Journal of Engineering and Applied Sciences 12 (2): 214-225, 2017
ISSN: 1816-949X

Implementation of Smart Mobile Parking Through Android Application Program

Syahrul and Ahmad Sobirin
Department of Jurusan Teknik Komputer, Univesitas Komputer Indonesia, Bandung, Indonesia

Abstract: In this study presented the results of research on the design and implementation of a system called smart mobile parking. The purpose of this research is to make the car parking system electronically and automatically. With this system is expected to facilitate the users to park their vehicles. The system is designed to apply the principles of IoT (Internet of Things). In addition, this system has the function to monitor the parking area, reservations and electronic payments (billing). To build this system required hardware is Raspberry Pi as a processor and an infrared sensor module is used to detect the presence of cars. Software applications that are built based on android and database. Tests on the hardware and the software is successfully carried out. The presence of cars in parking places can be detected by infrared light. Booking parking places conducted by the android application program. This android application is also supported by the database system. Overall the system has worked, ranging from infrared sensor readings in real time to the process of booking a place (reservations) and also on the electronic payment system.

Keywords: Parking, car, IoT, android, system

INTRODUCTION

With the large number of private vehicles, especially four-wheel vehicles, causing the car park becomes an important thing, especially in large buildings that exist in Indonesia (Liu *et al*., 2016). Therefore, the management of car parking should continue to improve its services, so that the parking system will be functioning optimally.

To improve the services of the parking system, it is necessary to evaluate the system already implemented. Starting from the improvement of parking management system and also the convenience of the user (Liu *et al*., 2016).

A parking system that regular will make the users become comfortable parking to park his vehicle. A good parking system that is providing the information the availability of the number of parking slots, easy to reserve (booking) a spot in realtime as well as ease of payment (billing processing). This issue should be supported by the development of technology in order to increase parking services well.

To overcome these problems, designed a system of smart mobile parking, by applying IoT (Internet of Things). This system operates in a manner M2M (Machine to Machine) which is the operating principles of smart mobile parking. The hardware is based on monitoring the parking area using a simple sensor module is an infrared sensor.

Users can monitor and select an empty parking slot in a manner mobile/online through its gadgets. Booking a parking area can be done at that time by clicking on the picture menu available parking area (on-screen display), recording payments (billing) for customers already started at that time by reducing the deposit money that had been deposited previously. While for non-customers can only monitor the empty parking area (could not make a reservation).

Literature review

Controller: One of the controllers that are currently widely used is the Raspberry Pi. Raspberry Pi is a Single Board Computer (SBC) with a small size that can be used for many things, just like an ordinary computer (Rakhman, 2014). Raspberry Pi could also be used to control more than one device, both near and distance (Bradbury and Everard, 2014). In contrast to the arduino, Raspberry Pi can control more than one unit of the device you wish to control (McRoberts, 2009). To control a piece of equipment, Raspberry Pi using Python language as a programming language (Dennis, 2013). Specifications of Raspberry Pi 2 is shown in Table 1.

Port on the Raspberry Pi 2 has many features that is 4 USB slots, 1 ethernet port, an HDMI port for connecting to a monitor, display connector serves to display the internal screen on Raspberry Pi and camera connector to connector to be connected to the internal camera on the Raspberry Pi. Figure 1 is shown physical image of the Raspberry Pi 2.

Corresponding Author: Syahrul, Department of Jurusan Teknik Komputer, Univesitas Komputer Indonesia, Bandung, Indonesia

Table 1: Specifications of the Raspberry Pi 2

Product name	Raspberry Pi 2, Model B
Chip	Broadcom BCM2836 SoC
Architecture	Quad-core ARM Cortex-A7
Clockspeed	900 MHz
GPU	Dual Core VideoCore IV®
Memory	1 GB LPDDR2
Operating system	Linux/Windows 10 IoT Core
Power	Micro USB Socket 5V, 2A
Connectors	Ethernet, HDMI, Audio Output, 4×USB, GPIO, Camera Connector, JTAG, Display Connector DSI (Display Serial Interface), Memory Card Slot

Fig. 1: Physical image of Raspberry Pi 2

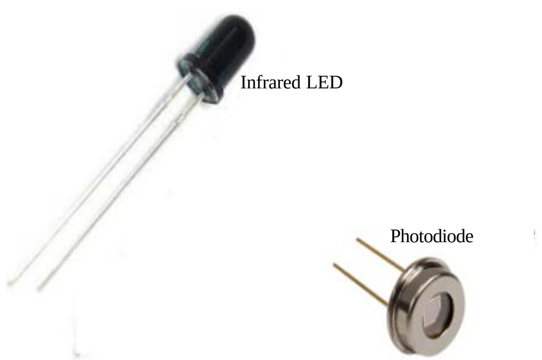

Fig. 2: Physical image infrared LED and photodiode

Infrared led and photodiode: Infrared light is electromagnetic radiation. Infrared light has a longer wavelength than visible light, but shorter than radio wave radiation. Infrared light is not visible. Infrared light can be used to detect solid objects. For this purpose, the infrared LED is used as a transmitter and a photodiode as the receiver. In Fig. 2 are shown the physical image infrared LED and photodiode.

Common infrared LED that emits infrared rays has the same appearance with visible light LED. Its appropriate operating voltage is around 1.4 V and the current is generally <20 mA. Current limiting resistances are usually connected in series in the infrared LED circuits to adjust the voltages, helping the LEDs to be adapted to different operating voltages.

