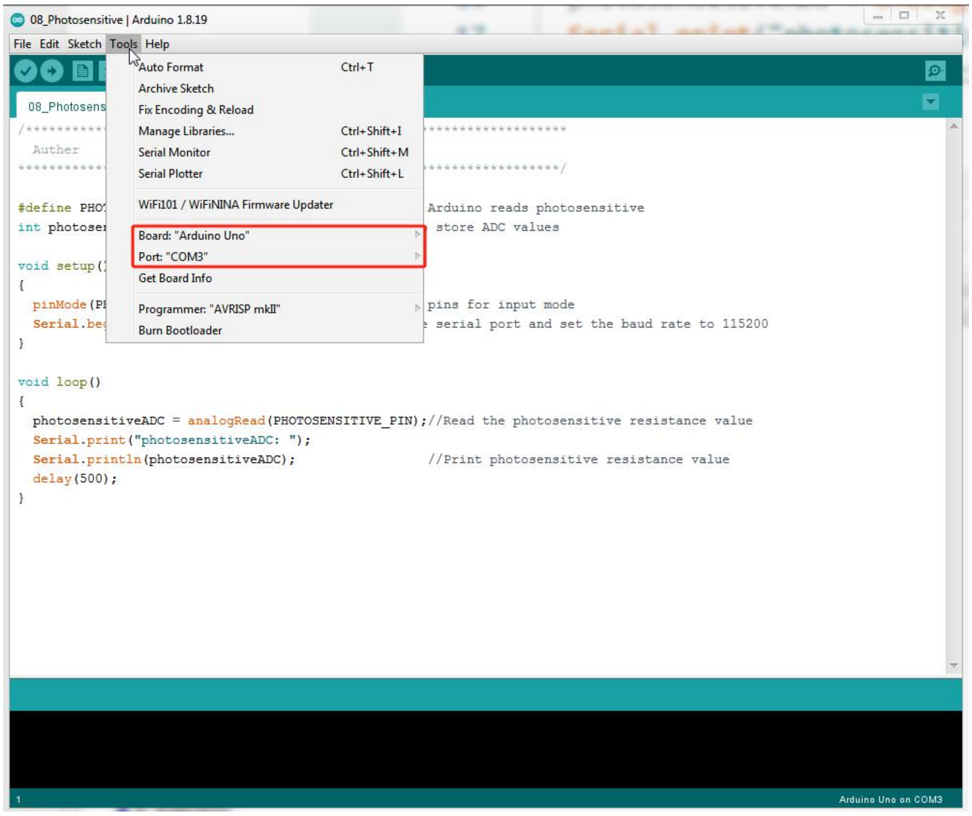


3. Select development board and serial port.

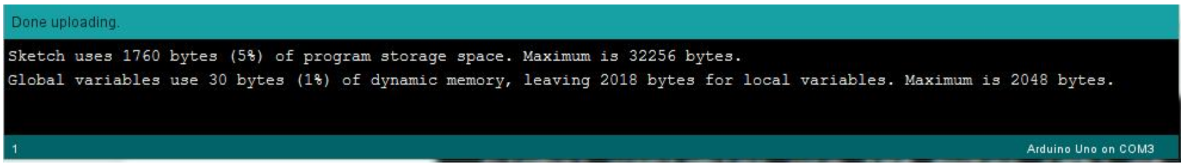
Board: Tools--->Board--->Arduino AVR Boards--->Arduino Uno

Port: Tools --->Port--->COMx

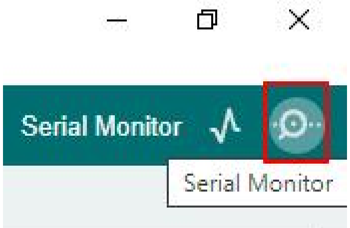
Note: The port number will be different in different computers.

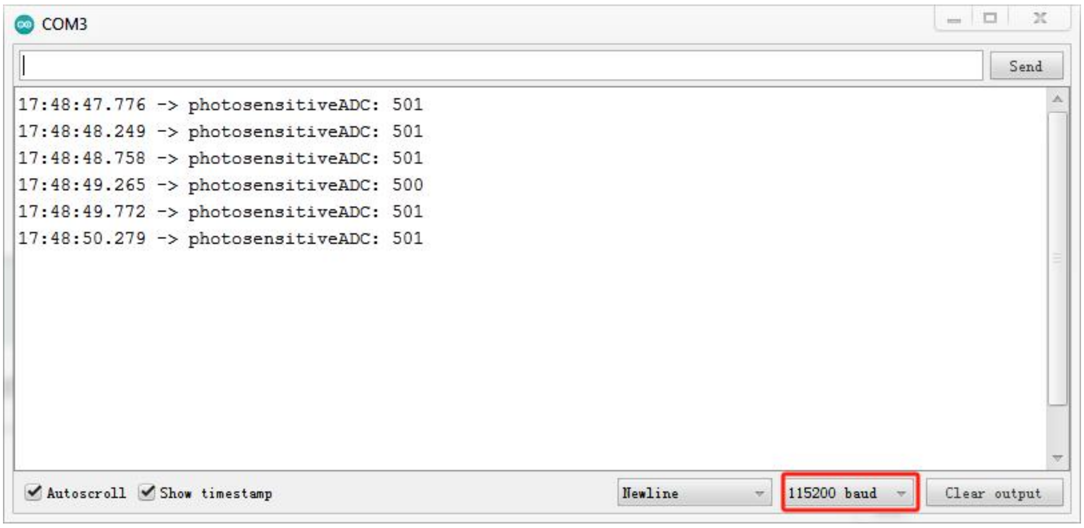


4. After opening, click  to upload the code program to the Arduino. If there is no error warning in the console below, it means that the Upload is successful.



5. Click Serial Monitor, Set the baud rate as 115200.





6. You will see the detected ADC value displayed on the screen.

12.6 Code

Complete code refer to 08_Photosensitive.ino

```
/***********************************************************************
  Auther      : www.adeept.com
***********************************************************************/

#define PHOTOSENSITIVE_PIN A0 //Define the pins that Arduino reads photosensitive
int photosensitiveADC;        //Defines a variable to store ADC values

void setup()
{
  pinMode(PHOTOSENSITIVE_PIN, INPUT);//Configure the pins for input mode
```

```
  Serial.begin(115200);              //Initialize the serial port and set the baud rate to 115200
}

void loop()
{
  photosensitiveADC = analogRead(PHOTOSENSITIVE_PIN);//Read the photosensitive resistance value
  Serial.print("photosensitiveADC: ");
  Serial.println(photosensitiveADC);                 //Print photosensitive resistance value
  delay(500);
}
```

Code explanation

Initialization Stage:

In the setup(), first set the photoresistor connection pin to input mode to read analog signals, then initialize serial communication and set the baud rate to 115200 for easy viewing of ADC values on the serial monitor.

Loop Control Process:

Stage 1: Read the analog value of the photosensitive resistor connection pin through analogRead() and convert it to an ADC value.

Stage 2: Next, use the serial communication function to print the current ADC value on the serial monitor.

Stage 3: Finally, by delay (500); Delay the program by 500ms to avoid frequent data output and facilitate observation of data changes.