Controlling Remote Robot with IOT & Motion Sensor

By – Sultan Morbiwala

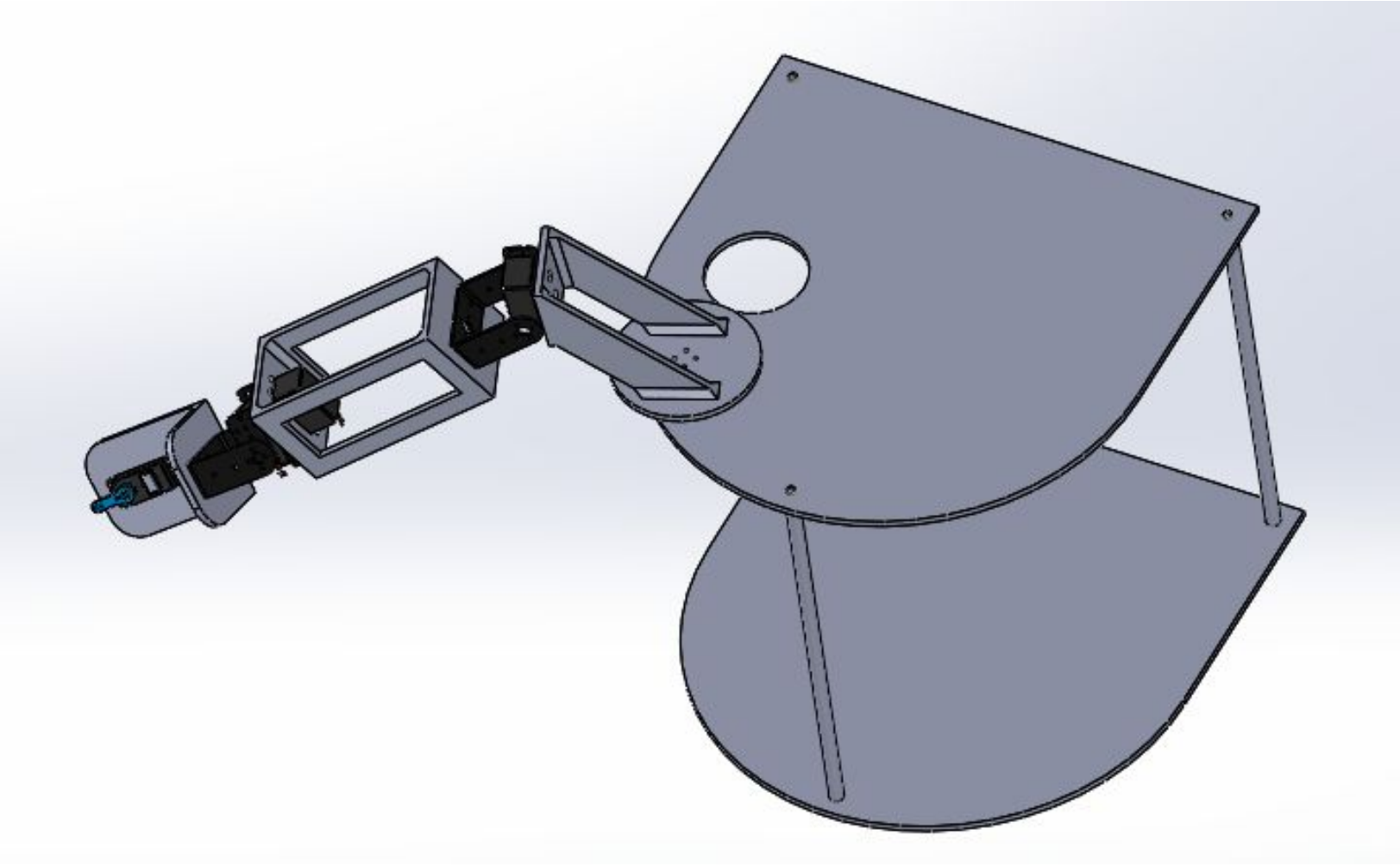
ID – 2015A4PS0367U

Objective & Purpose

- Design a mechanism to interact with physical environment without human presence
- Application:-

