

**RANCANG BANGUN ALAT PENCAMPUR CAT TEMBOK
OTOMATIS BERBASIS *PERSONAL COMPUTER* (PC)
(BAGIAN II)**

TUGAS AKHIR



Oleh:

DITA AYU INDAH PRAMESWARY

NIM. 081310213038

PROGRAM STUDI D3 OTOMASI SISTEM INSTRUMENTASI

DEPARTEMEN TEKNIK

FAKULTAS VOKASI

UNIVERSITAS AIRLANGGA

SURABAYA

2016

LEMBAR PERSETUJUAN TUGAS AKHIR

RANCANG BANGUN ALAT PENCAMPUR CAT TEMBOK
OTOMATIS BERBASIS *PERSONAL COMPUTER* (PC)

BAGIAN II

TUGAS AKHIR

Sebagai Salah Satu Syarat Untuk Memperoleh Gelar Ahli Madya (A.Md)

Bidang Otomasi Sistem Instrumentasi

Pada Departemen Teknik Fakultas Vokasi

Universitas Airlangga

Oleh :

DITA AYU INDAH PRAMESWARY

NIM. 081310213038

Disetujui Oleh,

Dosen Pembimbing

Dosen Konsultan


Winarno, S.Si., M.T.


Denv Arifianto, S.Si.

NIP. 198109122015041001

NIK. 139111263

LEMBAR PENGESAHAN NASKAH TUGAS AKHIR

JUDUL : RANCANG BANGUN ALAT PENCAMPUR CAT
TEMBOK OTOMATIS BERBASIS *PERSONAL*
COMPUTER (PC) (BAGIAN II)

PENYUSUN : Dita Ayu Indah Prameswary

NIM : 081310213038

PEMBIMBING : Winarno, S.Si., M.T.

KONSULTAN : Deny Arifianto, S.Si

TANGGAL UJIAN : 26 Juli 2016

Disetujui Oleh:

Dosen Pembimbing

Dosen Konsultan

Winarno, S.Si., M.T.

Deny Arifianto, S.Si.

NIP. 198109122015041001

NIK. 139111263

Mengetahui:

Ketua Departemen Teknik

Koordinator Program Studi
D3 Otomasi Sistem Instrumentasi

I. Dyah Herawatie, M.Si.

Winarno, S.Si., M.T.

NIP. 196711111993032002

NIP. 198109122015041001

PEDOMAN PENGGUNAAN TUGAS AKHIR

Tugas Akhir ini tidak dipublikasikan, namun tersedia di perpustakaan dalam lingkungan Universitas Airlangga. Diperkenankan untuk dipakai sebagai referensi kepustakaan, tetapi pengutipan seijin penulis dan harus menyebutkan sumbernya sesuai kebinasaan ilmiah.

Dokumen Tugas Akhir ini merupakan hak milik Universitas Airlangga.



Dita Ayu Indah Prameswary, 2016, *Rancang Bangun Alat Pencampur Cat Tembok Otomatis Berbasis Personal Computer (PC) (Bagian II)*. Tugas Akhir ini di bawah bimbingan Winarno, S.Si., M.T. dan Deny Arifianto S.Si. Prodi D3 Otomasi Sistem Instrumentasi Departemen Teknik Fakultas Vokasi Universitas Airlangga.

ABSTRAK

Cat tembok merupakan cat yang bahan pelarutnya dari air. Selain berfungsi untuk memperindah, cat tembok juga berfungsi sebagai pelindung dinding rumah. Kebutuhan akan warna yang beragam terus meningkat. Sedangkan penggunaan warna cat saat ini masih sangat tergantung oleh standar warna yang tersedia di pasaran.

Tugas akhir ini membahas tentang pembuatan rancang bangun alat pencampur cat tembok otomatis berbasis *personal computer (PC)*. *Personal computer* digunakan sebagai tampilan masukan warna dan tempat untuk menuliskan program sistem. Pengendali utama pada alat ini menggunakan mikrokontroler ATmega2560 pada board arduino mega yang diprogram dengan menggunakan aplikasi Arduino IDE dan menggunakan Delphi 7 sebagai *interface* pemilihan warna. Pada delphi 7 terdapat pilihan komposisi warna dasar cat dan kolom yang digunakan untuk masukan volume total cat yang diinginkan selanjutnya akan dikonversi perhitungan dari volume komposisi warna dasar yang dibutuhkan, kemudian data tersebut dikirim dan diproses oleh arduino. Arduino bekerja membaca data yang telah dikirim dari delphi 7 dan akan bekerja menurunkan aktuator pengaduk dan selang cat hingga mengenai *limit switch* bawah, kemudian mengontrol buka/tutup *valve* sehingga didapatkan volume yang sesuai dengan yang ditentukan. Setelah itu mengontrol lama waktu pengadukan selama 30 detik selanjutnya aktuator pengaduk dan selang cat akan kembali naik hingga mengenai *limit switch* atas dan proses selesai.

Dari hasil penelitian keberhasilan alat dalam melakukan pencampuran cat berdasarkan volume yang diinginkan dengan komposisi cat warna dasar, rancang bangun alat pencampur cat tembok berbasis *personal computer (PC)* ini dapat bekerja secara otomatis dengan hasil rata-rata selisih volume *set point* terhadap volume *output* yaitu 12,4 ml.

Kata Kunci : *Personal Computer (PC)*, Arduino Mega, Arduino IDE, Borland Delphi 7

KATA PENGANTAR

Segala puji syukur penulis panjatkan kehadirat Allah SWT atas rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan Tugas Akhir yang berjudul “Rancang Bangun Alat Pencampur Cat Tembok Otomatis Berbasis *Personal Computer (PC)* Bagian II” .

Tugas Akhir ini dapat selesai dengan baik berkat bantuan dari berbagai pihak. Untuk itu dengan segala kerendahan hati, penulis menyampaikan terima kasih kepada semua pihak yang turut membantu dalam menyelesaikan tugas akhir ini :

1. Bapak Winarno, S.Si., M.T. selaku Koordinator Program Studi D3 Otomasi Sistem Instrumentasi Universitas Airlangga Surabaya.
2. Bapak Winarno, S.Si., M.T. selaku Dosen Pembimbing yang telah banyak memberikan arahan, bimbingan, dan masukan kepada penulis sehingga terselesaikannya Proposal Proyek Akhir ini.
3. Bapak Deny Arifianto, S.Si selaku Konsultan yang banyak memberikan arahan, bimbingan, masukan, beserta ketulusan hati dalam membimbing.
4. Ibu Dr. Riries Rulaningtyas, S.T., M.T selaku Dosen penguji yang telah memberikan banyak masukan maupun saran dalam pembuatan Tugas Akhir ini.
5. Ayah, Ibu dan Udin yang selalu tiada henti untuk mendoakan, memberi semangat, dan juga dukungan hingga penulis mampu menyelesaikan tugas akhir ini.

6. Seluruh Dosen D3 OSI terimakasih atas ilmu dan bimbingannya.
7. Mbak Nadia TU OSI, Pak Jemawan terimakasih atas segala informasi dan doanya.
8. Pratama Bagus Baharsyah selaku partner tugas akhir yang banyak membantu serta terimakasih buat berbagi ilmunya dan berbagi waktunya untuk membuat campuran warna yang sangat indah.
9. Rumpita(Miming, Ifa, Sofi, Dewi, Wanda, Oneng, Aliyah, Media) terimakasih buat semangatnya, doanya, pelukan hangatnya. Andin terimakasih buat tangisannya.
10. ASTRAI (Zu, Albik, Hendrik, Sueb, Adreng, Aldy, Kak Al, Ilham, Acil, Bagir, Dimas) terimakasih buat bantuannya. Terimakasih Affan sebagai tim sukses yudisium OSI'13, Farid, Shela, Rilmen, Oscar, dan semua Teman-Teman OSI 2013 terimakasih atas dukungannya.
11. Choiriah, Feby Auliandini, Anik, Frisvi, Kimia Analisis 2013, Manda Risiwi dan Yovie. Terimakasih semangat jarak jauhnya.
12. Semua pihak yang telah memberikan bantuan kepada penulis.

Akhir kata, penulis sangat mengharapkan saran dan kritik yang membangun demi penyempurnaan tugas akhir ini.

Surabaya, 17 Juli 2016

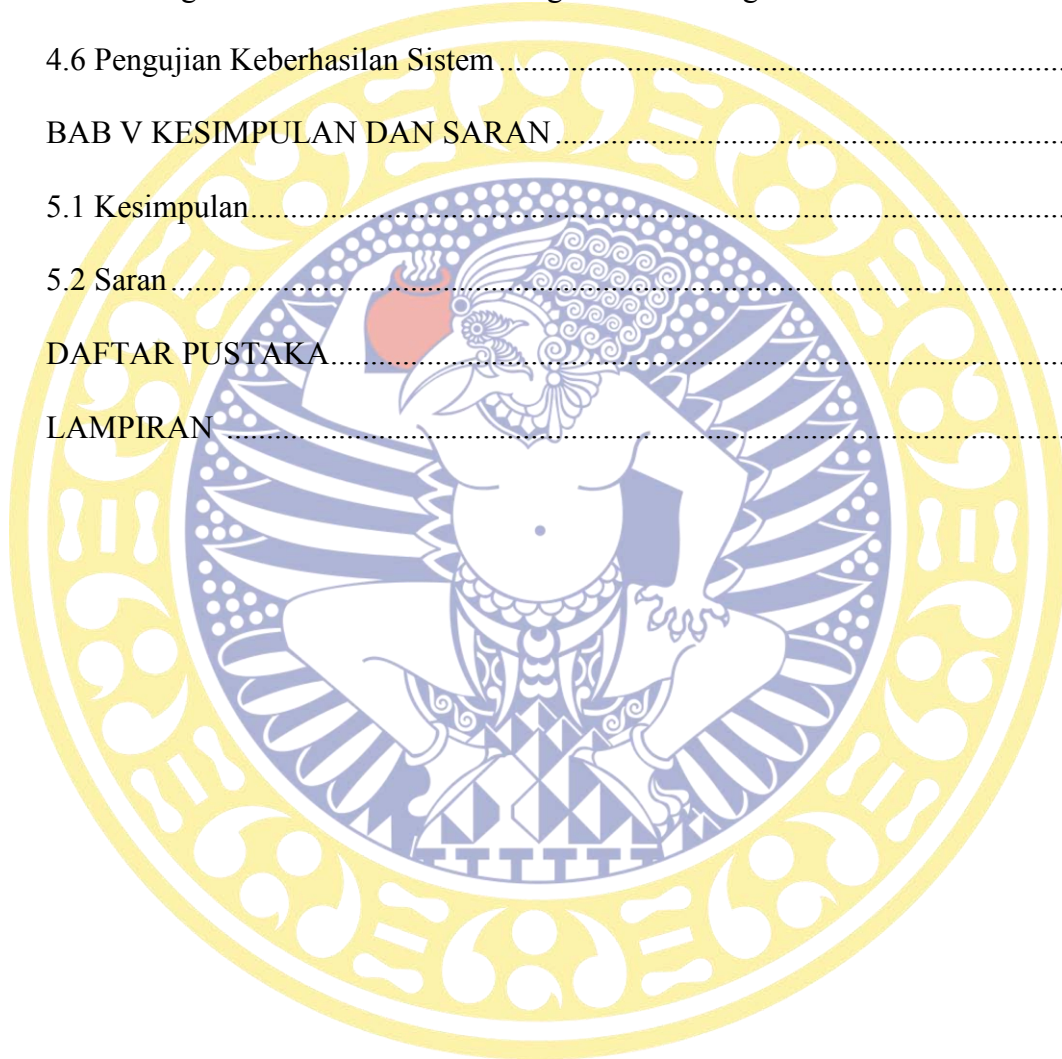
Penulis

DAFTAR ISI

HALAMAN SAMBUTAN	i
LEMBAR PERSETUJUAN.....	ii
LEMBAR PENGESAHAN	iii
PEDOMAN PENGGUNAAN TUGAS AKHIR	iv
ABSTRAK	v
KATA PENGANTAR	vi
DAFTAR ISI.....	viii
DAFTAR TABEL.....	xi
DAFTAR GAMBAR.....	xii
DAFTAR LAMPIRAN.....	xiii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	3
1.3 Batasan Masalah.....	4
1.4 Tujuan.....	4
1.5 Manfaat.....	4
BAB II TINJAUAN PUSTAKA.....	5
2.1 Cat Tembok.....	5
2.2 Pengolahan Citra.....	5
2.3 Representasi Warna.....	6
2.4.6 <i>Software</i>	7

2.4.1 Arduino IDE	7
2.4.2 Borland Delphi.....	8
BAB III METODE PENELITIAN.....	10
3.1 Waktu dan Tempat Pelaksanaan.....	10
3.2 Alat dan Bahan	10
3.2.1 Alat	10
3.2.2 Bahan	11
3.3 Prosedur Penelitian.....	11
3.3.1 Tahap Persiapan.....	13
3.3.2 Tahap Perancangan Alat.....	13
3.3.3 Tahap Perwujudan Alat	14
3.3.4 Tahap Pemrograman <i>Software</i>	15
3.4 Pengujian Sistem	22
3.4.1 Pengujian Sensor <i>Flowmeter</i>	22
3.4.2 Pengujian <i>Software</i>	22
3.5 Analisis Data	23
BAB IV HASIL DAN PEMBAHASAN.....	24
4.1 Hasil Pembuatan Interface Program Pada Delphi	24
4.2 Hasil Pembuatan Perangkat Lunak (<i>Software</i>).....	25
4.2.1 Sub Program Pemilihan Warna Cat Pada Delphi	25
4.2.2 Sub Program <i>Relay</i> Untuk Kendali <i>Valve</i>	26
4.2.3 Sub Program <i>Flowmeter</i>	27
4.2.4 Sub Program <i>Parsing</i> Data	30

4.3 Pengujian Kinerja <i>Relay</i>	32
4.4 Pengujian Sensor <i>Flowmeter</i>	33
4.4.1 Pengujian Awal Respons <i>Flowmeter</i> Terhadap Laju Aliran.....	34
4.4.2 Pengujian Respons <i>Flowmeter</i> menggunakan buka/tutup <i>valve</i>	35
4.5 Hubungan Antara Volume Cat Dengan Waktu Pengadukan.....	40
4.6 Pengujian Keberhasilan Sistem.....	41
BAB V KESIMPULAN DAN SARAN.....	42
5.1 Kesimpulan.....	42
5.2 Saran.....	43
DAFTAR PUSTAKA.....	44
LAMPIRAN.....	45



DAFTAR TABEL

Tabel 3.1 Tabel Pengalamatan Port Pada Arduino	17
Tabel 4.1 Tabel Pengujian <i>Relay</i>	32
Tabel 4.2 Tabel Hubungan Antara Volume Cat Dengan Volume <i>Flowmeter</i> Pada <i>Flowmeter Merah</i>	34
Tabel 4.3 Tabel Hubungan Antara Volume Cat Dengan Volume <i>Flowmeter</i> Pada <i>Flowmeter Hijau</i>	34
Tabel 4.4 Tabel Hubungan Antara Volume Cat Dengan Volume <i>Flowmeter</i> Pada <i>Flowmeter Biru</i>	35
Tabel 4.5 Tabel Hubungan Antara Volume <i>Set point</i> dengan Volume <i>Flowmeter</i> dan Volume Hasil Pada <i>Flowmeter Merah</i>	36
Tabel 4.6 Tabel Hubungan Antara Volume <i>Set point</i> dengan Volume <i>Flowmeter</i> dan Volume Hasil Pada <i>Flowmeter Hijau</i>	37
Tabel 4.7 Tabel Hubungan Antara Volume <i>Set point</i> dengan Volume <i>Flowmeter</i> dan Volume Hasil Pada <i>Flowmeter Biru</i>	38
Tabel 4.8 Tabel Hubungan Antara Volume Cat Dengan Waktu Pengadukan	40
Tabel 4.9 Tabel Keberhasilan Sistem	42

