

**PENGEMBANGAN INSTRUMEN PENGUKURAN
KETINGGIAN PESAWAT UDARA NIRAWAK (PUNA)
MENGUNAKAN SENSOR TEKANAN BAROMETRIK**

SKRIPSI

Disusun untuk melengkapi syarat-syarat guna memperoleh gelar
Sarjana Sains



MUHAMMAD ERWIN

3225061789

**PROGRAM STUDI FISIKA
JURUSAN FISIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS NEGERI JAKARTA**

2011

SURAT PERNYATAAN KEASLIAN SKRIPSI

Dengan ini saya yang bertanda dibawah ini, Mahasiswa Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Negeri Jakarta :

Nama : Muhammad Erwin

No.Reg : 3225061789

Jurusan : Fisika

Program Studi : Fisika

Menyatakan bahwa skripsi yang berjudul “PENGEMBANGAN INSTRUMEN PENGUKURAN KETINGGIAN PESAWAT UDARA NIRAWAK (PUNA) MENGGUNAKAN SENSOR TEKANAN BAROMETRIK” adalah:

1. Dibuat dan diselesaikan oleh saya sendiri, berdasarkan data yang diperoleh dari hasil penelitian pada bulan Oktober 2010 sampai dengan Pebruari 2011.
2. Bukan merupakan duplikat skripsi yang pernah dibuat oleh orang lain atau jiplakan karya tulis orang lain atau bukan terjemahan karya tulis orang lain.

Pernyataan ini saya buat dengan sesungguhnya dan saya bersedia menanggung segala akibat yang timbul jika pernyataan saya ini tidak benar.

Jakarta, Juli 2011

Yang membuat pernyataan

Muhammad Erwin

HALAMAN PERSEMBAHAN

*Kupersembahkanskripsiini kepadaOrangtuakutercinta,
kakakdankeuarga kutersayang. Atasdukungan danperhatian yang
sangatbesardanberharga.*

ABSTRAK

Muhammad Erwin. Pengembangan Instrumen Pengukuran Ketinggian Pesawat Udara NirAwak (PUNA) Menggunakan Sensor Tekanan Barometrik. Skripsi, Jakarta : Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Negeri Jakarta, 2011.

Dalam Pesawat Udara NirAwak (PUNA), salah satu parameter terpenting bagi sistem navigasinya adalah pengukuran ketinggian yang akurat. Dibutuhkan suatu perangkat pengukuran yang memiliki keakurasian yang baik. Pengaplikasian sensor tekanan barometrik dapat digunakan sebagai perangkat pengukur ketinggian PUNA. Perbandingan pengukuran ketinggian menggunakan sensor barometrik, GPS dan tabel ISA (International Standard Atmospheric) dapat digunakan untuk mengetahui *squared error* yang terjadi pada pengukuran, serta keakurasian pengukuran. Pengukuran ketinggian paling mendekati tabel ISA pada penelitian ini adalah setelah perangkat diberi low-pass filter dan juga kalman filter ($w=0.00001$) dengan RMSE (Root Mean Square Error) sebesar 5.83 yang merupakan hasil reduksi error sebesar 92.45%, dan keakurasian pengukuran tertinggi sebesar 99.86%.

Kata kunci : pengukuran ketinggian, sensor tekanan barometrik, sistem navigasi, PUNA

ABSTRACT

Muhammad Erwin. Development of Altitude Measurement Instruments in the unmanned aerial vehicle (UAV) Using Barometric Pressure Sensor. Thesis, Jakarta: Faculty of Mathematics and Natural Sciences, State University of Jakarta, 2011.

In unmanned aerial vehicle (UAV), one of the most important parameter for the navigation system is an accurate altitude measurements. It takes a measurement device that has a good accuracy. Application barometric pressure sensor can be used as measurement tools altitude. comparative measurement of altitude using barometric sensor, GPS and chart of ISA (International Standard Atmospheric) can be used to determine the squared error that occurs in measurement, and accuracy of measurement. Altitude measurement table closest to the ISA in this study is that after the device is given a low-pass filter and Kalman filter ($w = 0.00001$) with RMSE (Root Mean Square Error) of 5.83 which is the result of an error reduction of 92.45%, and the highest measurement accuracy of 99.86%.

Key words: measuring altitude, barometric pressure sensor, navigation system, UAV

KATA PENGANTAR

Puji dan syukur dihaturkan kepada Allah swt atas ilmu pengetahuan dan nikmat kesehatan yang diberikan kepada penulis, sehingga skripsi dengan judul “Pengembangan Instrumen Pengukuran Ketinggian Pesawat Udara NirAwak (PUNA) Menggunakan Sensor Tekanan Barometrik” dapat terselesaikan.

Selesainya penulisan skripsi ini tidak lepas dari bantuan dan dukungan banyak pihak. Oleh karena itu penulis ingin menyampaikan penghargaan dan ucapan terimakasih sebesar – besarnya kepada:

1. Dr.rer.nat Bambang Heru Iswanto, selaku pembimbing akademik, dosen pembimbing dan ketua jurusan Fisika FMIPA UNJ yang telah memberikan waktu, pengalaman, serta pengetahuannya.
2. Dr.Ir. Abdul Aziz, M.Sc, selaku dosen pembimbing II, serta Tim PUNA PTIPK-BPPT yang telah memberikan waktu, pengalaman, serta pengetahuannya.
3. Bapak dan Ibu dosen fisika yang telah memberikan kuliah selama masa perkuliahan berlangsung.
4. Ibu dan Ayah tercinta, atas kasih sayangnya yang tidak tergantikan dan terbalaskan.
5. Teman-temanku angkatan 2006; Adinda, Aminudin, Giri, Yuda, Ardi , Ato, MasJay, Priyo, Azis, dll untuk pengalamannya.
6. Teman – teman di lab instrumen dan komputasi, Hastoro, Arief budiman, Eka raga, Aditya I, Awaludin, Haikel, bang Fajrin, serta Teman – teman fisika dan pendidikan fisika angkatan 2004, 2005, 2006, 2007, 2008 Untuk Persahabatan dan persaudaraan.

Jakarta, Juli 2011

Muhammad Erwin

DAFTAR ISI

ABSTRAK.....	i
ABSTRACT.....	ii
KATA PENGANTAR	iii
DAFTAR ISI.....	iv
DAFTAR GAMBAR	vi
DAFTAR TABEL.....	vii
DAFTAR LAMPIRAN.....	viii
BAB IPENDAHULUAN.....	1
A. Latar Belakang Masalah.....	1
B. Identifikasi Masalah.....	2
C. Pembatasan Masalah.....	2
D. Perumusan Masalah	3
E. Tujuan Penelitian	3
F. Manfaat Penelitian	3
BAB I IKAJIAN TEORI	4
A. Alat Ukur Ketinggian (Altimeter).....	4
1.1. Tekanan Udara (Barometric Pressure)	4
1.2. Gelombang (Ultrasonik, Infra merah, maupun lainnya)	5
B. Mekanika Fluida	5
C. Pesawat Udara NirAwak (PUNA)	10
D. Sistem Akuisisi Data	12
4.1 Sumber.....	12
4.2 Sensor Tekanan.....	12
4.3 Analog To Digital Converter (ADC).....	14
4.4 Rangkaian Low-pass Filter	15
4.5 Perangkat Keras Sistem Akuisisi Data	16
4.6 Perangkat Lunak Sistem Akuisisi Data	18
E. Kalman Filter	20
BAB IIIMETODOLOGI PENELITIAN	26
A. Tujuan Operasional.....	26
B. Waktu dan Tempat Penelitian.....	26
C. Metode Penelitian	26
D. Instrumen Peneitian	27

E. Diagram Alir Penelitian	28
F. Skema Rangkaian Perangkat Pengukuran.	29
BAB IV HASIL PENELITIAN DAN PEMBAHASAN.....	30
A. Pengembangan Instrumen	30
1.1 Perancangan Instrumen	30
1.2 Kalibrasi Instrumen	30
1.3 Pengintegrasian Kalman Filter	32
1.4 <i>Graphics User Interface</i> Pengukuran.....	33
B. Hasil Pengujian dan Pembahasan	33
C. Identifikasi Pengukuran	38
3.1 Kesalahan Rata-rata Pengukuran (<i>RootMeanSquareError</i>).....	38
3.2 Reduksi Kesalahan (<i>ErrorReduction</i>)	39
3.3 Tingkat Keakuratan Pengukuran	40
BAB V KESIMPULAN DAN SARAN	41
A. Kesimpulan	41
B. Saran..	42
DAFTAR PUSTAKA	43
LAMPIRAN.....	44
A. Gambar Alat.....	44
B. Hasil Pengukuran	44
2.1 Hasil Pengukuran Tanpa Filter.....	44
2.2 Hasil Pengukuran dengan Filter	45
C. List Program.....	47
3.1 List Program pada Mikrokontroler	47
3.2 List Program pada GCS PUNA dengan Kalman Filter.....	51

DAFTAR GAMBAR

Gambar 1	Altimeter pesawat barometrik.....	5
Gambar 2	Gaya normal yang menghasilkan tekanan	6
Gambar 3	Tekanan absolute dan gauge	7
Gambar 4	PUNA yang dikembangkan oleh PTIPK – BPPT.....	10
Gambar 5	Skemasistemstandar PUNA	11
Gambar 6	MPX 4115 AP	13
Gambar 7	Kurva linearitas tekanan dan tegangan pada MPX4115AP	14
Gambar 8	Skema Sistem Standar Mikrokontroler	17
Gambar 9	Mikrokontroler ATmega 8535	17
Gambar 10	K-125i	18
Gambar 11	Siklus proses operasi kalman filter	23
Gambar 12	Gambaran lengkap operasi kalman filter	24
Gambar 13	Gambar Rangkaian Perangkat Pengukuran Ketinggian.....	30
Gambar 14.1	Perangkat pengukuran ketinggian	31
Gambar 14.2	Karakteristik sensor MPX4115AP	31
Gambar 15	<i>Graphics User Interface</i> monitoring Pesawat.....	33
Gambar 16.1	Grafik Ketinggian sebelum diberi filter	34
Gambar 16.2	Grafik Ketinggian (Low-pass) sebelum diberi kalman filter	34
Gambar 17	Grafik Ketinggian setelah diberi kalman filter ($w=0.1$).....	35
Gambar 18	Grafik Ketinggian setelah diberi kalman filter ($w=0.01$).....	35
Gambar 19	Grafik Ketinggian setelah diberi kalman filter ($w=0.001$).....	36
Gambar 20	Grafik Ketinggian setelah diberi kalman filter ($w=0.0001$).....	36
Gambar 21	Grafik Ketinggian setelah diberi kalman filter ($w=0.00001$)....	37

DAFTAR TABEL

Tabel1	RMSE.....	39
Tabel2	Reduksi error.....	40
Tabel3	Keakurasian pengukuran	40

DAFTAR LAMPIRAN

A. Gambar Alat	44
B. Hasil Pengukuran	44
2.1. Hasil Pengukuran Tanpa Filter	44
2.2. Hasil Pengukuran dengan Filter	45
C. List Program	47
3.1. List Program pada Mikrokontroler	47
3.2. List Program pada GCS PUNA dengan Kalman Filter	48

BAB I

PENDAHULUAN

A. Latar Belakang Masalah

Dewasa ini pertumbuhan di bidang fisika instrumentasi sangat berkembang pesat. Seperti perkembangan dibidang mikrokontroler, Robotik, dan juga sistem automatisasi. Di Indonesia sendiri perkembangan dibidang instrumentasi saat ini sangat berkembang, terlebih departemen pemerintahan juga ikut mengembangkan. Seperti Badan Pengkajian dan Penerapan Teknologi (BPPT) yang telah banyak mengembangkan penelitian dibidang ini. Salah satu dari produk pengembangan dari BPPT di bidang ini adalah Pesawat Udara NirAwak(PUNA), yang didalamnya terdapat pengembangan sistem instrumentasi.

