**ROBOTIC ARM REMOTE TOOLING**

**1) Core idea**

Implementation of the concept of remote tooling for robotic arm using cloud infrastructure.

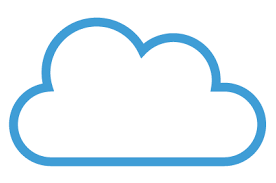
This idea enables effective control of a robotic arm remotely with very less delay in response. The remotely controllable robotic arm can then be used for a wide variety of applications involving remote tooling and for industrial robotic automation.

For this, the cloud infrastructure and its enormous features proves highly suitable and of great help. By implementing the concept of remote tooling of the robotic arm in cloud we are able to increase the remote range of access and also reduce the response time drastically and hence making the arm very reliable and quick to respond to the commands across even very large distances. By implementing the cloud infrastructure we are also providing high scalability and hence the capability for the system to adapt easily and reliably if the number of remote robotic arm is increased.

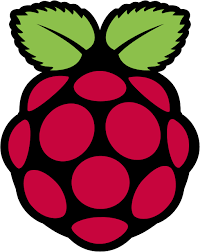
**2) Details of sub-modules/systems**

1. **Robotic Arm:** The robotic arm used here comprises of 8 servos where in each servo can take up rotation from 0 to 140 degrees. The robotic arm is controlled by an Arduino. The robotic arm is connected to the cloud using the raspberry pi module. The remote signal or commands given by the client is send to could and from the cloud its retrieved by the raspberry pi and then the pi sends the commands to the Arduino, where in the Arduino takes the required actions on the servos.
2. **Cloud:** The cloud infrastructure is the backbone of the project and is responsible for the communication of the signals or commands of the client located at the remote location. For the backend cloud infrastructure we are making use of the cloud service provided by PubNub. On the cloud, we are making use of “server less” architecture which is responsible for providing the high scalability and adaptability.
3. **Client:** The user or the client will be making use of a web portal that takes the input from the user and sends the data to the cloud service for further wireless data transmission**.**
4. **Communication protocol:** The communication protocol implemented is called Publication/Subscription (PUB/SUB) protocol. This protocol is used by both the client as well as the robotic arm for any sort of communication with the cloud.