In order to lengthen its controlling distance, infrared LED should be operated under pulse state as the effective transmitting distance of the pulsed light (modulated light) is in proportion with the wind-induced current of the pulses. Thus, by increasing the peak value (Ip) of the pulses, the emitting distance of the infrared LED can also be lengthened. One way to increase Ip is to diminish the duty ratio of the pulse that is to reduce the width of the pulse (T).

The controller with infrared LED can emit infrared rays to take control of correspondent unit and at the controlled unit end, there is also a receiving device to turn the infrared light into electricity such as infrared light receiving diode, photoelectric triode and so on. Emitting and receiving matched infrared diode has also been applied in practical use.

There are two emitting-receiving modes for infrared LED and the controlled unit, one is direct light emitting mode and the other is reflecting light mode. In the direct light emitting mode, the emitting diode and the receiving diode are installed in the emitting end and the controlled unit end respectively, with a certain distance between them. As to the reflecting light mode, the lighting diode and the receiving diode are in parallel. Only when the infrared rays emitted by the diode were reflected by something can the receiving diode get the infrared rays, thereby stimulate the controlled unit to operate. Besides, infrared emitting circuit with double diodes bears higher power and longer functional distance.

A photodiode is a semiconductor device that converts light into current. The current is generated when photons are absorbed in the photodiode. A small amount of current is also produced when no light is present. Photodiodes may contain optical filters, built-in lenses and may have large or small surface areas.

A photodiode is a kind of light detector which involves the conversion of light into voltage or current, based on the mode of operation of the device. It consists of built-in lenses and optical filters and has small or large surface areas. With an increase in their surface areas, photodiodes have a slower response time. Conventional solar cells, used for generating electric solar power are a typical photodiode with a large surface area.

Stepper motor: Stepper motors are DC motors that move in discrete steps (Kadir, 2014, 2013). They have multiple coils that are organized in groups called "phases". By energizing each phase in sequence, the motor will rotate, one step at a time. With a computer controlled stepping you can achieve very precise positioning and/or speed

Fig. 3: Physical image of a stepper motor

control. For this reason, stepper motors are the motor of choice for many precision motion control applications. Stepper motors come in many different sizes and styles and electrical characteristics.

The stepper motor is one type of motor that is widely used today as an actuator. The stepper motor is a DC motor that has no commutator. Generally stepper motors have only the coils in the stator while the rotor is a permanent magnet section. Because of this construction, the stepper motor can be positioned in a certain position or rotating in the desired direction. The stepper motor can spin or rotate with a step angle which can vary depending on the motor used. Step size may be in the range 0.9-90°C. For example, step angle 7.5°, 15°, 30° and so on depending on the desired application or needs. Figure 3 shows the physical image of a stepper motor.

LCD display 16×2: LCD stands for liquid crystal display. Many multinational companies like Philips Hitachi Panasonic make their own special kind of LCD's to be used in their products. All the lcd's performs the same functions (display characters numbers special characters ASCII characters, etc). Their programming is also same and they all have same 14 pins (0-13) or 16 pins (0-15). The 8-Data pins carries 8-bit data or command from an external unit such as microcontroller. Figure 4 are shown the physical image of a LCD display.

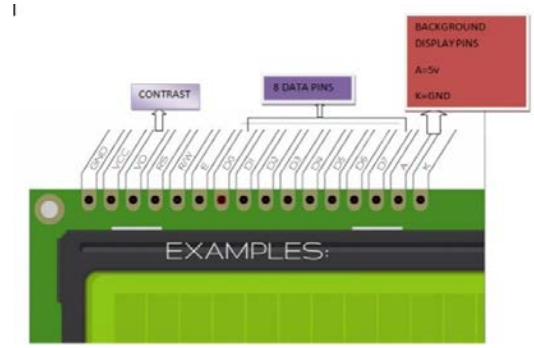

Fig. 4: Physical image of a LCD display

V_0 (set LCD contrast): Set LCD contrast here. Best way is to use variable resistor such as potentiometer. Output of the potentiometer is connected to this pin. Rotate the potentiometer knob forward and backward to adjust the LCD contrast.

RS (Register Select): Their are two registers in every LCD:

- Command register
- Data register

Command register: When we send commands to LCD these commands go to command register and are processed their. Commands with their full description are given in Fig. 4. When RS = 0 command register is selected.

Data register: When we send Data to LCD it goes to data register and is processed their. When RS = 1 data register is selected.

RW (Read -Write): When RW = 1 We want to read data from LCD. When RW = 0 we want to write to LCD.

EN (Enable signal): When you select the register (Command and Data) and set RW (Read-Write) now its time to execute the instruction. By instruction i mean the 8-bit data or 8-bit command present on data lines of LCD. This requires an extra voltage push to execute the instruction and EN(enable) signal is used for this purpose. Usually we make it en = 0 and when we want to exectue the instruction we make it high en = 1 for some milli seconds. After this we again make it ground en = 0. Data which we send to our LCD can be any alphabet (small or big), digit or ASCII character.

Keypad 4×4: Logic and interface of a matrix keypad (4×4) with microcontroller to reduce the number of port pins

Fig. 5: Physical image of a 4×4 matrix keypad

required to read a certain number of inputs (digital). The same logic applies to any matrix keypad of order N×N. Where, N is the order of the matrix.

Why matrix keypad?: Figure 5 shows the physical image of 4×4 matrix keypad. Typically one port pin is required to read a digital input into the controller. When there are a lot of digital input that has to be read, it is not feasible to allocate one pin for each of them. This is when a matrix keypad arrangement is used to reduce the pin count.

