

SOLAR TRACKING FOR ECONOMY

BY

MR. TRAIRAT THASSANASARIT

**A PROJECT SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE OF BACHELOR OF
ENGINEERING IN MECHATRONICS
KING MONGKUT'S INSTITUTE OF TECHNOLOGY
LADKRABANG
ACADEMIC YEAR 2021**

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

FACULTY OF ENGINEERING
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG
PROJECT CERTIFICATE

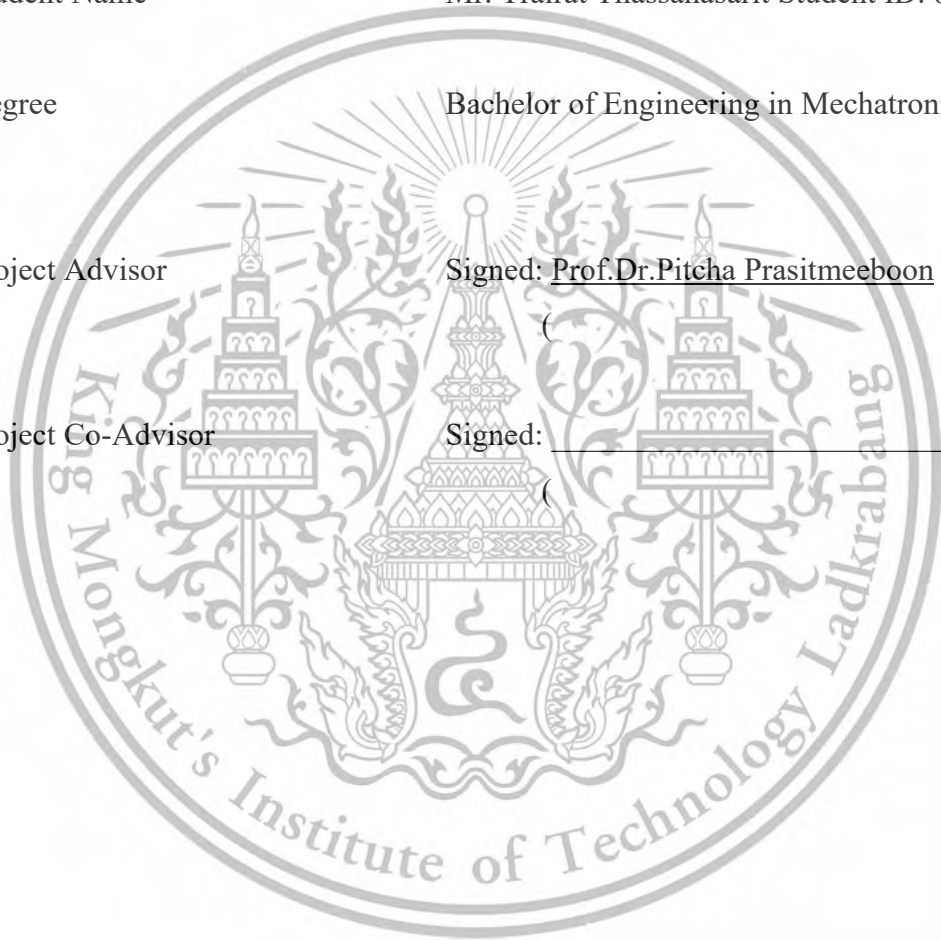
Project Title Solar tracking for economy

Student Name Mr. Trairat Thassanasarit Student ID. 62011284

Degree Bachelor of Engineering in Mechatronics

Project Advisor Signed: Prof.Dr.Pitcha Prasitmeeboon
()

Project Co-Advisor Signed: _____
()



Project Title	Solar tracking for economy
Student Name	Mr. Trairat Thassanasarit 62011284
Degree	Bachelor of Engineering in Mechatronics
Project Advisor	Assoc. <u>Prof.Dr.Pitcha Prasitmeeboon</u>
Project Co-Advisor	Assoc.Prof. International Program, Ph.D.
Academic Years	2022

ABSTRACT

Modern homes and businesses employ solar cell panels to convert solar energy into electricity. to obtain energy for usage in various energy sources, such factory electricity or a home's electrical system.

In order to collect the maximum electricity, most solar panels are typically erected by businesses facing the sun. However, the sun will always move in accordance with the Earth's orbit around the sun. because the solar panel is installed, it does not rotate in accordance with the movement of the sun, preventing the solar panel from receiving solar energy at its maximum efficiency.

Therefore, we developed an model to let the solar panel to continuously track solar energy until the sun sets. to enable us to obtain the most solar energy possible. for the effectiveness of being used in business or to provide electrical energy for homes Inadequate power reserves could contribute to the organization's damage.

ACKNOWLEDGEMENTS

Prof.Dr.Pitcha Prasitmeeboon, who offers guidance and funding from the payment of tax invoices from the Faculty of Electrical Engineering, International Program, supports our activity. that ensures the efficiency and thoroughness of this project, as well as assistance from elders and friends who can contribute advice or equipment. We now have enough money to finish this study endeavor.



This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

TABLE OF CONTENTS

	Page
ABSTRACT	(i)
ACKNOWLEDGEMENTS	(ii)
LIST OF TABLES	(iii)
LIST OF FIGURES	(ix)
LIST OF SYMBOLS/ABBREVIATIONS	(x)
CHAPTER 1 INTRODUCTION	
1.1 Background information and the issue's importance	1
1.2 Objective	1
1.3 Report Outline	2
CHAPTER 2 REVIEW OF YOUR SUBJECT AREA AND TOPICS	3
2.1 Solar energy	3
2.2 Arduino	4
2.3 Solar cell	23
2.4 The economics of solar power	29
2.5 Chapter Summary	34
CHAPTER 3 METHODOLOGY	35
3.1 Introduction	35
3.1.1 Hardware design	36
3.1.2 Schematics design	37

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

3.2 Equipment	37
3.3 Coding to Arduino	44
3.4 Problems in method	47
3.5 Proposed Solution	47
3.6 Summary	47
CHAPTER 4 EXPERIMENTAL RESULT	48
4.1 Introduction	48
4.2 Testing summary	48
4.3 Evaluation	51
4.4 Testing and Evaluation Summary	52
CHAPTER 5 CONCLUSION	53
5.1 Summary	53
5.2 Key Points	53
REFERENCES	54
APPENDICES	
APPENDIX A	59

LIST OF TABLES

Tables	Page
3.1 Block diagram Solar tracking	35
3.8 Length of pulse	42
4.3 Table time evaluation	51



This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

LIST OF FIGURES

Figures	Page
2.1 This is an Arduino Uno	4
2.2 This is a screenshot of the Arduino IDE	5
2.3 Uno Arduino	7
2.4 LDR	8
2.5 Servo motor	8
2.6 Arduino Uno board	9
2.7 Port A1-A5	9
2.8 Port 0-13	10
2.9 Reset switch	11
2.10 TX RX LEDs	11
2.11 Arduino code example	13
2.12 Serial monitor	14
2.13 Serial plotter	14
2.14 Library	16
2.15 Loop	18
2.16 Step 1	20
2.17 Uploading	20
2.18 Select port	21
2.19 ATmega328	22
2.20 Board Manager	23
2.21 Solar cell panel	24
2.22 JJ Thompson	25
3.2 Design solar tracking	36
3.3 Schematics design	37
3.4 MDF Board	37
3.5 LDR Sensor	39
3.6 Arduino	40
3.7 Servo motor	40

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

LIST OF FIGURES

Figures	Page
3.8 Processing servo motor	41
3.9 Resistor color code	43
4.1 Time 06.00-11.00 AM	48
4.2 Time 12.00 PM	49
4.3 Time 13.00-18.00 PM	50



LIST OF SYMBOLS/ABBREVIATIONS

Symbols/Abbreviations	Terms
EE	Electrical Engineering
SIIE	School of International Interdisciplinary Engineering Programs



This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

CHAPTER 1

INTRODUCTION

This chapter introduces the overarching themes of this report and places the motivation for the work into context. Thereafter, the rationale and goals defined for the investigation of the project are discussed, followed by a summary of the overall project. Finally, an overview of the dissertation is given on a per-chapter basis.

The goal of this study is to find a solution to the shortage of reserve power. because they cannot consistently collect solar energy at their highest efficiency resulting in inadequate power, therefore solar tracking was developed to address these issues.

1.1 Background information and the issue's importance

This research endeavor is inspired by the use of solar panels to transform solar energy into electricity today. Humans cannot watch the sun's movement throughout the day, which prevents the solar panels we install from turning to capture energy at maximum efficiency. Because our world revolves around the sun, which causes the sun that we can see with our naked eyes to move constantly. We became aware of this issue, which is why we investigated it.

Therefore, solar panels that follow the sun's path are crucial for obtaining enough energy with high efficiency to be utilized as a backup source to lower costs for households or businesses. It also keeps the system from disrupting the electrical power that comes from other sources. We will be able to know that we still have adequate backup power to use in the event of other sources of power failure. to avoid industry errors that have no negative effects on the economy.

1.2 Objective

1. In order to model while learning how to program the Arduino Uno board.
2. Track solar energy while studying how the Earth revolves around the Sun.
3. Investigate how different Arduino sensor function.
4. To avoid using inadequate backup power in homes and businesses.

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

5. Reduce household and industrial expenses.
6. Avoiding issues brought on by inadequate backup power.

1.3 Report Outline

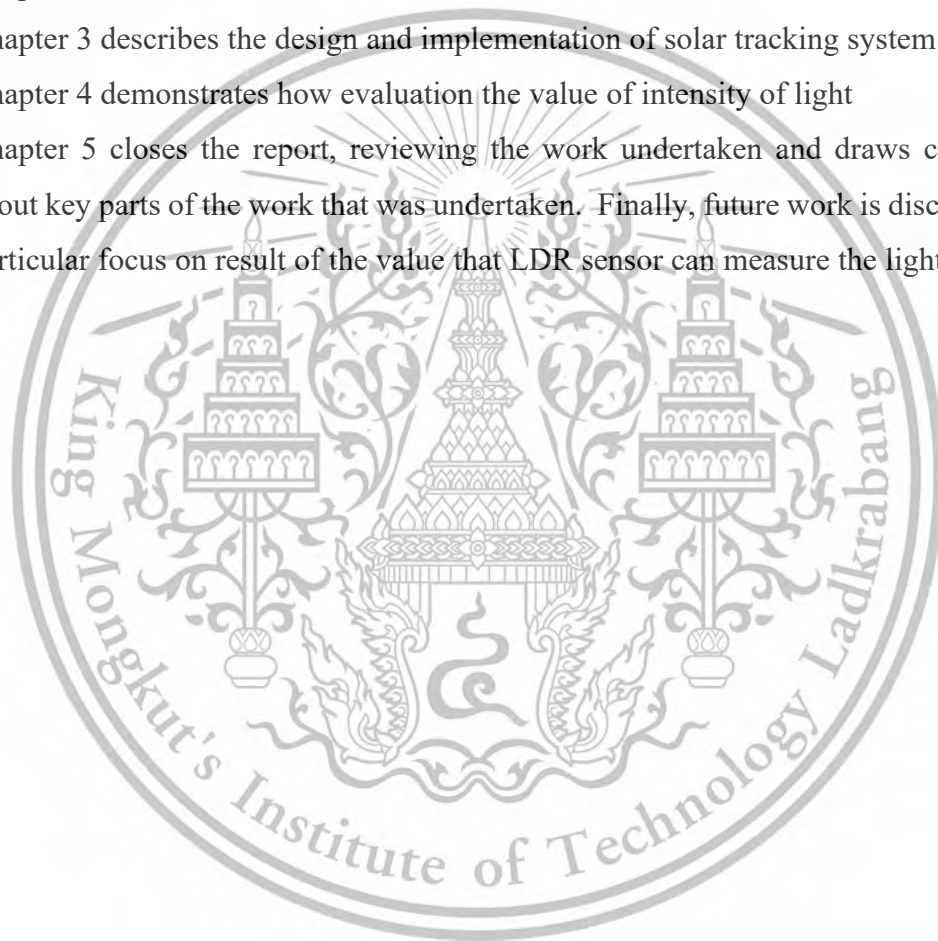
The rest of this report is organized as follows:

Chapter 2 reviews the introduction of solar cell

Chapter 3 describes the design and implementation of solar tracking system

Chapter 4 demonstrates how evaluation the value of intensity of light

Chapter 5 closes the report, reviewing the work undertaken and draws conclusions about key parts of the work that was undertaken. Finally, future work is discussed with particular focus on result of the value that LDR sensor can measure the light



CHAPTER 2

PRINCIPLES AND THEORIES

The goal of the solar tracking research project is to avoid having insufficient backup power for use in homes or businesses. due to the solar panel's inability to continuously track the sun's light as a result of the Earth's sun's orbit. causing the sun to move all the time We created the solar tracking specifically to track the sun all day long and avoid power interruptions for this reason. additionally capable of lowering costs in both households and businesses. by turning the solar panel in the direction of the highest energy by utilizing a light sensor to identify the side of the sun with the most energy. for the purpose of making solar energy consumption as efficient as possible.

2.1 SOLAR ENERGY

The sun is an amazing, renewable resource that can support life on earth and give all of its inhabitant access to clean, sustainable energy. In fact, the sun sends more energy to our planet in a single hour than the entire world's population uses in a whole year. Solar photovoltaic (PV) modules may turn solar energy into electricity (photo = light, voltaic = electricity).

Solar energy is captured by PV modules, which then transform it into an electrical current that may be used. The fact that the sun shines everywhere makes solar energy practical. Solar electric systems can be independent of the utility grid because to the ability of solar to be combined with batteries for energy storage, making them affordable for distant sites. Solar modules have no moving components making maintenance costs low, and they are highly reliable with a lengthy service life of 25+ years of guaranteed electricity. There is no need to drill for petroleum-based fuels, process them, or transport them to the location because solar electricity depends on the sun as its fuel source. As you can see, solar energy has a lot of benefits.