PUNA *Avionics and Payload System* terdiri dari *OnBoard System* dan *Ground Control System*. Untuk *OnBoard System* terdiri dari *Flight Control Subsystem* dan *Payload System*. Untuk *Ground Control System* terdiri dari *Pilot Control System* dan *Observer Control System* [6].

Dalam *OnBoard System (OBS)* terdapat bagian *flight control subsystem* yang terdiri dari ADAHRS dan *Control unit* yang berfungsi untuk parameter kontrol penerbangan. ADAHRS adalah bagian sensor-sensor telemetri untuk dijadikan parameter navigasi penerbangan, sedangkan GPS untuk mengetahui lokasi PUNA itu sendiri. Serta terdapat pula antena *transmitter data links* yang berfungsi untuk mengirimkan data-data dari ADAHRS ke *Ground Control System*.

Salah satu bagian dari sistem telemetri adalah *altimeter* yang berfungsi untuk pengukuran ketinggian pada PUNA. *Altimeter* yang digunakan pada PUNA ini menggunakan sensor Tekanan Barometrik. Pengukuran ketinggian ini sangat penting untuk salah satu parameter sistem navigasi PUNA, jika salah

ataupun kurang akurat dalam pengukuran ketinggian bisa berakibat salah mendarat, maupun bertabrakan dengan bukit atau gunung. Salah satu cara untuk peningkatan keakuratan pengukuran ketinggian adalah dengan menambahkan rangkaian *low – pass filter* pada perangkat integrasi sensor Tekanan Barometrik, serta ditambahkan *kalman filter* sebagai filter digital agar akurasi pengukuran menjadi lebih baik lagi.

B. Identifikasi Masalah

Berdasarkan uraian latar belakang dapat diidentifikasi beberapa masalah-masalah sebagai berikut :

1. Bagaimana pemilihan sensor Tekanan Barometrik yang sesuai dengan kebutuhan sistem navigasi PUNA?.
2. Bagaimana sistem antar muka sensor Tekanan Barometrik?.
3. Bagaimana karakteristik sensor Tekanan Barometrik yang digunakan untuk pengukuran ketinggian pada PUNA?.
4. Bagaimana merancang perangkat untuk mengukur ketinggian pada PUNA menggunakan sensor Tekanan Barometrik?.
5. Bagaimana keakuratan pengukuran ketinggian menggunakan sensor Tekanan Barometrik?.
6. Bagaimana pengintegrasian *Low-pass Filter* dan *kalman filter* sebagai filter digitalnya pada perangkat pengukuran?.

C. Pembatasan Masalah

Dalam penelitian ini, permasalahan hanya dibatasi pada perancangan perangkat untuk mengukur ketinggian dengan menggunakan sensor Tekanan Barometrik, serta tingkat keakuratan perangkat pengukurannya setelah diberi *low-pass Filter* dan *kalman filter* sebagai filter digitalnya.

D. Perumusan Masalah

Permasalahan yang akan dikaji pada penelitian ini adalah bagaimana merancang bangun perangkat pengukuran ketinggian dengan akurasi yang optimal.

E. Tujuan Penelitian

Merancang dan membangun perangkat pengukuran ketinggian yang berakurasi tinggi untuk parameter sistem navigasi yang optimal.

F. Manfaat Penelitian

1. Pengukuran ketinggian yang akurat dapat menjadi parameter yang optimal untuk sistem navigasi PUNA.
2. Pengukuran ketinggian yang akurat akan memudahkan bagi pengembangan di bagian desain sistem kontrol PUNA.
3. Sebagai salah satu referensi pengintegrasian *low-pass filter* dan kalman filter pada perangkat pengukuran *altitude*.
4. Sebagai salah satu input parameter untuk otomatisasi PUNA.
5. Sebagai referensi penggunaan sensor Tekanan Barometrik untuk pengukuran ketinggian pada PUNA.

BAB II

KAJIAN TEORI

A. Alat Ukur Ketinggian (Altimeter)

adalah alat untuk mengukur ketinggian suatu titik dari permukaan laut (*altitude*). Biasanya alat ini digunakan untuk keperluan navigasi dalam penerbangan, pendakian, dan kegiatan yang berhubungan dengan ketinggian [2]. *Altimeter* bekerja dengan beberapa metode, yaitu [2] :

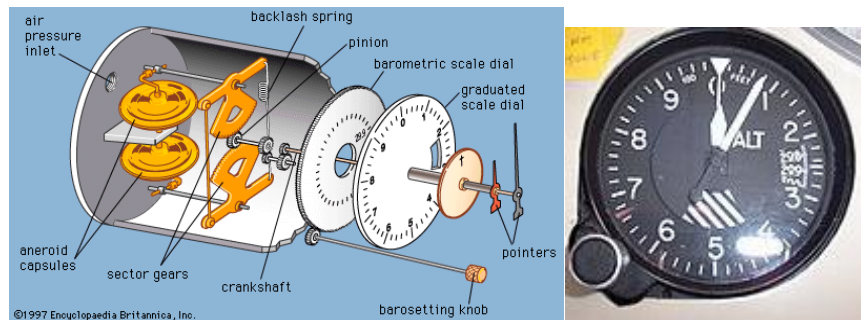
1.1. Tekanan Udara (Barometric Pressure)

Altimeter jenis ini umumnya ditemui pada pesawat udara. Altimeter *barometric* pada pesawat umumnya memakai mekanika mesin biasa (bukan sensor), dalam suatu altimeter terletak peralatan yang sama ditemukan dalam barometer. mereka mengukur jumlah tekanan udara statis diberikan pada altimeter untuk menentukan setinggi di atas permukaan laut. tekanan udara statis mengukur jumlah tekanan yang diberikan oleh suasana di ketinggian tertentu.

Altimeter tekanan beroperasi pada prinsip bahwa rata-rata tekanan atmosfer menurun secara linier dengan ketinggian. Altimeter harus dikalibrasi untuk menunjukkan langsung sebagai ketinggian di atas permukaan laut, sesuai dengan model matematis didefinisikan oleh International Standard Atmosphere (ISA). Pesawat yang lebih tua menggunakan barometer aneroid sederhana di mana jarum dibuat kurang dari satu revolusi di sekitar wajah dari nol sampai skala penuh. Pesawat modern menggunakan "altimeter sensitif" yang memiliki jarum utama yang membuat beberapa revolusi, dan satu atau lebih jarum sekunder yang menunjukkan jumlah revolusi, mirip dengan tampilan jam. Dengan kata lain, setiap jarum menunjuk ke sebuah angka yang berbeda dari pengukuran ketinggian saat ini.

Pada sebuah altimeter sensitif, permukaan laut tekanan referensi dapat disesuaikan dengan pengaturan kenop (*knob*). Tekanan referensi, dalam inci

merkuri di Kanada dan Amerika Serikat dan *hectopascals* (sebelumnya milibar) di tempat lain, akan ditampilkan di jendela Kollsman, terlihat di sisi kanan altimeter pesawat yang ditampilkan di sini. Hal ini diperlukan, karena permukaan laut tekanan referensi atmosfer bervariasi dengan suhu dan pergerakan sistem tekanan di atmosfer.



Gambar 1 Altimeter pesawat barometrik

1.2. Gelombang (Ultrasonik, Infra merah, maupun lainnya)

Altimeter radio mengukur jarak pesawat terhadap tanah bukan terhadap permukaan laut. Ketinggian sama dengan satu setengah waktu yang dibutuhkan sebuah pulsa energi gelombang dari pesawat ke tanah dan kembali dikalikan dengan kecepatan pulsa (setara dengan kecepatan cahaya).

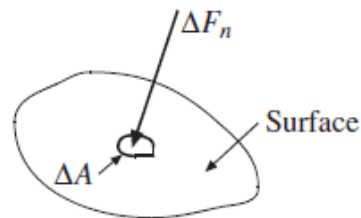
B. Mekanika Fluida

Fluida merupakan zat yang dapat mengalir dan mempunyai partikel yang mudah bergerak serta berubah bentuk tanpa pemisahan massa. Ketahanan fluida terhadap perubahan bentuk sangat kecil sehingga fluida dapat dengan mudah mengikuti bentuk ruang. Dalam penelitian ini jenis fluida yang dibahas adalah udara [1].

