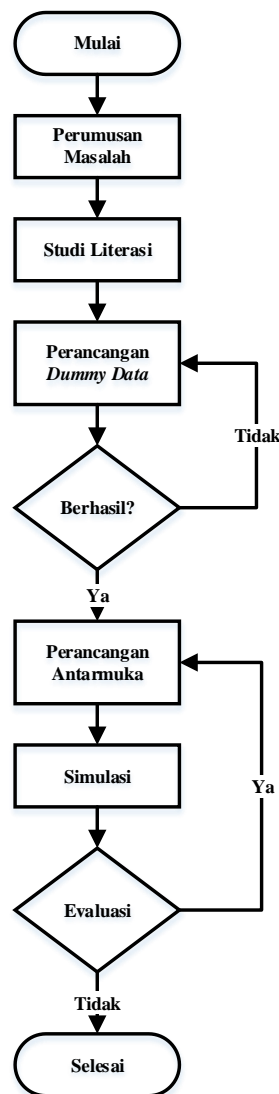


BAB III

METODE PENELITIAN

Dalam pengerjaan tugas akhir ini, dibutuhkan sebuah metode penelitian untuk menentukan langkah-langkah yang perlu dilakukan. Selain itu, metode penelitian ini akan mempermudah pembaca dalam memahami proses pengerjaan tugas akhir ini. Adapun langkah-langkah yang perlu dilakukan adalah sebagai berikut.



Gambar 3.1 Diagram alir metode penelitian
(Dibuat pada 31 Maret 2019 pukul 08.00)

Gambar 3.1 menjelaskan bagaimana pengerjaan tugas akhir dilakukan. Diawali dari mengangkat permasalahan yang ada, kemudian studi literasi, analisis dan pengembangan, simulasi, dan evaluasi. Proses evaluasi menentukan selesai atau tidaknya pengerjaan tugas akhir. Ketika masih banyak evaluasi yang belum terpecahkan, maka kembali ke proses analisis dan pengembangan. Hal ini akan berulang terus hingga evaluasi mencapai apa yang diinginkan dalam tugas akhir ini.

3.1 PERUMUSAN MASALAH

Langkah yang paling pertama adalah perumusan masalah. Hal ini dilakukan di awal, yaitu sebelum penulisan proposal tugas akhir. Permasalahan yang diangkat telah disampaikan di awal bab, pada bagian Rumusan Masalah. Dalam perumusan masalah, dilakukan pembatasan permasalahan yang diuraikan untuk memudahkan fokus pengerjaan pada tugas akhir.

3.2 STUDI LITERASI

Dalam proses tugas akhir, pasti membutuhkan proses mempelajari literasi apa saja yang diperlukan dalam melakukan penelitian. Proses ini dilakukan mulai dari awal perumusan masalah hingga penerapan ilmu pada perancangan dan pengembangan. Studi literasi dilakukan melalui banyak sumber, antara lain: buku, buku elektronik, jurnal, dan internet.

3.3 PERANCANGAN DAN PENGEMBANGAN

Perancangan dan pengembangan dilakukan menggunakan piranti lunak dan piranti keras, tetapi lebih dominan pada piranti lunak karena hal tersebut merupakan topik dari tugas akhir ini. Sebelum masuk ke dalam penjelasan mengenai perancangan dan pengembangan tugas akhir, berikut ini adalah alat dan bahan yang digunakan dalam perancangan tugas akhir ini.

1. Laptop Toshiba Satellite C50-B, Intel Core i3 1,8 MHz RAM 4 GB.
Sistem Operasi: Windows 8.1 SP 1.

2. Arduino UNO.
3. Arduino IDE.
4. LabVIEW 2017.

3.3.1 PERANCANGAN *DUMMY DATA*

Pada piranti keras dilakukan pemrograman untuk menghasilkan *dummy data*. Pemrograman tersebut dilakukan menggunakan fungsi yang menghasilkan data acak, sehingga dihasilkan data ideal. Data ideal ini dirancang berdasarkan format data yang dihasilkan dari pengukuran sensor. Berikut ini adalah prosedur dalam perancangan *dummy data* menggunakan Arduino.



