

## DAFTAR GAMBAR

Gambar 2.1 Contoh Code .....	11
Gambar 2.2 Contoh pinMode().....	11
Gambar 2.3 Contoh set/get power .....	12
Gambar 2.4 Budidaya semut rangrang .....	16
Gambar 3.1 Tahapan Alur Penelitian .....	18
Gambar 3.2 Desain Aplikasi Blynk .....	21
Gambar 3.3 Desain Box Kontrol Tampak Samping .....	21
Gambar 3.4 Desain Box Kontrol Tampak Atas .....	22
Gambar 3.5 Desain Alat .....	22
Gambar 3.6 Flowchart Sistem .....	23
Gambar 3.7 Penempatan Alat Tampak Depan.....	24
Gambar 3.8 Penempatan Alat Tampak Samping .....	24
Gambar 3.9 Penempatan Alat Tampak Atas.....	25
Gambar 4.1 Tampilan Blynk IoT.....	30
Gambar 4.2 <i>Define</i> dan <i>Include</i> Blynk .....	31
Gambar 4.3 Kode <i>Connect</i> Blynk.....	31
Gambar 4.4 <i>myTimerEvent</i> .....	32
Gambar 4.5 <i>Datastream 1</i> .....	32
Gambar 4.6 <i>Datastream 2</i> .....	33
Gambar 4.7 Pengaturan Virtual Pin pada Blynk.....	33
Gambar 4.8 Pengaturan Event .....	34
Gambar 4.9 Edit Event .....	34
Gambar 4.10 Tampilan Lcd.....	35
Gambar 4.11 Setup Lcd .....	35
Gambar 4.12 Kode Tampilan Lcd .....	35
Gambar 4.13 Setup Relay .....	35
Gambar 4.14 Relay Humidifier .....	37
Gambar 4.15 Relay Pompa Dc .....	38
Gambar 4.16 Grafik Hasil Pengaturan Suhu .....	46
Gambar 4.17 Grafik Hasil Pengaturan Kelembaban .....	46
Gambar 4.18 Grafik Perbedaan Suhu .....	47
Gambar 4.19 Penutupan Lubang Pintu .....	47
Gambar 4.20 Pengaturan Suhu Setelah Ditutup .....	51
Gambar 4.21 Pengaturan Kelembaban Setelah Ditutup .....	52
Gambar 4.22 Perbandingan Suhu Setelah Ditutup .....	52
Gambar 4.23 Perbandingan Pengaturan Suhu .....	53
Gambar 4.24 Perbandingan Pengaturan Kelembaban .....	53
Gambar 4.25 Kondisi Semut Selama Pengujian.....	55

*Halaman ini sengaja dikosongkan*

## DAFTAR TABEL

Tabel 2.1 Kajian Pustaka .....	7
Tabel 2.2 Mode DIP Switch .....	10
Tabel 2.3 Parameter elektrik hc-sr04.....	12
Tabel 2.4 Konfigurasi dht11 .....	15
Tabel 3.1 Perangkat keras yang digunakan .....	17
Tabel 3.2 Perangkat lunak yang digunakan .....	18
Tabel 3.3 Pengujian Hipotesa 1 .....	26
Tabel 3.4 Pengujian Hipotesa 2.....	27
Tabel 4.1 Uji Sensor 1 Hcsr04.....	39
Tabel 4.2 Uji Sensor 2 Hcsr04.....	39
Tabel 4.3 Uji Sensor 3 Hcsr04.....	39
Tabel 4.4 Pengukuran Suhu dan Kelembaban .....	40
Tabel 4.5 Uji Pemberian Air Gula Motor 2 .....	41
Tabel 4.6 Uji Pemberian Air Gula Motor 2.....	41
Tabel 4.7 Uji Pemberian Air Gula Motor 3 .....	41
Tabel 4.8 Hasil Pengujian Suhu Hari Pertama .....	43
Tabel 4.9 Hasil Pengujian Suhu Hari Kedua .....	44
Tabel 4.10 Hasil Pengujian Kelembaban.....	45
Tabel 4.11 Hasil Pengujian Suhu Hari Pertama setelah ditutup .....	48
Tabel 4.12 Hasil Pengujian Suhu Hari Kedua setelah ditutup.....	49
Tabel 4.13 Hasil Pengaturan Kelembaban Setelah Ditutup.....	50
Tabel 4.14 Hasil Pemberian air gula.....	54