DAFTAR GAMBAR

Gambar 2.1 <i>Void Setup</i> dan <i>Void Loop</i> Arduino IDE.....	7
Gambar 2.2 Tampilan Awal <i>Form</i> Delphi	9
Gambar 2.3 SerialNG.....	9
Gambar 3.1 Diagram Blok Prosedur Penelitian.....	12
Gambar 3.2 Blok Diagram Sistem Kontrol.....	14
Gambar 3.3 Alat Tampak Depan	14
Gambar 3.4 <i>Flowchart</i> Pada Delphi.....	18
Gambar 3.5 <i>Flowchart</i> Pembacaan Nilai <i>Red</i> Pada Arduino.....	19
Gambar 3.6 <i>Flowchart</i> Pembacaan Nilai <i>Green</i> dan <i>Blue</i> Pada Arduino.....	20
Gambar 3.7 <i>Flowchart</i> Proses Pengadukan Pada Arduino	21
Gambar 4.1 Perancangan <i>Interface</i> Pada Delphi	24
Gambar 4.2 Grafik Linieritas <i>Flowmeter</i> Merah	37
Gambar 4.3 Grafik Linieritas <i>Flowmeter</i> Hijau	38
Gambar 4.4 Grafik Linieritas <i>Flowmeter</i> Biru.....	39

DAFTAR LAMPIRAN

Lampiran 1 Hasil Rancang Bangun Alat Pencampur Cat Tembok Berbasis PC

Lampiran 2 Program Pada Delphi

Lampiran 3 Program Pada IDE Arduino



BAB I

PENDAHULUAN

1.1 Latar Belakang

Cat adalah suatu cairan yang digunakan untuk melapisi permukaan suatu bahan dengan tujuan mempercantik (*decorative*), memperkuat (*reinforcing*) serta melindungi (*protective*) bahan tersebut. Jenis cat sangat bervariasi dan dapat disesuaikan dengan kebutuhan. Berdasarkan bahan pelarutnya, cat terbagi menjadi dua jenis utama yaitu cat berbahan dasar air dan cat berbahan dasar minyak. Cat sendiri dapat digunakan pada hampir semua jenis objek, antara lain untuk menghasilkan karya seni (oleh pelukis untuk membuat lukisan), salutan industri (*industrial coating*), marka jalan, pengawet (untuk mencegah korosi atau kerusakan oleh air). [rumahbangun (diakses Desember 2015)]

Penggunaan warna cat saat ini masih sangat tergantung oleh standar warna yang tersedia di pasaran. Kebutuhan akan warna yang beragam terus meningkat, sedangkan ragam warna yang ada di pasaran masih sangat minim dan masih menggunakan standar warna yang sangat mendasar. (Yoseph Evana, 2008)

Selama ini dalam menghasilkan variasi warna yang sesuai dengan keinginan konsumen, hanya dengan melakukan pencampuran dari warna dasar cat menggunakan cara manual. Cara tersebut kurang *efisien* dan merepotkan, karena penjual/tukang cat harus menakar warna cat dasar terlebih dahulu kemudian mengujinya. Dalam membuat warna baru yang sesuai dengan keinginan, dapat dilakukan dengan pencampuran dari beberapa warna sehingga menjadi warna baru

sesuai yang diharapkan. Tetapi seringkali warna yang dihasilkan jauh dari yang diharapkan. Pencampuran cat secara manual memerlukan tenaga ahli yang sudah berpengalaman sehingga tidak semua orang dapat melakukannya. Untuk mendapatkan hasil yang sesuai dengan keinginan diperlukan penakaran komposisi warna yang akurat. Pada pencampuran secara manual, keahlian yang dimiliki oleh seseorang sangat menentukan hasil yang diperoleh dan sering terjadi perbedaan hasil antara pencampuran satu dengan yang lain. Untuk mengatasi hal tersebut maka diperlukan peralatan yang bekerja secara otomatis serta mempunyai kemampuan dalam penakaran yang tepat. Kemajuan ilmu pengetahuan dan teknologi telah mendorong manusia untuk melakukan otomatisasi dan *digitalisasi* pada perangkat-perangkat manual.

Berdasarkan permasalahan tersebut, dibuat inovasi pencampuran cat secara otomatis yaitu “Rancang Bangun Alat Pencampur Cat Tembok Otomatis Berbasis *Personal Computer (PC)*”. Rancang bangun alat pencampur cat otomatis digunakan untuk memudahkan dalam pencampuran warna yang dioperasikan secara otomatis dengan *input* warna yang diinginkan dengan menggunakan *Personal Computer (PC)*. Komponen *software* yang akan digunakan dalam pembuatan program pada rancangan alat pencampur cat yang telah dibuat adalah Delphi 7 dan Arduino IDE.

Delphi 7 merupakan *interface* yang digunakan untuk tampilan masukan warna yang diinginkan serta volume total dari cat yang dibutuhkan. Dari masukan volume total dan komposisi warna yang didapatkan maka selanjutnya akan dikonversi menjadi volume komposisi masing-masing warna dasar (merah, hijau,

biru) yang dibutuhkan dan akan dikirim ke Arduino IDE. Kemudian data berupa volume warna dasar cat yang didapat Arduino IDE akan diolah dan sistem akan melakukan pengendalian kerja aktuator pengaduk dan selang cat yang turun hingga mengenai *limit switch* bawah. Setelah itu sistem melakukan pencampuran warna cat tembok berdasarkan pengontrolan komposisi warna dasar (merah, biru, hijau) dengan mengendalikan volume cat menggunakan sensor *flow meter* digital, *valve* akan terbuka dan mengalirkan aliran warna dasar tersebut keluar ke wadah atau tempat hasil penampungan sesuai dengan yang ditentukan. Selanjutnya sistem akan melakukan pengadukan dengan waktu yang telah disesuaikan yaitu selama 30 detik, kemudian aktuator pengaduk dan selang cat akan kembali naik hingga mengenai batas *limit switch* atas dan proses pencampuran cat selesai. Dengan *software* yang telah dibuat pada rancang bangun alat pencampur cat diharapkan dapat melakukan proses pencampuran cat secara otomatis dan menjadi solusi bagi tukang cat dan industri kecil dalam melakukan pencampuran warna cat.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah disampaikan, maka disusun rumusan masalah yang mencakup :

1. Bagaimana membuat *software* program untuk mengendalikan aktuator naik/turun, motor buka/tutup *valve* dan motor pengaduk ?
2. Bagaimana rancangan program untuk sistem pencampur cat secara otomatis?

3. Bagaimana tingkat kestabilan sistem pada saat proses pencampuran cat?

1.3 Batasan Masalah

Dalam pembuatan tugas akhir, ada beberapa batasan masalah agar permasalahan tidak meluas diantaranya adalah :

1. Warna dasar cat yang digunakan yakni warna merah, hijau, biru.
2. Jenis cat yang digunakan adalah cat tembok.
3. Pencampuran cat maksimal 600ml.

1.4 Tujuan

Adapun tujuan yang ingin dicapai adalah :

1. Membuat program yang dapat mengendalikan kinerja dari motor pengaduk, aktuator naik/turun dan motor buka/tutup *valve*.
2. Mengetahui program yang dapat digunakan dalam sistem pencampuran cat secara otomatis.
3. Mengetahui kestabilan sistem dengan melakukan beberapa kali percobaan.

1.5 Manfaat

Adapun manfaat dari perancangan alat ini adalah untuk mempermudah dalam pencampuran warna cat dinding pada industri kecil dalam memenuhi keinginan dari konsumen untuk membuat warna cat dinding sesuai yang diinginkan.

BAB II

TINJAUAN PUSTAKA

2.1 Cat Tembok

Berdasarkan dari bahan pelarutnya, cat terbagi dalam dua jenis utama yaitu cat berbahan dasar air (*water-based paint*), dan cat berbahan dasar minyak (*solvent-base paint*). Pada umumnya cat tembok menggunakan cat berbahan dasar air. Cat berbahan dasar air memiliki karakter diantaranya adalah cepat kering, bau yang tidak terlalu menyengat, dan ketajaman warna yang cukup baik. Cat tembok memiliki beberapa fungsi diantaranya memberikan warna pada permukaan dinding, menutupi kekurangan bangunan dimana bidang bangunan yang sebelumnya retak atau terdapat cacat dapat terlihat rapi atau mulus setelah dilapisi cat. [edupaint (diakses Desember 2015)]

2.2 Pengolahan Citra (*Image processing*)

Pengolahan citra (*Image processing*) adalah suatu sistem dimana proses dilakukan dengan masukan (*input*) berupa citra (*image*) dan hasilnya (*output*) juga berupa citra (*image*). Dalam bidang komputer, citra atau disebut juga *image* merupakan representasi visual dari suatu objek setelah mengalami berbagai transformasi data dari berbagai bentuk rangkaian numerik.

Pengolahan citra terkait dengan pemrosesan citra atau *image processing*, khususnya dengan menggunakan komputer menjadi citra yang kualitasnya lebih baik. Dengan kata lain pengolahan citra adalah kegiatan memperbaiki kualitas

citra agar mudah diinterpretasi oleh manusia atau mesin (komputer). Pengolahan citra dapat dilakukan dengan melakukan pemrosesan warna atau representasi warna. Pengolahan warna dapat dilakukan dengan mengetahui RGB dan Histogram dari objek yang sudah berupa citra.

2.3 Representasi Warna

Representasi warna terdiri dari tiga unsur utama yaitu merah (*red*), hijau (*green*), dan biru (*blue*). Gabungan tiga warna tersebut membentuk warna-warna lainnya berdasarkan intensitas dari masing-masing warna tersebut dengan intensitas maksimal, dan warna hitam merupakan gabungan dari ketiga warna tersebut dengan intensitas minimal. Dari definisi tersebut, untuk menyajikan warna tertentu dapat dengan mudah dilakukan, yaitu dengan mencampurkan ketiga warna dasar RGB.

Dalam tugas akhir rancang bangun alat pencampur cat tembok otomatis berbasis PC menggunakan model warna RGB. Tingkat RGB pola bit dikomposisikan dari tiga warna tersebut dan masing-masing warna mempunyai 28 atau 256 bit (0 - 255). Nilai RGB ditentukan setelah mengambil *capture image* agar diketahui *range* warna dari *image* tersebut. Model warna RGB yang dapat dinyatakan dalam bentuk *indeks* warna RGB dengan cara menormalisasi setiap komponen warna dengan persamaan sebagai berikut :

$$r = \frac{R}{R+G+B} \dots\dots\dots(2.2)$$

$$g = \frac{G}{R+G+B} \dots\dots\dots(2.3)$$

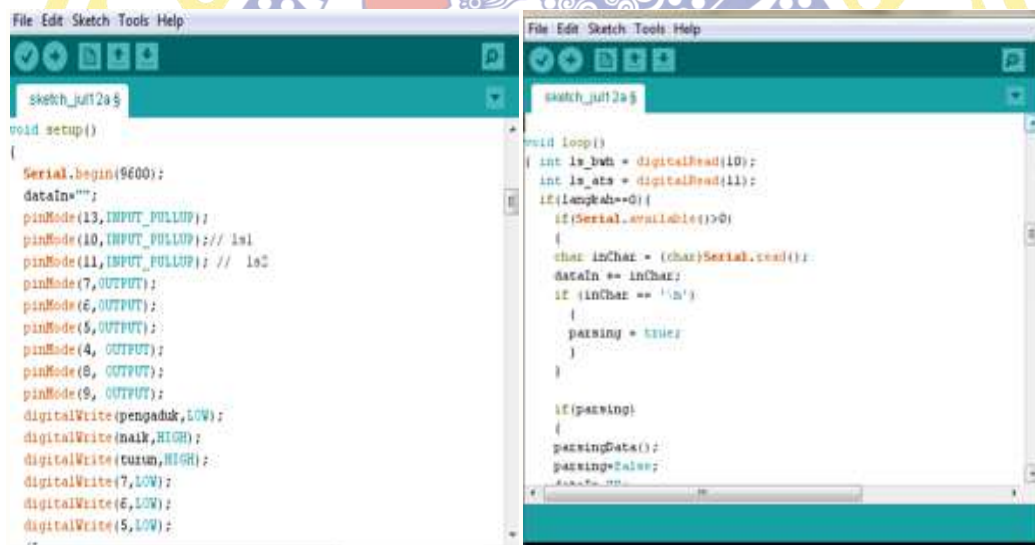
$$b = \frac{B}{R+G+B} \dots\dots\dots (2.4)$$

2.4 Software

Sistem pencampuran cat otomatis menggunakan 2 *software* diantaranya Arduino IDE dan Delphi 7.

2.4.1 Arduino IDE

Lingkungan *open-source Arduino* memudahkan untuk menulis kode dan meng-*upload* ke *board Arduino*. *Software IDE* Arduino dapat juga berfungsi untuk meng-*upload* program dengan hanya menekan tombol *upload* di *software IDE* Arduino. [arduino.cc]



```
File Edit Sketch Tools Help
sketch_jul12a.g
void setup()
{
  Serial.begin(9600);
  dataIn="";
  pinMode(13, INPUT_PULLUP);
  pinMode(10, INPUT_PULLUP); // 1s1
  pinMode(11, INPUT_PULLUP); // 1s2
  pinMode(7, OUTPUT);
  pinMode(6, OUTPUT);
  pinMode(5, OUTPUT);
  pinMode(4, OUTPUT);
  pinMode(8, OUTPUT);
  pinMode(9, OUTPUT);
  digitalWrite(pengaduk, LOW);
  digitalWrite(maik, HIGH);
  digitalWrite(cusam, HIGH);
  digitalWrite(7, LOW);
  digitalWrite(6, LOW);
  digitalWrite(5, LOW);
}

void loop()
{
  int ls_bwh = digitalRead(10);
  int ls_ats = digitalRead(11);
  if (langkah==0)
  {
    if (Serial.available()>0)
    {
      char inChar = (char)Serial.read();
      dataIn += inChar;
      if (inChar == '\n')
      {
        parsing = true;
      }
    }
    if (parsing)
    {
      parsingData();
      parsing=false;
    }
  }
}
```

Gambar 2.1 *Void Setup* dan *Void Loop* Arduino IDE

Pada gambar 2.1 menampilkan *void setup* dan *void loop* yang ada pada *software* Arduino IDE yang digunakan untuk mendeklarasikan *variabel* dan juga untuk tempat menuliskan eksekusi program. Setelah menuliskan *listing* program yang dibuat maka pilih *verify* untuk mengetahui apakah program yang telah dibuat ada kesalahan penulisan atau tidak. Selanjutnya memilih pilihan Tools-Board-ArduinoMega Or Mega 2560 karena *board*

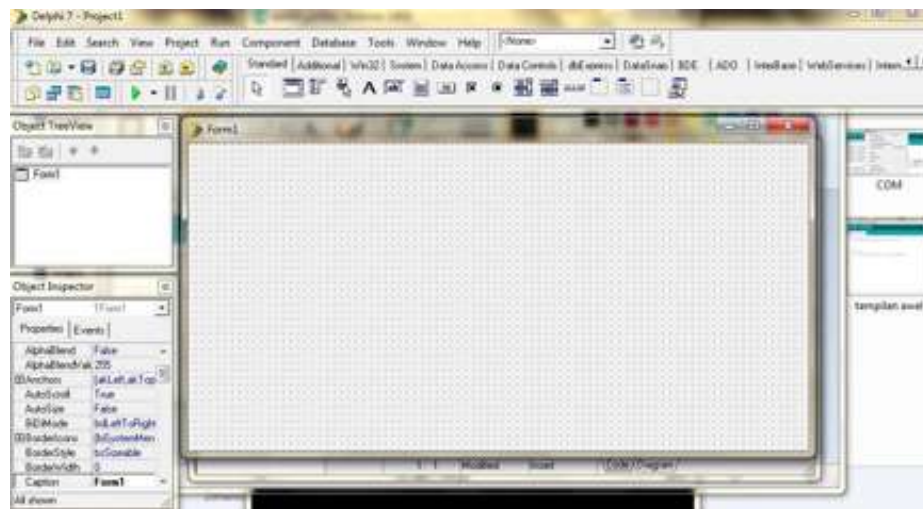
arduino yang digunakan pada tugas akhir ini adalah Arduino Mega. Setelah memilih board kemudian pilih COM yang sesuai dan nantinya COM juga di sesuaikan dengan yang ada di Delphi. Kemudian pilih *Upload* sehingga program dapat terkirim ke mikrokontroler.

