

## BAB I PENDAHULUAN

### 1.1 Latar Belakang

Semakin berkembangnya teknologi saat ini memberikan dampak positif pada perkembangan dunia elektronik khususnya di bidang sistem kontrol. Pada saat ini banyak teknologi yang telah menggunakan alat otomatisasi kontrol. Hal tersebut disebabkan oleh keinginan manusia untuk memenuhi kebutuhan dan mendapatkan fasilitas kemudahan dan efisiensi dalam berbagai macam hal.

*flyboard* merupakan olahraga baru di Indonesia yang dengan cepat menarik perhatian masyarakat. *flyboard* sendiri adalah sebuah jenis jet pack air yang melekat pada *personal watercraft* (PWC) atau wahana yang melekat pada kaki pengguna yang selanjutnya dipasok air bertekanan tinggi sehingga pengguna dapat terbang di udara akibat dorongan air tersebut, olahraga ini dikenal sebagai *flyboarding*. Seorang pengendara *flyboard* berdiri di atas papan yang dihubungkan dengan selang panjang yang terhubung ke mesin jetski. Air bertekanan tinggi untuk sepasang sepatu bot dengan nozel jet bawah yang memberikan dorongan bagi pengendara untuk terbang hingga mampu naik mencapai 15 meter di udara.

Sehubungan dengan pengontrolan keseimbangan *flyboard* yang dilakukan secara manual oleh pemain sehingga sulit bagi pengguna untuk mendapatkan keseimbangan awal yang stabil, terutama bagi pengguna baru dan wisatawan yang tidak berpengalaman sama sekali, selain itu juga melelahkan bagi operator karena harus mengulangi permainan secara terus menerus yang diakibatkan oleh seringnya pemain terjatuh ke air dikarenakan kesulitan untuk menyeimbangkan diri dengan baik, hal ini tentu sangat merugikan pengguna karena cukup banyak waktu yang terbuang untuk mempelajari penggunaan *flyboard* padahal biaya sewa *flyboard* itu sendiri cukup mahal untuk waktu yang singkat. maka dari itu penulis merancang sebuah alat pengontrol keseimbangan *flyboard* yang dapat digunakan untuk memepertahakan keseimbangan *flyboard* secara otomatis dan stabil sehingga memudahkan pengendalian serta menstabilkan posisi *flyboard* sesuai keinginan pengguna. Penelitian ini menerapkan sistem pengendalian tekanan air pada miniatur *flyboard* menggunakan kontroler PID (*Proportional Integral*

*Derivatif*) yang diharapkan bisa menghasilkan suatu sistem sesuai dengan *setpoint* yang ditentukan.

### 1.2 Rumusan Masalah

1. Bagaimana merancang *hardware* sistem kontrol pengendalian keseimbangan pada miniatur *flyboard* dengan menggunakan metode PID (*Proportional Integral Derivatif*)?
2. Bagaimana merancang *software* sistem kontrol pengendalian keseimbangan pada miniatur *flyboard* dengan menggunakan metode PID (*Proportional Integral Derivatif*)?

### 1.3 Batasan Masalah

Karena luasnya objek pengkajian maka perlu dilakukan pembatasan masalah adapun batasan masalah pada skripsi ini sebagai berikut;

- 1 *Flyboard* yang digunakan adalah dalam bentuk miniatur yang terbuat dari kardus berlapis cat agar tahan air.
  - 1) Kapasitas box kaca pada ukuran 30 cm x 100 cm dan digunakan sebagai tempat simulasi miniatur *flyboard*.
  - 2) *Setpoint* miniatur *flyboard* adalah keseimbangan sumbu x pada sudut  $180^{\circ}$ .
  - 3) Menggunakan dua buah pompa akuarium untuk mempertahankan keseimbangan posisi miniatur *flyboard*, sehingga manuver *flyboard* selain keseimbangan diabaikan.
  - 4) Pembahasan mengenai jenis air, mekanik, model matematis, serta rangkaian elektronika tidak dibahas secara mendalam.

### 1.4 Tujuan

Pada penelitian ini diharapkan dapat merancang suatu sistem kontrol kendali keseimbangan pada miniatur *flyboard* dengan menggunakan metode PID (*Proportional Integral Derivatif*) untuk mempertahankan keseimbangan sebagai *setpoint* pada sumbu x  $180^{\circ}$ .

### 1.5 Sistematika Penulisan

Agar penyusunan laporan skripsi ini dapat mencapai sasaran dan tidak menyimpang dari judul yang telah ditentukan, maka diperlukan sistematika



pembahasan yang jelas. Pembahasan dalam skripsi ini secara garis besar adalah sebagai berikut:

**BAB I           Pendahuluan**

Menjelaskan tentang latar belakang, rumusan masalah, batasan masalah, tujuan, manfaat dan sistematika penulisan.

**BAB II           Tinjauan Pustaka**

Menjelaskan teori dasar yang berisi penjelasan tentang teori pompa air aquilla P5900, sensor *accelerometer* ADXL3345, *zero crossing detector*, arduino uno, kontroller PID (*Proportional Integral Derivatif*).

**BAB III          Metodologi Penelitian**

Menjelaskan tentang metodologi penelitian yang terdiri atas studi literatur, perancangan alat, pembuatan alat, pengujian alat, serta pengambilan kesimpulan dan saran.

**BAB IV          Perancangan dan Pembuatan Alat**

Menjelaskan tentang perancangan dan pembuatan alat yang meliputi prinsip kerja alat, perancangan perangkat keras dan perangkat lunak.

**BAB V           Pengujian dan Analisis**

Menjelaskan tentang pengujian alat dan analisa yang meliputi pengujian bagian blok sistem secara keseluruhan.

**BAB VI          Penutup**

Menjelaskan tentang pengambilan kesimpulan sesuai dengan hasil perealisasiian dan pengujian alat sesuai dengan tujuan dan rumusan masalah, serta pemberian saran untuk pengembangan.



## BAB II

### TINJAUAN PUSTAKA

Dalam skripsi berjudul pengendalian keseimbangan pada miniatur *flyboard* menggunakan kontroler PID (*Proportional Integral Derivatif*) untuk mempermudah dalam memahami cara kerja pada rangkaian ataupun dasar perancangan alat ini maka diperlukan penjelasan dan uraian teori penunjang yang digunakan dalam penulisan ini. Teori penunjang yang akan dijelaskan dalam bab ini adalah sebagai berikut:

- Pompa air Aquila P5900
- Sensor *accelerometer* ADXL345
- *Zero Crossing Detector*
- Arduino Uno
- Kontroler PID (*Proportional Integral Derivatif*)

#### 2.1 Pompa Air Aquilla P5900

Pompa air aquilla P5900 berfungsi sebagai pemompa air menuju miniatur *flyboard* agar *flyboard* mendapatkan tekanan air yang bisa menimbulkan daya angkat. Pompa air aquilla P5900 sebenarnya adalah pompa air yang digunakan pada akuarium, pompa ini memiliki daya 150 watt yang mampu memompa air sebesar 5500 liter/jam dengan tinggi dorongan maksimum sejauh 5 meter.



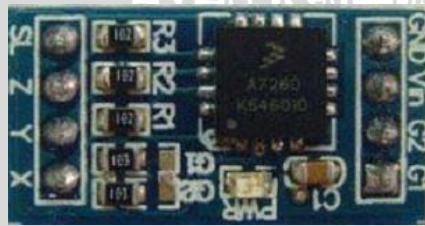
Gambar 2.1 Pompa Air Aquilla P5900



## 2.2 Sensor Accelerometer ADXL345

Accelerometer adalah sebuah *Tranduser* yang berfungsi untuk mengukur percepatan, mendeteksi dan mengukur getaran, ataupun untuk mengukur percepatan akibat gravitasi bumi. Accelerometer juga dapat digunakan untuk mengukur getaran yang terjadi pada kendaraan, bangunan, mesin, dan juga bisa digunakan untuk mengukur getaran yang terjadi di dalam bumi, getaran mesin, jarak yang dinamis, dan kecepatan dengan ataupun tanpa pengaruh gravitasi bumi.

Prinsip kerja dari *Tranduser* ini berdasarkan hukum fisika bahwa apabila suatu konduktor digerakkan melalui suatu medan magnet, atau jika suatu medan magnet digerakkan melalui suatu konduktor, maka akan timbul suatu tegangan induksi pada konduktor tersebut. Accelerometer yang diletakan di permukaan bumi dapat mendeteksi percepatan 1g (ukuran gravitasi bumi) pada titik vertikalnya, untuk percepatan yang dikarenakan oleh pergerakan horizontal maka accelerometer akan mengukur percepatannya secara langsung ketika bergerak secara horizontal. Hal ini sesuai dengan tipe dan jenis sensor Accelerometer yang digunakan karena setiap jenis sensor berbeda-beda sesuai dengan spesifikasi yang dikeluarkan oleh perusahaan pembuatnya. Saat ini hamper semua sensor/*Tranduser* accelerometer sudah dalam bentuk digital (bukan dengan sistem mekanik) sehingga cara kerjanya hanya bedasarkan temperatur yang diolah secara digital dalam satu chip. Adapun bentuk accelerometer ditunjukkan dalam Gambar 2.2.

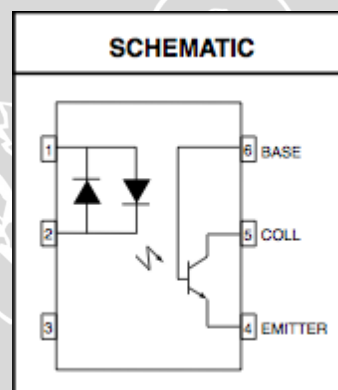


**Gambar 2.2 Sensor Accelerometer ADXL345**

### 2.3 Zero Cross Detector

Rangkaian *zero cross detector* berfungsi untuk mendeteksi titik persimpangan gelombang tegangan bolak-balik dengan titik nol. Untuk pengontrolan daya dorong pada flyboard membutuhkan rangkaian *zero cross detector* karena dengan menggunakan rangkaian tersebut dapat mengetahui referensi 0 V pada tegangan jala-jala. Sehingga nilai referensi tersebut dapat digunakan untuk mengatur sudut penyalan pada TRIAC. Hasil penyalan dengan menggunakan TRIAC berpengaruh pada tegangan yang masuk pada pompa air. Nilai tegangan yang masuk pompa air berpengaruh pada daya dorong *flyboard*.

Untuk menghasilkan sinyal *zero cross*, digunakan komponen elektronika berupa *optocoupler*. Salah satu yang dapat digunakan adalah *optocoupler* H11AA1. H11AA1 merupakan komponen elektronika yang berisi *diode*, LED, dan *transistor*. Deskripsi pin H11AA1 terlihat pada gambar 2.3.



**Gambar 2.3 Deskripsi Pin H11AA1**

Pada pin 1 adalah anoda yang nantinya dihubungkan pada sumber tegangan AC 220 V, dan pada pin 2 adalah katoda yang nantinya dihubungkan pada fasa netral dari sumber tegangan, sedangkan untuk pin tidak perlu dihubungkan pada apapun. Untuk pin 4 adalah emitor dari transistor yang nantinya dihubungkan dengan ground. Untuk pin 5 adalah kolektor dari transistor yang nantinya dihubungkan seperti pada gambar 2.25. Untuk pin 6 adalah basis dari transistor dan tidak perlu dihubungkan dengan apapun.

