

LAMPIRAN 1

1. *Source Code Gesture Mode*

```
1.  #include<Wire.h>
2.  #include<Servo.h>
3.  const int MPU_ADDR = 0x68;
4.  int16_t accelerometer_x, accelerometer_y, accelerometer_z;
5.  int16_t gyro_x, gyro_y, gyro_z;
6.  int16_t temperature;
7.  char tmp_str[7];
8.  Servo servo_1;
9.  Servo servo_2;
10. Servo servo_3;
11. Servo servo_4;
12. int minVal=265;
13. int maxVal=402;
14. const int flexPin = A0;
15. double x;
16. double y;
17. double z;
18. char* convert_int16_to_str(int16_t i) {
19.   sprintf(tmp_str, "%6d", i);
20.   return tmp_str;
21. }
22. void setup() {
23.   Serial.begin(9600);
24.   Wire.begin();
```

```

25. Wire.beginTransmission(MPU_ADDR);
26. Wire.write(0x6B);
27. Wire.write(0);
28. Wire.endTransmission(true);
29. servo_1.attach(2);
30. servo_2.attach(3);
31. servo_3.attach(4);
32. servo_4.attach(5);
33. }
34. void loop() {
35. Wire.beginTransmission(MPU_ADDR);
36. Wire.write(0x3B);
37. Wire.endTransmission(false);
38. Wire.requestFrom(MPU_ADDR,14, true);
39. accelerometer_x = Wire.read()<<8 | Wire.read();
40. accelerometer_y = Wire.read()<<8 | Wire.read();
41. accelerometer_z = Wire.read()<<8 | Wire.read();
42. temperature = Wire.read()<<8 | Wire.read();
43. gyro_x = Wire.read()<<8 | Wire.read();
44. gyro_y = Wire.read()<<8 | Wire.read();
45. gyro_z = Wire.read()<<8 | Wire.read();
46. int xAng = map(accelerometer_x,minVal,maxVal,-90,90);
47. int yAng = map(accelerometer_y,minVal,maxVal,-90,90);
48. int zAng = map(accelerometer_z,minVal,maxVal,-90,90);
49. x= RAD_TO_DEG * (atan2(-yAng, -zAng)+PI);
50. y= RAD_TO_DEG * (atan2(-xAng, -zAng)+PI);
51. z= RAD_TO_DEG * (atan2(-yAng, -xAng)+PI);
52. int flexValue;
53. int servoPosition;
54. flexValue = analogRead(flexPin);

```

```

55. servoPosition = map(flexValue, 870, 890, 0, 110);
56. servoPosition = constrain(servoPosition, 0, 110);
57. servo_3.write(servoPosition);
58. if(x >=0 && x <= 40)
59. {
60.   int mov1 = map(x,0,40,0,75);
61.   servo_1.write(mov1);
62. }
63. else if(x >=300 && x <= 360)
64. {
65.   int mov2 = map(x,360,300,0,90);
66.   servo_2.write(mov2);
67. }
68. if(y >=0 && y <= 55)
69. {
70.   int mov3 = map(y,0,55,90,180);
71.   servo_4.write(mov3);
72. }
73. else if(y >=270 && y <= 360)
74. {
75.   int mov3 = map(y,360,270,90,0);
76.   servo_4.write(mov3);
77. }
78. Serial.print("aX = ");
    Serial.print(convert_int16_to_str(accelerometer_x));
79. Serial.print(" | aY = ");
    Serial.print(convert_int16_to_str(accelerometer_y));
80. Serial.print(" | aZ = ");
    Serial.print(convert_int16_to_str(accelerometer_z));
81. Serial.print(" | tmp = "); Serial.print(temperature/340.00+36.53);

```

```

82. Serial.print(" | gX = "); Serial.print(convert_int16_to_str(gyro_x));
83. Serial.print(" | gY = "); Serial.print(convert_int16_to_str(gyro_y));
84. Serial.print(" | gZ = "); Serial.print(convert_int16_to_str(gyro_z));
85. Serial.print(" | Flex = ");
86. Serial.print(flexValue);
87. Serial.print(" | Cal. Sudut elbow & shoulder = ");
88. Serial.print(x);
89. Serial.print(" | Cal. Sudut base = ");
90. Serial.print(y);
91. Serial.println();
92. delay(10);
93. }

```

2. Source Code IoT Mode

```

1.  #define BLYNK_PRINT Serial
2.  #include <ESP8266WiFi.h>
3.  #include <BlynkSimpleEsp8266.h>
4.  #include <Servo.h>
5.  char auth[] = "5IIeRXpt3XmDdtyWT40xxxxxxxxxxxxxx";
6.  char ssid[] = "Kreatif xxx xxxxx";
7.  char pass[] = "kreatifxxxxxx";
8.  Servo servo1;
9.  Servo servo2;
10. Servo servo3;
11. Servo servo4;
12. const int LED1 = 16;
13. const int LED2 = 5;
14. const int LED3 = 12;
15. const int LED4 = 13;

