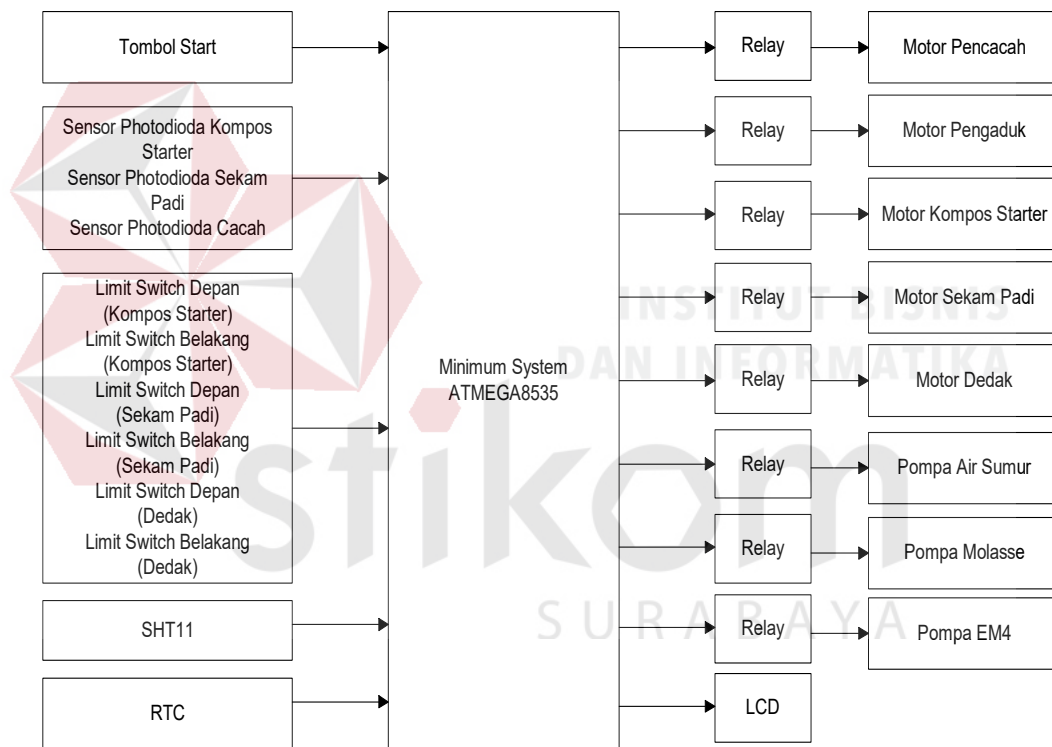


BAB III

METODE PENELITIAN

3.1 Pemodelan Sistem

Dalam pemodelan sistem, secara garis besar dapat digambarkan pada Gambar 3.1 berikut:



Gambar 3.1. Blok Diagram Keseluruhan Sistem

Dalam sistem secara keseluruhan yang nampak pada gambar 3.1 diatas terlihat bahwa sistem ini mempunyai 5 bagian input, antara lain tombol *Start*, sensor *Photodiode*, *Limit Switch*, SHT, dan RTC. Dan 9 pada bagian output, antara lain motor pencacah, motor pengaduk, motor kompos *starter*, motor sekam padi, motor

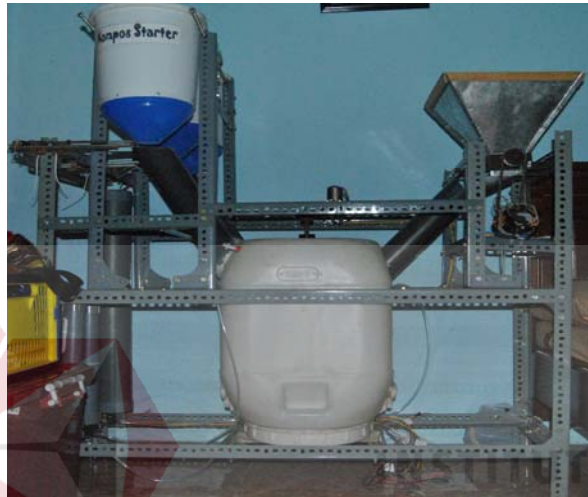
dedak, pompa air sumur, pompa *molasse*, pompa EM4, dan LCD. Dalam sistem ini tombol Start berfungsi sebagai masukan pertama untuk menghidupkan sistem secara keseluruhan, lalu sensor *Phodioda* berfungsi untuk mendeteksi apakah terdapat sampah atau bahan yang ada pada tempat penampungan lalu akan mengirimkan hasilnya ke mikrokontroler, begitu juga dengan *limit switch* yang berfungsi untuk membatasi gerak motor katup pada tempat bahan dimana data dari *limit switch* tersebut juga langsung dikirim ke mikrokontroler, sedangkan untuk SHT mikrokontroler akan mengirim beberapa perintah ke pada sensor SHT11, untuk membaca kelembapan sampah. Kemudian sensor SHT11 berfungsi sebagai *inputan*, dimana sensor ini akan mendeteksi atau memantau tingkat kelembaban sampah. Sedangkan RTC merupakan *integrated circuit* (IC) yang digunakan untuk mencatat waktu secara presisi dari detik hingga tahun. Hasil *input* dari sensor SHT11 dan RTC dieksekusi oleh mikrokontroler. Semua masukan yang diterima oleh mikrokontroler nantinya akan digunakan untuk menggerakkan motor dan pompa sesuai dengan kondisi inputan yang diterima dan akan langsung menampilkan hasilnya ke LCD.

