IoT BASED CONTROL OF LAUNDRY MACHINES

(Video URL: https://www.youtube.com/watch?v=3gmbnz073Cs)

Anchit Pandey,
Bits Pilani, Dubai Campus,
f20150050@dubai.bits-pilani.ac.in

Vrushali Patil,
Bits Pilani, Dubai Campus,
f20160079@dubai.bits-pilani.ac.in

Abstract

In this paper we introduce a novel framework for the IoT based control of laundry machines (washers and dryers) found in laundromats, through mobile device or a computer. The paper presents a unique way of activating the laundry machines through a Django based web application. A custom PCB is designed which interfaces between the machines and the server through a raspberry pi. The PCB facilitates the data transfer to and from the machines. A script running on the raspberry pi accepts and sends command to the web server which in turn makes specific GPIO pins of the raspberry pi high or low thus activating the particular machine. The user can also view the status of his washing or drying cycle in a similar fashion.

1. Introduction

The idea of creating a Smart-Living is now becoming possible with the emergence of the Internet of Things. IoT is a system of interrelated computing devices, mechanical and digital machines, objects, animals or people that are provided with unique identifiers (UIDs) and the ability to transfer data over a network without requiring human-to-human or human-to-computer interaction. [1] In this paper we develop a systematic architecture based on IoT, for activation of laundry machines. Laundry machines in laundromats are usually self-service and coin-operated. The coin system imposes certain security concerns as fake and bogus coins of nearly same dimensions can be inserted into the coin slider to operate the machines. Further the person in-charge of such machines also faces the issue of collecting and recycling the coins and other bookkeeping tasks of maintaining user list, preparing usage summary and so on. From the user's perspective, they have to come to the laundromat location to check if a machine is free. Once they have started a washing or a drying cycle, they have to periodically check if it is completed. Such laundromat services are currently offered in many universities. Also since it is a student environment, laundry users tend to play pranks (for e.g. using ear buds) to operate the machines, apart from fake coins. Some of these tasks can be automated by having a card-operated laundry machine, where each user is issued a special card or is allowed to use his/her debit/credit card. But to achieve this, each laundry machine needs to be equipped with a card reader and every user must have a card to operate the machine. There are security issues with loss of card. Further, altering the existing machines with card-readers, requires

substantial modifications to the existing hardware and hence increases the overall cost of conversion. Also if special cards are issued to users, then expensive kiosk systems need to be provided to top-up the card. These issues motivated us to consider an alternate solution, where the machine can be operated by the user through a web link from a mobile device. The prime idea is that the user can log into a web portal hosted on a central server, and check if any machine is available. If so, the user can load the machine with his clothes and then operate it by choosing the same machine from the displayed list. The user can also check the status online, whether his/her cycle is completed, without actually visiting the laundromat. Further, bookkeeping tasks like, maintaining the per user usage summary for accounting purposes, keeping track of working/non-working machines etc. can be automated, through the web-portal at the central server.

2. Working of the Coin System Machine

We have carefully examined the existing functioning of the laundry machines before introducing the new architecture. A small control unit is present inside every machine. This unit directs the operation of the dryer and the washing machine. The relevant components from our design's view are the micro switch and the IN USE LED. The IN USE LED states whether a machine is currently in a drying or a washing cycle or in an idle state while the micro switch controls the activation of the machine. The control circuitry provides different platforms for its working like the coin-slider method, Ethernet based control etc. In the existing coin-slider system, a lever is attached to the slider which closes a micro-switch as soon as the coin rolls in. The terminals of the micro-switch are connected to the control circuitry of the machine and when the control circuitry detects that these terminals are shorted, it activates the machine. We observed that the voltage across the terminals of the micro-switch remains at 5V, when the machine is idle (off state). Voltage drops to zero volts when the micro-switch is closed by the lever in the coin-sliding system and as a result the machine is activated. Our custom circuit mimics the shorting of the terminals of the micro switch through an optocoupler on the PCB which is energized when the specific machine is selected to operate. The existing mechanism is depicted in the left while the IOT based solution is shown in the right in the fig 2.1 below.

