Tribhuvan University

Institute of Engineering

Kathmandu Engineering College

Department of Electronics and Communication Engineering

Minor Project Report

On

Smart Car Parking Project

By

Dipesh Dhamala(72023)

Jeevan Sapkota(72026)

Jyoti Kc(72027)

Kailash Shrestha(72028)

Kathmandu, Nepal

AUGUST, 2018

ACKNOWLEDGEMENT

We sincerely thanks **Er. Sagun Manandhar**, Head of Department of Electronics and Communication Engineering for providing us with the opportunity to do this minor project. We extend our gratitude to **Er. Dipen Manandhar**, our project coordinator for his generous support and for helping us in different aspects for the development of project.

We express our deep sense of gratitude to **Er. Anand Kumar Shah**, for his precious time examining our project.

Sincere thanks to all the individuals who helped us with their valuable suggestions so as to build up this report.

ABSTRACT

This project is a simple demonstration that aims to provide a user friendly, reliable and automated car parking system. In this project, the design and implementation of Smart Car Parking System is mentioned that permits drivers to effectively locate and withhold the vacant parking spaces. This system allows the drivers to leave their car at the entrance which is further transported inside different parking stations of the buildings with mechanical structure. Similarly, they are retrieved by mechanical structure and placed at the exit for the owner to drive away. When a car arrives at the entrance, the driver stops the car. If the availability of Parking space is confirmed, the user is allowed to have a certain number as identity of their vehicle and the mechanical system parks the car at appropriate location. In order to retrieve the car, the user provides the same number and the mechanical system brings back the car to the exit. LEDs indicate the availability of parking stations. The system uses least number of required components without sacrificing the reliability thus it serves as an economical parking system.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ACRONYMS

AC	:	Alternating Current
DC	:	Direct Current
I/O	:	Input/output
LCD	:	Liquid Crystal Display
LED	:	Light Emitting Diode
MOSI	:	Master Out Slave In
PIO	:	Programmed Input/output
PWM	:	Pulse Width Modulation
SCK	:	Serial Clock
SCL	:	Serial Clock Input
SDA	:	Serial Data Input/output
SPI	:	Serial Peripheral Interface
TWI	:	Two Wire Interface
USB	:	Universal Serial Bus

CHAPTER 1

INTRODUCTION

1.1 BACKGROUND THEORY

In today's parking system there is not standard system to check the parking spaces. The system heavily relies on human interaction with physical space and entity. This leads to wastage of human manpower and also parking spaces at times. These parking lots are dependent on Human-to-Human Interaction (HHI) which is not efficient.

In urban areas, most of the cars which have reached their destination and are circling around looking for a parking space thus leading to problems like pollution and traffic congestion. It is essential to control the pollution cause by the burning of fuels using a robust car parking system that will be used for parking spot allocation in on spot resource allocation scenario. A system that does the entire work of parking and unparking in a systematic manner would remove the hassle of the user regarding parking and space management.

On the other hand, if a car is parked in such a way that it occupies two parking slots rather than one, this is called improper parking. Improper parking can happen when a driver is not careful about another driver's rights. This is tackled by the development of automated smart car parking system as the parking procedure is automatically done by the device itself without need of the human knowledge. An automated system built with a secure authentication mechanism would therefore add up extra features like security to the parked vehicles of the owner and would serve as the best solution for global rising parking issues.

1.2 PROBLEM STATEMENT

In today's context most of the parking lots depend upon human manpower itself which in not reliable. Most of the time when users go to malls and commercial complex, they experience that there is limited space for parking spots especially on prime hours. There is greater difficulty in finding vacant spaces quickly in a multilevel parking lot especially on weekends or public holidays. Bus stations or shopping malls are crowded at peak periods and difficulty in finding vacant slots at these places is a major problem for customers. Insufficient car park spaces lead to traffic congestion and driver frustration. Hence, there is a desperate need of robust parking system that will enable us to allocate the parking spots with vehicle security.

