

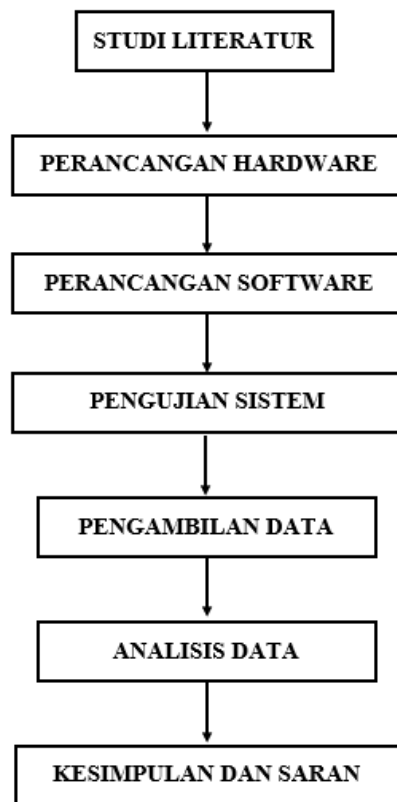
## **BAB 3**

### **METODE PENELITIAN**

Bab 3 ini akan menjelaskan mengenai bagaimana bentuk metode penelitian yang digunakan untuk merealisasikan rumusan masalah yang ada. Disini akan dijelaskan secara rinci mengenai alur penelitian yang dilakukan, alat dan bahan yang digunakan, perancangan *hardware*, perancangan *software*, serta pengujian sistem. Pada bab ini akan dijelaskan setiap proses yang dilalui dalam penelitian, mulai dari perumusan masalah, pengambilan data, analisis data, hingga pengambilan kesimpulan. Selanjutnya, penulis juga akan menjelaskan alat dan bahan yang digunakan, baik *hardware* maupun *software* yang dibutuhkan. Perancangan *hardware* dan *software* juga akan dijelaskan secara rinci, termasuk diagram blok dan skematik rangkaian yang digunakan dalam penelitian. Terakhir, penulis akan menjelaskan tentang pengujian sistem, baik pengujian fungsional maupun pengujian performa, guna mengetahui sejauh mana sistem yang telah dirancang dapat berjalan dengan baik sesuai dengan yang diharapkan. Untuk dapat memahami lebih jelas akan dijelaskan sebagai berikut.

#### **3.1 ALUR PENELITIAN**

Pada penelitian ini ada beberapa tahapan-tahapan yang akan dilakukan demi mewujudkan hasil yang diharapkan. Tahapan dalam penelitian ini yaitu meliputi Studi Literatur, Perancangan *hardware*, Perancangan *Software*, Pengujian Sistem, Pengambilan data, Analisis data serta yang terakhir adalah kesimpulan dan saran Berikut ini merupakan tahapan pelaksanaan penelitian pemanfaat *peripheral PWM* pada alat pengusir hama burung yang akan dijelaskan melalui diagram alur pada gambar 3.1 berikut:



**Gambar 3.1 Diagram Alur Penelitian**

Diagram alur pada gambar 3.1 berisi tentang proses yang akan dilalui dalam melaksanakan penelitian ini. Penjelasan untuk tahapan penelitian dari diagram alur diatas adalah sebagai berikut:

1) Studi Literatur

Pada tahap ini mempunyai tujuan untuk mengumpulkan, mempelajari, serta mengolah bahan penelitian mengenai pemanfaatan peripheral PWM, hama burung, Mikrokontroler Arduino dan gelombang ultrasonik/infrasonik. Sumber yang digunakan dapat berasal dari buku, jurnal, dan artikel lain yang terpercaya.

2) Perancangan *Hardware*

Tahapan kedua yang dilakukan setelah melakukan studi literatur adalah Perancangan *Hardware*. Dari informasi yang sudah diperoleh pada tahap studi literatur, pada tahapan ini akan dirancang *hardware* yang digunakan sebagai objek penelitian. *Hardware* tersebut dapat berupa alat yang digunakan pada penelitian ini. Pada tahapan ini akan dijelaskan mengenai blok diagram dari *hardware* yang digunakan, hubungan antara

IC NE555 dengan Arduino UNO R3 (meliputi pin, data yang masuk dan pemanfaatan sistem PWM), hubungan antara Arduino UNO R3 dan *Buzzer* dan yang terakhir adalah bagaimana mekanisme pemanfaatan fitur *Arduino Sleep Mode*.