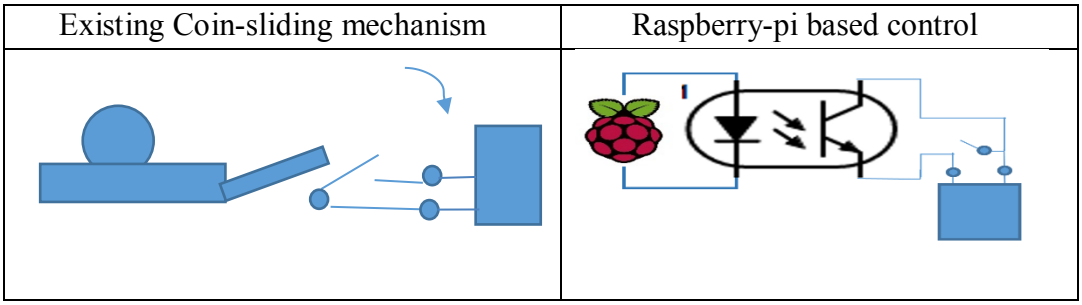


Fig 2.1 Contrast b/w existing system and IOT based system

3. Custom designed PCB for interfacing with the machines

In the view of resolving the aforementioned problems of the coin-slide system we propose a custom designed double sided PCB which activates the machines. The PCB is mounted with a decoder, optocouplers, comparators, RJ45 connectors and a 40 pin GPIO header. The front and back designs of the PCB are shown in Fig 3.1. The schematic diagram of the PCB is shown in Fig 3.2.