Dalam mekanika fluida kita sering menemui tekanan, yang dihasilkan oleh gaya per satuan luas yang bekerja tegak lurus terhadap permukaan. Secara umum didefinisikan sebagai [1] :

$$P = \lim_{\Delta A \rightarrow 0} \frac{\Delta F}{\Delta A} \quad 1$$

dimana, P = Tekanan (Pascal), A = Luas permukaan (m^2), F = Gaya (Newton)



Gambar 2 Gaya normal yang menghasilkan tekanan

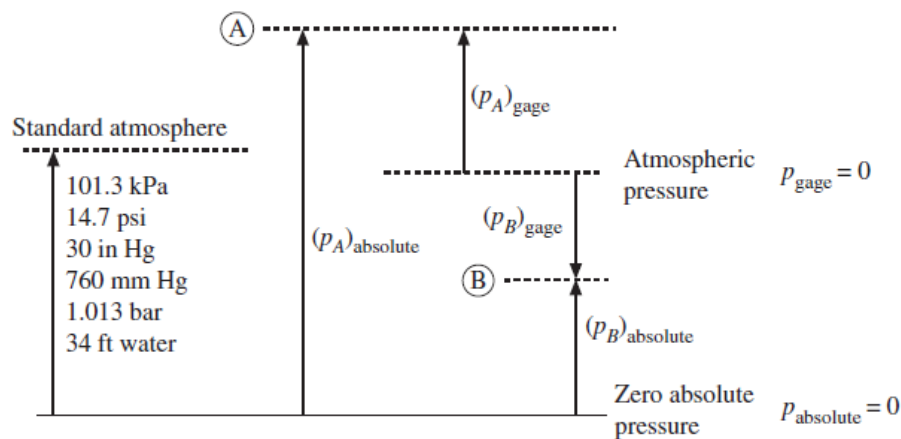
Dalam konsep ini tekanan didefinisikan sebagai gaya yang diberikan oleh fluida pada tempat yang memadahnya. Meskipun tekanan bernilai mutlak, pengukuran tekanan biasanya dibuat relatif terhadap tekanan udara ambient. Dalam kasus lain tekanan dibuat relatif terhadap tekanan vakum atau ke referensi lainnya. Berikut adalah istilah-istilah tekanan berdasarkan referensinya:

- **Absolute Pressure (tekanan absolut)** adalah nilai mutlak tekanan yang bekerja pada wadah tersebut yang merupakan referensi nol terhadap sebuah keadaan vakum yang sempurna. Jadi tekanan ini setara dengan tekanan gauge ditambah dengan tekanan atmosfer.

$$P_{\text{absolute}} = P_{\text{gauge}} + P_{\text{atm}} \quad 2$$

- **Gauge Pressure (tekanan relatif)** adalah Tekanan yang dinyatakan dan diukur relatif terhadap tekanan atmosfer. Jadi tekanan relatif adalah selisih antara tekanan absolute dengan tekanan atmosfer (1 atmosfer = 760 mmHg = 14.7 psia)
- **Differential Pressure (tekanan differential)** adalah Tekanan yang diukur terhadap tekanan yang lain.

- **Vacuum pressure (tekanan vakum)** adalah tekanan terhadap tekanan ruang vakum [3]



Gambar 3 Tekanan absolute dan gauge

Jenis pengukuran tekanan yang digunakan untuk pengukuran ketinggian adalah pengukuran *atmospheric pressure / barometric pressure*. Semakin besar ketinggian, semakin rendah tekanan. Ketika barometer diberikan dengan kalibrasi nonlinier sehingga untuk menunjukkan ketinggian, instrumen yang disebut altimeter tekanan atau altimeter barometric. Sebuah altimeter tekanan adalah altimeter yang digunakan hanya pada pesawat, dan skydivers menggunakan pergelangan versigenggam tujuan serupa. Pejalan kaki dan pendaki gunung menggunakan pergelangan tangan-mount atau altimeter genggam, selain alat navigasi lainnya seperti peta, kompas magnetik, atau penerima GPS.

Tekanan dapat didefinisikan sebagai ketinggian kolom fluida vertikal terhadap penampang silang dengan basisnya pada titik keseimbangan. Biarkan p menjadi tekanan pada ketinggian z dalam kolom gas. Kemudian [2]:

$$dp = -\rho g dz . \quad 3$$

dimana :

dp = perubahan tekanan setiap perubahan ketinggian dari dz

ρ = densitas fluida pada ketinggian z

g = percepatan terhadap gravitasi.

ρ adalah fungsi dari z.

Dari persamaan gas ideal [3]:

$$pv = nRT \quad 4$$

dimana :

p = tekanan

v = volume

T = Temperatur *Ambient*

R = konstanta gas 287.058 J/(kg·K)

n = the number of moles of fluid in the volume v.

Dengan mensubstitusikan persamaan 3 dan 4 menjadi :

$$\frac{dp}{dz} = -\frac{g}{RT} p \quad 5$$

Untuk T dan g konstan persamaannya menjadi :

$$z = -\frac{RT}{g} \log \frac{P}{P_0} \quad 6$$

Persamaan diatas dikenal dengan persamaan *hypsometric*, bergantung terhadap perubahan tekanan, serta mengasumsikan bahwa :

1. Suhu (T) adalah konstan
2. Percepatan gravitasi (g) adalah konstan.
3. *zero lapse rate* (L = 0).

Persamaan *hypsometric* hanya digunakan untuk pengukuran ketinggian statis misal pengukuran ketinggian gunung dimana gunung itu statis tidak

bergerak. Maka untuk pengukuran ketinggian untuk suatu obyek yang bergerak diperlukan koreksi pada suhu (T) dan *lapse rate* (laju perubahan suhu terhadap ketinggian), sehingga suhu adalah sebuah variable bukan konstan. Tetapi untuk pengukuran ketinggian perubahan suhu adalah linear menurut standar *International Standard Atmosphere* (ISA) dan *International Civil Aviation Organization* (ICAO) , yaitu :

$$L = \frac{-6.5^{\circ}C}{1000m} \quad 7$$

Maka persamaan ketinggian menjadi :

$$\frac{dp}{dz}[z] = -\frac{1}{R} \frac{g[z]}{T[z]} P[z] \quad 8$$

$$P[z] == P_0 \text{Exp}\left(-\frac{1}{R} \left(\int_0^z \frac{g[\zeta]}{T[\zeta]} d[\zeta]\right)\right) \quad 9$$

Dengan ζ hanya sebuah variable untuk memudahkan integrasi. Asumsikan g konstan karena jika masih dibawah tekanan 1atm koreksi nilai g hanya 5%, serta memasukan koreksi suhu (L), sehingga persamaan menjadi :

$$g \int_0^z \frac{d\zeta}{T_0 + L\zeta} == \frac{g}{L} (\log [T_0 + L\zeta] - \log[T_0]) \quad 10$$

Substitusikan persamaan diatas kepada persamaan P[z], maka persamaan pengukuran ketinggian (z) menjadi:

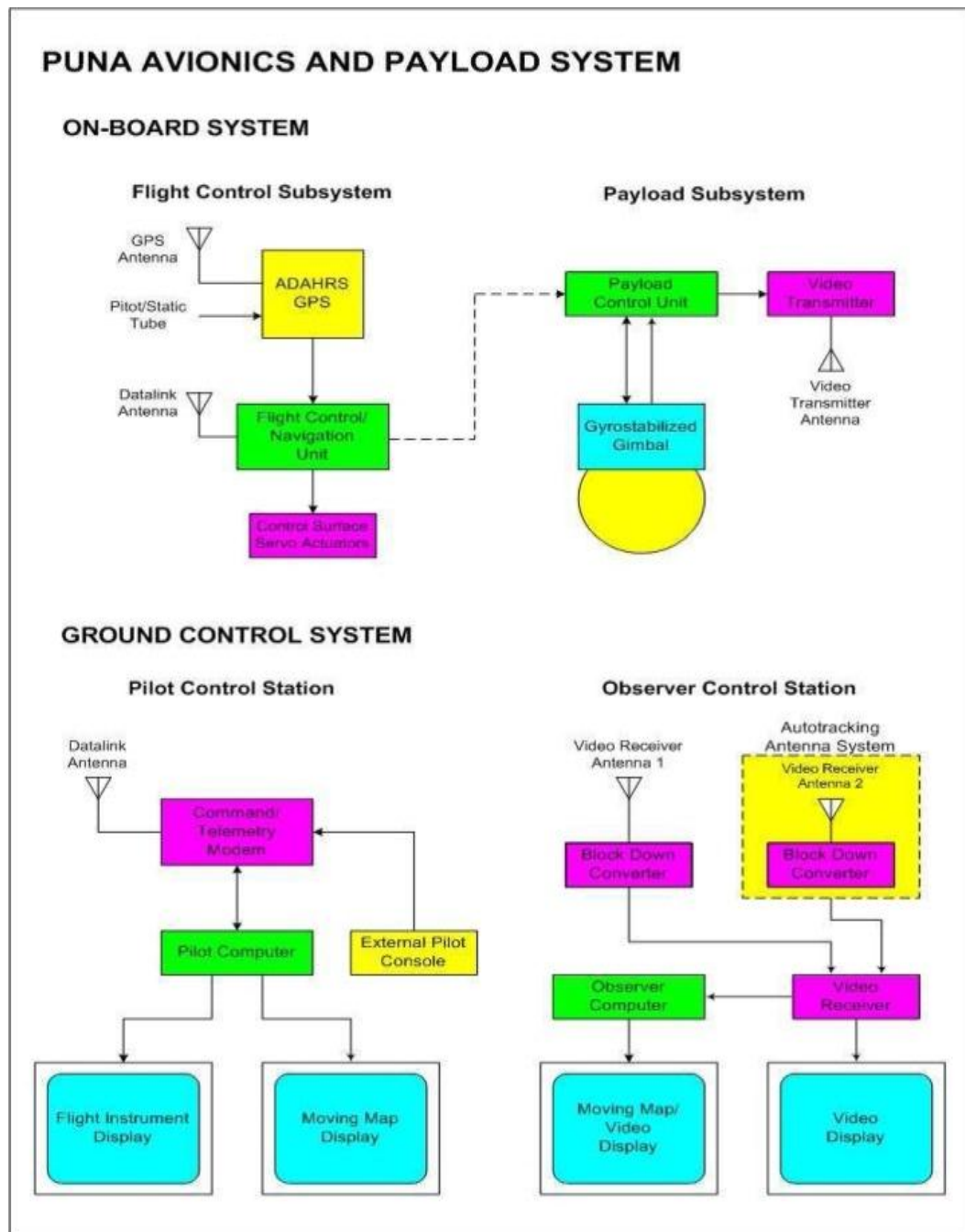
$$z == -\frac{T_0}{L} \left(\left(\frac{P}{P_0}\right)^{-L\frac{R}{g}} - 1 \right) \quad 11$$

C. Pesawat Udara NirAwak (PUNA)

Kendaraan Udara Tanpa Awak (Unmanned Aerial Vehicles=UAVs) adalah pesawat yang dikendalikan dengan remote atau “self-piloted” (mengendalikan diri sendiri) yang dapat membawa kamera, sensor, peralatan komunikasi atau peralatan lainnya. Mereka digunakan untuk peran pengintaian dan pengumpulan data intelejen sejak 1950an, dan peran lebih menantang juga diharapkan, termasuk misi pertempuran[4].Berikut adalah sistem standar dari PUNA sebagai berikut [6] :



Gambar 4 PUNA yang dikembangkan oleh PTIPK – BPPT



Gambar 5 Skemasistemstandar PUNA

PUNA memiliki sistem standar seperti diatas. Yaitu memiliki dua sistem secara keseluruhan, yaitu: (a) *OnBoard System* (OBS). Didalam *On Board System* terdapat bagian *flight control subsystem* yang terdiri dari ADAHRS, *Engine Monitoring Unit* (EMU), *Control unit* yang berfungsi untuk parameter kontrol

penerbangan. ADAHRS adalah bagian sensor-sensor telemetri untuk dijadikan parameter penerbangan, sedangkan GPS untuk mengetahui lokasi PUNA itu sendiri. Serta terdapat pula antena *transmitter data links* yang berfungsi untuk mengirimkan data-data dari ADAHRS ke *Ground Control System*. *Control unit* pada PUNA menggunakan *Rabbit Microcontroller*, yang memiliki kemampuan untuk mengontrol system dengan sangat baik. Dalam hal ini menggunakan seri RCM 4100. Sedangkan EMU menggunakan mikrokontroler jenis AVR untuk mengolah data- data keluaran dari sensor. (b) *Ground Control System* (GCS) yang secara umum adalah pusat kendali data, serta pusat kendali navigasi pada PUNA. Terdiri dari *pilot control station* dan *observer control station* [6].