1) Working in hazardous environment
2) Virtual Locomotion For disabled
3) No need to travel long distance

Design



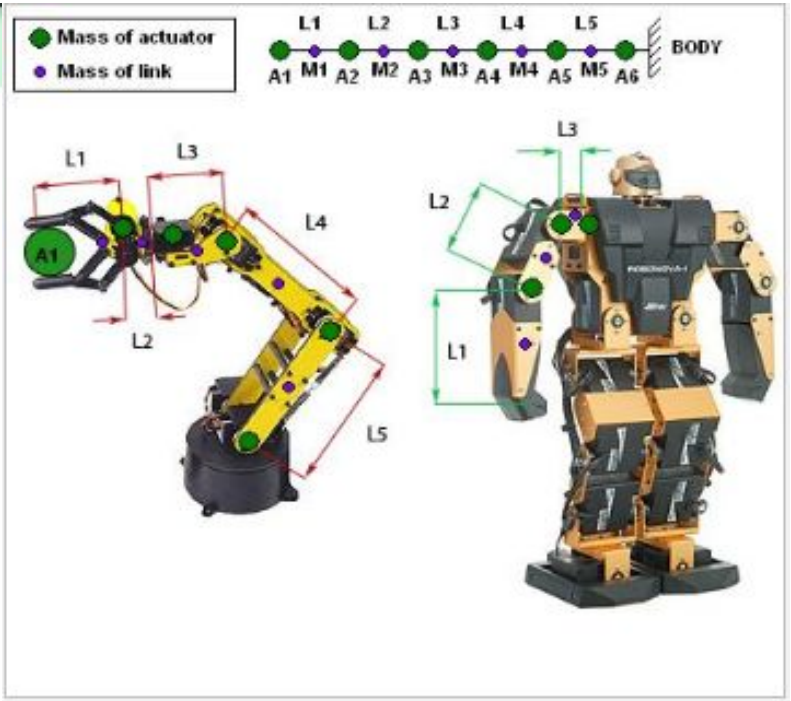
Hand Suction

Components

Force Calculation

Robot Arm Torque Calculator

L:	[cm]	M:	[kg]	A:	[kg]	T:	[kg cm]
L1:	0	M1:	0	A1:		T1:	0
L2:	3	M2:	0.02	A2:	0.2	T2:	0.6300000
L3:	15	M3:	0.08	A3:	0.06	T3:	5.43
L4:	20	M4:	0.12	A4:	0.06	T4:	15.029999
L5:	9	M5:	0.1	A5:	0.06	T5:	20.88
L6:	0	M6:	0	A6:	0	T6:	0



Robot Arm Torque Calculator

Actual Calculation

L2 = 3cm , F2 = 0.2kg

T2 = FL = 0.6 = **0.6kg.cm**

L3 = 15cm, F3 = 0.14kg

T3 = FL = 15x(0.14 + 0.2 + 0.05) = **5.85kg.cm**

L4 = 20cm, F4 = 0.18kg

T = 20x(0.18 + 0.14 + 0.2 + 0.15) = **13.4kg.cm**

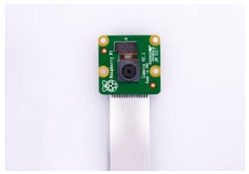
L5 = 9cm, F5 = 0.16kg

T = 9x(0.16 + 0.18 + 0.14 + 0.2 + 0.2) = 9x0.88 = **7.92kg.cm**

Motor Specs

Remote robot with IOT									
Name of motor	Quantity	Nominal Voltage	Operating Voltage Range	Nominal Current	Max Current	Nominal Power	Max Power	Stall Torque	Degree of rotation
MG995 Servo	2	6V	4.8 - 7.2V	170mA	1200mA			10kg.cm	180
Kuman Servo	2		6-7.4V	100mA				17kg.cm	270
jx Servo	1		6-8.4V					32kg.cm	180
Vacuum pump	1	12V		1A			12W	_	_