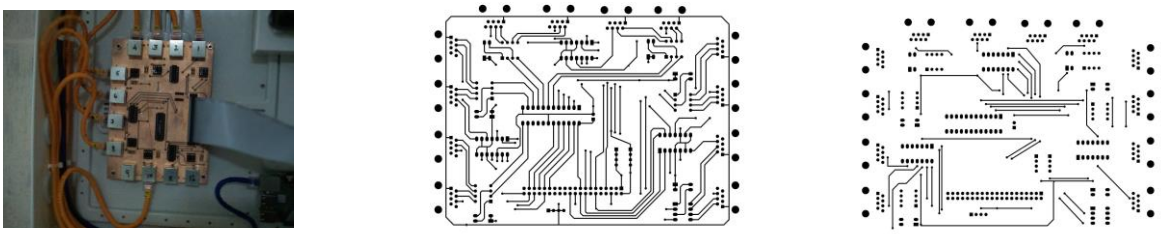


Fig. 3.1 Double sided PCB

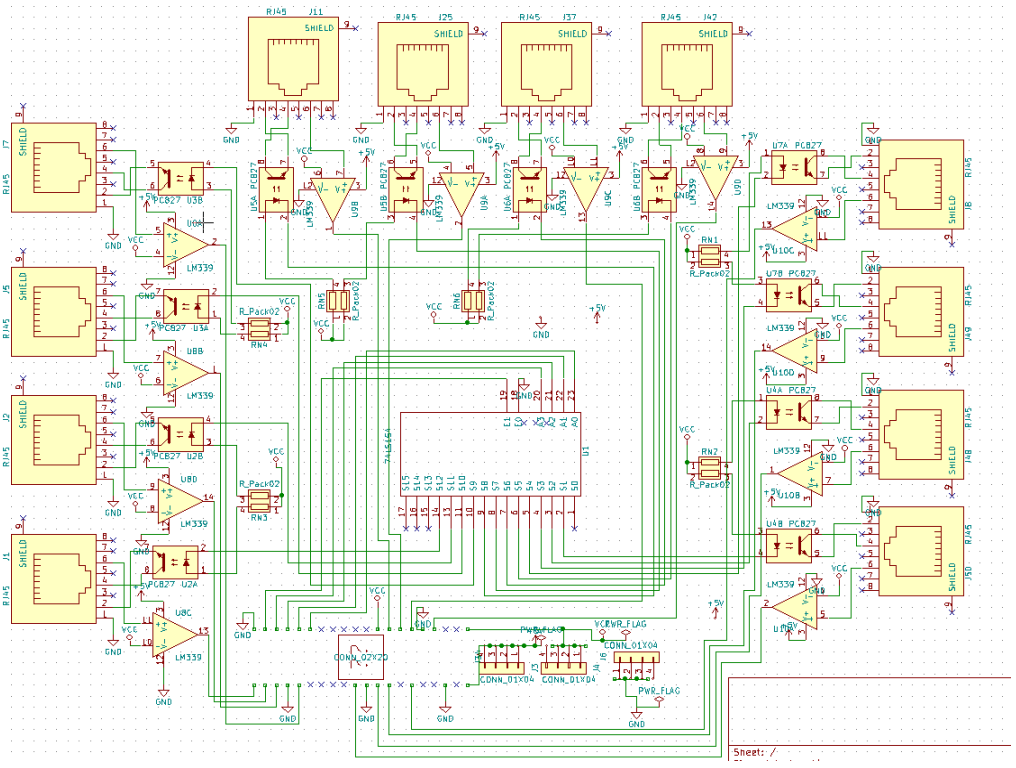


Fig. 3.2 Schematic of the PCB

Let us first discuss the role of each of the components mounted on the PCB circuit.

A. Decoder: 74LS154

A decoder is a combinational logic circuit which is constructed from individual logic gates and it transforms a set of digital input signals into an equivalent decimal code at its output. [2] (Fig. 3.4)

The 74LS154 decoder used in this circuitry is a 4 to 16 line decoder, that is, it has 4 input data lines, 2 strobe inputs to select or enable the decoder and 16 output combinations. This is an active low decoder which means it codes four-binary coded inputs into one of the 16 mutually exclusive outputs at logic "0". Both the strobe inputs G1 and G2 should be low. When either of the input strobe G1 or G2 is high, all the outputs are high (logic 1). The input pins A, B, C and D of the decoder are connected to the GPIO pins 19, 13, 6 and 5 of the raspberry pi respectively. The binary combination of the machine-id (machine number) appearing as output on the GPIO pins is fed to the input data lines of the decoder. On the basis of this binary pattern, specific output of the decoder is pulled low.

B. Optocoupler

Optocoupler is a device that couples two different electric circuits using light. They are typically used in electronic circuits to turn on the load in another circuit without loading the original signal. They contain an infrared LED on the input side and a photodetector such as a photodiode on the output side, combined in one package. It is highly desirable to provide isolation between the main equipment and the PCB circuit to prevent damage in the event of short circuits or triac failure. This is achieved through optocouplers. [3]

We use PC847 and PC827 ICs which are Phototransistor optocouplers. The GPIO pins of the raspberry pi can withstand voltage levels of 0 to 3.3v. Any signal higher than 3.3v and lesser than 0v is undesirable as it can damage the Pi. Hence in our application the optocouplers provide an isolation between a control signal from the raspberry pi and a much higher voltage output signal from the machine. Apart from this, we use it to enable the switching of machines by closing the micro switch within the machines. The infrared led within the optocoupler is forward biased. The positive terminal (anode) of the LED is supplied with constant input voltage. The optocoupler closes when the cathode receives a low input and the machine gets activated. The pins 1, 3, 5, 7 in Fig. 3.5 are always high (3.3v) while pins 2, 4, 6 and 8 are connected to respective decoder output pins. The output pins 16,15 etc. go to RJ45 ports of the corresponding machines which are in turn connected across the terminals of the micro switch via the Ethernet cable as seen in Fig 3.2.

C. Comparator

A comparator is a simple device that compares two input voltage levels. It has non-inverting (V+) and inverting (V-) input voltage pins as input and one output voltage pin. If the non-inverting input is less than the inverting input, the comparator produces a low output voltage. If the non-inverting voltage is greater than the inverting voltage, the output is high. [5] In our case the reference voltage