### 3) Perancangan *Software*

Setelah tahapan perancangan *hardware* sudah dilaksanakan, tahapan selanjutnya yaitu perancangan *software*. Tahap ini akan disusun program yang akan mendukung *hardware* yang sudah dirancang pada tahap sebelumnya. Pada tahapan ini akan dijelaskan mengenai flowchart atau diagram alur pada sistem penyusunan *software* yang digunakan.

### 4) Pengujian Sistem

Setelah tahap perancangan perangkat keras dan perancangan perangkat lunak telah berhasil diselesaikan, tahap selanjutnya dalam proses tersebut adalah pengujian sistem. Pengujian sistem sangat penting karena bertujuan untuk menentukan apakah alat yang telah dirancang sesuai dengan hasil yang diharapkan. Selama tahap pengujian ini, berbagai tahapan pengujian dilakukan. Tahapan ini mencakup pengujian IC NE555 dan arduino, mengevaluasi fungsionalitas arduino dan buzzer, menilai keefektifan fitur *Arduino Sleep Mode*, dan pada akhirnya melakukan pengujian komprehensif terhadap seluruh perangkat. Jika hasil yang diperoleh dari pengujian tersebut memenuhi kriteria yang diinginkan, maka tahap selanjutnya yang melibatkan pengumpulan data dapat dimulai.

### 5) Pengambilan Data

Setelah dilaksanakan tahapan pengujian sistem, hasil data yang diperoleh pada tahapan tersebut akan dikumpulkan untuk dapat dianalisis ke tahapan selanjutnya.

### 6) Analisis Data

Data yang sudah diperoleh pada tahapan pengumpulan data akan diolah dan di analisis pada tahapan ini. Proses analisis data berfungsi untuk menentukan keluaran dari penelitian yang telah dilaksanakan apakah sudah sesuai dengan tujuan yang diinginkan.

## 7) Kesimpulan dan Saran

Tahapan yang terakhir adalah penyusunan kesimpulan dan saran terkait hasil penelitian yang sudah dilaksanakan.

## 3.2 ALAT DAN BAHAN

### 3.2.1 *Hardware*

Dalam suatu penelitian, *hardware* dibutuhkan untuk mendukung proses berjalannya penelitian agar sesuai dengan yang diharapkan. Pada tabel 3.1 berikut ini akan dijelaskan mengenai *hardware* yang akan digunakan pada penelitian ini serta fungsinya.

**Tabel 3.1 *Hardware* yang digunakan**

No.	Nama <i>Hardware</i>	Fungsi
1.	Mikrokontroler Arduino UNO R3	Mendalikan dan memproses data dari perangkat <i>input</i> kemudian meneruskannya hasilnya ke perangkat <i>output</i> .
2.	IC NE555	Timer (pewaktu) yang dapat diprogram untuk membuat penundaan waktu tertentu.
3.	Breadboard	Board yang digunakan untuk menyusun rangkaian elektronik.
4.	<i>Buzzer</i>	Jenis perangkat suara yang dapat menghasilkan bunyi atau nada tertentu
5.	Kabel jumper female-male	Digunakan untuk menghubungkan komponen elektronik atau modul dengan mikrokontroler atau papan pengembangan seperti Arduino Uno R3.

No.	Nama <i>Hardware</i>	Fungsi
6.	Laptop	Menghubungkan perangkat mikrokontroler dengan menggunakan <i>software</i> Arduino IDE agar perangkat dapat berfungsi.