Gambar 3.2 Diagram alur perancangan dummy data menggunakan Arduino
(Dibuat pada 1 Maret 2019 pukul 08.32)

1. Membuka aplikasi Arduino IDE pada laptop.
2. Membuat program pada aplikasi Arduino IDE dengan cara mengetikkan kode program, dimulai dari inisialisasi atau deklarasi variabel, pengaturan serial dan pin Arduino, hingga program utama (Kode program terlampir).

```

randomSeed | Arduino 1.8.5
File Edit Sketch Tools Help
randomSeed
double randTemp, randPress, randAlt, randG, randC, randHD;

void setup() {
  Serial.begin(9600);
  randomSeed(analogRead(0));
}

void loop() {
  randTemp = random(50)+(random(10)/100.0)*10+random(10)/100.0;
  randPress = random(100)+(random(10)/100.0)*10+random(10)/100.0;
  randAlt = random(100)+(random(10)/100.0)*10+random(10)/100.0;
  randG = random(-120, 120)+(random(10)/100.0)*10+random(10)/100.0;
  randC = random(-120, 120)+(random(10)/100.0)*10+random(10)/100.0;
  randHD = random(-359, 359)+(random(10)/100.0)*10+random(10)/100.0;

  // String Data[] = {"t", randTemp, "p", randPress, "a", randAlt,
  // "accy", randG, "accz", randG, "lo", randAlt, "la", randAlt, "c",
  // "cy", randC, "cz", randC, "gx", randG, "gy", randG, "gz", randG
}
  
```

Gambar 3.3 Tampilan program dummy data pada Arduino IDE
(Diambil pada 28 Februari 2019 pukul 11.22)



3. Pengaturan pin dibuat menggunakan fungsi *randomSeed()*, sedangkan program utama menggunakan fungsi *random()* untuk menghasilkan angka acak pada setiap format data. Format datanya adalah:

```
t%fp%fa%faccx%faccy%faccz%flo%fla%fcx%fcy%fcz%fgx%fgy%fgz%fhd%f
```

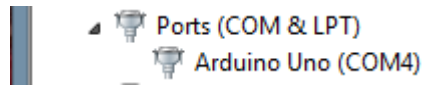
Format data tersebut diperoleh dari format pembacaan sensor yang digunakan pada penelitian sebelumnya. Hal ini kemudian dijadikan acuan dalam membuat data acak sebagai *dummy data*. Berikut ini adalah Tabel 4 yang memberikan keterangan atas format tersebut:

Tabel 4. Tabel keterangan format data untuk *dummy data*.

Format data	Penjelasan
t	Awalan untuk data suhu
p	Awalan untuk data tekanan
a	Awalan untuk data ketinggian
accx	Awalan untuk data akselero (x)
accy	Awalan untuk data akselero (y)
accz	Awalan untuk data akselero (z)
lo	Awalan untuk data garis lintang
la	Awalan untuk data garis bujur
cx	Awalan untuk data kompas (x)
cy	Awalan untuk data kompas (y)
cz	Awalan untuk data kompas (z)
gx	Awalan untuk data giro (x)
gy	Awalan untuk data giro (y)
gz	Awalan untuk data giro (z)
hd	Awalan untuk data sudut kepala
%f	Data acak yang dihasilkan

4. Setelah program selesai, kemudian program dicek terlebih dahulu, apakah masih ada kesalahan dari segi logika maupun format kode. Kemudian, ikon  (*Verify*) diklik.
5. Jika sudah keluar tulisan “*Done compiling*” pada bagian bawah Arduino IDE, maka program siap untuk diunggah ke Arduino UNO. Kemudian, ikon  (*Upload*) diklik. Namun, jika masih muncul tulisan selain “*Done compiling*”, maka kode program perlu dicek kembali untuk dikoreksi kesalahannya dan kembali ke langkah ke 4.

