# Design and Control of Automated Medical Bed System Using IoT

**Department of Robotics and Mechatronics Engineering**

Author:

Roll: 1216
Reg: 2016 918 724

Roll: 1202
Reg: 2015 616 919

# Introduction

The goal of the project is to design and develop an intelligent medical system with a sensor system that can be integrated into the monitoring of medical beds.

If the patient is near the emergency room, the control panel will send a signal to the bed. Automatically enter the emergency room; if the patient needs urgent care, the bed will be transferred to the emergency room; on the other hand, if the patient needs a conventional cabin, the automatic bed will be placed in the ordinary cabin.

# Proposed Medical Bed Design:

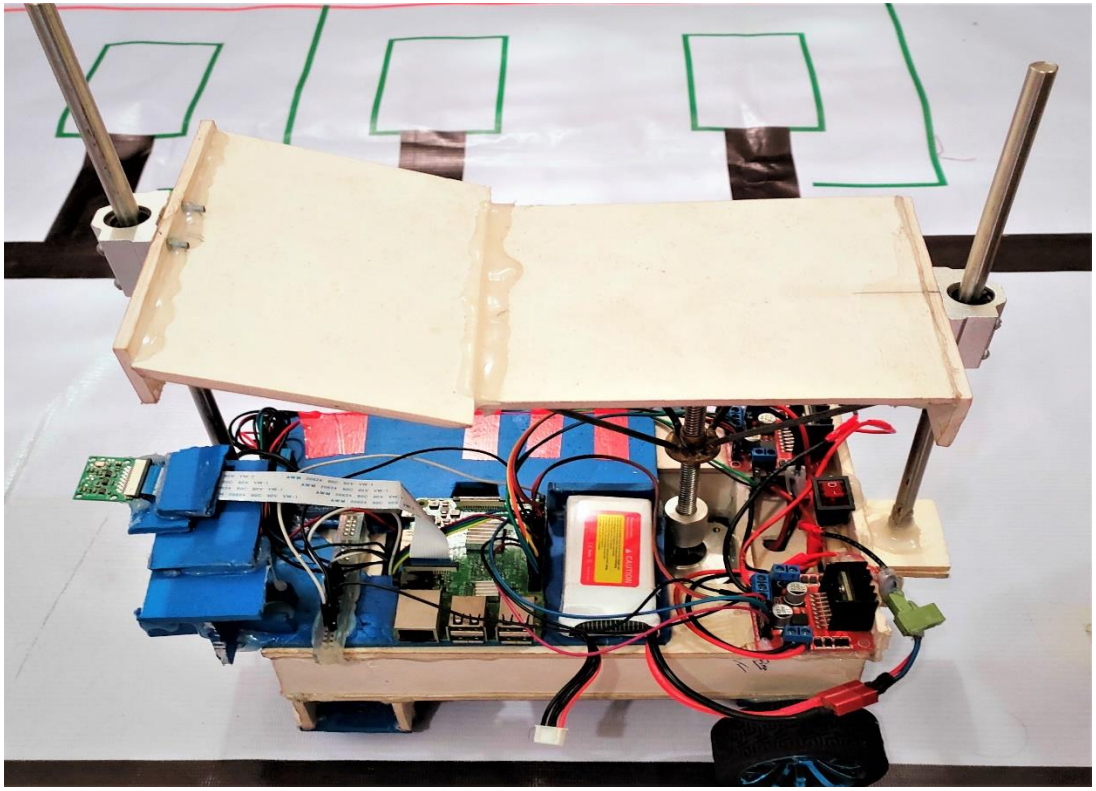


Fig:1-Medical Bed

# Mechanical Mechanism

This is a lead screw mechanism that transforms the rotational motion of the motor to the linear motion of the bed. For rotational motion, we connect it to a NEMA-17 Stepper motor.

Fig:2-Mehanical Mechanism

# Circuit Design of Medical Bed

Component List:

1. Raspberry Pi 3B.
2. Pi Camera.
3. ADC MCP-3008.
4. IR Sensor Array.
5. L298N Motor Controller.
6. Gear Motor 256 RPM.
7. Nema-17 Stepper Motor.
8. 1500mAh 12V Lipo Battery.
9. 10000 mAh 5V 2Amp Power Bank.
10. Lead Screw Mechanism of Bed.
11. Linear Bearing.
12. Stainless Steal Bar.

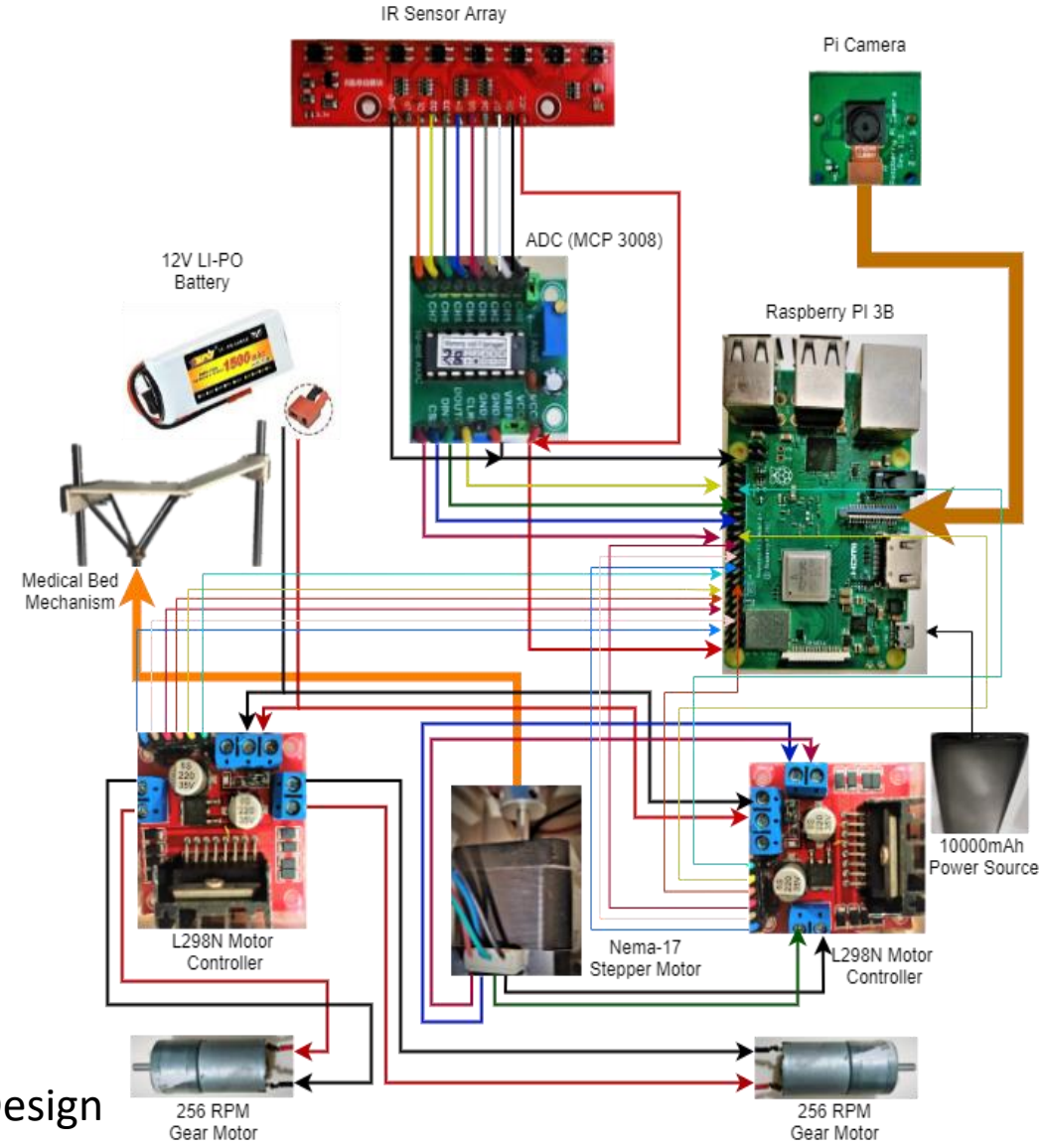



Fig:3- Circuit Design

# Proposed Map for Medical Bed

Medical map where our automated bed will move to the gate for picking patient and move back to its own cabin position. We also attach QR code to the map for bed position identification. Black line means the track where the bed moves by reading QR code.