### 3.2.2 *Software*

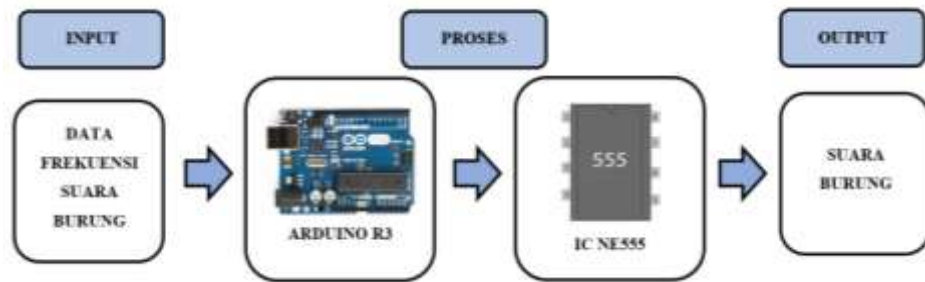
#### 1. *Arduino IDE*

Arduino IDE ialah *software* pemrograman yang memiliki fungsi utama untuk memprogram dan mengunggah kode ke *board* arduino atau mikrokontroler lain yang kompatibel. Dengan menggunakan Arduino IDE, pengguna dapat menulis program menggunakan bahasa pemrograman Arduino yang mudah dipahami. Setelah program selesai, pengguna dapat mengunggahnya ke *board* arduino melalui koneksi USB. Arduino IDE menyediakan berbagai fitur dan alat pengembangan, termasuk editor kode, pemantauan serial, serta kemampuan untuk mengompilasi dan mengunggah program ke dalam *board*. Dengan demikian, Arduino IDE memainkan peran kunci dalam proses pengembangan perangkat keras berbasis Arduino.

## 3.3 PERANCANGAN HARDWARE

### 3.3.1 Blok Diagram

Blok diagram yang akan untuk penelitian ini dapat dilihat pada Gambar 3.2 berikut. Pada blok diagram tersebut terdapat 3 tahapan pada penelitian yang terdiri dari tahapan Input, tahapan Proses dan tahapan Output.

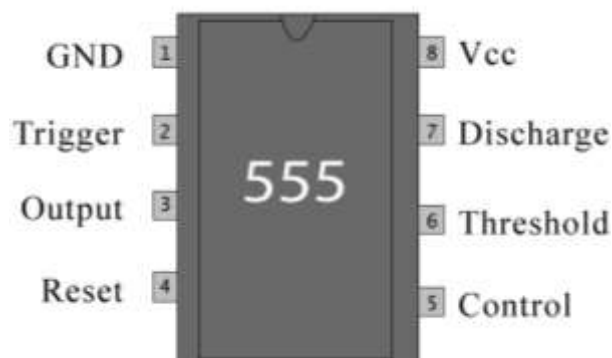


**Gambar 3.2 Blok Diagram Perancangan Sistem**

Dapat dilihat pada gambar 3.2 merupakan blok diagram yang digunakan pada penelitian pemanfaatan *pheriperal* pwm untuk otomasi alat pengusir hama burung berbasis sinyal ultrasonik. Untuk bentuk inputan pada alat tersebut merupakan data frekuensi suara burung yang akan dimasukkan dalam program arduino. Selanjutnya data tersebut akan dikirimkan melalui Arduino UNO R3. Pemanfaatan IC NE555 pada tahapan ini yaitu digunakan untuk memproses waktu pemutaran data frekuensi burung yang sebelumnya sudah di inputkan. Kemudian data yang sudah diproses hasilnya akan berupa suara ultrasonik yang outputnya akan dikeluarkan melalui *buzzer*.

### 3.3.2 Perancangan Rangkaian IC NE555

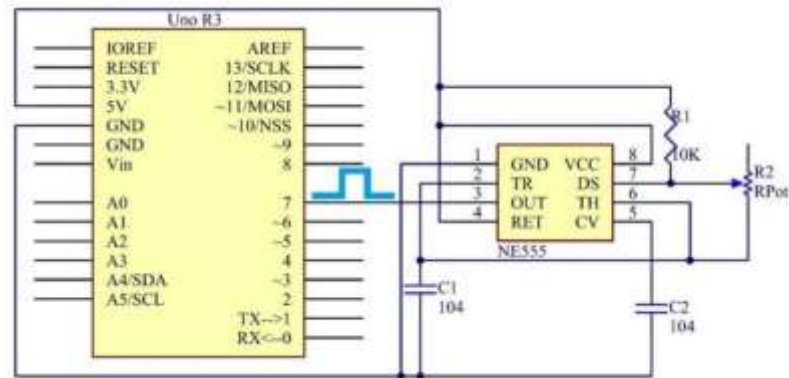
Rangkaian IC NE555 merupakan rangkaian elektronika yang digunakan sebagai IC *Timer* pada alat elektronik digital. IC NE555 pada dasarnya berfungsi untuk membangkitkan *clock*.