6. Sebelum program diunggah ke Arduino, dipastikan dulu Arduino telah terhubung ke laptop melalui Serial Port. Hal ini dapat dicek melalui *Run -> Device Manager -> Ports*.



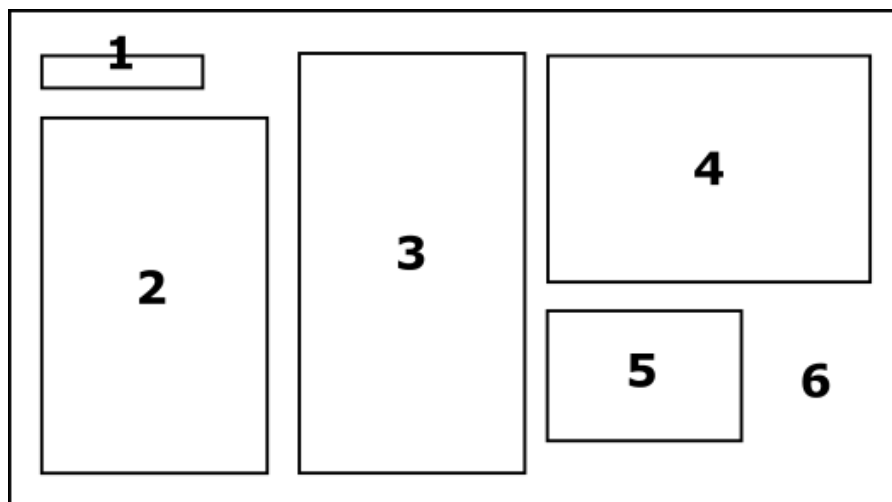
Gambar 3.4 Tampilan Arduino telah terhubung dengan laptop
(Diambil pada 28 Februari 2019 pukul 12.32)

Gambar di atas (Gambar 3.4) menunjukkan bahwa Arduino UNO telah terhubung dengan laptop melalui komunikasi serial dengan alamat COM4. Jika Arduino UNO belum terhubung ke laptop, maka laptop perlu dipasang *USB driver* Arduino terlebih dahulu.

7. Setelah program diunggah ke Arduino UNO, *Serial Monitor* dibuka untuk memastikan data telah terkirim ke serial atau belum.

3.3.2 PERANCANGAN ANTARMUKA

Dalam perancangan antarmuka (LabVIEW), diawali dari perancangan tampilan depan antarmuka. Hal ini dapat dicontoh dari antarmuka yang telah ada dari penelitian sebelumnya. Antarmuka yang dirancang merupakan pengembangan dari antarmuka yang telah ada sebelumnya. Berikut ini adalah gambaran antarmuka (*Front Panel*) yang dirancang (lihat Gambar 3.5).



Gambar 3.5 Rencana tampilan antarmuka LabVIEW
(Dibuat pada 4 Oktober 2018 pukul 08.18)

Tampilan tersebut terbagi ke dalam 6 bagian, dimana nomor 1 hingga 5 merupakan objek. Objek yang dimaksud adalah *Controls* yang tersedia pada LabVIEW. Berikut ini keterangan komponen pada Gambar 3.5.

1. Pengaturan port komunikasi serial;
2. Indikator keluaran hasil akhir proses;
3. Indikator data per satuan paket;
4. Indikator data per segmen *string* dan waktu proses;
5. Indikator galat pada proses akhir;
6. Panel kendali.


Selain dari perancangan tampilan antarmuka, tentu dirancang pula blok diagram yang akan digunakan untuk melakukan proses pengolahan data. Blok diagram dirancang berdasarkan kebutuhan yang ada, yaitu kemungkinan kesalahan yang dibuat oleh piranti keras. Berikut ini adalah alur kerja dari blok diagram yang dirancang.



