

## **BAB IV**

### **IMPLEMENTASI SISTEM**

#### **4.1 Hasil Perancangan**

Stelah selesai tahap perancangan yang akan dilakukan adalah menganalisa hasil dari pembuatan alat. Terbagi menjadi dua bagian pada pembuatan alat ini yaitu pertama pembuatan hardware mikrokontroler *arduino nano* sebagai pengatur beban dan pemroses data. Pada bagian kedua adalah *software* pada *arduino* dengan menggunakan dua program *transmitter* dan *receiver*

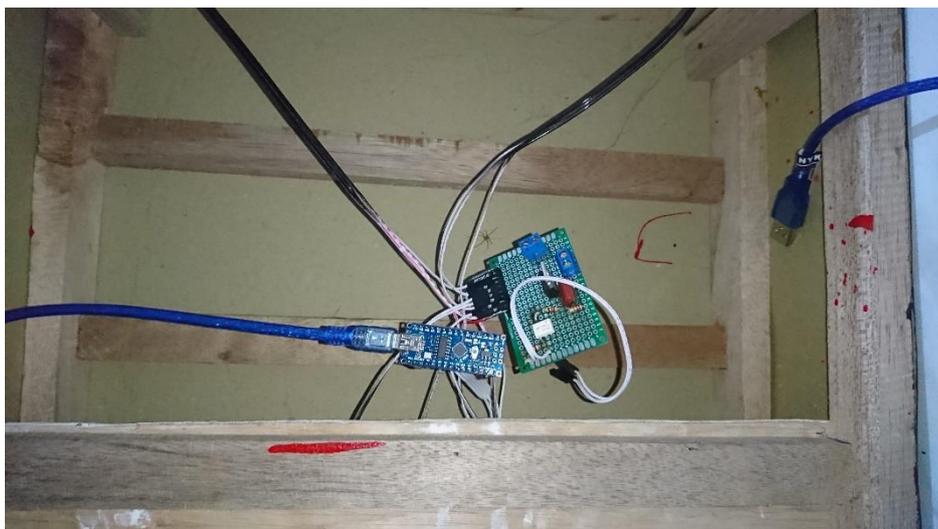
##### 4.1.1 Hasil Pembuatan *Hardware*

Dalam pembuatan *hardware* penulis membuat dua bagian rangkain dan cover pertama penulis membuat maket miniatur hutan dimana empat buah sensor dipasang disetiap sudut dan yang kedua bagian *receiver* yang berbentuk box, dengan menggunakan box ini terjamin dari segi keamanan dalam pengoperasiannya serta bentuk yang rapi dan minimalis mudah di bawa dengan bentuk yang cukup simpel.



**Gambar 4.1** Maket Bagian Atas

Pada gambar 4.1 terlihat bagian atas maket terdapat sensor asap di setiap sudut sebagai input data asap yang di proses oleh *arduino* serta adanya miniatur pohon sungai dan rumah.



**Gambar 4.2** Maket Bagian Bawah

Pada gambar 4.3 terlihat bagian bawah terdapat rangkaian *mikrokontroler* dimana terdapat arduino dan transmitter serta kabel penghubung rangkaian



**Gambar 4.3** Box Bagian Atas

Pada gambar 4.2 terlihat bagian atas terdapat tulisan sensor dari 1 sampai 4 ini menjelaskan bahwa bila lampu 1 menyala menandakan bahwa sensor satu dalam posisi mendeteksi terjadinya asap begitu juga yang lainnya dengan itu mempermudah pengguna.



**Gambar 4.4** Box Bagian Depan Atas

Pada gambar 4.3 terlihat bagian depan terdapat *buzzer* yang akan menjadi alarm atau indikator bila salah satu ataupun seluruh sensor mendeteksi adanya asap



**Gambar 4.5** Box Bagian Samping kanan

Pada gambar 4.4 terlihat bagian samping kanan terdapat saklar pemutus atau untuk menyalakan rangkaian *receiver*