Therefore, the number of pins that are required to interface a given number of inputs decreases with increase in the order of the matrix. Example: if the matrix is 2×2, you will need 2 pins for the rows and 2 pins for the columns in such a case there is no difference in the cost of reading that many inputs. But if you consider a 10×10 matrix you will just need 20 pins (10 for the rows and 10 for the columns) to read 100 digital inputs. From the circuit you can see that of one of the 16 buttons are pressed, a pair of pins are connected together. We will used this feature to detect with button was pressed.

MATERIALS AND METHODS

Mechanical frame design: In this research, we made three parking slots. Each parking slots required two infrared sensor module, two parking dispenser and two doorstop. Figure 6 shows the results of the design in the picture looks the parking slots that will be made along with the size and dimensions. Parking pillar designed to have a height of 100 cm and a width of 4×4 cm. This size was chosen because the average height of the car is not >170 cm, so that this measure is sufficient to detect the

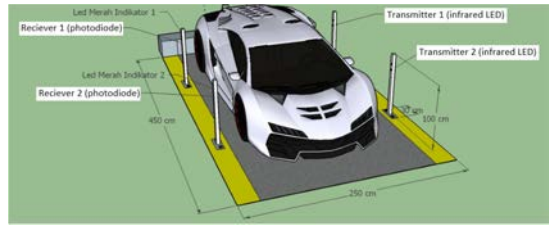

Fig. 6: The results of the design and size of the parking slot

presence of cars. In each parking slot there are four pillars, two parking pillars serves as a transmitter and the rest as a receiver.

Figure 7 shows the results of the design of parking dispenser. Parking dispenser boxes, pillars and frame support. This dispenser box length is 30 cm width is 30 cm and the height is 50 cm. The length of pillar is 100 cm and the supporting frame has a size of 15×15 cm. In dispenser box has an LCD and a keypad.

LCD serves as an interface unique code that is inputted by the user. While the keypad functions as a button, to enter the unique code from which booking system has been provided by the system through the gadget. In the box dispenser has a "Raspberry Pi" which serves as a controller.

Parking Portal has two components, namely boxes and crossbar at the entrance and exit. Box size length 38 cm, width 38 cm and height 111 cm. On the crossbar there is a stepper motor that serves to lift the crossbar by opening and closing by 90 degrees. Figure 8 shows the results of the design of a crossbar on the door.

Block diagram of system: The following describes the block diagram of the hardware design of the system that has been built (Fig. 9). The following will explain the block diagram of the hardware design in detail. From Fig. 8 can be seen that the main controller in the design of this system is the Raspberry Pi. Infrared sensor and the keypad is the input of the system. In addition, the system is also driving the actuators such as stepper motors as well as displays the character code on LCD 16×2. In this design, the Raspberry Pi used is 3 pieces, one for reading 6 infrared sensors and two Raspberry Pi as controlling the entrance and exit.

Operation: Raspberry Pi is a central processing unit. When the user will reserve parking slots of the gadget, the user select from a menu provided, if available slot is empty or not. The presence information of a vehicle in a parking slot is detected by the infrared sensor in the parking lot that is processed by the