applied is +3.3V to the V- terminal of the comparators. The V+ terminal is connected to the 6th pin of the RJ45 port for each machine which taps and reads the voltage at cathode of the IN-USE LED. The pins 4, 6, 8 and 10 are connected to 3.3v while pins 5, 7, 9 and 11 as shown in Fig 3.3 tap the voltage at the cathode of the IN USE LEDs for respective machines. The output of the comparator is fed to GPIO pins of the pi where each GPIO pin corresponds to a particular machine as shown in Fig 3.2. When the machine is in use, LED is ON, then the voltage at the cathode of the LED is lesser than 3.3v, i.e. non inverting input (V+) < inverting input voltage (V- = Vref = 3.3v) and the o/p of LM339 is LOW. When the machine is idle, the IN-USE led is OFF, voltage at the cathode is 5v, and the comparator o/p is HIGH. The output of IC LM339 is an open collector meaning that collector pin of a transistor is kept open in this IC which can be connected to any +ve supply rail through a right valued external (pull up or pull down) resistor. It is possible to have pull up resistors in hardware and also using software. The Raspberry Pi has built-in pull-up and pull-down resistors which can be enabled in software (internally). In the RPi.GPIO module we configure this using the GPIO.PUD_UP. Thus, open collector can be used to interface two different circuits of different operating voltage. i.e., The IC may be 5V, but Open Collector can be connected with 3.3V supply.

D. RJ45 Ports

A RJ45 connector, used for Ethernet networking, comprises of 8 pins to connect the 8 separate wires present in an Ethernet cable. The optocoupler output pins are connected to pins 2 and 4 of RJ45 port which then transmit the data via the Ethernet cable to close the micro-switch. To read the status of the machine, the cathode of IN-USE LED is tapped and the voltage is received at pin 6 of RJ45. This is then fed as input voltage to the V+ (non-inverting) terminal of the comparator. (Fig 3.6) [4]

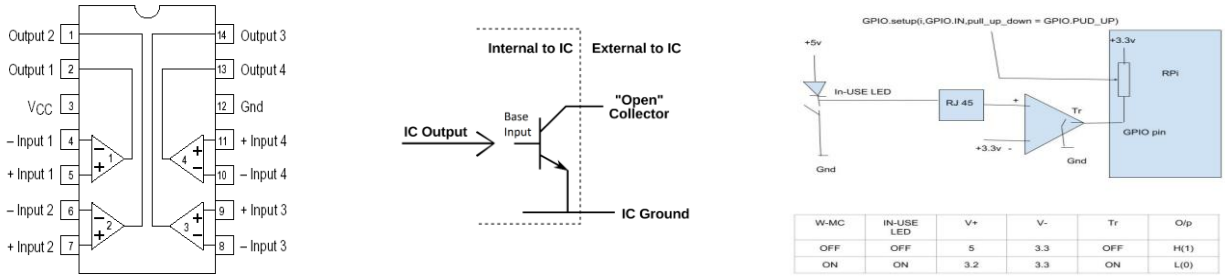


Fig. 3.3 Comparator

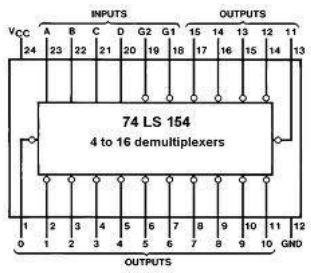


Fig. 3.4 Decoder

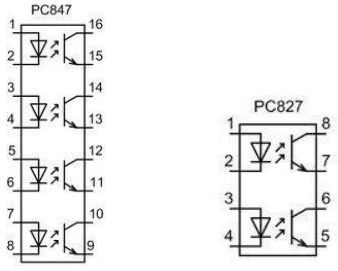


Fig. 3.5 Optocoupler

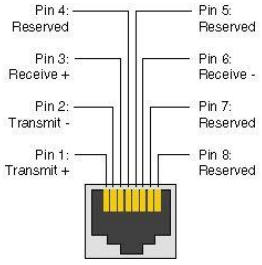


Fig. 3.6 RJ45 port

4. Raspberry Pi as a client and a server

Raspberry Pi is a mini computer and a powerful platform to build IOT based projects as it aids in fast prototyping and building of projects. It offers a complete Linux server in a tiny platform which is capable of multitasking, serving files, and communicating with other networked computers and cloud platforms. We use the Raspberry Pi 3 B+ model for this system. The General Purpose Input Output (GPIO) pins are connected to different components on the custom PCB board. Few pins act as an output to the custom board while few read input from the different machines via the custom PCB. The numerous components of the custom PCB draw power and ground from the Vcc and ground of the Pi. In short it acts as an interface between the custom PCB and the Django webserver.