**Gambar 4.6** Box Tampak Bagian Belakang

Pada gambar 4.5 terlihat bagian belakang terdapat catu daya atau baterai sebagai sumber dari rangkaian *receiver*



```
Nrftransmitter
#include <SPI.h>
#include <nRF24L01.h>
#include <RF24.h>
int pesan[1];
RF24 radio(9,10);
const int pipe = 33;
int pushbutton = 5;int pushbutton2 = 6;int pushbutton3 = 7;int pushbutton4 = 8;

void setup(void) {
  radio.begin();
  radio.setChannel(100);
  radio.openWritingPipe(pipe);
  radio.stopListening();
  Serial.begin(9600);
  pinMode(pushbutton, INPUT); pinMode(pushbutton2, INPUT);
  pinMode(pushbutton3, INPUT); pinMode(pushbutton4, INPUT);
}

void loop(void) {
  radio.write(pesan, 1);
  if (digitalRead(pushbutton) == HIGH) { pesan[0] = 19;radio.write(pesan, 1);Serial.println("Led 1");}
  if (digitalRead(pushbutton2) == HIGH) { pesan[0] = 29;radio.write(pesan, 1);}
  if (digitalRead(pushbutton3) == HIGH) { pesan[0] = 39;radio.write(pesan, 1);}
  if (digitalRead(pushbutton4) == HIGH) { pesan[0] = 49;radio.write(pesan, 1);}
  else {pesan[0] = 99;radio.write(pesan, 1);}
}
```

Done compiling.  
Sketch uses 3320 bytes (10%) of program storage space. Maximum is 30720 bytes.  
Global variables use 213 bytes (10%) of dynamic memory, leaving 1835 bytes for local variables. Maximum is 2048

**Gambar 4.7** Programing *Transmitter*



```
Nrfreceiver_baru
#include <SPI.h>
#include <nRF24L01.h>
#include <RF24.h>
int pesan[1];
RF24 radio(9,10);
const uint64_t pipe = 0xE8E8F0F0E1LL;
int LED = 5;int LED2 = 6;int LED3 = 7;int LED4 = 8;
int bazzzer = 4;
void setup(void) {
  radio.begin();
  radio.openReadingPipe(1,pipe);
  radio.startListening();
  pinMode(LED, OUTPUT);pinMode(LED2, OUTPUT);pinMode(LED3, OUTPUT);pinMode(LED4, OUTPUT);pinMode(bazzzer, OUTPUT);}

void loop(void) {
  radio.read(pesan, 1);
  if (pesan[0] == 19){digitalWrite(LED, HIGH);digitalWrite(bazzzer, HIGH);delay(50);}
  if (pesan[0] == 29){digitalWrite(LED2, HIGH);digitalWrite(bazzzer, HIGH);delay(50);}
  if (pesan[0] == 39){digitalWrite(LED3, HIGH);digitalWrite(bazzzer, HIGH);delay(50);}
  if (pesan[0] == 49){digitalWrite(LED4, HIGH);digitalWrite(bazzzer, HIGH);delay(50);}
  else// (pesan[0] == 0)
  {digitalWrite(LED, LOW);digitalWrite(LED2, LOW);
  digitalWrite(LED3, LOW);digitalWrite(LED4, LOW);digitalWrite(bazzzer, LOW);
  }}
}
```

Done compiling.  
Sketch uses 2162 bytes (7%) of program storage space. Maximum is 30720 bytes.  
Global variables use 28 bytes (1%) of dynamic memory, leaving 2020 bytes for local variables. Maximum is 2048

**Gambar 4.8** Programing *Receiver*

Dan pada gambar 4.6 dan 4.7 terlihat program yang dibuat sebagai alur sistem kerja alat terlihat dua program yaitu program pada sistem *transmitter* dan pada sistem *receiver*.

## **4.2 Cara Kerja Sistem**

Pada implementasinya sistem ini adalah sebuah perangkat keras dan perangkat lunak yang terintegrasi menjadi satu kesatuan yang mampu menyelesaikan, mempermudah, dan memberikan efisiensi dan efektivitas waktu dalam mendeteksi kebakaran hutan dan lahan melalui sensor asap yang dihubungkan ke *output* LED dengan menggunakan Radio *Transmitter*.

Cara kerja sistem ini sendiri yaitu pertama-tama menyalakan sistem hardware *Arduino nano* yang terhubung ke kabel usb, *Arduino nano* akan menghubungkan koneksi melalui radio *transmitter-receiver*. *Mikrokontroler* menerima data dari modul sensor asap MQ2 yang kemudian diproses dan dikirimkan ke *Arduino nano receiver* melalui *transmitter* untuk menyalakan indikator LED dan *buzzer* sebagai indikator

## **4.3 Pembahasan Unjuk Kerja Sistem**

### **4.3.1 Menghidupkan rangkaian transmitter**

Pertama menghidupkan rangkaian *transmitter* yang terdapat sensor MQ02 sebagai input serta mengecek apakah rangkaian benar-benar keadaan normal dan bekerja dengan baik, dapat terlihat dari lampu indikator pada sensor serta mensetting sensitifitas pada sensor secara manual karena memakai sistem digital agar mudah mendeteksi adanya asap, terlihat pada gambar 4.8 rangkain dalam

keadaan menyala dengan melihat dari lampu indikator pada sensor MQ02 posisi menyala.