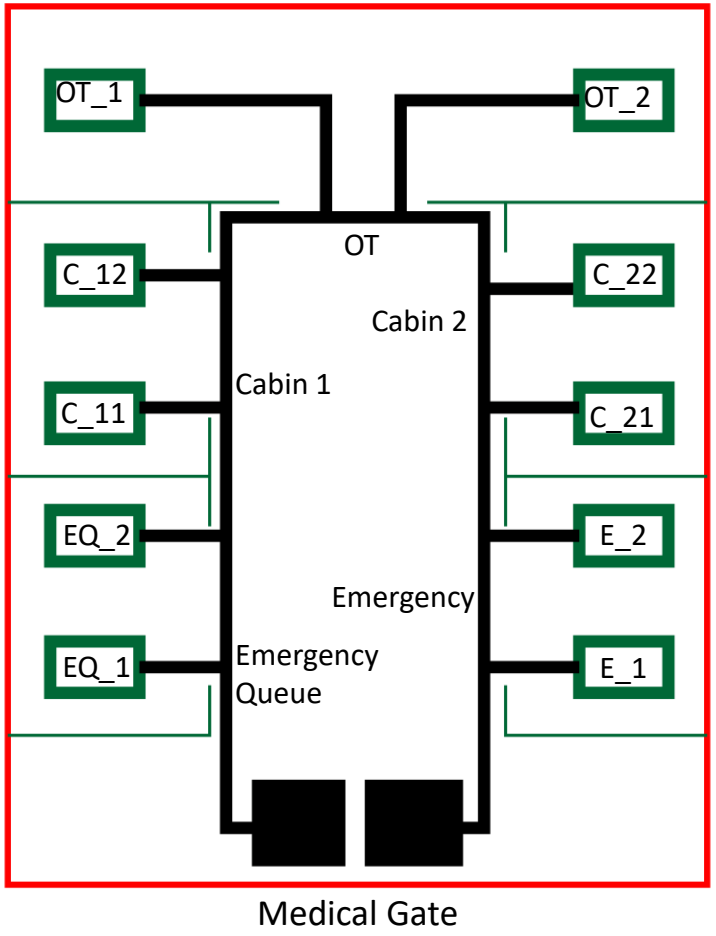



Fig:4- Proposed Map

# Methodology

- We run a webserver and publish through a router using an address.
- Any device connected with the same router can visit the webpage using the publishing address but can not manipulate the server data.
- Webserver data can be manipulated by controller.
- Each automated medical bed can read its own data that served by the router using webs crapping.
- Now according to the served data medical bed will define itself that it will move or not or what should it have to do.

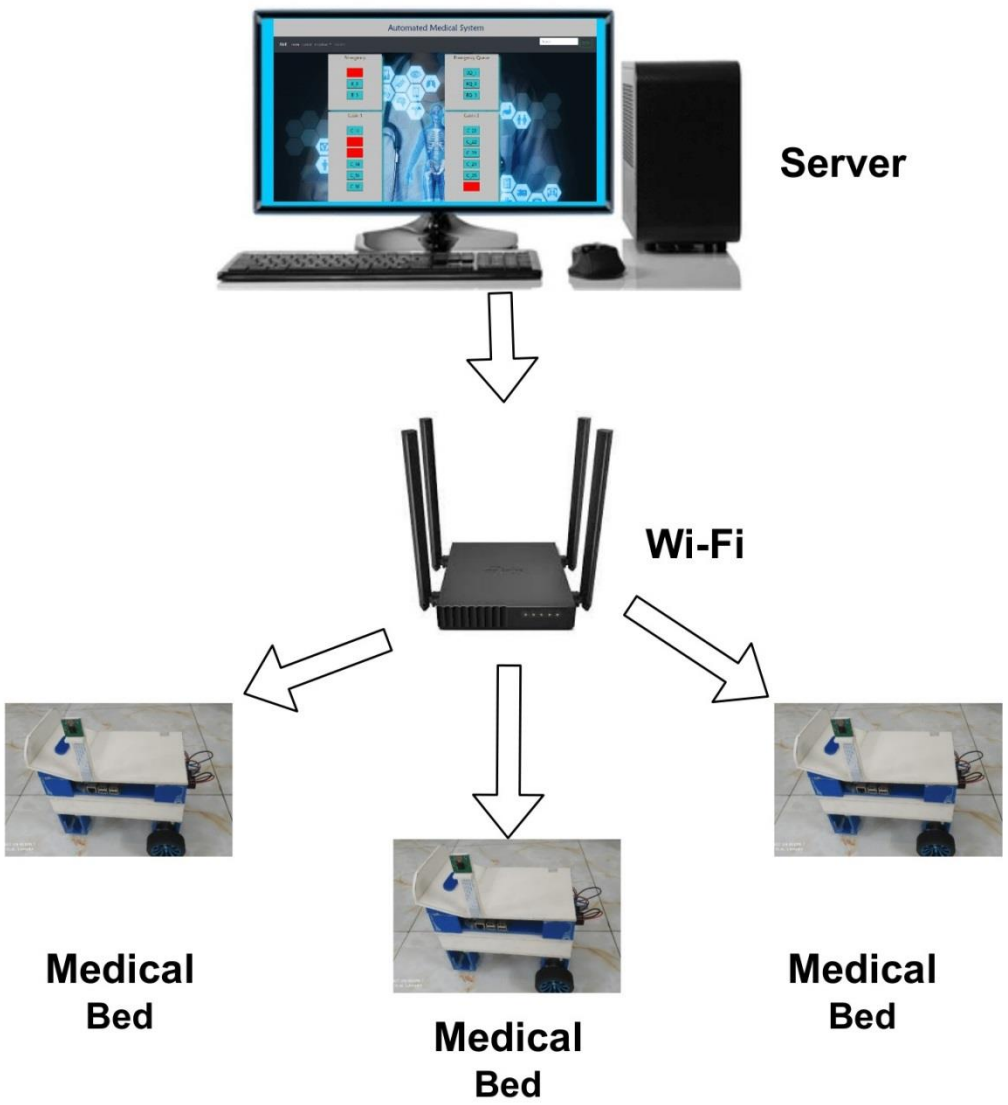
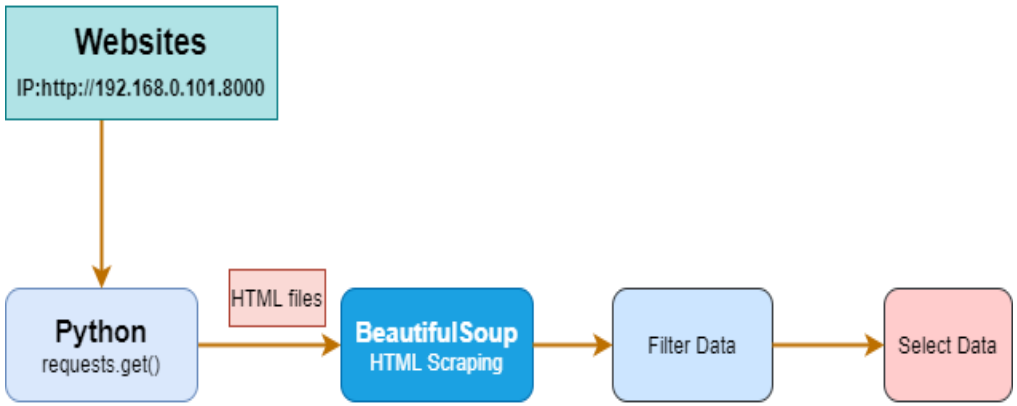