D. Sistem Akuisisi Data

Akuisisi data merupakan proses sampling dari kondisi besaran fisis dan konversi dari sampel yang dihasilkan menjadi data digital yang dapat dimanipulasi oleh komputer. Sistem akuisisi data biasanya melibatkan konversi bentuk gelombang analog menjadi data digital untuk diproses.

4.1 Sumber

Perolehan data dimulai dengan mengukur Tekanan Barometrik. Oleh karena itu dibutuhkan suatu sensor yang merupakan jenis transduser, yaitu sebuah perangkat yang mengubah besaran fisis menjadi sinyal listrik yang sesuai. Jenis sensor yang digunakan adalah sensor tekanan *barometric*. Data yang diambil kemudian diolah dengan mikrokontroler sehingga data keluaran berupa data ketinggian PUNA.

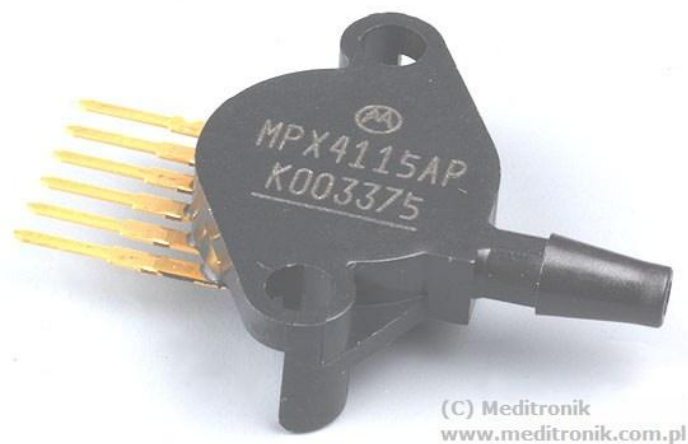
4.2 Sensor Tekanan

Sensor tekanan adalah sensor atau transducer yang mengukur tekanan. Sensor tekanan mempunyai ciri – ciri sebagai berikut :

- Bahan elastik dan sensor perpindahan (*displacement*),
- Besaran ukuran strain (tegang) atau *displacement* (perpindahan)
- Pengelompokan : tipe *absolute*, *gauge*, dan *differential*

Salah satu transduser tekanan yang paling sensitif adalah transduser tekanan yang berjenis piezoresistif. Sensor ini terdiri dari empat Piezoresistor semikonduktor yang hampir identik diletakkan di permukaan diafragma yang terbuat dari bahan silikon tipis yang melingkar. Tekanan menyebabkan diafragma untuk membungkuk, mendorong penekanan pada diafragma dan Piezoresistor dikuburkan.

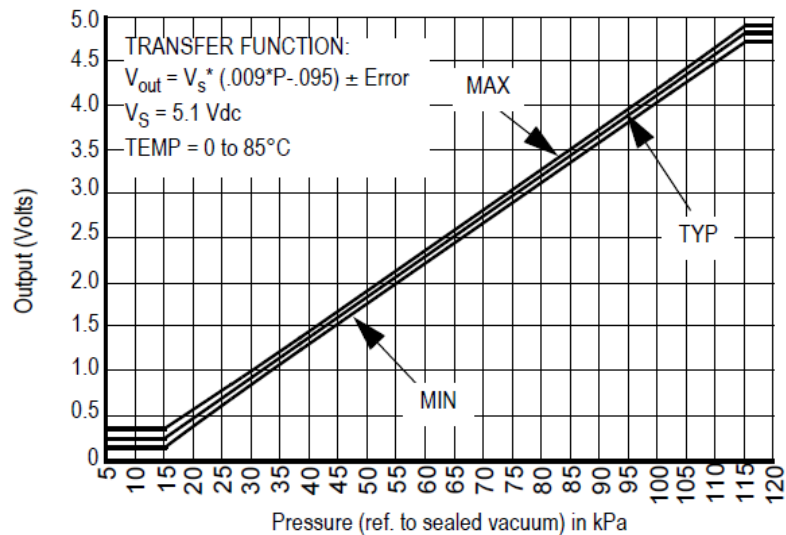
Sensor tekanan yang akan digunakan pada penelitian ini adalah sensor MPX 4115 AP. Sensor MPX 4115 AP ini bertipe *absolute Pressure* atau Tekanan Barometrik. Seri MPX4115AP dirancang untuk merasakan tekanan udara mutlak dalam altimeter atau aplikasi barometer (BAP). Sensor BAP mengintegrasikan *on-chip* sirkuit op – amp bipolar dan jaringan film tipis resistor untuk memberikan level sinyal analog output tinggi dan kompensasi suhu. Faktor sensor yang berukuran kecil dan kehandalan yang tinggi membuat integrasi *on-chip* BAP sensor menjadi pilihan yang logis dan ekonomis bagi para teknisi.



Gambar 6 MPX 4115 AP

Sensor MPX 4115 AP bekerja jika diberi tegangan sebesar 5 V DC dan memerlukan arus sebesar 10 mA DC. Sensor ini dapat mendeteksi beda tekanan dari 15 sampai 115 kPa. Tegangan keluaran dari sensor bernilai +1 volt ketika tekanan 15 kPa dan tegangan bernilai +5 volt jika tekanan 115 kPa, dengan sensitivitas sebesar 46mV/KPa, dan bekerja optimal pada suhu -40°C – 125°C .

Dengan demikian, terdapat hubungan linier antara tekanan dan tegangan keluaran sensor.



Gambar 7 Kurva linearitas tekanan dan tegangan pada MPX4115AP

4.3 Analog To Digital Converter (ADC)

ADC adalah proses pengubahan sinyal analog menjadi sinyal digital. sinyal dari sensor yang sudah berupa data digital sehingga dapat diolah oleh mikrokontroler. Proses pengubahan terjadi pada konverter/pengubah yang dikenal dengan analog to digital converter. Proses pengubahan ini dikenal juga dengan nama sistem akuisisi data. Dengan menggunakan ADC, kita dapat mengamati sinyal-sinyal dari perubahan-perubahan sinyal analog seperti perubahan temperature, kepekatan asap, tekanan udara, kecepatan angin, berat benda, kadar asam (pH), dan lain-lain yang semuanya dapat diamati melalui sensornya masing-masing.

Ada beberapa tipe ADC antara lain ADC eksternal, misalnya dengan menggunakan IC ADC 0804 dan ADC internal, seperti pada ADC yang terintegrasi dengan mikrokontroler AVR Atmega 8535. Fitur ADC internal inilah yang menjadi salah satu kelebihan mikrokontroler AVR Atmega 8535 bila dibandingkan dengan beberapa jenis mikrokontroler yang lain. Data konversi ADC dirumuskan sebagai berikut :

$$ADC = \frac{V_{in} \times n}{V_{ref}}$$

Dimana, n = banyaknya bit ADC yang digunakan

V_{in} = tegangan masukan

V_{ref} = tegangan referensi yang dipilih

Dan hal yang paling penting dalam suatu rangkaian ADC adalah resolusi, yaitu besaran analog terkecil yang masih dapat dikonversi menjadi satuan digital.

$$\text{Resolusi (r)} = \frac{1}{2^{n-1}} \times V_{ref} \quad 13$$

Dimana, n = banyaknya bit ADC;

V_{ref} = tegangan referensi yang digunakan

4.4 Rangkaian Low-pass Filter

Low – pass filter pada perangkat pengukuran ini digunakan untuk menjaga agar sinyal yang masuk ke ADC pada mikrokontroler hanya sinyal DC saja, tanpa ada *noise* yang masuk ke ADC, serta membatasi frekuensi yang diterima [7]. Low pass filter' pada rangkaian *discrete input* (DI) diperlukan karena seringkali terjadi 'ringing' akibat *unmatched impedance* dari *field wiring* sampai kedua ujung DI dan *field switch terminals*, sehingga timbul *signal reflection*. Dalam hal ini, *low pass filter* berfungsi untuk membuang 'ringing' dari *discrete signal* sebenarnya.

Pada umumnya DI circuit sudah dilengkapi dengan *low pass filter* dan umumnya dengan *fixed low pass frequency* (ada yang membuat dengan *adjustable frequency*), akan tetapi karena kebutuhan beberapa aplikasi proses yang membutuhkan *response time* cepat, *low pass frequency* dibuat cukup tinggi agar *fast respond time* bisa tercapai. Akibatnya, apabila *ringing frequency* berada dibawah *low pass frequency*.

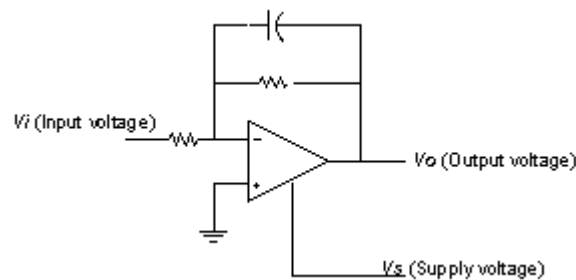
Apabila setiap perubahan status dari DI diterima sebagai beberapa kali open/close (atau terjadi ringing), bisa diatasi dengan beberapa cara:

1. Menambah low pass filter dengan rangkaian RC (Resistor dan Kapasitor) sebagai *signal conditioner*, atau

2. Mengkonfigurasi 'ON/OFF Delay Timer' (untuk aplikasi *normally OPEN/CLOSE respectively*)

Untuk kedua cara diatas, *respond time requirements* dari proses perlu diperhatikan. Apabila dengan menambahkan kedua hal diatas menyebabkan *respond time requirement* tidak terpenuhi, terpaksa harus menggunakan *matched impedance wiring* agar tidak terjadi *reflection*.

