

IMPLEMENTATION AND DEVELOPMENT OF A AUTOMATIC TICKETING AND BILL PAYMENT SYSTEM FOR METRO RAIL

Supervised by :

Submitted by :

Shahinur Rahman	Sajib Sarkar	ID: BME1803016530
Lecturer	Zobaer Ahammad Sajib	ID: BME1803016491
Department of Mechanical Engineering	Md. Al-Mamun	ID: BME1803016518
	Safikul Islam	ID: BME1803016507
	KOWSAR	ID: BME1803016512

A report submitted to the Department of Mechanical, Sonargaon University of Bangladesh in partial fulfillment of the requirements for the Award of Degree of Bachelor of Science in Mechanical Engineering.



Sonargaon University (SU)
Dhaka-1215, Bangladesh

August, 2022

ACKNOWLEDGEMENT

First, we started in the name of almighty Allah. This thesis is accomplished under the supervision of **Shahinur Rahman**, Lecturer, Department of Mechanical, Sonargaon University. It is a great pleasure to acknowledge our profound gratitude and respect to our supervisor for this consistent guidance, encouragement, helpful suggestion, constructive criticism and endless patience through the progress of this work. The successful completion of this thesis would not have been possible without his persistent motivation and continuous guidance.

The authors are also grateful to **Prof. Md. Mostofa Hossain**, Head of the Department of Mechanical Engineering and all respect teachers of the Mechanical Engineering Department for their co-operation and significant help for completing the thesis work successfully.

‘Authors’

ABSTRACT

Railway transport remain some of the most vital modes of travel all over the world. Therefore, all their operative techniques have to be simplified for deriving the best results. . The main concept of implementing such system will make the ticketing payment process makes much easier. These type of tags are used for the verification process and user can be used to avoid the fool proof respectively. Here, lot of passengers are using railways and they need to be somewhat more enriched in securities. These RFID tags are much more easier to scan and they are cheap and reusable too. These kind of technologies increases more revenues to the government and makes the ticketing process as much as easier as well. Here the way of using the RFID is much more different from all the usage among the real time examples the RFID were scanned and they check using the program which was dumped into it. By scanning the RFID tag makes much more initial process and they are safer to use when compare to bar code and other scanning type of process like QR code respectively.

TABLE OF CONTENTS

Acknowledgement	i
Abstract	ii
Table of Contents	iii
Table of Figure	v
CHAPTER-1	INTRODUCTION
1.1 Introduction	1
1.2 Background Study	1
1.3 Objective	2
1.4 Structure of the Projects	2
CHAPTER 2	LITERATURE REVIEW
2.1 Introduction	4
2.2 Literature Review	4
2.3 Summary	6
CHAPTER-3 HARDWARE AND SOFTWARE ANALYSIS	
3.1 Arduino Pro Mini	7
3.2 SMPS	10
3.3 Voltage Regulator	15
3.4 Servo Motor	15
3.5 LCD Display	18
3.6 RFID Module	20
3.7 Resistor	22
3.8 Arduino IDE	23
3.14 Proteus Software	25

CHAPTER-4	METHODOLOGY	
4.1 Methodology		27
4.2 Block Diagram		27
4.3 Schematic Diagram		28
4.4 Working Principle		28
4.5 Our Final Project View		29
CHAPTER-5	RESULT AND DISCUSSION	
5.1 Discussion		30
5.2 Result		31
5.3 Performance Test		31
5.4 Advantage		43
5.5 Application		43
CHAPTER-6	CONCLUSION	
6.1 Conclusion		44
6.2 Future Scope		44
Reference		45

TABLE OF FIGURES

FIGURE NO	FIGURE NAME	PAGE NO
3.1	Arduino Pro Mini	8
3.2	Arduino Pro Mini Pin Out	10
3.3	Micro Controller IC AT Mega	10
3.4	SMPS	11
3.5	Basic Working Concept of an SMPS	13
3.6	SMPS Diagram	14
3.7	DC Power Supply Way	15
3.8	5V Regulator IC	16
3.9	Servo Motor	17
3.10	Schematic Diagram of Servo Motor	18
3.11	LCD Display	19
3.12	Pin Out of LCD Display	19
3.13	Pin Description	20
3.14	LCD Display Interface	20
3.15	How RFID Works	21
3.16	RFID Module	22
3.17	Two Common Resistor	23
3.18	Arduino Software Interface IDE	25
3.19	Proteus Software Interface	26
4.1	Block Diagram	28
4.2	Schematic Diagram	29
4.3	Our Final System Overview	30

CHAPTER 1

INTRODUCTION

1.1 Introduction

RFID is also known as the frequency identification device and which was in the form of tags these are the upcoming technologies. RFID projects and these technologies were used and they are much more efficient too. These type of RFID tags are used to identify the radio waves automatically and in stores some kind of values into it. Generally, the RFID tag consists of the antenna which is inbuilt in the tag and which used as the reader the chip as well as possible respectively. The RFID may consists of two main parts like the reader part and the tag. The RFID tag and reader transfers the message through electro magnetic waves and they have prior modules. This RFID system uses with the technology of embedded systems and they provide such an automated ticketing system in the railways. These type of RFID project has been implemented and they have utilised as well as possible. This system makes the installation of these RFID circuits everywhere in the metro trains as well as possible. These Kind of reader circuit make the calculations much more easier and they simplify as well as the ticket charges with some valid time on it. Since these kind of systems are much more loved by the passengers respectively.

1.2 Background Study

As for the RFID application, it's been a widespread tool for both tracking the transit transports and for the public ticketing system. It's already been an outstanding achievement throughout the globe including big cities like London, Helsinki, Shanghai, Istanbul, Moscow, Porto and many more. The system can be implemented for subways, railways and public bus services for the sake of systematic operations in corresponding cases.

In the mega city Dhaka, the conventional system of public transport is based on paper based bus or railway tickets that ultimately lead to chaos among public, system loss, corruption and most of all traffic jam that is responsible for a huge wastage of time. No prior notification of the arrival and departure of the transports are available creating a lot of confusion among the passengers resulting in a rough argument between them and the bus supervisors or the operators. Again having no government authority to take control or

keep an eye over the whole scenario, the private sectors are creating a monopoly, taking control over the public transport and autocratic raise in bus fair.

The tracking and ticketing systems using RFID can be merged to solve the prevailing problems. Even though the GPS based system can be designed, we propose the RFID based tickets for its low cost, easy operation, portability, durability, reliability and being much more user friendly. Also the high speed RFID tags and detectors make the tracking system of a running bus merely a child's play. Public carrying RFID based electronic tickets will have access to any bus service of the city only entering his current location and his destination on the keypad attached to every bus. The data will directly be transferred to the server main database and the equivalent credit will be stored in the corresponding bus account. Also the screen at every bus stop will notify the passengers, the departure time of the last bus of any route. This automated system will save time, have a higher authoritative inspection and reduce chaos and confusion on the road.

1.3 Objective

We have some specific objectives for this project and they are pointed below:

- To **Design, Construction and Performance Test of Metro Rail Gate Control System.**
- To implement Automatic Bill Payment System on Metro Rail.
- To study the system performance for future reference and improvement purposes.

1.4 Structure of the Project

This Project is organized as follows:

Chapter 1 Introduction : The first chapter contains the statement of the introduction, our background study for the project, objectives of the study, methodology used in the project and the project outline.

Chapter 2 Background and Motivation: The chapter two contains our literature review part.

Chapter 3 Methodology: Chapter three describes the theoretical model. Here we mainly discuss about proposed system architecture in details with having block diagram, circuit diagram, project working principle, complete project image, project instrument cost analysis and discuss about Hardware and software development of our projectetc.

Chapter 4 Performance Evaluation : Chapter four deals with the result and discussion and discuss about our project advantages and application.

Chapter 5 Conclusion : Chapter five all about our project conclusion, limitations and future scope.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

In this section topics related to **Design, Construction and Performance Test of Metro Rail Gate Control System** are included. These provide a sampling of problems appropriate for application of **Metro Rail Gate Control System**. The references are summarized below.