3.2 Perancangan Mekanik

Perangkat keras merupakan bentuk Mesin pembuat pupuk kompos dan rangkaian elektronik yang digunakan, terdiri dari rangkaian regulator, rangkaian *microcontroller*, rangkaian *relay*, rangkaian RTC, rangkaian sensor *photodiode*.

3.2.1 Mesin Pembuat Pupuk Kompos

Mesin pembuat pupuk kompos ini memiliki dimensi panjang x lebar x tinggi berturut-turut yaitu 118 cm × 76 cm × 132 cm. Desain mekanik secara keseluruhan ditunjukkan pada Gambar 3.2.



Gambar 3.2. Desain Mekanik Secara Keseluruhan

Wadah yang digunakan untuk menampung sampah dan bahan berupa tong dengan dimensi panjang x lebar x tinggi berturut-turut yaitu 38 cm x 32 cm x 59 cm, sedangkan untuk pengaduknya terbuat dari besi yang didesain sedemikian rupa agar mampu mengaduk sampah dan bahan lainnya secara merata dan digerakan menggunakan motor DC 12V. Bentuk tong dapat dilihat pada Gambar 3.3 sedangkan pengaduk dapat dilihat pada Gambar 3.4.



Gambar 3.3. Bentuk Tong



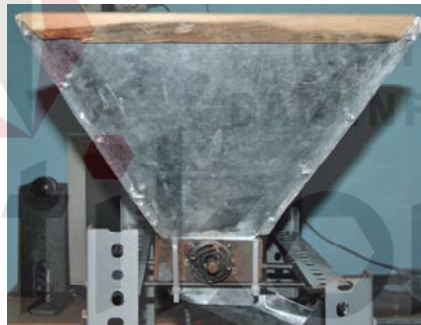
Gambar 3.4. Pengaduk

Untuk pemasangan sensor SHT11 akan terpasang di bagian dalam tong, sehingga akan memudahkan untuk pengaturan kelembapan sampah yang ada dalam tong. Peletakan sensor SHT11 dapat dilihat pada Gambar 3.5.



Gambar 3.5. Peletakan sensor SHT11

Tempat pencacah terbuat dari plat besi tipis atau seng dengan dimensi panjang x lebar x tinggi berturut-turut 30 cm x 30 cm x 24 cm dan penempatan sensor akan di taruh dibagian samping dari tempat pencacah. Tempat pencacah dan peletakan sensor dapat dilihat pada Gambar 3.6 dan Gambar 3.7.



Gambar 3.6. Tempat Pencacah



Gambar 3.7. Peletakan sensor pencacah

Wadah yang digunakan untuk bahan-bahan berasal dari toples plastik sebanyak 3 buah yang nantinya akan digunakan sebagai tempat kompos starter, sekam padi, dan dedak, dan untuk peletakan sensornya terletak dibagian bawah samping tepatnya dibagian corong dari wadah penampung. Tempat Bahan dan peletakan sensor bahan dapat dilihat pada Gambar 3.8 dan Gambar 3.9.



Gambar 3.8. Tempat Bahan



Gambar 3.9. Peletakan sensor pada bagian bahan

Sedangkan untuk katupnya terbuat dari besi yang dapat digeser lalu dipasang besi ulir yang akan terhubung langsung pada motor DC 12V yang berfungsi sebagai penggerak katup. Katup dapat dilihat pada Gambar 3.10.



Gambar 3.10. Katup

Tempat penampung cairan antara lain cairan EM4, *molasse*, dan air sumur terbuat dari pipa paralon dengan panjang masing-masing:

Molas4 = paralon 1,5", tinggi 37 cm

Air Sumur = paralon 3", tinggi 72 cm

Tempat penampung cairan dapat dilihat pada Gambar 3.11.