We can transition away from other polluting and unsustainable energy sources thanks to the design and widespread installation of PV systems. The demand for skilled people is increasing since the solar business is expanding.

In order to harness the power of the sun and acquire the technical PV design and installation skills necessary to offer clean power and good change to their communities,

This material is reserved for educational use only, not allowed for commercial use.

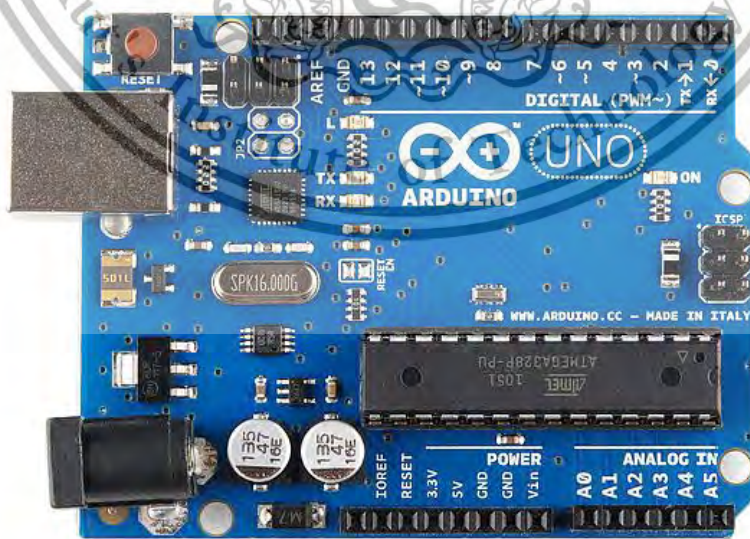
Forbidden to modify the content, and cite the document when use.

women and men from all over the world are trained by Remote Energy (RE), a 501(c)(3) for-impact organization. Programs for technicians equip the expanding labor force with the applicable skills needed to be competent and competitive. Programs for solar educators equip local instructors with the curriculum, mentoring opportunities, and expertise required to satisfy current regional training standards.

2.2 ARDUINO

The open-source Arduino platform is used to create electrical projects. With Arduino, you can write and upload computer code to a physical programmable circuit board (commonly called a microcontroller) using a piece of software called the IDE (Integrated Development Environment), which runs on your computer.

With those just getting into electronics, the Arduino platform has grown rather popular, and for good reason. The Arduino does not require a separate piece of hardware (referred to as a programmer) in order to load fresh code onto the board; instead, you can do so by using a USB cable, in contrast to the majority of earlier programmable circuit boards. Additionally, the Arduino IDE employs a condensed form of C++ that makes learning to program simpler. Finally, Arduino offers a standard form factor that separates the micro-functionality controller into a more usable packaging.

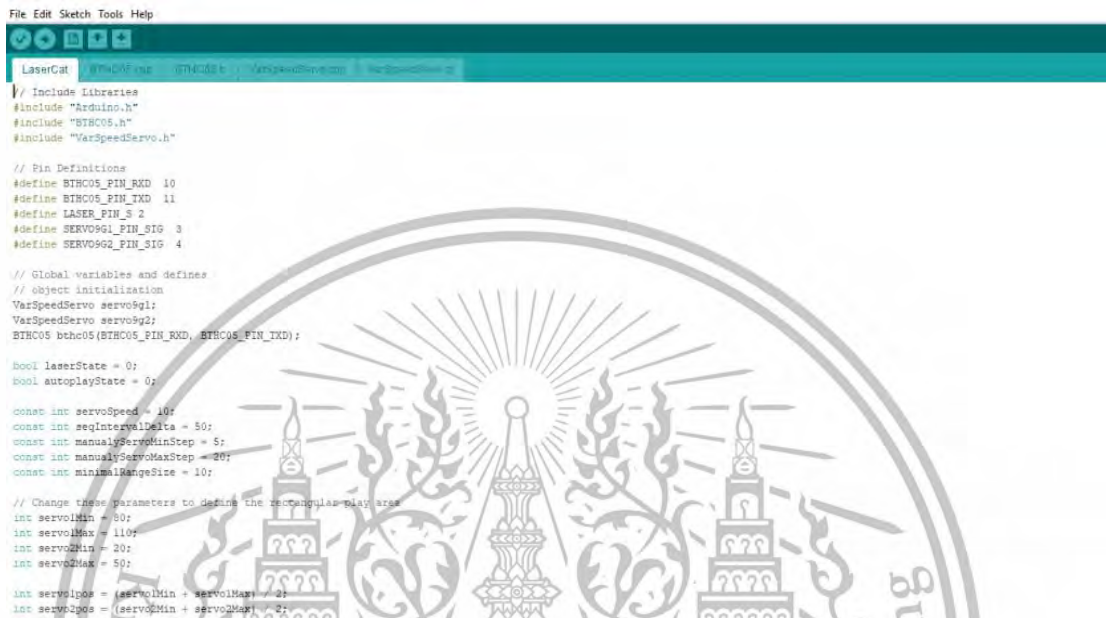


2.1 This is an Arduino Uno

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

One of the more well-liked Arduino boards and a wonderful option for novices is the Uno.



```

File Edit Sketch Tools Help
LaserCat
// Include Libraries
#include "Arduino.h"
#include "BTBC05.h"
#include "VarSpeedServo.h"

// Pin Definitions
#define BTBC05_PIN_RXD 10
#define BTBC05_PIN_TXD 11
#define LASER_PIN_S 2
#define SERVO01_PIN_SIG 3
#define SERVO02_PIN_SIG 4

// Global variables and defines
// object initialization
VarSpeedServo servo01;
VarSpeedServo servo02;
BTBC05 btbc05(BTBC05_PIN_RXD, BTBC05_PIN_TXD);

bool laserState = 0;
bool autoplayState = 0;

const int servoSpeed = 10;
const int seqIntervalDelta = 50;
const int manuallyServoMinStep = 5;
const int manuallyServoMaxStep = 20;
const int minimalRangeSize = 10;

// Change these parameters to define the rectangular play area
int servo1Min = 30;
int servo1Max = 110;
int servo2Min = 20;
int servo2Max = 50;

int servo1Pos = (servo1Min + servo1Max) / 2;
int servo2Pos = (servo2Min + servo2Max) / 2;

```

2.3 This is a screenshot of the Arduino IDE.

To put it briefly, an Arduino is an open hardware development board that makers, hobbyists, and tinkerers may use to create and construct objects that interact with the physical environment. While "Arduino" refers to a particular type of board design, it can also be used to describe a business that produces a particular version of these boards. It is also frequently used to refer to the community surrounding compatible boards produced by other people or businesses that perform similarly.

The circuit board of an Arduino contains a variety of parts that interface with one another. The layout has evolved through time, and some iterations now include additional components. However, a standard board is likely to have the following pieces:

1. There are other pins that can be utilized to link the Arduino to different components. There are two types of these pins:
 - 1.1 digital pins that only have two possible states: on and off. Digital I/O pins on Arduinos typically number 14.

1.2 For more precise control, analog pins, which can read a variety of values, are helpful. There are typically six of these analog pins on an Arduino.

These pins are set up in a certain way so that if you purchase an add-on board made to fit into them, sometimes referred to as a "shield," it should fit into most Arduino-compatible devices without too much difficulty.

2. a power connector that supplies power to the device itself as well as a low voltage that, if the linked components' power requirements are low enough, can power LEDs and different sensors. A tiny battery or an AC adaptor can be connected to the power connector.
3. The Arduino's main component, called a microcontroller, enables programming so that you may program it to carry out orders and make decisions depending on numerous inputs. Depending on the type of Arduino you purchase, the precise chip may vary, but they are often Atmel controllers, typically an ATmega8, ATmega168, ATmega328, ATmega1280, or ATmega2560. Although there are minor differences between these chips, a novice will mostly notice the variations in onboard memory sizes.
4. On most contemporary boards, a regular USB port serves as the serial connector. With the help of this connector, you can talk with the board from your computer and download fresh software for it. A USB port can frequently be used to power Arduinos, eliminating the need for a separate power source.
5. A number of additional minor parts, such as an oscillator and/or a voltage regulator, which are crucial to the board's functionality but which you normally don't contact with directly; just be aware that they are there.

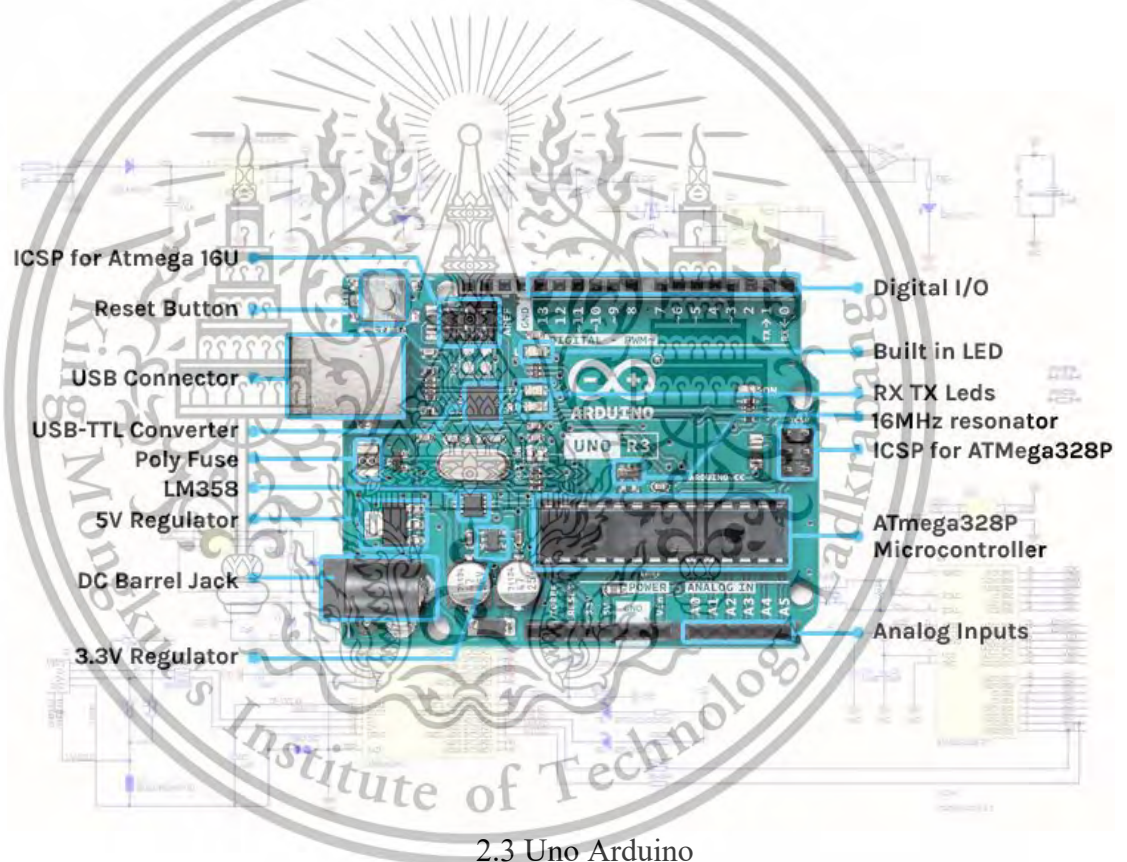
2.2.1 Arduino Sensor

The hardware board of Arduino has a microcontroller and a software IDE for uploading the code. Numerous sensors that are compatible with Arduino have been introduced as a result of Arduino's growing popularity among hobbyists.

The market is filled with a variety of Arduino sensors. These sensors support Arduino's ability to communicate with its environment and create new applications.

Examples of Arduino Sensor

- Uno Arduino
- LDR
- Servo motor
- Digital thermal sensor – Temperature sensor
- Rotary Encoder Module
- MQ-2 Gas sensor
- SW-420 Motion sensor
- Humidity and Rain Detection sensor
- Passive Buzzer Module
- Speed sensor Module



2.3 Uno Arduino

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.



2.2.2 Arduino Uno

It is a straightforward, frequently used prototype board that even complete beginners can use to learn about electronics. Being the basic one, it is vital for any electronics developer to comprehend its numerous components.

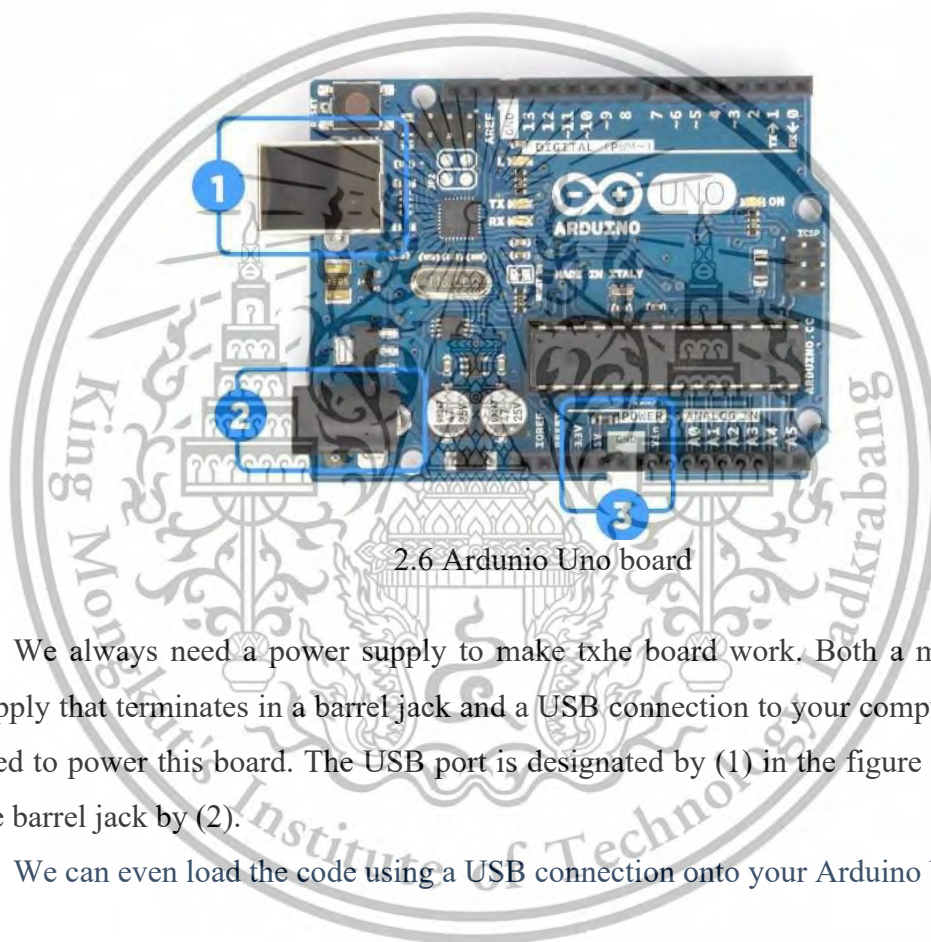
An ATmega328P microprocessor is included with UNO. There are two types of it: one has a surface mount type and the other has a through-hole microcontroller connection. If there is a problem with the through-hole model, you can swap out the chip with a new one.