**3) Implementation and procedure**



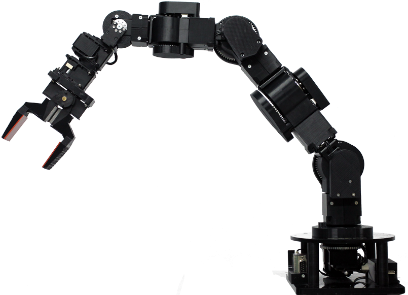
****

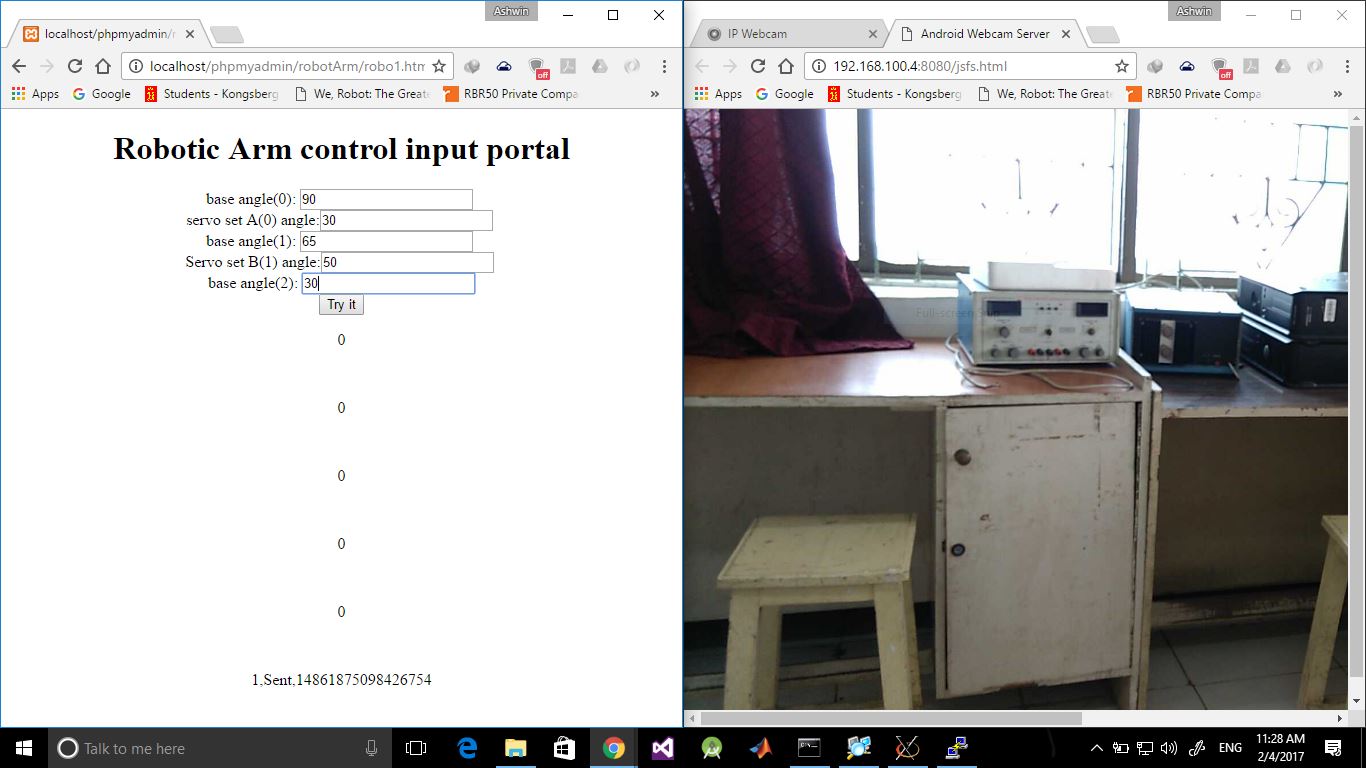
****



****

****

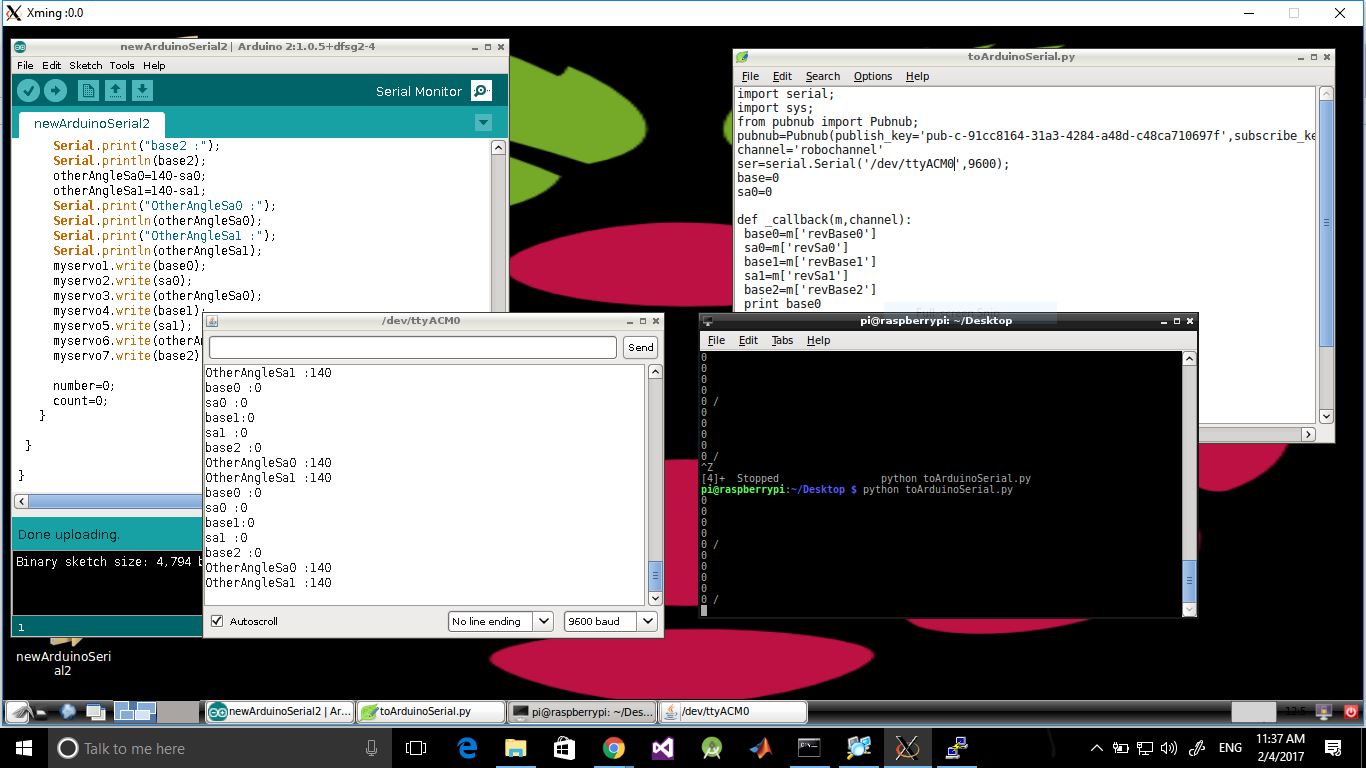
****

1. The robotic arm is fabricated using a compound that is light weight but has high tensile strength. The robotic arm has 8 servos used to provide the proper degree freedom.
2. The client side application which is used to take input from the user is a web portal. The web portal is made using the basic HTML,CSS,and JavaScript. The HTML and CSS, decide the basic structure and layout of the web page and JavaScript is used to execute the backend API to do communication with cloud. The information or commands in the form of degrees, which are to be send to the robotic arm are obtained from the user in the web portal using basic web forms. JavaScript is then used to retrieve the data from the forms and is then used with the cloud API to communicate and send the data.