2.2 Literature Review

The authors of proposed a system where cloud can be used for storage and retrieving of the data in the IoT environment. They also mentioned that when the IoT system used in a real-time system then the cloud will be very much efficient in terms of availability. Cloud gives the easy integration, quick and easy to process complex data, with an added advantage of low installation cost. In our proposed project, we used the cloud for remotely accessing the data from anywhere, and also for easy processing. [1]

Coming to the data transfer from and to the cloud, the authors of face two types of threats in cloud and IoT environments. One is data at rest and another is data in transit. Data in rest means when the data is in the cloud sitting idle. And data in transit means when data is in transmission from one node to another. To provide the security in both the aspects we used the two different algorithms for securing the data. [2]

Firstly we provide encryption in the Arduino itself using AES Algorithm so now the encrypted data goes to the cloud, where we process the data and encrypt again using the DES algorithm. So the data that is encrypted and secured both while sending and resting in the cloud. IoT making our life easy but as its use is increasing making us vulnerable also. [3]

Authors of suggest that the IoT is the future of our information technology as well as for electronic technology, as it deals with data, sometimes the data are very sensitive in terms of privacy and security at the time of sending can help to encrypt the data. We are using AES to encrypt data to store in cloud .we are using AES library in Arduino for encrypting the data of the fingerprint. [4]

The authors of proposed a model for the hybrid encryption model for securing the diagnostic text data of medical images. As IoT gives us to enhance any system in big scale health sector is also adopting the IoT based system, but the health system is data is fully sensitive because most of the data is personal data so. Due to security reason before sending to the cloud the data should be encrypted. For a Room Temperature Control and Fire Alarm/Suppression IoT Service using MQTT on the AWS system the authors of [5] uses the publish subscriber architecture and MQTT protocol, they used MQTT broker for connecting the IoT, Arduino is used to sending the data using Wi-Fi. [5]

Our main aim is to connect the Arduino (with Node MCU) to the AWS IoT Core in order to enable communication between them so that the devices can be controlled remotely. In Registration Device, we are going to get the fingerprint data of the new user which will be converted into a hexadecimal format. This data must be sent to the AWS IoT directly from Arduino using NodeMCU. The AWS IoT will have a trigger that gets activated when it receives a message and then this fingerprint data is stored in the backend cloud database. [6]

While in the entry/exit devices on the journey day, we have to depict whether the user is allowed to travel or not, which will be shown with the help of LED. If a Green LED glows we allow the user and if the red LED glows when the user is not allowed to travel. Here the messages will be published by the Arduino device under the topic LED while the AWS IoT will serve as the subscriber to the LED topic. The state of the LED will be changed from the cloud end itself. [7]

The authors of have proposed the intricacies of the AES algorithm and a new method has been proposed to improve the security of the algorithm by introducing the dynamic S-Box Generation and Dynamic key generation. In this approach, more complexity is added to the data to increase Confusion and diffusion in the ciphertext by using the Dynamic Key Generation and then by using this S-box to make it difficult for the attacker to do any study of the S-Box. [8]

Authors of have done a study of a new securing mechanism for low-power devices known as ECC (Elliptic Curve Cryptography). ECC is a public key encryption technique based on the elliptic curve theory that can be used to make faster and more effectual cryptographic keys. [9]

The keys are generated from the elliptic curve equation rather than the very traditional process of the generation as the product of very large prime numbers. Such a method of security is highly suitable for Arduino powered devices and also mobile devices as they provide a higher level of security compared to the RSA or DES with lower-key bits. This new method could be used for a future version of the same proposed system to make it more efficient and secure. [10]

2.3 Summary

We try to do this project by reading the above literature, and we have been able to make our project successful by reducing the mistakes of last year's project.

CHAPTER 3

HARDWARE AND SOFTWARE ANALYSIS

3.1 Arduino Pro Mini

The Arduino Pro Mini is a micro-controller board based on the ATmega168. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, an on-board resonator, a reset button, and holes for mounting pin headers. A six pin header can be connected to an FTDI cable to provide USB power and communication to the board.

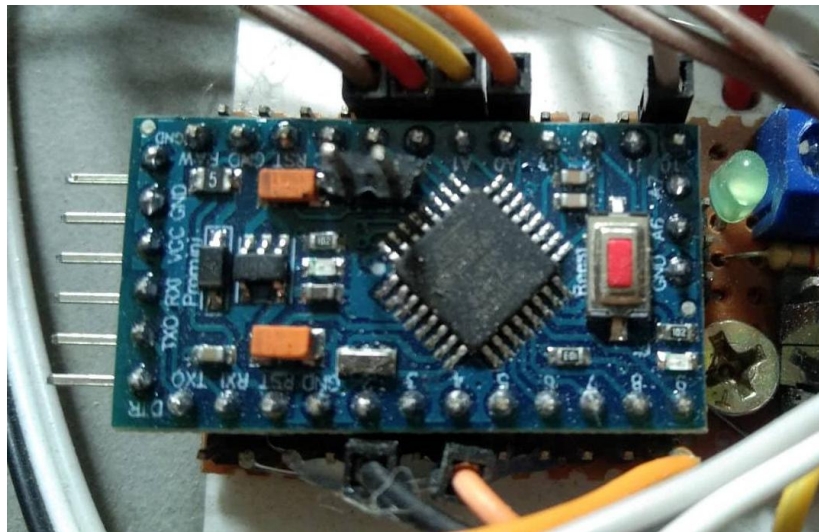


Figure 3.1: Arduino Pro Mini

Specification

- Micro-controller ATmega168
- Operating Voltage: 3.3V or 5V (depending on model)
- Input Voltage: 3.35 -12 V (3.3V model) or 5 - 12 V (5V model)
- Digital I/O Pins: 14 (of which 6 provide PWM output)
- Analog Input Pins: 6
- DC Current per I/O Pin: 40 mA
- Flash Memory: 16 KB (of which 2 KB used by bootloader)

- SRAM: 1 KB
- EEPROM: 512 bytes
- Clock Speed: 8 MHz (3.3V model) or 16 MHz (5V model)

Pin Out

Each of the 14 digital pins on the Pro Mini can be used as an input or output, using `pinMode()`, `digitalWrite()`, and `digitalRead()` functions. They operate at 3.3 or 5 volts (depending on the model). Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

- Serial: 0 (RX) and 1 (TX). Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the TX-0 and RX-1 pins of the six pin header.
- External Interrupts: 2 and 3. These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the `attachInterrupt()` function for details.
- PWM: 3, 5, 6, 9, 10, and 11. Provide 8-bit PWM output with the `analogWrite()` function.
- SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). These pins support SPI communication, which, although provided by the underlying hardware, is not currently included in the Arduino language.
- LED: 13. There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

The Pro Mini has 6 analog inputs, each of which provide 10 bits of resolution (i.e. 1024 different values). Four of them are on the headers on the edge of the board; two (inputs 4 and 5) on holes in the interior of the board. The analog inputs measure from ground to VCC. Additionally, some pins have specialized functionality:

- I2C: 4 (SDA) and 5 (SCL). Support I2C (TWI) communication using the Wire library.

There is another pin on the board:

- Reset. Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

See also the mapping between Arduino pins and ATmega168 ports.

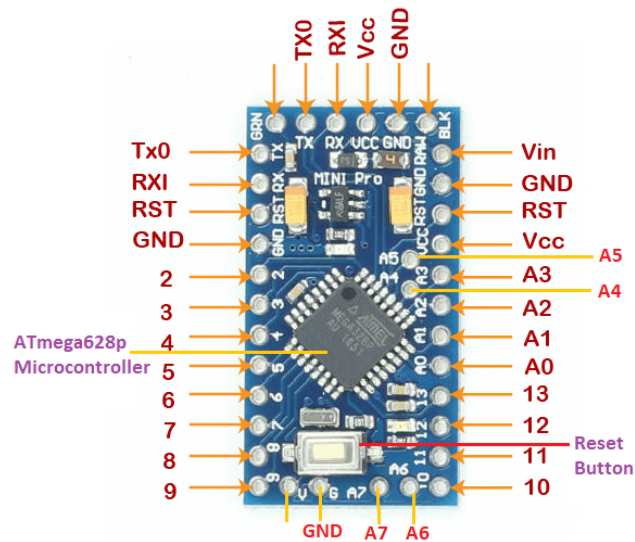


Figure 3.2: Arduino Pro Mini Pin Out

Microcontroller IC ATmega328p



Figure 3.3: Microcontroller IC AT Mega 328p

The high-performance Microchip Pico Power 8-bit AVR RISC-based microcontroller combines 32KB ISP flash memory with read-while-write capabilities, 1024B EEPROM,

2KB SRAM, 23 general purpose I/O lines, 32 general purpose working registers, three flexible timer/counters with compare modes, internal and external interrupts, serial programmable USART, a byte-oriented 2-wire serial interface, SPI serial port, a 6-channel 10-bit A/D converter (8-channels in TQFP and QFN/MLF packages), programmable watchdog timer with internal oscillator, and five software selectable power saving modes. The device operates between 1.8-5.5 volts. By executing powerful instructions in a single clock cycle, the device achieves throughputs approaching 1 MIPS per MHz, balancing power consumption and processing speed.

3.2 Switch Mode Power Supply (SMPS)

A switched-mode power supply (switching-mode power supply, switch-mode power supply, switched power supply, SMPS, or switcher) is an electronic power supply that incorporates a switching regulator to convert electrical power efficiently. Like other power supplies, an SMPS transfers power from a DC or AC source (often mains power) to DC loads, such as a personal computer, while converting voltage and current characteristics.