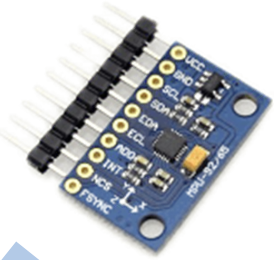
System



Video Feed System

Firebase

MIT APP INVENTOR



Robot Arm System

PubNub

Base Motion

Arduino Code

1) Initialization

```
#include <ESP8266WiFi.h>
#define PubNub_BASE_CLIENT WiFiClient
#define PUBNUB_DEBUG
#include <PubNub.h>
#define PUBNUB_DEFINE_STRSPN_AND_STRNCASECMP
#include "MPU9250.h"
#include <Servo.h>


Servo myservo;

const char *ssid =  "mym404";     // replace with your wifi ssid and wpa2 key
const char *pass =  "786110786";

const char * pubkey = "pub-c-4a9cdbca-5688-4939-a852-f63cf6743980";
const char * subkey = "sub-c-9b4067ae-1293-11e9-b4a6-026d6924b094";

const char * pubchannel = "demo_keyset";
const char * subchannel = "demo_keyset";

MPU9250 IMU(Wire,0x68);
int status;

int pos = 0;
```

2) Setup

```
void setup() {
    /* For debugging, set to speed of your choice */
    Serial.begin(115200);

       Serial.println("Connecting to ");
       Serial.println(ssid);

       WiFi.begin(ssid, pass);
       while (WiFi.status() != WL_CONNECTED)
          {
            delay(500);
            Serial.print(".");
          }
      Serial.println("");
      Serial.println("WiFi connected");

    /* Start the Pubnub library by giving it a publish and subscribe
       keys */
    PubNub.begin(pubkey, subkey);

    myservo.attach(12);
    myservo.write(0);
     // start communication with IMU
     status = IMU.begin();
     if (status < 0) {
       Serial.println("IMU initialization unsuccessful");
       Serial.println("Check IMU wiring or try cycling power");
       Serial.print("Status: ");
       Serial.println(status);
           while(1) {}

     }
  }
```