```

```
16.  const int LED5 = 15;
17.  int button1Presses = 0;
18.  boolean button2Pressed = false;
19.  boolean button1Pressed = false;
20.  int servo1PosSaves[] = {1,1,1,1,1};
21.  int servo2PosSaves[] = {1,1,1,1,1};
22.  int servo3PosSaves[] = {1,1,1,1,1};
23.  int servo4PosSaves[] = {1,1,1,1,1};
24.  int servo1Pos;
25.  int servo2Pos;
26.  int servo3Pos;
27.  int servo4Pos;
28.  WidgetTerminal terminal(V6);
29.  void setup()
30.  {
31.    Serial.begin(9600);
32.    Blynk.begin(auth, ssid, pass);
33.    servo1.attach(4); //base
34.    servo2.attach(0); //elbow
35.    servo3.attach(2); //shoulder
36.    servo4.attach(14); //gripper
37.    pinMode(LED1, OUTPUT);
38.    pinMode(LED2, OUTPUT);
39.    pinMode(LED3, OUTPUT);
40.    pinMode(LED4, OUTPUT);
41.    digitalWrite(LED1, LOW);
42.    digitalWrite(LED2, LOW);
43.    pinMode(LED5, OUTPUT);
44.  }
45.  void loop()
```

```

46.  {
47.  Blynk.run();
48.  servo1.write(servo1Pos);
49.  servo2.write(servo2Pos);
50.  servo3.write(servo3Pos);
51.  servo4.write(servo4Pos);
52.  if(button1Pressed){
53.  switch(button1Presses){
54.  case 1:
55.  servo1PosSaves[0] = servo1Pos;
56.  servo2PosSaves[0] = servo2Pos;
57.  servo3PosSaves[0] = servo3Pos;
58.  servo4PosSaves[0] = servo4Pos;
59.  digitalWrite(LED1, HIGH);
60.  terminal.println("Pos 1 saved");
61.  break;
62.  case 2:
63.  servo1PosSaves[1] = servo1Pos;
64.  servo2PosSaves[1] = servo2Pos;
65.  servo3PosSaves[1] = servo3Pos;
66.  servo4PosSaves[1] = servo4Pos;
67.  digitalWrite(LED2, HIGH);
68.  terminal.println("Pos 2 saved");
69.  break;
70.  case 3:
71.  servo1PosSaves[2] = servo1Pos;
72.  servo2PosSaves[2] = servo2Pos;
73.  servo3PosSaves[2] = servo3Pos;
74.  servo4PosSaves[2] = servo4Pos;
75.  digitalWrite(LED3, HIGH);

```

```

76. terminal.println("Pos 3 saved");
77. break;
78. case 4:
79. servo1PosSaves[3] = servo1Pos;
80. servo2PosSaves[3] = servo2Pos;
81. servo3PosSaves[3] = servo3Pos;
82. servo4PosSaves[3] = servo4Pos;
83. digitalWrite(LED4, HIGH);
84. terminal.println("Pos 4 saved");
85. break;
86. case 5:
87. servo1PosSaves[4] = servo1Pos;
88. servo2PosSaves[4] = servo2Pos;
89. servo3PosSaves[4] = servo3Pos;
90. servo4PosSaves[4] = servo4Pos;
91. digitalWrite(LED5, HIGH);
92. terminal.println("Pos 5 saved");
93. break;
94. }
95. }
96. if(button2Pressed){
97. for(int i = 0; i < 5; i++){
98. servo1.write(servo1PosSaves[i]);
99. servo2.write(servo2PosSaves[i]);
100. servo3.write(servo3PosSaves[i]);
101. servo4.write(servo4PosSaves[i]);
102. terminal.print(" Angles: "); terminal.println(i + 1);
103. terminal.println(servo1PosSaves[i]);
104. terminal.println(servo2PosSaves[i]);
105. terminal.println(servo3PosSaves[i]);

```

```

106. delay(1200);
107. }
108. terminal.flush();
109. }
110. }
111. BLYNK_WRITE(V1){
112.   servo1Pos = (param.asInt());
113. }
114. BLYNK_WRITE(V2) {
115.   servo2Pos = (param.asInt());
116. }
117. BLYNK_WRITE(V3) {
118.   servo3Pos = (param.asInt());
119. }
120. BLYNK_WRITE(V6) {
121.   servo4Pos = (param.asInt());
122. }
123. BLYNK_WRITE(V4){
124.   if(param.asInt() == 1){
125.     button1Presses++;
126.     button1Pressed = true;
127.   }
128.   else {
129.     button1Pressed = false;
130.   }
131. }
132. BLYNK_WRITE(V5){
133.   button2Pressed = true;
134. }
135. BLYNK_WRITE(V7){ //Reset

```



```
136. button1Presses = 0;
137. button2Pressed = false;
138. for(int i = 0; i < 5; i++){
139.   servo1PosSaves[i] = 0;
140.   servo2PosSaves[i] = 0;
141.   servo3PosSaves[i] = 0;
142.   servo4PosSaves[i] = 0;
143. }
144. digitalWrite(LED1, LOW);
145. digitalWrite(LED2, LOW);
146. digitalWrite(LED3, LOW);
147. digitalWrite(LED4, LOW);
148. digitalWrite(LED5, LOW);
149. terminal.println("Everything reset");
150. }
```