1.3 OBJECTIVES

The main objectives of our project are mentioned below:

- To develop an intelligent, user friendly automated car parking system which reduces the manpower and traffic congestion.
- To offer safe and secure parking slots within limited area in a reliable way.

1.4 PROJECT SCOPE AND APPLICATION

This proposed project can be applied in different fields as mentioned below:

- This system can be implemented in most busy areas like commercial complex, and parking spots of malls.
- Used in those areas where safety of the parked vehicles is necessary.
- It integrates the entire user experience into a unified action.

1.5 ORGANIZATION OF REPORT

This report is comprised of various sections which are briefly described below. Initially sincere appreciation to all the individuals who helped us in order to accomplish this project was recorded. In order to provide brief information about the report an abstract section is present. The introduction section includes the background which provides general idea of the project and also contains the problem statement which states different problems that are still prevalent among us upon which we were inspired to commence this project on. Objectives of this project is also listed out under this section. Literature Review is a section available in this report which includes different ways the project has been conducted by different people from round the globe. Outcomes of the project along with improvement in the current project are mentioned. Methodology section present in this report that describes about the circuit used for the project along with description of the components and the way they work. In order to picturize the operation process, algorithms and flowcharts are provided within this report. All the accomplished tasks along with the fulfillment of objectives are mentioned under the Conclusion section of this report. References section includes different sources from which we took help from so as to build up this project.

CHAPTER 2

LITERATURE REVIEW

Various methods are prevalent for development of autonomous or intelligent parking systems. Study of these systems shows that these require a little or more human intervention for the functioning. One of the intelligent systems for car parking has been proposed by making use of Image processing [1]. In this system, a brown rounded image on the parking lot is captured and processed to detect the free parking slot. The information about the currently available parking slots is displayed on the 7-segment display. Initially, the image of parking slots with brown-rounded image is taken. The image is segmented to create binary images. The noise is removed from this image and the object boundaries are traced. The image detection module determines which objects are round, by estimating each object's area and perimeter. Accordingly, the free parking space is allocated. A vision-based car parking system [2] is developed which uses two types of images (positive and negative) to detect free parking slot. In this method, the object classifier detects the required object within the input. Positive images contain the images of cars from various angles. Negative images do not contain any cars in them. The co-ordinates of parking lots specified are used as input to detect the presence of cars in the region. Haar-like features are used for feature detection. However, limitations may occur with this system with respect to the type of camera used. Also, the co-ordinate system used selects specific parking locations and thus camera has to be at a fixed location. Limited set of positive and negative images may impose limitations on the system.

Number Plate Recognition [3] technique for developing autonomous car parking system uses image processing basis to process the number plates of the vehicles. In this system, the image of the license number plate of the vehicle is acquired. It is further segmented to obtain individual characters in the number plate. Ultrasonic sensors are used to detect free parking slots. Then the images of number plate are

the parking fees. The LCD displays ‘FULL’ sign to indicate that a parking slot is taken and analyzed. Simultaneously, the current timing is noted so as to calculate not available. RFID [4] is also used in different areas for achieving smart car parking facility. However, some limitations with the system include background color being compulsorily black and character color white. Also, analysis is limited to number plates with just one row.

Smart Car Parking system designed proposed a mechanical model where availability of the parking stations would be indicated by the status of LED. The car would be parked with the use of lift at multiple levels. The whole process is completely autonomous. Thus, we aim to propose a car parking system that represents a fully automated model with minimum human intervention and overcome the limitations of existing systems.