Fig:5- Block Diagram

# How Server Command Transfer to Medical Bed?



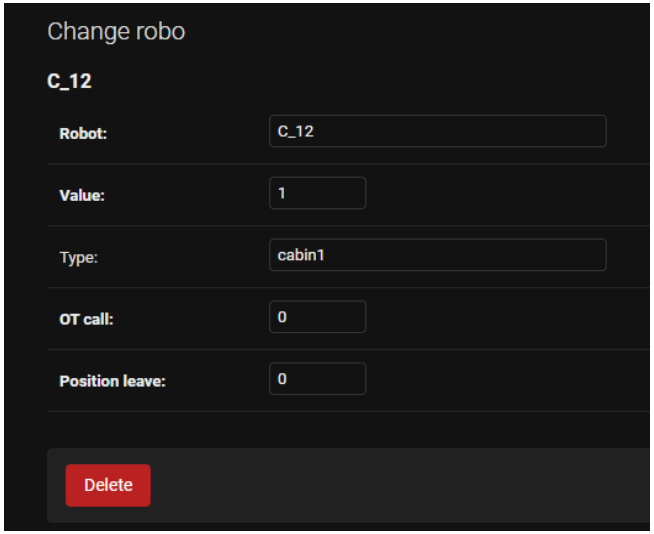




Fig:6-Beautiful Soup Library

1. Give any value to the backend webserver (http://192.168.0.101:8000/admin) like 0 or 1 to the field of Value, OT call, and Position leave.
2. Then the fields will update the value and also update the frontend of the website Contact page and publish it.
3. Then using requests.get('http://192.168.0.101:8000/contact/') function we can access the data of webpage.
4. After that using the BeautifulSoup library extract the data and filter our necessary data of the Contact webpage.
5. Finally store the data to an array and select data from this array according to our robotic medical bed.

# Bed moving Truth Table

Several conditions of the automated medical bed movement according to the server data field. This table defines when the medical bed will go to the gate for picking the patient or when it will go to the OT by carrying the patient or when it will not move from its current position.

| Index | Value | OT Call | Position leave |
|---|---|---|---|
| Stay in Cabin | 0 | 0 | 0 |
| No Move | 0 | 0 | 1 |
| No Move | 0 | 1 | 0 |
| Moving to OT | 0 | 1 | 1 |
| No Move | 1 | 0 | 0 |
| Moving to Hospital Gate | 1 | 0 | 1 |
| No Move | 1 | 1 | 0 |
| No Move | 1 | 1 | 1 |

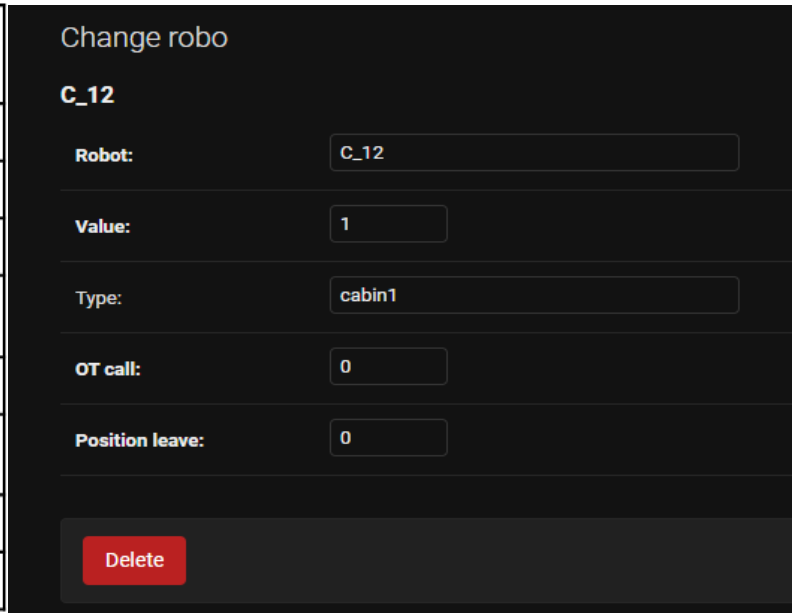



Fig:7-Truth Table

# QR Reading

**Step1:** Video stream to the Raspberry pi through Pi-camera.

**Step2:** Image extract frame by frame from stream video.

**Step3:** Image inserts into Zbar decoder.

**Step4:** Zbar detects 4 corners of the QR code.

**Step5:** According to the light intensity of the Black and White portion of the QR code it can detect the QR data.

**Step6:** According to the light intensity it gives output C_12.

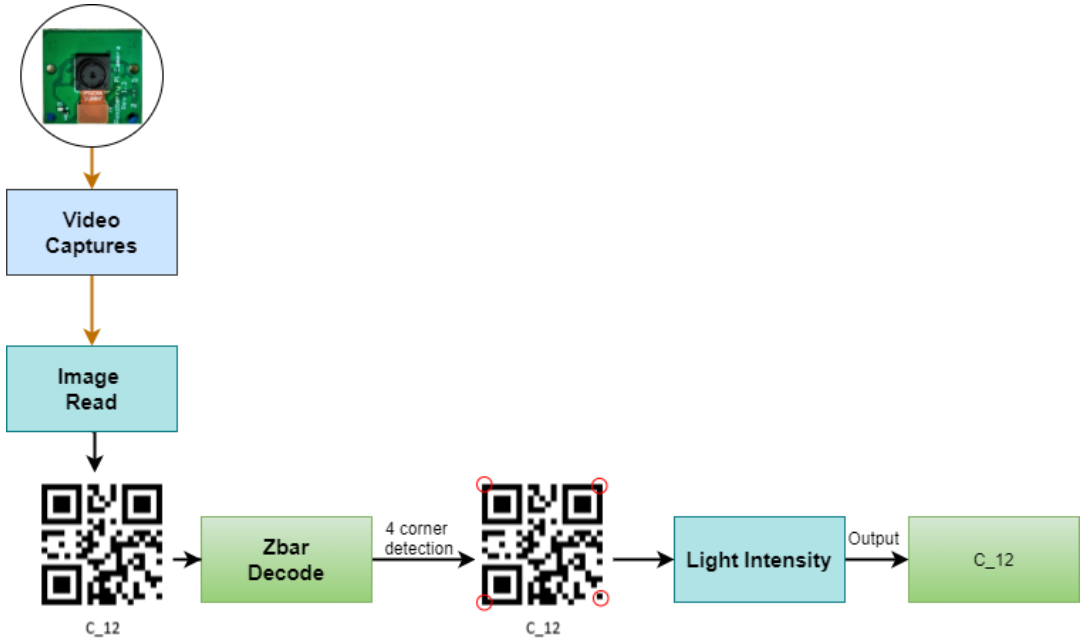
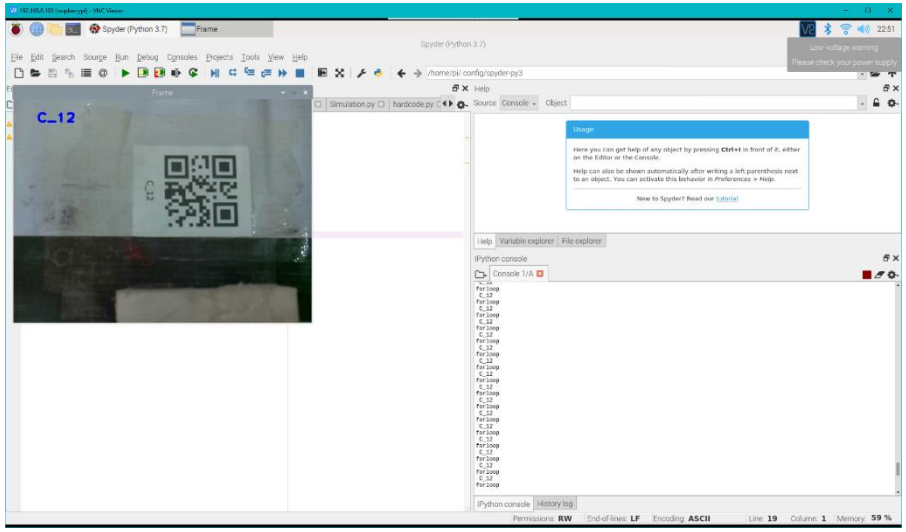