# LAMPIRAN 1

## CODING PROGRAM PADA ARDUINO

```
1 #include<LiquidCrystal_I2C_Hangul.h>
2 #include <DFRobot_DHT11.h>
3
4 #include "WiFiEsp.h"
5
6 //DHT11
7 DFRobot_DHT11 DHT;
8 #define DHT11_PIN 4
9
10 //Blynk
11 #define BLYNK_TEMPLATE_ID          "TMPLlpgW4etB"
12 #define BLYNK_DEVICE_NAME         "Clickstart Device"
13 #define BLYNK_AUTH_TOKEN          "ln0GSN7mhD0eX6Dsy2qdkIqRUg0v6$Xg"
14 // Comment this out to disable prints and save space
15 #define BLYNK_PRINT Serial
16 #include <ESP8266_Lib.h>
17 #include <BlynkSimpleShieldEsp8266.h>
18
19 //trigpin dan echopin HC-SR04
20 const int trigPin = 6, echoPin = 5, trigPin1 = 8, echoPin1 = 7, trigPin2 = 10, echoPin2 = 9;
21
22 LiquidCrystal_I2C_Hangul lcd(0x27,16,2); // set the LCD address to 0x27 for a 16 chars and 2 line display
23
24 long duration, duration1, duration2; // variable for the duration of sound wave travel
25 float distance, distance1, distance2; // variable for the distance measurement
26 String kipas = "OFF", heater = "OFF", humidifier = "OFF";
27 int relayPin1 = 22, relayPin2 = 23, relayPin3 = 34, relayPin4 = 25, relayPin5 = 27; //Assign Pin 8 to the relay control/signal pin
28 int subumin, suhumax, kelembabanmin, kelembabanmax, heatermin, heatermax;
29 unsigned int notif1, notif2, notif3, notif4, a, b, c, d;
30
31 char auth[] = BLYNK_AUTH_TOKEN;
```

```
32 // Your WiFi credentials.
33 // Set password to "" for open networks.
34 char ssid[] = "NM";
35 char pass[] = "Maleslapolapo";
36 // Hardware Serial on Mega, Leonardo, Micro...
37 #define EspSerial Serial3
38 // Your ESP8266 baud rate:
39 #define ESP8266_BAUD 115200
40 ESP8266 wifi(&EspSerial);
41 BlynkTimer timer;
42 // This function is called every time the Virtual Pin 0 state changes
43 BLYNK_WRITE(V0)
44 {
45   // Set incoming value from pin V0 to a variable
46   int value = param.asInt();
47
48   // Update state
49   Blynk.virtualWrite(V1, value);
50 }
51 BLYNK_WRITE(V12)
52 {
53   // Set incoming value from pin V12 to a variable
54   int suhu_min = param.asInt();
55   suhumin = suhu_min;
56   // Update state
57   Blynk.virtualWrite(V12, suhu_min);
58 }
59 BLYNK_WRITE(V13)
60 {
61   // Set incoming value from pin V13 to a variable
62   int suhu_max = param.asInt();
```

```

63  suhumax = suhu_max;
64  // Update state
65  Blynk.virtualWrite(V13, suhu_max);
66  }
67  BLYNK_WRITE(V14)
68  {
69  // Set incoming value from pin V14 to a variable
70  int kelembaban_min = param.asInt();
71  kelembabanmin = kelembaban_min;
72  // Update state
73  Blynk.virtualWrite(V14, kelembaban_min);
74  }
75  BLYNK_WRITE(V15)
76  {
77  // Set incoming value from pin V15 to a variable
78  int kelembaban_max = param.asInt();
79  kelembabanmax = kelembaban_max;
80  // Update state
81  Blynk.virtualWrite(V15, kelembaban_max);
82  }
83  BLYNK_WRITE(V16)
84  {
85  // Set incoming value from pin V16 to a variable
86  int heater_min = param.asInt();
87  heatermin = heater_min;
88  // Update state
89  Blynk.virtualWrite(V16, heater_min);
90  }
91  BLYNK_WRITE(V17)
92  {
93  // Set incoming value from pin V17 to a variable
94
95  int heater_max = param.asInt();
96  heatermax = heater_max;
97  // Update state
98  Blynk.virtualWrite(V17, heater_max);
99  }
100 // This function is called every time the device is connected to the Blynk.Cloud
101 BLYNK_CONNECTED()
102 {
103 // Change Web Link Button message to "Congratulations!"
104 Blynk.setProperty(V3, "offImageUrl", "https://static-image.nyc3.cdn.digitaloceanspaces.com/general/fce/congratulations.png");
105 Blynk.setProperty(V3, "onImageUrl", "https://static-image.nyc3.cdn.digitaloceanspaces.com/general/fce/congratulations_pressed.png");
106 Blynk.setProperty(V3, "url", "https://docs.blynk.io/en/getting-started/what-do-i-need-to-blynk/how-quickstart-device-was-made");
107 }
108 // This function sends Arduino's uptime every second to Virtual Pin 2.
109 void myTimerEvent()
110 {
111 // You can send any value at any time.
112 // Please don't send more that 10 values per second.
113
114 Blynk.virtualWrite(V4, DHT.temperature);
115 Blynk.virtualWrite(V5, DHT.humidity);
116 Blynk.virtualWrite(V6, distance);
117 Blynk.virtualWrite(V7, distancel);
118 Blynk.virtualWrite(V8, distancel2);
119 Blynk.virtualWrite(V9, kipas);
120 Blynk.virtualWrite(V10, heater);
121 Blynk.virtualWrite(V11, humidifier);
122 if (DHT.temperature >= suhumax && notifl == 0)
123 {
124 a = suhumax - 2;
125 notifl = 1;
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200