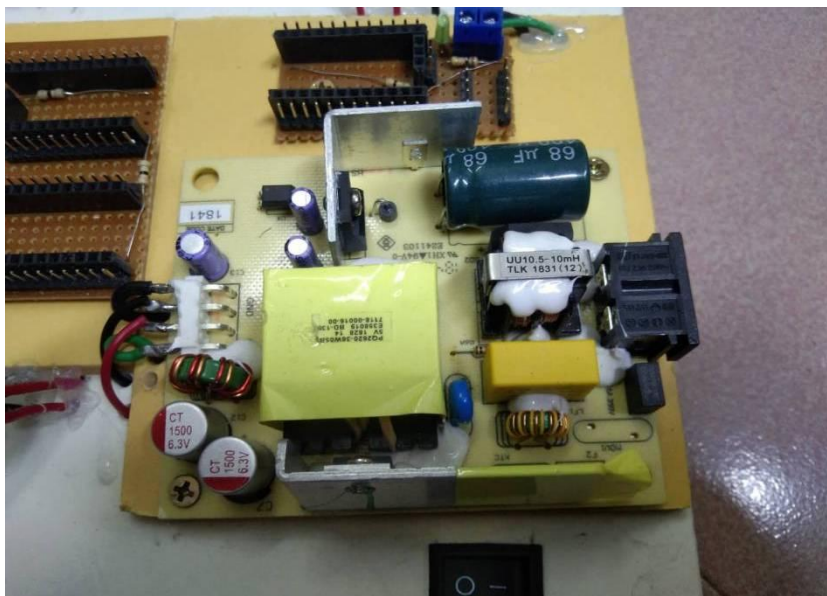


Figure 3.4 : SMPS

Unlike a linear power supply, the pass transistor of a switching-mode supply continually switches between low-dissipation, full-on and full-off states, and spends very little time in the high dissipation transitions, which minimizes wasted energy. A hypothetical ideal

switched-mode power supply dissipates no power. Voltage regulation is achieved by varying the ratio of on-to-off time (also known as duty cycles). In contrast, a linear power supply regulates the output voltage by continually dissipating power in the pass transistor. This higher power conversion efficiency is an important advantage of a switched-mode power supply. Switched-mode power supplies may also be substantially smaller and lighter than a linear supply due to the smaller transformer size and weight. Switching regulators are used as replacements for linear regulators when higher efficiency, smaller size or lighter weight are required. They are, however, more complicated; their switching currents can cause electrical noise problems if not carefully suppressed, and simple designs may have a poor power factor.

Switched-mode power supplies are classified according to the type of input and output voltages. The four major categories are:

- AC to DC
- DC to DC
- DC to AC
- AC to AC

A basic isolated AC to DC switched-mode power supply consists of:

- Input rectifier and filter
- Inverter consisting of switching devices such as MOSFETs
- Transformer
- Output rectifier and filter
- Feedback and control circuit

The input DC supply from a rectifier or battery is fed to the inverter where it is turned on and off at high frequencies of between 20 KHz and 200 KHz by the switching MOSFET or power transistors. The high-frequency voltage pulses from the inverter are fed to the transformer primary winding, and the secondary AC output is rectified and smoothed to produce the required DC voltages. A feedback circuit monitors the output voltage and instructs the control circuit to adjust the duty cycle to maintain the output at the desired level.

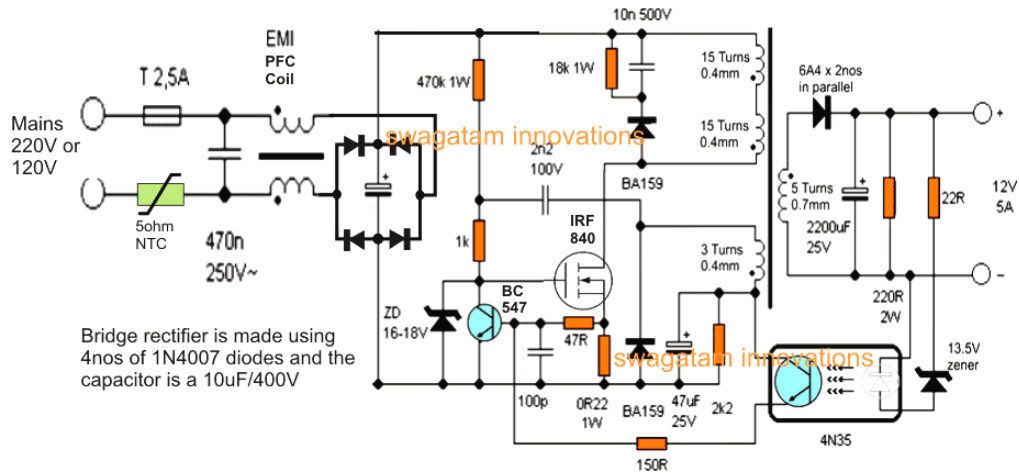


Figure 3.5 : Basic working concept of an SMPS

A switching regulator does the regulation in the SMPS. A series switching element turns the current supply to a smoothing capacitor on and off. The voltage on the capacitor controls the time the series element is turned. The continuous switching of the capacitor maintains the voltage at the required level.

Design basics

AC power first passes through fuses and a line filter. Then it is rectified by a full-wave bridge rectifier. The rectified voltage is next applied to the power factor correction (PFC) pre-regulator followed by the downstream DC-DC converter(s). Most computers and small appliances use the International Electrotechnical Commission (IEC) style input connector. As for output connectors and pinouts, except for some industries, such as PC and compact PCI, in general, they are not standardized and are left up to the manufacturer.

There are different circuit configurations known as topologies, each having unique characteristics, advantages and modes of operation, which determines how the input power is transferred to the output. Most of the commonly used topologies such as flyback, push-pull, half bridge and full bridge, consist of a transformer to provide isolation, voltage scaling, and multiple output voltages. The non-isolated configurations do not have a transformer and the power conversion is provided by the inductive energy transfer.

Advantages of switched-mode power supplies:

- Higher efficiency of 68% to 90%
- Regulated and reliable outputs regardless of variations in input supply voltage
- Small size and lighter
- Flexible technology
- High power density

Disadvantages:

- Generates electromagnetic interference
- Complex circuit design
- Expensive compared to linear supplies

Switched-mode power supplies are used to power a wide variety of equipment such as computers, sensitive electronics, battery-operated devices and other equipment requiring high efficiency.

Switch Mode Power Supply

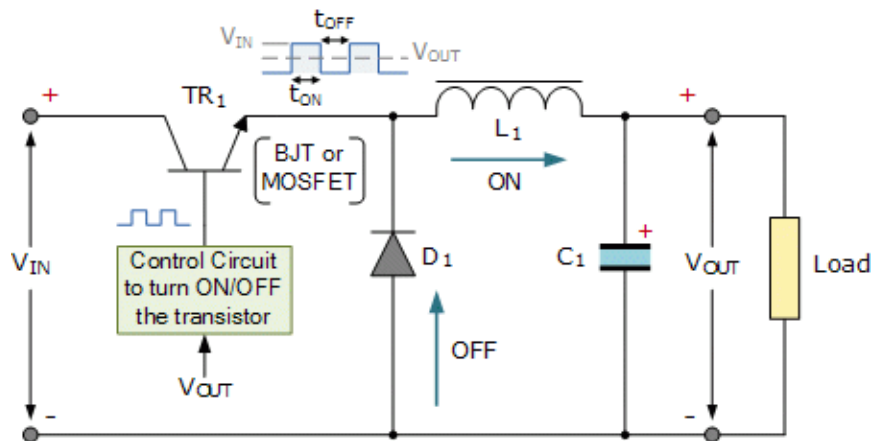


Figure 3.6 : SMPS Diagram

Linear voltage IC regulators have been the basis of power supply designs for many years as they are very good at supplying a continuous fixed voltage output. Linear voltage regulators are generally much more efficient and easier to use than equivalent voltage regulator circuits made from discrete components such as a zener diode and a resistor, or

transistors and even op-amps. The most popular linear and fixed output voltage regulator types are by far the positive output voltage series, and the negative output voltage series. These two types of complementary voltage regulators produce a precise and stable voltage output ranging from about 5 volts up to about 24 volts for use in many electronic circuits.

There is a wide range of these three-terminal fixed voltage regulators available each with its own built-in voltage regulation and current limiting circuits. This allows us to create a whole host of different power supply rails and outputs, either single or dual supply, suitable for most electronic circuits and applications. There are even variable voltage linear regulators available as well providing an output voltage which is continually variable from just above zero to a few volts below its maximum voltage output. Most d.c. power supplies comprise of a large and heavy step-down mains transformer, diode rectification, either full-wave or half-wave, a filter circuit to remove any ripple content from the rectified d.c. producing a suitably smooth d.c. voltage, and some form of voltage regulator or stabiliser circuit, either linear or switching to ensure the correct regulation of the power supplies output voltage under varying load conditions. Then a typical d.c. power supply would look something like this:

Typical DC Power Supply

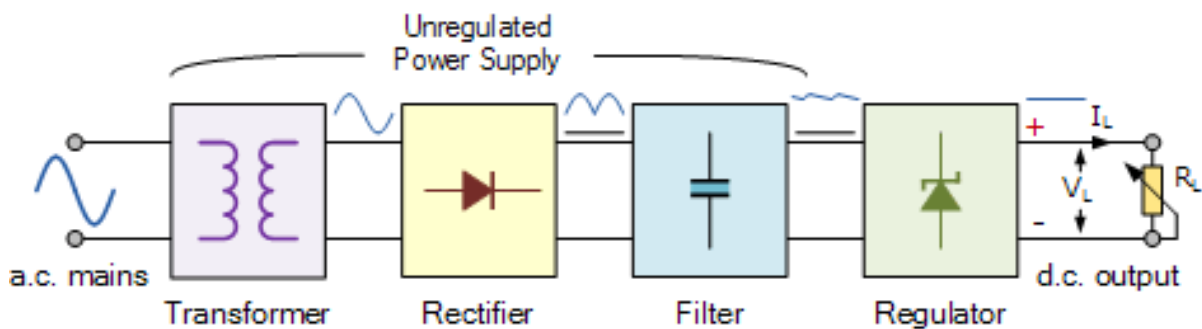


Figure 3.7: DC Power supply way

These typical power supply designs contain a large mains transformer (which also provides isolation between the input and output) and a dissipative series regulator circuit. The regulator circuit could consist of a single zener diode or a three-terminal linear series regulator to produce the required output voltage. The advantage of a linear regulator is that the power supply circuit only needs an input capacitor, output capacitor and some feedback resistors to set the output voltage.