CHAPTER 3

RELATED THEORY

For the implementation of any system, hardware and software components are the essential parts. Thus, the integral parts of Fingerprint Based Authentication system are as mentioned:

3.1 HARDWARE DESCRIPTIONS

Arduino Mega:

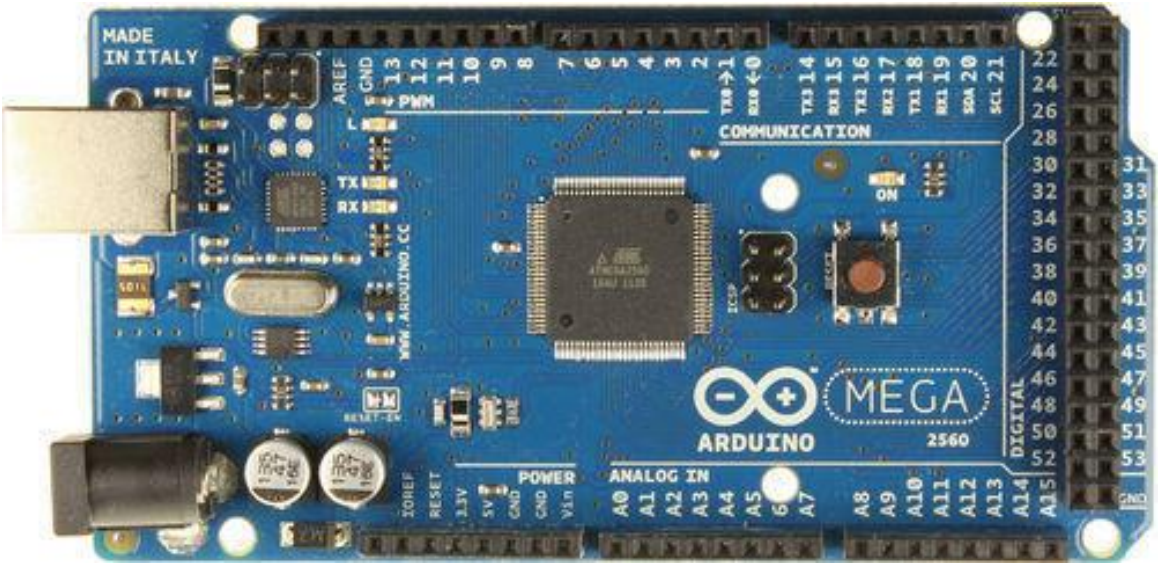

Figure 3.1: Arduino Mega 2560

Arduino is an open-source electronics platform based on easy-to-use hardware and software. It's intended for making interactive projects. The Arduino Mega is a microcontroller board based on the ATmega2560. It has 54 digital input/output pins (of which 14 can be used as PWM outputs), 16 analog inputs, 4 UARTs (hardware serial ports), a 16 MHz crystal oscillator, a USB connection, a power jack, and ICSP header, and a reset button.

Each of the 54 digital pins on the board can be used as an input or output, using pin

Mode(), digital Write(), and Read() functions. They operate at 5 volts. Each pin can provide or receive 40 mA as recommended operating condition and has an internal pull-up resistor (disconnected by default) of 20-50k ohm. A maximum of 12V is the value that must not be exceeded on any I/O pin to avoid permanent damage to the microcontroller. In addition, some pins have specialized functions:

- Serial: 0 (RX) and 1 (TX); Serial 1: 19 (RX) and 18 (TX); Serial 2: 17 (RX) and 16 (TX); Serial 3: 15 (RX) and 14 (TX). Used to receive (RX) and transmit (TX) TTL serial data. Pins 0 and 1 are also connected to the corresponding pins of the FTDI USB-to-TTL Serial chip.
- External Interrupts: 2 (interrupt 0), 3 (interrupt 1), 18 (interrupt 5), 19 (interrupt 4), 20 (interrupt 3), and 21 (interrupt 2). 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: 2 to 13 and 44 to 46. Provide 8-bit PWM output with the analogWrite() function.
- SPI: 50 (MISO), 51 (MOSI), 52 (SCK), 53 (SS). These pins support SPI communication, which, although provided by the underlying hardware, is not currently included in the Arduino language. The SPI pins are also broken out on the ICSP header, which is physically compatible with the Duemilanove and Diecimila.
- 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.
- I2C: 20 (SDA) and 21 (SCL). Support I2C (TWI) communication using the Wire library (documentation on the Wiring website). Note that these pins are not in the same location as the I2C pins on the Duemilanove or Diecimila.
- AREF. Reference voltage for the analog inputs. Used with analogReference().