**Gambar 3.3 Pin Penyusun IC NE555**

Dapat dilihat pada gambar 3.3 merupakan pin penyusun IC NE555. IC NE555 tersusun dari 8 pin yang mempunyai fungsi serta karakteristiknya masing-masing. Fungsi dari setiap pin pada IC NE555 adalah sebagai berikut:

- a. Pin 1 (*Ground*): Terhubung pada *Ground*
- b. Pin 2 (*Trigger*): *Trigger* yang berfungsi sebagai pemicu agar pewaktuan bekerja. Ketika tegangan pada pin berkurang menjadi  $1/3$  dari VCC atau berada di ambang batas yang telah ditentukan oleh *board control*, maka terminal output akan mengirimkan ke *High Level*.
- c. Pin 3 (*Output*): *Output* yang dikeluarkan oleh IC NE555 dibagi menjadi dua keadaan yaitu 0 (*low*) atau 1 (*high*) sesuai dengan tingkatan yang diinputkan. Dalam keadaan high arus keluaran maksimumnya hingga 200mA.
- d. Pin 4 (*Reset*): *Reset* ini berfungsi ketika *output* yang dihasilkan berada di kondisi *low* (0) maka secara sistem akan berhenti secara otomatis kerja IC.
- e. Pin 5 (*CV*): *Control Voltage* berfungsi untuk mengontrol tegangan agar dibagikan secara merata pada threshold sebesar  $1/3$  serta  $2/3$  akan dibagikan ke Vcc.
- f. Pin 6 (*Threshold*): Pin ini berfungsi untuk menghentikan kinerja IC apabila tegangan pada Pin 6 berada dibawah  $2/3$  Vcc.
- g. Pin 7 (*Discharge*): Pin ini berfungsi untuk membuang muatan pada kapasitor yang disinkronkan dengan pin 3 yang menggunakan level logika yang sama. Tetapi pin ini tidak menghasilkan arus.
- h. Pin 8 (Vcc): Pin yang terhubung dengan Vcc. Terminal positif untuk IC timer NE555 yang memiliki tegangan mulai dari +4.5V hingga +16V.



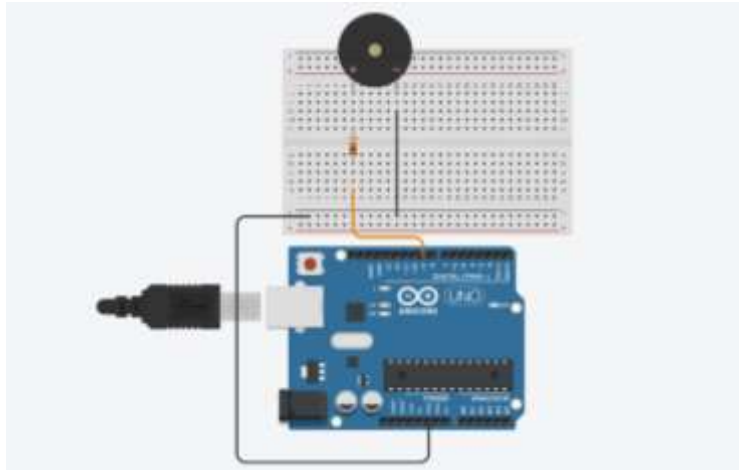
**Gambar 3.4 Rangkaian Skematik IC NE555**

Sesuai dengan penjelasan pin pada IC NE555 pada gambar 3.3, selanjutnya dapat dilihat pada gambar 3.4 merupakan rangkaian penyusun IC NE555 dengan Arduino Uno. Pin *output* 5v pada arduino terhubung ke pin 4 (*Reset*) IC NE555 yang di paralelkan dengan pin 8 (*VCC*) dan pin 7 (*Discharge*). Kemudian untuk pin 3 (*Output*) terhubung ke pin D7 pada arduino. Pin D7 pada arduino uno menandakan bahwa data yang dikirimkan melalui pin 8 dalam bentuk digital. Pin GND (Ground) Arduino UNO terhubung ke IC NE555 melalui pin 1 (GND) yang terhubung juga secara paralel dengan pin 2 (*Trigger*) dan pin 6 (*Threshold*). Pada rangkaian tersebut juga menggunakan Resistor 10K (R1) serta kapasitor 104 (C1 dan C2).