Prinsip kerja dari rangkaian ini adalah sumber AC dihubungkan dengan pin 1 dan netral nya dihubungkan dengan pin 2. Setelah dihubungkan arus akan melewati LED yang terdapat di dalam optocoupler dan akan membuat LED menyala dan kemudian memicu phototransistor untuk bekerja. Ketika lampu LED



menyala maka arus collector (Ic) akan meningkat secara drastis dan begitu juga sebaliknya jika LED mati maka Ic akan bernilai kecil sekali. Dengan kata lain fungsi dari phototransistor yang terdapat di dalam H11AA1 adalah berfungsi seperti saklar dengan parameter LED *on&off*.

## 2.4.1 Arduino

### 2.4.1 Pengertian Arduino UNO R3

Arduino adalah pengendali mikro *single-board* yang bersifat *open-source*, diturunkan dari *Wiring platform*, dirancang untuk memudahkan penggunaan elektronik dalam berbagai bidang. *Hardware*nya memiliki prosesor Atmel AVR dan *software*nya memiliki bahasa pemrograman sendiri. Arduino adalah kit mikrokontroler yang serba bisa dan sangat mudah penggunaannya. Untuk membuatnya diperlukan *chip programmer* (untuk menanamkan *bootloader* Arduino pada *chip*).

Arduino merupakan *single board hardware* yang *open-source* dan juga *software*nya pun dapat menikmati secara *opensource* juga. Di sisi *software* arduino dapat dijalankan di multi *platform*, yaitu linux, windows, atau juga mac. *Hardware* arduino merupakan mikrokontroler yang berbasis AVR dari ATMEL yang didalamnya sudah diberi *bootloader* dan juga sudah terdapat standart pin I/O nya. Tampilan arduino uno ditunjukkan dalam Gambar 2.4.



**Gambar 2.4 Tampilan Arduino Uno**

Arduino Uno berbeda dari semua board Arduino sebelumnya, Arduino UNO tidak menggunakan chip driver FTDI USB-to-serial. Sebaliknya, fitur-fitur



Atmega16U2 (Atmega8U2 sampai ke versi R2) diprogram sebagai sebuah pengubah USB ke serial. Revisi 2 dari board Arduino Uno mempunyai sebuah resistor yang menarik garis 8U2 HWB ke *ground*, yang membuatnya lebih mudah untuk diletakkan ke dalam DFU mode. Revisi 3 dari board Arduino UNO memiliki fitur-fitur baru sebagai berikut:

- Pinout 1.0: ditambah pin SDA dan SCL yang dekat dengan pin AREF dan dua pin baru lainnya yang diletakkan dekat dengan pin RESET, IOREF yang memungkinkan *shield-shield* untuk menyesuaikan tegangan yang disediakan dari board. Untuk ke depannya, *shield* akan dijadikan kompatibel/cocok dengan *board* yang menggunakan AVR yang beroperasi dengan tegangan 5V dan dengan Arduino Due yang beroperasi dengan tegangan 3.3V. Yang ke-dua ini merupakan sebuah pin yang tak terhubung, yang disediakan untuk tujuan kedepannya.
- Sirkuit RESET yang lebih kuat.
- Atmega 16U2 menggantikan 8U2.

“Uno” berarti satu dalam bahasa Italia dan dinamai untuk menandakan keluaran (produk) Arduino 1.0 selanjutnya. Arduino UNO dan versi 1.0 akan menjadi referensi untuk versi-versi Arduino selanjutnya. Arduino UNO adalah sebuah seri terakhir dari board Arduino USB dan model referensi untuk papan Arduino, untuk suatu perbandingan dengan versi sebelumnya. Berikut adalah ringkasan dari arduino yang ditunjukkan dalam Tabel 2.1.

**Tabel 2.1. Ringkasan Spesifikasi Arduino Uno**

Mikrokontroler	ATmega328
Tegangan pengoperasian	5V
Tegangan input yang disarankan	7-12V
Batas tegangan input	6-20V
Jumlah pin I/O digital	14 (6 di antaranya menyediakan keluaran PWM)
Jumlah pin input analog	6
Arus DC tiap pin I/O	40 mA
Arus DC untuk pin 3.3V	50 mA
Memori Flash	32 KB (ATmega328), sekitar 0.5 KB digunakan oleh bootloader
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Clock Speed	16 MHz

Referensi desain Arduino dapat menggunakan sebuah Atmega8, 168, atau 328, model saat ini menggunakan Atmega328, tetapi Atmega8 ditampilkan pada skema sebagai referensi. Konfigurasi pin identik pada semua ketiga prosesor tersebut.

#### 2.4.2 Daya (Power)

Arduino UNO dapat disuplai melalui koneksi USB atau dengan sebuah power suplai eksternal. Sumber daya dipilih secara otomatis. Suplai eksternal (non-USB) dapat diperoleh dari sebuah adaptor AC ke DC atau battery. Adaptor dapat dihubungkan dengan mencolokkan sebuah center-positive plug yang panjangnya 2,1 mm ke *power jack* dari *board*. Kabel *lead* dari sebuah *battery* dapat dimasukkan dalam *header/kepala* pin *Ground* (Gnd) dan pin *Vin* dari konektor *POWER*. *Board* Arduino UNO dapat beroperasi pada sebuah suplai eksternal 6 sampai 20 Volt.

Jika disuplai dengan yang lebih kecil dari 7 V, kiranya pin 5 Volt mungkin mensuplai kecil dari 5 Volt dan board Arduino UNO bisa menjadi tidak stabil. Jika menggunakan suplai yang lebih dari besar 12 Volt, *voltage regulator* bisa kelebihan panas dan membahayakan board Arduino UNO. Range yang direkomendasikan adalah 7 sampai 12 Volt. Pin-pin dayanya adalah sebagai berikut:

- VIN. Tegangan input ke Arduino board ketika board sedang menggunakan sumber suplai eksternal (seperti 5 Volt dari koneksi USB atau sumber tenaga lainnya yang diatur). Tegangan dapat mensuplai melalui pin ini, atau jika penyuplaian tegangan melalui *power jack*, aksesnya melalui pin ini.
- 5V. Pin output ini merupakan tegangan 5 Volt yang diatur dari regulator pada board. Board dapat disuplai dengan salah satu suplai dari DC *power jack* (7-12V), USB connector (5V), atau pin VIN dari board (7-12). Penyuplaian tegangan melalui pin 5V atau 3,3V membypass regulator, dan dapat membahayakan board. Hal itu tidak dianjurkan.
- 3V3. Sebuah suplai 3,3 Volt dihasilkan oleh regulator pada board. Arus maksimum yang dapat dilalui adalah 50 mA



- GND. Pin *ground*.
- IOREF. Pin ini di papan Arduino memberikan tegangan referensi dengan yang mikrokontroler beroperasi. Sebuah perisai dikonfigurasi dengan benar dapat membaca pin tegangan IOREF dan pilih sumber daya yang tepat atau mengaktifkan penerjemah tegangan pada output untuk bekerja dengan 5V atau 3.3V. ATmega328 mempunyai 32 KB (dengan 0,5 KB digunakan untuk *bootloader*). ATmega 328 juga mempunyai 2 KB SRAM dan 1 KB EEPROM (yang dapat dibaca dan ditulis (RW/read and written) dengan EEPROM library).

ATmega328 memiliki 32 KB (dengan 0,5 KB digunakan untuk *bootloader*). Ini juga memiliki 2 KB dari SRAM dan 1 KB EEPROM (yang dapat dibaca dan ditulis dengan perpustakaan EEPROM).

### 2.4.3 Input dan Output

Setiap 14 pin digital pada Arduino Uno dapat digunakan sebagai input dan output, menggunakan fungsi `pinMode()`, `digitalWrite()`, dan `digitalRead()`. Fungsi-fungsi tersebut beroperasi di tegangan 5 Volt. Setiap pin dapat memberikan atau menerima suatu arus maksimum 40 mA dan mempunyai sebuah resistor pull-up (terputus secara default) 20-50 kOhm. Selain itu, beberapa pin mempunyai fungsi-fungsi spesial:

- Serial: 0 (RX) dan 1 (TX). Digunakan untuk menerima (RX) dan memancarkan (TX) serial data TTL (Transistor-Transistor Logic). Kedua pin ini dihubungkan ke pin-pin yang sesuai dari chip Serial Atmega8U2 USB-ke-TTL.
- External Interrupts: 2 dan 3. Pin-pin ini dapat dikonfigurasi untuk dipicu sebuah interrupt (gangguan) pada sebuah nilai rendah, suatu kenaikan atau penurunan yang besar, atau suatu perubahan nilai. Lihat fungsi `attachInterrupt()` untuk lebih jelasnya.
- PWM: 3, 5, 6, 9, 10, dan 11. Memberikan 8-bit PWM output dengan fungsi `analogWrite()`.

- SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). Pin-pin ini mensupport komunikasi SPI menggunakan SPI library.
- LED: 13. Ada sebuah LED yang terpasang, terhubung ke pin digital 13. Ketika pin bernilai HIGH LED menyala, ketika pin bernilai LOW LED mati.

Arduino UNO mempunyai 6 input analog, diberi label A0 sampai A5, setiapnya memberikan 10 bit resolusi (contohnya 1024 nilai yang berbeda). Secara default, 6 input analog tersebut mengukur dari ground sampai tegangan 5 Volt, dengan itu mungkin untuk mengganti batas atas dari rangenya dengan menggunakan pin AREF dan fungsi *analogReference()*. Di sisi lain, beberapa pin mempunyai fungsi spesial:

- TWI: pin A4 atau SDA dan pin A5 atau SCL. Mensupport komunikasi TWI dengan menggunakan *Wire library*.
- AREF. Referensi tegangan untuk input analog. Digunakan dengan *analogReference()*.
- Reset. Membawa saluran ini LOW untuk mereset mikrokontroler. Secara khusus, digunakan untuk menambahkan sebuah tombol reset untuk melindungi yang memblock sesuatu pada board.

#### 2.4.4 Komunikasi

Arduino UNO mempunyai sejumlah fasilitas untuk komunikasi dengan sebuah komputer, Arduino lainnya atau mikrokontroler lainnya. Atmega 328 menyediakan serial komunikasi UART TTL (5V), yang tersedia pada pin digital 0 (RX) dan 1 (TX). Sebuah Atmega 16U2 pada channel board serial komunikasinya melalui USB dan muncul sebagai sebuah port virtual ke *software* pada komputer. Firmware 16U2 menggunakan driver USB COM standar, dan tidak ada driver eksternal yang dibutuhkan. Bagaimanapun, pada Windows, sebuah file inf pasti dibutuhkan. *Software* Arduino mencakup sebuah serial monitor yang memungkinkan data tekstual terkirim ke dan dari board Arduino. LED RX dan TX pada board akan menyala ketika data sedang ditransmit melalui chip USB-to-serial dan koneksi USB pada komputer (tapi tidak untuk komunikasi serial pada pin 0 dan 1).



Sebuah *SoftwareSerial* library memungkinkan untuk komunikasi serial pada beberapa pin digital UNO. Atmega328 juga mensupport komunikasi I2C (TWI) dan SPI. *Software Arduino* mencakup sebuah *Wire* library untuk memudahkan menggunakan bus I2C, lihat dokumentasi untuk lebih jelas. Untuk komunikasi SPI, gunakan SPI library.