Fig:8- Zbar library

# QR Reading Simulation

When the medical bed move by following a QR code then it can recognize the QR data and execute the automated command of the microcontroller for reaching a destination. These figures are the simulated result of the spyder of Raspberry Pi 3.



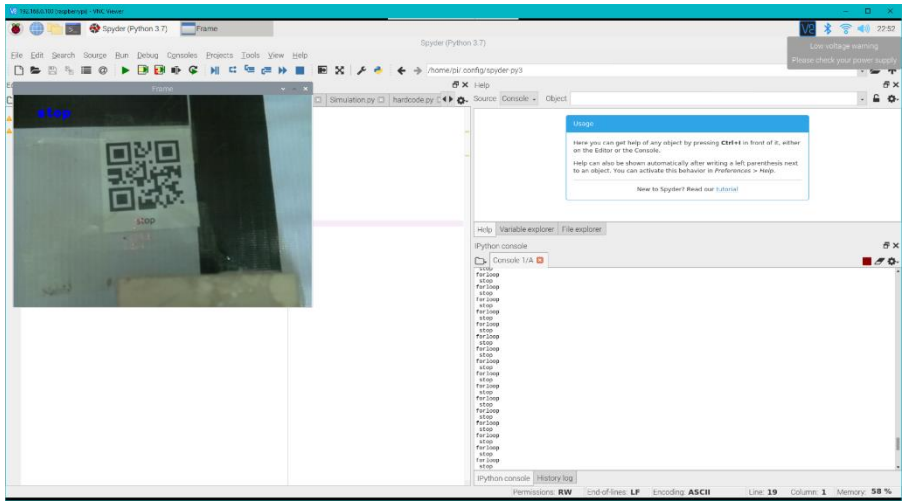




Fig:9-QR Code Reading

# Gate call of Medical Bed

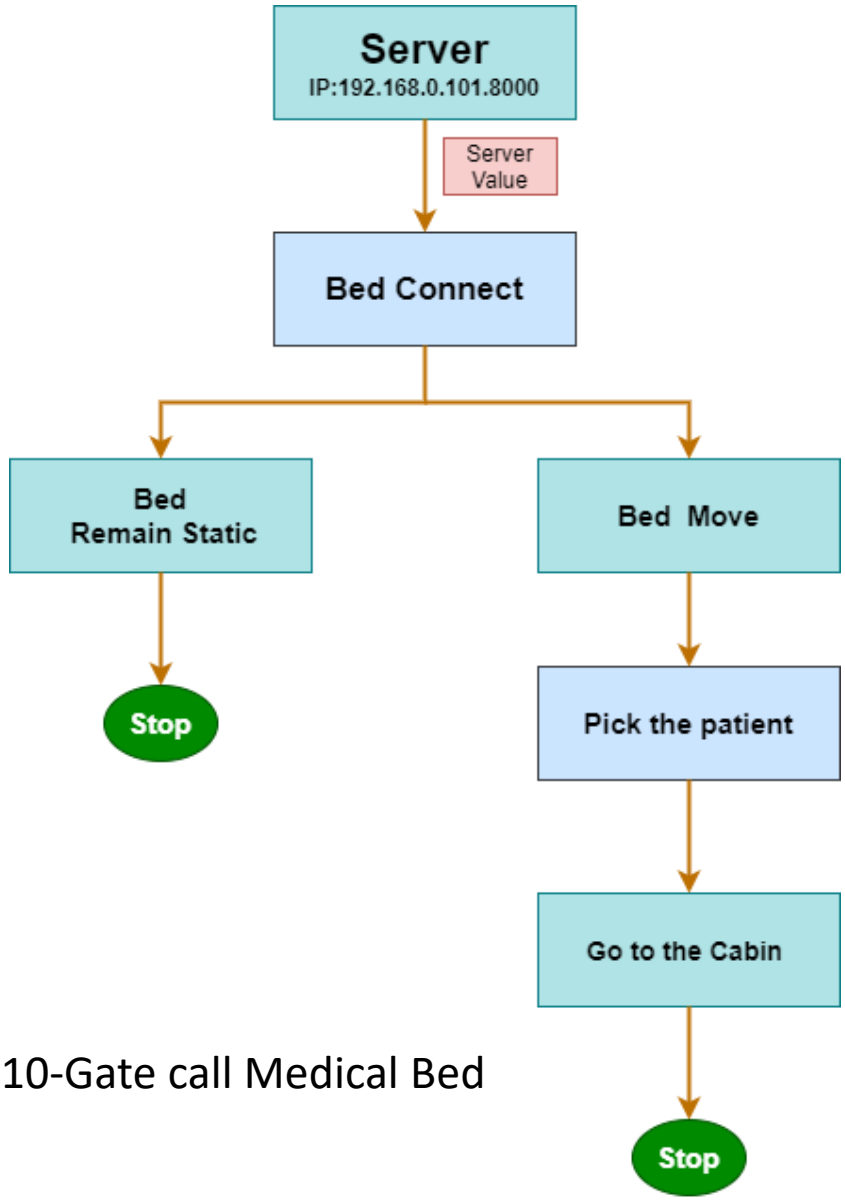
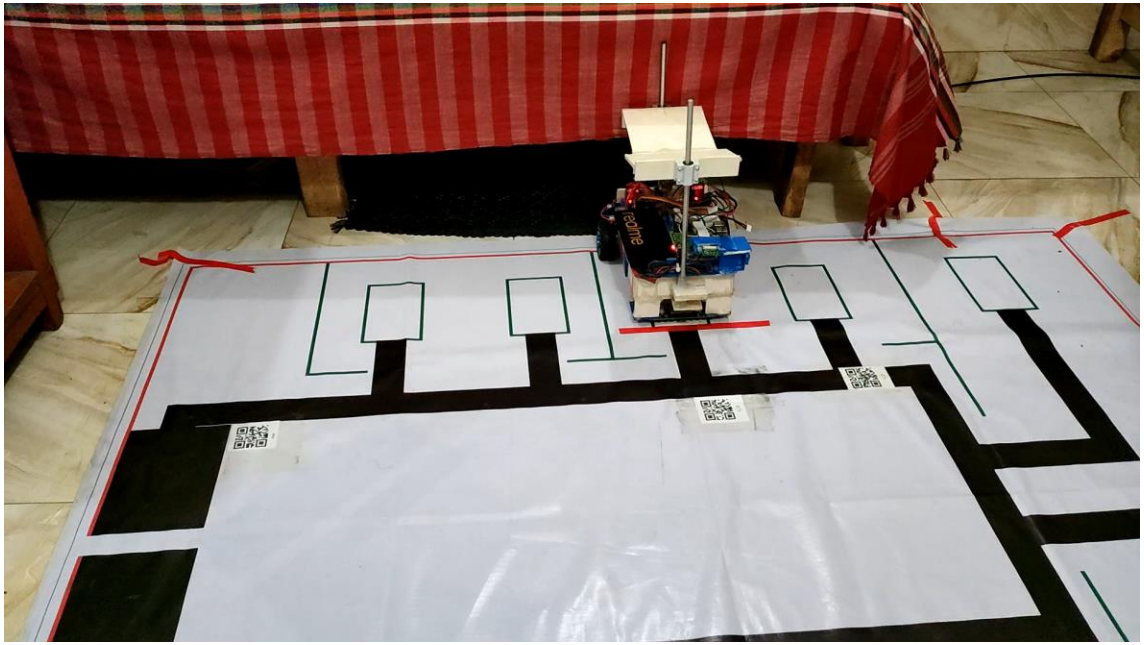