### 3.3.3 Perancangan Rangkaian Arduino dengan *Buzzer*

Skema rangkaian Arduino dengan buzzer dapat dilihat pada Gambar 3.5 berikut ini. Diagram ini menguraikan susunan komponen dan koneksinya. Untuk mendapatkan pemahaman yang komprehensif tentang sirkuit, serangkaian langkah terperinci akan dijelaskan dalam penelitian ini.





**Gambar 3.5 Skema Perancangan Rangkaian Arduino dengan *Buzzer***

Dimana pada rangkaian yang ditampilkan gambar 3.5 tersebut *part* yang digunakan yaitu terdiri dari 1 Arduino Uno R3, 1 *piezo buzzer* dan 1 buah resistor serta kabel jumper untuk menghubungkan antar perangkat. Dari rangkaian tersebut dapat dilihat salah satu kaki *piezo buzzer* terhubung ke *ground* sedangkan untuk kaki yang lainnya terhubung ke pin D9 Arduino Uno. Sebelum terhubung ke pin D9 Arduino, terlebih dahulu melewati resistor 1K yang berfungsi untuk mengtur besarnya arus yang digunakan. Karena pada rangkaian tersebut untuk menghubungkan Arduino UNO dengan *buzzer* melalui pin D9, maka data yang dihasilkan adalah dalam bentuk data digital.

### 3.3.4 *Arduino Sleep Mode*

*Arduino sleep mode* merupakan sistem pada mikrokontroler khususnya pada Atmega328P yang memungkinkan pengguna untuk menghentikan atau mematikan modul yang tidak digunakan dengan tujuan untuk mengurangi konsumsi daya. Ada beberapa mode pilihan pada fitur *Arduino sleep mode* yang akan dijelaskan sebagai berikut.

a. *Idle Mode*

Untuk mengaktifkan *Idle Sleep Mode*, tuliskan bit SM [2,0] dari kontroler “000”. Mode ini akan menghentikan CPU tetapi mengizinkan SPI, antarmuka serial 2 kabel, USART, *Watchdog* (sistem pengawas), *counters*, *analog comparator* untuk pengoprasian. Pada dasarnya pada *Idle Mode* akan menghentikan  $CLK_{CPU}$  dan  $CLK_{FLASH}$ . Ardunio dapat aktif

kembali kapan saja dengan menggunakan interupsi secara eksternal atau internal.

**Code untuk *Idle Sleep Mode*:**

```
LowPower.idle(SLEEP_8S, ADC_OFF, TIMER2_OFF,  
TIMER1_OFF, TIMER0_OFF, SPI_OFF, USART0_OFF,  
TWI_OFF);
```

Dengan menggunakan kode diatas arduino akan tidur selama 8 detik dan bangun kembali secara otomatis. Dapat dilihat pada kode diatas bahwa pada *Idle Mode* akan mematikan semua pengatur waktu, SPI, USART dan TWI (antarmuka serial 2 kabel).

b. *ADC Noise Reduction Mode*

Untuk mengaktifkan mode ini, tuliskan bit SM [2,0] ke “001”. Pada mode ini akan menghentikan CPU tetapi mengizinkan ADC, interupsi eksternal, USART, antarmuka serial 2 kabel, *Watchdog* (sistem pengawas) dan *counters* untuk sistem pengoperasian. Pada dasarnya mode ini akan menghentikan CLK<sub>CPU</sub>, CLK<sub>I/O</sub> dan CLK<sub>FLASH</sub>. Untuk membangunkan pada mode ini dapat menggunakan metode *External Reset*, Interupsi perubahan pin, dan *Timer/Counter Interrupt*.

c. *Power-Down Mode*

Pada mode ini menghentikan semua klok yang dihasilkan serta hanya mengizinkan pengoperasian modul asinkron. Mode ini dapat diaktifkan dengan menulis bit SM [2,0] ke “010”. Pada mode ini osilator eksternal akan mati tetapi antarmuka serial 2 kabel, *Watchdog* (sistem pengawas), dan interupsi eksternal akan terus beroperasi. Untuk membangunkan pada mode ini dapat menggunakan metode *External Reset*, Interupsi perubahan pin, dan *External level interrupt on INT*.