This page is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

An 8-bit microcontroller with AVR architecture, the Arduino UNO provides a variety of functions and functionalities.

There are 14 digital input-output (I/O) pins in all on the UNO, which you can utilize for input or output. The six pins available from these 14 pins can be used to generate PWM signals. This board's pins operate at 5V with a 20mA current.



2.6 Arduino Uno board

We always need a power supply to make the board work. Both a main power supply that terminates in a barrel jack and a USB connection to your computer can be used to power this board. The USB port is designated by (1) in the figure above, and the barrel jack by (2).

We can even load the code using a USB connection onto your Arduino board.



2.7 Port A1-A5

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

The above image shows the following:

1. GND (3): GND is short for "Ground," which is what your circuit uses to ground itself.
2. 5V (4) & 3.3V (5): Power can be supplied at five volts using the 5V pin, and 3.3 volts using the 3.3V pin.
3. Analog (6): These pins with the letters (A0–A5) are referred to as analog pins. An analog sensor will be changed into a digital one.



2.8 Port 0-13

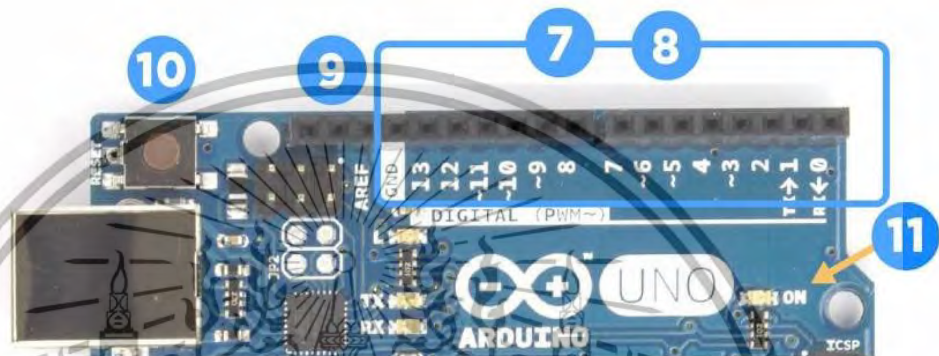
The top right of the above image highlights the 14 I/O pins that can perform specific functions, as stated below:

1. We can use pins 0 and 1 for serial communication to receive and send serial data. They allow you to interact with users via the serial monitor and program the Arduino board.
2. We can provide external interrupts using pins 2 and 3. These pins cause an outside event.
3. Eight-bit PWM output uses six pins, from 3 to 11.
4. In particular for SPI communication, pins 10, 11, 12, and 13 are for SS, MOSI, MISO, and SCK, respectively.

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

5. An LED connection is already established into pin 13. The LED is turned on when this pin is set to HIGH, and it is turned off when it is set to LOW.
6. Analog Reference, often known as AREF, is a term used to set an external reference voltage (0 -5 Volts).
- 7.



2.9 Reset switch

1. The reset button is indicated by (10) at the upper left of the above figure. This button connects the reset pin to the ground and restarts the uploaded code. You can try again by repeatedly pressing the reset button in case your code doesn't work.
2. The Power LED Indicator, which will illuminate when you connect your Arduino to a source, is designated by the number (11).



2.2 TX RX LEDs

This material is reserved for educational use only, not allowed for commercial use.

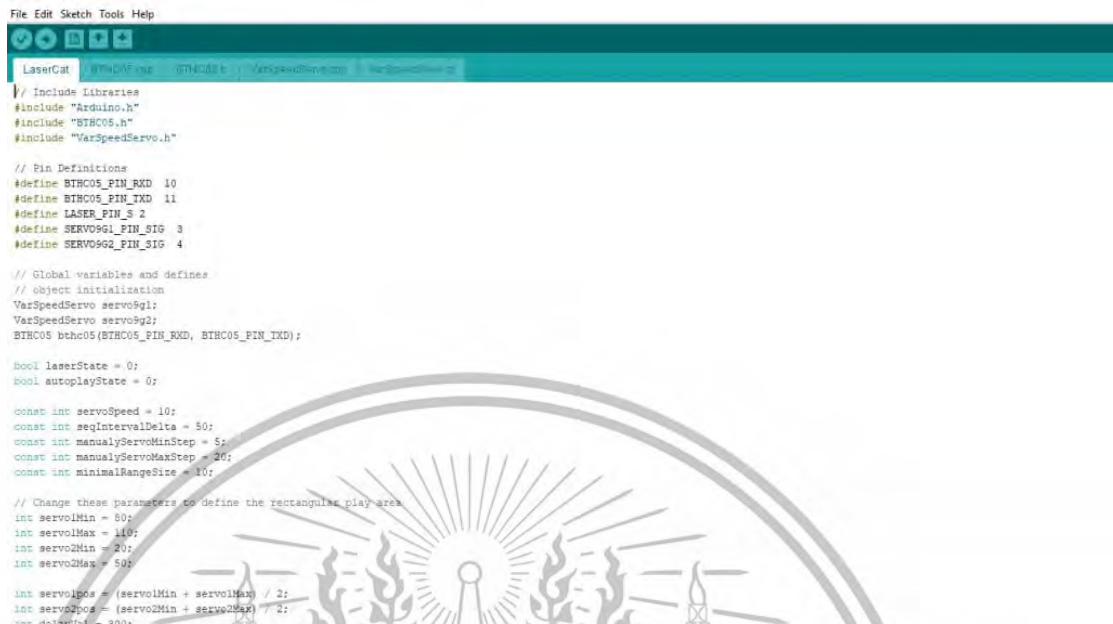
Forbidden to modify the content, and cite the document when use.

1. The TX RX LEDs is shown in the image above by (12), where TX stands for transmit and RX for receive. When communicating serially, these are employed. These LEDs provide visual cues while receiving or delivering data using Arduino.
2. The integrated circuit (13) also referred to as the brain of Arduino is described. On the top of the IC, you can see the IC type listed.
3. The voltage regulator, which aids in regulating the amount of voltage delivered to the Arduino board, is described in figure 14. As a gatekeeper, it stops more voltage from entering the circuit. Do not connect the Arduino to greater over 20 volts because it has limitations.

2.2.3 Arduino coding

With the inclusion of unique methods and functions, which we'll discuss later, Arduino programming is written in C++. Human-readable programming languages include C++. A "sketch" the name given to Arduino code files is created and then processed before being translated into machine language.

The primary text editor used for Arduino programming is the Arduino Integrated Development Environment (IDE). Before uploading your code to the board, you wish to program, you will type it up there. Sketches are the name given to Arduino code.



```

LaserCut #Pin:05.png 07/14/2018 1:11:11 VarSpeedServo.cpp VarSpeedServo.h
// Include Libraries
#include "Arduino.h"
#include "BH1750.h"
#include "VarSpeedServo.h"

// Pin Definitions
#define BH1750_PIN_RXD 10
#define BH1750_PIN_TXD 11
#define LASER_PIN_S 2
#define SERVO1_PIN_SIG 3
#define SERVO2_PIN_SIG 4

// Global variables and defines
// object initialization
VarSpeedServo servo1;
VarSpeedServo servo2;
BH1750 bh1750(BH1750_PIN_RXD, BH1750_PIN_TXD);

bool laserState = 0;
bool autoplayState = 0;

const int servoSpeed = 10;
const int seqIntervalDelta = 50;
const int manuallyServoMinStep = 5;
const int manuallyServoMaxStep = 20;
const int minimalRangeSize = 10;

// Change these parameters to define the rectangular play area
int servoMin = 80;
int servoMax = 110;
int servo2Min = 20;
int servo2Max = 50;

int servo1pos = (servoMin + servoMax) / 2;
int servo2pos = (servo2Min + servo2Max) / 2;

```

2.11 Arduino code example

As we can see, the IDE is designed simply. The menu bar has only 5 headers, and there are several buttons that you can use to validate and upload your sketches underneath. In essence, your sketches are converted and compiled into code that Arduino can comprehend via the IDE. Our Arduino code is then uploaded to the board's memory after it has been compiled.

The user only needs to push a button to begin compiling their sketch (a guide to this can be found below).

A warning message that asks the user to make adjustments will appear if the Arduino code contains any errors. Most new users often face trouble with compiling due of Arduino's tight syntax requirements. When using Arduino, any punctuation errors will result in the code failing to build and an error message.

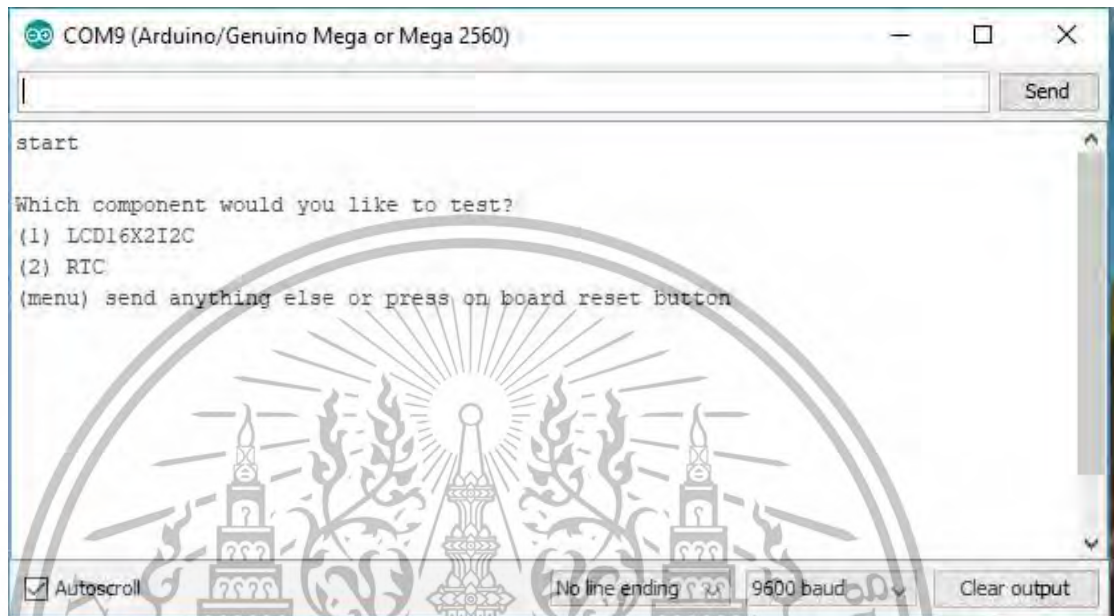
A serial plotter and monitor

By selecting the magnifying glass icon on the IDE's top right side or from the tool's menu, the Arduino serial monitor may be launched. The serial monitor is a fantastic tool for real-time monitoring and debugging and is primarily used for computer-based communication with the Arduino board. We must utilize the Serial class in order to use the monitor.

This material is reserved for educational use only, not allowed for commercial use.

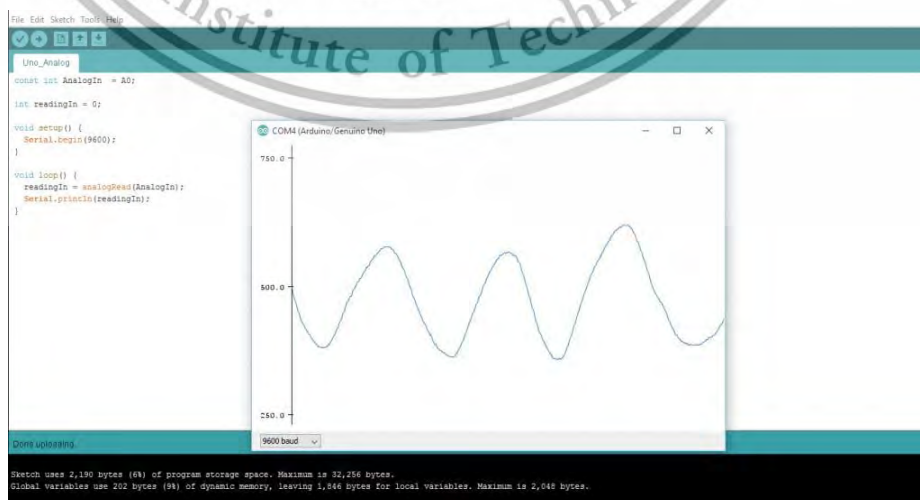
Forbidden to modify the content, and cite the document when use.

As we can see in the screenshot below, the code you get from circuito.io includes a test section that enables you to test each component using the serial monitor:



2.12 Serial monitor

The Arduino IDE also includes the Arduino serial plotter, which enables you to create a real-time graph of your serial data. We may much more easily analyze your data with the serial plotter's visual display. We can make graphs, graphs with negative values, and analyze waveforms.



2.13 Serial plotter

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

Arduino Hardware and Code Debugging

Unlike other platforms for writing software, Arduino lacks a built-in debugger. For monitoring and debugging, users can either use third-party software or the serial monitor to print the active operations of Arduino.