```

```

125     Blynk.logEvent("suhu_diatas_ideal");
126 } else if(DHT.temperature < a){
127     notif1 = 0;
128 }
129 if (DHT.temperature <= heatermin && notif2 == 0)
130 {
131     b = heatermin + 2;
132     notif2 = 1;
133     Blynk.logEvent("suhu_dibawah_ideal");
134 } else if(DHT.temperature < b){
135     notif2 = 0;
136 }
137 if (DHT.humidity >= kelembabanmax && notif3 == 0)
138 {
139     c = kelembabanmax - 2;
140     notif3 = 1;
141     Blynk.logEvent("kelembaban_diatas_ideal");
142 } else if(DHT.humidity < c){
143     notif3 = 0;
144 }
145 if (DHT.humidity <= kelembabanmin && notif4 == 0)
146 {
147     d = kelembabanmin + 2;
148     notif4 = 1;
149     Blynk.logEvent("kelembaban_dibawah_ideal");
150 } else if(DHT.humidity < d){
151     notif4 = 0;
152 }
153 }
154
155 void lcdset(){
156
157     lcd.init();
158     lcd.clear();
159     lcd.backlight(); // Make sure backlight is on
160 }
161
162 void hcsrset(){
163     pinMode(trigPin, OUTPUT); // Sets the trigPin as an OUTPUT
164     pinMode(echoPin, INPUT); // Sets the echoPin as an INPUT
165     pinMode(trigPin1, OUTPUT); // Sets the trigPin as an OUTPUT
166     pinMode(echoPin1, INPUT); // Sets the echoPin as an INPUT
167     pinMode(trigPin2, OUTPUT); // Sets the trigPin as an OUTPUT
168     pinMode(echoPin2, INPUT); // Sets the echoPin as an INPUT
169 }
170
171 void relayset(){
172     pinMode(relayPin1, OUTPUT); //Setting the Relay pin as an Output Pin
173     pinMode(relayPin2, OUTPUT); //Setting the Relay pin as an Output Pin
174     pinMode(relayPin3, OUTPUT); //Setting the Relay pin as an Output Pin
175     pinMode(relayPin4, OUTPUT); //Setting the Relay pin as an Output Pin
176     pinMode(relayPin5, OUTPUT); //Setting the Relay pin as an Output Pin
177     pinMode(relayPin6, OUTPUT); //Setting the Relay pin as an Output Pin
178
179     digitalWrite(relayPin4, HIGH); //memastikan relay pin 4(humidifier) mati terlebih dahulu.
180     digitalWrite(relayPin5, HIGH); //memastikan relay pin 5(heater) mati terlebih dahulu
181     digitalWrite(relayPin6, HIGH); //memastikan relay pin 6(kipas) mati terlebih dahulu
182 }
183
184 void lcdloop(){
185     DHT.read(DHT11_PIN);
186     lcd.setCursor(0,0); //Set cursor to character 2 on line 0
187     lcd.print("Suhu = ");
188     lcd.print(DHT.temperature);