2.4.2 Borland Delphi 7

Delphi adalah bahasa pemrograman yang menggunakan visualisasi, sama seperti bahasa pemrograman *Visual Basic* (VB). Namun Delphi menggunakan bahasa yang hampir sama dengan pascal (sering disebut obyek pascal), sehingga lebih mudah untuk digunakan. Bahasa pemrograman Delphi dikembangkan oleh *CodeGear* sebagai divisi pengembangan perangkat lunak milik *Embarcadero*. Divisi tersebut awalnya milik Borland, sehingga bahasa ini memiliki versi Borland Delphi.

Delphi juga menggunakan konsep yang berorientasi objek (OOP), maksudnya pemrograman dengan membantu sebuah aplikasi yang mendekati keadaan dunia yang sesungguhnya. Hal tersebut bisa dilakukan dengan cara mendesain objek untuk menyelesaikan masalah. OOP memiliki beberapa unsur yaitu: *Encapsulation* (pemodelan), *Inheritance* (Penurunan), *Polymorphism* (Polimorfisme).

Software delphi 7 digunakan untuk *interface* tampilan pilihan warna serta volume total cat dan sebagai tempat penulisan program untuk konversi perhitungan volume komposisi cat warna dasar.

Gambar 2.2 Tampilan Awal *Form* Delphi

Komponen serialNG adalah komponen delphi yang digunakan untuk mengakses port serial. Dengan menggunakan komponen serialNG, data dapat dibaca dan dikirim melalui port serial yang ada pada PC dengan menggunakan bahasa pemrograman delphi.



Gambar 2.3 Komponen SerialNG

BAB III

METODE PENELITIAN

3.1 Tempat dan Waktu Pelaksanaan Penelitian

Perancangan dan pembuatan alat ini dilakukan di Laboratorium Bengkel Mekanik, Departemen Fisika, Fakultas Sains dan Teknologi, Universitas Airlangga selama kurang lebih 4 bulan yang dimulai dari bulan April 2016 sampai Juli 2016.

3.2 Alat dan Bahan Penelitian

3.2.1 Alat – alat Penelitian

Alat perangkat keras (*hardware*):

1. Multimeter
2. Solder
3. Catudaya
4. Laptop/PC (*Personal Computer*)
5. *Downloader* Mikrokontroler

Alat perangkat lunak (*software*):

1. *IDE Arduino*
2. *Delphi 7*

3.2.2 Bahan – bahan Penelitian

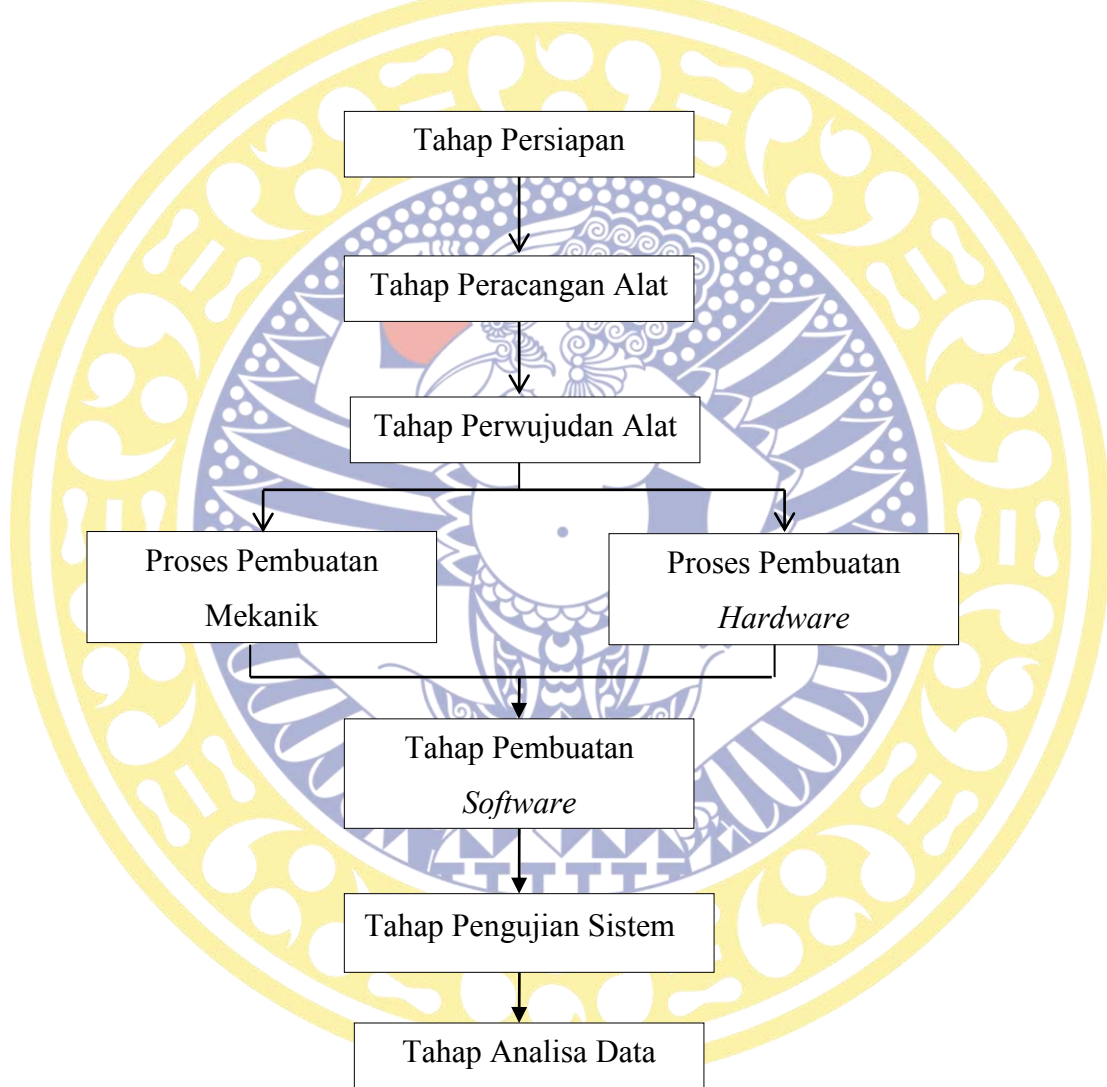
1. Motor *DC*
2. *Flow meter*
3. *Stir* / Pengaduk
4. ArduinoMega
5. *Valve*
6. Botol penampung cat
7. *Relay*
8. Cat Tembok Aga

3.3 Prosedur Penelitian

Pada prosedur penelitian dilakukan beberapa tahapan dalam pengerjaan alat. Prosedur yang digunakan dalam perancangan dan pembuatan alat adalah sebagai berikut :

1. Tahap Persiapan (Pembuatan sketsa mekanik *plan* yang dirancang serta studi literatur).
2. Tahap Perancangan Alat (proses pembuatan cara kerja alat dan alur kerja alat dari *input* yang digunakan dan *output* yang dihasilkan).
3. Tahap Perwujudan Alat (Pembuatan perangkat keras (*hardware*), *system* mekanik alat serta pemrograman alat).
4. Melakukan pengujian *hardware* dan *software*.
5. Analisis Data.

Masing-masing tahapan yang dilakukan penulis saling berkesinambungan satu sama lain, oleh sebab itu setiap tahapan yang dilakukan harus dipastikan sudah sesuai dengan yang diharapkan sebelum dilanjutkan ke tahap berikutnya. Berikut gambar 3.1 merupakan diagram blok tahapan prosedur penelitian :



Gambar 3.1 Diagram Blok Prosedur Penelitian

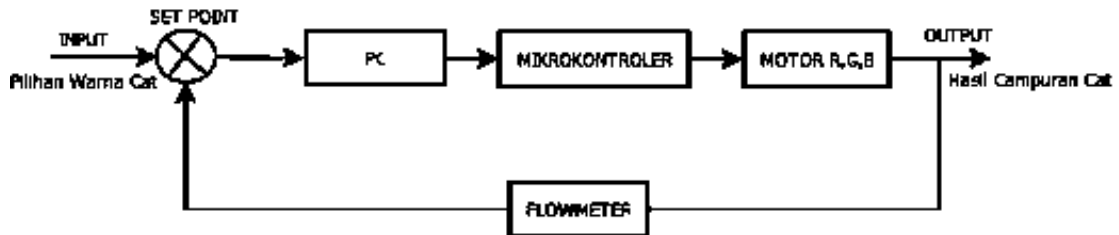
3.3.1 Tahap Persiapan

Tahap persiapan merupakan tahapan awal dalam melakukan penelitian, pada tahap persiapan penulis melakukan studi literatur dengan mencari berbagai acuan serta *referensi* pada buku, jurnal, artikel maupun tugas akhir mengenai penelitian serta cara kerja komponen yang digunakan, mempelajari teori-teori yang akan diaplikasikan pada sistem. Menganalisis kelebihan dan kekurangan sistem untuk menemukan solusi perbaikan dan pengembangan.

3.3.2 Tahap Perancangan Alat

Tahap perancangan alat terdiri dari perancangan *hardware* dan perancangan mekanik *system* alat. Sistem yang dibuat adalah meliputi pembuatan rangkaian *relay* yang akan bekerja mengatur bukaan *valve* sehingga aliran cat keluar ke wadah yang telah disediakan. PC digunakan untuk memberikan masukan pada arduino. Arduino bertugas mengatur bukaan *valve* menggunakan rangkaian *relay* untuk mengatur kadar cat agar sama dengan yang telah ditentukan dan juga mengatur motor dc dalam proses pengadukan cat. *Set point* berupa pilihan dari gradasi warna yang kita inginkan. Selanjutnya oleh mikrokontroler akan diolah. Kemudian mikrokontroler akan bekerja mengatur bukaan *valve* sesuai dengan kebutuhan dari warna yang telah ditentukan. Setelah cat keluar ke tempat hasil penampungan cat, sistem melakukan pencampuran cat dan didapatkan hasil campuran cat yang diinginkan.

Berikut gambar 3.2 adalah diagram kontrol dari sistem :



Gambar 3.2 Blok Diagram Sistem Kontrol

3.3.3 Tahap Perwujudan Alat

Tahap perwujudan alat dilakukan oleh bagian I meliputi pembuatan mekanik alat dan pembuatan *hardware* alat sesuai dengan perancangan yang sudah dibuat.

Tahap pembuatan mekanik terdiri atas pembuatan alat yang terdiri dari tempat cat, *valve* untuk mengatur aliran cat, *flowmeter* sebagai sensor, dan *stir*/pengaduk yang dirancang menjadi satu seperti pada desain mekanik pada gambar berikut ini:



Gambar 3.3 Alat Tampak Depan

Tahap pembuatan *hardware* terdiri atas pembuatan beberapa rangkaian elektronik yang dapat menjalankan rancang bangun alat pencampur cat secara otomatis. Adapun rancangan *hardware* dari sistem yang dibuat adalah sebagai berikut: Rangkaian Arduino, dan rangkaian *relay*. Rangkaian *relay* digunakan untuk mengontrol motor DC untuk mengatur bukaan *valve* aliran cat dan rangkaian *relay* dikendalikan oleh Arduino. *Flowmeter* yang digunakan sebagai sensor volume cat dan dihubungkan arduino agar nilai yang terbaca pada *flowmeter* dapat diolah oleh arduino yang dihubungkan pada pin *interupt*. Sensor *flowmeter* digunakan sebagai kontrol volume. *Valve* akan otomatis membuka sebanyak volume yang dibaca oleh *flowmeter* dan akan menutup apabila volume yang didapat telah sesuai dengan yang ditentukan. Selanjutnya rangkaian *relay* juga digunakan untuk mengontrol motor DC yang mengatur *axis* pergerakan naik turun pada selang cat dan pengaduk yang akan secara otomatis berhenti ketika mengenai *limit switch*. Rangkaian *relay* juga digunakan untuk mengontrol motor DC yang mengatur putaran pengadukan cat.

3.3.4 Tahap Pemrograman *Software*

Tahap pembuatan *software* meliputi pembuatan program untuk mengeksekusi rancangan *hardware* yang telah dibuat. *Software* yang digunakan yakni IDE Arduino dan Delphi 7 sebagai *input* pengontrol.

Pada delphi 7 ditampilkan 3 pilihan warna yang telah ditentukan nilai *red*, *green*, *blue* sebagai *input* pencampuran warna cat yang akan dicampur dan volume

total. Pada tahap awal pembuatan *software* yaitu melakukan inisialisasi sensor dalam sistem yakni sensor *flowmeter*. Sensor *flowmeter* memiliki *output* berupa sinyal dikonversi menjadi keluaran *digital* yang akan dibaca oleh mikrokontroler. Digunakan sebagai kontrol volume yang telah ditentukan. selanjutnya inisialisasi *relay* untuk mengendalikan motor pengaduk, aktuator naik turun, serta buka tutup *valve* untuk aliran cat.