Fig:10-Gate call Medical Bed

# Gate call of Medical Bed



# Simulated Result When Medical Bed Move to The Gate

Simulation result 1, when medical bed got value=1 but position leave=0 from server and it does not move.

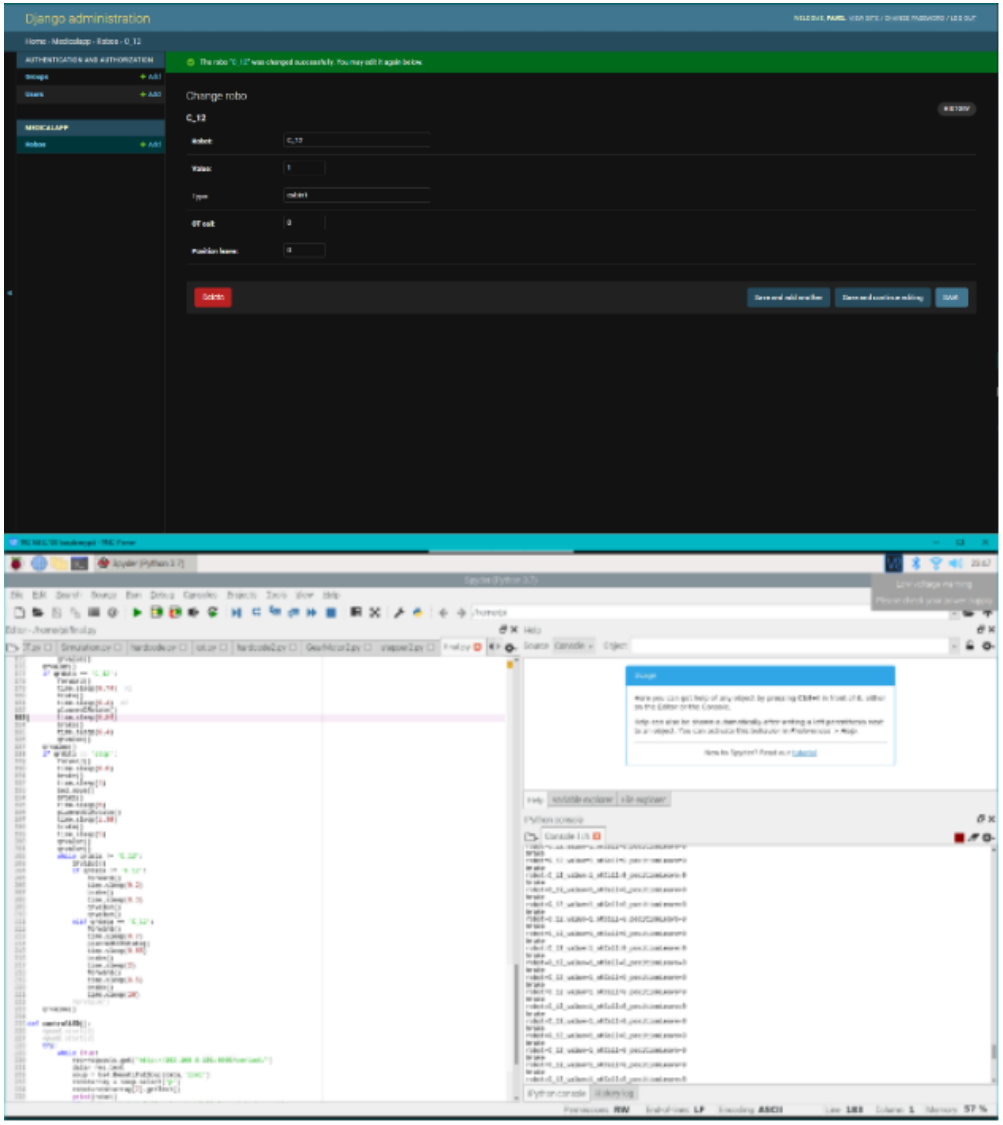



Fig:11-Simulated Result

# Simulated Result When Medical Bed Move to The Gate

Simulation result 2, when medical bed finding the path of the gate by reading QR code according to server value=1 and position leave=1.

