

Printout

Wednesday, September 18, 2024 9:12 AM

Section 1

MANE 3351

Laboratory 7 Session

Classroom Management

Agenda

- Topics
 - Light Detecting Resistors
 - Fritzing Diagrams
 - Circuit Diagrams
 - Analog to Digital Conversion
- Discuss Lab 3 Assignment
- Distribute materials needed for Lab Three

Resources

Assignments

- Collect materials needed for Laboratory Assignment Three
- Complete Lab 3 before 9/25/2024 at 2:00 pm

Note: laboratory session will not meet 9/23. Dr. Timmer will be in his office during the regularly scheduled laboratory session and office hours. Please demonstrate your working system before the deadline.

Light Detecting Resistor (LDR)

- A Light Detecting Resistor (LDR), also known as a photoresistor or photocell, is an electronic component that changes its electrical resistance based on the amount of light it is exposed to.
 - Resistance decreases as light intensity increases.
 - Resistance increases in the dark.

LDR material created by ChatGPT

Key Characteristics of LDRs

- ① Made from semiconductor materials like cadmium sulfide (CdS).
- ② Resistance decreases with increased light exposure.
- ③ Has a slower response time compared to photodiodes or phototransistors.
- ④ Typically used in light-sensing circuits, brightness control, and photography.

Working Principle of LDR

- In the dark, LDR has high resistance (megaohms range).
- When exposed to light, photons provide energy to electrons, decreasing resistance.
- More light means more electrons are excited, resulting in lower resistance.

Common Uses of LDRs

- Automatic street lights and night lamps.
- Brightness control for screens and displays.
- Light level measurement in cameras.

Limitations of LDRs

- Sensitive to environmental conditions, such as temperature.
- Not suitable for precise light measurements.
- Relatively slow response time to changing light conditions.

Fritzing Diagrams

A Tool for Visualizing and Designing Electronic Circuits
Fritzing Diagram slides generated by ChatGPT

#####

What

is a

Fritz-

ing

Dia-

gram?

- Vi-

sual

rep-

re-

sen-

ta-

tion

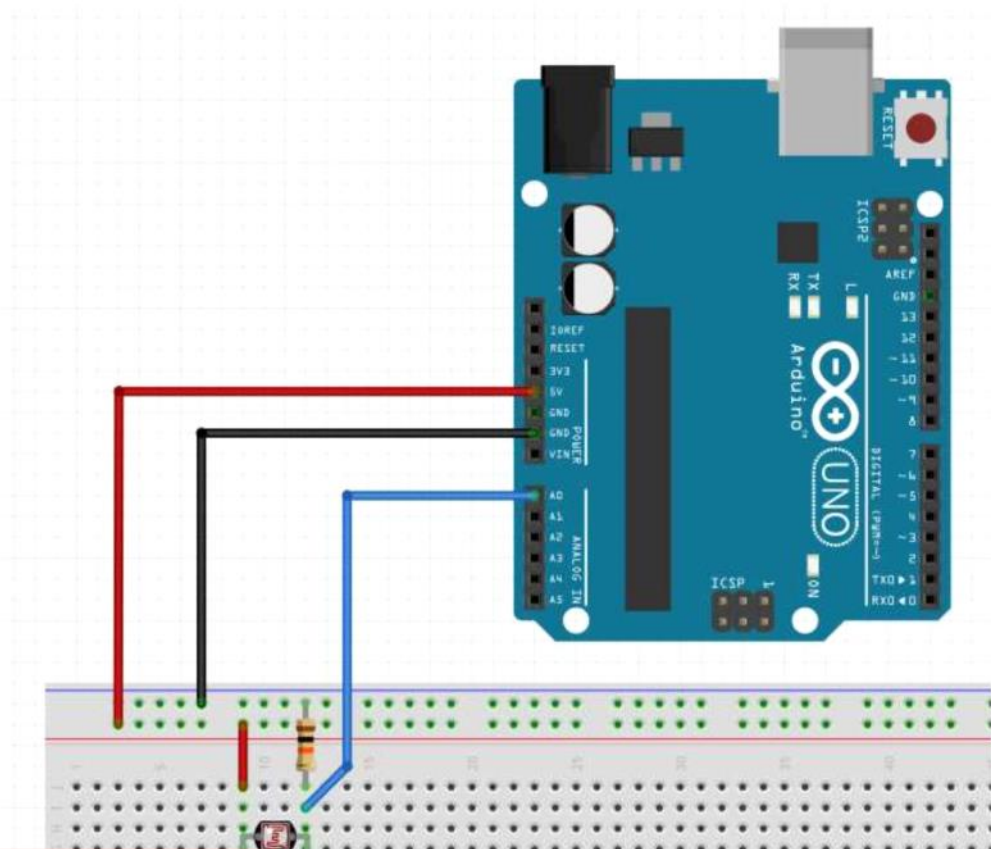
Why Use Fritzing Diagrams?