Dari nilai volume total yang ditentukan dan pilihan warna yang telah dipilih selanjutnya akan diproses di delphi 7. Seperti pada pilihan warna 1 telah ditentukan nilai *red* adalah 255, nilai *green* adalah 255, dan nilai *blue* adalah 0. Dengan menggunakan parameter aturan r, g, b maka akan didapatkan volume dari masing-masing komposisi warna dasar. Berikut adalah perhitungan yang ada di delphi 7 untuk mendapatkan hasil volume komposisi warna cat dasar:

$$\text{Volume Red} = \left(\frac{r}{(r+g+b)} \times 100\% \right) \times \text{volume total}$$

$$\text{Volume Green} = \left(\frac{g}{(r+g+b)} \times 100\% \right) \times \text{volume total}$$

$$\text{Volume Blue} = \left(\frac{b}{(r+g+b)} \times 100\% \right) \times \text{volume total}$$

Selanjutnya volume hasil perhitungan tersebut dikirimkan ke Arduino IDE. Arduino IDE akan mengirimkan intruksi kepada arduino mega, dan arduino bekerja dengan mengaktifkan *relay* yang mengendalikan kerja motor aktuator turun hingga mengenai *limit switch* bawah. Selanjutnya mengaktifkan *relay* yang terhubung dengan motor bukaan *valve*, *valve* akan terbuka dan mengalirkan cat dan akan menutup apabila volume cat yang terdeteksi oleh *flowmeter* sudah sesuai dengan *set*

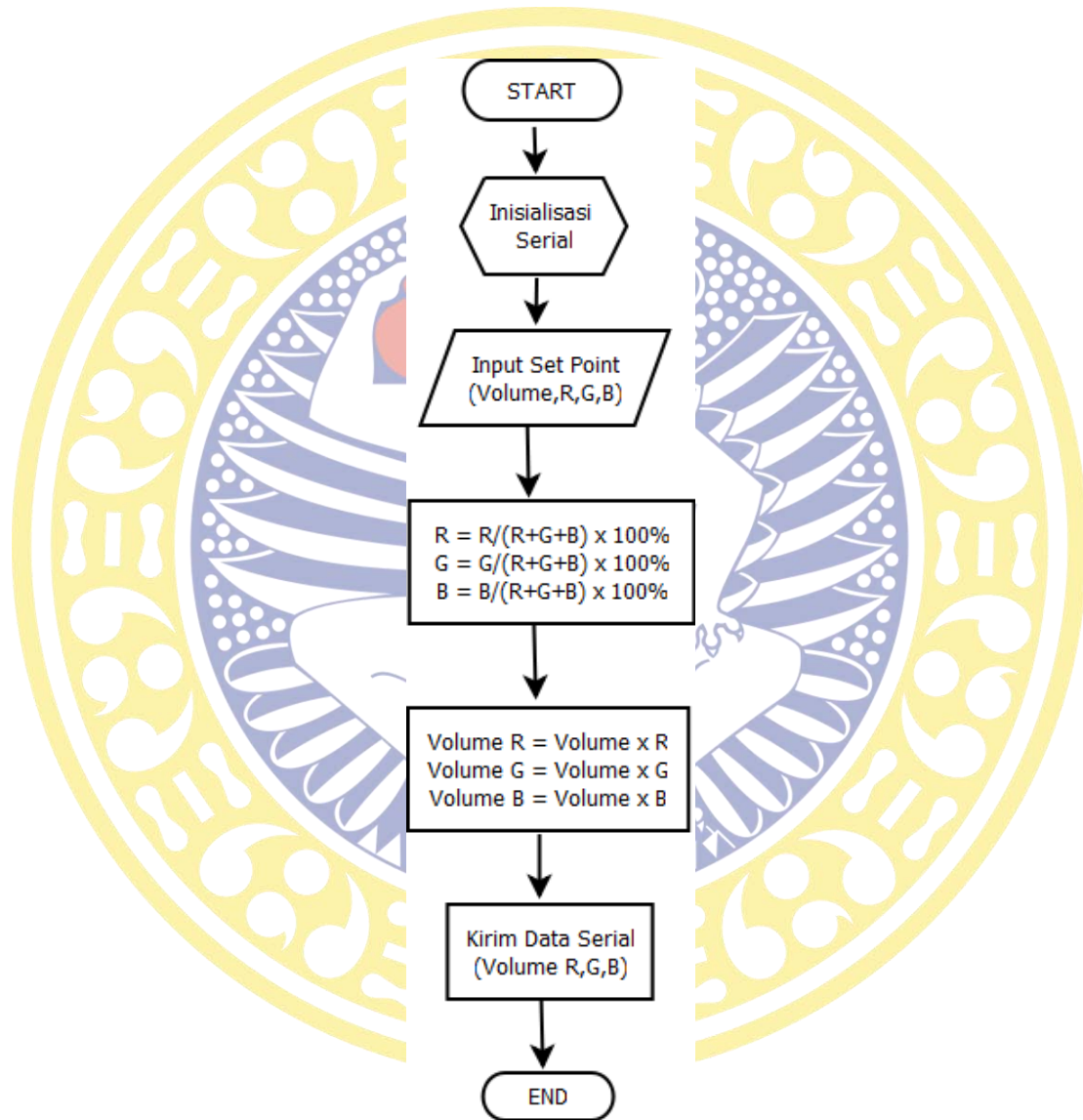
point. Kemudian *relay* yang terhubung pada motor pengaduk akan aktif dan sistem akan melakukan pengadukan cat selama *delay* waktu 30 detik. Setelah pengadukan selesai maka *relay* yang terhubung pada motor aktuator naik akan aktif dan aktuator akan bergerak naik hingga mengenai *limit switch* atas dan program pencampuran cat selesai. Pada tabel 3.1 ditampilkan pengalamatan untuk *port-port* yang akan digunakan dalam pembuatan *software* rancang bangun pencampur cat berbasis PC pada arduino :

Tabel 3.1 *Port* Pengalamatan Pada Arduino

<i>Port</i> Arduino	<i>Hardware</i>
<i>Port</i> 4	Motor Pengaduk
<i>Port</i> 5	Motor <i>Valve</i> 3
<i>Port</i> 6	Motor <i>Valve</i> 2
<i>Port</i> 7	Motor <i>Valve</i> 1
<i>Port</i> 8	Motor Turun (LS1)
<i>Port</i> 9	Motor Naik (LS2)
<i>Port</i> 10	<i>Limit Switch</i> Bawah
<i>Port</i> 11	<i>Limit Switch</i> Atas
<i>Port</i> 2	<i>Flowmeter</i> 1
<i>Port</i> 3	<i>Flowmeter</i> 2
<i>Port</i> 21	<i>Flowmeter</i> 3

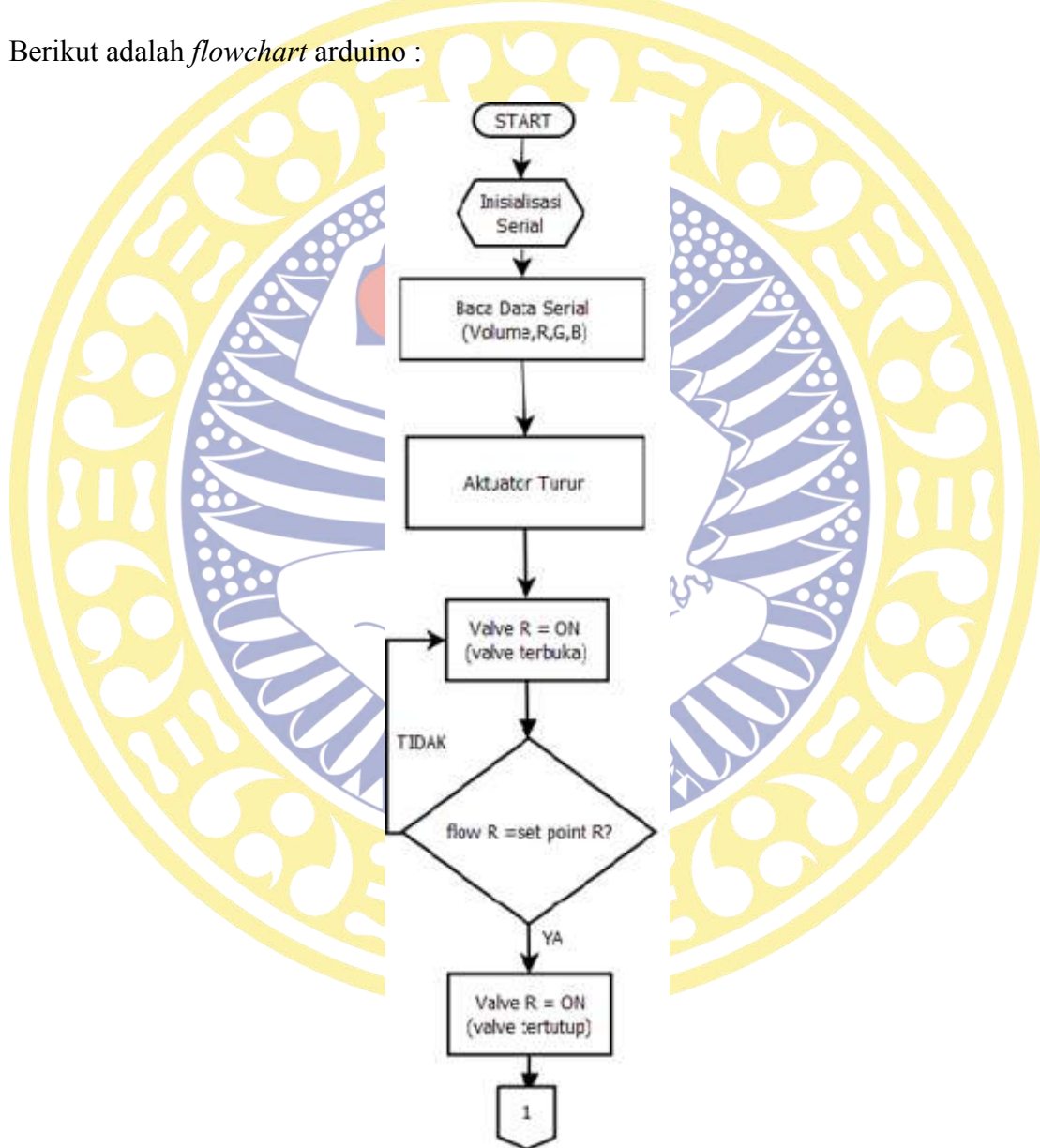
Sedangkan untuk tahap selanjutnya yaitu pembuatan diagram alir atau *flowchart* program Rancang Bangun Alat Pencampur Cat Tembok Berbasis PC.

Berikut gambar 3.4 adalah diagram *flowchart* program :