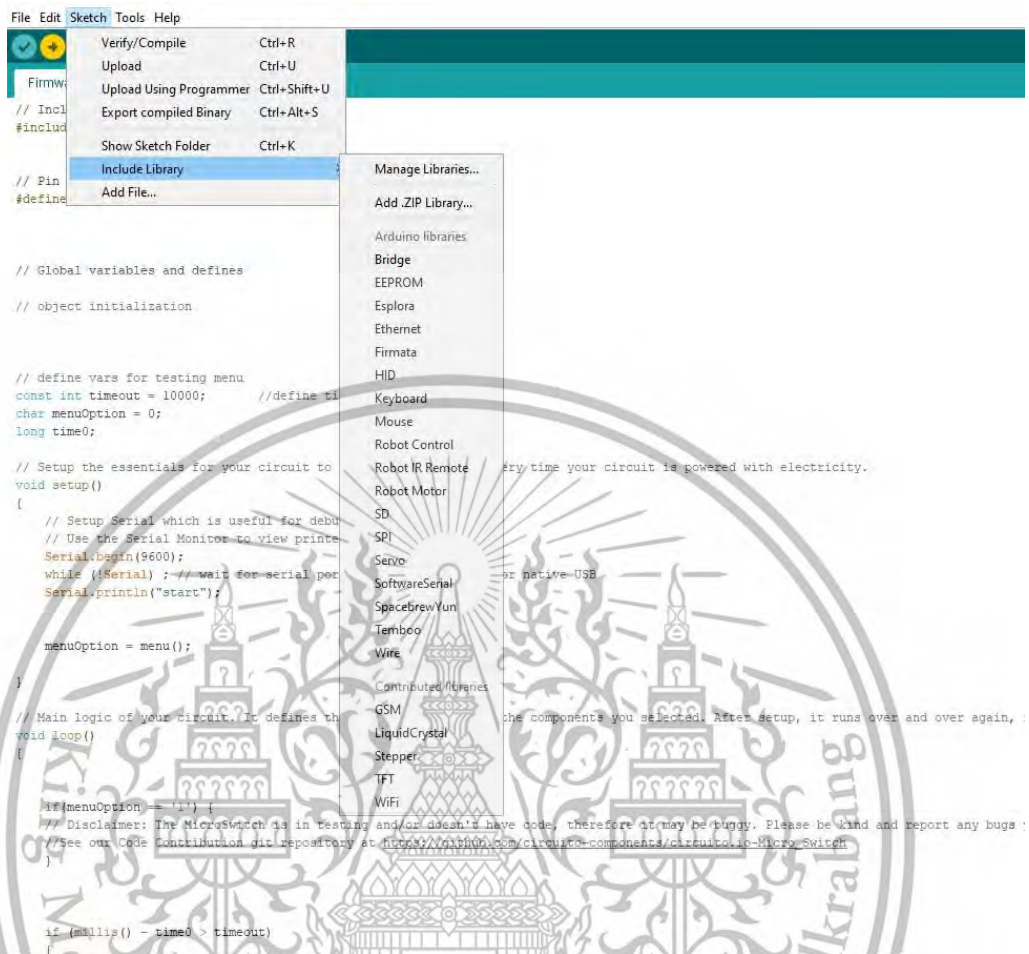
We can print variable values and debugging notes to the serial monitor by using the Serial class. This will use serial pins 0 and 1, which are connected to the USB port, on the majority of Arduino models.

Code Structure

Library Systems

Similar to other popular programming platforms, Arduino has built-in libraries that offer fundamental functionality. The Arduino board's capabilities and functionalities can also be expanded by importing additional libraries. Libraries that interface with a certain component or libraries that implement new functionalities are roughly how these libraries are divided.

We must select Sketch > Import Library in order to import a new library.



2.14 Library

You must also use '#include' at the top of your .ino file to include external libraries. Additionally, you can make your own libraries to use with isolated sketches.

Definitions of Pin

We must specify which pin is being utilized and its purpose in order to use the Arduino pins. Using: is an easy way to specify the used pins.

```
"#define pinName pinNumber," etc.
```

The setup portion uses the pinMode () method to describe the functionality, which might be input or output.

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

Statements and Variables

We must define global variables and instances whenever you use Arduino so that they can be utilized later. In essence, a variable enables you to name and save a value for later use. For instance, you might archive sensor data so you can use it later. We just need to specify a variable's type, name, and initial value in order to declare it.

It is important to note that declaring global variables is not a must. To make it simple to use your values later on, it's advised that you declare your variables up front.

Instances

A class in computer programming is a group of variables and functions that are kept in one location. A constructor is a unique function that each class has that is used to generate an instance of the class. We must declare an instance of the class before we can utilize any of its functions.

Setup()

There must be a setup function in every Arduino sketch. This function, which only executes once, establishes the Arduino's initial state during boot.

We'll define the following terms here:

1. Pin functionality using the pinMode function
2. Initial state of pins
3. Initialize classes
4. Initialize variables
5. Code logic

Loop()

Every Arduino sketch also needs to have the loop method, which runs after setup(). It is the primary function, and as its name implies, it repeatedly executes in a loop. Your circuit's primary logic is described in the loop.

For instance:



```

File Edit Sketch Tools Help

Firmware

// define vars for testing menu
const int timeout = 10000; //define timeout of 10 sec
char menuOption = 0;
long time0;

// Setup the essentials for your circuit to work. It runs first every time your circuit is powered with electricity.
void setup()

// Setup Serial which is useful for debugging
// Use the Serial Monitor to view printed messages
Serial.begin(9600);
while (!Serial) ; // wait for serial port to connect. Needed for native USB
Serial.println("start");

menuOption = menu();
}

// Main logic of your circuit. It defines the interaction between the components you selected. After setup, it runs over and over again, in an eternal loop.
void loop()

if(menuOption == '1') {
// Disclaimer: The MicroSwitch is in testing and/or doesn't have code, therefore it may be buggy. Please be kind and report any bugs you may find.
//See our Code Contribution git repository at https://github.com/circuitcomponents/circuitcomponents-Micro_Switch
}
}

if (millis() - time0 > timeout)
{
menuOption = menu();
}
}

```

2.15 Loop

Program Arduino

The "if-then" structure of the fundamental logic in Arduino programming may be broken down into four blocks:

Setup - Usually found in the setup section of the Arduino code, this function carries out one-time tasks such sensor calibration.

Read the inputs at the beginning of the loop. The ambient light reading from an LDR using analogRead will be one example of how these values are used as criteria ("if").

Data Manipulation: This part is used to change the data's format or do calculations. For instance, the AnalogRead() function returns a reading between 0 and 1023, which can be mapped to a range between 0 and 255 for PWM purposes. Observe analogWrite()

Output: Using the information calculated in the preceding step, this part defines the final result of the logic ("then"). Only activate an LED when the ambient light level falls below a predetermined threshold, keeping in mind our LDR and PWM example. This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

Library Organization in Arduino Code

A folder containing C++ (.cpp) code files and C++ (.h) header files is called a library.

The library's structure is described in the.h file, which also declares all of its variables and functions.

The function implementation is stored in the.cpp file.

Bringing in Libraries

The first step is to choose the library you want to utilize from the numerous that are offered online. We simply need to launch Arduino IDE and select Sketch > Include Library > Manage Libraries after downloading it to your PC. The library that we want to import into the IDE can then be chosen. The library will be available in the sketch menu once the process is finished.

Instead of including the external libraries stated earlier in the circuito.io code, we provide them access to the firmware subdirectory. When using #include in this situation, the IDE knows where to look for them.

Hardware to Software Transition

There is plenty to be said about the software capabilities of the Arduino platform, but it's crucial to keep in mind that the platform is made up of both software and hardware. Together, they operate a sophisticated operating system.

Code → Compile → Upload → Run

The capability to compile and run the code lies at the heart of Arduino.

We must upload the code to the Arduino after writing it in the IDE. If the code passed compilation, clicking the Upload button (the right-facing arrow icon) will compile it and upload it. Your upload will be finished, and then the program will launch itself.

We can also carry out the following actions sequentially:

1. First, compile the code. Simply click the check icon to accomplish this, or select Sketch > Verify / Compile from the menu bar.



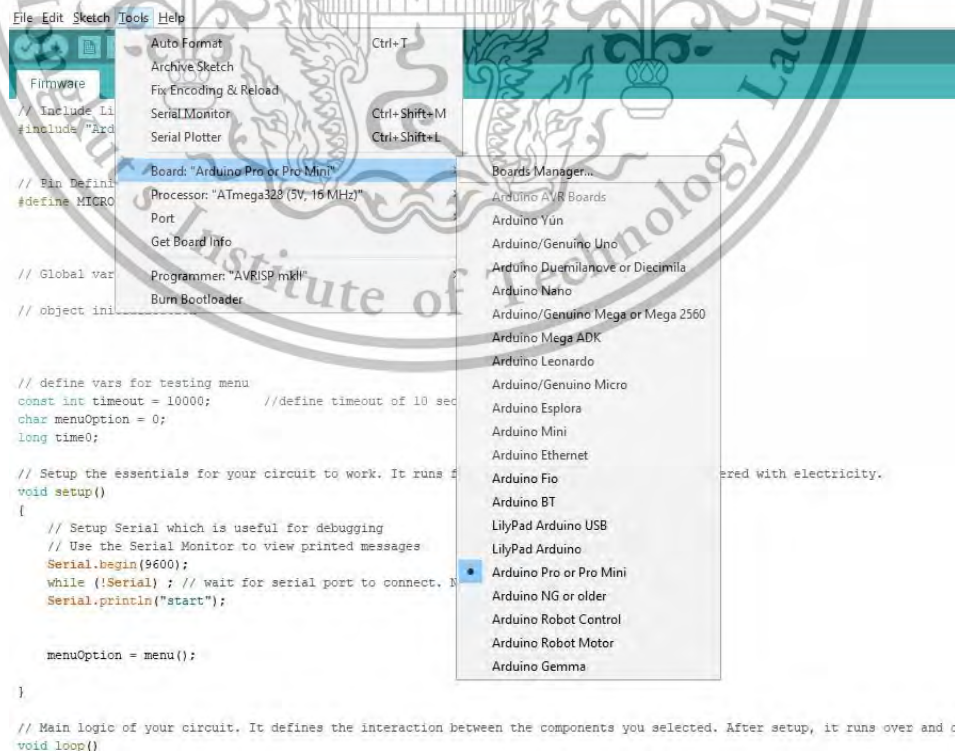
1. Choose the serial port your Arduino is currently connected to. To do this, select your preferred serial port by clicking Tools > Serial port in the menu (as shown earlier above). After that, you can upload the assembled sketch.
2. Click the upload icon next to the tick to submit the sketch. As an alternative, you can select File> Upload from the menu. Once the data is being sent, your Arduino LEDES will start to flicker.

Configuring Your IDE

A USB cable is required to connect an Arduino board to your computer. When transferring data from one board to another, especially when utilizing USB: The arduino is powered via a USB wire. You can also use an external power source to power your Arduino.

Before you can upload the code, there are various options that you need to configure.

Select your board: You must decide which Arduino board you will employ. Click Tools > Board > Your Board to accomplish this.

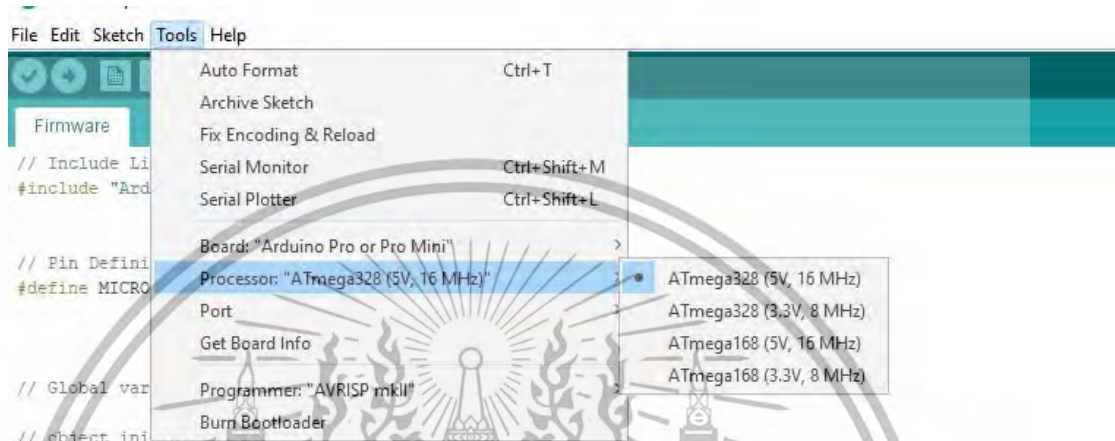


2.18 Select port

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

Select your processor carefully because you may need to indicate what processor model you have for some boards (such as the Arduino Pro Mini). The model you have should be selected under tools > processor.



2.19 ATmega328

Choose your port: Go to tools > Port > COMX Arduino (this is the serial port for Arduino) to select the port to which your board is attached.

Install Non-Native Boards (e.g. NodeMCU)

We must install some board types in the Arduino IDE before you can upload code because they are not pre-installed.

We must: in order to install a non-native board like NodeMCU:

1. To access the Boards Manager, select Tools > Boards.
2. Use the search bar to choose the board you wish to add, then click "install."