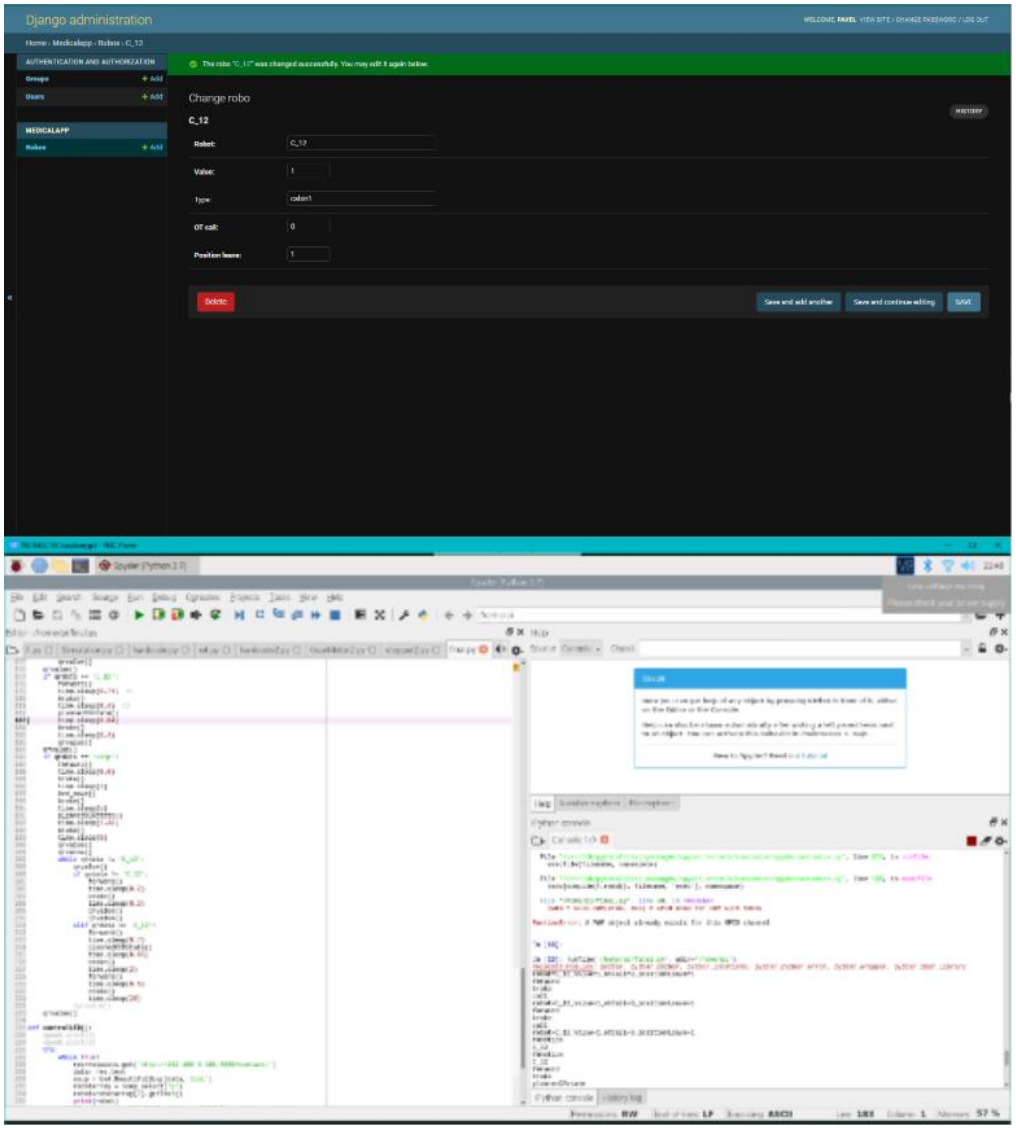



Fig:12-Simulated Result

# OT call of Medical Bed

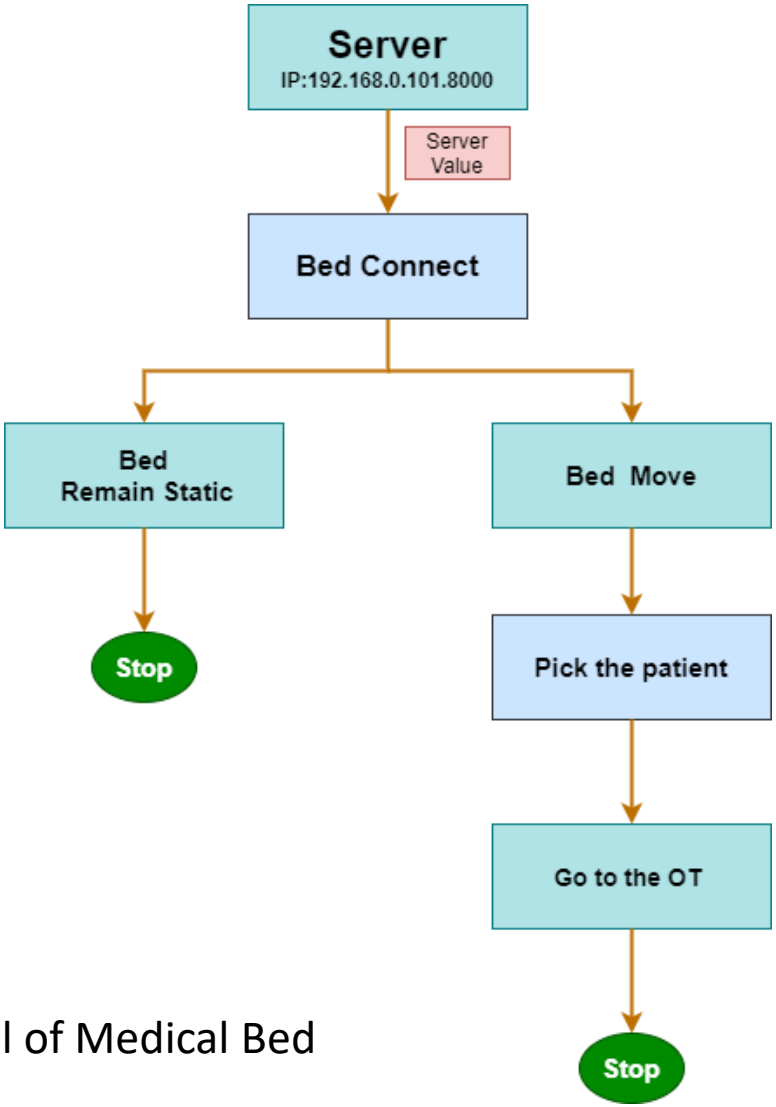



Fig:13-OT call of Medical Bed

# OT call of Medical Bed

# Simulated Result When Medical Bed Move to The OT

Simulation result 1, when medical bed got OT call=1 and position leave=1 from server and it moves to OT.

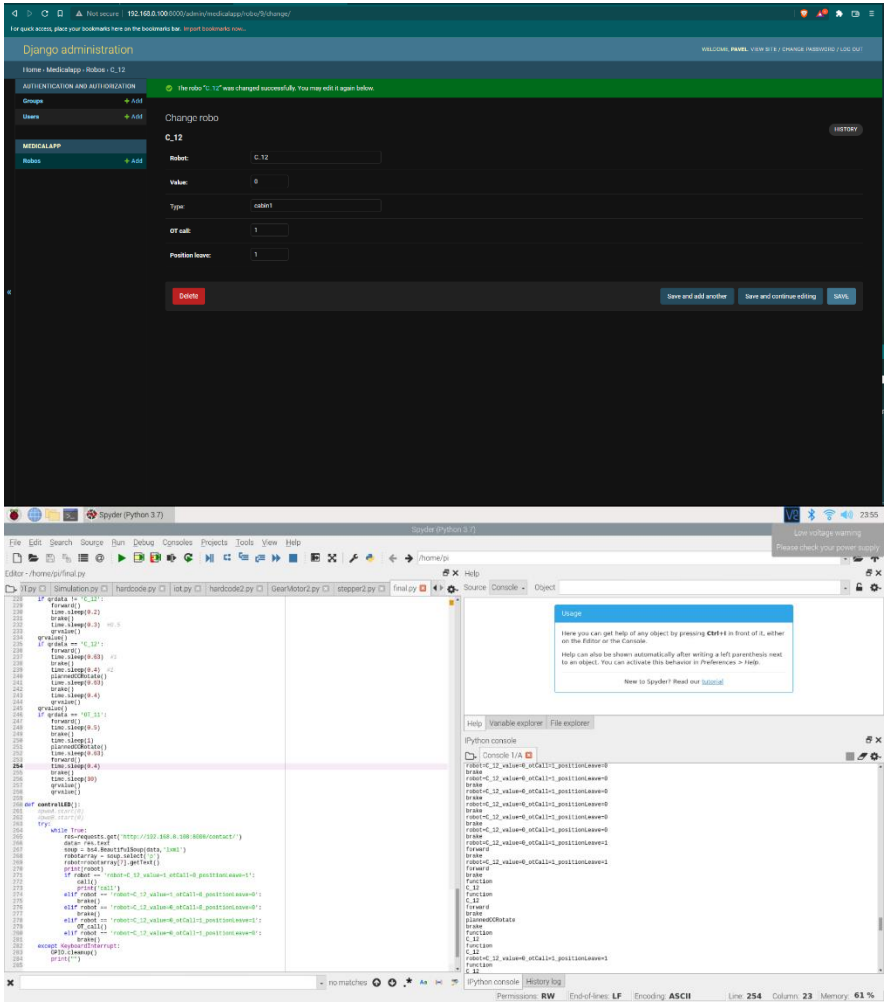



Fig:14-Simulated Result

# Simulated Result When Medical Bed Move to The OT

Simulation result 2, when medical bed got OT call=1 and position leave=1 from server and it stops to OT when it finds the stop QR code inside operation theater.