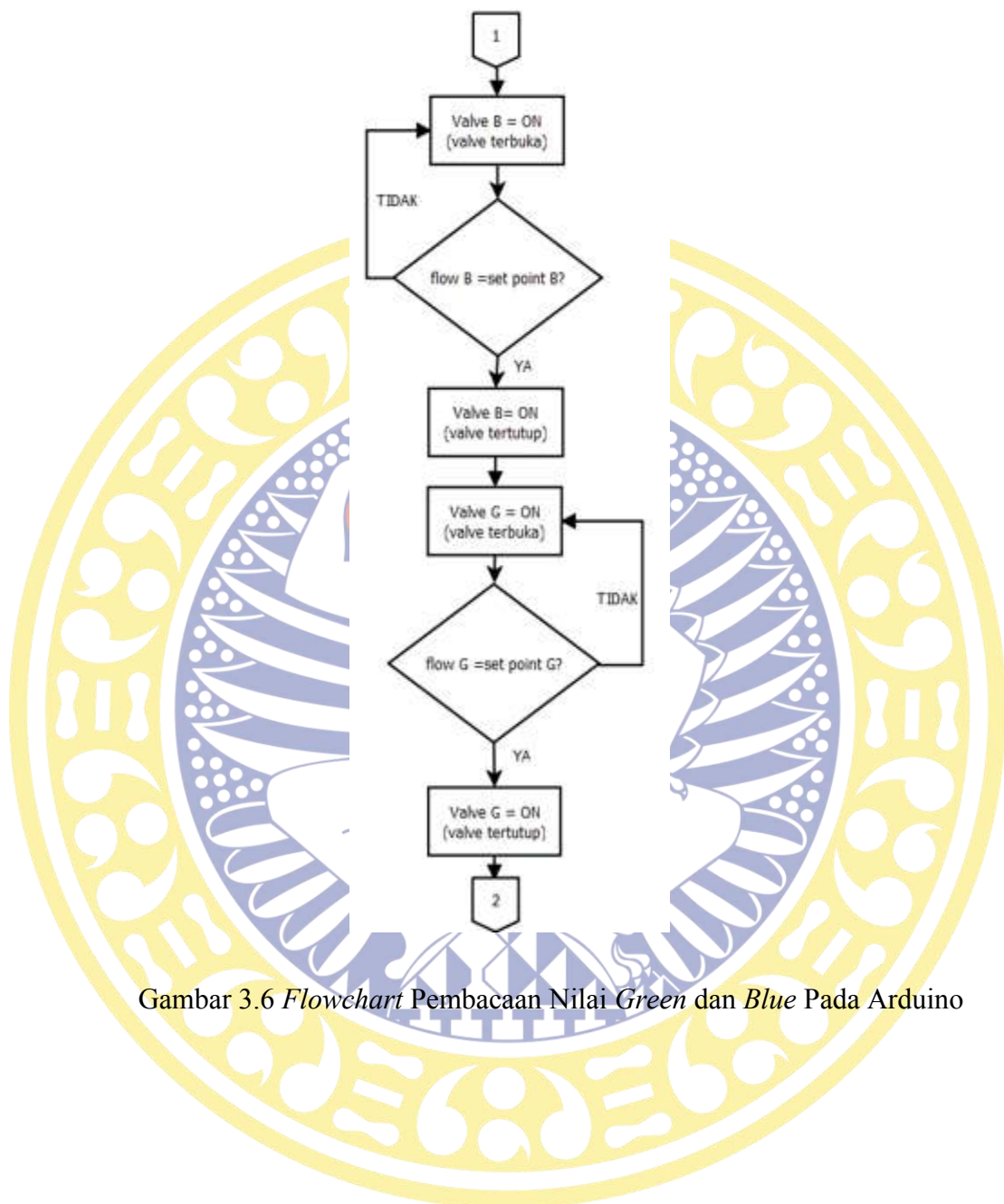


Gambar 3.4 *Flowchart* Pada Delphi

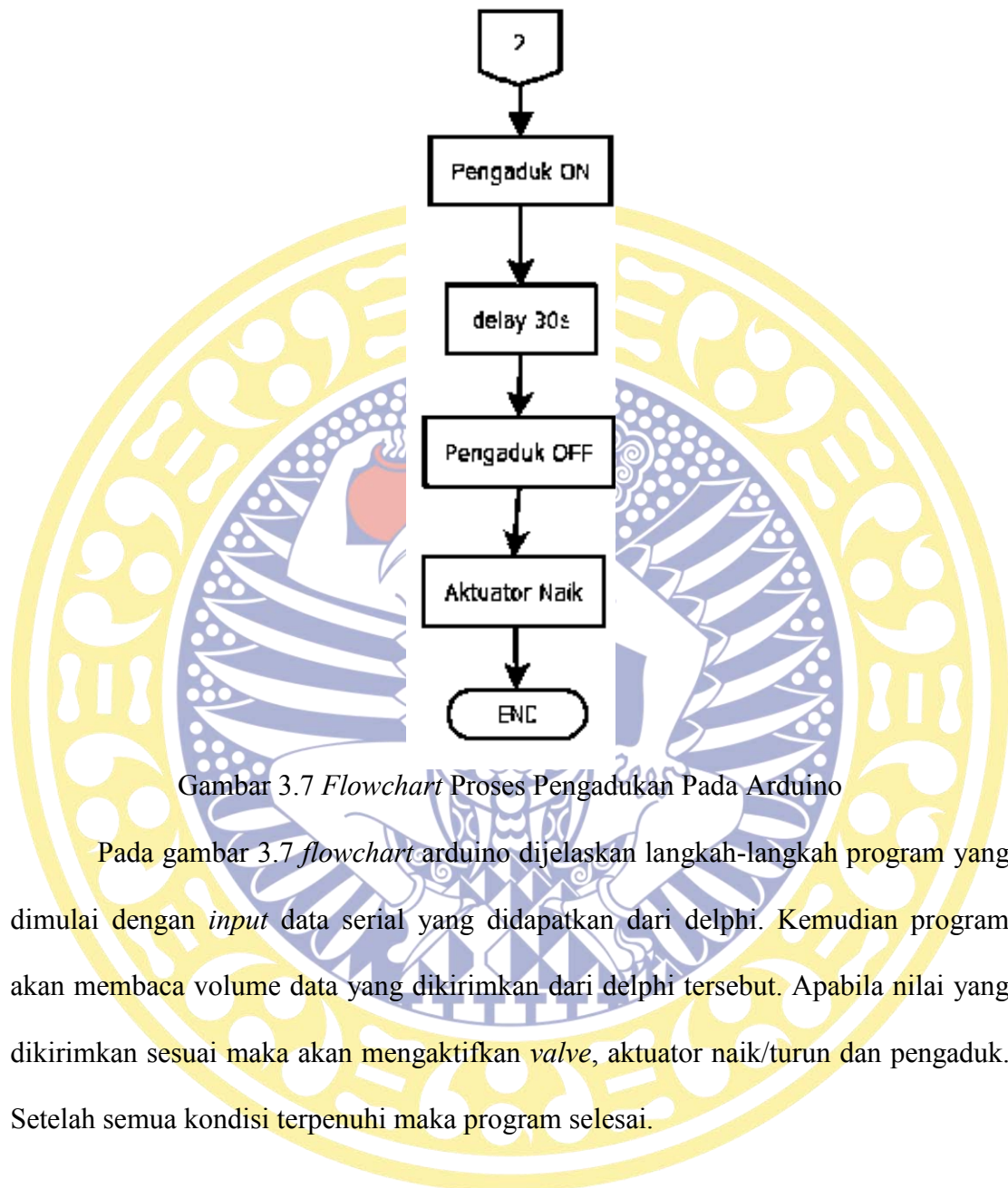
Pada gambar 3.4 *flowchart* delphi dijelaskan langkah-langkah pembuatan program yang dimulai dengan inialisasi arduino. Selanjutnya menuliskan perhitungan hasil volume dari masing-masing komposisi warna dasar. Kemudian data dikirim ke arduino dengan komunikasi serial menggunakan komponen SerialNG. Berikut adalah *flowchart* arduino :



Gambar 3.5 *Flowchart* Pembacaan Nilai Red Pada Arduino



Gambar 3.6 *Flowchart* Pembacaan Nilai *Green* dan *Blue* Pada Arduino



Gambar 3.7 *Flowchart* Proses Pengadukan Pada Arduino

Pada gambar 3.7 *flowchart* arduino dijelaskan langkah-langkah program yang dimulai dengan *input* data serial yang didapatkan dari delphi. Kemudian program akan membaca volume data yang dikirimkan dari delphi tersebut. Apabila nilai yang dikirimkan sesuai maka akan mengaktifkan *valve*, aktuator naik/turun dan pengaduk. Setelah semua kondisi terpenuhi maka program selesai.

3.4 Tahap Pengujian Sistem

Tahap pengujian sistem terdiri dari pengujian seluruh sistem alat yang telah dibuat meliputi pengujian sensor *flowmeter*, pengujian rangkaian modul *relay*, pengujian motor, dan uji *software*. Pengujian sistem dimaksudkan untuk mengetahui karakteristik dari sensor maupun sistem yang digunakan dalam penelitian. Berikut penjelasan masing-masing pengujian yang dilakukan :

3.4.1 Pengujian Sensor *Flowmeter*

Pengujian dilakukan dengan membandingkan antara volume *input* yang telah terukur dengan mengukur volume *output* dari *valve*. Dari melakukan perbandingan tersebut maka dapat dilakukan linieritas hubungan antara volume masukan terhadap volume keluaran yang terukur oleh *flowmeter* dengan menggunakan cat. Sehingga dapat diketahui karakteristik dari sensor *flowmeter* yang digunakan.

3.4.2 Pengujian *Software*

Pengujian *software* pada penelitian meliputi pengujian *respons hardware* terhadap program yang sudah dikirimkan ke mikrokontroler. Tahapan pengujian *software* digunakan untuk mengetahui alat yang sudah dikerjakan dapat membaca dan mengeksekusi perintah dari program yang telah dibuat atau tidak.

3.5 Analisis Data

Pengambilan data dilakukan untuk mengetahui seberapa efektif *hardware* dan *software* yang telah dibuat sehingga alat dapat bekerja sesuai dengan yang diharapkan. Uji kinerja alat dan analisis data diantaranya pengujian bahan yang akan digunakan *flowmeter*, pengujian kontrol kerja motor dan pengujian sistem secara keseluruhan. Pengujian pada *flowmeter* berupa pengambilan data dari *flowmeter* dengan cara mengalirkan air dengan ukuran volume bervariasi dan mengukur volume keluaran dari *flowmeter*. Tidak hanya air tetapi juga pengambilan data dilakukan dengan menggunakan cat tembok. Data yang didapat dari percobaan adalah volume cat yang akan dicampur dengan lama waktu pengadukan. Selain itu dilakukan analisis terhadap kinerja *relay* untuk kontrol kerja *on/off* pada *valve* aliran cat, motor pengaduk dan aktuator. Pengujian sistem secara keseluruhan bertujuan untuk mengetahui kinerja *hardware*, *software* maupun sistem pada alat secara keseluruhan yang dilakukan dengan melihat tingkat keberhasilan sistem.

BAB IV

HASIL DAN PEMBAHASAN

4.1 Pembuatan *Interface* Program Pada Delphi

Pada bagian delphi ditampilkan antar muka untuk memasukkan nilai, pilihan warna dan hasil perhitungan dari komposisi warna cat dasar. Gambar 4.1 berikut merupakan *interface* pada delphi pada rancang bangun alat pencampur cat tembok otomatis berbasis *personal computer* (PC).



Gambar 4.1 Perancangan *Interface* Pada Delphi

Pembuatan *interface* pada delphi juga diisi dengan program yang dibutuhkan. Terdapat kolom volume untuk memasukkan nilai volume total yang dibutuhkan. Komponen *radiobutton* digunakan untuk pilihan variasi warna yang diinginkan. *Button* “start” merupakan *button* untuk mengeksekusi hasil dari pilihan variasi warna dan volume total yang dibutuhkan. Selanjutnya hasil tampilan volume komposisi masing-masing warna cat akan ditampilkan pada

kolom R, G dan B. *Button* “ *kirim*” merupakan *button* yang digunakan untuk mengirim data serial dari delphi dengan arduino. SerialNG digunakan untuk komunikasi serial antara delphi dengan arduino.

4.2 Hasil Pembuatan Perangkat Lunak (*Software*)

Sub bab program *software* membahas tentang pembuatan dan pengujian perangkat lunak (*software*) sistem kontrol volume rancang bangun alat pencampur cat tembok berbasis PC.

4.2.1 Sub Program Pemilihan Warna Cat Pada Delphi

Sub bab pemilihan warna cat membahas mengenai program pada delphi yang digunakan untuk menampilkan hasil perhitungan dari volume cat yang dibutuhkan. Berikut adalah program pada delphi :

```
var
    volume:integer;
    merah,hijau,biru,r,g,b,volumeR,volumeG,volumeB:real;
procedure TForm1.Button1Click(Sender: TObject);
begin
    if RadioButton1.Checked then //pilihan warna 1
    begin
        volume:=StrToInt(Edit1.Text);
        r:=255;
        g:=255;
        b:=0;
        merah:=(r/(r+g+b))*100;
        hijau:=(g/(r+g+b))*100;
        biru :=(b/(r+g+b))*100;
        volumeR:=volume*merah/100;
```



```

volumeG:=volume*hijau/100;
volumeB:=volume*biru/100;
Edit2.Text:=FloatToStrF(volumeR,ffFixed,3,0);
Edit3.Text:=FloatToStrF(volumeG,ffFixed,3,0);
Edit4.Text:=FloatToStrF(volumeB,ffFixed,3,0);
end;
procedure TForm1.Button2Click(Sender: TObject);
begin
SerialPortNG1.Active:=True;
SerialPortNG1.SendString(FloatToStrF(volumeR,ffFixed,3,0)+' '+FloatToStrF(v
olumeG,ffFixed,3,0)+' '+FloatToStrF(volumeB,ffFixed,3,0)+'#'+#10);

```

Program pemilihan warna cat menjelaskan tentang perhitungan dari komposisi cat warna dasar dari volume total cat yang dibutuhkan dan dengan memilih pilihan warna pada *radiobutton*. Pada *radiobutton* terdapat komposisi warna yang telah ditentukan. Program *button* “start” berisikan *list* program perhitungan komposisi masing-masing warna dasar. Program *button* “kirim” berisikan program untuk mengirimkan data serial ke arduino.

4.2.2 Sub Program *Relay* Untuk Kendali *Valve*

Sub bab program *relay* membahas mengenai program untuk menyalakan *relay* sehingga *relay* dapat menjadi kendali motor DC yang bekerja untuk kontrol buka/tutup *valve*. Program *relay* untuk mengendalikan motor DC dibuat pada aplikasi arduino. Berikut program yang digunakan :

```

#define relay3
pinMode(5,OUTPUT);
int langkah=0;

```

```

voidsetup()
{ pinMode(5,OUTPUT);
  digitalWrite(5,LOW);}
voidloop()
{ if(langkah==0){int mulai=digitalRead(13);
  if(mulai==LOW){langkah=1;}}
  if(langkah==1){digitalWrite(relay3,HIGH);delay(200);langkah=2;}
  if(langkah==2){digitalWrite(relay3,LOW);langkah=3; }
  if(langkah==4){digitalWrite(relay3,HIGH);delay(230);langkah=5;}
  if(langkah==5){digitalWrite(relay3,LOW);langkah=0;}

```

Berdasarkan *listing* program dapat diketahui bahwa pembacaan program dimulai dengan menginisialisasikan pin 5 pada arduino menjadi variabel *relay*. Program kemudian disetting dalam *void setup* bahwa mode pin *relay* tersebut digunakan sebagai *output* dan kondisi awal dari *relay* adalah *LOW* atau berlogika 0.

Program kemudian dieksekusi pada *void loop*. Pada program dijelaskan kondisi *relay HIGH* atau *relay* bekerja membuka *valve* selama 200ms. Kemudian kondisi *relay HIGH* atau *relay* bekerja menutup *valve* dengan waktu 230ms.

4.2.3 Sub Program *Flowmeter*

Sub bab program *flowmeter* membahas mengenai program untuk inialisasi dan perhitungan *flowmeter* sehingga didapatkan nilai volume dalam satuan ml. Berikut program yang digunakan :

```

byte sensorInterrupt = 0;// 0 = digital pin 2
byte sensorPin = 2;
float calibrationFactor = 4.5;

```

```

volatile byte pulseCount; //variabel byte pulsecount
float flowRate; // karena flowrate bernilai pecahan maka tipe data yang digunakan
float
unsigned int flowMilliLitres;
unsigned long totalMilliLitres;
unsigned long oldTime;
} tipe data agar tidak bernilai negatif

{pinMode(sensorPin, INPUT);digitalWrite(sensorPin, HIGH);
pulseCount = 0;
flowRate = 0.0;
flowMilliLitres = 0;
totalMilliLitres = 0;
oldTime = 0
} deklarasi nilai awal

// konfigurasi untuk trigger sehingga kondisi pulsa berubah dari HIGH ke LOW
attachInterrupt(sensorInterrupt, pulseCounter, FALLING); }
voidloop()
{ if(langkah==6){
if((millis() - oldTime) > 1000) // counter per detik
{ detachInterrupt(sensorInterrupt); //mengaktifkan sensor interupt
flowRate = ((1000.0 / (millis() - oldTime))*pulseCount)/calibrationFactor;
//perhitungan untuk mendapatkan nilai debit
oldTime = millis();
flowMilliLitres = (flowRate / 60) * 1000; // perhitungan untuk mendapatkan nilai
volume tiap 1 s
totalMilliLitres += flowMilliLitres; // mengkalkulasi nilai total volume yang
terdeteksi
unsigned int frac;
/Untuk Menampilkan Flowrate (debit)
Serial.print("Flow rate: "); //menampilkan tulisan "Flow Rate:"
Serial.print(int(flowRate)); // menampilkan flowrate dengan tipe data integer
Serial.print("."); // menampilkan titik desimal
frac = (flowRate - int(flowRate)) * 10; // frac menyimpan data dari flowrate

```


sebelumnya dikurangi dengan data yang baru dikalikan 10

```

Serial.print(frac, DEC) ; // menampilkan frac
Serial.print("L/min");// menampilkan "L/min"
//Menampilkan MilliLitres (volume/detik)
Serial.print(" Current Liquid Flowing: ");
Serial.print(flowMilliLitres); // menampilkan flowMillilitres
Serial.print("mL/Sec"); menampilkan "mL/Sec"
//Menampilkan Total Keseluruhan
Serial.print(" Output Liquid Quantity: ");
Serial.print(totalMilliLitres); //menampilkan totalMilliLitres
Serial.println("mL"); //menampilkan "mL"
//Reset counter pulsa untuk dapat di increment kan kembali
pulseCount = 0;
attachInterrupt(sensorInterrupt, pulseCounter, FALLING);//mengaktifkan interupt
void pulseCounter()
{ pulseCount++;} /increment counter pulsa

```

Berdasarkan *listing* program dapat diketahui bahwa pembacaan program dimulai dengan mendeklarasikan *interrupt* 0 yang terdapat pada pin 2 sebagai pin *interupt* yang dipakai pada arduino. Selain itu mendeklarasikan faktor kalibrasi. Selanjutnya mendeklarasikan variabel yang digunakan untuk perhitungan dari sensor *flowmeter*. Kemudian program disetting dalam *void setup* bahwa Sensorpin tersebut sebagai *input* dengan kondisi awal *HIGH*. karena *flowmeter* yang digunakan adalah aktif *LOW* maka kondisi awal yang diberikan adalah *HIGH* agar *flowmeter* dalam keadaan *off* atau tidak membaca sinyal. Kemudian dikonfigurasi dari keadaan *HIGH* ke *LOW*.

Program dieksekusi di *void loop*. Langkah awal yang dieksekusi telah dijelaskan pada sub program *relay*. Selanjutnya menjelaskan pembacaan

flowmeter. Jika langkah sama dengan 3 maka akan menghitung apakah hasil dari *millis* dikurangi dengan *old time* kurang dari 1000. Kemudian melakukan perhitungan debit yang disimpan dalam variabel *flowrate*. Selanjutnya memanggil *oldTime* yang bernilai = *millis* artinya lama waktu dalam 1 *milisecond*. Kemudian menghitung volume liter dalam 1 detik dengan cara hasil debit dibagi 60 dikalikan 1000. Menghitung total keseluruhan volume dengan cara menambahkan dengan nilai sebelumnya. *Sintaks serial print* dimaksudkan untuk memonitoring nilai perhitungan yang telah dieksekusi oleh *flowmeter* karena tidak menggunakan LCD untuk menampilkan hasil perhitungan volume.