#### 2.4.5 Pemrograman

Arduino UNO dapat diprogram dengan *software* Arduino (download). Pilih “Arduino Uno dari menu Tools > Board (termasuk mikrokontroler pada board). Untuk lebih jelas, lihat referensi dan tutorial. ATmega328 pada Arduino Uno hadir dengan sebuah *bootloader* yang memungkinkan untuk mengupload kode baru ke ATmega328 tanpa menggunakan pemrogram *hardware* eksternal. ATmega328 berkomunikasi menggunakan protokol STK500 asli (referensi, file C header). Contoh program pada arduino ditunjukkan dalam Gambar 2.5.

```
#define LED_PIN13
|
void setup () {
  pinMode (LED_PIN, OUTPUT); // enable pin 13 for digital output
}

void loop () {
  digitalWrite (LED_PIN, HIGH); // turn on the LED
  delay (1000); // wait one second (1000 milliseconds)
  digitalWrite (LED_PIN, LOW); // turn off the LED
  delay (1000); // wait one second
}
```

**Gambar 2.5 Contoh Program Pada Arduino**

Pada pemrograman arduino juga dapat membypass *bootloader* dan program mikrokontroler melalui kepala/header ICSP (In-Circuit Serial Programming); lihat instruksi untuk lebih jelas Sumber kode firmware ATmega16U2 (atau 8U2 pada board revisi 1 dan revisi 2) tersedia. ATmega16U2/8U2 diload dengan sebuah *bootloader* DFU, yang dapat diaktifkan dengan:

- Pada board Revisi 1: Dengan menghubungkan jumper solder pada belakang board (dekat peta Italy) dan kemudian mereset 8U2.

- Pada board Revisi 2 atau setelahnya: Ada sebuah resistor yang menarik garis HWB 8U2/16U2 ke ground, dengan itu dapat lebih mudah untuk meletakkan ke dalam mode DFU. Dalam hal ini dapat menggunakan *software* Atmel's FLIP (Windows) atau pemrogram DFU (Mac OS X dan Linux) untuk meload sebuah firmware baru. Atau dapat menggunakan header ISP dengan sebuah pemrogram eksternal (mengoverwrite *bootloader* DFU). Lihat tutorial user-contributed ini untuk informasi selengkapnya.

#### 2.4.6 Reset Otomatis

Dari pada mengharuskan sebuah penekanan fisik dari tombol reset sebelum sebuah penguploadan, Arduino Uno didesain pada sebuah cara yang memungkinkannya untuk direset dengan *software* yang sedang berjalan pada pada komputer yang sedang terhubung. Salah satu garis kontrol aliran *hardware* (DTR) dari ATmega8U2/16U2 sihubungkan ke garis reset dari ATmega328 melalui sebuah kapasitor 100 *nanofarad*. Ketika saluran ini dipaksakan (diambil rendah), garis reset jatuh cukup panjang untuk mereset chip. *Software* Arduino menggunakan kemampuan ini untuk memungkinkan untuk *mengupload* kode dengan mudah menekan tombol upload di *software* Arduino. Ini berarti bahwa *bootloader* dapat mempunyai sebuah batas waktu yang lebih singkat, sebagai penurunan dari DTR yang dapat menjadi koordinasi yang baik dengan memulai penguploadan.

Pengaturan ini mempunyai implikasi. Ketika Arduino Uno dihubungkan ke sebuah komputer lain yang sedang running menggunakan OS Mac X atau Linux, Arduino Uno mereset setiap kali sebuah koneksi dibuat dari *software* (melalui USB). Untuk berikutnya, setengah-detik atau lebih, *bootloader* sedang berjalan pada Arduino UNO. Ketika Arduino UNO diprogram untuk mengabaikan data yang cacat/salah (contohnya apa saja selain sebuah penguploadan kode baru) untuk menahan beberapa bit pertama dari data yang dikirim ke board setelah sebuah koneksi dibuka. Jika sebuah sketch sedang berjalan pada board menerima satu kali konfigurasi atau data lain ketika sketch pertama mulai, memastikan



bahwa *software* yang berkomunikasi menunggu satu detik setelah membuka koneksi dan sebelum mengirim data ini.

Arduino Uno berisikan sebuah jejak yang dapat dihapus untuk mencegah reset otomatis. Pad pada salah satu sisi dari jejak dapat disolder bersama untuk mengaktifkan kembali. Pad itu diberi label “RESET-RN” yang juga dapat menonaktifkan reset otomatis dengan menghubungkan sebuah resistor 110 ohm dari tegangan 5V ke garis reset; lihat thread forum ini untuk lebih jelasnya.

#### **2.4.7 Proteksi Arus USB**

Arduino UNO mempunyai sebuah sekering reset yang memproteksi port USB komputer dari hubungan pendek dan arus lebih. Walaupun sebagian besar komputer menyediakan proteksi internal sendiri, sekering menyediakan sebuah proteksi tambahan. Jika lebih dari 500 mA diterima port USB, sekering secara otomatis akan memutuskan koneksi sampai hubungan pendek atau kelebihan beban hilang.

#### **2.4.8 Karakteristik Fisik**

Panjang dan lebar maksimum dari PCB Arduino UNO masing-masingnya adalah 2.7 dan 2.1 inci, dengan konektor USB dan power jack yang memperluas dimensinya. Empat lubang sekrup memungkinkan board untuk dipasangkan ke sebuah permukaan atau kotak. Sebagai catatan, bahwa jarak antara pin digital 7 dan 8 adalah 160 milimeter, bukan sebuah kelipatan genap dari jarak 100 milimeter dari pin lainnya. ([www.arduino.cc/en/Main/ArduinoBoardUno](http://www.arduino.cc/en/Main/ArduinoBoardUno),2016)

### **2.5 Kontroler**

Sistem pengendalian dirancang untuk melakukan dan menyelesaikan tugas tertentu. Syarat utama sistem pengendalian adalah harus stabil. Di samping kestabilan mutlak, maka sistem harus memiliki kestabilan secara relatif, yakni tolak ukur kualitas kestabilan sistem dengan menganalisis sampai sejauh mana batas-batas kestabilan sistem tersebut jika dikenai gangguan (Ogata K.,1997). Selain itu analisis juga dilakukan untuk mengetahui bagaimana kecepatan sistem dalam merespons *input*, dan bagaimana peredaman terhadap adanya lonjakan (*over shoot*).

Suatu sistem dikatakan stabil jika diberi gangguan maka sistem tersebut akan kembali ke keadaan *steady state* di mana *output* berada dalam keadaan tetap seperti tidak ada gangguan. Sistem dikatakan tidak stabil jika *output*nya berosilasi terus menerus ketika dikenai suatu gangguan. Karena suatu sistem pengendalian biasanya melibatkan penyimpanan energi maka *output* sistem ketika diberi suatu *input*, tidak dapat mengikuti *input* secara serentak, tapi menunjukkan respons transien berupa suatu osilasi teredam sebelum mencapai *steady state*.

Dalam sistem pengendalian terdapat 2 macam *loop*:

1. Pengendalian dengan *loop* terbuka

Sistem kontrol *loop* terbuka adalah sistem kontrol yang keluarannya tidak berpengaruh pada aksi pengontrolan. Jadi pada sistem kontrol *loop* terbuka, keluaran tidak diukur atau diumpun balik untuk dibandingkan dengan masukan.

2. Pengendalian dengan *loop* tertutup

Sistem kontrol *loop* tertutup adalah sistem kontrol yang keluarannya mempunyai pengaruh langsung pada aksi pengontrolan. Disebut juga sistem kontrol yang menggunakan umpan balik untuk memperkecil kesalahan sistem.

## 2.6 Kontroler Proporsional

Kontroler proporsional memiliki keluaran yang sebanding dengan sinyal kesalahan (selisih antara besaran yang diinginkan dengan harga aktualnya). Untuk kontroler dengan aksi kontrol proporsional, hubungan antara keluaran kontroler  $m(t)$  dan sinyal kesalahan penggerak  $e(t)$  adalah:

$$m(t) = K_p \cdot e(t) \dots \dots \dots (2.1)$$

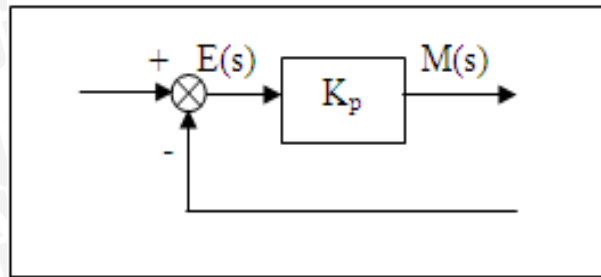
atau, dalam besaran transformasi Laplace,

$$\frac{M(s)}{E(s)} = K_p \dots \dots \dots (2.2)$$

Di mana  $K_p$  adalah kepekaan proporsional atau penguatan.

Apapun wujud mekanisme yang sebenarnya dan apapun bentuk daya penggerakannya, kontroler proporsional pada dasarnya merupakan penguat dengan penguatan yang dapat diatur (Ogata K.,1997). Diagram blok kontroler proporsional ditunjukkan dalam Gambar 2.6.





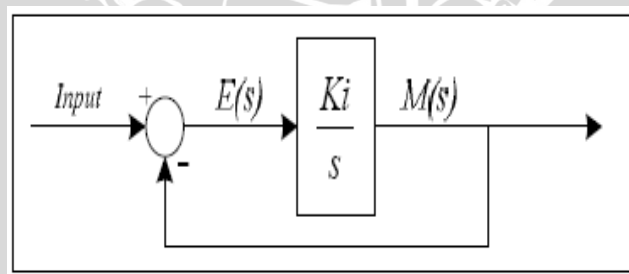
Gambar 2.6 Diagram Blok Kontroler Proporsional

2.7 Kontroler Integral

Kontroler integral berfungsi mengurangi kesalahan keadaan mantap pada kontroler proporsional sebelumnya. Pada kontroler dengan aksi integral, harga keluaran kontroler  $m(t)$  diubah dengan laju yang sebanding dengan sinyal kesalahan penggerak  $e(t)$ . Jadi,

$$\frac{dm(t)}{dt} = K_i \cdot e(t) \dots\dots\dots (2.3)$$

Jika harga  $e(t)$  diduakalikan, maka harga  $m(t)$  berubah dengan laju perubahan menjadi dua kali semula. Jika kesalahan penggerak nol, maka harga  $m(t)$  tetap stasioner. Aksi kontrol integral seringkali disebut *control reset* (Ogata K.,1997). Diagram blok kontroler integral ditunjukkan dalam Gambar 2.7.