- 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.
- VIN. The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- 5V. The regulated power supply used to power the microcontroller and other components on the board. This can come either from VIN via an on-board regulator, or be supplied by USB or another regulated 5V supply.

16*2 LCD Display:

A 16x2 LCD display is very basic module and is very commonly used in various devices and circuits. These modules are preferred over seven segments and other multi segment LEDs. The reasons being: LCDs are economical; easily programmable; have no limitation of characters animations and so on. A 16x2 LCD means it can display 16 characters per line and there are 2 such lines. In this LCD each character is displayed in 5x7 pixel matrix.

This LCD has two registers, namely, Command and Data. The command register stores the command instructions given to the LCD. A command is an instruction given to LCD to do a predefined task like initializing it, clearing its screen, setting the cursor position, controlling display etc. The data register stores the data to be displayed on the LCD.

Figure 3.2:16x2 LCD Display Module

Keypad

The 16 buttons keypad provides a useful human interface component for microcontroller. Convenient adhesive backing provides a simple way to mount the keypad in a variety of applications. Matrix keypad uses a combination button states to the host device, typically a microcontroller. Underneath each key is a pushbutton, with one end connected to one row and the other end connected to one column. In order for the microcontroller to determine which button is pressed, it first needs to pull each of the four columns (pins 1-4) either low or high one at a time, and then poll the states of the four rows (pins 5-8). Depending on the states of the columns, the microcontroller can tell which button is pressed. For example, say your program pulls all four columns low and then pulls the first row high. It then reads the input states of each column, and reads pin 1 is high. This means that a contact has been made between column 4 and row 1, so button A has been pressed.

Figure 3.3: 4x4 Keypad Module

Stepper Motor

A stepper motor is a brushless, synchronous electric motor that converts digital pulses into mechanical shaft rotations. Each rotation of a stepper motor is divided into a set number of steps, sometimes as many as 200 steps. The stepper motor must be sent a separate pulse for each step. The stepper motor can only receive one pulse and take one step at a time and each step must be the same length. Since each pulse results in the motor rotating a precise angles-typically 1.8 degrees-we can precisely

control the position of the stepper motor without any feedback mechanism. As the digital pulses from the controller increase in frequency, the stepping movement converts into a continuous rotation with the velocity of the rotation directly proportional to the frequency of the control pulses. Stepper motors are widely used because of their low cost, high reliability, and high torque at low speeds. Their rugged construction enables us to use stepper motors in a wide environmental range

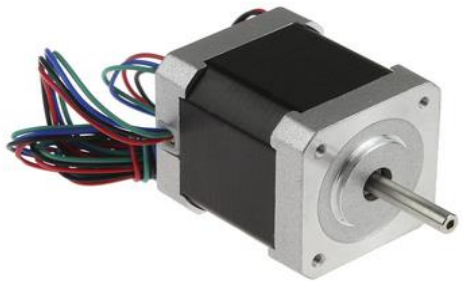

Figure 3.4: 4x4 Stepper Motor

3.2 SOFTWARE DESCRIPTIONS

The software is another integral part of the system design which controls the hardware. Thus, software must be designed with minimization of flaws so that the operation may be carried out effectively.

Software development process

In order to develop any system, there must be a series of logical steps. Software is a foothold which provides those series of stems to build up a system. It also provides stability and control of the project. Software also makes the implementation of hardware much easier with the help of series of codes compiled together called program. Since software enhances visibility of the project it should be developed in a way that it provides consistency to the system. The series of steps should be analyzed properly during the development phase for effective implementation in later phase.

Arduino is an open source electronics platform based on easy to use software too. This means that we can easily program the microcontroller the way we want to. Thus, we used Arduino IDE software so as to write the program and achieve our desired functionality. Programs for different components were written, compiled and then uploaded to the Arduino board.