4.2.4 Sub Program *Parsing Data*

```
int langkah=0; //langkah=0 tipe datanya integer
String dataIn; //dataIn tipe datanya string
String dt[3]; //tipe data array string dan jumlah array 3
int i; //i tipe datanya integer
boolean parsing=false; //tidak ada parsing data
int volumeR;
int volumeG;
int volumeB; } deklarasi nilai R,G,B

void loop () {
if (langkah==0) {if (Serial.available()>0)
{char inChar = (char)Serial.read();
dataIn += inChar;
if (inChar == '\n')
{parsing = true;}}
if (parsing)
{parsingData();
parsing=false;
```

```

dataIn="";
langkah=1;}}
if(langkah==19){volumeR=dt[0].toInt();volumeG=dt[1].toInt();volumeB=dt[2].to
Int();langkah=20;}

void parsingData()
{int j=0;
//inisialisasi variabel, (reset isi variabel)
dt[j]="";
//proses parsing data
for(i=0;i<dataIn.length();i++)
{if ((dataIn[i] == '#') || (dataIn[i] == ','))//pengecekan tiap karakter dengan
karakter (#) dan (,)
{ j++;//increment variabel j, digunakan untuk merubah index array penampung
dt[j]=""; } //inisialisasi variabel array dt[j]
else
{dt[j] = dt[j] + dataIn[i];} //proses tampung data saat pengecekan karakter selesai
}}

```

Pada sub program *parsing* data dijelaskan program untuk *parsing* data dari delphi. Dimulai dengan mendeklarasikan variabel yang akan digunakan. Selanjutnya program dituliskan pada *void loop*. Jika langkah sama dengan 0 maka akan mengeksekusi pembacaan kondisi serial variabel. Apabila serial variabel lebih dari 0 maka akan melakukan pembacaan dengan tipe data char. dataIn didapatkan dari penjumlahan data inChar. Jika inChar telah memenuhi data selanjutnya *parsing* akan bernilai benar. Mengisikan data yang dikirim sesuai batas karakter yang ditentukan.

Kemudian mengeksekusi program yang ada pada *void parsingData*, yang

pertama inialisasi data j yang digunakan untuk variabel penambah data *array* ke- n dengan nilai awal 0. kemudian proses *parsing* dari data dimulai, dengan menghitung panjang data yang masuk ke dalam *array* dengan bantuan variabel i yang nilai awalnya bernilai 0, sampai data yang terbaca yaitu *separator* atau pemisah data ke-1 dengan yang lainnya. Apabila data yang terbaca bukan merupakan *separator* maka data *input* ke i (data yang disimpan pada variabel i) akan ditampung pada *array* yang sama sampai data *input* yang disimpan di i bernilai *separator* ('#' atau ','), dan apabila data *input* i bernilai *separator* maka nilai variabel j akan diincrementkan atau ditambah yang menunjukkan nilai *array* bertambah (*array* 0 ke *array* 1 dan seterusnya). Setelah *parsing* data selesai maka nilai dari data *input* akan kembali dikosongkan atau *reset*.

4.3 Pengujian Kinerja Relay

Pengujian kinerja *relay* digunakan untuk mengetahui kondisi dari *relay*, motor dc dan *flowmeter* dalam keadaan baik atau tidak. *Relay* pada sistem berfungsi sebagai pengontrol yang mengatur kerja dari motor dc. Motor dc digunakan untuk mengatur gerakan naik turun aktuator selang dan pengaduk, putaran pengaduk dan juga buka tutup *valve*. Berikut tabel 4.1 adalah tabel pengujian *relay* :

Tabel 4.1 Pengujian Relay

<i>Input power supply</i>	Kontrol	<i>Output relay</i>	Keterangan
H	L	H	NO (<i>Normaly Open</i>)
H	L	L	NC (<i>Normaly Close</i>)
H	H	H	NO (<i>Normaly Open</i>)
H	H	L	NC (<i>Normaly Close</i>)

Pengujian dilakukan dengan cara menghubungkan rangkaian *relay*, motor dc dan *flowmeter* dengan arduino mega dan menuliskan program pada Arduino IDE. Rangkaian *relay* dihubungkan pada *port* arduino mega sebagai keluaran yang selanjutnya keluaran dari rangkaian *relay* dihubungkan pada motor dc, sedangkan *flowmeter* dihubungkan pada *port interrupt* pada arduino mega sebagai masukan. Pengujian yang pertama yaitu dengan memberikan kondisi *HIGH* atau nilai 1, jika *relay* kondisi *ON* atau *N* maka *relay* aktif *HIGH*. Apabila *relay* kondisi *OFF* maka *relay* aktif *LOW*. Selanjutnya pada pengujian kedua memberikan nilai 0 atau kondisi *LOW*, jika *relay* kondisi *ON* maka *relay* aktif *LOW*. Apabila *relay* kondisi *OFF* maka *relay* aktif *HIGH*. Dari keadaan tersebut maka diketahui bahwa *relay*, *flowmeter* dan motor dc yang digunakan dalam kondisi yang baik untuk digunakan.

4.4 Pengujian Sensor *Flowmeter*

Pengujian dilakukan dengan membandingkan antara volume masukan yang telah terukur dengan mengukur volume keluaran dari *flowmeter* dan dari *valve*. Dari melakukan perbandingan tersebut maka dapat dilakukan linieritas hubungan antara volume masukan terhadap volume keluaran yang terukur oleh *flowmeter*. Didapatkan hasil selisih pembacaan antara volume masukan dan volume keluaran dari *flowmeter* dan *valve* sehingga dapat diketahui karakteristik dari sensor *flowmeter* yang digunakan.

4.4.1 Pengujian Awal *Respons Flowmeter* Terhadap Laju Aliran

Pengujian dilakukan untuk mengetahui apakah sensor *flowmeter* yang telah dirangkai pada sistem sudah dapat membaca laju dari aliran yang dilewatkan *flowmeter* atau tidak. Berikut tabel 4.2, 4.3 dan 4.4 adalah tabel data volume cat dan volume *flowmeter* menggunakan cat tembok tanpa buka/tutup *valve*.

Tabel 4.2 Tabel Hubungan Antara Volume Cat dengan Volume *Flowmeter* Pada *Flowmeter* Merah

No.	Volume Cat (ml)	Volume <i>Flowmeter</i> (ml)
1.	50	43
2.	100	106
3.	150	184
4.	200	195
5.	250	250

Tabel 4.3 Tabel Hubungan Antara Volume Cat dengan Volume *Flowmeter* Pada *Flowmeter* Hijau

No.	Volume Cat (ml)	Volume <i>Flowmeter</i> (ml)
1.	50	38
2.	100	93
3.	150	188
4.	200	255
5.	250	299

Tabel 4.4 Tabel Hubungan Antara Volume Cat dengan Volume *Flowmeter* Pada *Flowmeter* Biru

No.	Volume Cat (ml)	Volume <i>Flowmeter</i> (ml)
1.	50	45
2.	100	91
3.	150	99
4.	200	128
5.	250	145

Pada tabel 4.2, 4.3, dan 4.4 didapatkan data hubungan antara volume cat dengan volume *flowmeter* dengan menggunakan perbandingan campuran bahan cat dan air yang digunakan yaitu 60:40 pada masing-masing *flowmeter* merah, hijau, dan biru. Karena pada saat menggunakan komposisi perbandingan cat 75 : air 25 yaitu dimana bahan cat lebih banyak daripada air maka sensor *flowmeter* tidak dapat membaca volume dari campuran komposisi bahan tersebut. Dari data didapatkan hubungan setiap kenaikan ukuran volume masukan menghasilkan ukuran volume keluaran yang juga naik dan dapat diketahui bahwa sensor *flowmeter* dapat bekerja sesuai yang diinginkan.

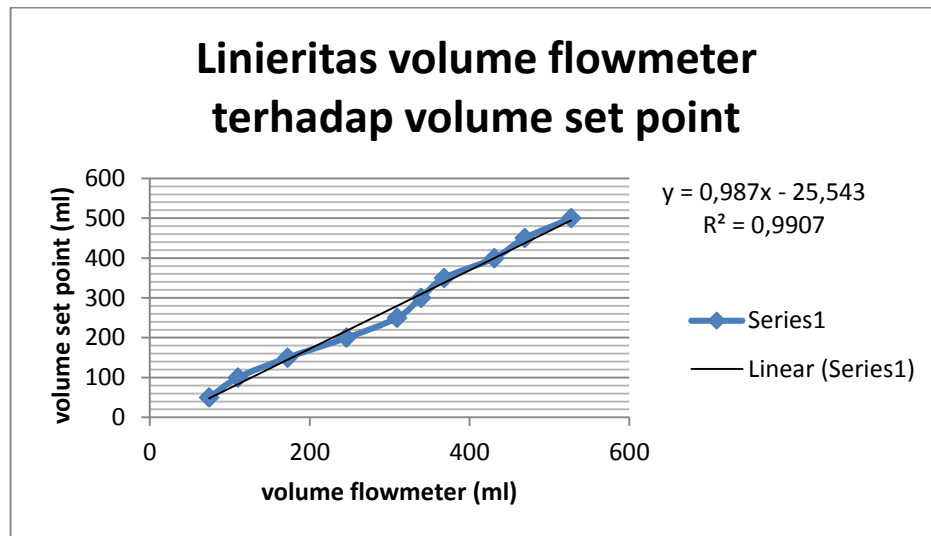
4.4.2 Pengujian *Respons Flowmeter* Menggunakan Buka/Tutup *Valve*

Metode yang digunakan hampir sama dengan pengujian tahap awal hanya saja pada tahap awal dilakukan secara langsung tanpa menggunakan buka/tutup *valve*. Pada pengujian *respons flowmeter* menggunakan buka/tutup *valve* dilakukan dengan kondisi awal *valve* dalam keadaan tertutup. Kemudian mengisi

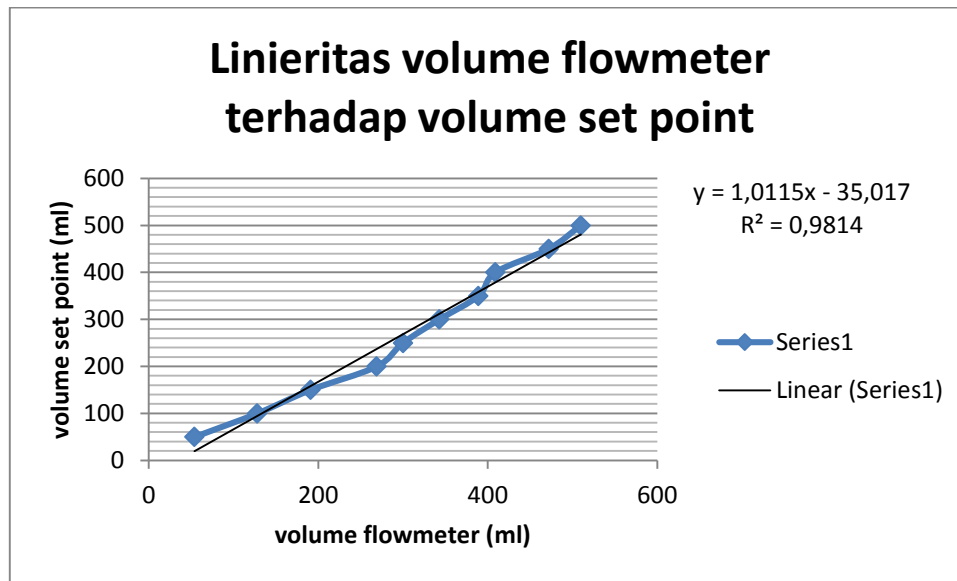
cat hingga penuh pada tempat *input* cat dan mengaktifkan motor dc untuk kontrol bukaan *valve* sehingga didapatkan volume *output* sama seperti volume yang ditentukan. Pengujian dilakukan dengan membandingkan antara volume cat yang telah dimasukkan dengan volume pembacaan dari *flowmeter* dan volume hasil yang diukur dengan menggunakan *beacker glass*. Berikut adalah tabel data volume *set point* dengan volume *flowmeter* dan volume hasil pada masing-masing *flowmeter*.

Tabel 4.5 Tabel Hubungan Antara Volume *Set point* dengan Volume *Flowmeter* dan Volume Hasil Pada *Flowmeter* Merah

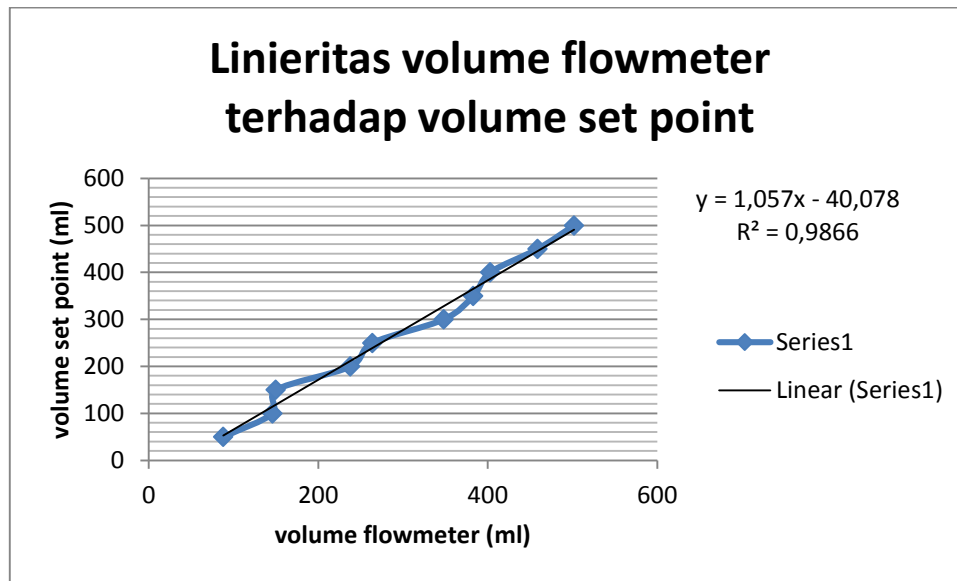
No.	<i>Set point</i> (ml)	Volume <i>Flowmeter</i> (ml)	Volume Hasil (ml)	ΔV <i>Set point</i> dan <i>Flowmeter</i> (ml)	ΔV <i>Flowmeter</i> dan Hasil (ml)
1.	50	74	75	24	1
2.	100	110	107	10	3
3.	150	172	160	22	12
4.	200	246	219	46	27
5.	250	309	255	59	54
6.	300	339	304	39	35
7.	350	368	330	18	38
8.	400	431	350	31	81
9.	450	469	395	19	74
10.	500	527	490	27	37
$\overline{\Delta V}$ (ml)				29,5	36,2

Gambar 4.2 Grafik Linieritas *Flowmeter* MerahTabel 4.6 Tabel Hubungan Antara Volume *Set point* dengan Volume *Flowmeter* dan Volume Hasil Pada *Flowmeter* Hijau

No.	Volume <i>Set point</i> (ml)	Volume <i>Flowmeter</i> (ml)	Volume Hasil (ml)	ΔV <i>Set point</i> dan <i>Flowmeter</i> (ml)	ΔV <i>Flowmeter</i> dan Hasil (ml)
1.	50	54	68	4	14
2.	100	128	113	28	15
3.	150	191	155	41	36
4.	200	269	238	69	31
5.	250	300	246	50	54
6.	300	343	270	43	73
7.	350	389	316	39	73
8.	400	409	340	9	69
9.	450	472	400	22	72
10.	500	510	497	10	13
$\bar{\Delta V}$ (ml)				31,5	45