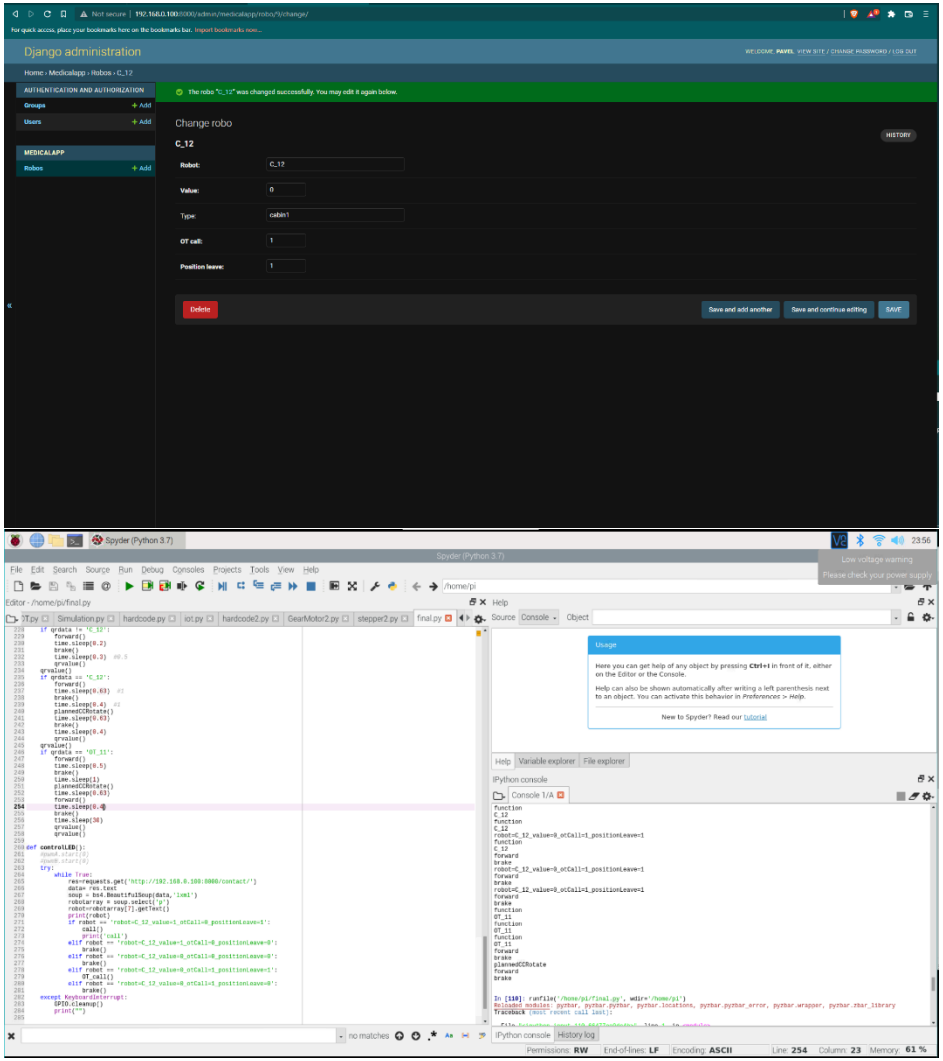



Fig:15-Simulated Result

# Complete System Algorithm

**Step1:** Start the Medical Bed
**Step2:** Read the value from the Database.
**Step3:** If Robot != C_12 && value!=1?
**Step4:** Go to step2
**Step5:**If Robot = =C_12 && value==1?
**Step6:** C_12 medical bed move forward
**Step7:** Then the bed brake
**Step8:** Find the QR Code
**Step9:** if qrvalue!=C_12?
**Step10:** Go to step6
**Step11:** if qrvalue==C_12?
**Step12:** Clockwise Rotation
**Step13:** The bed move forward
**Step14:** Then the bed brake
**Step15:** if qrvalue!=stop?
**Step16:** Go to step13
**Step17:** if qrvalue==stop?
**Step18:** Then the bed move forward
**Step19:** Then the bed brake
**Step20:** Then the bed move downward
**Step21:** Pick the Patient
**Step22:** Then the bed move upward
**Step23:** Counterclockwise rotation
**Step24:** Then the bed move forward
**Step25:** Then the bed brake
**Step26:** if qrvalue!=C_12?
**Step27:** go to step24
**Step28:** if qrvalue==C_12?
**Step29:** Counterclockwise rotation
**Step30:** Then the bed move forward
**Step31:** Then the bed brake
**Step32:** Go to setp2
**Step33:** Stop the medical bed by keyboard interruption.