Gambar 3.11. Tempat Penampung Cairan

Mesin pencacah terbuat dari plat besi yang didesain menyerupai pisau sebagai pemotong dan ditempatkan didasar tempat pencacah, dimana pada kedua sisinya dipasang pisau *cutter* dan plat besi tipis atau seng yang dibentuk zigzag lalu penggeraknya menggunakan motor AC. Mesin pencacah dapat dilihat pada Gambar 3.12 sedangkan pisau pencacah dapat dilihat pada Gambar 3.13.



Gambar 3.12. Mesin Pencacah

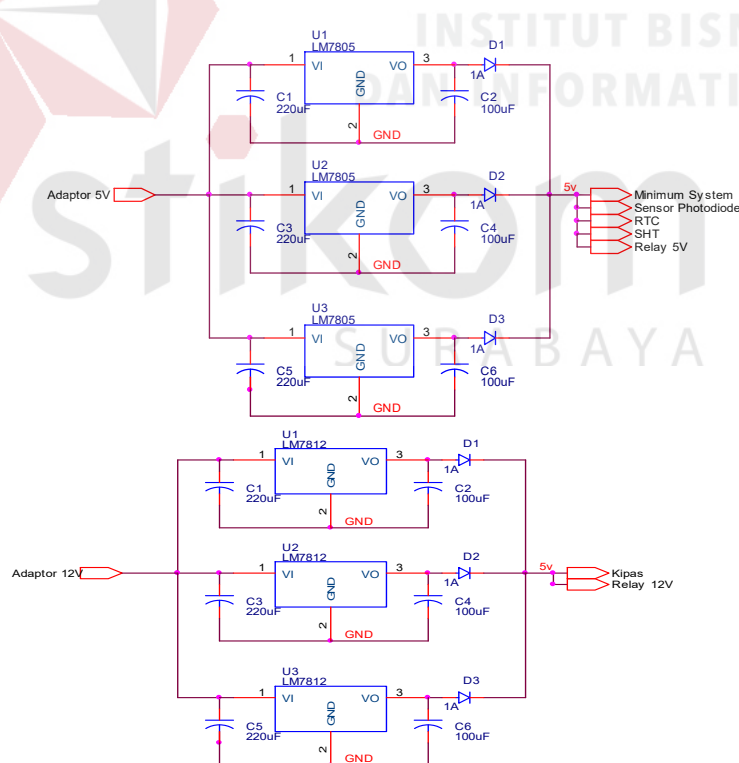


Gambar 3.13. Pisau Pencacah

3.3 Perancangan Perangkat Keras Elektronika

3.3.1 Rangkaian Regulator

Catu daya merupakan pendukung utama bekerjanya suatu sistem. Catu daya yang biasa digunakan untuk menyuplai tegangan sebesar 5 Volt adalah catu daya DC yang memiliki keluaran +5 volt. Catu daya ini digunakan untuk mensuplay tegangan sebesar 5 volt. IC 7805 (IC regulator) digunakan untuk menstabilkan tegangan searah. Kapasitor digunakan untuk mengurangi tegangan kejut saat pertama kali saklar catu daya dihidupkan. Sehingga keluaran IC regulator 7805 stabil sebesar 5 volt DC, sedangkan untuk menyuplai tegangan 12V menggunakan IC regulator 7812. Rangkaian regulator terlihat pada Gambar 3.14. berikut:



Gambar 3.14. Rangkaian regulator

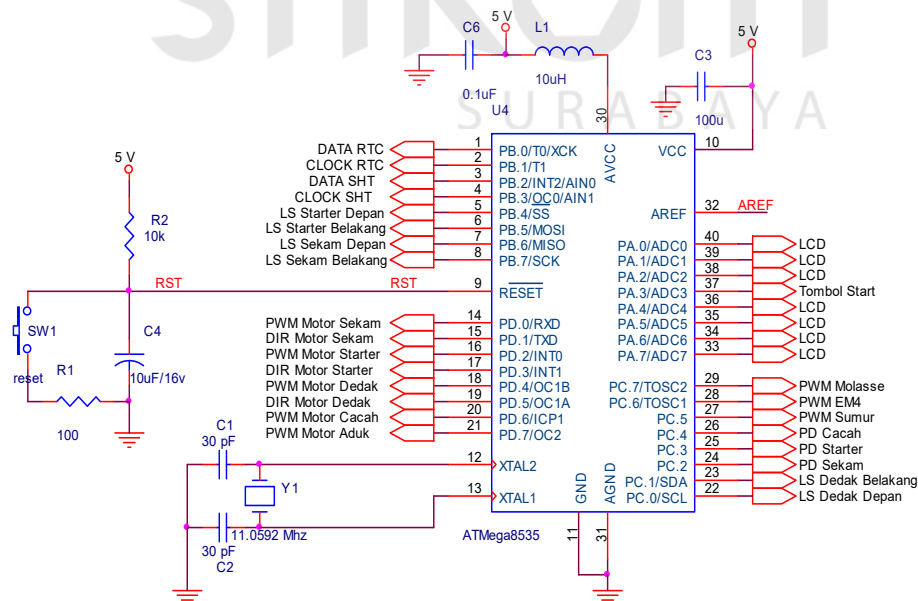
3.3.2 Rangkaian Mikrokontroler

Pada proyek akhir ini dibuat piranti pengendali menggunakan mikrokontroler keluaran AVR, yaitu ATmega8535. Untuk mengaktifkan atau menjalankan mikrokontroler ini diperlukan rangkaian *minimum system*. Rangkaian *minimum system* tersebut terdiri rangkaian *reset*, rangkaian *oscillator*.

A. Rangkaian Minimum System

Untuk menjalankan mikrokontroler dibutuhkan sebuah rangkaian minimum agar mikrokontroler tersebut dapat bekerja dengan baik. Rangkaian minimum mikrokontroler terdiri dari rangkaian *reset* dan rangkaian *oscillator*.

Pada pin VCC diberi masukan tegangan operasi berkisar antara 4,5 Volt sampai dengan 5,5 Volt. Pin RST mendapat *input* dari *manual reset*. Rangkaian *minimum system* dapat dilihat pada Gambar 3.15 berikut:



Gambar 3.15. Rangkaian *Minimum System* ATmega8535

Berdasarkan Gambar 3.15, *pin Vcc* diberi tegangan operasi yang besarnya berkisar 5 volt sampai dengan 6,5 volt. Dalam Gambar 3.15. di atas, *pin XTAL1* dan *XTAL2* dihubungkan dengan komponen XTAL sebesar 11,5092 MHz. Pemilihan *frekuensi osilasi* dari XTAL tersebut berdasarkan penggunaan mikrokontroler agar setiap *clock* mikrokontroler berlangsung setiap 1,042 μ s. Berikut adalah perhitungan besar *clock cycle* yang dipakai :

$$\text{Clock cycle} = \frac{12}{\text{Frekuensi Kristal}}$$

Frekuensi Kristal

$$= \frac{12}{11,5092 \times 10^6}$$

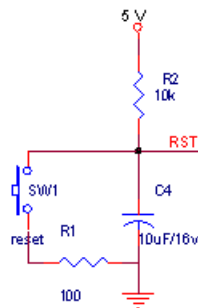
$$= 1,042 \times 10^{-6}$$

$$= 1,042 \times 10^{-6}$$

$$\text{Clock cycle} = 1,042 \mu\text{s}$$

B. Rangkaian Reset

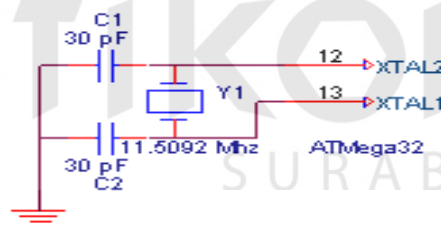
Reset pada mikrokontroler ATmega8535 terjadi dengan adanya logika *high* “1” selama dua *cycle* pada kaki RST pada mikrokontroler ATmega8535. Setelah kondisi pin RST kembali *low*, maka mikrokontroler akan menjalankan program dari alamat 0000H. Dalam hal ini *reset* yang digunakan adalah *manual reset*. Rangkaian *reset* dapat dilihat pada Gambar 3.16 berikut:



Gambar 3.16. Rangkaian Reset

C. Rangkaian Oscillator

Pin XTAL1 dan XTAL2 merupakan pin *oscillator* bagi mikrokontroler ATmega8535. Pin XTAL1 berfungsi sebagai *input* dan XTAL2 sebagai *output oscillator*. *Oscillator* ini bisa berasal dari kristal. Rangkaian *oscillator* dapat dilihat pada Gambar 3.17 berikut:



Gambar 3.17. Rangkaian Oscillator

D. Interface I/O

Rangkaian *I/O* dari mikrokontroler mempunyai kontrol direksi yang tiap bitnya dapat dikonfigurasi secara *individual*, maka dalam perancangan *I/O* yang digunakan ada yang berupa operasi *port* ada pula yang dikonfigurasi tiap bit *I/O*.

Berikut ini akan diberikan konfigurasi dari *I/O* mikrokontroler tiap bit yang ada pada masing-masing *port* yang terdapat pada mikrokontroler.

Mikrokontroler

1. Port A

Port A digunakan untuk LCD dan tombol Start.