3.3 5V Regulator IC

Voltage sources in a circuit may have fluctuations resulting in not providing fixed voltage outputs. A voltage regulator IC maintains the output voltage at a constant value. 7805 IC, a member of 78xx series of fixed linear voltage regulators used to maintain such fluctuations, is a popular voltage regulator integrated circuit (IC). The xx in 78xx indicates the output voltage it provides. 7805 IC provides +5 volts regulated power supply with provisions to add a heat sink.

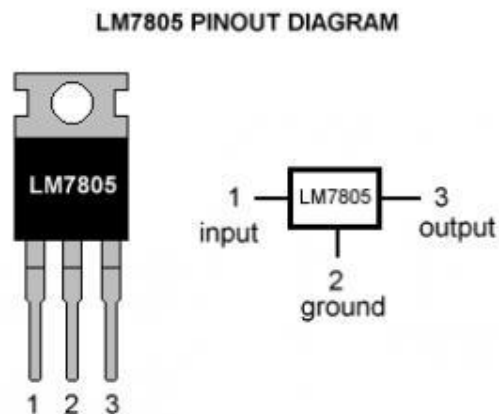


Figure 3.8: 5V Regulator IC

7805 IC Rating:

- Input voltage range 7V- 35V
- Current rating $I_c = 1A$
- Output voltage range V. Max=5.2V ,V. Min=4.8V

3.4 Servo Motor

A servo motor is an electrical device that can push or rotate an object with great precision. If you want to rotate an object at certain angles or distances, you use servo motors. It is made by a simple motor which is driven by a servo mechanism. If the motor is DC driven then it is called DC servo motor and if it is AC driven motor then it is called AC servo motor. We can get a very high torque servo motor in a small and light weight package. These are being used in various applications like toy cars, RC helicopters and planes, robotics, machines etc.



Figure 3.9: Servo Motor

Servo motors are rated at kg / cm (centimeters per kilogram). Most hobby servo motors are rated at 3 kg / cm or 6 kg / cm or 12 kg / cm. This kg / centimeter tells you how much weight your servo motor can lift at a certain distance. For example: A 6 kg / cm servo motor should be able to lift 6 kg if the load is suspended 1 cm away from the motor shaft, to be less than the carrying capacity. The position of the servo motor is determined by the electric pulse and its circuitry is placed next to the motor.

Servo Mechanism

It consists of three parts:

1. Controlled device
2. Output sensor
3. Feedback system

All motors have three wires. Of which two will be used for the supply (positive and negative) and one will be used for the signal transmitted from the MCU. The servo motor is controlled by PWM (Pulse with Modulation) which provides the control wires. Has minimum pulse, maximum pulse and repetition rate. The servo motor can rotate 90 degrees in two directions, creating its neutral position. The servo motor expects to see a pulse every 20 milliseconds (ms) and the pulse length will determine how far the motor rotates. For example, a 1.5 mm pulse will turn the motor to 90 °, for example, if the branch is less than 1.5 mm shaft, it will move to 0 to, and if it is longer than 1.5 mm, the servo will turn to 180 to.

The servo motor operates on the PWM (pulse width modulation) principle, meaning that the angle of rotation is controlled by the duration of the pulse applied to its control pin. Basically, the servo motor is made up of a DC motor which is controlled by a variable resistor (potentiometer) and some gears. The high-speed force of the DC motor is converted to torque by the gears. We know that work = force x DISTANCE, less in DC motor force and more in distance (speed) and in servo, forces are higher and distance is less. The potentiometer server is connected to the output trench to calculate the angle and stop the DC motor at the required angle.

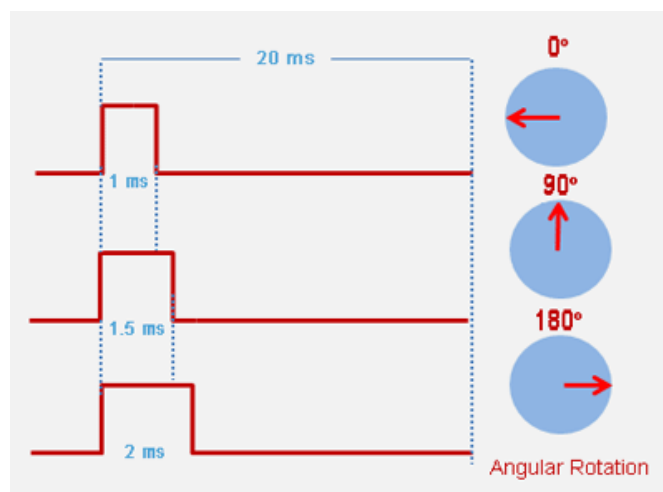


Figure 3.10: Schema Diagram of servo motor

The servo motor can be rotated from 0 to 180 degrees but depending on the output it can go up to 210 degrees. This degree of rotation can be controlled by applying an electric pulse of appropriate width to its control pin. Servo examines pulses every 20 milliseconds. The 1 ms (1 millisecond) width pulse servo can rotate 0 degrees, 1.5 mm 90 degrees (neutral position) and 2 ms pulse it can rotate 180 degrees. All servo motors work directly with your + 5V supply rails but we need to be careful about how much current the motor will use. If you plan to use more than two servo motors, you should design a suitable servo solder.

3.5 LCD Display

The LCD (liquid crystal display) screen is an electronic display module and looks for various applications. A 16x2 LCD display is a very basic module and it is very commonly used in various devices and circuits. These modules are preferred over seven sections and many other segments.



Figure 3.11: 16*2 LCD Display

The reasons for having LCD are economic; Easily programmable, special and even custom characters (different in seven sections), there are no restrictions on displaying animations. A 16x2 LCD means it can display 16 characters per line and contains 2 lines. Each character on this LCD is displayed in a 5x7 pixel matrix.

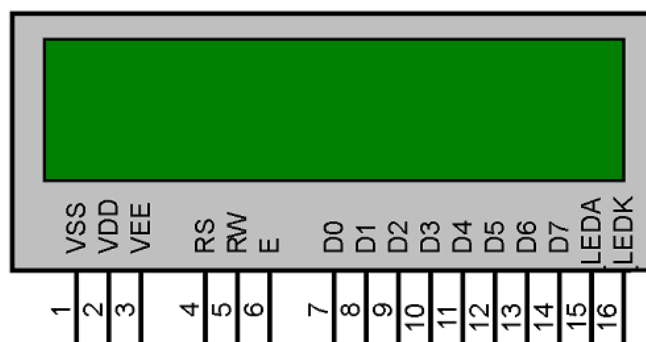


Figure 3.12: Pin out of 16*2 LCD Display

Pin No	Symbol	Level	Description
1	V _{SS}	0 V	Ground
2	V _{DD}	5.0V	Supply voltage for logic
3	V _o	variable	Operating voltagefor LCD
4	RS	H/L	H data/L instruction code
5	R/W	H/L	H/Read (MPU-module) L/write (MPU-module)
6	E	H, H-L	Chip enable signal
7	DB0	H/L	Data bit 0
8	DB1	H/L	Data bit 1
9	DB2	H/L	Data bit 2
10	DB3	H/L	Data bit 3
11	DB4	H/L	Data bit 4
12	DB5	H/L	Data bit 5
13	DB6	H/L	Data bit 6
14	DB 7	H/L	Data bit 7
15	A		Power supply for LED backlight (+)
16	K		Power supply for LED backlight (-)

Figure 3.13: 16*2 LCD Display Pin Description

This LCD contains two articles called Command and Data. The command register stores the command instructions on the LCD. The command is a command given to the LCD to perform a predefined task such as starting it, clearing its screen, locating the cursor, controlling the display, etc.