Gambar 4.3 Grafik Linieritas *Flowmeter* HijauTabel 4.7 Tabel Hubungan Antara Volume *Set point* dengan Volume *Flowmeter* dan Volume Hasil Pada *Flowmeter* Biru

No.	Volume <i>Set point</i> (ml)	Volume <i>Flowmeter</i> (ml)	Volume Hasil (ml)	ΔV <i>Set point</i> dan <i>Flowmeter</i> (ml)	ΔV <i>Flowmeter</i> dan Hasil (ml)
1.	50	88	84	38	4
2.	100	146	146	46	0
3.	150	150	150	0	0
4.	200	238	220	38	18
5.	250	264	250	14	14
6.	300	348	293	48	55
7.	350	383	347	33	36
8.	400	403	360	3	43
9.	450	459	400	9	59
10.	500	502	420	2	82
$\overline{\Delta V}$ (ml)				23,1	31,1



Gambar 4.4 Grafik Linieritas *Flowmeter* Biru

Dari data pada tabel 4.5, 4.6 dan 4.7 dapat diketahui bahwa semakin besar volume yang dimasukkan maka volume yang dibaca *flowmeter* dan dihasilkan semakin besar. Dengan memberikan lama waktu *delay* buka dan tutup *valve* sehingga cat dapat mengalir keluar ke wadah penampungan cat dan menggunakan komposisi perbandingan bahan cat dengan air yaitu 60:40 didapatkan selisih rata-rata volume antara *set point* dengan hasil dan selisih rata-rata volume *set point* dengan volume *flowmeter*. Pada tabel 4.5, 4.6 dan 4.7 didapatkan selisih rata-rata terbesar antara volume *flowmeter* dengan volume hasil adalah *flowmeter* hijau yaitu 45 ml dan yang terkecil adalah *flowmeter* biru yaitu 31,1 ml. Selisih rata-rata terbesar antara volume *set point* dengan volume *flowmeter* adalah *flowmeter* hijau yaitu 31,5 ml dan yang terkecil adalah *flowmeter* biru yaitu 23,1 ml. Dari tabel dapat dibuat grafik dan didapatkan linieritas pada *flowmeter* merah 0,990 ; *flowmeter* hijau 0,981 dan biru 0,986.

4.5 Hubungan Antara Volume Cat Dengan Waktu Pengadukan

Untuk mengetahui hubungan antara cat dengan waktu pengadukan agar cat dapat tercampur merata maka dilakukan pengambilan data setiap kenaikan 50 ml dengan lama waktu pengadukan. Data yang diambil adalah volume cat yang akan dicampur dengan lama pengadukan. Dengan memberikan *delay* waktu buka tutup *valve* dan lama waktu pengadukan maka cat yang telah ditentukan maka *valve* akan terbuka sesuai lama *delay* dan cat akan keluar mengalir sesuai volume yang diinputkan kemudian pengaduk *ON* sesuai *delay*. Hasilnya akan didapat tercampur atau tidaknya campuran cat tersebut sesuai dengan lamanya pengadukan dan volume cat. Parameter dari tercampur atau tidaknya cat yaitu dengan melakukan pengamatan secara langsung/secara *visual*. Dengan membandingkan hasil antara pengamat 1 dengan pengamat 2. Berikut adalah tabel 4.8 data dari hubungan antara volume cat dan waktu lama pengadukan :

Tabel 4.8 Tabel Hubungan Antara Volume Cat dengan Waktu Lama Pengadukan

No	Volume Cat (ml)	t pengadukan (s)		Tercampur/Tidak Tercampur
		Pengamat 1	Pengamat 2	
1.	450	5	5	Tidak Tercampur
		10	10	Tidak Tercampur
		15	15	Tidak Tercampur
		20	20	Tercampur
		25	25	Tercampur
2.	500	5	5	Tidak Tercampur
		10	10	Tidak Tercampur
		15	15	Tidak Tercampur
		20	20	Tercampur
		25	25	Tercampur
3.	550	5	5	Tidak Tercampur
		10	10	Tidak Tercampur
		15	15	Tidak Tercampur
		20	20	Tidak Tercampur
		25	25	Tidak Tercampur
		30	30	Tercampur

Setelah melakukan pengujian hubungan antara lama waktu pengadukan dengan volume cat maka lama waktu yang digunakan dalam proses pencampuran cat adalah 30 detik selain itu tipe dari bentuk pengadukan juga diperhatikan agar cat dapat tercampur merata.

4.6 Pengujian Keberhasilan Sistem

Pengujian sistem secara keseluruhan bertujuan untuk mengetahui kinerja *hardware, software* maupun sistem pada alat secara keseluruhan yang dilakukan dengan melihat tingkat keberhasilan sistem. Keberhasilan dari sistem didapat dengan menghitung jumlah selisih volume antara *set point* dengan volume *Output* dibagi dengan banyak percobaan. Berikut tabel 4.9 adalah tabel data sistem keberhasilan dari sistem secara keseluruhan :

Tabel 4.9 Data Sistem Keberhasilan

Percobaan ke-	Set point (ml)	Output (ml)	ΔV (ml)	Warna Campuran Manual	Warna Hasil
1	200	230	30	<input type="checkbox"/>	<input type="checkbox"/>
2		215	15		
3		220	20		
4		238	38		
5		210	10		
1	200	200	0	<input type="checkbox"/>	<input type="checkbox"/>
2		220	20		
3		200	0		
4		205	5		
5		197	3		
1	200	195	5	<input type="checkbox"/>	<input type="checkbox"/>
2		210	10		
3		220	20		
4		210	10		
5		100	0		
$\sum \Delta V$ (ml)			186		
$\overline{\Delta V}$ (ml)			12,4		

Dari data tabel 4.9 dapat diketahui *nilai* keberhasilan sistem secara keseluruhan dengan menghitung nilai rata-rata dari selisih volume *set point* dengan volume yang dihasilkan dan rancang bangun alat pencampur cat tembok otomatis berbasis *personal computer* dapat bekerja secara otomatis dengan rata-rata selisih volume *set point* terhadap volume *output* yaitu 12,4 ml.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

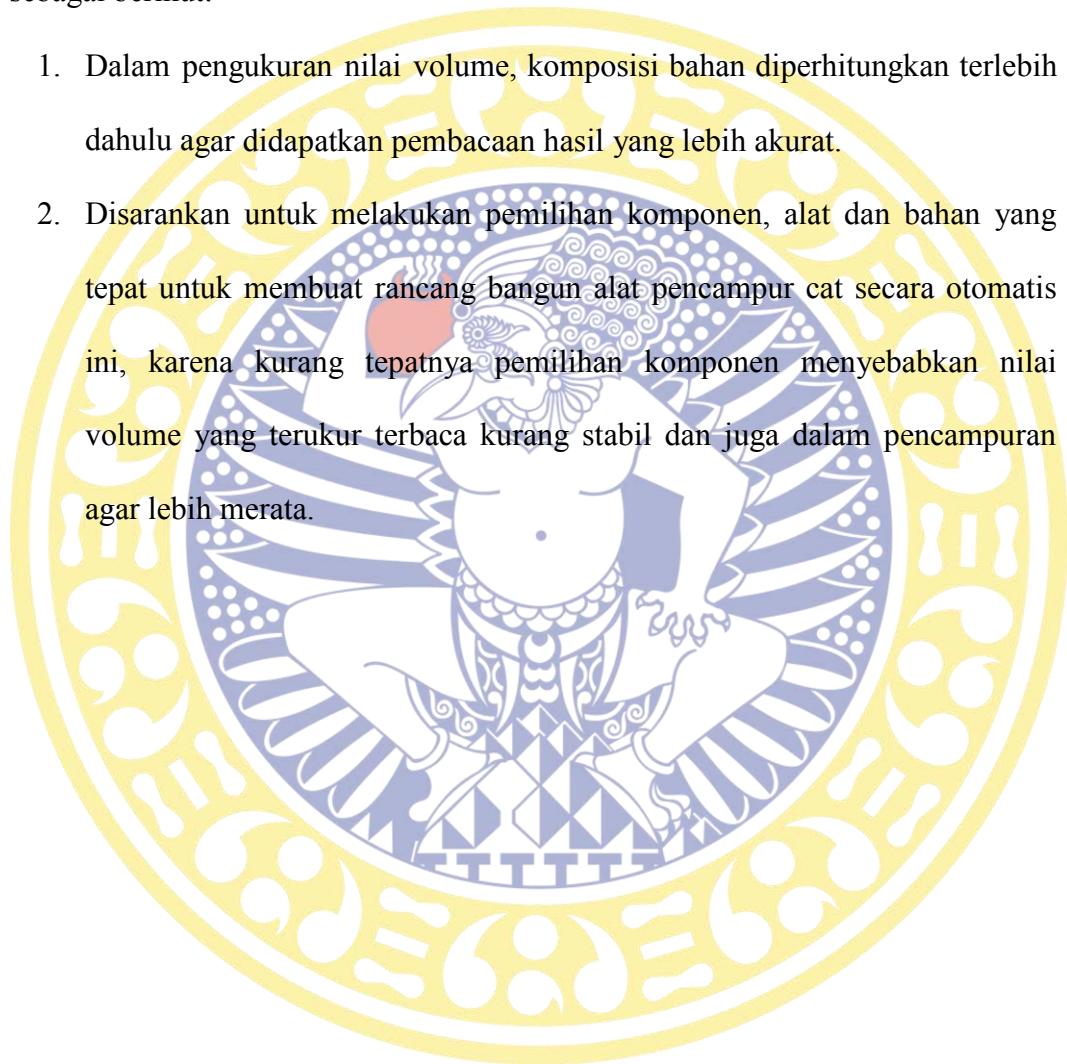
Dari kegiatan pengujian Tugas Akhir dengan judul Rancang Bangun Pencampur Cat Tembok Otomatis Berbasis PC yang sudah dilakukan maka dapat diambil kesimpulan sebagai berikut:

1. *Software* dalam sistem pencampur cat tembok otomatis mampu mengendalikan kinerja aktuator naik/turun, motor buka/tutup *valve* dan motor pengaduk dalam melakukan pencampuran cat.
2. Tugas akhir dirancang dengan menggunakan program arduino dan delphi 7. Program arduino digunakan untuk konversi perhitungan pulsa menjadi debit pada *flowmeter* serta kendali motor buka/tutup *valve*, pengaduk dan aktuator naik/turun. Program delphi 7 digunakan untuk *interface* dan konversi perhitungan pilihan komposisi warna.
3. *Software* dalam sistem pencampur cat tembok otomatis mampu melakukan pencampuran cat sesuai dengan nilai *set point* dengan pemberian *delay* pada buka/tutup *valve* didapatkan volume yang sesuai dengan yang ditentukan serta mampu mencampur warna cat sesuai yang dikehendaki. Dengan hasil $\overline{\Delta V}$ pada *flowmeter* merah yaitu 29,5 ml; $\overline{\Delta V}$ pada *flowmeter* hijau yaitu 31,5 ml; $\overline{\Delta V}$ pada *flowmeter* biru yaitu 23,1 ml.

5.2 Saran

Penulis mengharapkan agar rancang bangun alat pencampur cat tembok otomatis berbasis PC bisa dikembangkan sehingga lebih baik lagi dalam hal pengontrolan volume. Beberapa saran yang dapat penulis sampaikan adalah sebagai berikut:

1. Dalam pengukuran nilai volume, komposisi bahan diperhitungkan terlebih dahulu agar didapatkan pembacaan hasil yang lebih akurat.
2. Disarankan untuk melakukan pemilihan komponen, alat dan bahan yang tepat untuk membuat rancang bangun alat pencampur cat secara otomatis ini, karena kurang tepatnya pemilihan komponen menyebabkan nilai volume yang terukur terbaca kurang stabil dan juga dalam pencampuran agar lebih merata.



DAFTAR PUSTAKA

- Evana, Yoseph; Fany Indriaty; dan Hugeng. 2008. Sistem Pencampuran Cat Menggunakan Mikrokontroler Dengan Interface PC. Vol 10, No.2 . Jakarta.
- Winarno. 2015. “Petunjuk Praktikum Perangkat Antar Muka 1”. Surabaya : Universitas Airlangga.
- Pambudi, Siswo. 2007. “Rancang Bangun Sistem Elektromekanik Pada Otomasi Alat Pencampur Cat Menggunakan Mikrokontroler AT89S51” Tugas Akhir D-3, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Diponegoro, Semarang.
- Syahrul; Ryan Saputra. “Rancang Bangun Pencampur Cat Otomatis Berbasis Mikrokontroler”. Bandung : Teknik Komputer Unikom.
- Pratikto, Aryanto Hari. 2009. “Prototype Pencampur Warna Otomatis Berbasis Mikrokontroler AT89S51”. Surakarta : Universitas Muhammadiyah.
- Gumelar, Ramdan; Ade Gafar Abdullah; Maman Somantri. 2013. “Simulator Sistem Pencampur Warna Otomatis Berbasis PLC Terintegrasi Human Machine Interface”. VOL.12, NO.2, 115- 126. Bandung : Teknik Elektro FPTK UPI.
- Fadlilah, Umi; Fatah Yasin Al Irsyadi; Aryanto Hari Pratikto. 2009. “Prototype Pencampur Warna Otomatis Berbasis Mikrokontroler AT89S51”. Surakarta : Teknik Elektro, Universitas Muhammadiyah.
- Nuswantara, Satria M. 2009. “Perancangan Proses Otomatis Pada Sistem Kontrol Servo *Valve* Untuk Pencampuran *Fluida* Warna Berbasis Mikrokontroler”. Semarang : Teknik Elektro, Universitas Diponegoro.
- Syahid, Andre Afrilian, Dedi Tri Widodo, Hizkia Pandhega, Rama Permata Senja. “Rancang Bangun Miniatur Pencampuran Warna Primer Cat Menjadi Warna Sekunder Secara Otomatis Berbasis PLC dan SCADA”. Semarang : Politeknik Negeri Semarang.