CHAPTER 4

METHODOLOGY

4.1 BLOCK DIAGRAM

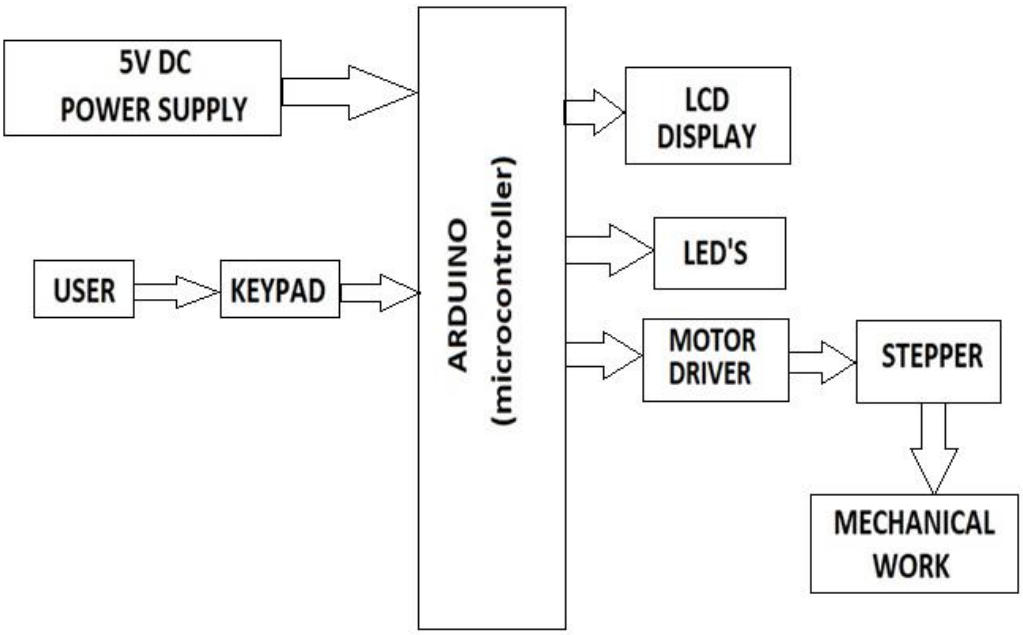

Figure 4.1: Block Diagram of Smart Car Parking System

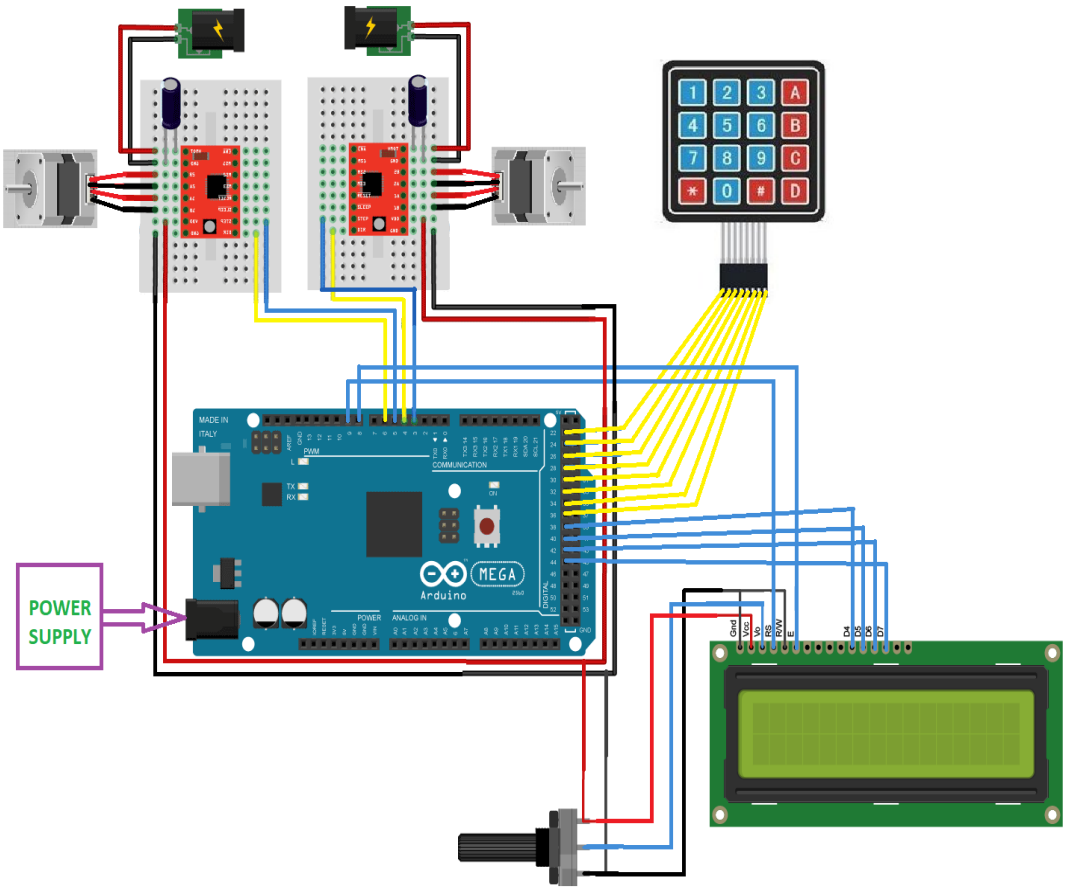

Figure 4.2: Schematic Diagram

LCD Module:

16x2 LCD module displays the message as per the signal received from microcontroller. As per the key pressed by the user on the keypad the microcontroller decides and sends a signal to the LCD module to display the appropriate message like displaying modes, asking for the code number etc.

Keypad:

4x4 keypad is used where the user presses a code that selects if the car is to be parked or retrieved. Initially a code is displayed either to park or retrieve the car. When parking code is pressed user is again allowed to register a number of the parking lot where he wants to keep the car. When user wants to retrieve the car, he needs to repress the same number in retrieve mode.

Microcontroller:

The microcontroller used is Arduino Mega. The Arduino is a microcontroller board based on the ATmega2560. The Arduino receives electrical signal from the sensor and operates upon the signal. All the decision-making processes are performed by the controller. It identifies which mode the user has entered then sends another signal that grants user to select a parking number of different stations to park his car. The number pressed by the user is saved by the controller and used as verification while user tries to retrieve his car. The controller prescribes the required action to the stepper motor to move along different parking blocks.

Power Supply:

Power supply unit or PSU are the components that supply power to the components including sensor, microcontroller, stepper motor and other components.

Stepper Motor:

Stepper motor receives the decisive signal produced by the microprocessor and operates upon that signal. During parking process after the code registration process

is complete the motor carries the car to different parking stations and parks the car. It is responsible to move the car in different directions as per the information supplied through the sensors. Also, during the retrieval process the motor moves to an appropriate location as per the code number and carries the car back to the exit.

LED:

LED (Light Emitting Diode) is used to represent the status of the parking station. The number of vacant parking lots is indicated by the LED's turning off. When the parking stations are filled with the cars the status of the LED is automatically updated.