2. Port B

Port B digunakan untuk RTC, SHT, dan *Limir Switch*.

3. Port C

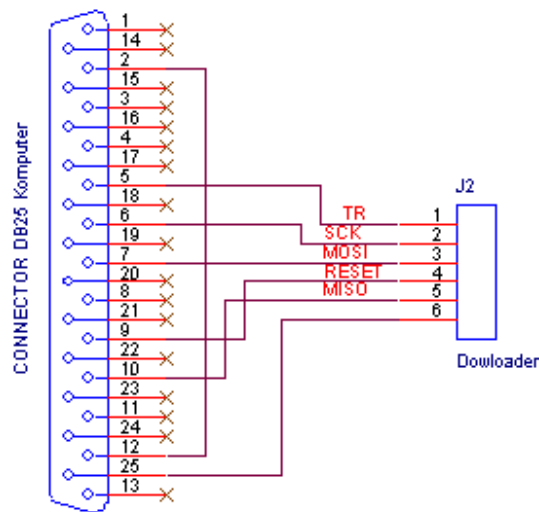
Port C digunakan untuk *Limit Switch*, sensor *Photodiode*, *Relay*.

4. Port D

Port D digunakan untuk *Relay*.

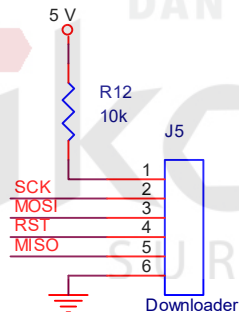
E. Program Downloader

Untuk melakukan proses *downloading* program dalam format *.HEX* dari komputer ke dalam *memory* program *internal* mikrokontroler, penulis menggunakan kabel *downloader* dengan *interface* DB25 yang dihubungkan pada *port* LPT1 pada komputer. Konfigurasi kabel *downloader* terdapat pada Gambar 3.18 berikut:



Gambar 3.18. Rangkaian kabel *downloader* pada port LPT

Sedangkan untuk konektor *downloader* pada mikrokontroler ATmega8535 dapat dilihat pada Gambar 3.19 berikut:

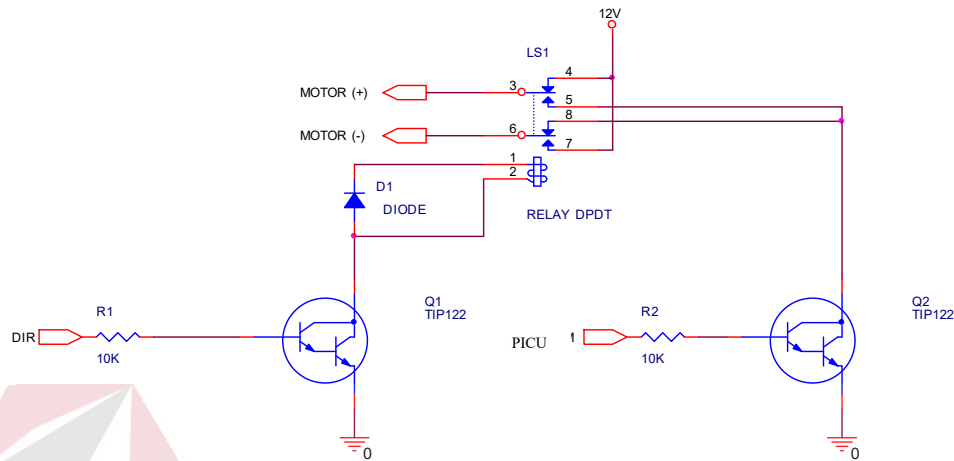


Gambar 3.19. Konektor *Downloader* pada Mikrokontroler ATmega8535

3.3.3 Rangkaian *Relay*

Relay digunakan sebagai *switch* tegangan dan arus yang lebih tinggi dari mikrokontroler. Salah satu cara mengontrol motor adalah dengan menggunakan *relay*. motor dihubungkan dengan sumber daya melalui saklar yang ada pada *relay*, dan

relay tersebut dihubungkan dengan *port* keluaran suatu kontroler melalui sebuah *transistor*. Rangkaian *relay* dapat dilihat pada Gambar 3.20.



Gambar 3.20. Rangkaian *Relay*.

Dari gambar 3.20 terlihat bahwa jika masukan berlogika 1, maka *relay* akan aktif sehingga saklar yang ada pada *relay* akan menutup dan menyebabkan motor hidup. Bila masukan berlogika 0, maka *transistor* akan mati dan *relay* juga mati sehingga saklar pada *relay* akan membuka dan menyebabkan motor mati.