**Gambar 4.9** Posisi Rangkaian *Transmitter* Menyala

#### 4.3.2 Menghidupkan rangkaian *receiver*

kedua menghidupkan saklar pada kotak *receiver* agar alat *standby* untuk menerima data dari sensor yang akan dikirim melalui transmitter bila sensor MQ02 mendeteksi adanya asap yang mengandung gas yang mudah terbakar, terlihat pada gambar 4.2 rangkaian dalam keadaan menyala dengan melihat dari lampu indikator pada *arduino* posisi menyala.

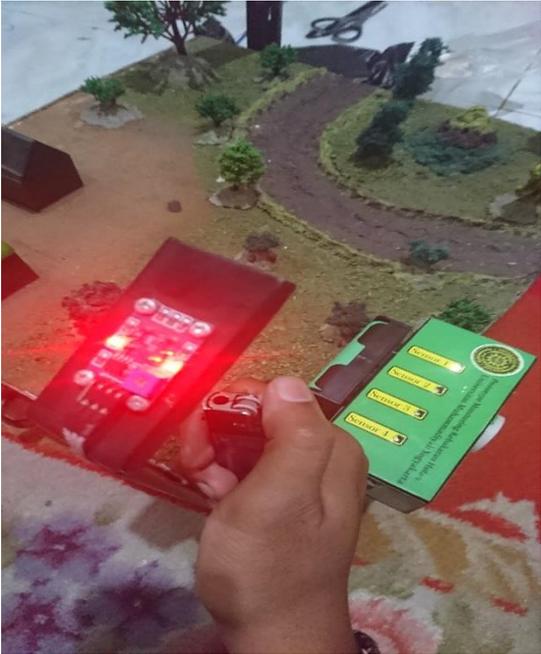


**Gambar 4.10** Posisi Rangkaian *Receiver* Menyala

#### 4.3.3 Pengujian Alat

Dalam tahap pengujian pendeteksi kebakaran hutan dan lahan ini akan dilakukan uji coba dengan mendeteksi sensor asap menggunakan gas amonia dan pembakaran yang menghasilkan asap. Pada pengaplikasian, proses awal yang dilakukan sistem yaitu menghubungkan arduino bagian sensor dengan catu daya agar *mikrokontroler* dan sensor dapat berjalan dan memproses data. Kemudian proses pendeteksi kebakaran hutan dan lahan dapat dibaca oleh sensor, dikirm ke receiver dan diprose oleh arduino untuk dikeluarkan ke led dan *buzzer* sebagai indikator serta jarak maksimal komunikasi alat. Terlihat pada tabel 5.1 dan 5.2.

**Tabel 4.1** Pengujian Pendeteksi Kebakaran Hutan dan Lahan

No	Pengujian prototype monitoring Kebakaran Hutan dan Lahan	Keterangan
1	<p>Pendeteksi Sensor 1 Menggunakan Gas Amonia mewakili sensor 3 dan 4</p> 	<p>Pada saat gas amonia ditekan maka sensor 1 akan mendeteksi dan diterima oleh <i>receiver</i> dengan indikator led 1 menyala serta <i>buzzer</i> berbunyi</p>
2	<p>Pendeteksi Sensor 2 Menggunakan Gas Amonia mewakili sensor 3 dan 4</p> 	<p>Pada saat gas amonia ditekan maka sensor 2 akan mendeteksi dan diterima oleh <i>receiver</i> dengan indikator led 2 menyala serta <i>buzzer</i> berbunyi</p>

No	Pengujian prototype monitoring Kebakaran Hutan dan Lahan	Keterangan
3	<p>Pendeteksi Sensor 3 Menggunakan obat nyamuk mewakili sensor 1 dan 2</p> 	<p>Pada saat gas amonia ditekan maka sensor 3 akan mendeteksi dan diterima oleh <i>receiver</i> dengan indikator led 3 menyala serta <i>buzzer</i> berbunyi</p>
4	<p>Pendeteksi Sensor 4 Menggunakan Pembakaran Obat nyamuk</p> 	<p>Pada saat gas amonia ditekan maka sensor 4 akan mendeteksi dan diterima oleh <i>receiver</i> dengan indikator led 4 menyala serta <i>buzzer</i> berbunyi</p>

Tabel 4.2 Pengujian Jarak Komunikasi Radio NRF24L01

No	Jarak (Meter)	Status komunikasi (Terhubung / Tidak Terhubung)
1	3	Terhubung
2	6	Terhubung
3	9	Terhubung
4	12	Terhubung
5	15	Terhubung
6	18	Tidak Terhubung