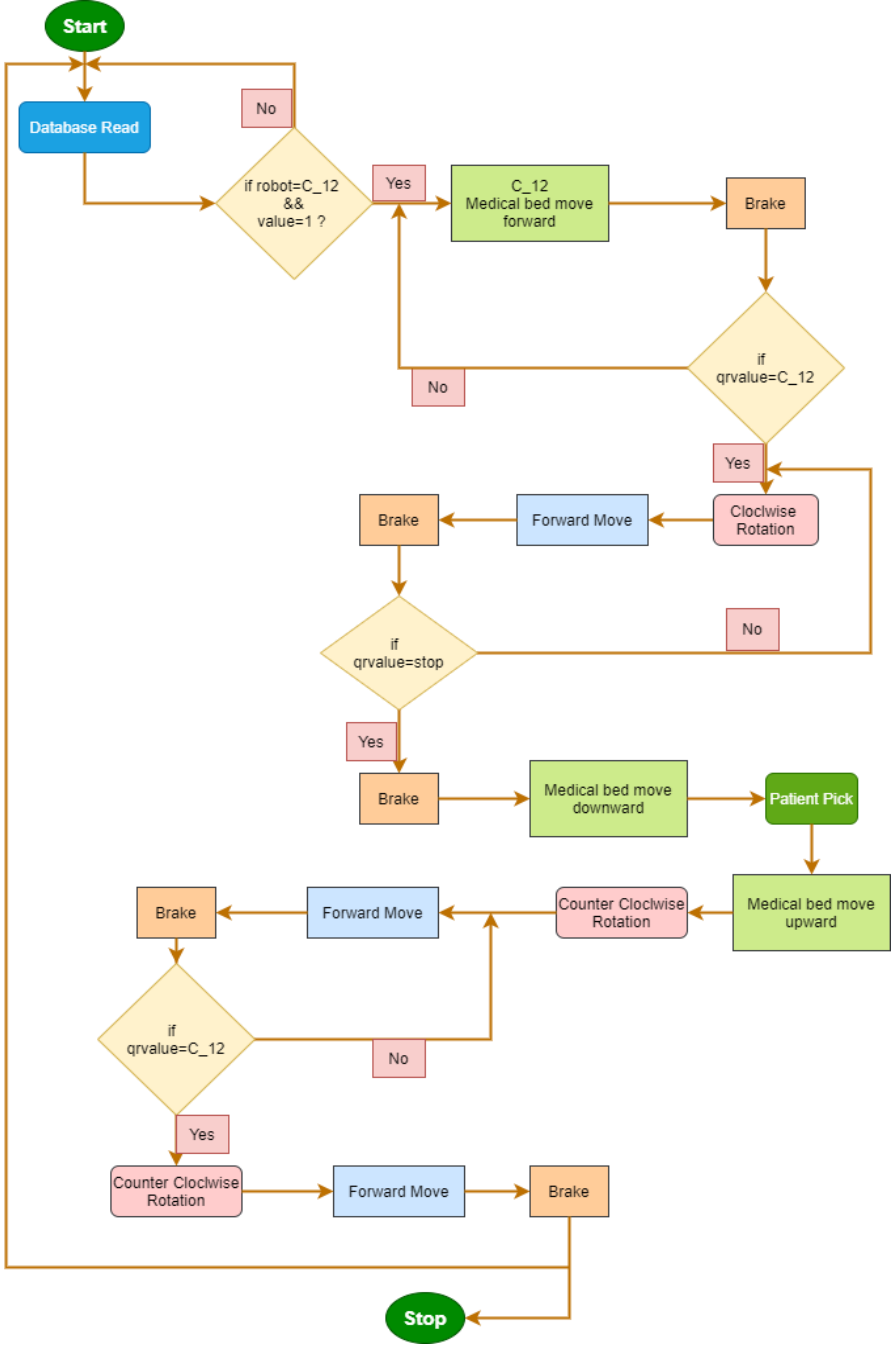
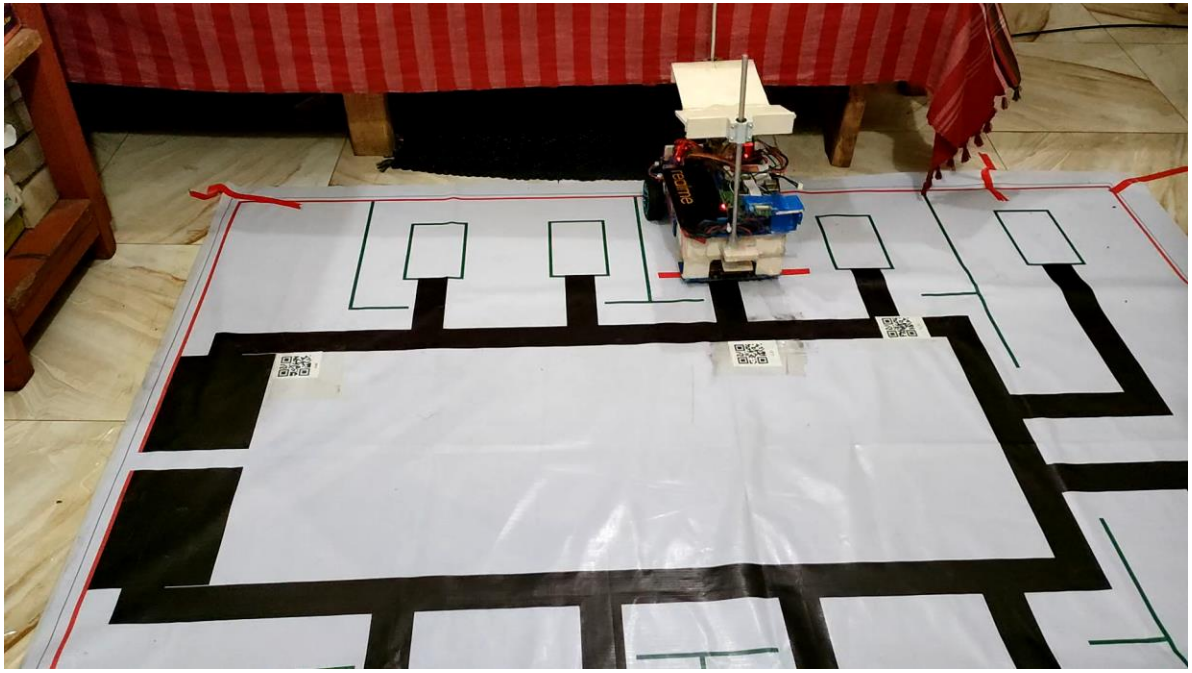



Fig:16-Complete system of Medical Bed

# Medical Bed Can Not Reach Cabin:

We gave the server command for picking patients from the gate but it could not find the cabin due to the displacement of the camera capture area from QR code area. It counts as a failure case or an error case.



# Result:

We ran the medical bed around 14 times for moving it from cabin to gate and picking the patient and then returning it to the cabin. The result is shown below:

| No of trials | Success | Failure |
|---|---|---|
| 1st | Yes | No |
| 2nd | Yes | No |
| 3rd | No | Yes |
| 4th | Yes | No |
| 5th | Yes | No |
| 6th | Yes | No |
| 7th | No | Yes |
| 8th | Yes | No |
| 9th | Yes | No |
| 10th | No | Yes |
| 11th | Yes | No |
| 12th | Yes | No |
| 13th | No | Yes |
| 14th | Yes | No |
| Total | 9 times | 5 times |

Fig:16-Result

Success Yes or Failure No means the bed moved to the Gate and pick the patient from the gate then returned itself to the cabin successfully. Success No or Failure Yes means the bed missed the track of QR code and provide unwanted movement and could not be returned to the cabin.

## Result:

$$Accuracy = \frac{Number\ of\ Success}{Total\ number\ of\ trials} \times 100\%$$

$$Accuracy = \frac{9}{14} \times 100\%$$

$$Accuracy = 64.28\%$$

$$Error\ rate = \frac{Number\ of\ Failures}{Total\ number\ of\ trials} \times 100\%$$

$$Error\ rate = \frac{5}{14} \times 100\%$$

$$Error\ rate = 35.72\%$$

## **Limitations**:

1. Due to the lack of surface friction some times wheel slips the rotation or forward movement.
2. Sometimes camera missed capturing the QR code.
3. We used L298N controller for controlling the Stepper motor.

# **Future Work**:

1. We can add some intelligent system that the robot can extract paths using image processing.
2. Also we can use a line segmentation-based system that the robot will draw some boundaries against track in-stream video and move through it.
3. We can synchronize the medical bed with lift for moving one floor to another floor.
4. We could add some smart feature in future.

# Conclusion

This paper based on IoT technology and vision system design and control of an automated medical system. The proposed system can provide information about the position of the bed and record the information. Smart Bed monitoring system with IoT is a system that collects information.

Firstly, we used the IR sensor array but it did not complete the task that moves the cabin to gate and gate to cabin. Because the medical robot out from the track and rotate itself or moves forward to finding the track but it did not find the track for a long time and our goal can not achieve.

In our Result error is around 35% which can be minimized using the YOLO algorithm by detecting the line from the stream image of the map. Also, our error maximized due to the slip of the wheel during forwarding rotation.

# Thank You