3.3.4 Rangkaian RTC

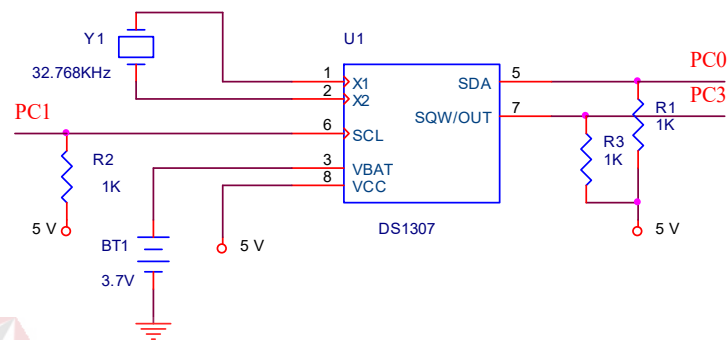
RTC ini merupakan RTC yang menggunakan teknologi *2-wire bus* sehingga memungkinkan untuk berbagi jalur dengan komponen lain yang juga mendukung teknologi tersebut. Dengan menggunakan 2 jalur data, alokasi pin untuk berkomunikasi dengan mikrokontroler menjadi lebih hemat karena untuk mengakses data RTC hanya menggunakan 2 pin yang masing-masing untuk jalur data dan *clock*.

Baterai yang digunakan adalah baterai kancing 3V dan *kristal quartz* 32.768kHz sebagai *power supply* cadangan untuk operasional osilator. Sedang pada

pin 8 dan 6 dihubungkan dengan resistor 1K dimana resistor tersebut sebagai *pull-up*.

Hal ini untuk menjaga agar saat jalur *bus free* tidak *pull-down* (berada di logika *high*).

Rangkaian RTC DS1307 dapat dilihat pada Gambar 3.21 berikut:



Gambar 3.21. Rangkaian RTC

dan berikut potongan program untuk RTC.

```
unsigned char bcd2dec(unsigned char input)
{
    unsigned char tmp_data, tmp1;
    tmp_data = input;
    tmp1 = tmp_data % 16;
    if (tmp_data > 15) tmp_data = tmp_data / 16;
    else tmp_data = 0;
    tmp_data = (tmp_data * 10) + tmp1;
    return tmp_data;
}
```

```
void read_rtc(void)
{
    unsigned char i, tmp_data;
    i2c_start();
    i2c_write(RTC_ADDR);
    i2c_write(0);
    i2c_stop();
    i2c_start();
    i2c_write(RTC_ADDR | 1);
    for (i=0; i<6; i++)
    {
        tmp_data = bcd2dec(i2c_read(1));
        //if (i!=3)
        data_rtc[i]=tmp_data;
    }
    tmp_data = bcd2dec(i2c_read(0));
    data_rtc[6]=tmp_data;
    i2c_stop();
}
```

```

    }

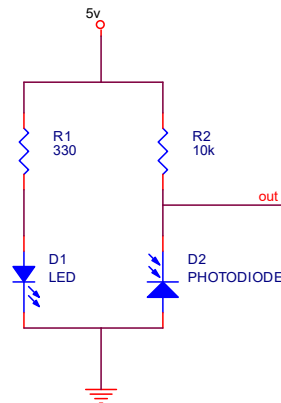
unsigned char dec2bcd(unsigned char input)
{
    unsigned char tmp_data;
    if (input > 9) tmp_data = ((input / 10)*16) + (input % 10);
    else
        tmp_data = input;
    return tmp_data;
}

void write_rtc(unsigned char alamat, unsigned char data)
{
    i2c_start();
    i2c_write(RTC_ADDR);
    i2c_write(alamat);
    if (alamat < 7)
        i2c_write(dec2bcd(data));
    else
        i2c_write(data);
    i2c_stop();
}