F
e
a
t
u
r
e
s
o
f
L

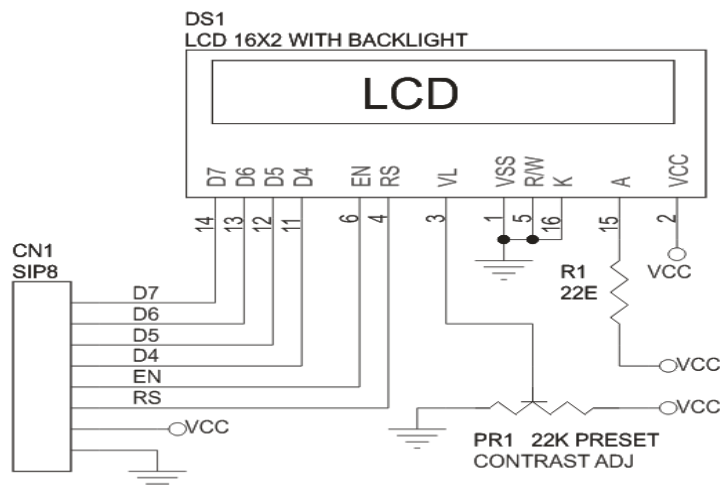


Figure 3.14 : 16*2 LCD Display interfacing with Arduino

Feature

The features of this LCD mainly include the following.

- The operating voltage of this LCD is 4.7V-5.3V
- It includes two rows where each row can produce 16-characters.
- The utilization of current is 1mA with no backlight
- Every character can be built with a 5×8 pixel box
- The alphanumeric LCDs alphabets & numbers
- Its display can work on two modes like 4-bit & 8-bit
- These are obtainable in Blue & Green Backlight
- It displays a few custom generated characters

3.6 RFID Module

RFID or Radio Frequency Identification system consists of two main components, a transponder/tag attached to an object to be identified, and a Transceiver also known as interrogator/Reader.

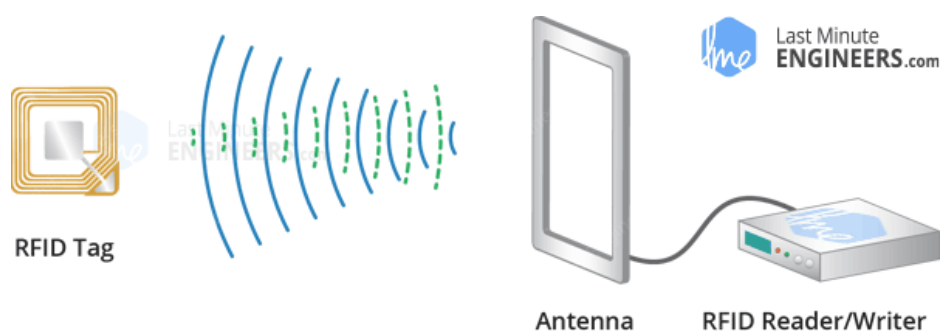


Figure 3.15: How RFID works

A Reader consists of a Radio Frequency module and an antenna which generates high frequency electromagnetic field. On the other hand, the tag is usually a passive device, meaning it doesn't contain a battery. Instead it contains a microchip that stores and processes information, and an antenna to receive and transmit a signal.

To read the information encoded on a tag, it is placed in close proximity to the Reader (does not need to be within direct line-of-sight of the reader). A Reader generates an electromagnetic field which causes electrons to move through the tag's antenna and subsequently power the chip.

The powered chip inside the tag then responds by sending its stored information back to the reader in the form of another radio signal. This is called backscatter. The backscatter, or change in the electromagnetic/RF wave, is detected and interpreted by the reader which then sends the data out to a computer or microcontroller.

Our RDM6300 RFID 125KHz card reader mini-module is designed for reading code from 125KHz card compatible read-only tags and read/write card. It can be applied in office/home security, personal identification, access control, anti-forgery, interactive toy and production control systems etc.

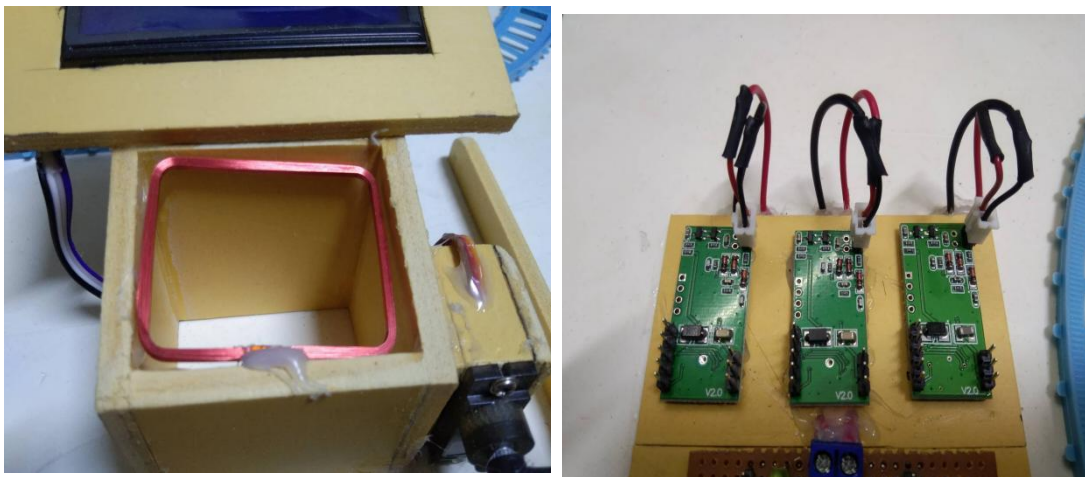


Figure 3.16: RDM 6300 RFID Module

Features:

- Support external antenna
- Maximum effective distance up to 50mm
- Less than 100ms decoding time
- Uart interface
- Support EM4100 compatible read only or read/write tags

- Small outline design

3.7 Resistor

Resistors are electronic components which have a specific, never-changing electrical resistance. The resistor's resistance limits the flow of electrons through a circuit. They are passive components, meaning they only consume power (and can't generate it). Resistors are usually added to circuits where they complement active components like op-amps, microcontrollers, and other integrated circuits. Commonly resistors are used to limit current, divide voltages, and pull-up I/O lines.

The electrical resistance of a resistor is measured in ohms. The symbol for an ohm is the Greek capital-omega: Ω . The (somewhat roundabout) definition of 1Ω is the resistance between two points where 1 volt (1V) of applied potential energy will push 1 ampere (1A) of current. As SI units go, larger or smaller values of ohms can be matched with a prefix like kilo-, mega-, or giga-, to make large values easier to read. It's very common to see resistors in the kilohm ($k\Omega$) and megaohm ($M\Omega$) range (much less common to see milliohm ($m\Omega$) resistors). For example, a $4,700\Omega$ resistor is equivalent to a $4.7k\Omega$ resistor, and a $5,600,000\Omega$ resistor can be written as $5,600k\Omega$ or (more commonly as) $5.6M\Omega$. All resistors have two terminals, one connection on each end of the resistor. When modeled on a schematic, a resistor will show up as one of these two symbols:

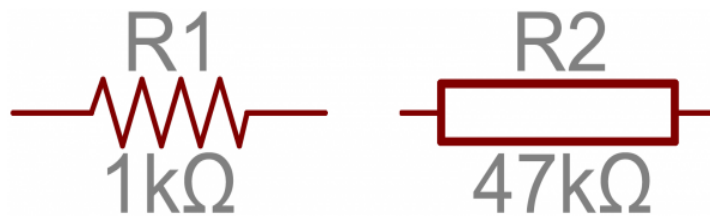


Figure 3.17: Two common resistor schematic symbols

3.8 Arduino IDE

The digital microcontroller unit named as Arduino Nano can be programmed with the Arduino software IDE. There is no any requirement for installing other software rather than Arduino. Firstly, Select "Arduino Nano from the Tools, Board menu (according to the microcontroller on our board). The IC used named as ATmega328 on the Arduino Nano comes pre burned with a boot loader that allows us to upload new code to it without the use of an external hardware programmer.

Communication is using the original STK500 protocol (reference, C header files). We can also bypass the boot loader and programs the microcontroller through the ICSP (In Circuit Serial Programming) header. The ATmega16U2 (or 8U2 in the rev1 and rev2 boards) firmware source code is available. The ATmega16U2/8U2 is loaded with a DFU boot loader, which can be activated by:

On Rev1 boards: connecting the solder jumper on the back of the board (near the map of Italy) and then resetting the 8U2. On Rev2 or later boards: there is a resistor that pulling the 8U2/16U2 HWB line to ground, making it easier to put into DFU mode.

The Arduino Nano is one of the latest digital microcontroller units and has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega328 provides UART TTL at (5V) with serial communication, which is available on digital pins 0 -(RX) for receive the data and pin no.1 (TX) for transmit the data. An ATmega16U2 on the board channels this serial communication over USB and appears as a virtual com port to software on the computer. The '16U2 firmware uses the standard USB COM drivers, and no external driver is needed. However, on Windows, an .in file is required. The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the Arduino board.