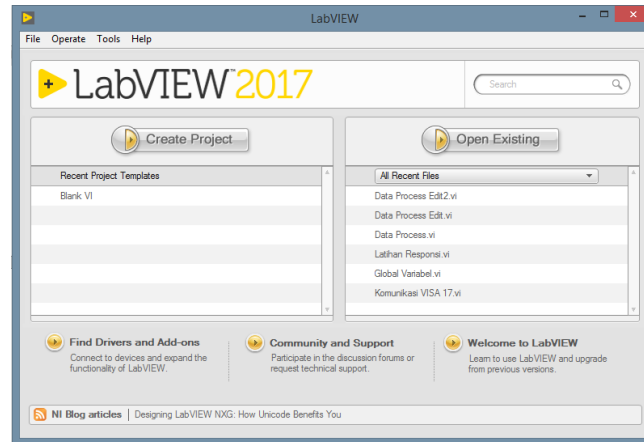
Gambar 3.6 Diagram alur perancangan blok diagram LabVIEW
(Dibuat pada 1 Maret 2019 pukul 08.45)

3.4 SIMULASI



Simulasi dilakukan setelah perancangan dan pengembangan dirasa sudah cukup siap. Tahap ini juga dapat disebut sebagai tahap uji coba antarmuka, yaitu antarmuka digunakan untuk memproses *dummy data* yang dihasilkan oleh piranti keras (mikrokontroler). Proses ini dilakukan melalui beberapa tahap, antara lain:

1. Arduino dihubungkan ke laptop.
2. Aplikasi Arduino IDE dibuka pada laptop, kemudian *Serial Monitor* dibuka dengan cara menekan ikon  (*Serial Monitor*).
3. Jika data sudah muncul pada *Serial Monitor*, maka baru bisa diuji coba ke LabVIEW.

4. NI LabVIEW 2017 dibuka, kemudian buka VI yang telah dibuat melalui bagian *Open Existing* atau di bawahnya.



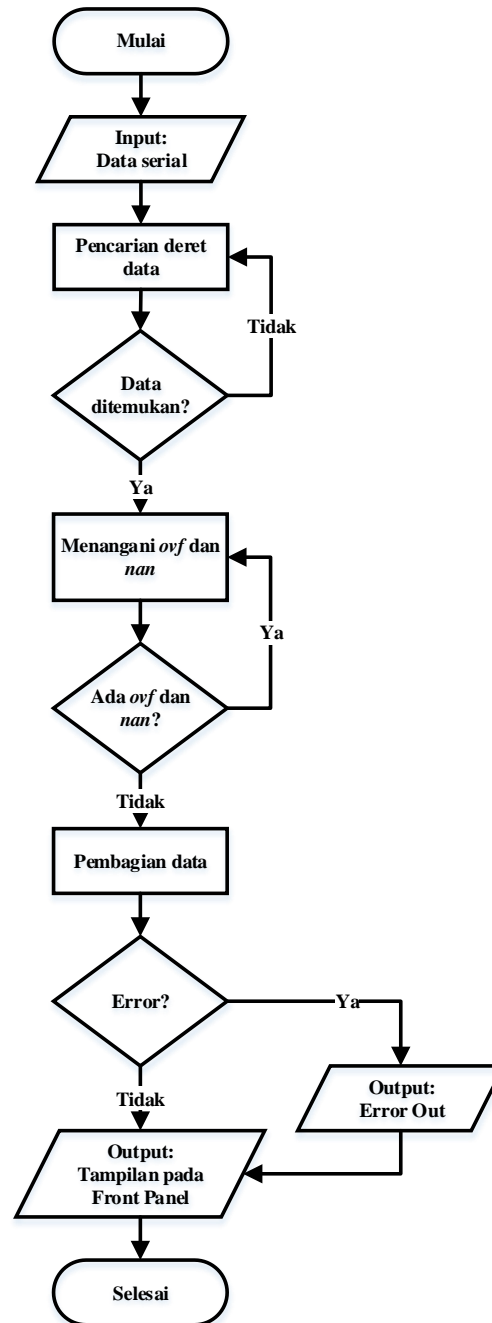
Gambar 3.7 Tampilan awal LabVIEW 2017
(Diambil pada 1 Maret 2019 pukul 10.08)