The Raspberry Pi Model 3 B+ is connected to the PCB (designed with Kicad Software) via the 40 pin connector on one end and wirelessly to the Institute Server on the other. The Pi is provided a static IP address and is connected to the Wi-Fi Network of the institute. It runs a continuous python server-client socket connection script. When the user activates the machine, the machine id of that machine is sent to the raspberry pi. The script running on the pi converts the machine id (decimal number) into its corresponding binary representation and stores the value in variables A0, A1, A2 and A3 where A0 represents the Least Significant Bit (LSB) and A3 the Most Significant Bit (MSB). The GPIO pins 19, 13, 6, 5 are programmed as output pins and are fed with values of the variables A0, A1 A2 and A3 respectively. These variables are input to the decoder on the PCB as shown in Fig 3.2. The GPIO pins that correspond to the machine-id in binary are turned high which then activate the pins on the PCB via the 40 pin connector. The flowchart shown in Fig 4.2 depicts the working of the pi server and activation of machine in short.

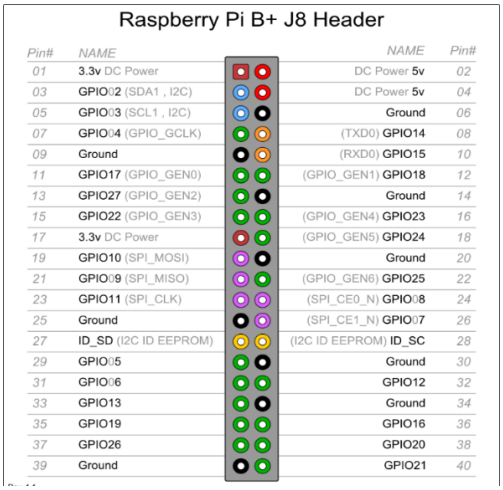


Fig. 4.1 Raspberry Pi GPIO Header

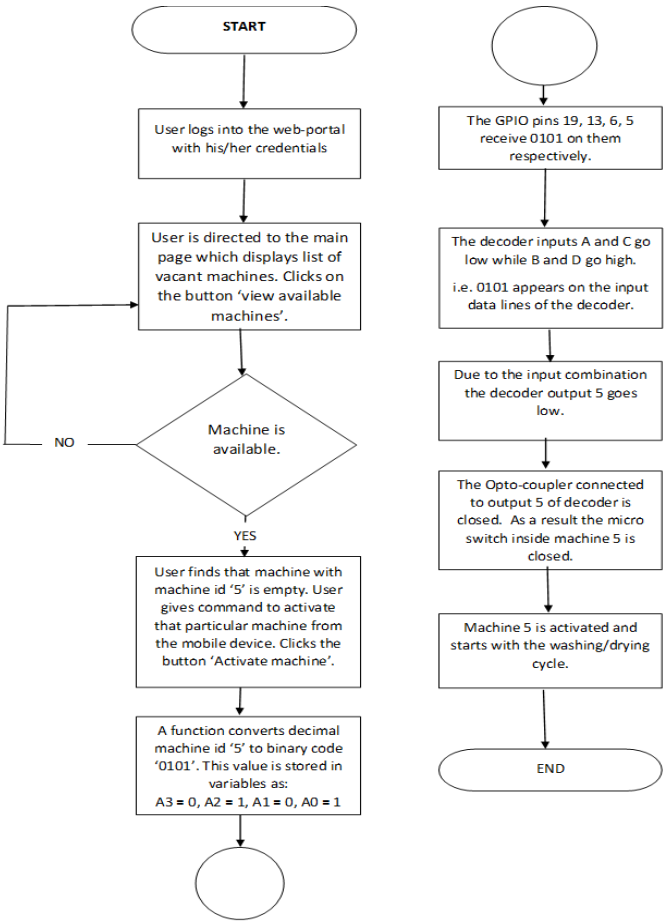


Fig. 4.2 Steps for activation of machine

5. Django based web application and the database

Django is a free open source web application framework that is written in python programming language. It is a collection of certain modules that help to ease the development of web based applications. It provides functionalities like contact forms, comment boxes, authentication support, admin panels, file upload support. These functionalities are already built in and hence it is quite easy to configure them to meet the requirements of the web application.