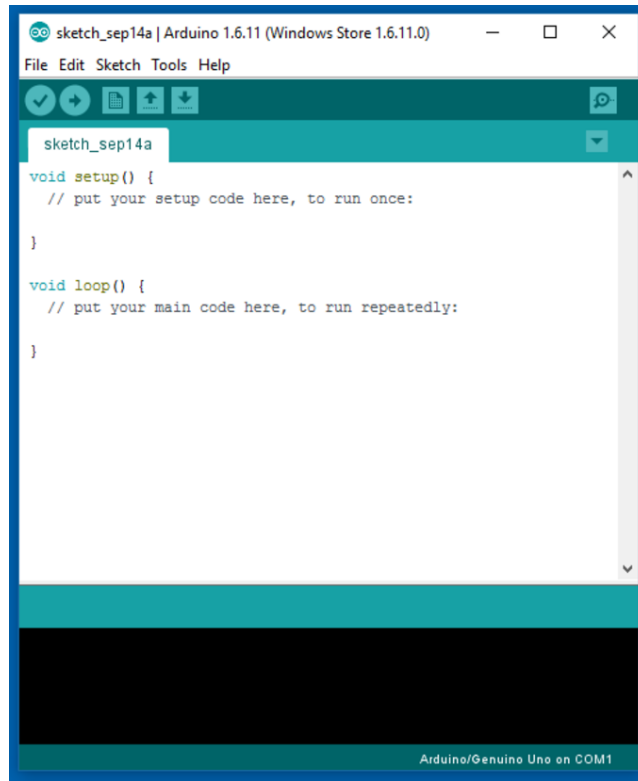


Figure 3.18: Arduino Software Interface IDE

The RX and TX LEDs on the board will flash when data is being transmitted via the USB-to-serial chip and USB connection to the computer (but not for serial Communication on pins 0 and 1). A Software Serial library allows for serial communication on any of the Nano's digital pins. The ATmega328 also supports I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I2C bus. Arduino programs are written in C or C++ and the program code written for Arduino is called sketch. The Arduino IDE uses the GNU tool chain and AVR Lab to compile programs, and for uploading the programs it uses argued. As the Arduino platform uses Atmel microcontrollers, Atmel's development environment, AVR Studio or the newer Atmel Studio, may also be used to develop software for the Arduino.

The Arduino Integrated Development Environment - or Arduino Software (IDE) - contains a text editor for writing code, a message area, a text console, a toolbar with buttons for common functions and a series of menus. It connects to the Arduino and Genuino hardware to upload programs.

Writing Sketches

Programs written using Arduino Software (IDE) are called sketches. These sketches are written in the text editor and are saved with the file extension. ino. The editor has features for cutting/pasting and for searching/replacing text. The message area gives feedback while saving and exporting and also displays errors. The console displays text output by the Arduino Software (IDE), including complete error messages and other information. The bottom right-hand corner of the window displays the configured board and serial port. The toolbar buttons allow you to verify and upload programs, create, open, and save sketches, and open the serial monitor.

3.9 Proteus Software

The Proteus Design Suite is a proprietary software tool suite used primarily for electronic design automation. The software is used mainly by electronics design engineers and technicians to create schematics and electronics prints for manufacturing printed circuit boards.

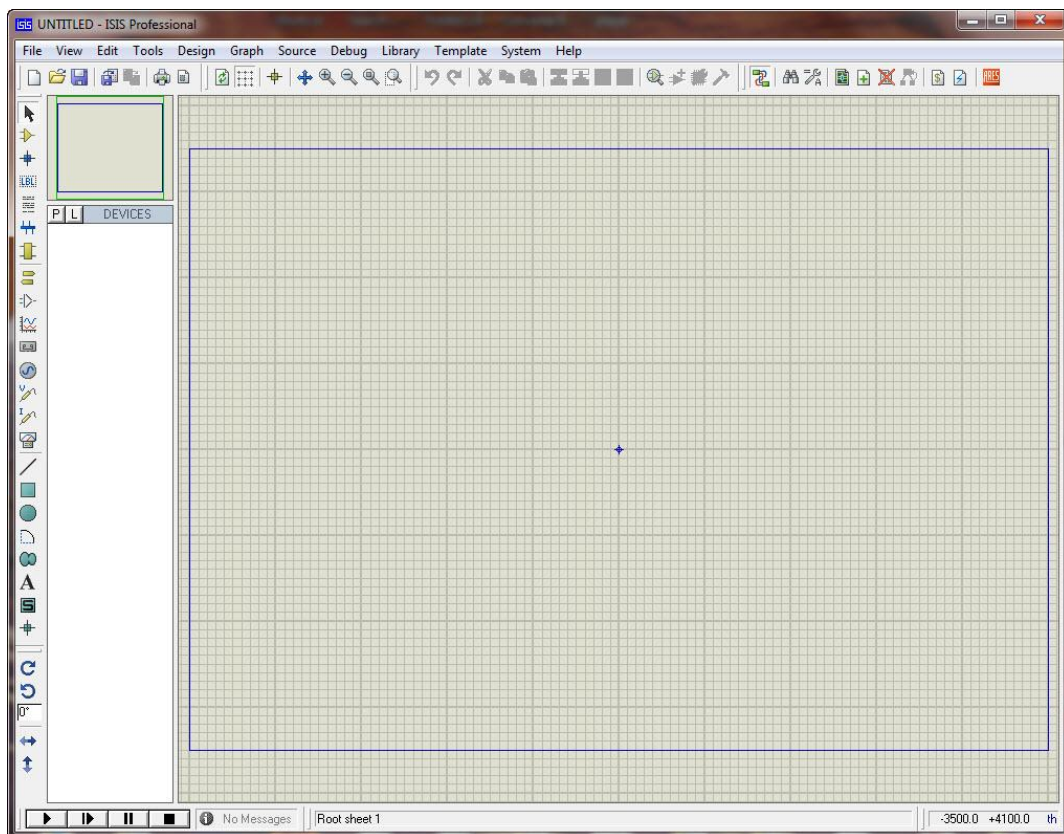


Figure 3.19: Proteus Software Interface

The first version of what is now the Proteus Design Suite was called PC-B and was written by the company chairman, John Jameson, for DOS in 1988. Schematic Capture support followed in 1990 with a port to the Windows environment shortly thereafter. Mixed mode SPICE Simulation was first integrated into Proteus in 1996 and microcontroller simulation then arrived in Proteus in 1998. Shape based auto routing was added in 2002 and 2006 saw another major product update with 3D Board Visualization. More recently, a dedicated IDE for simulation was added in 2011 and MCAD import/export was included in 2015. Support for high speed design was added in 2017. Feature led product releases are typically biannual, while maintenance-based service packs are released as required.

CHAPTER 4 METHODOLOGY

4.1 Our methodologies for the project

Our methodologies for the project:

- Creating an idea for **Design, Construction and Performance Test of Metro Rail Gate Control System**. And designing a block diagram & circuit diagram to know which components we need to construct it.
- Collecting all the components and programming the micro controller to control the whole system.
- Setting up all the components in a PCB board & then soldering. Lastly, assembling all the blocks in a board and to run the system & for checking purposes.

4.2 Block Diagram

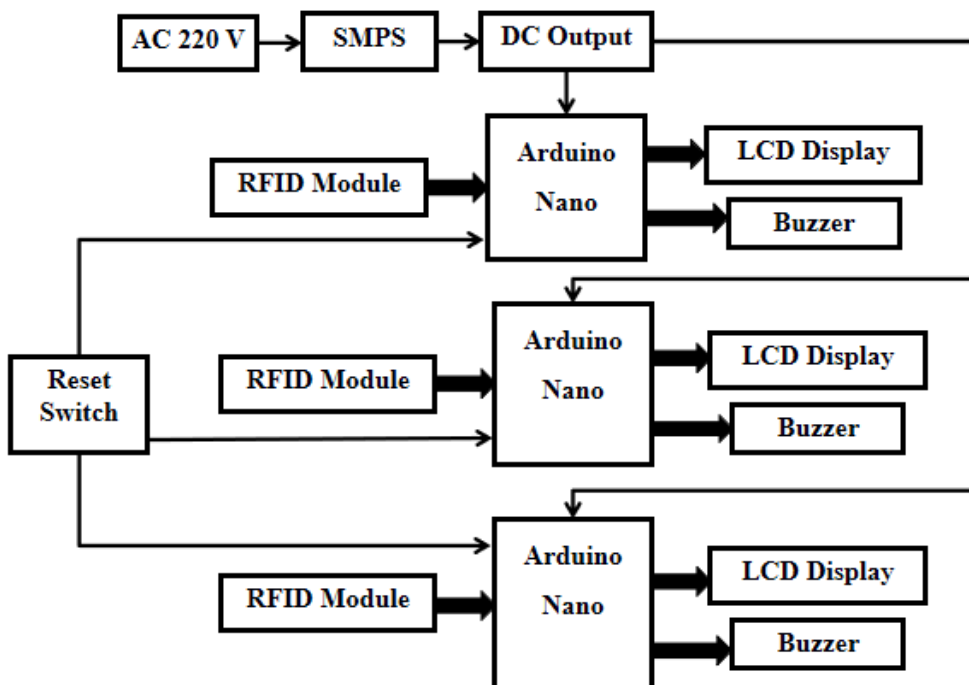


Figure 4.1: Block Diagram of Our System

4.3 Schematic Diagram

The schematic diagram here is representing the electrical circuit and the components of the project. Here we have used standardized symbols and lines.