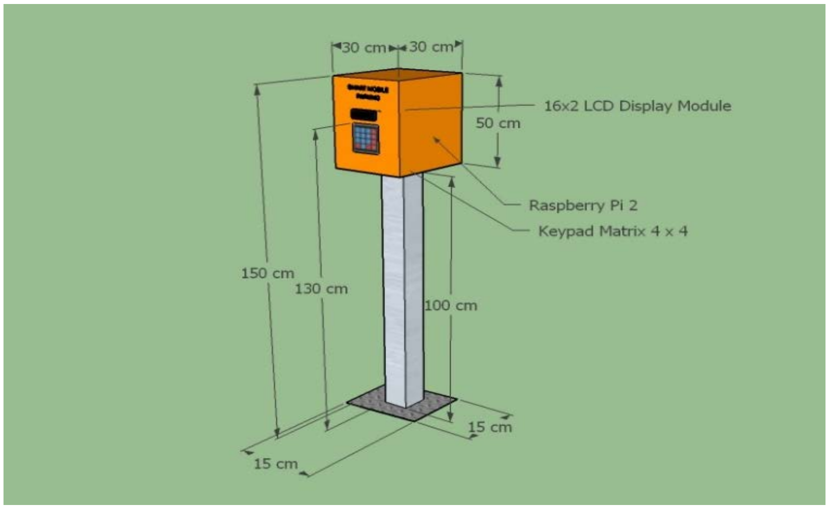

Fig. 7: The results of the design and size of the parking dispenser

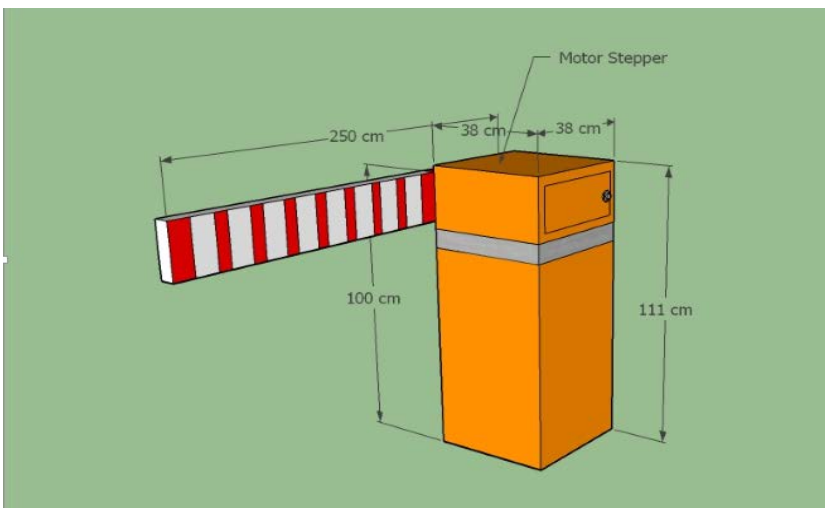

Fig. 8: The results of the design of a crossbar on the door

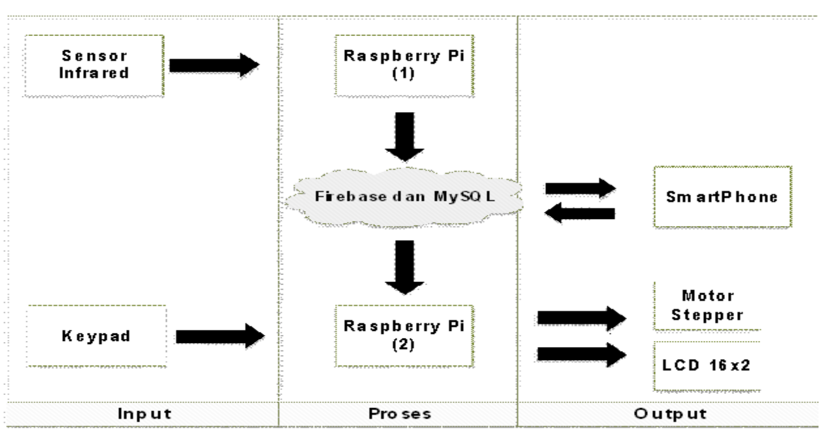

Fig. 9: Block diagram of a simplified system

Table 2: Allocation of the pins used on Raspberry Pi

Devices	Function	GPIO	Status	Connections
Raspberry Pi	As a reader sensors	GPIO 18	Input	Infrared 1 Slot Parkir 1
		GPIO 23	Input	Infrared 2 Slot Parkir 1
		GPIO 24	Input	Infrared 1 Slot Parkir 2
		GPIO 25	Input	Infrared 2 Slot Parkir 2
		GPIO 12	Input	Infrared 1 Slot Parkir 3
		GPIO 16	Input	Infrared 1 Slot Parkir 3
Raspberry Pi	As the crossbar entrance and exit	GPIO 2 (I2C1SDA)	Output	LCD
		GPIO3 (I2C1SCL)	Output	LCD
		GPIO 18	Input	Keypad
		GPIO 23	Input	Keypad
		GPIO 4	Input	Keypad
		GPIO 27	Input	Keypad
		GPIO 22	Input	Keypad
		GPIO 5	Input	Keypad
		GPIO 6	Input	Keypad
		GPIO 13	Input	Keypad
		GPIO 24	Output	Stepper motor
		GPIO 25	Output	Stepper motor
		GPIO 12	Output	Stepper motor
		GPIO 16	Output	Stepper motor

Raspberry Pi. Furthermore, the presence of the vehicle information is sent back to the user via the Raspberry Pi into the gadget.

If the user gets an empty slot and has been chosen, then the payment is calculated and then deposit money belongs to the customers began to be reduced in accordance with the time. If all slots are occupied and none of the empty slot, then the system will send an alert signal to the user. Furthermore, users have to try again at another time. If the users are not members, they could just browsing.

Hardware design: Table 2 shows the allocation of the pins on the Raspberry Pi is used to perform data processing and control systems. On the other hand, there are smartphones (gadgets) which functioned as an input device held by the user. From Table 2, it can be seen GPIO allocation is as a sensor reader. GPIO ports 18, 23, 24, 25, 12, 16 is used to read input from the infrared sensor. While on the Raspberry Pi which serves as a crossbar controller entrance and exit. GPIO ports 2 and 3 are used for output to the LCD 16×2 with an I2C chip, so it takes only two GPIO. GPIO ports 18, 23, 4, 27, 22, 5, 6, 13, 24, 25.12 and 16 is a GPIO allocation for Keypad. The allocation for the keypad is very much dependent on the number of buttons used. The design of the keypad used is a 4×4 matrix keypad. GPIO ports 24, 25, 12 and 16 are used to output the stepper motor.

Infrared module design: For a distance range of approximately three to five meters, beam of infrared light must be modulated to avoid data loss due to noise. The main principle of infrared sensor circuit like a switch that gives the voltage changes when there is a barrier between the transceiver and receiver. Infrared sensor module has two devices, namely transmitter circuit (infrared LED) and receiver circuit (photodiode). The transmitter circuit emits infrared light, then radiance is received by the receiver (photodiode) so that the voltage generated is equivalent to the ground voltage. And conversely if it does not receive an infrared light beam, it will produce a voltage greater than ground.

Figure 10 is shown a schematic of infrared sensor and connection with Raspberry Pi. In the schematic, there is a comparator circuit (IC LM741). In this comparator circuit compares the input photodiode (pin 3) with voltage of the variable resistor (pin 2). When photodiode obtain infrared light, then the output on pin 6 will generate high. Conversely, if the photodiode infrared light is not obtained, then the output from pin 6 will generate low. There is also the IC 555 as a pulse generator to trigger infrared light.