Through the Board Manager, some boards cannot be located. In this situation, you'll need to add them manually. To accomplish this:

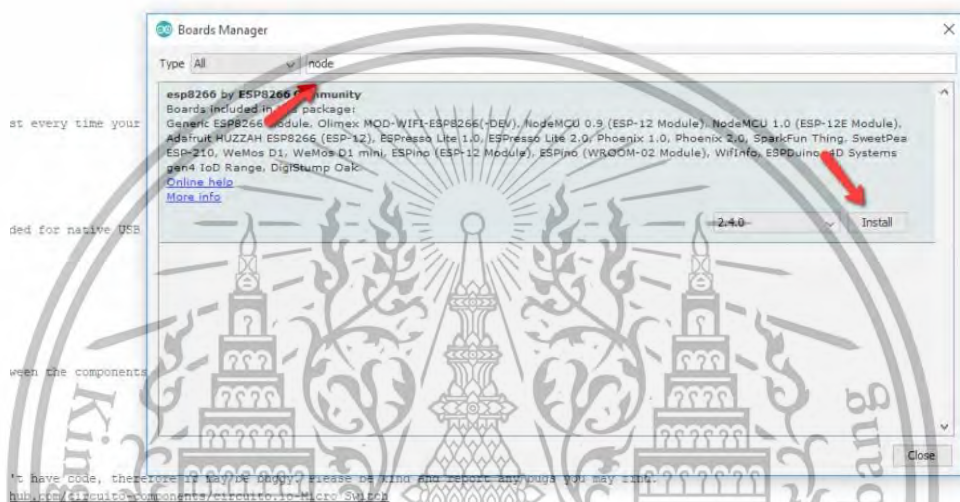
1. Select "File" > "Preferences."
2. Paste the URL of your board's installation package into the Additional Boards Manager area. Add the following URL, for instance, for nodeMCU:

http://arduino.esp8266.com/stable/package_esp8266com_index.json

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

3. Click OK.
4. select "tools" > "boards" > "boards manager"
5. Use the search bar to choose the board you wish to add, then click "install."



2.20 Board Manager

After completing this step, the installed boards will appear in the boards list under tools.

2.3 Solar cell

The most plentiful source of energy is solar energy, which may even be used under cloudy conditions. The rate at which the Earth absorbs solar energy is around 10,000 times higher than the rate at which people use energy.

For a wide range of applications, solar systems can provide heat, cooling, natural lighting, power, and fuels. Solar technologies can use photovoltaic panels or solar radiation-concentrating mirrors to turn sunlight into electrical energy.

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

Although not all nations have the same access to solar energy, direct solar energy can nevertheless make a major contribution to any nation's energy mix.

Solar cells, also termed photovoltaic cells, transform the energy of light into electrical energy via the photovoltaic effect. The majority of these are silicon cells, which range from amorphous silicon cells (non-crystalline) to polycrystalline and monocrystalline (single crystal) silicon types and have varying conversion efficiencies and prices.

Solar systems don't use chemical reactions and don't need fuel, in contrast to batteries. Additionally, unlike electric generators, solar cells don't have any moving parts. Domestic solar systems generate power from about 20% of the sunshine they receive, although more expensive commercial solar systems can generate up to 40%. However, it is anticipated that as technology progresses, solar panel efficiency will increase.

Solar farms can be built using the largest solar cell configuration, known as an array, which is made up of thousands of individual cells and can convert sunshine into electricity for extensive commercial, industrial, and domestic use.



2.21 Solar cell panel

Solar cell panels, or more often, solar panels, are smaller collections of cells. The many varieties of solar panels can be used for a range of purposes, including providing electricity in places where traditional sources are not available or are prohibitively

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

expensive to install, as well as replacing or enhancing home electrical supplies. Due to the lack of moving parts or fuels, solar panels are also frequently employed in space, notably for satellites and space stations, however diffusion of the Sun's energy in the distant portions of the solar system mean that these panels cannot be used to launch probes out to interstellar space.

The smallest solar cells can be found in toys, calculators, and radios, among other consumer goods. In addition to sunlight, these solar cells may also generate energy using artificial light.

A quick overview of light's history

It is known that the Ancient Greeks were the first to conduct research on the nature of light. During this time, philosophers such as Plato, Socrates, and Pythagoras all contributed their ideas and theories on the topic, with scientists from the medieval times coming along later to work on theories of light and vision.

We didn't fully comprehend the nature of light, though, until Christian Huygens devised a method for demonstrating how light flows in waves and where light waves spread, in 1678. It was not seen as sufficient proof at the time, but subsequent studies in the 19th century by Thomas Young (1801) and James Clerk Maxwell (1865) led to the understanding of what a beam of light is today—a traveling wave of electric and magnetic fields.

Learning about the photoelectric effect

In order to test James Clerk Maxwell's theory, Heinrich Hertz designed some experiments using a spark gap generator in 1887. The earliest electromagnetic wave transmission and reception are credited to these experiments.

In the experiment, sparks from a transmitter that produced sparks between two tiny metal spheres caused sparks to leap between two brass knobs in a copper wire loop that served as a receiver. Then, a very small spark abruptly leaped between these two electrodes. Hertz realized that he could intensify the receiver spark by shining UV light on the electrodes. This was the photoelectric effect's first recorded observation.

Although Hertz did not explain the observed phenomena (the photoelectric effect), his observations were corroborated by Wilhelm Hallwachs roughly one year later. He

This material is reserved for educational use only, not allowed for commercial use.

demonstrated how photoelectric current was produced when UV light was shone on an evacuated quartz bulb with two zinc plates serving as electrodes and linked to a battery (electron emission).

This is further supported by Stoletov's law, sometimes known as the "first law of photoeffect," which states that the generated photoelectric current is proportional to the strength of electromagnetic radiation acting on a metallic surface.

2.3.1 Electrons and the theory of Einstein

JJ Thompson made the discovery of electrons, or "corpuscles" as they were known at the time. Then, in 1899, he demonstrated that the increased sensitivity in Hertz's experiments was caused by light pushing on corpuscles and postulated a structure for the atom. Philipp Lenard also demonstrated in 1899 that UV light irradiation of metals may result in the production of photoelectrons. He discovered that these photoelectrons' kinetic energy was unrelated to the brightness of light with the same frequency. Despite this, he did concur that a bright source would emit more photoelectrons than a dim one.

A model of the atom as a positively charged core (nucleus) with electrons (negative charges) circling at various distances around it was put forth 13 years later by Thompson's pupil, Rutherford. This atom's nuclear structure is still recognized today. As a result of his contributions, Rutherford has come to be known as "the father of nuclear physics."

In his attempt to explain the photoelectric phenomenon, Einstein revived Isaac Newton's century-old theory of light corpuscles. Max Planck's theories that light is composed of energy bundles and that a light beam contains hundreds of "quanta" were also taken into consideration. He realized that Planck's theory actually held true: light is a stream of discrete quanta, not a continuous wave of electromagnetic energy. Einstein's study resulted in a crucial formula for quantum physics known as the Planck-Einstein relation:

$$E\bar{e} = \frac{1}{2} \cdot m \cdot v^2 = E_y - W = h \cdot f - W$$

Where $E\bar{e}$ is the work function, a constant that depends on the metal, E is the energy of the light quantum, v is the speed of the light quantum, m is the mass of the electron, and E is the energy of the electron.

W is the amount of power needed to liberate one electron from a metal in order to create photoelectrons. The metal, its crystal structure, and how polished the surface are all factors that affect how many photoelectrons are released. According to Einstein's theory, kinetic energy is transferred from electrons to W when light strikes a metal. Electrons are discharged at zero velocity when the energy input matches the work done (W). This explains Hertz's observation that more sparks were generated at increasing UV light intensity.

Einstein won the Nobel Prize in 1921 for his efforts and "services to Theoretical Physics, and particularly for his discovery of the law of the photoelectric effect."

Although Gilbert Lewis' work from 1926 suggested that we should regard a new form of atom, the photon, as the carrier of light instead of the light quantum, we now commonly refer to these as "photons" despite the fact that Einstein's work is still sound science even today. The term "photon" is seen to be a good substitute for Einstein's study on the quantum nature of light.



2.22 JJ Thompson

This material is reserved for educational use only, not allowed for commercial use.

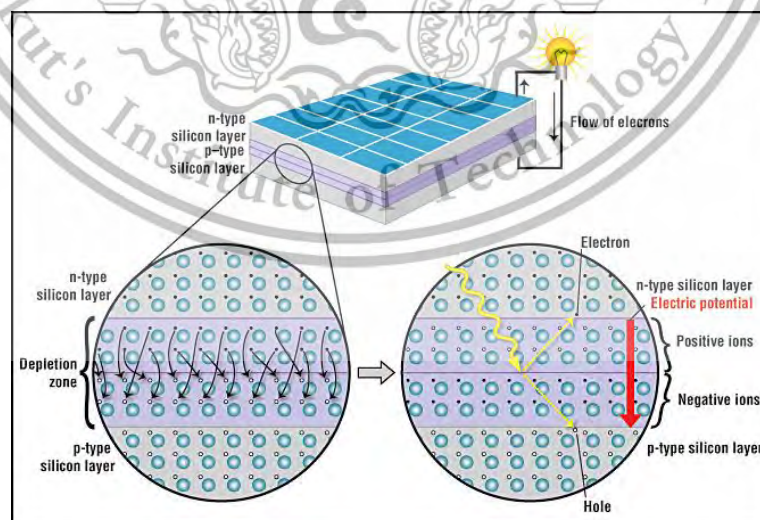
Forbidden to modify the content, and cite the document when use.

2.3.2 Solar cell principle

P-type and n-type silicon are two different types of semiconductors that are used to make solar cells. Atoms with one fewer electron in their outer energy level than silicon, like boron or gallium, are added to create p-type silicon. An electron vacancy, sometimes known as a "hole," is generated in boron because it has one fewer electron than is necessary to make the bonds with the neighboring silicon atoms.

By adding elements, like phosphorus, that have one extra electron in their outer level than silicon, the n-type silicon is created. The outer energy level of phosphorus contains five electrons, not four. It connects with its silicon neighbor atoms, however one electron is not involved in bonding. It can instead move around freely inside the silicon framework.

A layer of p-type silicon is sandwiched between a layer of n-type silicon to form a solar cell (Fig. 1). There are too many electrons in the n-type layer and too many positively charged holes in the p-type layer (which are vacancies due to the lack of valence electrons). The electrons on one side of the junction (n-type layer) migrate into the holes on the opposite side of the junction close to the junction of the two layers (p-type layer). Due to this, the electrons fill the holes in the depletion zone that forms surrounding the connection (Fig. 1, closeup).



2.22 schematic representation of a solar cell, showing the n-type and p-type layers, with a close-up view of the depletion zone around the junction between the n-type and p-type layers.

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

The p-type side of the depletion zone, where holes were initially present, is now filled with negatively charged ions, while the n-type side of the depletion zone, where electrons were previously present, is now filled with positively charged ions. These ions opposite charges provide an internal electric field that inhibits the n-type layer's electrons from filling the p-type layer's holes.

When sunlight strikes a solar cell, silicon's electrons are evacuated, which causes "holes" to form—the voids the outgoing electrons left behind. The electric field will transport holes to the p-type layer and electrons to the n-type layer if this occurs. Electrons will move from the n-type layer to the p-type layer by crossing the depletion zone and then go through the external wire back of the n-type layer, causing an electrical flow, if the n-type and p-type layers are connected by a metallic wire.

2.4 The economics of solar power

In many locations, solar energy is now even more affordable than coal or other fossil fuels thanks to recent drops in price. The number of solar installations is rapidly rising globally, thanks to hefty tax credits and subsidies. We discuss a few of the solar energy's economic factors below.

2.4.1 Understanding Solar Power's Economics

Even with the enormous advancements in technology, fossil fuels have not yet been replaced by renewable energy sources. Governments have imposed tax credits for solar and wind energy, which, until recently, were significantly more expensive than the status quo, in order to encourage the adoption of renewable energy sources.

However, the direct costs of solar and wind energy for consumers have reduced as a result of increased production, government subsidies, and growing environmental concerns. In some markets, renewable energy is actually produced more affordably for customers than fossil fuels. Solar energy offers both commercial and household applications, in contrast to wind energy, which is typically used in wind farms.

2.4.2 Affordability of Fossil Fuels

The depletion of fossil fuels is predicted to occur in fewer than 100 years, with oil by 2052, gas by 2060, and coal by 2090, according to several predictions.

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

The availability of coal, natural gas, and crude oil has been declining, but usage of these fuels has not.

Fossil fuels outperform nuclear and renewable energy among all energy sources. About 85% of all energy consumed in 2019 came from fossil fuels, up from 80% in 2014.

Fossil fuels are not only nonrenewable, but they also have a number of negative environmental repercussions. The main source of anthropogenic CO₂ that contributes considerably to climate change is the burning of fossil fuels.

Global warming, Arctic ice loss, rising sea levels, and subpar food yields are notable repercussions.

Increasing Financial Costs

Even though the United States spends over \$1 trillion year on fossil fuels, the negative consequences of burning them continue to increase the cost of living. In fact, the U.S. spent \$649 billion on fossil fuel subsidies alone in 2015.

According to research, sickness and deaths caused by air pollution cost the European Union's economy \$1.6 trillion annually.

The estimated true cost of fossil fuels, when accounting for healthcare costs, environmental damage, and other factors, is \$5.2 trillion year.

Cost of Solar Energy

Although it only makes up a small portion of all energy used, the United States is the world's largest user of renewable energy. However, solar energy still only makes up 2.3% of the total energy used in the United States, despite the fact that solar energy availability has increased over the past ten years. When it comes to preferred renewable energy sources, hydropower and wind are ahead of solar power, which accounts for 11.5% of all renewable energy consumption in the United States in 2019.

Currently, only two forms of solar technology exist that are capable of converting the sun's energy into a source of power: solar thermal and photovoltaic. Solar thermal collectors take in solar energy to warm a building or water. Sunlight is used by photovoltaic technology to supplement or replace the electricity generated by the utility grid.

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

Use of solar energy

Solar energy systems were previously only available to the wealthy or the fervent. However, the availability of solar paneling systems is expanding due to dramatically falling costs. The typical American solar system cost \$10 per watt in the early 2000s.

According to a 2017 report by the International Renewable Energy Agency (IRENA) titled "Renewable power generating prices in 2017," the price of solar photovoltaic (PV) had decreased to \$0.10 per kWh.