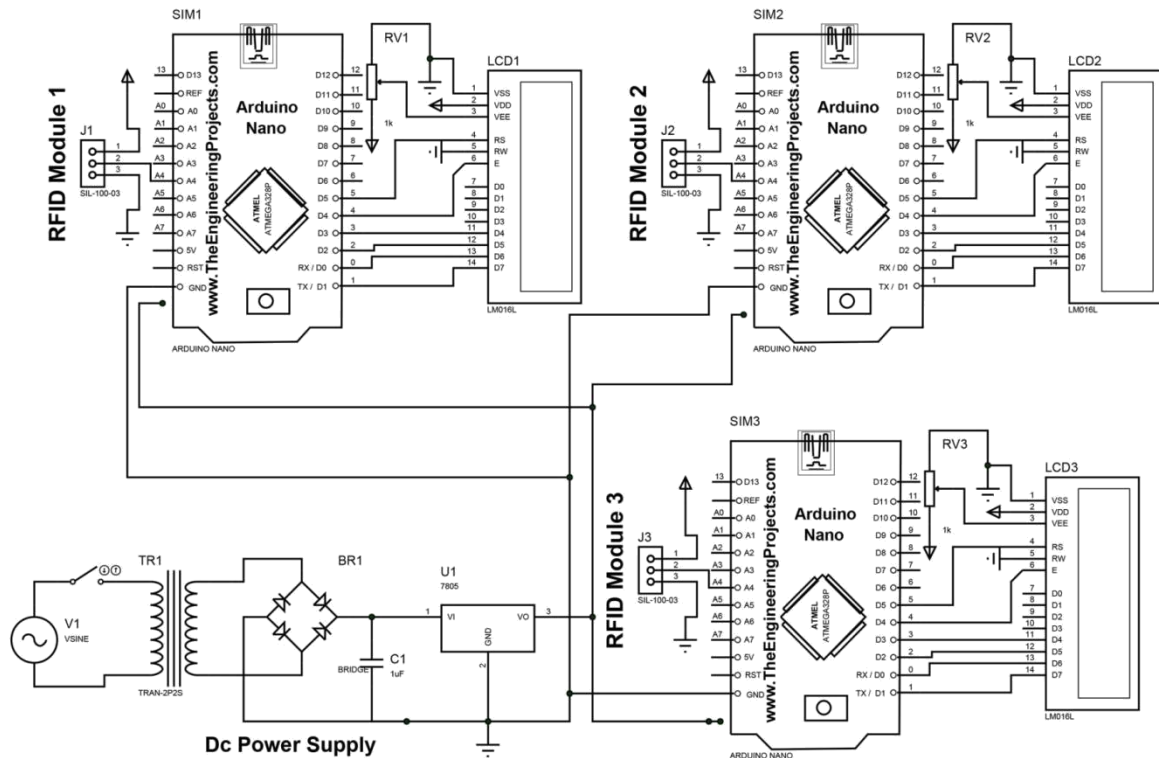


Figure 4.2: Schematic Diagram of the Project

4.4 Working Principle

Our main power supply for this system is AC 220 voltage. This power is connected to the SMPS. The main function of SMPS is to convert AC to DC. Since all our instruments in this project are DC, we need DC power. By using a voltage regulator we get a fixed 5 voltage DC for our DC source. Here Arduino Nano Micro Controller is used as the main brain of our system. It can be controlled by programming in this controller. In our system, when a user enters the station, he/she will enter with a card. After that, the user has to scan the card again to travel on the metro rail and get off at the next stop. Our system will then deduct the payment from the card. This is the main function of our system.

4.5 Our Final System View



Figure 4.3: Our Final System Overview

CHAPTER 5

RESULT AND DISCUSSION

5.1 Discussion

While working on our project, we did face some difficulties as it is a very complex system but the end results, we came up with were quite satisfactory. We have put the whole system through several tasks to validate our work and also have taken necessary notes for future improvements. Some future recommendations that we have involves improvement in system design and wiring, adding features for more efficient.

5.2 Result

Now, it's time to talk about the results. We have written our commands using the Arduino IDE and the following things can happen:

- After power this project then it will be able to operate.
- Here we make 3 gates on our system.
- When a person wants to travel to metro rail then scan the RFID Card.
- And payment will be cut from the card.

5.3 Performance Test

```
#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x27 ,16,2);

#include <Servo.h>

Servo myservo;

#include <rdm6300.h>

#define RDM6300_RX_PIN 2 // read the SoftwareSerial doc above! may need to change
this pin to 10...

int rfid_data;

Rdm6300 rdm6300;

int data1 = 3;

int data2 = 4;

int data3 = 6;

int gate2_data1;

int data4 = 7;

int gate3_data1;

int buzzar = A0;

void setup() {

  Serial.begin(9600); // opens serial port, sets data rate to 9600 bps

  rdm6300.begin(RDM6300_RX_PIN);

  lcd.init();

  lcd.backlight();
```

```

lcd.begin(16,2);
lcd.clear();
lcd.setCursor(0,0);
lcd.print(".... Please ....");
lcd.setCursor(0,1);
lcd.print("Punch Your Card ");
myservo.attach(5); // attaches the servo on pin 9 to the servo object
myservo.write(0); // tell servo to go to position in variable 'pos'
pinMode(data3,INPUT);
pinMode(data4,INPUT);

pinMode(data1,OUTPUT);
pinMode(data2,OUTPUT);
digitalWrite(data1 ,LOW);
digitalWrite(data2 ,LOW);
}
void(* resetFunc) (void) = 0; //declare reset function @ add

void loop() {

if (rdm6300.update())
rfid_data = rdm6300.get_tag_id();
Serial.println(rfid_data);

gate2_data1 = digitalRead(data3);
Serial.println(gate2_data1);
gate3_data1 = digitalRead(data4);

```

```

Serial.println(gate3_data1);

if(rfid_data == 7333 && gate3_data1 == HIGH){
myservo.write(90);          // tell servo to go to position in variable 'pos'
lcd.clear();
lcd.setCursor(0,0);
lcd.print("... Thank You ...");
lcd.setCursor(0,1);
lcd.print(" Your Fee 200 Tk.");
delay(3000);
myservo.write(0);          // tell servo to go to position in variable 'pos'
delay(1000);
resetFunc();
}

if(rfid_data == 7333 && gate2_data1 == HIGH){
myservo.write(90);          // tell servo to go to position in variable 'pos'
lcd.clear();
lcd.setCursor(0,0);
lcd.print("... Thank You ...");
lcd.setCursor(0,1);
lcd.print(" Your Fee 100 Tk.");
delay(3000);
myservo.write(0);          // tell servo to go to position in variable 'pos'
delay(1000);
resetFunc();
}

```



```

if(rfid_data == 7333){
myservo.write(90);          // tell servo to go to position in variable 'pos'
lcd.clear();
lcd.setCursor(0,0);
lcd.print(" Thank You..Have ");
lcd.setCursor(0,1);
lcd.print(" a Nice Journey ");

delay(2000);

myservo.write(0);          // tell servo to go to position in variable 'pos'
digitalWrite(data1 ,HIGH);
digitalWrite(data2 ,HIGH);
delay(10000);
resetFunc();

}

if(rfid_data == 10350){
myservo.write(0);          // tell servo to go to position in variable 'pos'
lcd.clear();
lcd.setCursor(0,0);
lcd.print(" .. Fake Card .. ");
delay(2000);
resetFunc();
delay(500);
}

delay(1000);

```

```
}
```

```
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
```

```
#include <LiquidCrystal_I2C.h>
```

```
LiquidCrystal_I2C lcd(0x27 ,16,2);
```

```
#include <Servo.h>
```

```
#include <EEPROM.h>
```

```
Servo myservo;
```

```
#include <rdm6300.h>
```

```
#define RDM6300_RX_PIN 2 // read the SoftwareSerial doc above! may need to change  
this pin to 10...
```

```
int rfid_data;
```

```
Rdm6300 rdm6300;
```

```
int data1 = 4;
```

```
int gate1_data1;
```

```
int data2 = 7;
```

```
int gate3_data1;
```

```
int gate1_output = 5;
```

```
int gate3_output = 6;
```

```
int address1 = 1;
```

```
int eeprom_value1;
```

```
void setup() {
```

```

Serial.begin(9600); // opens serial port, sets data rate to 9600 bps

Serial.println(EEPROM.length());

rdm6300.begin(RDM6300_RX_PIN);

lcd.init();

lcd.backlight();

lcd.begin(16,2);

lcd.clear();

lcd.setCursor(0,0);

lcd.print(".... Please ....");

lcd.setCursor(0,1);

lcd.print("Punch Your Card ");

myservo.attach(3); // attaches the servo on pin 9 to the servo object

myservo.write(0);          // tell servo to go to position in variable 'pos'

pinMode(data1,INPUT);

pinMode(data2,INPUT);

pinMode(gate1_output,OUTPUT);

pinMode(gate3_output,OUTPUT);

digitalWrite(data1 ,LOW);

digitalWrite(gate1_output ,LOW);

digitalWrite(gate3_output ,LOW);

}

void(* resetFunc) (void) = 0; //declare reset function @ add

void loop() {