Keypad interface design: Figure 11 shows the keypad connection with the Raspberry Pi. In this design, the 4×4 keypad is used to enter a unique code that is received from the android application (Leea, 2013; Juhara, 2016). Pins required on the keypad are eight. In this design, only a keypad module that serves to enter the unique code on the system. The system will compare with the existing code in the database. When the code is the same, the stepper motor will open the crossbar with a 90°C rotation.

LCD interface design: LCD is used to display a unique code that is entered by the user via the keypad. LCD is placed on each dispenser box entrance and exit. Figure 12 is shown 16×2 LCD connection with the Raspberry Pi.

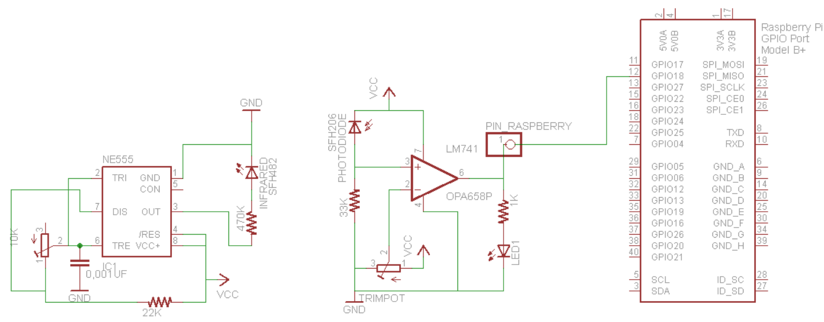

Fig. 10: Schematic of infrared sensors and connections with Raspberry Pi

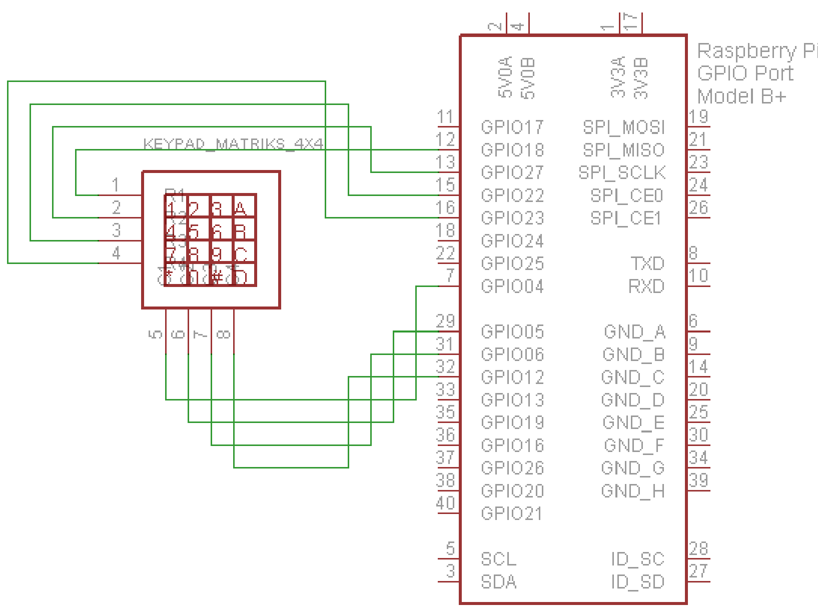

Fig. 11: Keypad connection with the Raspberry Pi

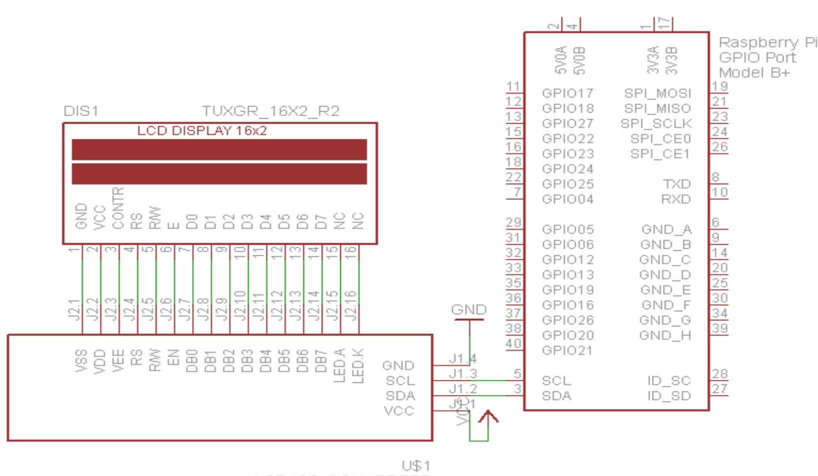

Fig. 12: LCD connection with the Raspberry Pi

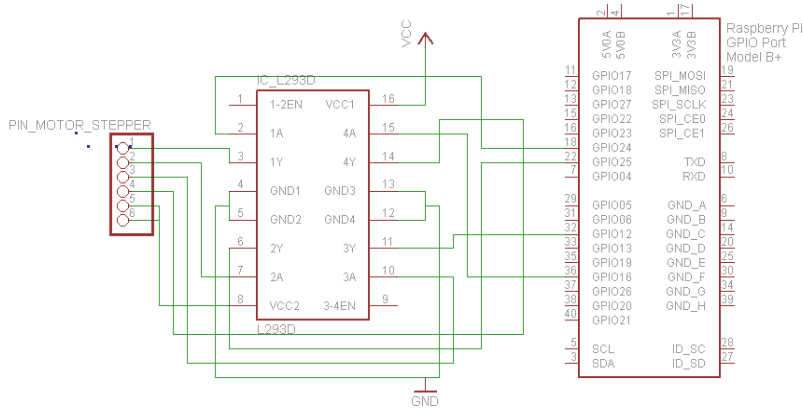

Fig. 13: The connections of stepper motor with L293D and Raspberry Pi

LCD display on the dispenser box using additional components, namely I2C chip. The use of I2C aims to cable routing of Raspberry Pi is reduced, using only two cables. Use of I2C can only be used for one LCD only. Pins used on the Raspberry Pi is both SDA and SCL pins contained in GPIO GPIO 2 and 3.