By 2020, the U.S. Office of Energy Efficiency & Renewable Energy aimed to make solar electricity competitively priced with traditionally produced electricity, free of subsidies. Utility-scale photovoltaic (PV) solar power costs had decreased to \$0.06 per kilowatt-hour as of 2017. (kWh). Residential and commercial solar cost targets were now \$0.16 and \$0.11 per kWh, respectively. (In comparison, in 2017, power produced by fossil fuels typically ran from \$0.05 to \$0.17 per kWh).

As a result, both residential and commercial buildings in the U.S. now have significantly more solar systems installed. Solar energy capacity increased from 0.34 gigawatts in 2008 to 97.2 gigawatts in 2021.

A Worldwide Rise

Global consumption of solar energy has increased as more nations become aware of the negative effects of burning fossil fuels. The cost of installation has dramatically decreased as a result of increased competition within the solar power sector.

Solar energy is now being used in many of the world's top economies, including those of the United States, China, India, and several European countries. China has spearheaded the world's largest drive toward renewable energy and erected a significant amount of photovoltaics in an effort to reduce pollution.

India, which suffers from severe pollution, has set a goal to generate 175 GW of renewable energy by 2022.

Over the next five years, it is anticipated that the United States' capacity for solar photovoltaic installations will more than double.

Big Companies

Large corporations are also making investments in reusable solar systems. Some stores, offices, and facilities at Walmart (WMT), Verizon (VZ), and Apple (AAPL) have already switched to solar power.

In the fall of 2019, Google acquired 1,600 megawatts from 18 different suppliers in the largest solar procurement transaction in history.

Solar energy still makes up a small portion of the total energy supply, but both the household and commercial sectors are gradually embracing renewable energy. Solar energy systems are anticipated to become increasingly widespread as costs continue to drop. In Europe, it's anticipated that the cost per kilowatt-hour will drop to between 4 and 6 cents in 2025 and even lower to as little as 2 cents in 2050.

Tax credits for solar energy

Even though solar energy systems are now more affordable, the government continues to provide incentives for both home and business use. Solar energy consumers' tax obligations are reduced in the US thanks to the Renewable Energy Tax Credit. Depending on date the property was put into operation, a taxpayer can deduct 30%, 26%, or 22% of eligible expenses for systems that serve an occupied space. Both wind and geothermal energy systems qualify for the same credit from the US government.

To make renewable energy systems more appealing, many European nations apply a feed-in-tariff system. Under a feed-in-tariff arrangement, renewable energy system owners can earn money from the government. Prices are determined per kilowatt-hour (kWh), with differences between nations.

Is solar energy cost-effective?

Depending on the amount of sunlight and weather where the installation is taking place, solar energy can be less expensive than fossil fuels. According to some estimates, solar energy is currently the least expensive energy source in the United States, with costs as low as \$0.70 per watt.

What Are the Drawbacks of Solar Energy?

Different minerals that must be mined from the soil are used to make solar panels. Like the extraction of coal and copper, this technique can harm the local ecosystem. Additionally, when a panel reaches the end of its useful life, it becomes e-waste. Additionally, not all locations are appropriate for solar electricity because some have more consistent exposure than others.

How Much Does It Cost to Install Solar Power in a House?

Unless you were ready to go without electricity at night, it would be challenging to power a home entirely using solar energy. It costs between \$16,000 and \$35,000, plus installation fees, according to some estimates. The local energy market will determine whether doing so will be profitable.

The conclusion

The commitment to renewable resources has primarily come from private citizens, large corporations, and nations. Companies like Google (GOOG) and Amazon (AMZN) have committed to utilize wind energy in addition to solar energy to power their facilities. With huge organizations, individuals, and governments continuing to move to renewable energy sources, unfavorable environmental effects from burning fossil fuels can hopefully be minimized.

Compete without risk for \$100,000 in virtual money.

Utilize our FREE Stock Simulator to put your trading abilities to the test. Trade your way to the top while competing against hundreds of other Investopedia traders! Before risking your personal money, place trades in a virtual environment. Practice your trading techniques so that you will have the necessary experience when it comes time to enter the real market.

2.5 Chapter Summary

In Chapter 2 we proposed the Solar energy, Arduino, Solar cell and the economics of solar power

Then we will discuss how to use solar tracking to improve your perception. And by using the aforementioned idea together with a variety of tools to make this work, you can learn more about how to build solar tracking. You will be given detailed instructions on how to complete each step of the project, including the coding, in order to make it as efficient as possible.



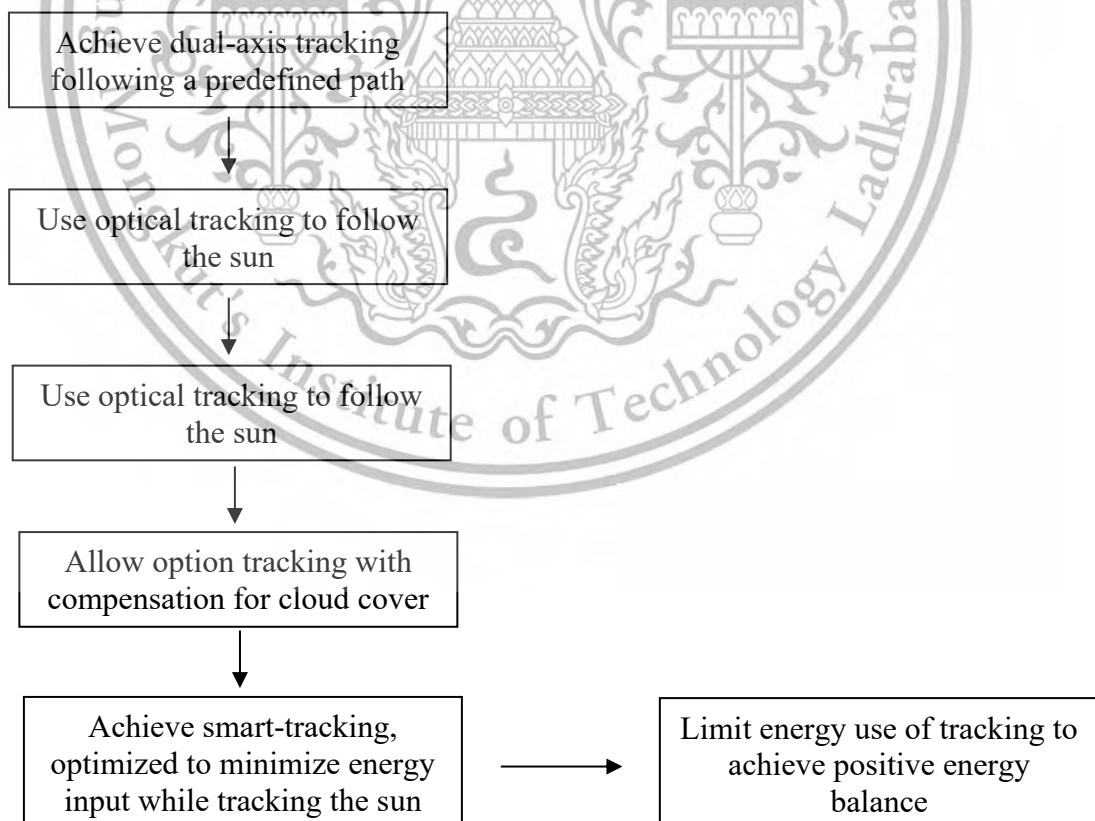
CHAPTER 3

METHODOLOGY

3.1 Introduction

We will demonstrate how to design solar tracking in this project, together with solar tracking tools and solar tracking code. to provide information to arduino that will enable solar tracking to function and provide us with a theory about solar refraction so that our solar tracking will deflect in the direction of receiving solar energy and receive energy effectively. should utilize it as inspiration to build a more effective and superior solar tracking. to safeguard backup power not just in the event of a domestic or commercial power failure

To make it simpler to understand and implement, we will start by constructing a block diagram algorithm.



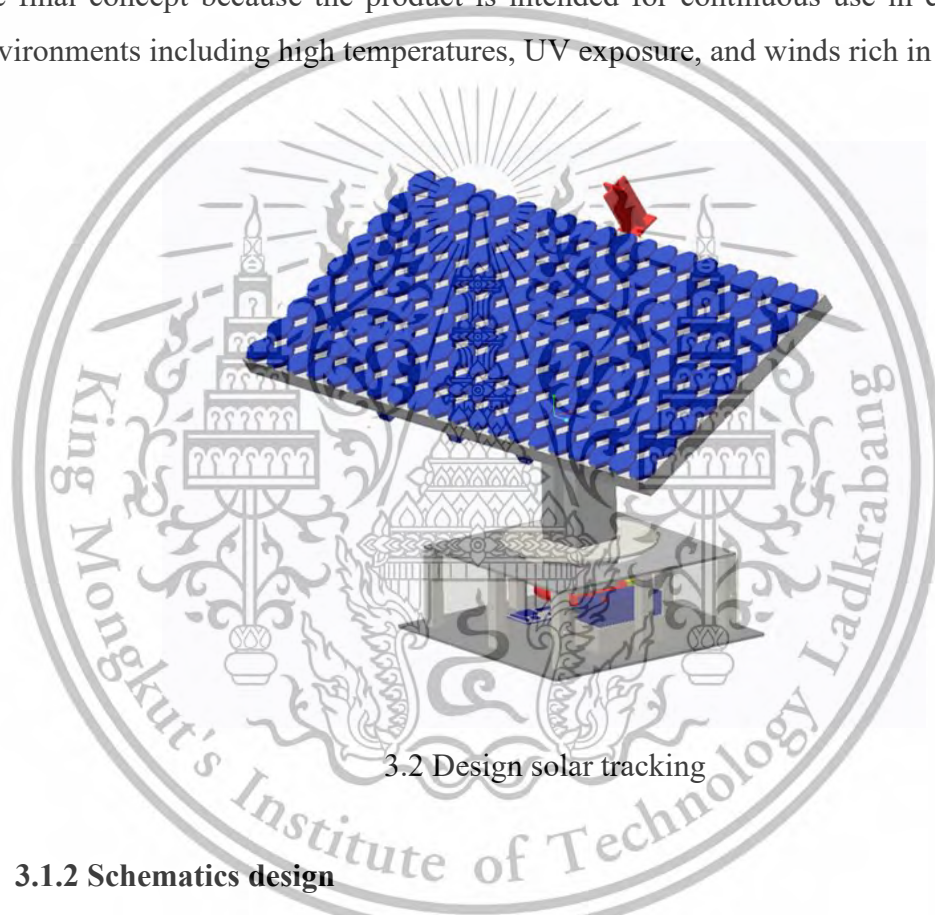
3.1 Block diagram Solar tracking

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

3.1.1 Hardware design

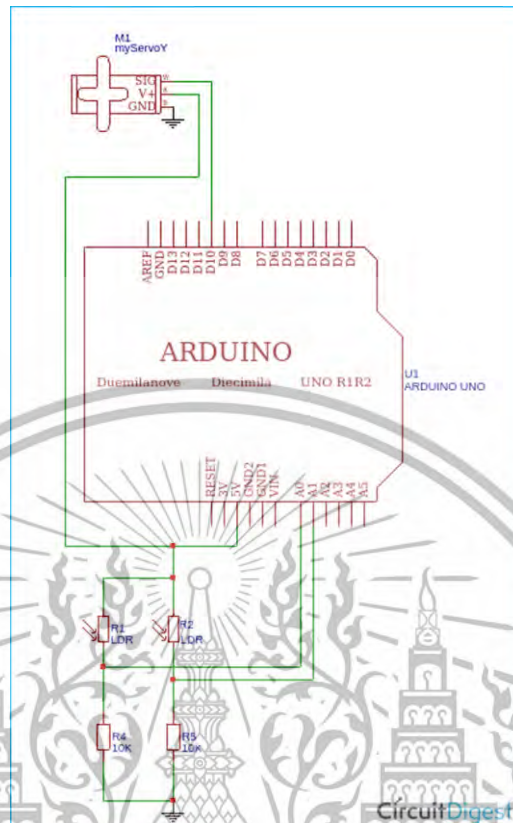
We kept the idea of separate azimuth- and zenith-tracking systems in our final design proposal. To reduce the number of moving parts, we made the design streamlined and straightforward. Durability and reliability were key considerations in the final concept because the product is intended for continuous use in challenging environments including high temperatures, UV exposure, and winds rich in dust.



3.2 Design solar tracking

3.1.2 Schematics design

The circuit's connection is really simple. Here, I connected the two LDRs to analogue pins A0 and A1 on an Arduino Uno that I was using as the controller. The servo motor is attached to Arduino's pin 9. Since we utilized a 5V servomotor, we don't need an extra power source because the Arduino can easily power all of the components. The diagram below shows every connection.



3.3 Schematics design

3.2 Equipment

3.2.1 Building the base is the first thing to do before assembling our solar tracker. I'm planning to utilize an MDF board to construct the base. The MDF board must first be cut into rectangular pieces of 12*8 cm and 12*2 cm, as illustrated in the figure.

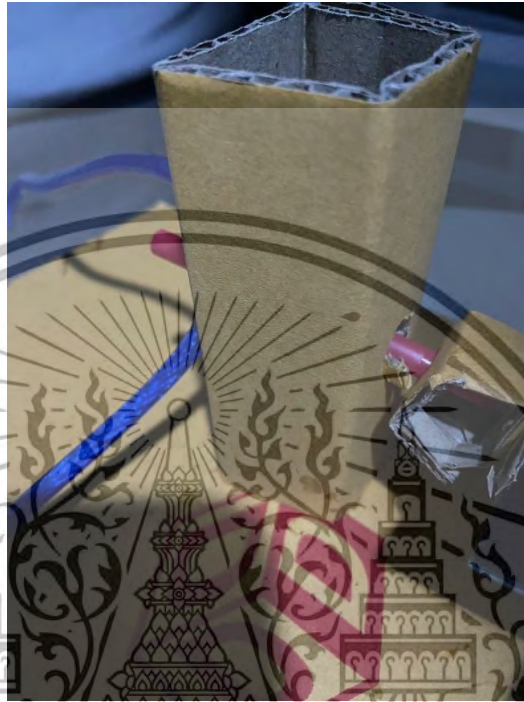


This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

3.4 MDF board

Then, as shown in the figure, adhere the 12*2cm piece vertically to the 12*8cm piece.