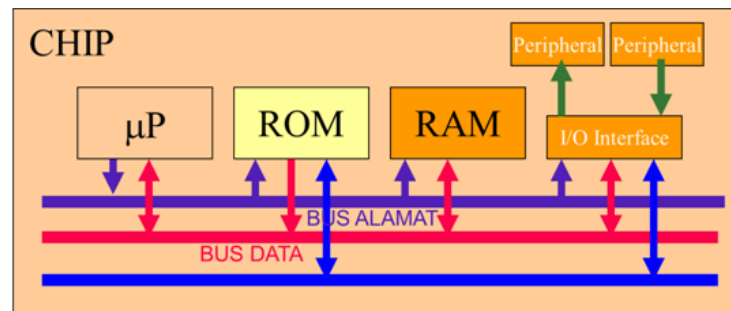
Rangkaian *Low – pass filter* yang digunakan pada perangkat ini adalah *passive Low – pass filter*, dengan menggunakan resistor dan kapasitor. Filter ini berfungsi untuk menghilangkan tegangan/amplitudo yang melebihi dari batas toleransi yang dibuat, sehingga sinyal-sinyal lain yang tidak diharapkan seperti noise dapat dihilangkan. Berikut skema *passive Low – pass filter* yang digunakan :



4.5 Perangkat Keras Sistem Akuisisi Data

4.5.1 Mikrokontroler

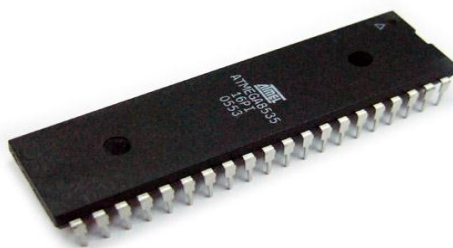
Mikrokontroler adalah *One –Single Chip* komputermengkombinasikan CPU dengan memori dan sistem I/O. Dengan demikian suatu mikrokontroler tidak membutuhkan tambahan *RAM*, *ROM* dan *I/O* yang memiliki kemampuan untuk diprogram dan digunakan untuk tugas-tugas yang berorientasi atau berbasis kontrol [5].



Gambar 8 Skemasistemstandar Mikrokontroler

Pada perangkat pengukuran ini digunakan mikrokontroler jenis AVR ATmega 8535 yang bersifat *low cost* dan *high performance*, dengan fitur yang cukup lengkap, mudah didapat, dan harga yang relatif terjangkau. Berikut adalah spesifikasi ATmega 8535 :

- 8K bytes *In-System Programmable Flash* dengan kemampuan *Read-While-Write*.
- 512 bytes EEPROM (*Electrically Erasable Programmable Read Only Memory*).
- 1 Kilo bytes SRAM (*Static Random Acces Memory*).
- 23 saluran I/O.
- 32 saluran register.
- Programmable serial USART (*Universal Synchronous and Asynchronous Receiver and Transmitter*).
- 10-bit ADC (*Analog to Digital Converter*) sebanyak 8 saluran.

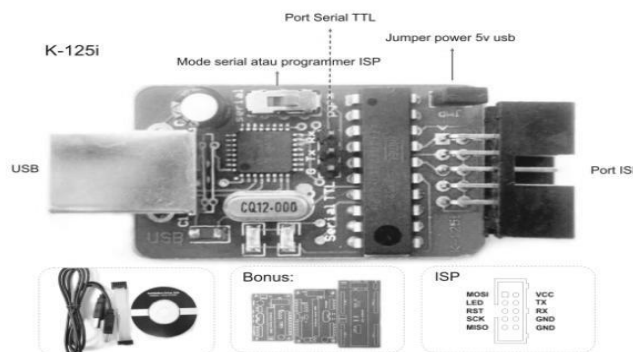


Gambar 9 Mikrokontroler ATmega 8535

4.5.2 USB ISP (In System Programming) dan Serial TTL (Transistor-Transistor Logic).

Untuk mendownload program ke dalam mikrokontroler dibutuhkan suatu perangkat yang dinamakan downloader. K-125i adalah salah satu jenis downloader yang sangat simple dan sudah disertai dengan koneksi USB (Universal Serial Bus) yang mempermudah pengguna dalam mendownload program.

K-125i juga disertai dengan serial TTL dimana ini adalah suatu bentuk jaringan yang dapat menghubungkan logika level tegangan yang dihasilkan dari mikrokontroler ke PC secara langsung.



Gambar 10 K-125i

4.6 Perangkat Lunak Sistem Akuisisi Data

4.6.1 Codevision AVR

Codevision AVR merupakan salah satu perangkat lunak kompilasi C yang khusus digunakan untuk mikrokontroler keluaran AVR Atmel. Kompilasi C melaksanakan semua elemen dari bahasa C ANSI, sebagaimana yang diperbolehkan oleh arsitektur AVR, dengan beberapa fitur yang ditambahkan untuk memaksimalkan dari spesifikasi arsitektur AVR dan kebutuhan. Untuk debugging embedded system, yang menggunakan komunikasi serial, IDE (Integrated Development Environment) memiliki built-in terminal.

4.6.2 Visual C# 2008

C# (dibaca: **C sharp**) merupakan sebuah bahasa pemrograman yang berorientasi objek yang dikembangkan oleh Microsoft sebagai bagian dari inisiatif

kerangka .NET Framework. Bahasa pemrograman ini dibuat berbasiskan bahasa C++ yang telah dipengaruhi oleh aspek-aspek ataupun fitur bahasa yang terdapat pada bahasa-bahasa pemrograman lainnya seperti Java, Delphi, Visual Basic, dan lain-lain) dengan beberapa penyederhanaan. Menurut standar *ECMA-334 C# Language Specification*, nama C# terdiri atas sebuah huruf Latin C (U+0043) yang diikuti oleh tanda pagar yang menandakan angka # (U+0023). Tanda pagar # yang digunakan memang bukan tanda kres dalam seni musik (U+266F), dan tanda pagar # (U+0023) tersebut digunakan karena karakter kres dalam seni musik tidak terdapat di dalam *keyboard* standar.

Standar European Computer Manufacturer Association (ECMA) mendaftarkan beberapa tujuan desain dari bahasa pemrograman C#, sebagai berikut:

- Bahasa pemrograman C# dibuat sebagai bahasa pemrograman yang bersifat *bahasa pemrograman general-purpose* (untuk tujuan jamak), berorientasi objek, modern, dan sederhana.
- Bahasa pemrograman C# ditujukan untuk digunakan dalam mengembangkan komponen perangkat lunak yang mampu mengambil keuntungan dari lingkungan terdistribusi.
- Portabilitas *programmer* sangatlah penting, khususnya bagi *programmer* yang telah lama menggunakan bahasa pemrograman C dan C++.
- Dukungan untuk internasionalisasi (multi-language) juga sangat penting.
- C# ditujukan agar cocok digunakan untuk menulis program aplikasi baik dalam sistem klien-server (*hosted system*) maupun sistem embedded (*embedded system*), mulai dari perangkat lunak yang sangat besar yang menggunakan sistem operasi yang canggih hingga kepada perangkat lunak yang sangat kecil yang memiliki fungsi-fungsi terdedikasi.
- Meskipun aplikasi C# ditujukan agar bersifat 'ekonomis' dalam hal kebutuhan pemrosesan dan memori komputer, bahasa C# tidak ditujukan untuk bersaing secara langsung dengan kinerja dan ukuran perangkat lunak yang dibuat dengan menggunakan bahasa pemrograman C dan bahasa rakitan.

- Bahasa C# harus mencakup pengecekan jenis (*type checking*) yang kuat, pengecekan larik (*array*), pendeteksian terhadap percobaan terhadap penggunaan Variabel-variabel yang belum diinisialisasikan, portabilitas kode sumber, dan pengumpulan sampah (*garbage collection*) secara otomatis.

Bahasa C# dalam penelitian ini digunakan untuk membuat interface perangkat pengukuran dengan *ground control system* (GCS), serta sebagai bahasa pemrograman untuk *kalman filter* pada pengukuran ini.

E. Kalman Filter

Kalman filter merupakan suatu algoritma yang menggabungkan model dan pengukuran. Data pengukuran terbaru menjadi bagian penting dari algoritma Kalman filter karena data terakhir akan mengoreksi hasil prediksi pengukuran, sehingga hasil estimasi selalu mendekati kondisi yang sebenarnya.

Kalman filter mengatasi masalah dengan mencoba mengestimasi keadaan x dari sebuah proses yang terkontrol dengan persamaan dasar yang digunakan dalam pengestimasi, persamaan linier sebagai berikut: [8]

$$x_{(k+1)} = Ax_k + Bu_k + w_k, \quad 14$$

dengan besar nilai pengukuran

$$z_k = Hx_k + v_k. \quad 15$$

Variabel acak w_k dan v_k menunjukkan *noise* proses dan *noise* pengukuran dengan diasumsikan masing-masing dengan nilai distribusi

$$p(w) \sim N(0, Q), \quad 16$$

$$p(v) \sim N(0, R). \quad 17$$

Dalam keadaan tertentu, matriks kovarian *noise* proses Q dan kovarian *noise* pengukuran R mungkin berubah setiap kali pengukuran, namun kali ini kita asumsikan konstan karena kita mengukur dengan ketinggian tertentu saja.

Dengan x_k adalah keadaan pada saat t_k , sedangkan x_{k-1} adalah keadaan pada saat t_{k-1} . A merupakan matrik yang menghubungkan antara keadaan pada waktu t_{k-1} dengan t_k . Matrik B menghubungkan antara input kendali u dengan keadaan x . Matrik H menghubungkan antara keadaan x dengan besar nilai pengukuran z_k . w adalah *noise* proses dan v adalah *noise* pengukuran.