```

```

if (rdm6300.update())
  rfid_data = rdm6300.get_tag_id();
Serial.println(rfid_data);

gate1_data1 = digitalRead(data1);
Serial.println(gate1_data1);
gate3_data1 = digitalRead(data2);
Serial.println(gate3_data1);

if(rfid_data == 7333 && gate1_data1 == HIGH){
myservo.write(90);      // tell servo to go to position in variable 'pos'
lcd.clear();
lcd.setCursor(0,0);
lcd.print("... Thank You ...");
lcd.setCursor(0,1);
lcd.print(" Your Fee 100 Tk.");
delay(3000);
myservo.write(0);      // tell servo to go to position in variable 'pos'
delay(1000);
resetFunc();
}

if(rfid_data == 7333 && gate3_data1 == HIGH){
myservo.write(90);      // tell servo to go to position in variable 'pos'
lcd.clear();
lcd.setCursor(0,0);

```

```

lcd.print("... Thank You ...");
lcd.setCursor(0,1);
lcd.print(" Your Fee 100 Tk.");
delay(3000);
myservo.write(0);          // tell servo to go to position in variable 'pos'
delay(1000);
resetFunc();
}

if(rfid_data == 7333){
myservo.write(90);        // tell servo to go to position in variable 'pos'
lcd.clear();
lcd.setCursor(0,0);
lcd.print(" Thank You..Have ");
lcd.setCursor(0,1);
lcd.print(" a Nice Journey ");
delay(2000);
myservo.write(0);        // tell servo to go to position in variable 'pos'
digitalWrite(gate1_output ,HIGH);
digitalWrite(gate3_output ,HIGH);
delay(10000);
resetFunc();
}

if(rfid_data == 10350){
myservo.write(0);        // tell servo to go to position in variable 'pos'
lcd.clear();

```

```
lcd.setCursor(0,0);  
lcd.print(" .. Fake Card .. ");  
delay(2000);  
resetFunc();  
delay(500);  
}  
delay(1000);  
  
}
```

```
////////////////////////////////////////////////////////////////
```

```
#include <LiquidCrystal_I2C.h>  
LiquidCrystal_I2C lcd(0x27 ,16,2);  
#include <Servo.h>  
#include <EEPROM.h>  
  
Servo myservo;  
#include <rdm6300.h>  
#define RDM6300_RX_PIN 2 // read the SoftwareSerial doc above! may need to change  
this pin to 10...  
int rfid_data;  
Rdm6300 rdm6300;  
  
int data1 = 3;  
int gate1_input;
```

```

int data2 = 5;

int gate2_input;

int gate1_data = 6;

int gate2_data = 7;

void setup() {

  Serial.begin(9600); // opens serial port, sets data rate to 9600 bps
  rdm6300.begin(RDM6300_RX_PIN);

  lcd.init();
  lcd.backlight();
  lcd.begin(16,2);
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print(".... Please ....");
  lcd.setCursor(0,1);
  lcd.print("Punch Your Card ");
  myservo.attach(4); // attaches the servo on pin 9 to the servo object
  myservo.write(0); // tell servo to go to position in variable 'pos'

  pinMode(gate1_data,OUTPUT);
  pinMode(gate2_data,OUTPUT);
  digitalWrite(gate1_data ,LOW);
  digitalWrite(gate2_data ,LOW);

  pinMode(data1,INPUT);
  pinMode(data2,INPUT);

```

```

}

void(* resetFunc) (void) = 0; //declare reset function @ add

void loop() {

  if (rdm6300.update())

    rfid_data = rdm6300.get_tag_id();

  Serial.println(rfid_data);

  gate1_input = digitalRead(data1);

  Serial.println(gate1_input);

  gate2_input = digitalRead(data2);

  Serial.println(gate2_input);

  if(rfid_data == 7333 && gate1_input == HIGH){

    myservo.write(90);          // tell servo to go to position in variable 'pos'

    lcd.clear();

    lcd.setCursor(0,0);

    lcd.print("... Thank You ...");

    lcd.setCursor(0,1);

    lcd.print(" Your Fee 200 Tk.");

    delay(3000);

    myservo.write(0);          // tell servo to go to position in variable 'pos'

    delay(1000);

    resetFunc();

  }

```



```

if(rfid_data == 7333 && gate2_input == HIGH){
myservo.write(90);          // tell servo to go to position in variable 'pos'
lcd.clear();
lcd.setCursor(0,0);
lcd.print("... Thank You ...");
lcd.setCursor(0,1);
lcd.print(" Your Fee 100 Tk.");
delay(3000);
myservo.write(0);          // tell servo to go to position in variable 'pos'
delay(1000);
resetFunc();
}

```

```

if(rfid_data == 7333){
myservo.write(90);          // tell servo to go to position in variable 'pos'
lcd.clear();
lcd.setCursor(0,0);
lcd.print(" Thank You..Have ");
lcd.setCursor(0,1);
lcd.print(" a Nice Journey ");

delay(2000);
myservo.write(0);          // tell servo to go to position in variable 'pos'
digitalWrite(gate1_data ,HIGH);
digitalWrite(gate2_data ,HIGH);
delay(10000);

```

```

resetFunc();
}
if(rfid_data == 10350){
myservo.write(0);          // tell servo to go to position in variable 'pos'
lcd.clear();
lcd.setCursor(0,0);
lcd.print(" .. Fake Card .. ");
delay(2000);
resetFunc();
delay(500);
}
delay(1000);
}

```

5.4 Advantage

There are many advantages of our project because of its accuracy. Some of the advantages are pointed out below:

- Transparent and convenient mode of billing/ ticketing.
- RFID cards are more reliable as compared to paper ticketing system.
- RFID tags are reusable.
- The RFID card can be used at any temperatures and any climatic conditions without affecting its performance.
- The RFID system facilitates smooth functioning of metro train system by avoiding quarrelling and corruption.

5.5 Application

Our project has many application areas and actually we need to use it in many places to verified the exact person which have the proper access . Some of the application areas of the project has been pointed out below:

- This System Can Be Widely Used Any Ticketing System.

CHAPTER 6 CONCLUSION

6.1 Conclusion

The system is expected to be fully automated, reliable, transparent and convenient. The whole system can also be used in vehicle on highways, their toll payment and in the railway ticketing system with small or no modification. The cards being reusable, they are much more convenient compared to the paper based ticketing system. The card also can be used to be a universal travel pass card that will allow any transportation on any route. Any unwanted events can be avoided as all the person carrying RFID tickets are monitored every time they travel. Also the possibilities of reducing traffic jams, chaos in the bus stoppage that we usually experienced in Dhaka city are immense. To save the human time for ticking or payment of billing is a challenge of the era of modern science and technology. The working model was fabricated within the laboratory premises. The results exhibit that it is one of the expedient approach for secure railway ticket payment system. In every station have an RFID Scanner where passengers scan their card and payment easily.

6.2 Future Scope

We are thinking about adding many features to our project in the future to get more desirable outcomes. Some of the steps that we are thinking about taking are given below:

- In future, we are looking forward to improving our whole system design to make it more efficient.
- In future, we are thinking about adding more features to the system such as cc cameras.
- In future, we are thinking about making the system IoT enabled so that wirelessly the whole system can be observed.

REFERENCE

- [1] Atlam, Hany & Alenezi, Ahmed & Alshdadi, Abdulrahman & Walters, Robert & Wills, Gary. (2017). Integration of Cloud Computing with Internet of Things: Challenges and Open Issues. 10.1109/iThings-GreenCom-CPSCom-SmartData.2017.105.
- [2] Albugmi, Ahmed & Alassafi, Madini & Walters, Robert & Wills, Gary. (2016). Data Security in Cloud Computing.
- [3] S. M. Babu, A. J. Lakshmi and B. T. Rao, "A study on cloud based Internet of Things: CloudIoT," 2015 Global Conference on Communication Technologies (GCCT), Thuckalay, 2015, pp. 60-65.
- [4] M. Elhoseny, G. Ramrez-Gonzalez, O. M. Abu-Elnasr, S. A. Shawkat, A. N and A. Farouk, "Secure Medical Data Transmission Model for IoT-Based Healthcare Systems," in IEEE Access, vol. 6, pp. 20596-20608, 2018.
- [5] D. Kang et al., "Room Temperature Control and Fire Alarm /Suppression IoT Service Using MQTT on AWS," 2017 International Conference on Platform Technology and Service (PlatCon), Busan, 2017, pp. 1-5.
- [6] F. J. D'souza and D. Panchal, "Advanced encryption standard (AES) security enhancement using hybrid approach," 2017 International Conference on Computing, Communication and Automation (ICCCA), Greater Noida, 2017, pp. 647-652.
- [7] A. Bansal and A. Agrawal, "Providing security, integrity and authentication using ECC algorithm in cloud storage," 2017 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, 2017, pp. 1-5.
- [8] M. J. Kaur and P. Maheshwari, "Building smart cities applications using IoT and cloud-based architectures," 2016 International Conference on Industrial Informatics and Computer Systems (CIICS), Sharjah, 2016, pp. 1-5.
- [9] Rijmen, V., & Daemen, J. (2001). Advanced encryption standard. Proceedings of Federal Information Processing Standards Publications, National Institute of Standards and Technology, 19-22.
- [10] S. P. Jena, S. Aman, R. Das Computerized Green House Data Acquisition System Using Arduino with LabVIEW, International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, Vol. 4, Issue 4, April 2015