Online client portal Real time video stream

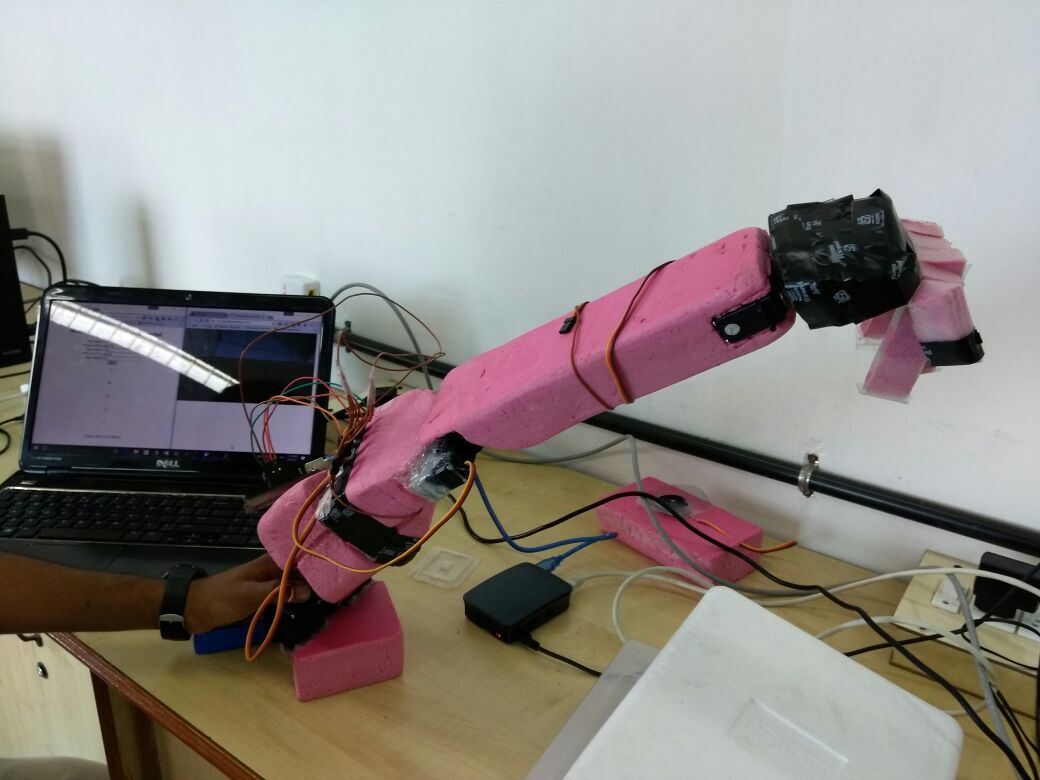
From robotic arm

1. The PubNub cloud service is used here for cloud services and the JavaScript API of the cloud service is used to do communication between the webportal.
2. The raspberry pi is used to connect to connect to the cloud to retrieve data or command from the cloud on the other side to be send to the robotic arm. In the raspberry pi , a python script consisting of the python API of the cloud service is used to retrieve data from the cloud on to the raspberry pi.
3. The raspberry pi is further connected to an Arduino. The raspberry pi does serial communication using the USB ports available on the raspberry pi with the Arduino to give it the data or commands that it retrieved from the cloud. The python code for serial communication is add to the same script which is used to retrieve the data from the cloud.
4. The Arduino is connected to the raspberry and it is uploaded with the necessary program to receive the serial data from the raspberry pi. The Arduino upon receiving the commands or degrees acts on the necessary servos to make them rotate up to the required angle and hence makes the arm take the required posture.



The image shows a session of the raspberry pi, where in the window on right is the python script retrieving the data from the cloud and the window on left is the program of the Arduino receiving and executing the commands received.

1. This project uses only one robotic arm, but due to the high scalability available to us because of the cloud service the same backend can be used to connect multiple robotic systems and still the system would be stable and provide smooth operation by automatically scaling with required backend resources. The project can be further used to do robotic automation by implementing a common instruction script on the client side , this common instruction script will contain the predefined set of repetitive operations a robotic system has to execute and if multiple robotic systems connect to the cloud service then, all the robotic systems would perform the same tasks simultaneously.

****

****

**3) Equipment to be used**

1. Arduino
2. Raspberry pi
3. Servos
4. Cloud infrastructure

**4) Application of the project**

1. The remote tooling can be used to for robotic automation.
2. The robotic remote tooling can prove to be highly helpful for executing task in environments that could be harmful for humans like radiation exposed area, or chemically contaminated area etc.
3. Robotic arm are can used for quick and effective large scale welding
4. The usage of robotic tools and remote tooling is increasing extensively in the field of medical science.
5. The robotic arm remote tooling is used in space to carry out many procedures.