5. Apabila sudah keluar tampilan *Front Panel*, dipilih nama alamat yang tersedia (misal COM4) dari komunikasi serial antara laptop dengan Arduino. Jika belum tersedia nama alamat, maka perlu dicek kembali prosedur perancangan piranti keras.
6. Ikon  (*Run*) diklik untuk menjalankan aplikasi LabVIEW secara sekali jalan, sedangkan ikon  (*Run Continuously*) diklik untuk menjalankan aplikasi LabVIEW secara berulang.

Dalam proses pengambilan data dari hasil simulasi menggunakan LabVIEW, digunakan dua metode pengambilan data, antara lain:

1. LabVIEW dijalankan secara satu per satu beberapa kali untuk mendapatkan data yang bersifat kuantitatif. Kemudian, dari sejumlah percobaan yang dilakukan tersebut nantinya dilihat berapa besar akurasi yang dihasilkan dari antarmuka.
2. LabVIEW dijalankan secara kontinyu dalam interval waktu tertentu untuk mendapatkan data bersifat kualitatif. Hal ini digunakan sebagai indikator kehandalan dari antarmuka yang dibuat.

Untuk mempermudah dalam memahami langkah-langkah simulasi dan metode pengambilan data, berikut ini disajikan diagram alir melalui Gambar 3.8. Diagram alir tersebut menggambarkan proses data yang terjadi pada proses simulasi.



Gambar 3.8 Diagram alir simulasi LabVIEW
(Dibuat pada 20 Maret 2019 pukul 07.45)

Setelah dilakukan langkah-langkah di atas, maka proses simulasi telah selesai dan diperoleh hasil. Jika pada proses simulasi masih terjadi kesalahan dan belum diperoleh hasil, maka dilakukan prosedur ulang terhadap simulasi. Selanjutnya, data yang telah diperoleh dari simulasi tersebut akan dievaluasi.

3.5 EVALUASI

Setelah simulasi dilakukan, maka hasil yang diperoleh dari simulasi tersebut perlu untuk dievaluasi. Pada tahap evaluasi, data hasil dianalisis dan didiskusikan bersama dosen pembimbing. Hasil analisis tersebut kemudian dinilai, apakah sudah menjawab tujuan dari tugas akhir atau belum. Proses evaluasi dilakukan secara kualitatif dan kuantitatif. Kualitatif yaitu menilai kehandalan dari antarmuka yang telah dibuat, sedangkan kuantitatif yaitu melihat jumlah kesalahan yang terjadi dari beberapa kali pengambilan data secara satu per satu. Berikut ini adalah tahap-tahap evaluasi.

1. Pengecekan *Dummy data* yang ditampilkan pada *Serial Monitor*, apakah data sudah bisa tampil dan berjalan normal.
2. Antarmuka diidentifikasi, yaitu dengan membandingkan hasil dengan format yang sudah dirancang.
3. LabVIEW dijalankan, kemudian dilihat waktu yang dibutuhkan dalam sebuah proses *tracing data*.
4. LabVIEW diidentifikasi, apakah ada error yang menyebabkan LabVIEW berhenti berjalan.
5. Jika ada error selain di atas, yaitu error dalam proses *parsing data*, cukup dilihat posisi errornya dimana.

Jika hasil tersebut masih jauh dari tujuan yang ingin dicapai, maka proses penelitian akan kembali ke bagian perancangan dan pengembangan. Namun, jika hasil tersebut sudah sesuai dengan tujuan yang ingin dicapai, maka proses selanjutnya adalah pembuatan laporan akhir.