Gambar 2.7 Diagram Blok Kontroler Integral

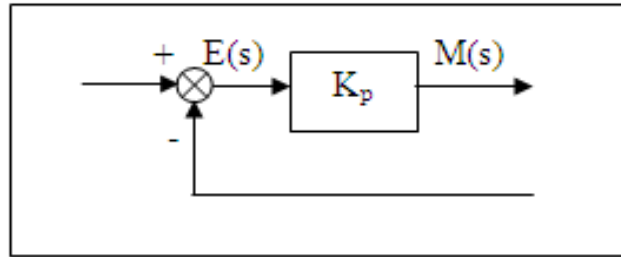
2.8 Kontroler Diferensial

Kontroler ini digunakan untuk memperbaiki atau mempercepat respons transien sebuah sistem kontrol dengan cara memperbesar *phase lead* terhadap penguatan kontrol dan mengurangi *phase lag* penguatan tersebut (Ogata K.,1997). Kontroler diferensial tidak dapat mengeluarkan *output* bila tidak ada perubahan

input, selain itu kontroler differensial tidak dapat digunakan untuk proses yang mengandung *noise*. Hubungan antara keluaran kontroler  $m(t)$  dan sinyal kesalahan penggerak  $e(t)$  adalah :

$$\frac{M(s)}{E(s)} = Kd \cdot s \dots\dots\dots(2.4)$$

Diagram blok kontroler differensial ditunjukkan dalam Gambar 2.8.



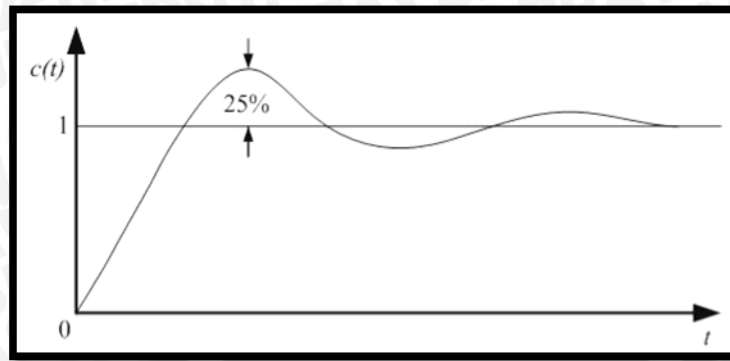
Gambar 2.8 Diagram Blok Kontroler Diferensial

## 2.9 Metode Kontroler Proporsional Integral Diferensial (PID) (*Proportional Integral Derivatif*)

Ziegler dan Nichols mengemukakan aturan-aturan untuk menentukan nilai dari gain proporsional  $K_p$ , waktu integral  $T_i$ , dan waktu derivatif  $T_d$  berdasarkan karakteristik respon transien dari *plant* yang diberikan. Penentuan parameter kontroler PID (*Proportional Integral Derivatif*) atau penalaan kontroler PID (*Proportional Integral Derivatif*) tersebut dapat dilakukan dengan bereksperimen dengan plan. (Ogata, K., 1997).

Terdapat dua metode yang disebut dengan aturan penalaan Ziegler-Nichols, pada kedua metode tersebut memiliki tujuan yang sama yaitu untuk mencapai 25% *maximum overshoot* pada respon unit step, ditunjukkan dengan Gambar 2.9.

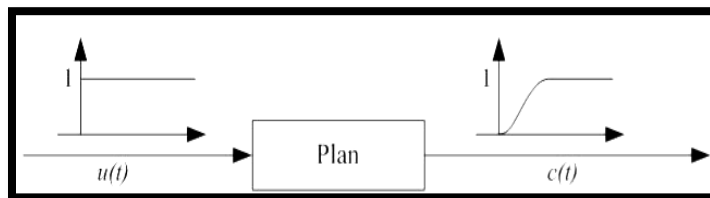




**Gambar 2.9 Kurva Respon Unit Step yang Menunjukkan 25% Maximum Overshoot**

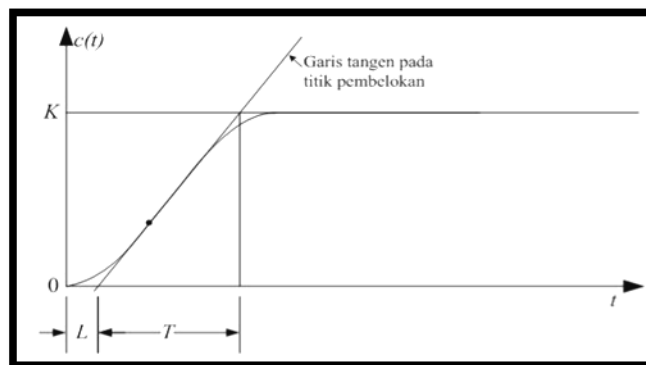
1) Metode Pertama

Metode pertama atau sering disebut metode kurva reaksi, respon dari plan dapat diperoleh secara eksperimental dengan masukan berupa unit step, seperti yang ditunjukkan dengan Gambar 2.10.



**Gambar 2.10 Respon Plant Terhadap Masukan Berupa Unit Step**

Jika dalam plan tersebut terdapat integrator atau *dominan complex-conjugate poles*, maka kurva respon unit step berbentuk seperti huruf S, seperti ditunjukkan dengan Gambar 2.11 jika respon tidak memberikan bentuk kurva S, maka metode ini tidak berlaku. (Ogata, K., 1997).



**Gambar 2.11 Kurva Respon yang Berbentuk S**

Kurva berbentuk S tersebut dapat dikarakteristikan menjadi dua konstanta yaitu waktu tunda  $L$  dan konstanta waktu  $T$ . Waktu tunda dan konstanta waktu ditentukan dengan menggambar sebuah garis tangen pada titik pembelokan dari kurva S, dan menentukan perpotongan antara garis tangen dengan sumbu waktu  $t$  dan sumbu  $c(t) = K$ , seperti yang telah ditunjukkan dalam Persamaan (6-1). Fungsi alih  $C(s)/U(s)$  dapat dilakukan pendekatan dengan sistem orde satu dengan persamaan sebagai berikut:

$$\frac{C(s)}{U(s)} = \frac{Ke^{-Ls}}{Ts+1} \quad \dots\dots\dots(2.5)$$

Ziegler dan Nichols menyarankan untuk menentukan nilai-nilai dari  $K_p$ ,  $T_i$  dan  $T_d$  berdasarkan pada formula yang ditunjukkan dalam Tabel 2.1 (Ogata, K., 1997)

**Tabel 2.2 Aturan Penalaan Ziegler-Nichols Berdasarkan Respon Unit Step Dari Plan**  
(Sumber: Ogata, K., 1997)

<b>Tipe Kontroler</b>	$K_p$	$T_i$	$T_d$
<b>P</b>	$\frac{T}{L}$	$\infty$	0
<b>PI</b>	$0,9\frac{T}{L}$	$\frac{L}{0,3}$	0
<b>PID (<i>Proportional Integral Derivatif</i>)</b>	$1,2\frac{T}{L}$	$2L$	$0,5 L$

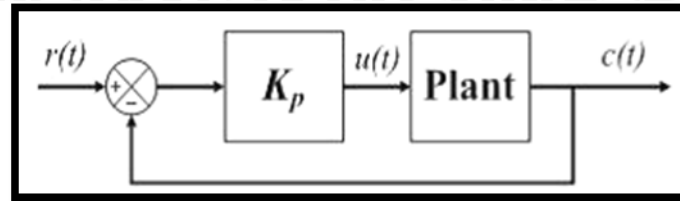
## 2) Metode Kedua

Dalam metode kedua ziegler-nichols, mula-mula yang dilakukan adalah membuat  $T_i = \infty$  dan  $T_d = 0$ . Kemudian hanya dengan menggunakan tindakan kontrol proporsional, harga ditingkatkan dari nol ke suatu nilai kritis  $K_{cr}$ , disini mula-mula keluaran memiliki osilasi yang berkesinambungan (Jika keluaran tidak memiliki osilasi berkesinambungan untuk nilai  $K_p$  manapun yang telah diambil,

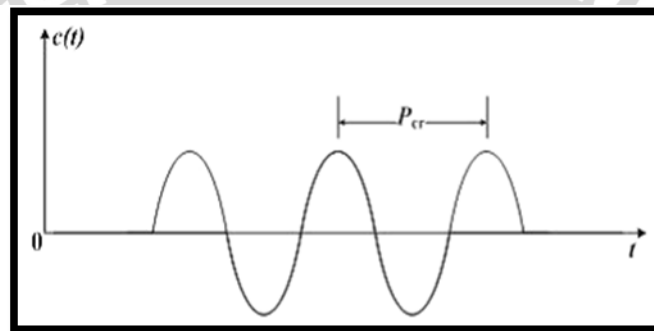


maka metode ini tidak berlaku). Dari keluaran yang berosilasi secara berkesinambungan, penguatan kritis  $K_{cr}$  dan periode  $P_{cr}$  dapat ditentukan.

Diagram blok sistem loop tertutup dengan kontroler proporsional ditunjukkan dengan Gambar 2.12 dan untuk osilasi berkesinambungan dengan periode  $P_{cr}$  ditunjukkan dengan Gambar 2.13.



Gambar 2.12 Sistem Loop Tertutup dengan Kontroler Proporsional



Gambar 2.13 Osilasi Berkesinambungan dengan Periode  $P_{cr}$

Ziegler dan Nichols menyarankan penyetelan nilai parameter  $K_p, T_i, T_d$  dan berdasarkan rumus yang diperlihatkan dalam Tabel 2.3. (Ogata, K., 1997)

Tabel 2.3 Aturan Dasar Ziegler-Nichols Berdasarkan Critical Gain  $K_{cr}$  dan Critical Period  $P_{cr}$

Type Kontroler	$K_p$	$T_i$	$T_d$
P	$0.5 K_{cr}$	$\infty$	0
PI	0.45 $K_{cr}$	0.5 $P_{cr}$	0
PID ( <i>Proportional Integral Derivatif</i> )	0.60 $K_{cr}$	0.5 $P_{cr}$	0.125 $P_{cr}$





## BAB III

### METODE PENELITIAN

Kajian dalam skripsi ini merupakan penelitian yang bersifat aplikatif, yaitu merancang dan membuat suatu miniatur *flyboard* dengan menitikberatkan pengendalian keseimbangan dengan menggunakan kontroler PID (*Proportional Integral Derivatif*) yang bertujuan agar dapat menampilkan performansi sistem sesuai dengan yang direncanakan.

Langkah-langkah yang perlu dilakukan untuk merealisasikan alat yang akan dibuat adalah sebagai berikut:

1. Spesifikasi alat
2. Perancangan dan realisasi pembuatan alat
3. Pengujian alat
4. Pengambilan kesimpulan

#### 3.1 Spesifikasi Alat

Adapun spesifikasi alat yang akan direalisasikan adalah sebagai berikut:

1. Mikrokontroler yang digunakan untuk mengatur proses otomatisasi adalah Arduino Uno.
2. Aktuator yang digunakan untuk menjaga keseimbangan *flyboard* adalah 2 buah pompa air.
3. Sensor keseimbangan yang digunakan untuk membaca posisi kemiringan *flyboard* adalah sensor *accelerometer* ADXL345.
4. Pemrograman menggunakan *compiler* arduino.ide.

#### 3.2 Perancangan dan Realisasi Pembuatan Alat

##### 3.2.1 Perancangan Perangkat Keras

- a. Pembuatan diagram blok secara lengkap
- b. Penentuan dan perhitungan komponen yang akan digunakan
- c. Merakit perangkat keras (*hardware*) untuk masing-masing blok.

##### 3.2.2 Perancangan Perangkat Lunak

Perancangan perangkat lunak dilakukan setelah mengetahui nilai parameter *setpoint* keseimbangan. Perancangan dimulai dari pembuatan

*flowchart*, kemudian dilanjutkan dengan penulisan *listing coden* pada *compiler arduino.ide*.

