CHAPTER 5
IMPLEMENTATION AND TESTING

5.1 Implementation

In this project C++ is the programming language used in arduino. When the program starts, the first is to read the pin and declared, then set the internet connection using the SSID and password, in this project SSID and password must match, if not match ESP-01 will not connect with internet and can not send data to IoT Thingspeak.

---

```c
const int pinSensor = 7; //pin infrared
int LDRreset1 = A1; //pin LDR1
int LDRreset2 = A0; //pin LDR2
int LDRreset3 = A2; //pin LDR3
int LDRreset4 = A3; //pin LDR4
int rileiLDR1 = 0;
int rileiLDR2 = 0;
int rileiLDR3 = 0;
int rileiLDR4 = 0;
// initialize sensing2 variables
int hitung = 0; //default hitung 0
int kondisi1 = 0; //default hitung 0
int status1;

Servo servoS;

/// keperluan IoT ke thingspeak

char ssid[] = "m2"; // your network SSID (name)
char pass[] = ""; // your network password
int status = WL_IDLE_STATUS; // the WifI radio's status
```

---

Illustration 5.1: Code pin and SSID

SSID is used to connect with IoT, if it is connected to the internet so data from IR Obstacle sensor can be sent to IoT Thingspeak. Sending data from IR Obstacle sensor to IoT Thingspeak takes 6 seconds until 8 second depending on internet speed.
Queue and upload Thingspeak is used to save data and send to Thingspeak from IR Obstacle. Data from IR Obstacle will be uploaded based on the time set, if the time is appropriate then the data will be sent in the order of data already entered in the queue. Queue can hold as many 200 data and will send 60 seconds once. The queue data to be sent to IOT Thingspeak have to sequence (first in first out) so that data does not overlap when going into Thingspeak.

Illustration 5.2: Set queue and upload Thingspeak

```c
// tips for queue untuk antrean upload thingspeak
const uint8_t QUEUE_SIZE = 200;
MD_ClrQueue antreanThingspeak(QUEUE_SIZE, sizeof(int));

// jalankan prosedur uploadThingspeak dengan interval 60 detik sekali
void uploadThingspeak(Task* me){
    Task taskUploadThingspeak(60000, uploadThingspeak);
    // jalankan prosedur loop/program utama dengan interval 1 ms sekali
    void mainLoop(Task* me);
    Task taskMainLoop(1, mainLoop);
}
15

Illustration 5.3: Code IR Obstacle

This program is inputted into IR Obstacle sensor so can be work. IR Obstacle sensor will be counting vehicle axle, then data from Obstacle IR sensor will be sent to IoT. When data from Obstacle IR sensor has been sent to IoT Thingspeak, IR Obstacle variable will return 0 (reset). Before being sent to IoT the data will be saved first to queue, when data is saved to IoT then the serial monitor will appear "Data baru ditampung dalam antrian", but when failed will appear notification "Gagal menampung data baru dalam antrian".
ESP-01 code will be used to connect to the internet using SSID and password, installation Esp-01 to connect internet will show on LCD as notification. ESP-01 is very important because the data from the IR Obstacle sensor can be sent to IoT Thingspeak.

5.2 Testing

After all the sensors are working, next step is test using a miniature vehicle. Testing using different type of vehicle, this test sees that the entire sensor works well or not. in this test the car runs with an average speed of
0.35 m/second, but it also tries at a higher speed but the result is not successful (not send to IoT Thingspeak)

Test Vehicle when entering at toll gate

<table>
<thead>
<tr>
<th>Test</th>
<th>Class of Vehicle</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Class 1</td>
<td>Success</td>
</tr>
<tr>
<td>2</td>
<td>Class 1</td>
<td>Success</td>
</tr>
<tr>
<td>3</td>
<td>Class 1</td>
<td>Failed</td>
</tr>
<tr>
<td>4</td>
<td>Class 2</td>
<td>Success</td>
</tr>
<tr>
<td>5</td>
<td>Class 2</td>
<td>Success</td>
</tr>
<tr>
<td>6</td>
<td>Class 2</td>
<td>Failed</td>
</tr>
<tr>
<td>7</td>
<td>Class 3</td>
<td>Success</td>
</tr>
<tr>
<td>8</td>
<td>Class 3</td>
<td>Success</td>
</tr>
<tr>
<td>9</td>
<td>Class 3</td>
<td>Failed</td>
</tr>
<tr>
<td>10</td>
<td>Class 4</td>
<td>Success</td>
</tr>
<tr>
<td>11</td>
<td>Class 4</td>
<td>Success</td>
</tr>
<tr>
<td>12</td>
<td>Class 4</td>
<td>Failed</td>
</tr>
</tbody>
</table>

Table 5.1: Toll Gate Testing Table
Table 5.1 shows the overall test results. From this test success or failure is influenced by the speed of the vehicle. When the data fails the reads are "1", the success rate of data is read according to the number of vehicle axis.

After testing use different types of vehicles, data from IR Obstacle will be sent to IoT Thingspeak. The data show on IoT Thingspeak is the number of axis, date and time. The number of tests will appear and the last time the data is uploaded to IoT Thingpeak.