```

3.3.5 Rangkaian *Photodiode*

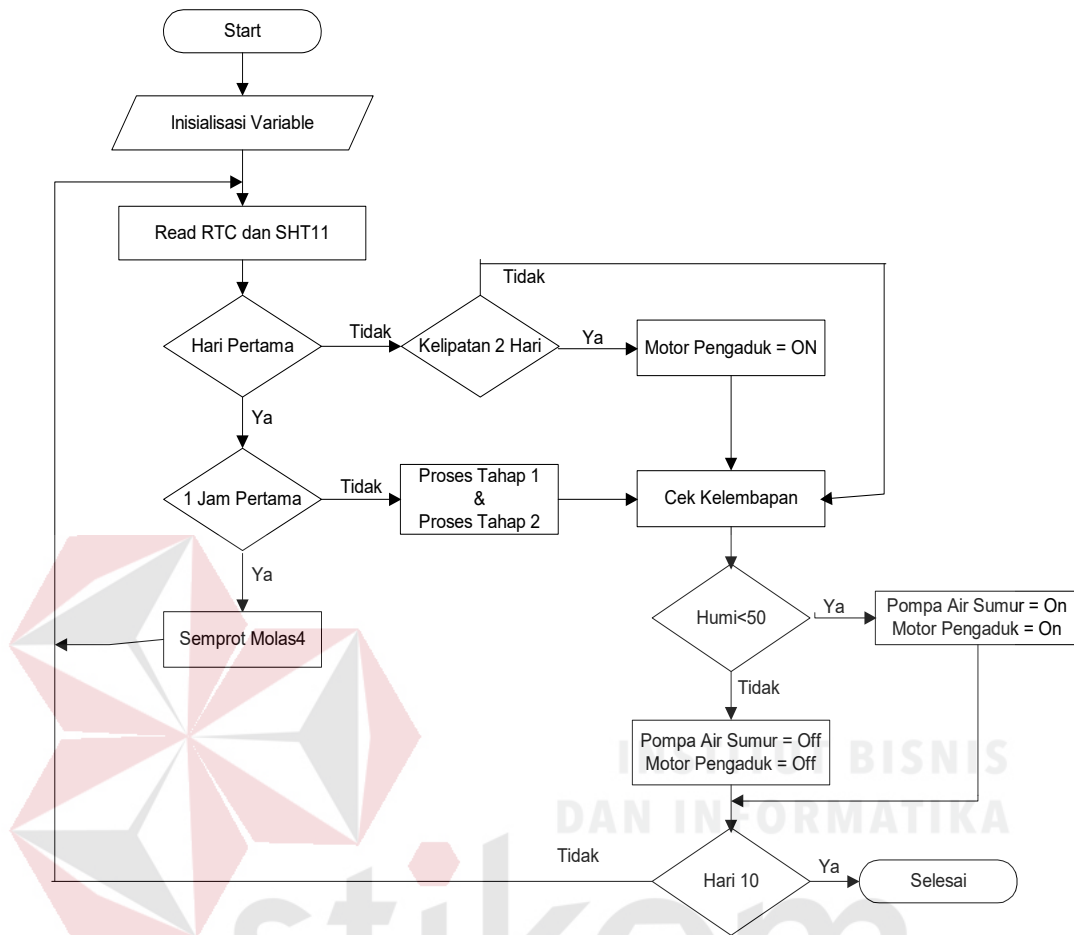
Photodiode digunakan untuk mendeteksi apakah terdapat sampah yang ada pada tempat pencacah dimana hasil dari pembacaan sensor ini akan digunakan untuk menghidupkan motor pencacah, sedangkan pada tempat penampung sensor *Photodiode* digunakan untuk mendeteksi apakah terdapat bahan didalam tempat penampung, jika terdeteksi adanya bahan baku didalam tempat penampung maka sensor ini akan mengirimkan data kemikrokontroler untuk menjalankan motor sekam, motor *starter*. Berikut Gambar rangkaian sensor *Photodiode* tampak pada Gambar 3.22.



Gambar 3.22. Rangkaian *Photodiode*

3.4 Perancangan Perangkat Lunak

Perancangan perangkat lunak bertujuan untuk memperoleh dan menampilkan data kelembaban sampah yang akurat. Perancangan perangkat lunak untuk sistem antara lain program secara keseluruhan dan program rutin baca kelembaban,. Diagram alir perangkat lunak secara umum dapat dilihat pada Gambar 3.23 berikut:

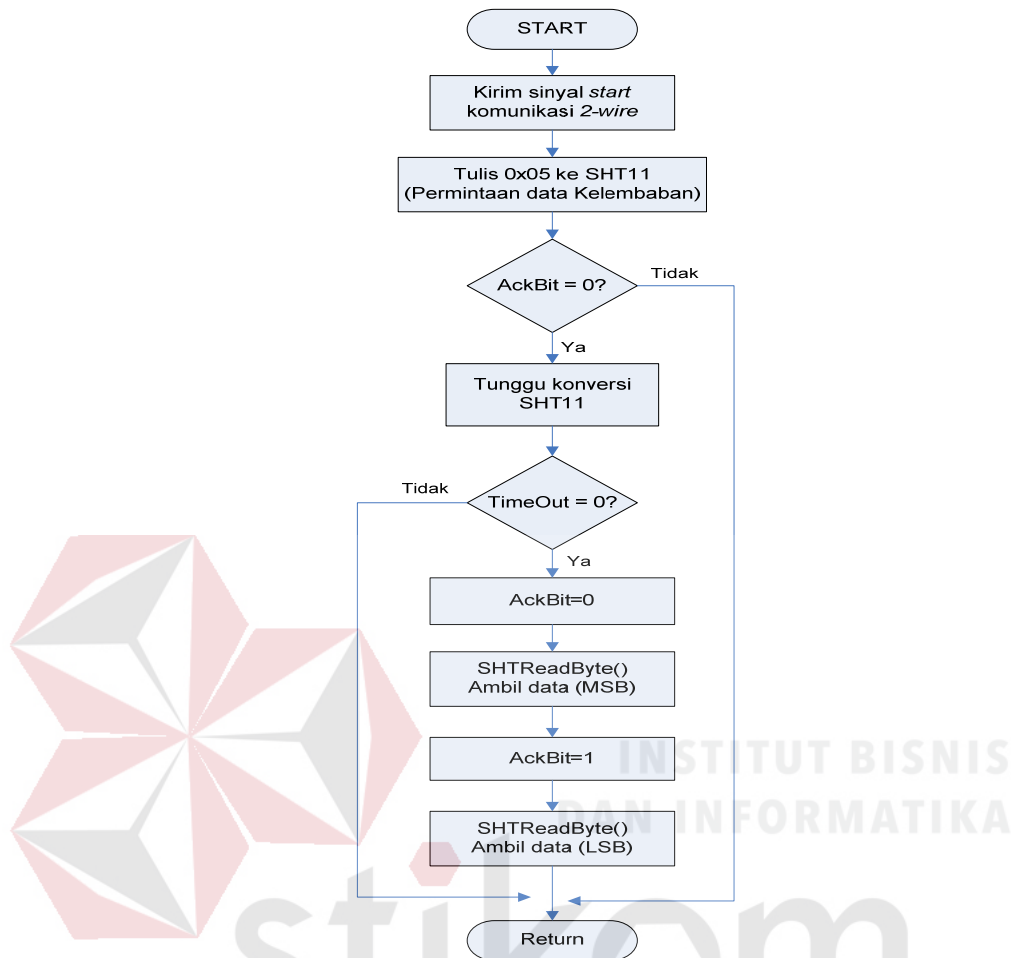


Gambar 3.23. Diagram alir program secara umum

3.4.1 Program Mikrokontroler

a. Program Rutin Baca Kelembaban

Diagram alir untuk pembacaan kelembaban SHT11 terdapat pada Gambar 3.24 berikut :



Gambar 3.24. Diagram alir pembacaan rutin kelembaban

Pada Gambar 3.24 diagram alir pembacaan rutin kelembaban, pembacaan kelembaban dimulai dengan mengirim sinyal *start* untuk memulai komunikasi serial 2-wire. Setelah itu program mengirim 0x05 ke SHT11 yang merupakan perintah untuk memulai pengukuran suhu. Rutin *SHTWriteByte(0x05)* akan memberikan nilai ACK yang disimpan dalam variabel *AckBit*. Jika variabel *AckBit* bernilai 0, maka program akan menunggu selesainya pengukuran SHT11 dengan memanggil rutin *SHTWait()*. Rutin *SHTWait()* akan memberikan suatu nilai yang kemudian disimpan pada variabel *TimeOut*. Variabel *TimeOut* akan bernilai 0 jika pengukuran SHT11

selesai dan data siap. Setelah pengukuran selesai, data suhu akan dibaca LSB dulu kemudian MSB. Pembacaan data LSB dilakukan dengan memberi nilai variabel AckBit = 0, sedangkan pembacaan data MSB dilakukan dengan memberi nilai variabel AckBit = 1. Berikut potongan program pembacaan rutin kelembaban:

```
// Atur delay untuk baca kelembaban
else if (BacaSuhu == 60)
{
    SHTReadHumidity();
}
    else if (BacaSuhu == 80)
    {
        BacaSuhu = 0;
        SHTWait();           // Tunggu sampai pengukuran selesai
        if (TimeOut==0)
        {
            AckBit=0;        // Kirim ACK untuk menerima byte berikutnya
            SHTReadByte();    // Ambil Byte MSB
            DataRHSHT = DataRead;
            DataRHSHT <<= 8;
            AckBit=1;        // Kirim NACK untuk mengakhiri pengambilan data
            SHTReadByte();    // Ambil byte LSB
            DataRHSHT |= DataRead;

            // Olah Data Humidity dari SHT
            DataRead = DataRHSHT * 0.0405;
            DataRHSHT = DataRHSHT / 1000;
            DataRHSHT = DataRHSHT * DataRHSHT * 2.8;
            DataRHSHT = DataRead - DataRHSHT - 4;

            ResetSHT();
        }
    }
}
```