Stepper motor interface design: Figure 13 shows the connection of stepper motor with L293D and Raspberry Pi. Stepper motors are used as actuators for entrance and exit bars. Required a driver chip (L293D), to drive the motor, because the stepper motors using relatively large currents while the output current to be drawn from the Raspberry Pi is not sufficient. L293D has 4 input pins, which are connected to the GPIO 24, 25, 12 and 16 on the Raspberry Pi. L293D has four output pins are 3, 6, 14 and 11.

Software design: In the software, the design is divided into two parts, namely software design on the Raspberry Pi and android app design.

The following is an algorithm (steps) of the software on the Raspberry Pi (Algorithm 1):

Step 1
Start
Step 2: GPIO Initializations on Raspberry Pi; Its function is to read the GPIO pins that will be used, so that Raspberry Pi can recognize the functions on those pins
Step 3: display an input message to a unique code on LCD; Its function is to display a message so that the user can enter a unique code on the LCD panel
Step 4: inputs to the unique code on the keypad; In this study the user must enter the unique code on the keypad
Step 5: Check the input from the keypad; Checks whether the input is equal to the database? If yes, then display a message that the code is valid and continue, if not, then display a message that the code is invalid and back again at the entry to enter a unique code (to step 4)
Step 6: Stepper motors receive an electrical voltage to run; If the unique code is valid, it will activate the electric voltage on the stepper motor on the crossbar
Step 7: Crossbar of the door is opened; Stepper motor will be unbarred at an angle of 90° and after the car passed then closed again
Step 8: Infrared sensor detects the presence of the car
Step 9: End

The following is an algorithm (steps) of the software on the android app (Algorithm 2):

Step
Step 1: Start
Step 2:open the app "Smart Mobile Parking"; Opening the smart mobile parking application on android smartphone
Step 3: check account?; Check whether the user has an account application? If there is, please login, if not, create an account
Step 4: Login; Enter your username and password
Step 5: Check whether the login is successful?; Check whether the login is successful? If successful, then go to the main menu, if it does not work, go back to the login menu (to step 4)
Step 6: The main menu of Smart Mobile Parking; The main menu contains the balance amount and parking plan
Step 7: Check the empty parking slot? Balance is sufficient or not?; Check whether the slot is empty and the balance is sufficient? If yes then go into the booking form, otherwise return to the main menu.
Step 8: enter the vehicle identification number and type of car
Step 9: Submit
Step 10:Parking Time Display; The time display will run backwards in accordance with the time of booking
Step 11: Display Unique Code; A unique code obtained from database
Step 12: Ends Parking?; Check whether the user ends the parking? If yes, then calculate the cost of parking, if not, go back to the parking time display
Step 13: counting parking time; Calculate how long a user to park their vehicles
Step 14: reduction in cash balance; Reduce balances in accordance with the length of time
Step 15: end

RESULTS AND DISCUSSION

Data analysis: This study will be explained the results and analysis on every block of the hardware components and software components. Each block of the hardware will be tested whether it is appropriate to the function or not. The hardware will be tested ie infrared sensor module. In addition to hardware testing as well as testing and