Dengan menetapkan \hat{x}_k^- sebagai estimasi keadaan *apriori* pada saat k diberikan informasi dari proses sebelum saat k , sedangkan \hat{x}_k sebagai estimasi keadaan *aposteriori* pada saat k diberikan pengukuran z_k . Maka kita dapat menetapkan error estimasi *apriori* dan *aposteriori* sebagai

$$e_k^- \equiv x_k - \hat{x}_k^-, \text{ dan } e_k \equiv x_k - \hat{x}_k \quad 18$$

kovarian error estimasi *priori*

$$P_k^- = E[e_k^- e_k^{-T}], \quad 19$$

Dan kovarian error estimasi *posteriori*

$$P_k = E[e_k e_k^T]. \quad 20$$

Untuk menurunkan persamaan Kalman filter, kita mulai dengan mencari persamaan untuk menghitung estimasi keadaan *aposteriori* \hat{x}_k sebagai kombinasi linier dari estimasi keadaan *apriori* \hat{x}_k^- dan besar perbedaan antara pengukuran sebenarnya z_k dan prediksi pengukuran $H\hat{x}_k^-$, seperti yang ditunjukkan pada persamaan dibawah ini.

$$\hat{x}_k = \hat{x}_k^- + K(z_k - H\hat{x}_k^-). \quad 21$$

Perbedaan $(z_k - H\hat{x}_k^-)$ merupakan inovasi pengukuran, atau residual. Residual mencerminkan perbedaan antara prediksi pengukuran $H\hat{x}_k^-$ dan pengukuran sebenarnya z_k . Jika residual bernilai 0 maka tidak ada perbedaan antara prediksi pengukuran dengan pengukuran sebenarnya.

K merupakan *gain* atau faktor penggabungan (*blending factor*) yang meminimalkan kovarian *error a posteriori*. Minimalisasi ini dapat diperoleh dengan mensubstitusikan (8) ke dalam persamaan diatas untuk e_k , ke dalam persamaan (7), dengan menurunkan hasil yang berhubungan dengan K , menetapkan bahwa hasil sama dengan nol, lalu cari nilai K . Salah satu hasil K ditunjukkan oleh

$$K_k = P_k^- H^T (H P_k^- H^T + R)^{-1} = \frac{P_k^- H^T}{H P_k^- H^T + R} \quad 22$$

Pada (9) terlihat bahwa pengukuran kovarian *error* R mendekati nol, *gain* K memiliki residu yang lebih besar, seperti

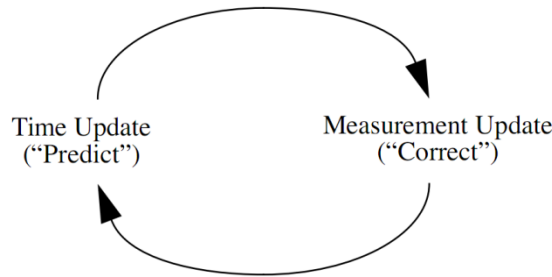
$$\lim_{R_k \rightarrow 0} K_k = H^{-1}$$

Di sisi lain, sebagai kovarian *error* estimasi *apriori* P_k^- mendekati nol, *gain* K memiliki residu yang tidak terlalu besar,

$$\lim_{P_k^- \rightarrow 0} K_k = 0$$

Kalman filter mengestimasi proses dengan menggunakan kontrol umpan balik : filter mengestimasi proses pada suatu waktu dan kemudian mendapat umpan balik dalam bentuk pengukuran *noise* (gangguan). Dengan demikian, persamaan untuk Kalman filter menjadi dua kelompok : persamaan *update* waktu dan persamaan *update* pengukuran. Persamaan *update* waktu bertanggung jawab untuk memproyeksikan kedepan (waktu) keadaan saat ini dan estimasi kovarian *error* mengestimasi untuk mendapatkan sebuah estimasi *apriori* untuk langkah waktu berikutnya. Persamaan *update* pengukuran bertanggung jawab atas umpan balik untuk menggabungkan pengukuran baru ke dalam estimasi *apriori* untuk mendapatkan estimasi *a posteriori* yang lebih baik.

Persamaan *update* waktu juga dapat dianggap sebagai persamaan prediktor, sedangkan persamaan *update* pengukuran dapat dianggap persamaan korektor. Algoritma estimasi menyerupai algoritma prediktor-korektor untuk memecahkan masalah numerik.



Gambar 11 Siklus proses operasi kalman filter

Persamaan Kalman filter *update* waktu

$$\hat{x}_k^- = A\hat{x}_{k-1} + Bu_{k-1} \quad 23$$

$$P_k^- = AP_{k-1}A^T + Q \quad 24$$

Persamaan Kalman filter *update* pengukuran

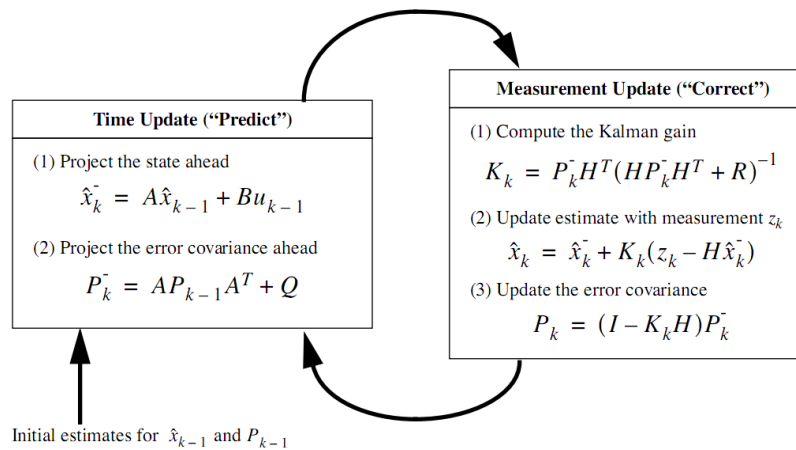
$$K_k = P_k^- H^T (HP_k^- H^T + R)^{-1} \quad 25$$

$$\hat{x}_k = \hat{x}_k^- + K_k(z_k - H\hat{x}_k^-) \quad 26$$

$$P_k = (1 - K_k H)P_k^- \quad 27$$

Tugas pertama selama *update* pengukuran adalah untuk menghitung *gain* Kalman, K_k . Langkah selanjutnya adalah mengukur proses untuk mendapatkan z_k , kemudian untuk menghasilkan estimasi keadaan *a posteriori* dengan memasukkan pengukuran ke (13). Langkah terakhir adalah untuk mendapatkan estimasi *a posteriori* kovarian error melalui (14).

Setelah waktu dan pengukuran terbaru, proses akan diulang dengan estimasi *a posteriori* sebelumnya digunakan untuk memprediksi estimasi *a priori* baru. Sifat rekursif merupakan salah satu fitur yang sangat menarik dari Kalman filter, ini membuat implementasi lebih layak daripada (misalnya) implementasi dari Wiener Filter yang dirancang untuk beroperasi pada semua data langsung untuk setiap estimasi. Kalman filter bersifat rekursif pada semua pengukuran yang lalu. Gambar 12 dibawah ini adalah gambaran lengkap dari operasi filter,



Gambar 12 Gambaran lengkap operasi kalman filter

Dalam implementasi sebenarnya dari filter ini, kovarian *noise* pengukuran R biasanya diukur sebelum adanya operasi dari filter ini. Mengukur kovarian error pengukuran R umumnya secara praktik karena kita harus mengukur proses tetap (saat pengoperasian filter) jadi kita seharusnya mengambil beberapa sampel pengukuran untuk menentukan varian dari *noise* pengukuran.

Penentuan dari kovarian *noise* proses Q umumnya lebih sulit karena kita tidak biasa memiliki kemampuan untuk mengamati secara langsung proses pengestimasi. Kadang-kadang relatif mudah suatu model proses dapat menghasilkan hasil yang baik jika dalam satu "suntikan" cukup dalam proses pemilihan nilai Q . Tentu saja dalam hal ini diharapkan bahwa proses pengukuran dapat diandalkan.

Dalam kedua kasus, apakah kita memiliki dasar untuk memilih parameter, sering kali kinerja filter yang baik dapat diperoleh dengan menyetel parameter filter Q dan R . Penyetelan biasanya dilakukan secara *off-line* (proses filterisasi belum dimulai), seringkali dengan bantuan Kalman filter yang lain dalam proses yang umumnya disebut dengan identifikasi sistem.

Dalam kondisi dimana Q dan R konstan, baik estimasi kovarian error P_k dan *gain* Kalman K_k akan stabil dengan cepat dan kemudian tetap konstan. Jika

hal ini terjadi, parameter ini dapat dihitung dengan menjalankan filter, atau misalnya dengan menentukan nilai *steady state* dari P_k .

Hal ini sering terjadi bagaimanapun kesalahan pengukuran tidak akan konstan. Selain itu, *noise* proses Q kadang-kadang berubah secara dinamik ketika filter beroperasi menjadi Q_k untuk menyesuaikan dengan dinamika yang berbeda. Dalam kasus Q_k mungkin dapat dipilih untuk nilai ketidakpastian.

BAB III

METODOLOGI PENELITIAN

A. Tujuan Operasional

1. Dapat merancang bangun antarmuka sensor Tekanan Barometrik dengan mikrokontroler ATmega sebagai perangkat pengukuran ketinggian.
2. Dapat mengintegrasikan *low-pass filter* dan juga *Kalman filter* pada perangkat pengukuran ketinggian.
3. Membandingkan hasil pengukuran dari perangkat yang telah diberi *low-pass filter* dan *kalman filter* dengan data pengukuran dari perangkat yang belum diintegrasikan dengan *low-pass filter* dan *kalman filter*, sebagai acuan akurasi perangkat digunakan data dari GPS yang sudah sesuai dengan *International Standard Atmosphere (ISA)*.

B. Waktu dan Tempat Penelitian

Penelitian ini dilaksanakan di kantor dan laboratorium PTIPK – BPPT Jakarta, lapangan udara Kopasus Batujajar Bandung, Lab instrument dan komputasi Fisika UNJ. Penelitian ini berlangsung selama oktober 2010 sampai dengan febuari 2010.