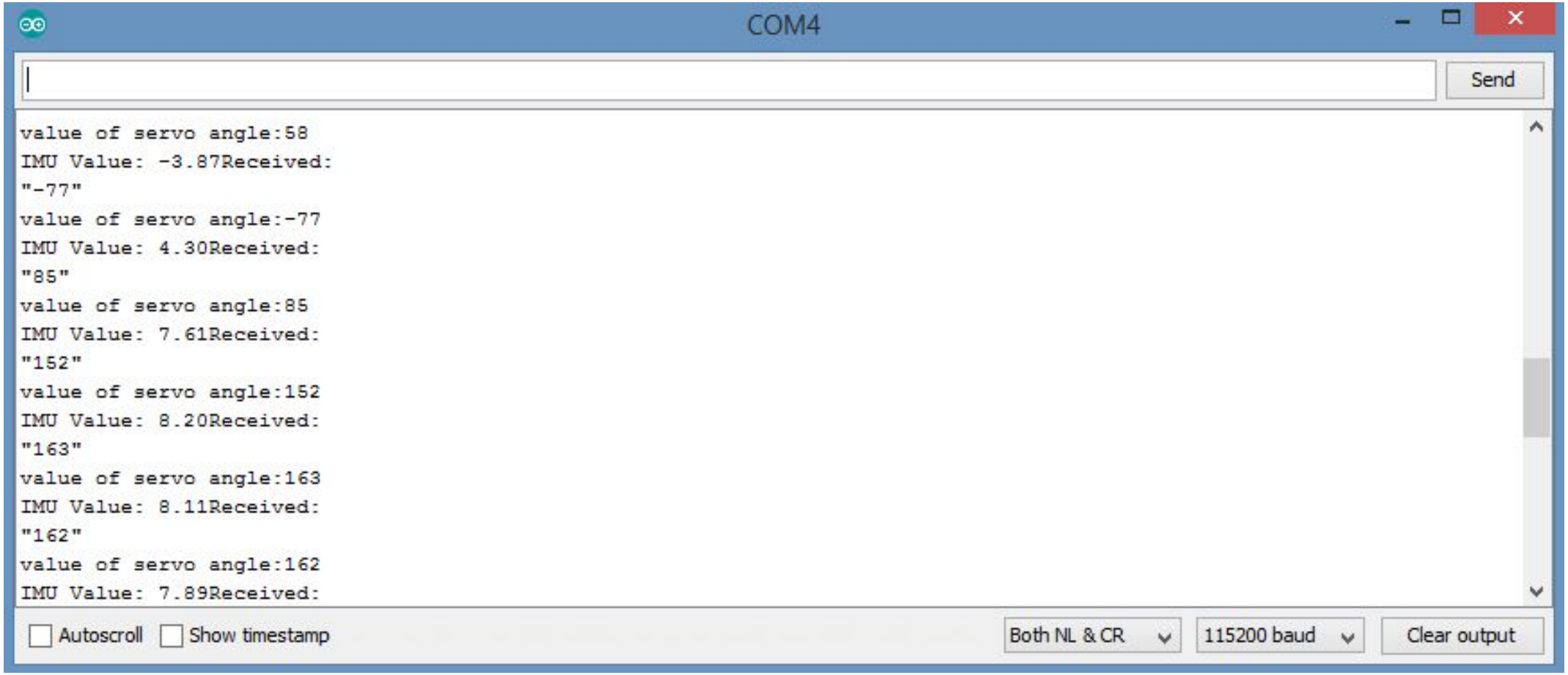
3) Main Loop – Publishing Data

```
void loop() {


//char buf[40] = { 0, };
  //sprintf(buf, "{\"columns\":[[\"Coffee\",\"%d\"]]}", TWI_fullMsg);

//IMU.getAccelX_mss()

    IMU.readSensor();

    char buf[500] = {};

    int imu = IMU.getAccelX_mss() * 20;

    sprintf(buf, "\"%d\"", imu);

    //Serial.print(buf);



    PubNonSubClient *pclient = PubNub.publish(pubchannel, buf);
    if (!pclient) return;
    PublishCracker cheez;
    cheez.read_and_parse(pclient);

    pclient->stop();
```