3.4 Stick MDF board

3.2.2 LDR

The most common uses for photoresistors, often referred to as light dependent resistors (LDR), are to detect the presence or absence of light or to gauge the strength of the light. When the LDR sensor is exposed to light, the resistance reduces rapidly, perhaps even to a few ohms, depending on the light intensity. In the dark, their resistance is very high, often reaching 1 M. LDRs are nonlinear devices having variable sensitivity depending on the wavelength of applied light. Although they are utilized in many applications, other gadgets like photodiodes and phototransistors frequently carry out this light sensing function. LDRs constructed of lead or cadmium have been outlawed in some nations due to environmental safety concerns.

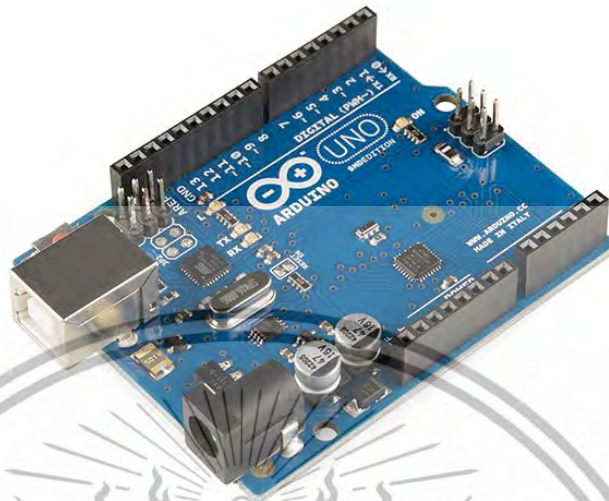


3.5 LDR Sensor

3.2.3 Arduino Uno

The finest board for learning electronics and coding is the Arduino UNO. The UNO is the most durable board you may start playing with if this is your first time dabbling with the platform. The Arduino family's UNO board is the most popular and well-documented model.

A microcontroller board called Arduino UNO is based on the ATmega328P. It contains 6 analog inputs, a 16 MHz ceramic resonator, 14 digital input/output pins (six of which can be used as PWM outputs), a USB port, a power jack, an ICSP header, and a reset button. It comes with everything needed to support the microcontroller; to get started, just plug in a USB cable, an AC-to-DC adapter, or a battery. You can experiment with your UNO without being overly concerned that you'll make a mistake; in the worst case, you can replace the chip for a few dollars and start over.

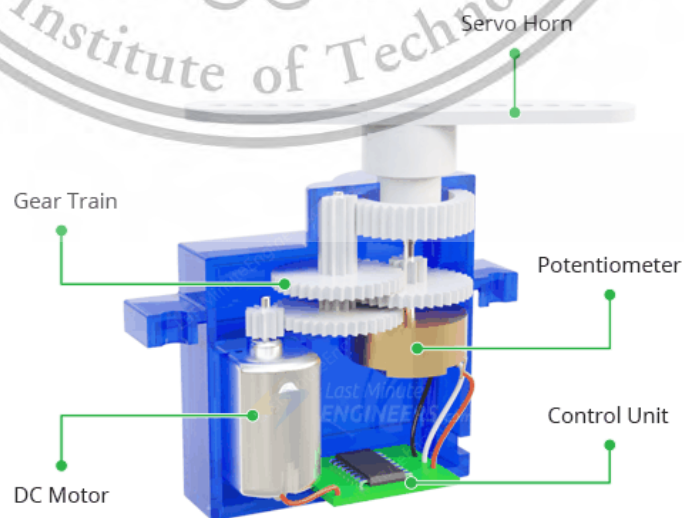


3.5 Arduino Uno

3.2.4 Servo motor

Because they typically move to a location rather than rotating constantly, servo motors let you accurately regulate physical movement. Because the motor driver is already present in them, connecting and controlling them is straightforward.

A small DC motor is built into servos and is gear-connected to the output shaft. The output shaft is connected to a potentiometer and operates a servo horn (pot).



3.6 Servo motor

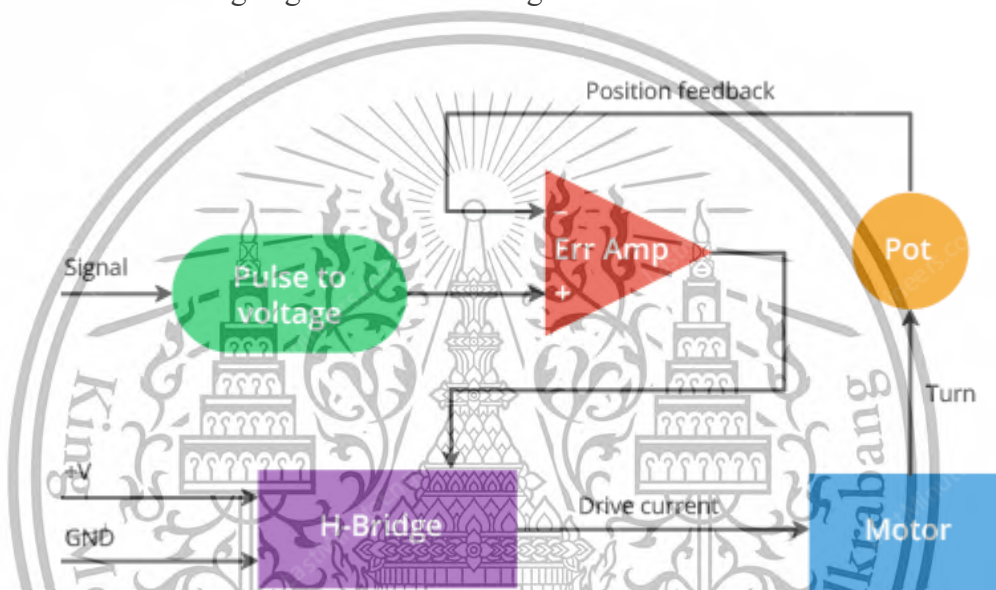
This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

The error amplifier in the control unit compares the motor's current position to the desired position using position feedback from the potentiometer.

The control unit corrects the inaccuracy by moving the motor's present position toward the target position.

This device is referred to as a servomechanism, or servo, in control engineering. It is a closed-loop control system that makes adjustments to the speed and direction of the motor using negative feedback to get the desired outcome.

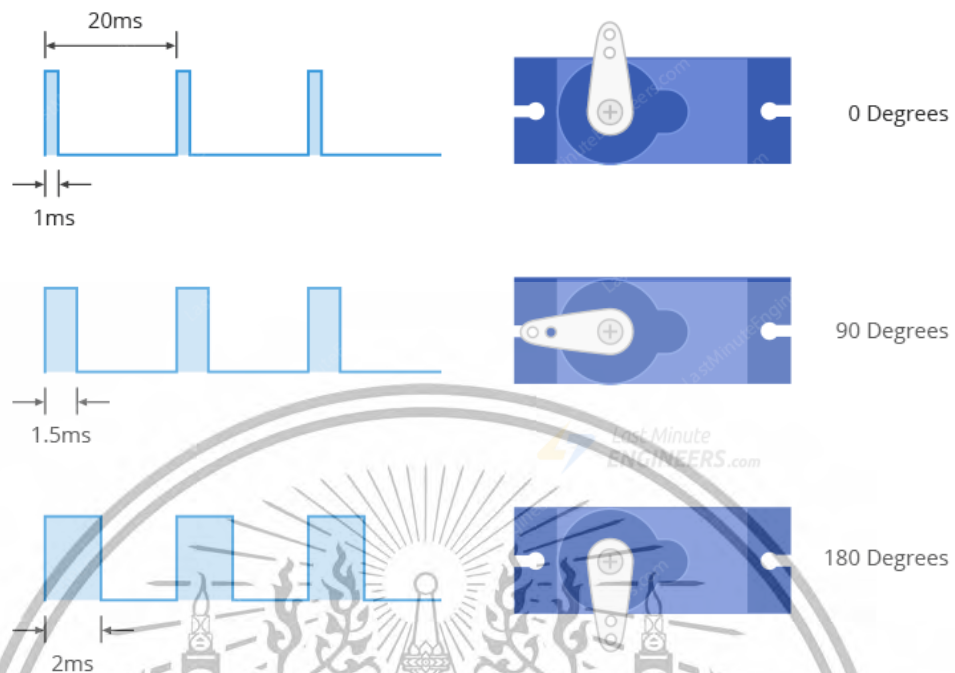


3.7 Processing servo motor

3.2.4.1 The Function of Servo Motors

Sending the servo motor a series of pulses allows you to control it. Every 20 milliseconds, a typical servo motor expects to receive a pulse (i.e., the signal should be 50Hz).

The servo motor's position is determined by the pulse's length.



3.8 Length of pulse

1. The servo will rotate to 0 degrees with a brief pulse of 1 ms or less (one extreme).
2. The servo will rotate 90 degrees with a 1.5 ms pulse (middle position).
3. The servo will rotate to 180 degrees with an approximately 2-ms pulse (other extreme).

The servo will spin to a position proportional to the pulse width for pulse widths between 1 and 2 milliseconds. You can better grasp the connection between pulses and position by watching the animation below.

3.2.4 Resistor

A resistor is an electrical component that controls or restricts how much electrical current can pass across a circuit in an electronic device. A specified voltage can be supplied via resistors to an active device like a transistor.

In a direct-current (DC) circuit, the current through a resistor is directly proportional to the voltage across it and inversely proportional to its resistance, all other things being equal. The well-known Ohm's Law is this. This criterion also

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

holds true for alternating-current (AC) circuits as long as the resistor has no inductance or capacitance.

3.2.4.1 How to read resistor color code?

Once you comprehend the logic and arithmetic behind each band used to represent resistance value, tolerance, and occasionally even the temperature coefficient, reading resistor color codes becomes simple. We've made a straightforward chart to help you understand resistor color coding.

Resistors are available in many different values, shapes, and physical sizes. Practically all leaded resistors with a power rating up to one watt have a pattern of colored bands that are used to indicate resistance value, tolerance, and sometimes even the temperature coefficient. There can be anywhere from three to six colored bands on the body of a resistor, with four bands being the most common variation. The first few bands always represent digits in the value of resistance. Then you will find a multiplier band to signify moving the decimal right or left. The last bands represent tolerance and the temperature coefficient.

How to Read Resistor Color Codes

6-Band $274 \cdot 10^2 \pm 2\%$ = 274 Ω \pm 2%, 250 ppm/K

Color	1st Digit	2nd Digit	3rd Digit	Multiplier	Tolerance	Temperature Coefficient
Black	0	0	0	1 Ω		250 ppm/K
Brown	1	1	1	10 Ω	\pm 1%	100 ppm/K
Red	2	2	2	100 Ω	\pm 2%	50 ppm/K
Orange	3	3	3	1k Ω		15 ppm/K
Yellow	4	4	4	10k Ω		10 ppm/K
Green	5	5	5	100k Ω	\pm 0.5%	20 ppm/K
Blue	6	6	6	1M Ω	\pm 0.25%	10 ppm/K
Violet	7	7	7		\pm 0.1%	5 ppm/K
Grey	8	8	8			1 ppm/K
White	9	9	9			
Gold				0.1 Ω	\pm 5%	
Silver				0.01 Ω	\pm 10%	

4-Band $12 \times 10^2 \pm 5\%$ = 1,200 k Ω \pm 5%

5-Band $100 \times 10^2 \pm 1\%$ = 10,000 Ω \pm 1%

3.9 Resistor color code

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

3.3 Coding to Arduino

The bottom of this page contains a link to the project's complete source code. Downloading the Servo Library is the first thing to do before writing any code. To manage the servo's motion, we require a servo library.

1. First, I included the servo library and created a servo object and named it as 'servo'.
2. I've declared the variables for sensor values and allocated the analogue pins A0 and A1 pins for LDR.
3. If you are utilizing the exact same LDRs on both sides, you can leave this value at 0 for system calibration. But if you are using different LDRs then you should use this to calibrate. Follow the directions in the next paragraph to calibrate.
4. Place a light source directly above the solar panel and serially print the sensor data to confirm each sensor's reading at noon. If the reading reveals the same values, you can keep this as is; if it reveals any differences, you must duplicate those values into this location.
5. This variable is to store the servo position.
6. I have defined the servo pin as pin 9

```
#include <Servo.h>

Servo servo;

int eastLDR = 0;
int westLDR = 1;

int east = 0;

int west = 0;

int error = 0;

int calibration = 600;

int servoposition = 90;

void setup()
```

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

7. The first step in the loop section is to read the LDR values using the Arduino's analog read function and save them in the east and west variables.
8. If any of the LDRs registers a low value, the solar panel will turn back to the east side, according to this if condition.

```

void loop()
{
    east = calibration + analogRead(eastLDR);
    west = analogRead(westLDR);
    if (east < 350 && west < 350)
    {
        while (servoposition <= 150)
        {
            servoposition++;
            servo.write(servoposition);
            delay(100);
        }
    }
}

```

9. Here, we compute the difference between the readings taken to the east and west. East has more intensity if the error value is positive, and west has more intensity if the error value is negative. So, based on this error number, we can turn the servo to the side with lower intensity.
10. The east side has higher intensity if the error is positive and greater than 30. As a result, the system will initially examine the servo's starting position and rotate to the east if it is less than 150 degrees. These angles can be modified to fit your system.

```

error = east - west;
if (error > 30)
{
    if (servoposition <= 150)
    {
        servoposition++;
        servo.write(servoposition);
    }
}

```

```

else if (error < -30)
{
    if (servoposition > 20)
    {
        servoposition--;
        servo.write(servoposition);
    }
}

```

11. The servo spins to the west side if the error is negative and smaller than -30, indicating that the west side is more intense.

```

else if (error < -30)
{
    if (servoposition > 20)
    {
        servoposition--;
        servo.write(servoposition);
    }
}

```

3.4 Problems in method

the aforementioned no matter how flawless our programming is, hardware issues like soldering wires can still occur. If we are not careful, the heat from the solder can harm the LDR sensor. Our LDR sensor has malfunctioned as a result of improper soldering, which prevents the solar tracking from tracking the light intensity and causes the servo motor to be pointed in the wrong direction.