### 3.3 Pengujian Alat

Setelah semua komponen pada alat sudah terhubung sesuai dengan diagram blok sistem yang telah dirancang dan perangkat lunak untuk mendukung sistem yang telah dibuat, maka diadakan pengujian dan analisa alat. Metode pengujiannya adalah sebagai berikut:

1. Pengujian Rangkaian *Zero Crossing*

Pengujian dilakukan dengan cara menghubungkan rangkaian *zero crossing* dengan tegangan jala-jala. Kemudian tegangan jala-jala dan keluaran rangkaian *zero crossing* dihubungkan dengan probe osiloskop

2. Pengujian Rangkaian Penyulutan TRIAC

Pengujian dilakukan dengan cara menghubungkan rangkaian *zero crossing* dan penyulutan triac pada arduino. Kemudian rangkaian yang sudah terhubung satu sama lain diberikan tegangan jala-jala yang keluaran rangkaian penyulutan TRIAC akan dihubungkan dengan osiloskop untuk mengetahui sinyal yang dihasilkan.

3. Pengujian Sensor *Accelerometer* ADXL345

Pengujian dilakukan dengan cara memberikan kemiringan pada sensor accelerometer, dimana nilai kemiringan diukur menggunakan busur derajat. Kemudian nilai kemiringan busur dibandingkan dengan nilai kemiringan yang dibaca oleh arduino. Pada pengujian kali ini hanya menguji satu axis yaitu axis x.

4. Pengujian PID (*Proportional Integral Derivatif*)

Pengujian PID (*Proportional Integral Derivatif*) dilakukan dengan cara memberikan nilai Kp dari nilai terkecil sampai respon mempunyai osilasi yang stabil, dimana nilai Ki dan Kd diberikan nilai 0. Setelah didapatkan grafik dengan sistem osilasi yang cukup stabil maka diperoleh parameter Kcr dan Pcr. Dimana nilai tersebut dimasukkan ke dalam tabel aturan dasar *Ziegler-Nichols* yang kemudian didapatkan nilai Kp, Ki dan Kd

5. Pengujian Keseluruhan Sistem



Pengujian dilakukan dengan cara mengaktifkan *flyboard* dan kemudian ketika respon sistem mulai *steady*, *flyboard* diberikan gangguan dengan cara menyentuh *flyboard* agar keseimbangan *flyboard* terganggu.

### 3.4 Pengambilan Kesimpulan

Kesimpulan diambil berdasarkan data yang diperoleh dari pengujian sistem secara keseluruhan. Jika hasil yang didapatkan telah sesuai dengan yang direncanakan sebelumnya, maka sistem kendali tersebut telah berhasil memenuhi harapan dan tentunya memerlukan pengembangan lebih lanjut untuk penyempurnaan.

UNIVERSITAS BRAWIJAYA







## BAB IV

### PERANCANGAN DAN PEMBUATAN ALAT

Dalam bab ini akan dibahas cara perancangan dan pembuatan Alat Pengendali Keseimbangan *Flyboard* Menggunakan Metode PID (*Proportional Integral Derivatif*). Perancangan alat meliputi perancangan mekanik, perancangan elektrik dan perancangan *software*.

Perancangan perangkat elektrik terdiri dari perancangan rangkaian catu daya, perancangan rangkaian *dimmer*, perancangan rangkaian sensor *accelerometer*. Sedangkan untuk perancangan *software* meliputi *flowchart* dan program untuk arduino uno.

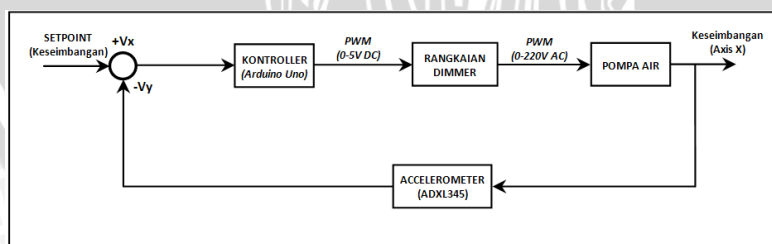
#### 4.1 Spesifikasi Sistem

Spesifikasi sistem yang dirancang adalah sebagai berikut:

- Menggunakan pompa air aquilla P5900 sebagai pendorong *flyboard*.
- Menggunakan Arduino uno sebagai pemroses utama sistem keseimbangan *flyboard*.
- Menggunakan rangkaian *dimmer* sebagai pengatur tegangan kekuatan dorong pompa air.
- Menggunakan rangkaian *zero crossing* untuk mengetahui sudut  $0^0$  tegangan jala-jala.
- Menggunakan sensor *accelerometer* ADXL345 sebagai sensor keseimbangan.

#### 4.2 Diagram Blok Sistem

Diagram blok controller ditunjukkan dalam Gambar 4.1

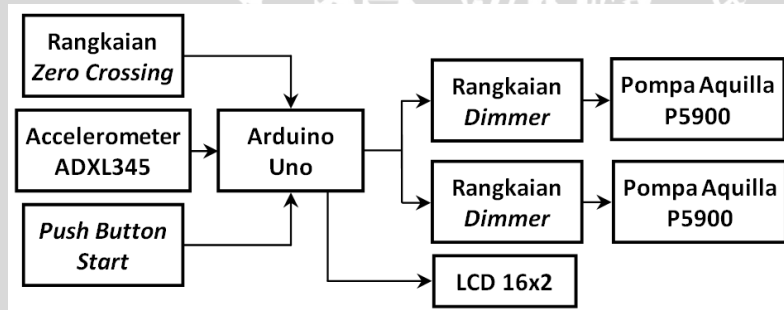


Gambar 4.1 Diagram Blok Controller (Perancangan, 201

Keterangan blok diagram dalam Gambar 4.1 adalah

- *Input* berupa nilai *setpoint* parameter keseimbangan *flyboard*.
- Sinyal *feedback* dari sensor *accelerometer ADXL345* berupa nilai keseimbangan (PV). pembacaan sensor *accelerometer ADXL345* yang diumpanbalikkan.
- *Error* berupa selisih nilai pengurangan antara *setpoint* (SP) dengan nilai keseimbangan yang terjadi pada saat itu.
- *Kontroller* yang digunakan adalah *Proportional, Integral dan Diferensial* dengan *arduino uno* untuk mengolah nilai *PID (Proportional Integral Derivatif)* yang digunakan sebagai nilai kontrol keseimbangan *flyboard*.
- *Aktuator* berupa 2 (dua) pompa air *aquilla P5900* yang digunakan sebagai tenaga pendorong *flyboard*.
- *Plant* berupa keseimbangan *flyboard*.
- *Output* berupa nilai hasil kontrol yang berpengaruh pada kekuatan gaya dorong *flyboard*.

Sistem pengaturan keseimbangan *flyboard* secara keseluruhan ditunjukkan dalam Gambar 4.2.



Gambar 4.2 Diagram Blok Sistem (Perancangan, 2016)

Keterangan diagram blok dalam Gambar 4.2 adalah:

- Sensor keseimbangan (*Accelerometer ADXL345*) memberikan nilai pembacaan keseimbangan *flyboard*.
- *Rangkaian Zero Crossing* berfungsi untuk membaca sudut  $0^0$  tegangan jala-jala yang akan digunakan sebagai nilai masukan untuk mengaktifkan rangkaian *dimmer*.
- *Push Button Start* berfungsi sebagai penanda untuk sistem mulai aktif.



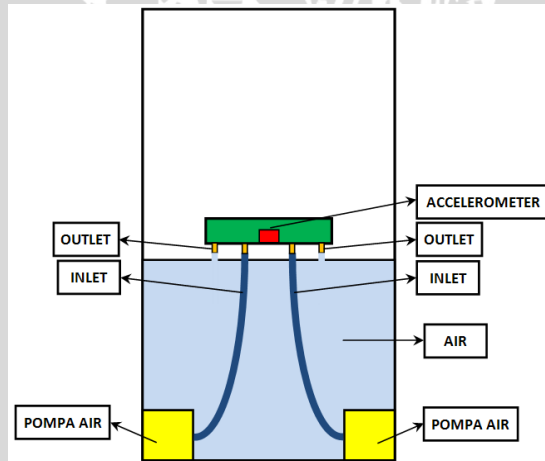
- Arduino uno mengolah nilai *error* dari simpangan antara nilai *setpoint* (SP) sebagai input dan kecepatan motor (PV) untuk menghasilkan sinyal kontrol yang kemudian dikirimkan rangkaian *dimmer*.
- Rangkaian *Dimmer* berfungsi untuk mengatur nilai efektif tegangan jala-jala dengan cara mencacah sinyal *sinusoida* pada tegangan jala-jala.
- LCD 16x2 berfungsi sebagai penampil nilai keseimbangan *flyboard*.
- Pompa Air Aquilla P5900 berfungsi sebagai pendorong *flyboard* agar dapat terangkat ke udara.

### 4.3 Perancangan Perangkat Keras

Perangkat keras terdiri dari mekanik rangka *flyboard*, rangkaian sistem keseluruhan, rangkaian sensor accelerometer ADXL3435, rangkaian LCD16x2, rangkaian *push button*, rangkaian *zero crossing*, rangkaian *dimmer*.

#### 4.3.1 Perancangan Mekanik *Flyboard*

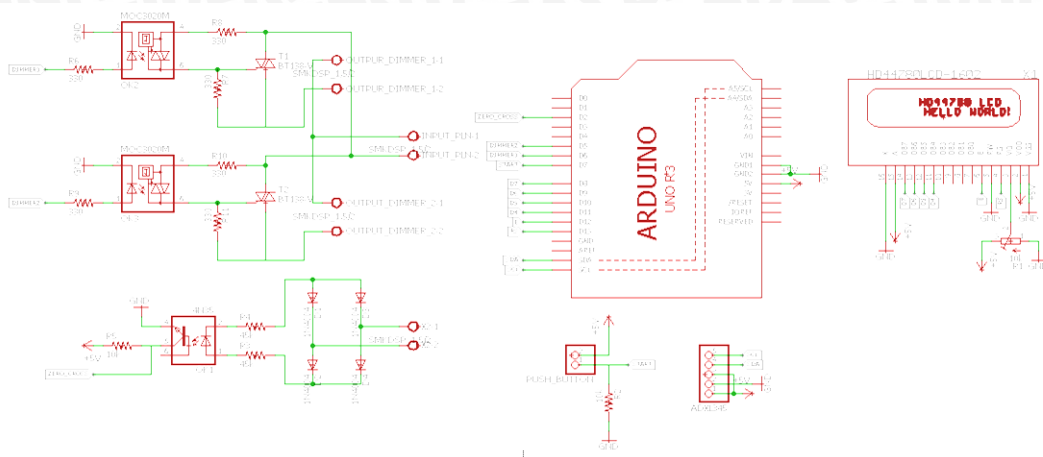
Perancangan mekanik simulasi *flyboard* dilakukan sebagai dasar dari komponen-komponen yang akan diletakkan. Secara umum mekanik untuk simulasi *flyboard* ditunjukkan dalam Gambar 4.3.