4.3 FLOWCHART

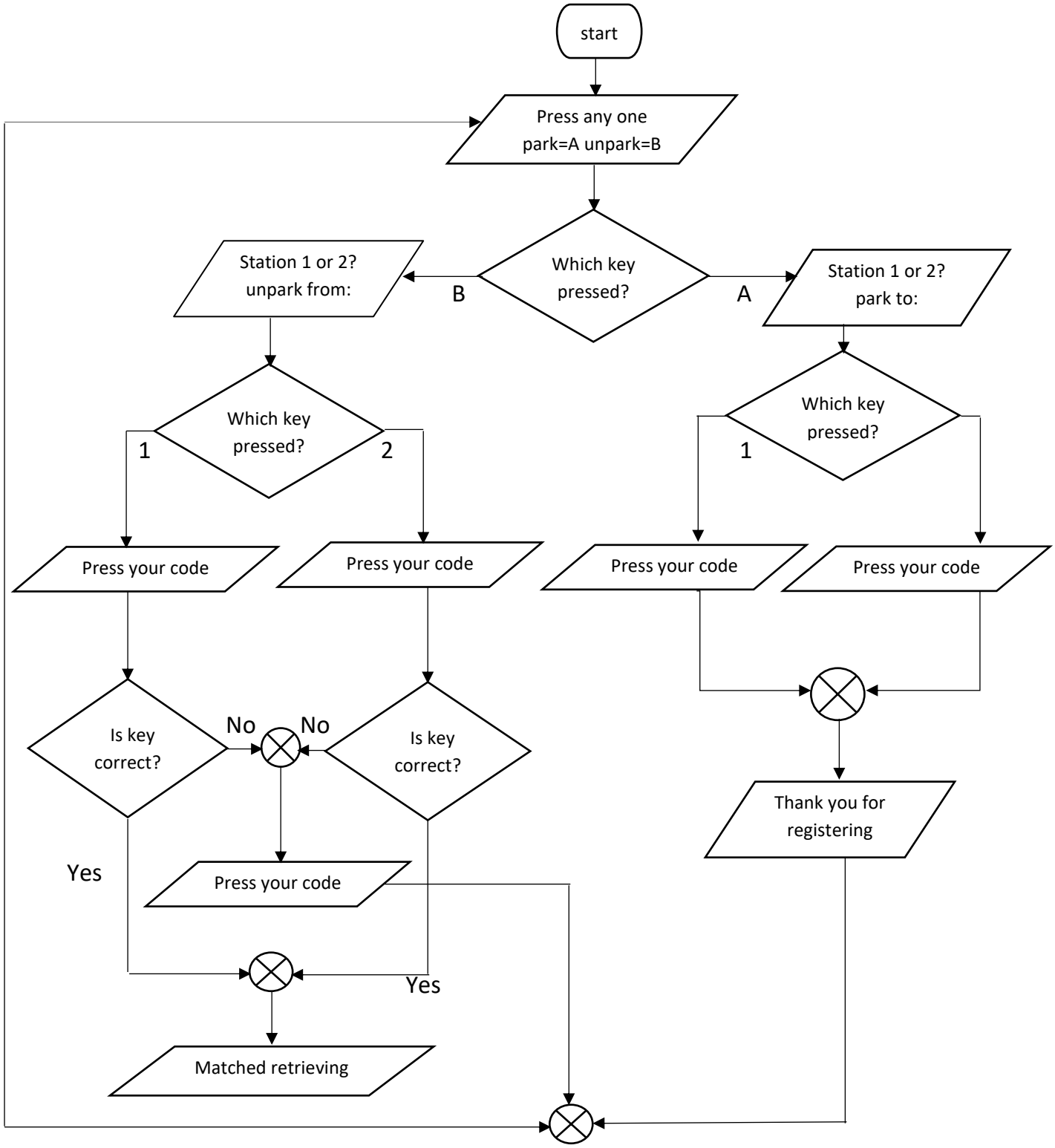

Figure 4.3: Flowchart

Flow Chart Explanation

Initially the user stops his car at the entrance. The user is prompted with the message on the LCD screen either to press A or B. In order to park the vehicle user needs to press A. Once A is pressed the user is re-prompted with the message to press the number of the parking station that user favors to park the vehicle. Once the number is selected the user is asked to press his secret code and then the car is autonomously parked by automatic parking mechanism with the help of stepper action. During the retrieval process the user needs to get through a series of authentication. The user is again asked to select the mode. When he presses B, the controller recognizes that the user is intending to retrieve the vehicle. Now the user is prompted with a message to enter the parking station number. If the user enters the correct parking station number then he is re-prompted with secret code and if the code matches the autonomous mechanical action brings back the car from stored parking location. But if the code entered is incorrect then the error message is prompted and the car isn't retrieved. Parking and retrieving of car immediately change the status of LED so that the next user in sequence would be well informed about the parking lot.