For this project we have created a web application coded on Django framework, for which the user can log into and access the laundry machines of the laundromat. The web application is hosted on the institute server. The institute sever can be accessed using the student Wi-Fi thereby allowing the users in the campus to log into their accounts. Initially, an account is created for every user. Every user is assigned a random username and a password. The users are informed about their username and password via an email. The database used for storing the information of the users is the built in Django database PostgreSQL. The tables stored in the database for the project are shown in the Figure 5.1 below.

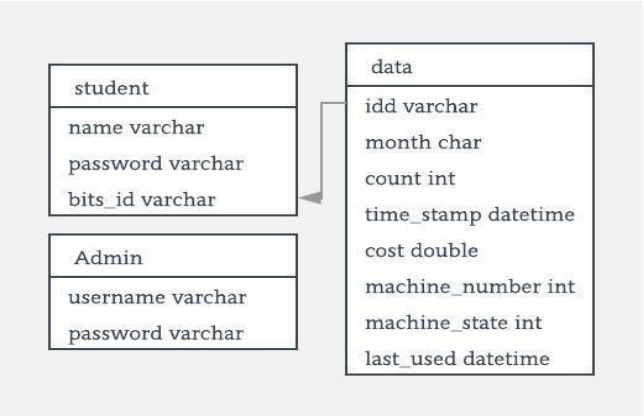


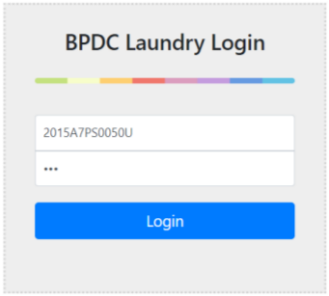
Fig. 5.1 Tables stored in database

The following are the description of the fields as described by the Fig. 4.2

a. name: Stores the full name of the user
b. password: Stores the password of the user's account
c. bits_id: Unique ID assigned to all the users (Primary Key)
d. username: Stores Username of the Administrator
e. idd: Foreign key that references the bits_id field
f. month: The current month in which the application is being used
g. count: Total number of the times the user activated the machines in a semester
h. time_stamp: The current timestamp
i. cost: The additional charges incurred on the user for exceeding the count level
j. machine_number: Indicates the machine id that the user has selected
k. machine_state : indicates whether the user's machine is active or not
l. last_used : indicates the timestamp when the user last used the machine

The Fig 5.2a shows the login page of the application. The user logs in with his unique id and password. After logging in, the user is directed to main screen as shown in Fig 5.2b. The user can view his fines, the number of times he used and the last used date. There are 3 buttons:

a. View Machine: Shows a list of machines that are in ideal state (not in use) in the laundromat
b. Activate : Activates a machine selected from the list
c. View History: Shows the laundry history of the user.



Here is your History

Month	Cumulative_Month_Freqeuncy	TimeStamp	Penalty
December	1	Dec. 26, 2017, 6:54 p.m.	0.00
December	2	Dec. 27, 2017, 4:59 a.m.	0.00
December	3	Dec. 27, 2017, 5:04 a.m.	0.00
December	4	Dec. 27, 2017, 5:09 a.m.	0.00
December	5	Dec. 27, 2017, 5:09 a.m.	0.00
December	6	Dec. 27, 2017, 5:09 a.m.	0.00
December	7	Dec. 27, 2017, 5:09 a.m.	0.00
December	8	Dec. 27, 2017, 5:09 a.m.	0.00
December	9	Dec. 27, 2017, 5:12 a.m.	2.50
December	10	Dec. 27, 2017, 5:12 a.m.	5.00
December	11	Dec. 27, 2017, 5:12 a.m.	7.50
December	12	Dec. 27, 2017, 5:12 a.m.	10.00
December	13	Dec. 27, 2017, 5:21 a.m.	12.50

Fig. 5.2a Login Page **Fig. 5.2b Main Screen** **Fig 5.2c User History Page**

6. Working of the system

We now describe the complete working of the system in the following steps:

1. The user types the URL of the web application and is directed to the login page of the server
2. After successfully logging in, he is taken to a page shown in Fig 5.2b.
3. The user can view his history as shown in Fig 5.2c by clicking the View history button
4. When the user clicks on View Machine, the voltage at the IN USE LED, is sampled by using a comparator IC (LM339) and level translated to 3.3v which is compatible with the raspberry pi input voltage.
5. The output is connected to one pair of wires in the Ethernet cable. This signal is fed via the RJ45 connector to the GPIO pins of the Pi and then it is relayed back to the central-server. The status of the machines are then displayed on the portal. The machines that are ideal are only displayed in the machine list.
6. When the user selects a specific machine on this web portal, it is communicated by the main server to the raspberry-pi over the TCP/IP network of the institute. A small python based server runs in the raspberry-pi, which listens to the commands from the main server
7. After selecting the machine, when the user clicks on Activate button, the machine number of the selected machine (in decimal) is sent to the Psi where it is converted into 4 bit binary variables. These variables are fed as input to the 4:16 decoder on the PCB via the 40 pin connector.
8. On the basis of this binary pattern, a specific output of the decoder is pulled low. This decoder output then closes the specific optocoupler connected to it. The terminals of the micro switch are not directly driven by the decoder but instead through an optocoupler as it provides electrical isolation and prevention from current overdrive. The closing of optocoupler is equivalent to the closing of the micro-switch, which activates the machine. New data entry is stored in the database.

7. Results and Conclusions

In this paper, we discussed the design of the PCB (made via KiCad Software). We also discussed how the individual components on the PCB work to determine the number of idle machines and for activation of machines. The PCB acts as a base station which connects a number of machine nodes via RJ45 port and Ethernet cable. We choose RJ45 connectors and Ethernet cable for robustness. The Raspberry pi acts like a processing unit by gathering the information about the machines through the PCB. The information is stored as high or low logic in its GPIO pins. This information is then sent back to the central server. Whenever the user activates a machine, the record is maintained by the server in his account. The administrator can log into his account and download the pdf summary of all the users. The proposed system is robust and more secure. It is convenient to the users as they do not have to visit the laundromat frequently. Book keeping issues are eliminated. This solution is scalable and can be implemented in different situations.

We now discuss the possible future modifications of this work. Firstly, heat sensors can be added to the dryers and interfaced with the Pi. They can gather the temperature data of the machine and send it to the central server. Thus, the user can come to know if the dryer is functioning properly and that his clothes are getting dried. The user and the administrator can hence be notified in case of a defective machine. Further, the functionality of booking a particular machine for a short span can be added. As a result, a queue can be created and the users can view the entire queue that is waiting for that machine before them. A time slot of 5 min can be allotted such that if a user fails to activate the machine in that slot, then his entry in the queue can be deleted and the next entry can be processed. Further, the voltages at each component and the heat generated at the board can be sent as input to a machine learning model running on the pi, trained to predict the time when the machine will go off or fail to function properly. This will allow the users and particularly, the admin to know in advance the status of each machine in future, hence preventing any failure inconvenience. Another modification will be to use smaller microprocessors and to print the PCB on a commercial level so as to squeeze the whole system to mini size level.

8. Acknowledgements

We would like to sincerely thank Dr. Santhosh Kumar and our hostel staff for their guidance and encouragement in carrying out this project work. We wish to express our gratitude towards Dr. R.N Saha, the director of Bits Pilani Dubai Campus, for providing us the resources for the completion of this project and his valuable guidance from time to time. In the end, we would like to thank Rahul Manikam and Ethan Antonio who helped us in creating the video for our project.

9. References

[1] https://internetofthingsagenda.techtarget.com/definition/Internet-of-Things-IoT
[2] https://assets.nexperia.com/documents/data-sheet/74HC_HCT154.pdf
[3] https://www.electronics-tutorials.ws/blog/optocoupler.html
[4] https://www.showmecables.com/blog/post/rj45-pinout/
[5] https://www.electronics-tutorials.ws/opamp/op-amp-comparator.html