C. Metode Penelitian

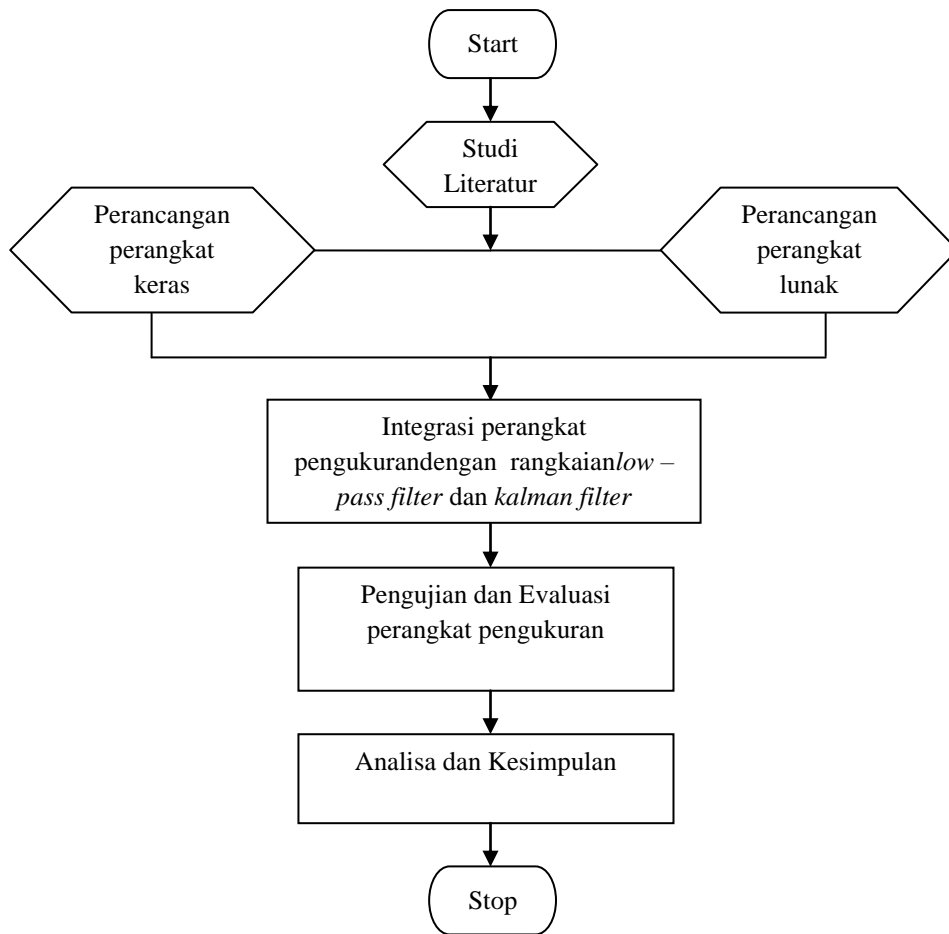
Metode penelitian yang digunakan dalam penelitian ini adalah metode studi literatur dan eksperimen. Secara umum pada tahap studi literature adalah Studi pendahuluan berupa studi pustaka dengan mempelajari tentang pengukuran ketinggian pada PUNA.

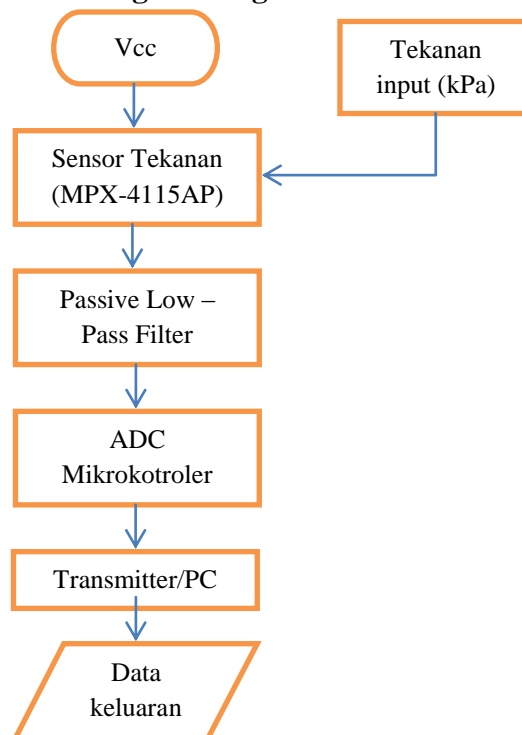
Secara umum pada tahap eksperimen terdiri dari: (1) merancang bangun perangkat pengukuran ketinggian dengan cara antarmuka sensor Tekanan Barometrik dengan mikrokontroler ATmega, (2) Pengintegrasian rangkaian *Low-*

Pass Filter pada perangkat, (3) pengkalibrasian perangkat pengukuran ketinggian, (4) Pengintegrasian *kalman filter* (5) Pengujian dan evaluasi kinerja perangkat pengukuran yang sudah terintegrasi *low-pass Filter* dan *kalman filter* dengan membandingkan terhadap perangkat yang belum terintegrasi *low-pass Filter* dan *kalman filter*, sebagai acuan akurasi perangkat adalah menggunakan data dari GPS yang sudah sesuai dengan *International Standard Atmosphere (ISA)*.

D. Instrumen Penelitian

1. Mikrokontroler
Sistem minimum mikrokontroler ATmega 8535.
2. Sensor Tekanan Barometrik
sensor yang digunakan dalam penelitian ini adalah MPX 4115 AP
3. Laptop / PC
4. RS 232
5. Serial to USB converter
6. ISP k-125
7. Resistor
8. Kapasitor
9. PCB
10. Multimeter
11. DC adaptor 6V
12. Solder
13. Timah

E. Diagram Alir Penelitian

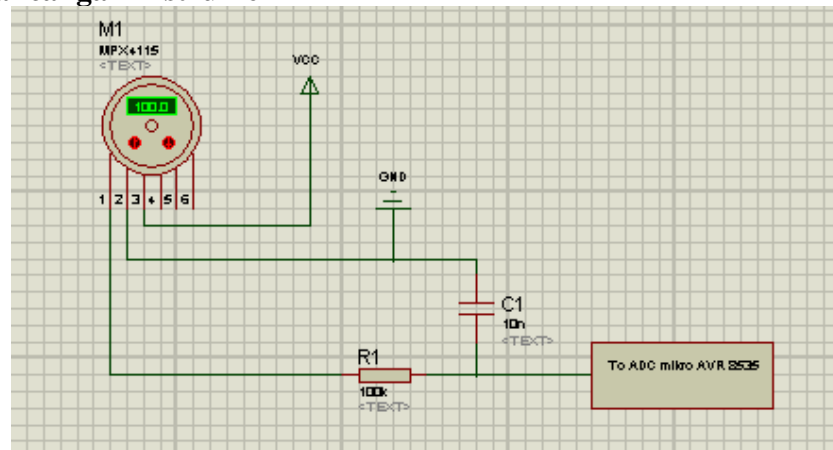
F. Skema Rangkaian Perangkat Pengukuran.

BAB IV

HASIL PENELITIAN DAN PEMBAHASAN

A. Pengembangan Instrumen

1.1 Perancangan Instrumen



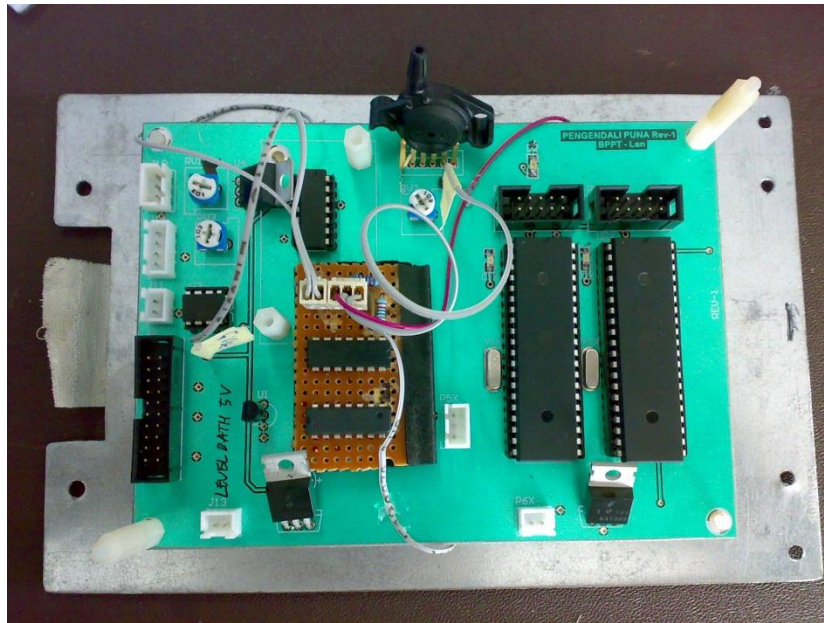
Gambar 13 Gambar Rangkaian Perangkat Pengukuran Ketinggian

Gambar di atas merupakan rangkaian pengukur ketinggian menggunakan sensor Tekanan Barometrik MPX-4115AP setelah diberi Low-Pass filter. Sinyal keluaran dari sensor kemudian dihubungkan ke dalam rangkaian pengkondisian sinyal sebelum masuk ke dalam sistem minimum mikrokontroler. Pada rangkaian pengkondisian sinyal tersebut adalah rangkaian low-pass filter. Rangkaian *filter lolos rendah* dengan frekuensi *cut off* 1500 Hz terdiri atas resistor dengan nilai $R = 100\text{ K}\Omega$ dan kapasitor dengan nilai $C = 10\text{ nF}$. Setelah dilakukan perangkaian perangkat selesai, lalu dilakukan pengkalibrasian perangkat.

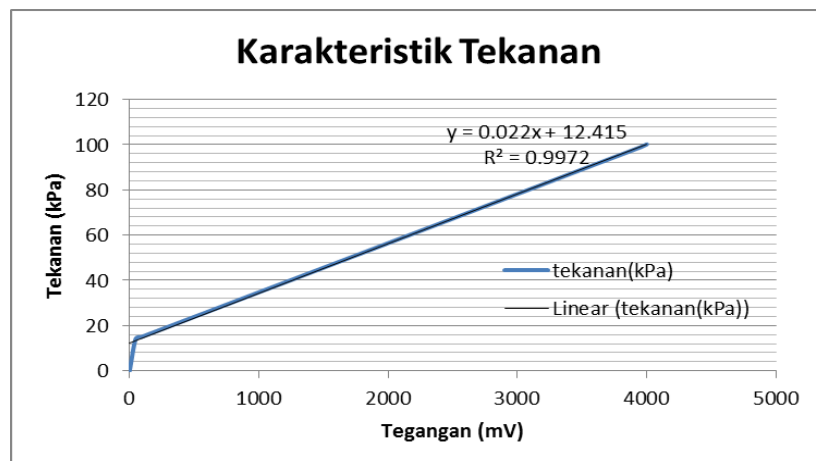
1.2 Kalibrasi Instrumen

Kalibrasi adalah kegiatan menghubungkan nilai yang ditunjukkan oleh instrument ukur dengan nilai-nilai yang sudah diketahui nilai kebenarannya (yang berkaitan dengan besaran yang diukur). Nilai yang sudah diketahui ini biasanya merujuk ke suatu nilai dari kalibrator atau standar, yang tentunya harus memiliki akurasi yang lebih tinggi dari pada alat ukur yang akan diuji. Proses kalibrasi pada sensor tekanan MPX-4115 AP dilakukan untuk mengetahui perubahan tegangan

output pada sensor terhadap perubahan tekanan udara sekitar pesawat. Data pembandingan yang digunakan berasal dari *pressure gaugemeter*(Altimeter manual) yang biasa untuk mengukur tekanan lingkungan.