ALGORITHM

step 1: start

step 2: print: smart car parking project

step 3: print: press any one

park=A unpark=B

step 4: if A is pressed go to step 5

else go to step 12

step 5: print: station 1 or 2?

park to:

step 6: if 1 is pressed go to step 7

else go to step 10

step 7: print: press your code

step 8: thank you for registering

step 9: go to step 3

step 10: press your code

step 11: go to step 8

step 12: print: station 1 or 2?

park to:

step 13: if 1 is pressed go to step 14

else go to step 19

step 14: if key is correct go to step 15

else go to step 17

step 15: print: matched retrieving car

step 16: go to step 3

else go to step 17

step 17: print: error try again

step 18: go to step 3

step 19: if key is correct go to step 15

else go to step 17

CHAPTER 5

RESULT AND CONCLUSION

5.1 RESULTS

We were able to cover up the following aspects regarding our project:

- Interfacing 16x2 LCD with Arduino MEGA

  We made necessary connections between 16x2 LCD and Arduino MEGA and were able to display message through the display module.

- Interfacing 4x4 Keypad with Arduino MEGA

  We also connected 4x4 Keypad module with Arduino MEGA and were able to provide character inputs using this module.

- Programming each component using Arduino IDE Software

  We programmed these components using Arduino IDE Software.

- Constructing mechanical structure for autonomous parking

  We made a mechanical structure capable of lifting the car to different parking lot.

- Interfacing NIMA 17 Stepper Motor with Arduino Mega and programming it

  We made necessary connection between Stepper motor and Arduino Mega.

- Attaching Stepper to mechanical structure along with different accessories

  We attached stepper with the mechanical structure and made the whole structure upright using different accessories.

- Building a parking lot or station

  We made a two-step parking lot with wooden structure covered with sun board.

Some snapshots of our project are present below,

Fig 5.1: A complete structure

5.2 CONCLUSION

We drew following conclusions from our project:

- We were able to develop an intelligent, user friendly automated car parking system which would efficiently reduce manpower effort and traffic congestion.
- An autonomous system with secure authentication mechanism was implemented which would provide safe and securing parking slots within limited area in a reliable way.

REFERENCES

1. M.A.R. Sarkar, A.A. Rokoni, M.O. Reza, M.F. Ismail, "*Smart Parking system with image processing facility*", I.J. Intelligent Systems and Applications, 2012, vol. 3, pp. 41-47.
2. Hamada R.H.AI-Absi,Patrick Sebastian , "*Vision-Based Automated Parking System*" in 10th International Conference on Information science,2010.
3. K. Finkenzeller, "*Fundamentals and Applications in Contactless Smart Cards and Identification.*" John Wiley and Sons Ltd, 2003.
4. K. M. R. Sudeep Dogra, "*Radio frequency identification (RFID) applicatio ns: A brief introduction, advanced engineering in-formatics.*" The IUP jour nal of Electrical and Electronics Engineering, 2011, p. 3.
5. Arduino.cc. (2018). Arduino - Home. [online] Available at:https://www.ard uino.cc/en/uploads/Hacking/PinMap2560big [Accessed 5 Aug. 2018].
6. Seed Studio. (2018). LCD. [online] Available at:http://www.seedstudio.com /lcd16x2 [Accessed 5 Aug. 2018].
7. Seed Studio. (2018). Keypad. [online] Available at:http://www.seedstudio.c om/4x4keypad[Accessed 5 Aug 2018].
8. Inventables.com. (2018). Inventables. [online] Available at:http://www.inve ntables.com/techonologies/stepper-motor-nema-17 [Acessed 5 Aug. 2018].