4) Subscribing

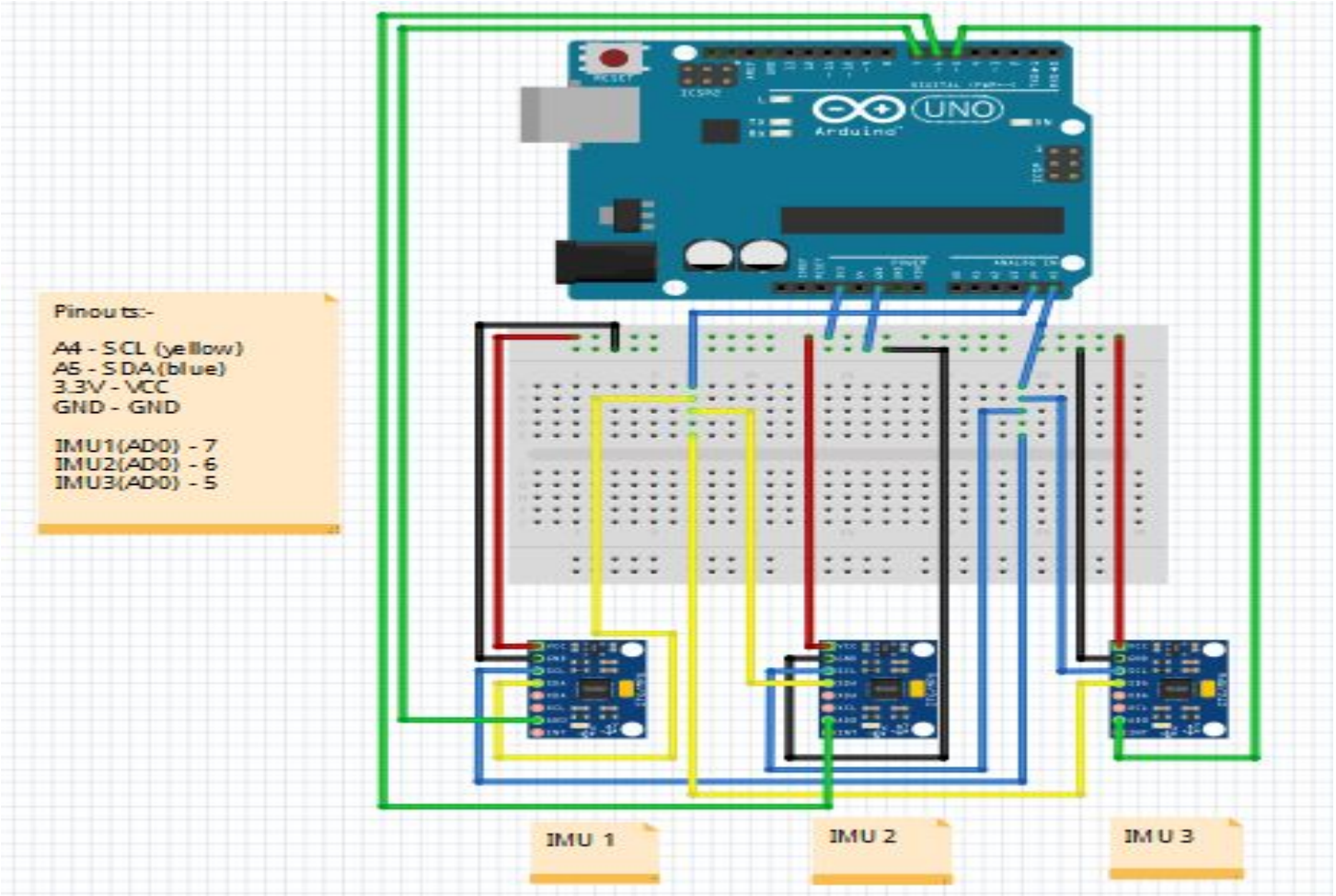
```
/* Wait for news. */
PubSubClient *sclient = PubNub.subscribe(subchannel);
if (!sclient) return; // error
String msg;
SubscribeCracker ritz(sclient);
while (!ritz.finished()) {
    ritz.get(msg);
    if (msg.length() > 0) {
        //Serial.print("Received: "); Serial.print(msg);
        msg.remove(0,1); msg.remove(msg.length());
        pos = msg.toInt();
        Serial.print("value of servo angle: "); Serial.println(pos);
        myservo.write(pos);
    }

}
sclient->stop();
```

Port Reading



I2C Bit Bashing

- Technique to connect multiple I2C devices on same bus



Code

```
void setup() {
  // serial to display data
  Serial.begin(115200);
  pinMode(imu1, OUTPUT);
  pinMode(imu2, OUTPUT);

  IMU1.begin();
}

void loop() {

digitalWrite(imu1, HIGH);

IMU1.readSensor();

Serial.println("IMU 1 is active: "); Serial.print(IMU1.getAccelX_mss(),6);

digitalWrite(imu1, LOW);
digitalWrite(imu2, HIGH);

IMU1.readSensor();


Serial.println("IMU 2 is active: "); Serial.print(IMU1.getAccelX_mss(),6);

digitalWrite(imu2, LOW);

delay(200);
}
```

Port Reading