```

```

187 lcd.print(" °C");
188 lcd.setCursor(0,1); //Move cursor to character 2 on line 1
189 lcd.print("Kelembaban = ");
190 lcd.print(DHT.humidity);
191 lcd.print("%");
192 delay(1000);
193 }
194
195 void relayloop(){
196
197 //humidifier
198 if (DHT.humidity <= kelembabanmin) {
199     digitalWrite(relayPin4, LOW); //humidifier menyala
200     humidifier = "ON";
201 }
202 if (DHT.humidity >= kelembabanmax) {
203     digitalWrite(relayPin4, HIGH); //humidifier mati
204     humidifier = "OFF";
205 }
206 //kipas angin
207 if (DHT.temperature >= suhumax) {
208     digitalWrite(relayPin5, LOW); //kipas menyala
209     kipas = "ON";
210 }
211 if (DHT.temperature <= suhumin) {
212     digitalWrite(relayPin5, HIGH); //kipas mati
213     kipas = "OFF";
214 }
215 //lampu heater
216 if (DHT.temperature <= heatermin) {
217     digitalWrite(relayPin6, LOW); //heater menyala
218
219     heater = "ON";
220 }
221 if (DHT.temperature >= heatermax) {
222     digitalWrite(relayPin6, HIGH); //heater mati
223     heater = "OFF";
224 }
225 // sensor hcsr04 ke-1 dan motor 1
226 // Clears the trigPin condition
227 digitalWrite(trigPin, LOW);
228 delayMicroseconds(2);
229 // Sets the trigPin HIGH (ACTIVE) for 10 microseconds
230 digitalWrite(trigPin, HIGH);
231 delayMicroseconds(10);
232 digitalWrite(trigPin, LOW);
233 // Reads the echoPin, returns the sound wave travel time in microseconds
234 duration = pulseIn(echoPin, HIGH);
235 // Calculating the distance
236 distance = duration * 0.034 / 2; // Speed of sound wave divided by 2 (go and back)
237 // Displays the distance on the Serial Monitor
238 if ( distance <= 11 ) // set value when you want to turn OFF water pump
239 {
240     digitalWrite(relayPin1, HIGH);
241 }
242 if ( distance > 11 ) // set value when you want to turn ON water pump
243 {
244     digitalWrite(relayPin1, LOW);
245 }
246 // sensor hcsr04 ke-2 dan motor 2
247 // Clears the trigPin condition
248 digitalWrite(trigPin1, LOW);

```



---

```

249 delayMicroseconds(2);
250 // Sets the trigPin HIGH (ACTIVE) for 10 microseconds
251 digitalWrite(trigPin1, HIGH);
252 delayMicroseconds(10);
253 digitalWrite(trigPin1, LOW);
254 // Reads the echoPin, returns the sound wave travel time in microseconds
255 duration1 = pulseIn(echoPin1, HIGH);
256 // Calculating the distance
257 distance1 = duration1 * 0.034 / 2; // Speed of sound wave divided by 2 (go and back)
258 // Displays the distance on the Serial Monitor
259 if ( distance1 <= 11 ) // set value when you want to turn OFF water pump
260 {
261   digitalWrite(relayPin2, HIGH);
262 }
263 if ( distance1 > 11 ) // set value when you want to turn ON water pump
264 {
265   digitalWrite(relayPin2, LOW);
266 }
267
268 // sensor hcsr04 ke-3 dan motor 3
269 // Clears the trigPin condition
270 digitalWrite(trigPin2, LOW);
271 delayMicroseconds(2);
272 // Sets the trigPin HIGH (ACTIVE) for 10 microseconds
273 digitalWrite(trigPin2, HIGH);
274 delayMicroseconds(10);
275 digitalWrite(trigPin2, LOW);
276 // Reads the echoPin, returns the sound wave travel time in microseconds
277 duration2 = pulseIn(echoPin2, HIGH);
278 // Calculating the distance
279 distance2 = duration2 * 0.034 / 2; // Speed of sound wave divided by 2 (go and back)

```

---

```

280 // Displays the distance on the Serial Monitor
281 if ( distance2 <= 11 ) // set value when you want to turn OFF water pump
282 {
283   digitalWrite(relayPin3, HIGH);
284 }
285 if ( distance2 > 11 ) // set value when you want to turn ON water pump
286 {
287   digitalWrite(relayPin3, LOW);
288 }
289 }
290 void setup()
291 {
292   // initialize serial for debugging
293   Serial.begin(115200);
294   //Connect Wifi
295   // Set ESP8266 baud rate
296   EspSerial.begin(ESP8266_BAUD);
297   delay(10);
298
299   Blynk.begin(auth, wifi, ssid, pass);
300   // You can also specify server:
301   //Blynk.begin(auth, wifi, ssid, pass, "blynk.cloud", 80);
302   //Blynk.begin(auth, wifi, ssid, pass, IPAddress(192,168,1,100), 8080);
303
304   // Setup a function to be called every second
305   timer.setInterval(1000L, myTimerEvent);
306   //LCD
307   lcdset();
308   // HCSR.
309   hcsrset();
310   //Relay

```

---

```
311 | relayset();
312 |
313 | //BLYNK
314 | timer.setInterval(100L, myTimerEvent);
315 |
316 | }
317 |
318 | void loop(){
319 |   lcdloop();
320 |   relayloop();
321 |   Blynk.run();
322 |   timer.run();
323 | }
```

*Halaman ini sengaja dikosongkan*

## LAMPIRAN 2

### FOTO ALAT