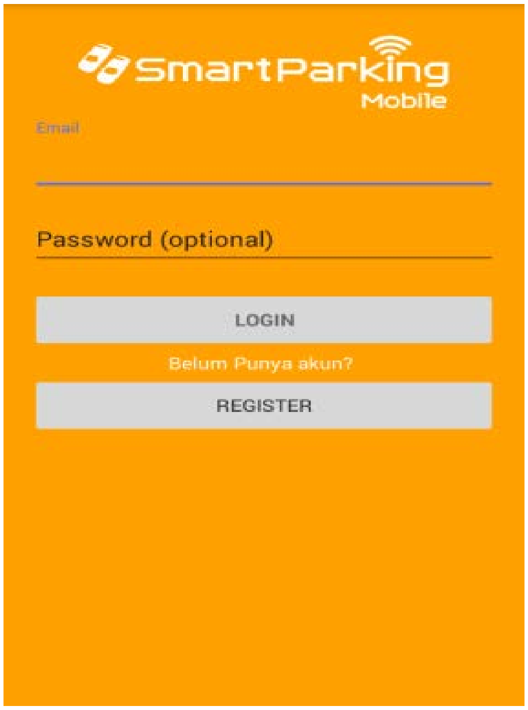

Fig. 14: Display of login page

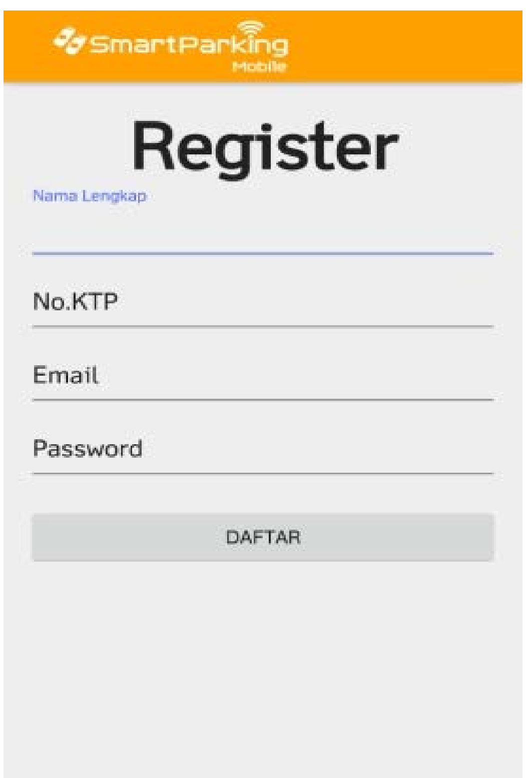

Fig. 15: Display of register page

Fig. 16: Display of parking slots

analysis of the software (android). This experiment is testing alpha using blackbox. Blackbox testing method focuses on the functional purposes of the application that has been built. Infrared sensor module has been able to operate when tested. Infrared light can be detected by a photodiode with a distance of about 4 m. This means it can recognize the car body width about 4 m and this is already sufficient.

The following describes the stages and the results of testing on android applications that run on android smartphones. The test is make a reservation to the parking slot. Figure 14 is a display of the page to login. On this page, users are required to enter the username and password that have been made previously. If users do not already have an account, required to create an account by pressing the button register. If users do not already have an account, users are required to sign up to the register menu. This menu allows the user to fill information full name, ID number, email and password number as in Fig. 15.

After display the login form, it will display the home menu. In the home menu there are parking slots that serves to provide information to the user which slots are empty as shown in Fig. 16. There is a slide on the home

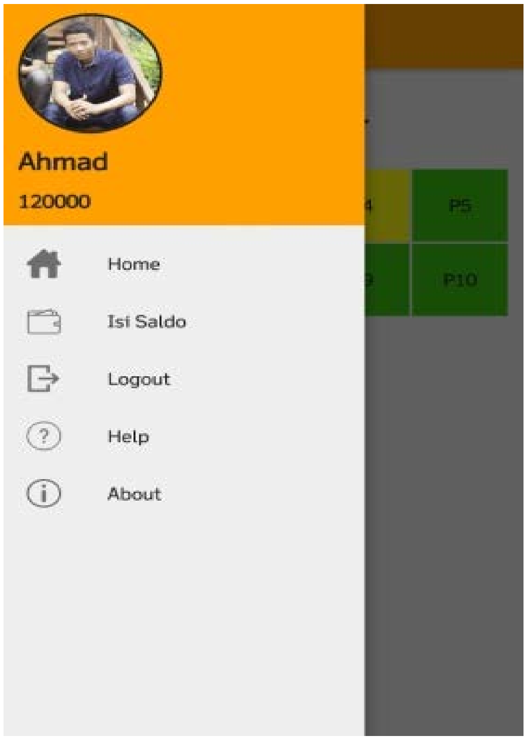

Fig. 17: Display of slide menu

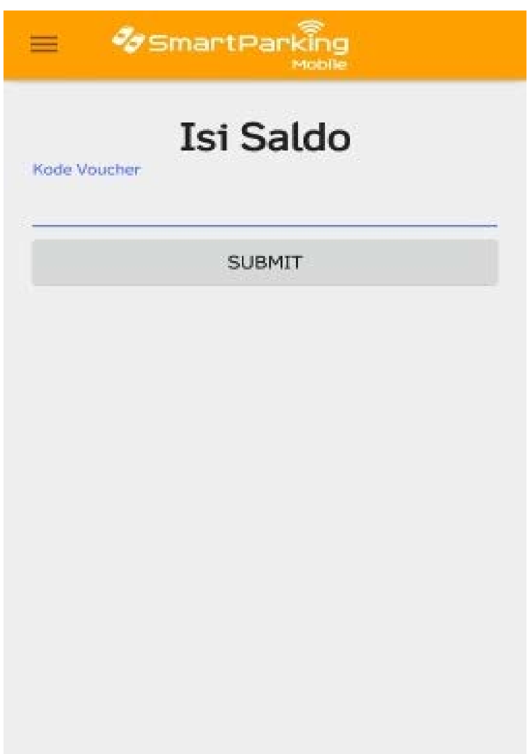

Fig. 18: Display of balance money

Fig. 19: Display of confirmation of balance refill

menu to simplify navigation of this application, the menu contains some features that go back to the menu button, the contents of balance, logout, help and about (Fig. 17). In the home menu the user can fill the balance in advance by pressing the contents of the balance on the home menu. Users simply enter the voucher code previously purchased on the owner of the parking lot and press the submit button (Fig. 18).

When the user presses the submit button it will show a confirmation message as shown in Fig. 19. If the user is already filling the balance of money, users can reserve an empty parking slot. It will display the parking booking, the user must fill a number of vehicles, the brand and type of car and then press the submit button as in Fig. 20. The next display is the time of booking. Users are given a unique code that will be automatically entered in the parking dispenser. To end parking, users can simply press the finish button, the display will return to the main page as shown in Fig. 21.