**Code untuk *Power-Down Periodic Mode*:**

```
LowPower.powerDown(SLEEP_8S, ADC_OFF, BOD_OFF);
```

Dengan menggunakan kode diatas arduino akan tidur selama 8 detik dan bangun kembali secara otomatis.

### Code untuk *Power-Down Interrupt Mode*

```
void loop()
{
    // Izinkan pin bangun untuk memicu interupsi
    rendah (low).
    attachInterrupt 0, wakeUp, LOW);
    LowPower.powerDown(SLEEP_FOREVER, ADC_OFF,
    BOD_OFF);
    // Nonaktifkan interupsi pin eksternal pada
    pin pengaktifan.
    detachInterrupt(0);
    // Do something here
}
```

#### d. *Power-Save Mode*

Mode ini dapat diaktifkan dengan menulis bit SM [2,0] ke “011”. Mode ini hampir sama dengan *Power-Down Mode*, dengan satu pengecualian yaitu jika *timer/counter* diaktifkan, maka akan tetap keadaan berjalan pada saat *sleep* diaktifkan. Perangkat dapat dihubungkan kembali dengan menggunakan *timer overflow*. Jika tidak menggunakan *timer/counter*, disarankan untuk menggunakan *power-down* untuk mode hemat daya.

#### e. *Standby Mode*

*Mode Standby* ini identik dengan *Power-Down*, satu satunya perbedaan diantar keduanya adalah osilator eksternal akan tetap berjalan pada mode ini. Mode ini dapat diaktifkan dengan menulis bit SM [2,0] ke “110”.

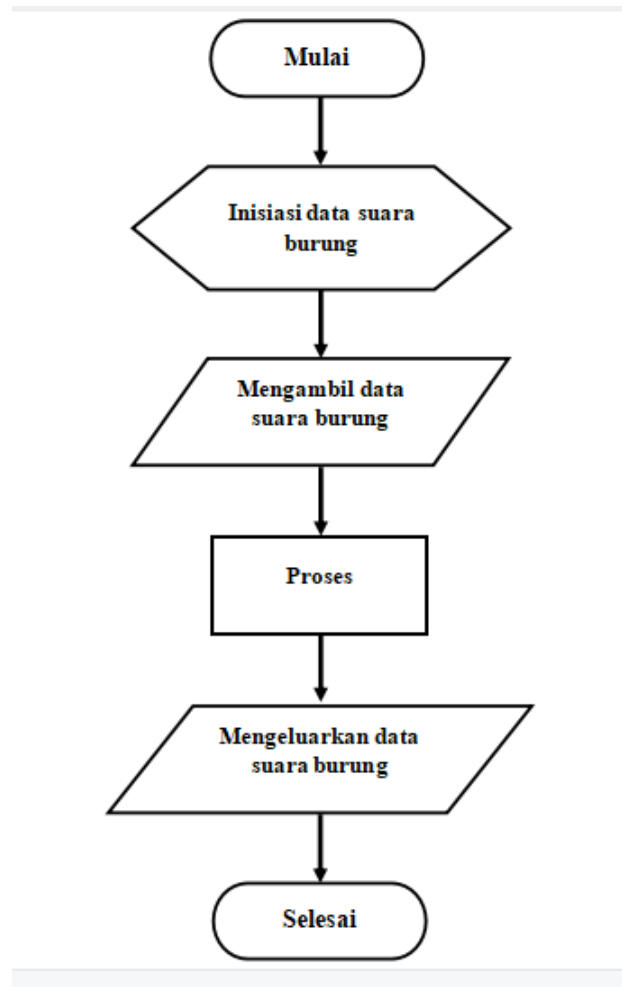
#### f. *Extended Standby Mode*

Mode ini mirip dengan *Power-Save Mode* dengan pengecualian osilator akan tetap berjalan. Perangkat akan memasuki mode ini apabila menuliskan bit SM [2,0] ke “111”. Perangkat akan menggunakan enam klok untuk memperpanjang *Standby Mode*.