Gambar 4.3 Mekanik *Flyboard* (Perancangan, 2016)

#### 4.3.2 Perancangan Rangkaian Sistem Keseluruhan

Perancangan rangkaian sistem keseluruhan bertujuan agar sistem dapat berjalan *sinkron* dengan fungsi masing-masing komponen. Yang pada akhirnya sistem dapat berjalan sesuai dengan apa yang diharapkan. Adapun rangkaian sistem keseluruhan ditunjukkan dalam Gambar 4.4.



**Gambar 4.4 Rangkaian Sistem Keseluruhan (Perancangan, 2016)**

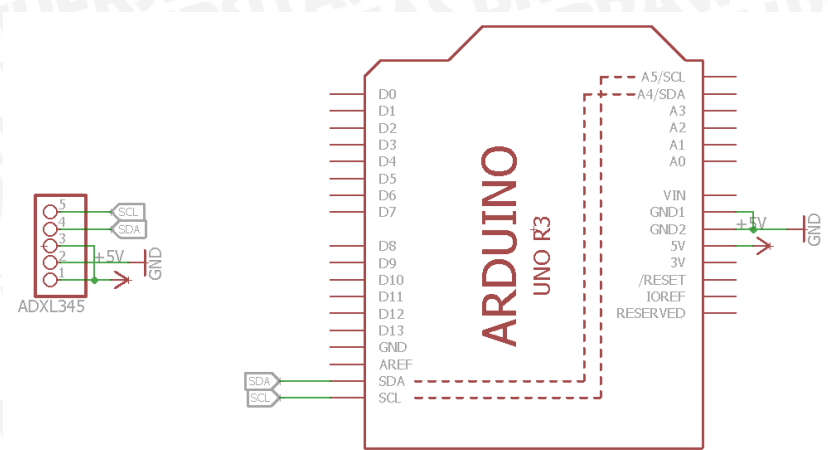
Berdasarkan hasil perancangan rangkaian sistem keseluruhan dapat diketahui bahwa arduino uno dibagi menjadi 2 (dua) fungsi sistem, yaitu fungsi masukan dan keluaran. Pin masukan dan keluaran pada perancangan ditunjukkan dalam Tabel 4.1.

**Tabel 4.1 Fungsi Pin Arduino Uno (Perancangan, 2016)**

No.	Pin	Fungsi
1	D13	Output Pin RS LCD
2	D12	Output Pin E LCD
3	D11	Output Pin D4 LCD
4	D10	Output Pin D5 LCD
5	D9	Output Pin D6 LCD
6	D8	Output Pin D7 LCD
7	D7	Input Push Button Start
8	D6	Output Pin Dimmer 1
9	D5	Output Pin Dimmer 2
10	D2	Input Pin Zero Crossing
11	SDA	Input Pin SDA ADXL345
12	SCL	Input Pin SCL ADXL345

#### 4.3.3 Perancangan Rangkaian Sensor Accelerometer ADXL345

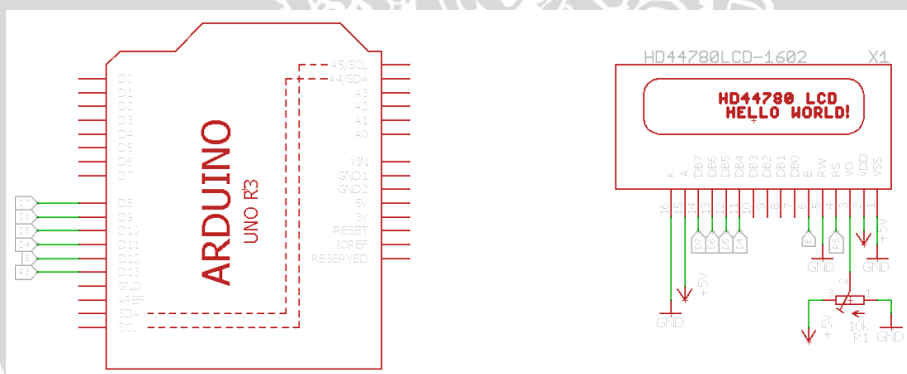
Sensor *accelerometer* ADXL345 berfungsi untuk membaca kemiringan suatu bidang. Dimana pada umumnya suatu bidang dibagi menjadi 3 bagian (axis), yaitu axis x, axis y dan axis z. Pada penelitian ini digunakan sensor accelerometer ADXL345 untuk membaca kemiringan bidang *flyboard*. Perancangan rangkaian sensor *accelerometer* ADXL345 pada arduino uno ditunjukkan dalam Gambar 4.5.



Gambar 4.5 Rangkaian Sensor Accelerometer ADXL345

#### 4.3.4 Perancangan Rangkaian LCD 16x2

LCD 16x2 berfungsi untuk menampilkan karakter tulisan yang sudah diprogram pada arduino uno. Pada penelitian kali ini LCD 16x2 digunakan untuk menampilkan data-data pemrosesan yang dilakukan oleh arduino uno sebagai pemroses utama dalam sistem. Perancangan rangkaian LCD 16x2 ditunjukkan dalam Gambar 4.6.



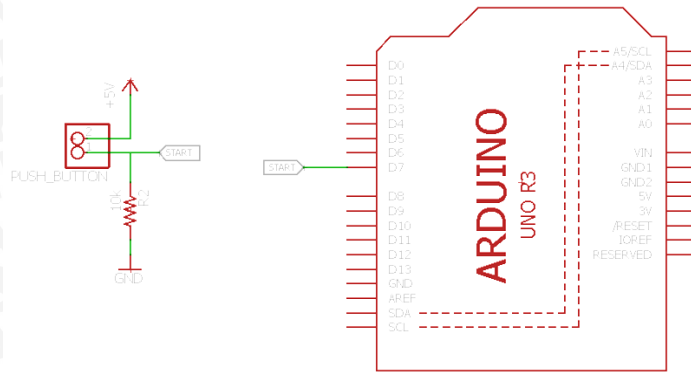
Gambar 4.6 Rangkaian LCD 16x2 (Perancangan, 2016)

#### 4.3.5 Perancangan Rangkaian Push Button

*Push button* mempunyai fungsi sebagai saklar, yang mana ketika push button ditekan maka akan memberikan pembacaan logika digital arduino uno. Dalam penelitian ini *push button* akan difungsikan sebagai penanda sistem akan mulai bekerja. Pada perancangan *push button* kali ini apabila push button belum ditekan, maka akan memberikan logika 0 (0V) pada arduino uno. Kemudian ketika *push button* ditekan, maka akan memberikan logika 1 (5V) pada arduino uno. Perancangan rangkaian push button ditunjukkan dalam Gambar 4.7.

4.7.

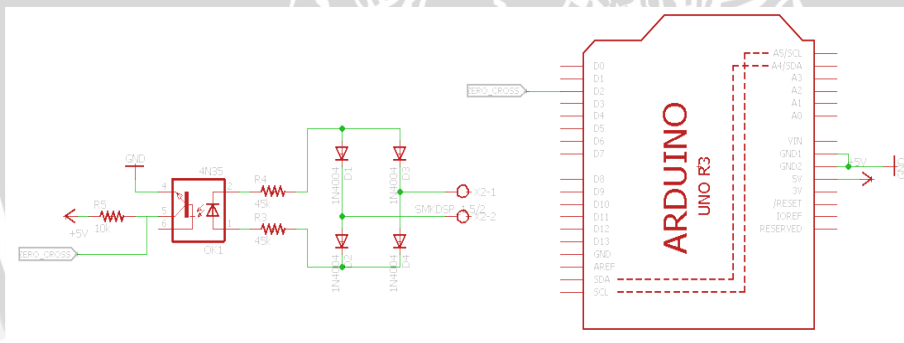




Gambar 4.7 Rangkaian Push Button (Perancangan, 2016)

#### 4.3.6 Perancangan Rangkaian Zero Crossing

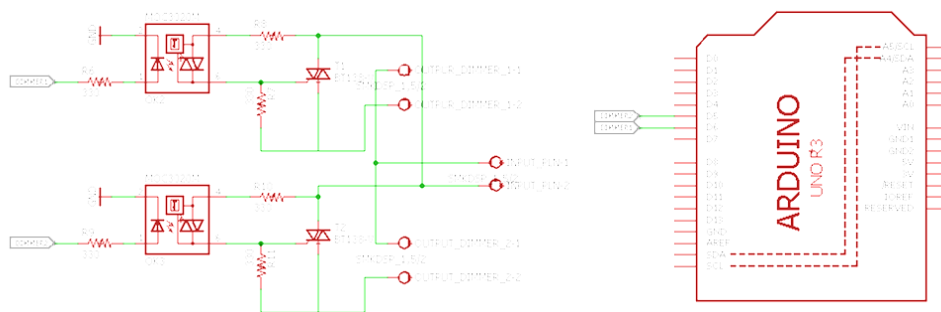
*Zero crossing* berfungsi untuk mendeteksi sudut  $0^0$  pada tegangan jala-jala PLN. Pada penelitian ini menggunakan rangkaian *zero crossing* yang digunakan sebagai bagian dari pengaturan tegangan jala-jala yang akan digunakan untuk mensupply pompa air. Dimana rangkaian zero crossing akan memberikan logika 1 (5V) ketika sinyal *sinusoida* tegangan jala-jala PLN dalam posisi 0V. Perancangan rangkaian zero crossing ditunjukkan dalam Gambar 4.8.



Gambar 4.8 Rangkaian Zero Crossing (Perancangan, 2016)

#### 4.3.7 Perancangan Rangkaian Dimmer

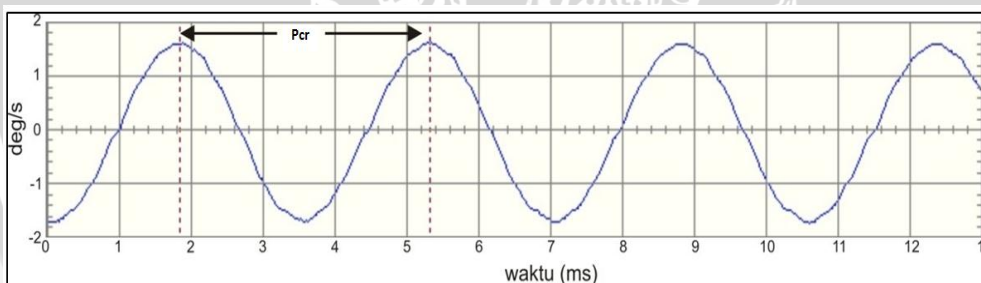
Rangkaian *dimmer* berfungsi untuk memotong atau mencacah sinyal *sinusoida* pada tegangan jala-jala PLN. Dimana pencacahan sinyal sinusoida akan berpengaruh pada tegangan rata-rata yang dihasilkan oleh rangkaian *dimmer*. Tegangan keluaran rangkaian *dimmer* akan mensupply pompa air untuk mendorong *flyboard*. Perbedaan tegangan jala-jala akan berpengaruh pada kekuatan pompa air dalam mendorong *flyboard*. Pada penelitian ini menggunakan 2 buah rangkaian *dimmer*, dimana masing-masing difungsikan untuk mensupply 1 (satu) pompa air. Perancangan rangkaian *dimmer* ditunjukkan dalam Gambar 4.9.