LAMPIRAN

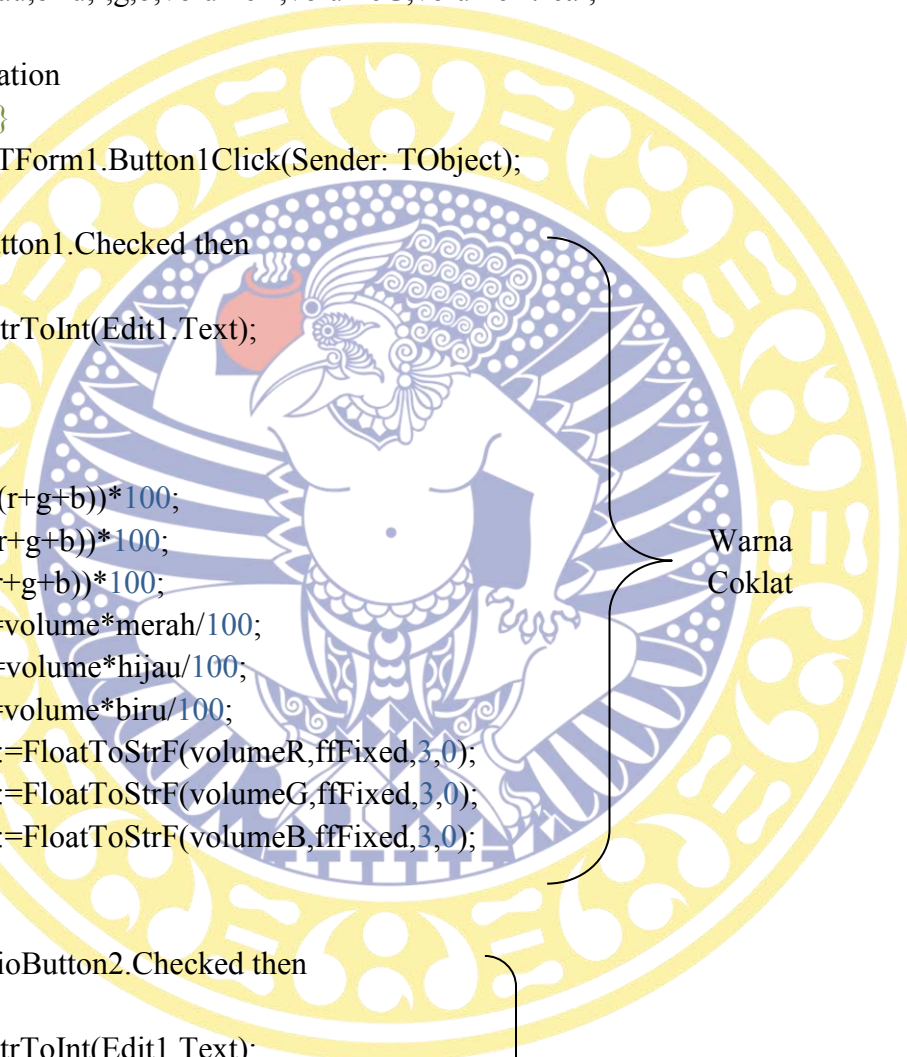
Program Pada Delphi :

```

var
  Form1: TForm1;
  volume: integer;
  merah, hijau, biru, r, g, b, volumeR, volumeG, volumeB: real;

implementation
  {$R *.dfm}
  procedure TForm1.Button1Click(Sender: TObject);
  begin
    if RadioButton1.Checked then
    begin
      volume:=StrToInt(Edit1.Text);
      r:=255;
      g:=255;
      b:=0;
      merah:=(r/(r+g+b))*100;
      hijau:=(g/(r+g+b))*100;
      biru :=(b/(r+g+b))*100;
      volumeR:=volume*merah/100;
      volumeG:=volume*hijau/100;
      volumeB:=volume*biru/100;
      Edit2.Text:=FloatToStrF(volumeR,ffFixed,3,0);
      Edit3.Text:=FloatToStrF(volumeG,ffFixed,3,0);
      Edit4.Text:=FloatToStrF(volumeB,ffFixed,3,0);
    end

    else if RadioButton2.Checked then
    begin
      volume:=StrToInt(Edit1.Text);
      r:= 0;
      g:=255;
      b:=255;
      merah:=(r/(r+g+b))*100;
      hijau:=(g/(r+g+b))*100;
      biru :=(b/(r+g+b))*100;
  
```



Warna Coklat

Warna Biru Laut


```

volumeR:=volume*merah/100;
volumeG:=volume*hijau/100;
volumeB:=volume*biru/100;
Edit2.Text:=FloatToStrF(volumeR,ffFixed,3,0);
Edit3.Text:=FloatToStrF(volumeG,ffFixed,3,0);
Edit4.Text:=FloatToStrF(volumeB,ffFixed,3,0);
end

```

} Warna Biru Laut

```

else if RadioButton3.Checked then
begin
volume:=StrToInt(Edit1.Text);
r:=255;
g:= 0;
b:=255;
merah:=(r/(r+g+b))*100;
hijau:=(g/(r+g+b))*100;
biru :=(b/(r+g+b))*100;
volumeR:=volume*merah/100;
volumeG:=volume*hijau/100;
volumeB:=volume*biru/100;
Edit2.Text:=FloatToStrF(volumeR,ffFixed,3,0);
Edit3.Text:=FloatToStrF(volumeG,ffFixed,3,0);
Edit4.Text:=FloatToStrF(volumeB,ffFixed,3,0);
end
end;

```

} Warna Magenta / Ungu

```

procedure TForm1.Button2Click(Sender: TObject);
begin
//kirim data serial
SerialPortNG1.Active:=True;
SerialPortNG1.SendString(FloatToStrF(volumeR,ffFixed,3,0)+''+FloatToStrF(vo
lumeG,ffFixed,3,0)+''+FloatToStrF(volumeB,ffFixed,3,0)+'#+#10);
end;
end.

```

LAMPIRAN

Hasil Rancang Bangun Alat Pencampur Cat Tembok Otomatis Berbasis PC



LAMPIRAN

Program Arduino :

```

#define relay1 7
#define relay2 6
#define relay3 5
#define pengaduk 4
#define naik 9
#define turun 8

```

} define output

```

byte sensorInterrupt = 0; //0=digital pin 2
byte sensorInterrupt1 = 1; //1=digitalpin3
byte sensorInterrupt2 = 2; //2=digitalpin21
byte sensorPin = 2;
byte sensorPin1 = 3;
byte sensorPin2 = 21;

```

} deklarasi pin interupt

```

float calibrationFactor = 4.5;
float calibrationFactor1 = 4.5;
float calibrationFactor2 = 4.5;

```

} kalibrasi flowmeter

```

volatile byte pulseCount;
volatile byte pulseCount1;
volatile byte pulseCount2;

```

} variabel byte pulsecount

```

float flowRate;
float flowRate1;
float flowRate2;
unsigned int flowMilliLitres;
unsigned int flowMilliLitres1;
unsigned int flowMilliLitres2;
unsigned long totalMilliLitres;
unsigned long totalMilliLitres1;
unsigned long totalMilliLitres2;

```

} flowrate bernilai pecahan tipe data yang digunakan float
tipe data agar tidak bernilai negatif

```

unsigned long oldTime;
unsigned long oldTime1;
unsigned long oldTime2;

```

```

String dataIn;
String dt[3]; //tipe data array string dan jumlah array 3
int i;
boolean parsing=false; //tidak ada parsing data
int langkah=0;
int volumeR;
int volumeG;
int volumeB;

```

} deklarasi nilai awal parsing data
deklarasi nilai R,G,B


```

void setup()
{
  Serial.begin(9600); //komunikasi serial
  dataIn="";
  pinMode(10,INPUT_PULLUP); // ls1
  pinMode(11,INPUT_PULLUP); // ls2
  pinMode(7,OUTPUT);
  pinMode(6,OUTPUT);
  pinMode(5,OUTPUT);
  pinMode(4, OUTPUT);
  pinMode(8, OUTPUT);
  pinMode(9, OUTPUT);
  digitalWrite(pengaduk,LOW);
  digitalWrite(naik,HIGH);
  digitalWrite(turun,HIGH);
  digitalWrite(7,LOW);
  digitalWrite(6,LOW);
  digitalWrite(5,LOW);

  pinMode(sensorPin, INPUT);
  digitalWrite(sensorPin, HIGH);
  pinMode(sensorPin1, INPUT);
  digitalWrite(sensorPin1, HIGH);
  pinMode(sensorPin2, INPUT);
  digitalWrite(sensorPin2, HIGH);

  pulseCount = 0;
  flowRate = 0.0;
  flowMilliLitres = 0;
  totalMilliLitres = 0;
  oldTime = 0;

  pulseCount1 = 0;
  flowRate1 = 0.0;
  flowMilliLitres1 = 0;
  totalMilliLitres1 = 0;
  oldTime1 = 0;

  pulseCount2 = 0;
  flowRate2 = 0.0;
  flowMilliLitres2 = 0;
  totalMilliLitres2 = 0;
  oldTime2 = 0;

  attachInterrupt(sensorInterrupt, pulseCounter, FALLING);
  attachInterrupt(sensorInterrupt1, pulseCounter1, FALLING);
  attachInterrupt(sensorInterrupt2, pulseCounter2, FALLING); }

```

deklarasi pin dan kondisi awal

baca sensorPin flowmeter

flowmeter1

flowmeter2

nilai awal flowmeter

flowmeter3

konfigurasi trigger sehingga kondisi pulsa berubah dari HIGH ke LOW

```

void loop()
{ int ls_bwh = digitalRead(10);
  int ls_ats = digitalRead(11); } limit switch

if(langkah==0){
  if(Serial.available()>0)
  {
    char inChar = (char)Serial.read(); //baca serial
    dataIn += inChar;
    if (inChar == '\n')
      {parsing = true;}
    }
  if(parsing)
  {parsingData();
  parsing=false;
  dataIn="";
  langkah=1; } }

if(langkah==1){volumeR=dt[0].toInt();volumeG=dt[1].toInt();volumeB=dt[2].toInt();langkah=2;}
// data volume R,G,B

if(langkah==2){digitalWrite(turun,LOW);
  if(ls_bwh==LOW){langkah=3;}}
if(langkah==3){digitalWrite(turun,HIGH);delay(2000);langkah=4;}
if(langkah==4){digitalWrite(relay1,HIGH);delay(200);langkah=5;}
if(langkah==5){digitalWrite(relay1,LOW);langkah=6;}

if(langkah==6){
  if((millis() - oldTime) > 1000)
  { detachInterrupt(sensorInterrupt);
    flowRate = ((1000.0 / (millis() - oldTime)) * pulseCount) / calibrationFactor;
    oldTime = millis();
    flowMilliLitres = (flowRate / 60) * 1000;
    totalMilliLitres += flowMilliLitres;
  }
  unsigned int frac;
  Serial.print("Flow rate: ");
  Serial.print(int(flowRate));
  Serial.print("."); // Print the decimal point
  frac = (flowRate - int(flowRate)) * 10;
  Serial.print(frac, DEC);
  Serial.print("L/min");
  Serial.print(" Current Liquid Flowing: ");
  Serial.print(flowMilliLitres);
  Serial.print("mL/Sec");
  Serial.print(" Output Liquid Quantity: ");
  Serial.print(totalMilliLitres);
  Serial.println("mL");
}

```

langkah 0 baca data serial

aktuator turun ON

valve 1 ON membuka

menampilkan Flowrate(debit)

menampilkan MilliLitres (volume/detik)

menampilkan Total keseluruhan

```

// Reset the pulse counter so we can start incrementing again
pulseCount = 0;
// Enable the interrupt again now that we've finished sending output
attachInterrupt(sensorInterrupt, pulseCounter, FALLING);

if(totalMilliLitres>=volumeR){langkah=7;} //volume R
}
if(langkah==7){digitalWrite(relay1,HIGH);delay(200);langkah=8;} } valve 1ON Tutup
if(langkah==8){digitalWrite(relay1,LOW);langkah=9;}

if(langkah==9){digitalWrite(relay2,HIGH);delay(200);langkah=10;} } valve 2 Buka
if(langkah==10){digitalWrite(relay2,LOW);langkah=11;}

if(langkah==11){
if(millis() - oldTime1) > 1000)
{ detachInterrupt(sensorInterrupt1);
flowRate1 = ((1000.0 / (millis() - oldTime1)) * pulseCount1) / calibrationFactor1;
oldTime1 = millis();
flowMilliLitres1 = (flowRate1 / 60) * 1000;
totalMilliLitres1 += flowMilliLitres1;
unsigned int frac1;

Serial.print ("Flow rate 2: ");
Serial.print (int(flowRate1));
Serial.print (" ");
frac1 = (flowRate1 - int(flowRate1)) * 10;
Serial.print (frac1, DEC) ;
Serial.print ("L/min");
Serial.print (" Current Liquid Flowing 2: ");
Serial.print (flowMilliLitres1);
Serial.print ("mL/Sec");
Serial.print (" Output Liquid Quantity 2: ");
Serial.print (totalMilliLitres1);
Serial.println ("mL");
} menampilkan Flowrate(debit)
} menampilkan MilliLitres (volume/detik)
} menampilkan Total keseluruhan

// Reset the pulse counter so we can start incrementing again
pulseCount1 = 0;
// Enable the interrupt again now that we've finished sending output
attachInterrupt(sensorInterrupt1, pulseCounter1, FALLING);

if(totalMilliLitres1>=volumeG){langkah=12;} //volume G
}
if(langkah==12){digitalWrite(relay2,HIGH);delay(200);langkah=13;} } valve 2 ON Tutup
if(langkah==13){digitalWrite(relay2,LOW);langkah=14;}

if(langkah==14){digitalWrite(relay3,HIGH);delay(200);langkah=15;} } valve 3 ON Buka
if(langkah==15){digitalWrite(relay3,LOW);langkah=16;}

```



```

if(langkah==16){
if((millis() - oldTime2) > 1000)
{ detachInterrupt(sensorInterrupt2);
  flowRate2 = ((1000.0 / (millis() - oldTime2)) * pulseCount2) / calibrationFactor2;
  oldTime2 = millis();
  flowMilliLitres2 = (flowRate2 / 60) * 1000;
  totalMilliLitres2 += flowMilliLitres2;
  unsigned int frac2;

  Serial.print ("Flow rate 3: ");
  Serial.print (int(flowRate2));
  Serial.print (" ");
  frac2 = (flowRate2 - int(flowRate2)) * 10;
  Serial.print (frac2, DEC) ;
  Serial.print ("L/min");
  Serial.print (" Current Liquid Flowing 3: ");
  Serial.print (flowMilliLitres2);
  Serial.print ("mL/Sec");
  Serial.print (" Output Liquid Quantity 3:");
  Serial.print (totalMilliLitres2);
  Serial.println("mL");

  // Reset the pulse counter so we can start incrementing again
  pulseCount2 = 0;
  // Enable the interrupt again now that we've finished sending output
  attachInterrupt(sensorInterrupt2, pulseCounter2, FALLING);

  if(totalMilliLitres2>=volumeB){langkah=17;} //volume B
  }
  if(langkah==17){digitalWrite(relay3,HIGH);delay(200);langkah=18;}
  if(langkah==18){digitalWrite(relay3,LOW);delay(3000);langkah=19;}

  if(langkah==19){digitalWrite(pengaduk,HIGH);delay(35000);langkah=20;} //pengaduk ON
  if(langkah==20){digitalWrite(pengaduk,LOW);delay(5000);langkah=21;} //pengaduk OFF
  if(langkah==21){digitalWrite(naik,LOW);
    if(!s_atn==LOW){langkah=22;} }
  if(langkah==22){digitalWrite(naik,HIGH);langkah=0;}
  }

void pulseCounter()
{ pulseCount++; }
void pulseCounter1()
{ pulseCount1++; }
void pulseCounter2()
{ pulseCount2++; }

```

menampilkan Flowrate(debit)

menampilkan MilliLitres (volume/detik)

menampilkan Total keseluruhan

valve 3 ON Tutup

aktuator naik ON

interrupt service routine

```
void parsingData()
{
    int j=0; //inisialisasi variabel, (reset isi variabel)
    dt[j]="";
    for(i=0;i<dataIn.length();i++)
    {
        //pengecekan tiap karakter dengan karakter (#) dan (,)
        if((dataIn[i] == '#') || (dataIn[i] == ','))
        { j++; //increment variabel j, digunakan untuk merubah index array penampung
          dt[j]=""; //inisialisasi variabel array dt[j]
        }
        else
        { dt[j] = dt[j] + dataIn[i]; } //proses tampung data saat pengecekan karakter selesai.
    }
}
```