- **Learning Tool:** Simplifies circuit design and helps beginners understand how circuits work.
- **Easy Prototyping:** Quickly create visual representations for planning and avoiding mistakes.
- **Clear Communication:** Universally understood diagrams for reports, presentations, and online sharing.

Fritzing Diagram Summary

- Fritzing diagrams help visualize and design electronic circuits.
- Move from an idea to a physical prototype with clear, easy-to-understand diagrams.
- Useful for learning, communicating, and building electronic projects effectively.

Fritzing Diagram for LDR



Circuit Diagram

- A **circuit diagram** (or schematic) is a graphical representation of an electrical circuit.
- It uses **standardized symbols** to show how different components are connected.
- Essential for understanding, designing, and troubleshooting electrical and electronic systems.

Circuit Diagram slides created by ChatGPT

Key Features of Circuit Diagrams

- **Standardized Symbols:**
 - Symbols represent different components: resistors, capacitors, transistors, diodes, etc.
 - Universal language for engineers worldwide.
- **Connections and Lines:**
 - Lines represent electrical connections (wires).
 - Dots or nodes show where multiple connections meet.

Types of Circuit Diagrams

① Schematic Diagrams:

- Show logical connections and functionality.
- Focus on how components are electrically connected.

② Wiring Diagrams:

- Show the physical layout and connections of components.
- Used for installation and assembly.

③ Block Diagrams:

- High-level overview of a system.
- Shows main functional units and their relationships.

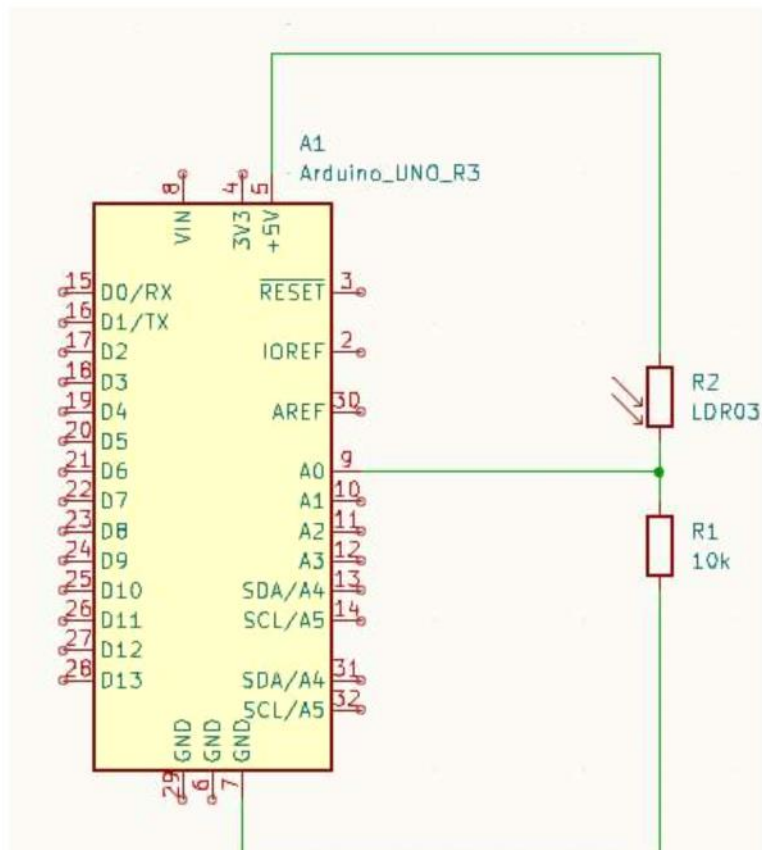
Why Use Circuit Diagrams?

- **Understanding Circuit Operation:**
 - Visualize how electrical components interact.
- **Design and Development:**
 - Plan and create new electronic devices.
- **Troubleshooting and Repair:**
 - Identify faults and issues in existing circuits.
- **Documentation and Communication:**
 - Share designs and modifications clearly.

Circuit Diagram Summary

- Circuit diagrams use symbols and lines to represent electrical circuits.
- They are crucial for understanding, designing, analyzing, and communicating about electronic systems.
- Learning to read and create circuit diagrams is a fundamental skill for engineering students.

Circuit Diagram for LDR



Circuit Diagrams vs. Fritzing Diagrams

What is a Circuit Diagram?