Gambar 4.9 Rangkaian Dimmer

#### 4.4 Perancangan Parameter PID (*Proportional Integral Derivatif*)

Perancangan parameter PID (*Proportional Integral Derivatif*) bertujuan untuk menentukan nilai  $K_p$ ,  $K_i$  dan  $K_d$  dari sistem yang akan dibuat. Pada penelitian kali ini sistem mempunyai *setpoint* berupa keseimbangan *flyboard* pada Axis X pada sudut  $180^\circ$ . Yang pertama kali dilakukan untuk menentukan nilai parameter  $K_p$ ,  $K_i$  dan  $K_d$  adalah dengan mencari nilai  $P_{cr}$ . Nilai  $P_{cr}$  dapat diperoleh dengan cara memberikan nilai  $K_{cr}$  bertahap dari paling kecil sampai sistem mencapai osilasi dengan periode yang stabil. Adapun grafik sistem osilasi yang stabil ditunjukkan dalam Gambar 4.8.



Gambar 4.10 Grafik Sistem Osilasi yang Stabil

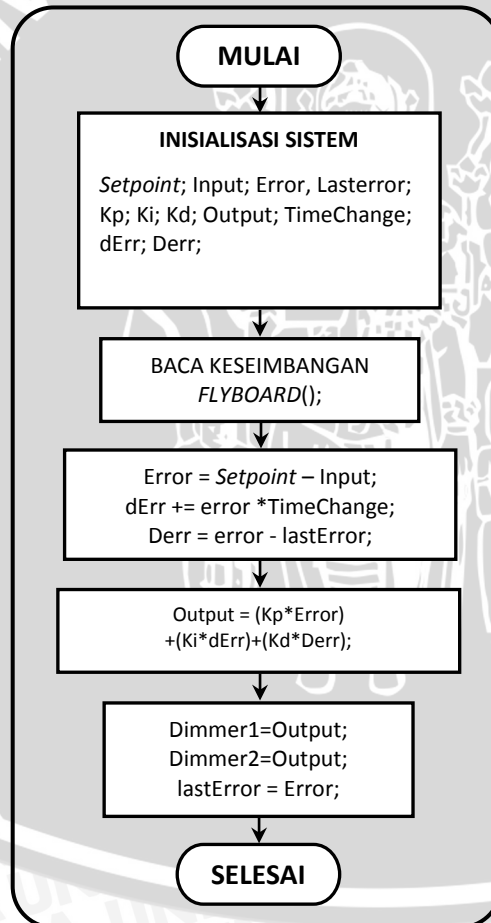
Setelah menemukan grafik osilasi yang stabil, maka nilai  $K_{cr}$  dan  $P_{cr}$  dimasukkan ke dalam tabel aturan Ziegler-Nichols yang ditunjukkan dalam Tabel 4.2.

**Tabel 4.2 Aturan Dasar Ziegler-Nichols Berdasarkan *Critical Gain Kcr* dan *Critical Period Pcr***

Typo Kontroler	Kp	Ti	Td
P	$0.5 K_{cr}$	$\infty$	0
PI	$0.45 K_{cr}$	$\frac{1}{1.2} P_{cr}$	0
PID ( <i>Proportional Integral Derivatif</i> )	$0.6 K_{cr}$	$0.5 P_{cr}$	$0.125 P_{cr}$

#### 4.5 Perancangan Perangkat Lunak

Perancangan perangkat lunak pada penelitian ini menggunakan *compiler* arduino.ide. Sebelum menulis program pada arduino, terlebih dahulu dilakukan perancangan *flowchart* sistem. Adapun *flowchart* sistem ditunjukkan dalam Gambar 4.9.



**Gambar 4.11 Flowchart Perangkat Lunak Sistem (Perancangan, 2016)**



Pada *flowchart* perangkat lunak sistem yang dirancang, yang pertama kali dilakukan adalah inialisasi variabel. Kemudian setelah itu program akan memulai membaca keseimbangan *flyboard*, yang mana nilai keseimbangan *flyboard* akan menjadi *feedback* sistem yang akan dibandingkan dengan nilai *setpoint* sumbu x pada sudut  $180^{\circ}$ . Dari nilai *error* yang diperoleh akan diolah dengan menggunakan algoritma PID (*Proportional Integral Derivatif*). Hasil dari pemrosesan algoritma PID (*Proportional Integral Derivatif*) berupa nilai *output* yang akan digunakan sebagai sinyal kontrol menyeimbangkan *flyboard*.





## BAB V

### PENGUJIAN DAN ANALISIS

Tujuan pengujian sistem ini adalah untuk menentukan apakah alat yang telah dibuat berfungsi dengan baik dan sesuai dengan perancangan. Pengujian pada sistem ini meliputi pengujian setiap blok maupun pengujian secara keseluruhan. Pengujian setiap blok ini dilakukan untuk menemukan letak kesalahan dan mempermudah analisis pada sistem apabila alat tidak bekerja sesuai dengan perancangan. Pengujian dibagi menjadi beberapa bagian, yaitu:

1. Pengujian Rangkaian *Zero Crossing*
2. Pengujian Rangkaian Penyulutan TRIAC
3. Pengujian Sensor *Accelerometer*
4. Pengujian PID (*Proportional Integral Derivatif*)
5. Pengujian Keseluruhan Sistem

#### 5.1 Pengujian Rangkaian *Zero Crossing*

Pengujian ini bertujuan untuk mengetahui apakah rangkaian zero crossing detector sudah berfungsi dengan baik sebagai penanda titik nol gelombang AC yang digunakan arduino sebagai acuan pemicuan TRIAC.

##### 5.1.1 Peralatan Pengujian

1. Osiloskop Siglent SDS1022DL
2. Rangkaian *Zero Crossing*

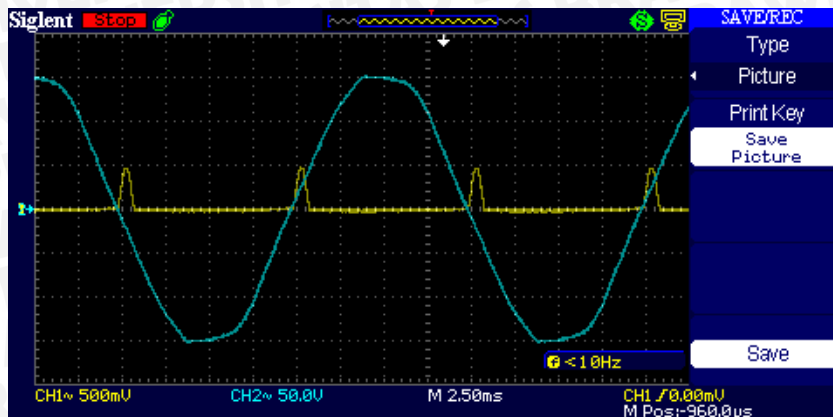
##### 5.1.2 Prosedur Pengujian

Pengujian dilakukan dengan cara menghubungkan rangkaian *zero crossing* dengan tegangan jala-jala. Kemudian tegangan jala-jala dan keluaran rangkaian *zero crossing* dihubungkan dengan probe osiloskop

##### 5.1.3 Hasil Pengujian

Hasil pengujian tegangan catu daya ditunjukkan dalam Gambar 5.1.





Gambar 5.1 Hasil Pengujian Rangkaian Zero Crossing (Pengujian, 2016)

Pengujian rangkaian *zero crossing* pada Gambar 5.1 menunjukkan gelombang tegangan AC jala-jala pada grafik yang berwarna biru. Kemudian sinyal keluaran rangkaian *zero crossing* ditunjukkan pada grafik yang berwarna kuning. Berdasarkan hasil pengujian rangkaian *zero crossing* yang ditunjukkan dalam Gambar 1 menunjukkan bahwa rangkaian *zero crossing* sudah dapat berfungsi sebagaimana mestinya. Hal ini dapat diketahui ketika tegangan AC jala-jala pada posisi 0V, maka rangkaian *zero crossing* akan mengeluarkan sinyal yang nantinya akan dikirim ke arduino sebagai penanda bahwa tegangan jala-jala pada posisi 0V.

## 5.2 Pengujian Rangkaian Penyulutan TRIAC

Pengujian ini bertujuan untuk mengetahui apakah rangkaian penyulutan TRIAC yang telah dirancang mampu mengatur gelombang sinusoida tegangan AC sebagai sumber tegangan pompa air.

### 5.2.1 Peralatan Pengujian

1. Osiloskop Siglent SDS1022DL
2. Rangkaian *Zero Crossing*
3. Rangkaian Penyulutan TRIAC
4. Arduino Uno

### 5.2.2

### Prosedur

### Pengujian

Prosedur pengujian dilakukan dengan cara menghubungkan rangkaian *zero crossing* dan penyulutan triac pada arduino. Kemudian rangkaian yang sudah terhubung satu sama lain diberikan tegangan jala-jala yang keluaran rangkaian penyulutan TRIAC akan dihubungkan dengan osiloskop untuk mengetahui sinyal yang dihasilkan.

### 5.2.3 Hasil Pengujian

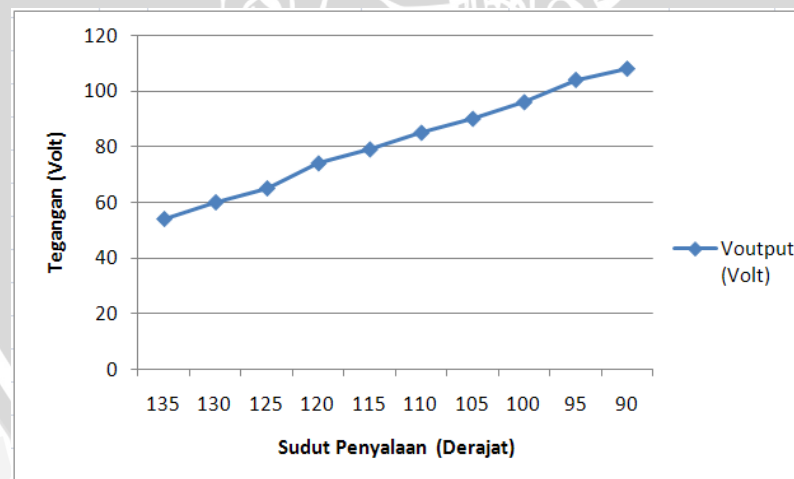
Hasil pengujian rangkaian penyulutan TRIAC ditunjukkan dalam Tabel 5.1.

Tabel 5.1 Hasil Pengujian Rangkaian Penyulutan TRIAC

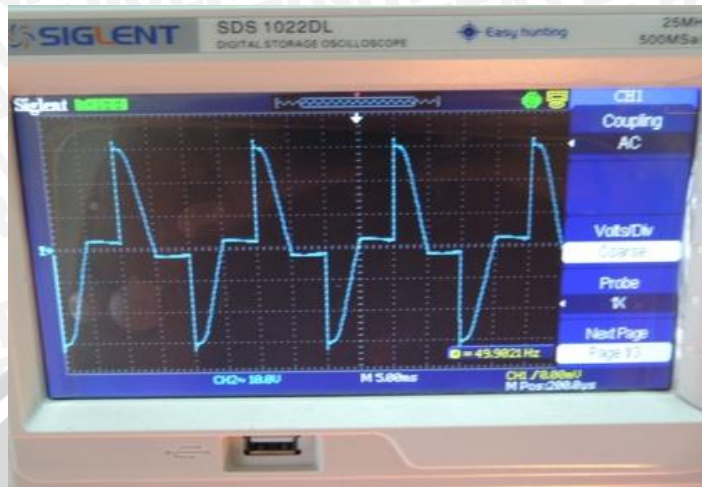
No.	Sudut Penyalaan ( $^{\circ}$ )	Vinput (Volt)	Voutput (Volt)
1	135	219	54
2	130	219	60
3	125	219	65
4	120	219	74
5	115	219	79
6	110	219	85
7	105	219	90
8	100	219	96
9	95	219	104
10	90	219	108

Gambar 5.2 Hasil Pengujian LCD 16x2 (Pengujian, 2016)

Grafik hasil pengujian penyulutan TRIAC dan bentuk gelombang keluaran TRIAC ditunjukkan dalam Gambar 5.2 dan 5.3.