3.5 Proposed Solution

soldering needs to be done carefully because it could harm our processor. This will have an impact on our electrical circuit and lead to an erroneous outcome since solar tracking requires numerical information from sensor LDR in order to force servo motor to rotate in the direction of the brightest light intensity. so that we will obtain the desired outcome.

3.6 Summary

Overall, using the technique we just described to you, solar tracking can be directed in the direction of the brightest light, giving us adequate backup power to use in an emergency. Power outage because the sensor LDR turns servo motor in the appropriate direction throughout the day by measuring the light intensity and sending a number to the software. from the emergence of the sun to dusk This enables us to make sure that our backup power is adequate to stop problems brought on by a temporary loss of electrical power.

Therefore, this model can be further developed through the development of more efficient devices and code. And also make people at home feel at ease and not worry about power failures, such as data loss prevention or accidents that occur during power failures. and eliminates the problem of old solar panels that do not receive enough solar energy.

CHAPTER 4

EXPERIMENTAL RESULT

4.1 Introduction

In order to determine whether solar tracking genuinely functions over time without error, we will examine whether solar tracking can accurately track the light intensity over time for this experiment.

4.2 Testing Summary

We first begin with the experiment that took place between 6:00 AM and 11:00 AM. You'll see that solar tracking is pointing eastward in the proper direction. due to the fact that morning light is the brightest. The servo motor rotates the solar panel in the proper direction based on data from the LDR sensor.



4.1 Time 06.00-11.00 AM

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

Second action We conducted our experiment from 12:00 to 12:59 p.m. As you can see, the solar tracking can pivot flat 180 degrees depending on the light, however our solar panel will move since the LDR sensor detects back and forth. It might have frequently gotten an inaccurate numeric value, leading the servo motor to move nearly constantly.

We can thus assume that around 12.00 PM, this may be the worst fault of project solar tracking, which results in false data and utilizes power because LDR is always moving. The servo motor rotates in the direction where the light intensity is the highest when there are two sides emitting light at once and the average light intensity is the same.

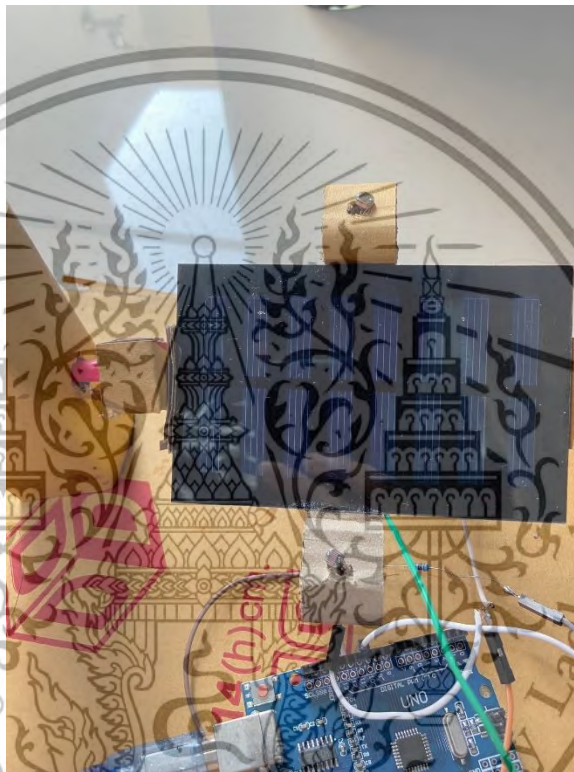


4.2 Time 12.00 PM

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

In the final phase, we tested solar tracking between 13.00 to 18.00 PM, and we can see that solar tracking rotates in the proper direction based on the amount of light detected by LDR. This causes servo motor to revolve in the proper direction before the monk. The sun will set. This guarantees that our program is flawless. Our code runs smoothly.



4.3 Time 13.00-18.00 PM

4.3 Evaluation

In this experiment, the test results were recorded in a table with 7 days for each period and an hour between seeing the test results. utilizing trustworthy sources to contrast with our solar tracking results.

Using a trial period of seven or one week, we documented the daily work of solar tracking in this table to see if it could be directed in the appropriate way or not.

Time Day	06.00 to 07.00	08.00 to 09.00	10.00 to 11.00	12.00 to 13.00	14.00 to 15.00	16.00 to 18.00
1	○	○	○	✗	○	✗
2	○	○	○	○	✗	○
3	○	○	✗	○	○	○
4	○	✗	✗	○	○	○
5	○	○	○	✗	○	✗
6	○	✗	○	○	✗	✗
7	○	○	✗	○	○	○

This table was created using the daily average of which side of the variable solar tracking faces which direction. and compare objectively which direction and time of day the light is at its brightest.

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

We can infer that solar tracking may be prone to errors from the data in the table. Numerous conditions, including air humidity and foggy days, prevent the LDR sensor from detecting an effective light intensity. But because the sensor LDR is accurate, it can still be a trustworthy reference. This enables precise tracking of the maximum light source's intensity.

4.4 Testing and Evaluation Summary

We can infer from the testing that solar tracking operates effectively but might have a little inaccuracy depending on the day's weather. causing the information from sensor LDR to be incorrect and making servo motor rotate in the other direction from what we anticipated, but solar tracking can give us a general notion of where the light intensity is. so that we can fully utilize solar energy and extend its use as an alternative or backup energy that is essential to lessen the effects of failures in home electrical systems. It could result in data loss or an accident

Additionally, we have conducted tests at various locations, including our homes and student residences 30 km away. It is evident from these tests that the solar tracking is still effective and reliably measures light intensity, indicating that solar tracking ours is moving in the right path. However, it also depends on the day's weather. Because it will have an impact on how the LDR sensor operates and could produce an inaccurate reading of the data.



This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

CHAPTER 5

CONCLUSION

5.1 Summary

From our studies and research for this solar tracking project experiment, we know that a variety of factors, including the environment and solder-damaged sensors, can alter how well a sensor measures light intensity. This could result in an inaccurate measurement of light intensity by our LDR sensor, spinning the servo motor in the wrong direction or the wrong way. Pressing the reset button is sometimes necessary since the model solar tracking won't work or use coding to identify different variables to use as a point of reference and learn more about measuring light intensity. Determine the orientation of servo motor so that it will rotate in the appropriate direction when the sensor identifies the side with the highest light intensity.

Going deeper and being able to improve it even more than before is the value of this endeavor, either enhancing the program's functionality or the resources utilized in its creation in order to lower household costs and guarantee there is enough backup power to handle power outages, in order to avoid the negative effects that will result.

5.2 Key Points

This project also has a number of safety measures, like the day's weather, which could lead to our solar tracking malfunctioning because false readings from the LDR sensor would cause our servo motor to spin in the wrong direction or move constantly. This could prevent us from obtaining energy efficiently, but additionally, our solar tracking is typically in a normal climate, can function effectively and make the appropriate turn.

REFERENCES

1. Solar Tracking
<https://circuitdigest.com/microcontroller-projects/building-your-own-sun-tracking-solar-panel-using-arduino>
2. Solar cell electronics
<https://www.britannica.com/technology/solar-cell>
3. OLEDSOLAR Project
<https://www.twi-global.com/technical-knowledge/faqs/what-are-solar-cells>
4. ENERGY.GOV RESOURCES
<https://www.energy.gov/eere/solar/solar-photovoltaic-cell-basics>
5. Paolo Ciambelli, ... Luca Vitale, in Studies in Surface Science and Catalysis, 2020
<https://www.sciencedirect.com/topics/chemistry/solar-cell>
6. S.C. Bhatia, in Advanced Renewable Energy Systems, 2014
7. Tetsuo Soga, in Nanostructured Materials for Solar Energy Conversion, 2006
8. Otwin Breitenstein, in Semiconductors and Semimetals, 2013
9. Muhammad Bilal Tahir, ... Ghulam Nabi, in Chemistry of Nanomaterials, 2020
10. Shantonu Roy, Soumya Pandit, in Microbial Electrochemical Technology, 2019
<https://www.sciencedirect.com/topics/chemistry/solar-cell>
11. Tetsuo Soga, in Nanostructured Materials for Solar Energy Conversion, 2006
12. Yotsarayuth Seekaew, ... Chatchawal Wongchoosuk, in Carbon-Based Nanofillers and Their Rubber Nanocomposites, 2019
13. Michael T. Taschuk, ... Michael J. Brett, in Handbook of Deposition Technologies for Films and Coatings (Third Edition), 2010
14. Umesh A. Fegade, Ganesh N. Jethave, in Interface Science and Technology, 2021

REFERENCES

15. Solar cell structure
<https://www.pveducation.org/pvcdrom/solar-cell-operation/solar-cell-structure>
16. Energy Education
https://energyeducation.ca/encyclopedia/Photovoltaic_cell
17. Solar Energy
<https://www.britannica.com/science/solar-energy>
18. National geographic
<https://education.nationalgeographic.org/resource/solar-energy>
19. How solar is use?
<https://www.seia.org/initiatives/about-solar-energy>
20. Transfer energy
<https://www.twi-global.com/technical-knowledge/faqs/what-is-solar-energy>
21. How does solar work?
<https://www.energy.gov/eere/solar/how-does-solar-work>
22. Just energy
<https://justenergy.com/blog/what-is-solar-energy/>
23. PV
<https://www.irena.org/Energy-Transition/Technology/Solar-energy>
24. Full cell
https://www.fuelcellstore.com/blog-section/renewable-energy-rundown-solar-energy?gclid=CjwKCAiAy_CcBhBeEiwAcoMRHE1xceLBCt1Bo7xNY7ZNlnjwUI-je0MtRZJnx6GSeJGAeCwIHAKw4xoCwQcQAvD_BwE
25. Solar portal
<https://energypedia.info/wiki/Portal:Solar>

REFERENCES

26. The journey
https://humanjourney.us/economics-renewable-energy-us/?gclid=CjwKCAiAy_CcBhBeEiwAcoMRHOxdL0ulKs6NKbndzOP8PSXxXJo_h5R3O50IpNF_FGOkYNblg05aQRoC_BkQAvD_BwE
27. What is Arduino?
https://www.dfrobot.com/blog-588.html?gclid=CjwKCAiAy_CcBhBeEiwAcoMRHDigiw6_j0lfS-jrP5CNBnCddk7LqahhRQwjlodVcb9FRThOrm6ziRoCo-sQAvD_BwE
28. Arduino spark fun
<https://learn.sparkfun.com/tutorials/what-is-an-arduino/all>
29. Open source
<https://opensource.com/resources/what-arduino>
30. How Arduino work?
<https://all3dp.com/2/what-is-arduino-simply-explained/>
31. Beginner to understand Arduino
<https://hackr.io/blog/what-is-arduino>
32. Meaning of Arduino
<https://www.techopedia.com/definition/27874/arduino>
33. What can you do with Arduino?
<https://www.makeuseof.com/tag/arduino-technology-explained/>
34. Overview of Arduino
<https://www.seeedstudio.com/blog/2019/12/04/introduction-to-the-arduino-what-is-arduino/>
35. How to read color resistor code?
<https://www.arrow.com/en/research-and-events/articles/resistor-color-code>
36. Improve the skill
<https://eepower.com/resistor-guide/resistor-standards-and-codes/resistor-color-code/>
37. Resistor color code
<https://www.allaboutcircuits.com/tools/resistor-color-code-calculator/>

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

REFERENCES

38. Everything you need to know about Arduino code
<https://www.circuito.io/blog/arduino-code/>
39. Arduino Coding Basics
<https://www.javatpoint.com/arduino-coding-basics>
40. C++ get start
https://www.w3schools.com/cpp/cpp_getstarted.asp
41. Learn C++
<https://www.learncpp.com/>
42. Code structure
<https://javascript.info/structure>
43. 5 Lessons I've Learned on How to Structure Code
<https://levelup.gitconnected.com/5-lessons-ive-learned-on-how-to-structure-code-6d662df0fd1f>
44. Programing structure
<https://www.cs.utah.edu/~germain/PPS/Topics/structures.html>
45. What are Photoresistors?
<https://eepower.com/resistor-guide/resistor-types/photo-resistor/>
46. Light dependent resistor
<https://www.watelectronics.com/light-dependent-resistor-ldr-with-applications/>
47. Arduino servo motor
<https://www.instructables.com/Arduino-Servo-Motors/>
48. Servo motor
https://www.tutorialspoint.com/arduino/arduino_servo_motor.htm
49. Uno R3
<https://docs.arduino.cc/hardware/uno-rev3>
50. What is Arduino UNO? | Arduino Uno Pinout & Program
<https://www.rs-online.com/designspark/what-is-arduino-uno-a-getting-started-guide>

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.



This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.



This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

```
#include <Servo.h>

Servo servo ;

int eastLDR = 0;

int westLDR = 1;

int east = 0;

int west = 0;

int error = 0;

int calibration = 600;

int servoposition = 90;

void setup()
{
  servo.attach(9);
}

void loop()
{
  east = calibration + analogRead(eastLDR);
  west = analogRead(westLDR);
  if (east < 350 && west < 350)
  {
    while (servoposition <= 150)
    {
      servoposition++;

      servo.write(servoposition);

      delay(100);
```

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

```
    }  
  }  
  
  error = east - west;  
  
  if (error > 30)  
  {  
    if (servoposition <= 150)  
    {  
      servoposition++;  
      servo.write(servoposition);  
    }  
  }  
  else if (error < -30)  
  {  
    if (servoposition > 20)  
    {  
      servoposition--;  
      servo.write(servoposition);  
    }  
  }  
  }  
  
  delay(100);  
}
```