- A graphical representation of electrical circuits using standardized symbols.
- Focuses on logical and functional connections between components.
- Used for designing, analyzing, and troubleshooting circuits.

What is a Fritzing Diagram?

- A visual representation of components connected on a breadboard or PCB.
- Uses realistic images of components to show their physical connections.
- Ideal for beginners, education, and rapid prototyping.

Purpose and Usage

- **Circuit Diagrams (Schematics)**
 - Used to understand, design, and troubleshoot electrical circuits.
 - Focus on the logical flow and electrical functionality.
- **Fritzing Diagrams**
 - Used to visualize physical connections on a breadboard or PCB.
 - Focus on practical assembly and hands-on learning.

Representation Style

- **Circuit Diagrams**

- Use standardized symbols (e.g., resistors, capacitors) to represent components.
- Show electrical functionality and connections abstractly.

- **Fritzing Diagrams**

- Use realistic images/icons of components.
- Represent physical layout and connections on a breadboard or PCB.

Level of Detail

- **Circuit Diagrams**

- Provide high-level abstraction, focusing on component values and electrical paths.
- Suitable for detailed analysis and design.

- **Fritzing Diagrams**

- Provide a tangible, easy-to-understand physical view.
- Suitable for practical assembly, prototyping, and educational purposes.

Ease of Use and Accessibility

- **Circuit Diagrams**
 - Require knowledge of electrical symbols and concepts.
 - Best for professionals and advanced users.
- **Fritzing Diagrams**
 - Easier for beginners; visually intuitive.
 - Ideal for learning, teaching, and hobby projects.

Software Tools

- **Circuit Diagrams**
 - Created with tools like **Eagle, KiCad, Altium Designer**.
 - Offer advanced features for simulation and PCB design.
- **Fritzing Diagrams**
 - Created using the **Fritzing** software.
 - Designed for rapid prototyping, education, and easy visualization.

When to Use Each Type of Diagram

- **Use Circuit Diagrams When:**
 - You need to understand the electrical function and behavior.
 - Working on complex projects requiring detailed analysis.
 - Designing circuits for professional or advanced purposes.
- **Use Fritzing Diagrams When:**
 - Building physical prototypes on a breadboard or PCB.
 - Teaching or learning basic electronics concepts.
 - Creating projects in a maker or hobbyist environment.

Summary

- **Circuit Diagrams:** Focus on electrical functionality, design, and analysis.
- **Fritzing Diagrams:** Focus on physical layout, ease of use, and rapid prototyping.
- Both tools are essential for different stages of learning and building electronics.

Analog-to-Digital Conversion on Arduino Uno

What is Analog-to-Digital Conversion (ADC)?

- **Analog-to-Digital Conversion (ADC)** is the process of converting an analog signal (continuous voltage) into a digital signal (discrete values).
- Digital devices like the Arduino Uno need to convert real-world analog signals into a format they can process (binary values).

How Does the Arduino Uno Perform ADC?

- The Arduino Uno uses a built-in **Analog-to-Digital Converter (ADC)** to read analog signals.
- The ADC on the Arduino Uno has a **10-bit resolution**, which means it converts an analog input into a digital value between **0** and **1023**.
- The ADC reads the voltage on one of its analog input pins (A0 to A5) and converts it to a digital value.

ADC Resolution and Voltage Range

- **10-bit resolution** means the input voltage is divided into **1024 levels** (2^{10}).
- The Arduino Uno's ADC maps input voltages from **0V** (ground) to **5V** (reference voltage) by default.
- Each step in the digital output corresponds to approximately **4.9 mV** ($5V/1024$).

$$\text{Voltage} = 5.0 \times (\text{SensorValue} / 1023.0)$$

How to Use ADC on Arduino Uno

① **Connect the Analog Sensor:**

- Connect the output of an analog sensor (e.g., potentiometer, temperature sensor) to one of the analog input pins (A0 to A5).

② **Use `analogRead()` Function:**

- The `analogRead(pin)` function reads the voltage on the specified analog pin.
- Returns a value between **0** (0V) and **1023** (5V).

Example: Reading an Analog Sensor

```
void setup() {  
  Serial.begin(9600); // Initialize serial communication at 9600  
}  
  
void loop() {  
  int sensorValue = analogRead(A0); // Read the input on analog  
  Serial.println(sensorValue); // Print the value from analog  
  delay(500); // Wait for 500 milliseconds  
}
```

Sensor Value = 880

$$\text{Voltage} = 5.0 \left(\frac{880}{1023} \right) = 4.3 \text{ V} \checkmark$$