Gambar 5.2 Grafik Sudut Penyalaan Terhadap Tegangan



Gambar 5.3 Sinyal Gelombang Keluaran TRIAC Pada Sudut Penyalan  $90^{\circ}$

Berdasarkan hasil pengujian penyulutan triac dapat disimpulkan bahwa triac dapat berfungsi sebagaimana mestinya. Hal ini ditunjukkan pada nilai tegangan yang terukur pada multimeter ketika sudut penyalan dirubah. Nilai tegangan semakin kecil apabila sudut penyalan besar, dan begitupun sebaliknya.

### 5.3 Pengujian Sensor *Accelerometer*

Pengujian sensor accelerometer ADXL345 bertujuan untuk mengetahui karakteristik dan fungsional dari sensor accelerometer.

#### 5.3.1 Peralatan Pengujian

1. Arduino Uno
2. Sensor *Accelerometer* ADXL345
3. Busur

#### 5.3.2 Prosedur Pengujian

Pengujian dilakukan dengan cara memberikan kemiringan pada sensor accelerometer, dimana nilai kemiringan diukur menggunakan busur derajat. Kemudian nilai kemiringan busur dibandingkan dengan nilai kemiringan yang dibaca oleh arduino. Pada pengujian kali ini hanya menguji satu axis yaitu axis x.



### 5.3.3 Hasil Pengujian

Hasil pengujian sensor *accelerometer* ditunjukkan dalam Tabel 5.2.

Tabel 5.2 Hasil Pengujian Sensor *Accelerometer* ADXL345 (Pengujian, 2016)

No	Kemiringan		Pengukuran ( <sup>o</sup> )
1	Miring Kanan	10 <sup>o</sup>	10,2
2	Miring Kanan	20 <sup>o</sup>	20
3	Miring Kanan	30 <sup>o</sup>	30,1
4	Miring Kanan	40 <sup>o</sup>	40,2
5	Miring Kiri	10 <sup>o</sup>	-10,1
6	Miring Kiri	20 <sup>o</sup>	-19,9
7	Miring Kiri	30 <sup>o</sup>	-30
8	Miring Kiri	40 <sup>o</sup>	-40,2

Berdasarkan hasil pengujian yang ditunjukkan dalam Tabel 5.2 menunjukkan bahwa *accelerometer* dapat berfungsi sebagaimana mestinya. Hal ini ditunjukkan dari pembacaan *arduino* yang mengikuti arah dan sudut kemiringan.

### 5.4 Pengujian PID (*Proportional Integral Derivatif*)

Pengujian PID (*Proportional Integral Derivatif*) bertujuan untuk menentukan parameter nilai  $K_p$ ,  $K_i$  dan  $K_d$  berdasarkan aturan dasar *Ziegler-Nichols*. Pada pengujian PID (*Proportional Integral Derivatif*) ini diharapkan *flyboard* mempunyai kestabilan.

#### 5.4.1 Peralatan Pengujian

1. *Arduino Uno*
2. Pompa Air

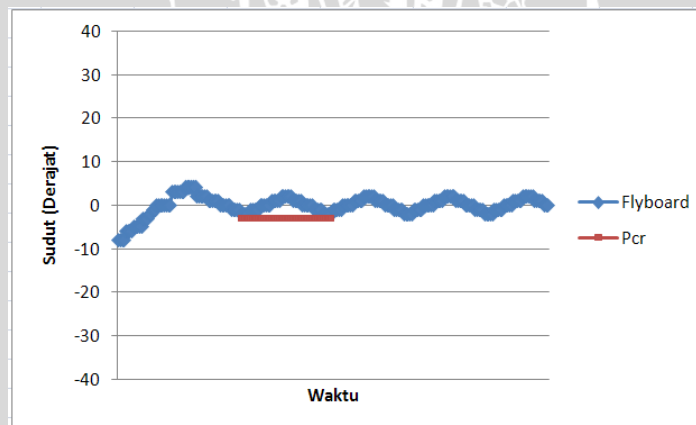
3. Rangkaian *Zero Crossing*
4. Rangkaian Penylutan TRIAC
5. Sensor Accelerometer ADXL345
5. Laptop

#### 5.4.2 Prosedur Pengujian

Pengujian PID (*Proportional Integral Derivatif*) dilakukan dengan cara memberikan nilai  $K_p$  dari nilai terkecil sampai respon mempunyai osilasi yang stabil, dimana nilai  $K_i$  dan  $K_d$  diberikan nilai 0. Setelah didapatkan grafik dengan sistem osilasi yang cukup stabil maka diperoleh parameter  $K_{cr}$  dan  $P_{cr}$ . Dimana nilai tersebut dimasukkan ke dalam tabel aturan dasar *Ziegler-Nichols* yang kemudian didapatkan nilai  $K_p$ ,  $K_i$  dan  $K_d$ .

#### 5.4.3 Hasil Pengujian

Hasil tuning PID (*Proportional Integral Derivatif*) untuk menentukan nilai  $K_{cr}$  dan  $P_{cr}$  didapatkan grafik osilasi yang ditunjukkan dalam Gambar 5.4.



Gambar 5.4 Grafik Osilasi *Flyboard*

Hasil pengujian sampai terjadi grafik osilasi yang stabil didapatkan pada nilai  $K_{cr} = 2,4$ . Pada pengujian flyboard didapatkan hasil yang cukup steady di *setpoint* sudut 180 derajat. Hal ini dikarenakan masing-masing pompa mempunyai karakteristik yang hampir sama. Sehingga pengaturan keseimbangan *flyboard* relatif mudah dilakukan. Untuk menguji apakah sistem pengontrolan berjalan dengan sesuai yang diinginkan, maka

*flyboard* perlu diberikan gangguan. Gangguan yang diberikan adalah dengan memberikan sentuhan pada *flyboard*.

Pada pengujian PID (*Proportional Integral Derivatif*) didapatkan parameter nilai sebagai berikut:

- $K_{cr} = 2,4$
- $P_{cr} = 0,47$

Hasil nilai  $K_{cr}$  dan  $P_{cr}$  dimasukkan ke dalam tabel aturan dasar *ziegler-nichols* metode 2. Maka diperoleh nilai  $K_p$ ,  $K_i$  dan  $K_d$  sebagai Berikut:

- $K_p = 0,6 \times K_{cr}$   
 $= 0,6 \times 2,4$   
 $= 1,44$
- $K_i = 0,5 \times P_{cr}$   
 $= 0,5 \times 0,47$   
 $= 0,235$
- $K_d = 0,125 \times P_{cr}$   
 $= 0,125 \times 0,47$   
 $= 0,058$

## 5.5 Pengujian Keseluruhan Sistem

Pengujian keseluruhan sistem bertujuan untuk mengetahui karakteristik dari *flyboard* ketika dikontrol menggunakan metode PID (*Proportional Integral Derivatif*) dengan parameter  $K_p$ ,  $K_i$  dan  $K_d$  yang sebelumnya diperoleh menggunakan metode 2 *ziegler-nichols*.

### 5.4.1 Peralatan Pengujian

1. Arduino Uno
2. Pompa Air
3. Rangkaian *Zero Crossing*
4. Rangkaian Penylutan TRIAC
5. Sensor Accelerometer ADXL345
5. Laptop

### 5.4.2

### Prosedur

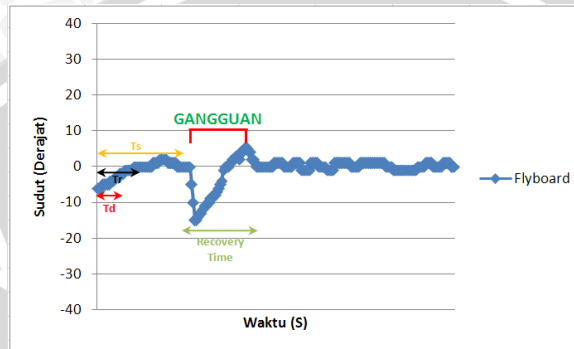
### Pengujian



Pengujian dilakukan dengan cara mengaktifkan *flyboard* dan kemudian ketika respon sistem mulai *steady*, *flyboard* diberikan gangguan dengan cara menyentuh *flyboard* agar keseimbangan *flyboard* terganggu.

### 5.4.3 Hasil Pengujian

Hasil Pengujian ditunjukkan dalam Gambar 5.5



Gambar 5.5 Grafik Sudut Terhadap Waktu

Berdasarkan hasil pengujian sistem keseluruhan didapatkan parameter  $T_d=0,11S$ ,  $T_r=0,23S$ ,  $T_s=0,5S$ ,  $recovery\ time=0,62s$ . Pada pengujian keseluruhan dapat diketahui ketika *flyboard* diberikan gangguan, respon sistem dapat kembali menuju ke titik *setpoint* sumbu  $x\ 180^0$  dan dapat kembali *steady*.

## BAB VI PENUTUP

### 6.1 Kesimpulan

1. Sistem pengendalian kestabilan miniatur *flyboard* dengan menggunakan metode PID (*Proportional Integral Derivatif*) *zielger-nichols* dengan plan yang dikontrol kestabilan *flyboard* menggunakan sensor accelerometer ADXL345 sebagai sensor keseimbangan. Didapatkan hasil akhir dengan parameter  $K_p=1.44$ ,  $K_i=0.235$  dan  $K_d=0.058$ . Pada pengujian keseluruhan didapatkan respon keseimbangan yang mempunyai  $T_d=0,11S$ ,  $T_r=0,23S$ ,  $T_s=0,5S$ ,  $recovery\ time=0,62s$ .
2. Penentuan parameter kontrol PID (*Proportional Integral Derivatif*) dengan menggunakan metode *ziegler-nichols* untuk mendapatkan nilai  $K_p$ ,  $K_i$  dan  $K_d$ . Pencarian parameter diawali dengan cara tuning nilai  $K_c$  dari kecil kemudian dinaikkan secara perlahan sampai terjadi osilasi yang mempunyai periode ( $P_c$ ) yang stabil / tetap. Kemudian nilai  $K_c$  dan  $P_c$  dimasukkan dalam aturan dasar *Ziegler-Nichols*.

### 6.2 Saran

Dalam perancangan dan pembuatan alat ini masih terdapat beberapa kelemahan. Untuk memperbaiki kinerja *flyboard* dan pengembangan lebih lanjut disarankan:

1. Menggunakan aktuator perputaran knob untuk menghasilkan hasil keseimbangan yang lebih baik.
2. Menggunakan metode lain selain PID (*Proportional Integral Derivatif*) sebagai perbandingan respon keseimbangan *flyboard*.
3. Melakukan pengembangan penelitian tentang pengontrolan terhadap 3 axis keseimbangan.