### 3.4 PERANCANGAN SOFTWARE

#### 3.4.1 *Flowchart* Sistem

Pada tahap perancangan *software* ini ada beberapa tahapan atau proses yang dilewati untuk mencapai hasil penelitian yang diinginkan. Pada gambar 3.6 merupakan *flowchart* yang digunakan dalam proses penyusunan *software*.



**Gambar 3.6** *Flowchart* Perancangan Sistem

Sesuai dengan gambar 3.6 proses yang pertama adalah mulai. Selanjutnya proses yang kedua yaitu inisiasi data suara burung yang meliputi dari beberapa suara burung yang ditakuti oleh burung pipit sesuai dengan frekuensi yang telah ditentukan. Ketiga adalah mengambil data suara burung dimana dari hasil inisiasi data suara burung yang diperoleh maka akan ditentukan suara burung seperti apa yang akan digunakan. Sebagai contoh pada tahap tersebut suara burung yang digunakan dapat berupa hasil rekaman suara burung predator yang ditakuti oleh

hama burung atau menggunakan suara buatan sendiri sesuai dengan frekuensi yang telah ditentukan. Tahap selanjutnya adalah proses, pada tahap ini data suara burung yang ada akan diproses melalui arduino. Selanjutnya setelah melewati arduino data akan di olah melalui IC NE555 sebagai sistem yang mengatur frekuensi yang dikeluarkan kemudian suaranya akan dikeluarkan melalui *buzzer*.

### **3.5 PENGUJIAN SISTEM**

#### **3.5.1 Pengujian Arduino dengan IC NE555**

Pada tahapan pengujian ini akan dilakukan pengujian terhadap pin pada IC NE555. IC NE555 sendiri memiliki 8 pin yang dapat terhubung. Sekema pengujian yang akan dilakukan adalah ingin mengetahui bagaimana fungsi masing-masing pin yang ada di IC NE555. Selanjutnya pada pengujian ini juga akan diketahui pada pin berapa IC NE555 dapat terhubung dengan sensor keluaran / *output* yang digunakan khususnya pada pengujian ini yang memanfaatkan *buzzer*.

#### **3.5.2 Pengujian Arduino dengan *Buzzer***

Pengujian ini dilakukan dengan cara membuat suara sesuai dengan frekuensi yang telah ditentukan. Frekuensi yang digunakan pada pengujian ini menggunakan rentan frekuensi yang dapat didengar oleh burung pipit dengan rentan 20 KHz - 25 KHz. Kemudian mengamati bagaimana kondisi *buzzer* saat suara dihasilkan apakah dalam keadaan hidup atau mati.

#### **3.5.3 Pengujian Arduino *Sleep Mode***

Pengujian ini dilaksanakan dengan tujuan untuk mengetahui tingkat efisiensi daya yang digunakan oleh sistem dengan memanfaatkan fitur Arduino *Sleep mode*. Pada pengujian ini akan dilakukan dua skema pengujian. Pengujian yang pertama yaitu Sistem tanpa menggunakan Arduino *Sleep Mode*. Kemudian untuk pengujian yang kedua Sistem menggunakan Arduino *Sleep Mode*. Pengujian dilaksanakan dengan menyesuaikan waktu burung pipit aktif yaitu pada pagi hari pukul 07.00-11.00 dan sore hari pada pukul 14.00-16.00. Selanjutnya untuk

mengetahui konsumsi tegangan yang digunakan maka akan dilakukan pengukuran tegangan sebelum alat digunakan dan setelah alat digunakan.

#### **3.5.4 Pengujian Keseluruhan Sistem**

Pengujian ini dilakukan untuk mengujian keseluruhan sistem yang sudah dirangkai. Untuk mengetahui tingkat efektifitas alat yang sudah dibuat, akan diamati kondisi burung dengan jarak 1-4 meter. Dari pengujian ini akan diketahui bagaimana kondisi burung (diam, bergerak, dan pergi) serta daya keseluruhan yang digunakan dengan memanfaatkan frekuensi yang berbeda. Pengujian keseluruhan sistem akan dilaksanakan dengan kondisi yang berbeda yaitu ketika sistem menggunakan arduino *sleep mode* dan ketika sistem tidak menggunakan arduino *sleep mode*.