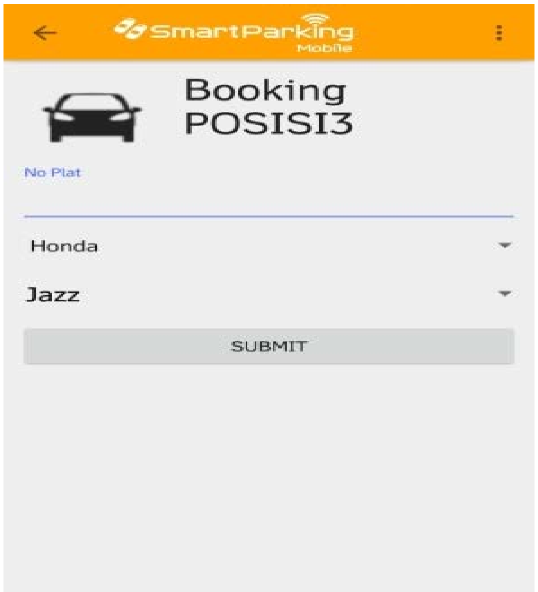

Fig. 20: Display of booking a parking slot

Fig. 21: Display the time of booking on android App

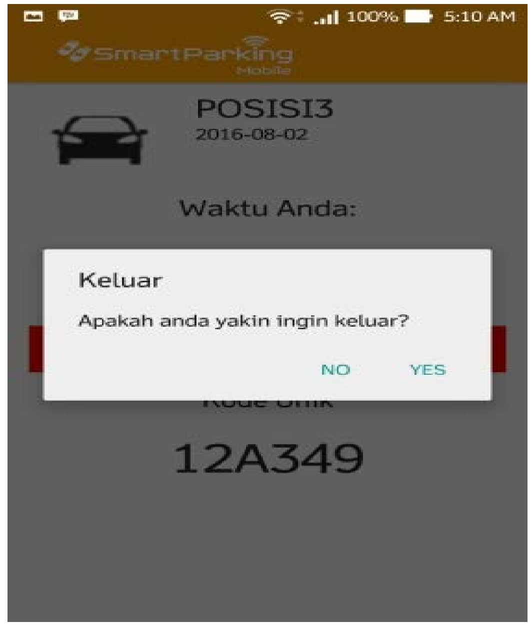

Fig. 22: Display of the confirmation message that parking is terminated

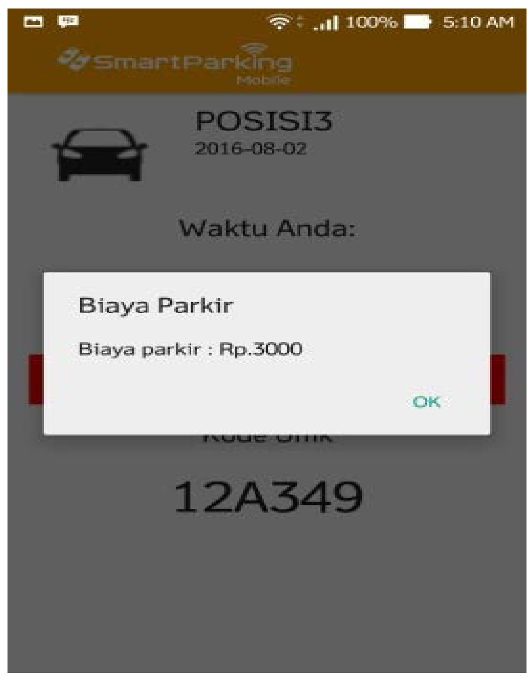

Fig. 23: Display of the parking fee

If the user presses the park is completed, it will display a confirmation message to end the parking time as in Fig. 22. When the user presses YES, then the system will automatically calculate the parking fee and will subtract the amount of your balance as in Fig. 23. All of the above menu has been successfully tested and can operate with a good. Weakness infrared sensor module that we have designed are within range. Distance reaching only about 4 m. This can affect less flexibility on when they want to park the vehicle. Moreover, if the width of the car is somewhat larger. This occurs, due to circuit of infrared light transmitter. To enlarge the range of distances, it is advisable to used the circuit of modulated infrared light transmitter.

CONCLUSION

Users can access the data occupancy to determine the availability and then pay for them with their mobile phones. Smart mobile parking improve the utilization of existing parking which leads to ease of monitoring and searching for a parking slot available, other benefits are ease of payment, greater revenue for the owners of parking.

The system benefits of smart mobile parking go well beyond avoiding the needless circling of city blocks. It also enables cities to develop fully integrated multimodal intelligent transportation systems.

Developing smart parking solutions within a city requires data standardization and management; mobile phone integration; hardware and software innovation and coordination among various stakeholders.

REFERENCES

Bradbury, A. and B. Everard, 2014. Learning Python with Raspberry Pi. John Wiley & Sons, UK., Europe, ISBN:978-1-118-71703-5, Pages:161.

Dennis, A.K., 2013. Raspberry Pi Home Automation with Arduino. Packt Publishing, Birmingham, England,.

Juhara, Z.P., 2016. The Complete Guide to Programming Android. Andi Publisher, Yogyakarta, Indonesia,.

Kadir, A., 2013. Practical Guide to Learning Application and Programming Using Arduino Microcontroller. Andi Publisher, Yogyakarta, Indonesia,.

Kadir, A., 2014. From Zero to A Pro Android. Andi Publisher, Yogyakarta, Indonesia,.

Lee, W.M., 2013. Android™ Application Development Cookbook: 93 Recipes for Building Winning Apps. John Wiley & Sons, Indianapolis, Indiana,.

Liu, Z., W. Chen and Z. Li, 2016. Finding your car in the car park: Only use a smartphone. J. Adv. Technol. Eng. Res., 2: 87-99.

McRoberts, M., 2009. Beginning Arduino. Apress, New York, USA.,.

Rakhman, E., 2014. Raspberry Pi that Tiny Microcontroller Solutions Can. Andi Publisher, Bandung, Indonesia,.