Gambar 14.1 Perangkat pengukuranketinggian



Gambar 14.2 Karakteristik sensor MPX4115AP

Gambar 14.1 adalah gambar perangkat pengukuran setelah diintegrasikan dengan low-pass filter. Gambar 14.2 menunjukkan perubahan tegangan *output* sensor terhadap tekanan yang bersifat linier. Berdasarkan grafik, diperoleh

persamaan regresi hubungan antara nilai tegangan keluaran sensor dengan nilai tekanan altimeter manual:

$$P = 0.022 \times V + 12.415$$

$$R^2 = 0.9972$$

Dimana :

P = tekanan input (kPa)

V = tegangan keluaran sensor (mV)

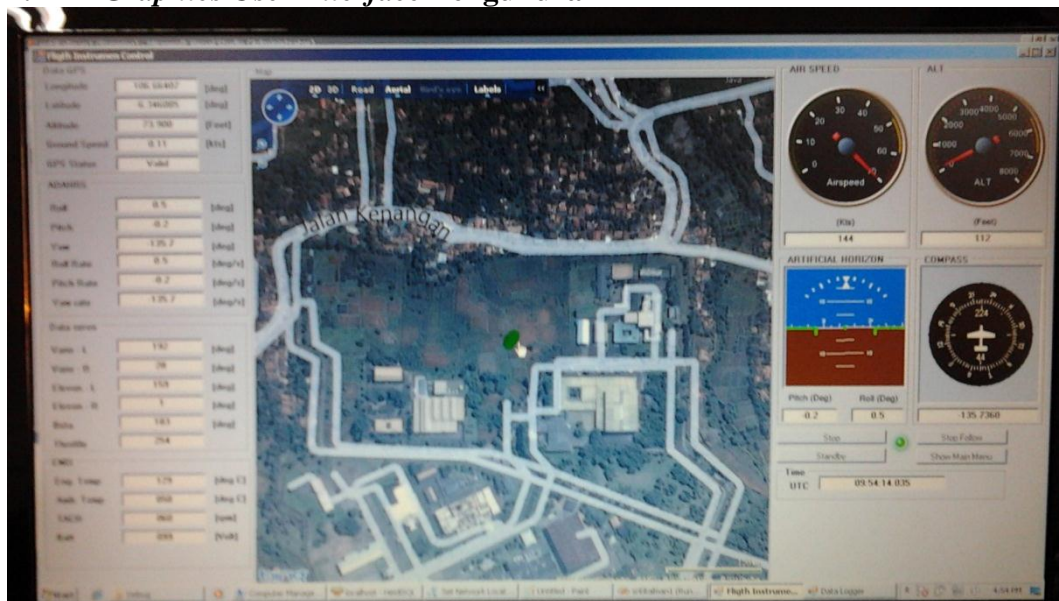
R = Koefisien korelasi

Kelinieran suatu sensor dapat diketahui juga dari nilai koefisien korelasinya. Koefisien korelasi adalah seberapa besar keterkaitan antara *variable* bebas dan *variable* tak bebas. Pada grafik di atas sensor tekanan MPX-4115AP mempunyai koefisien korelasi sebesar 0.9972 antara tegangan keluarannya dengan tekanan dari pembacaan *Altimeter* manual, hal ini menunjukkan bahwa sensor yang digunakan mempunyai tingkat korelasi yang baik dengan alat ukur standar. Jadi, MPX4115AP mempunyai kelinieran yang baik untuk pengukuran tekanan barometrik antara 15 KPa – 115 KPa.

1.3 Pengintegrasian Kalman Filter

Pengintegrasian Kalman filter pada penelitian ini dilakukan pada perangkat lunak, dalam hal ini dilakukan pada Visual C#. yang pertama dilakukan adalah membuat class iterasi untuk proses Kalman Filter. Kedua setelah dibuat *class* kalman filer, class itu digunakan untuk pemfilteran pengukuran ketinggian pesawat udara nirawak. Setelah didapatkan hasil perhitungan kalman ditampilkan pada *graphics user interface* untuk memonitoring pergerakan pesawat dari waktu ke waktu.

1.4 *Graphics User Interface Pengukuran*

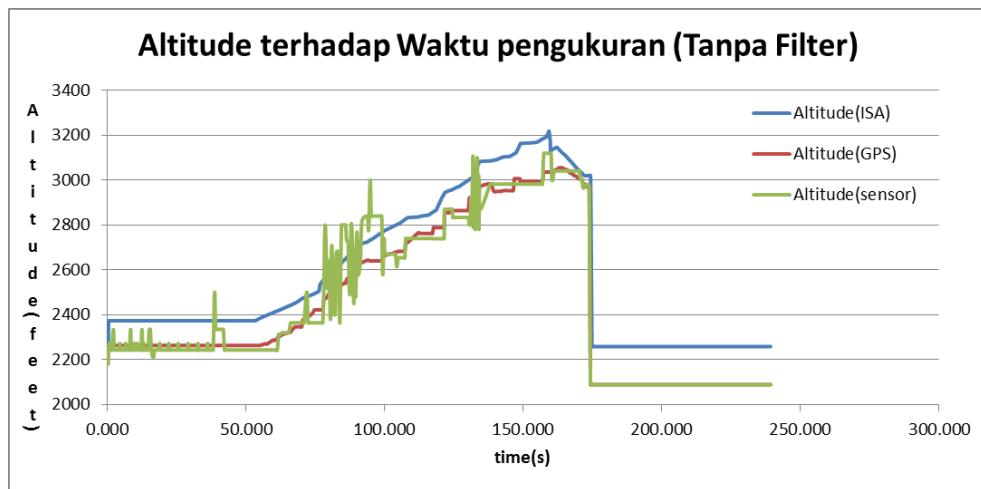


Gambar 15. *Graphics User Interface* monitoring Pesawat

Gambar diatas adalah gambar sebagian dari *Graphics user Interface* untuk memonitor pergerakan pesawat. Disana terdapat gauge virtual untuk memonitor pergerakan ketinggian pesawat (bagian ALT).

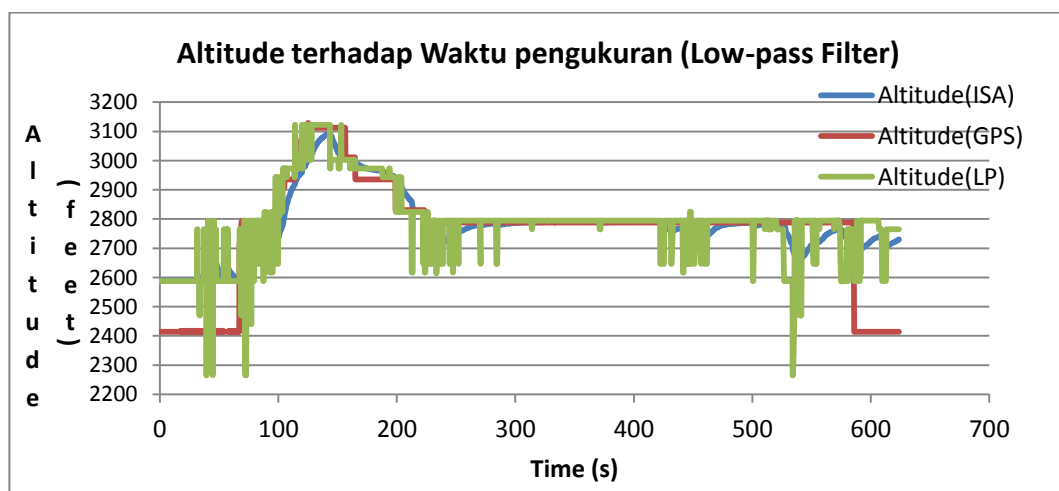
B. Hasil Pengujian dan Pembahasan

Pengukuran ketinggian sebelum diberi low-pass filter maupun kalman filter dilakukan di lapangan udara kopasus batujajar Bandung 2 Desember 2010. Pada pengujian ini didapat data selama 3 menit 59 detik sebanyak 585 data seperti pada Gambar 16.1 dibawah. Garis biru adalah ketinggian berdasarkan tabel ISA, garis yang berwarna merah menunjukkan pengukuran ketinggian dari perangkat GPS, dan garis yang berwarna hijau menunjukkan data pengukuran ketinggian dari perangkat sensor barometrik. Hasil pengukuran perangkat sensor barometrik sebelum diberi low-pass filter dibandingkan dengan tabel ISA mendapat nilai RMSE sebesar 79.19.



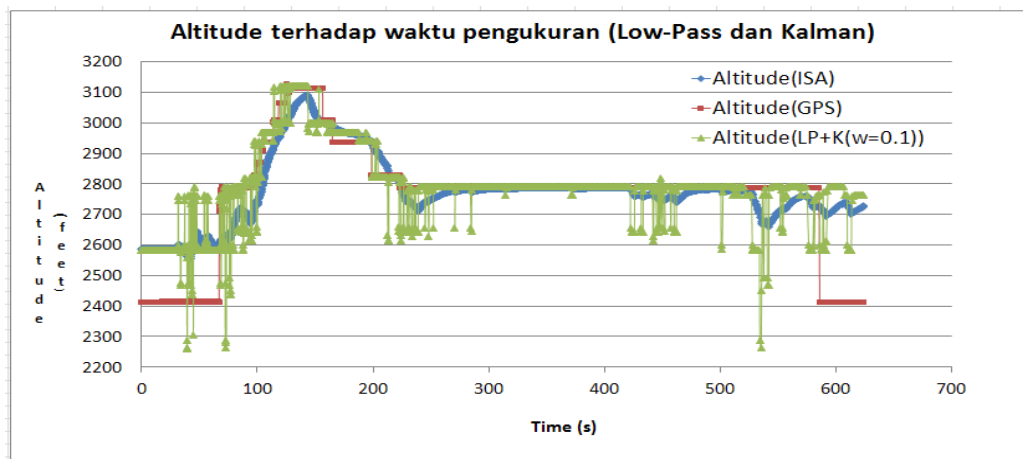
Gambar 16.1 Grafik Ketinggian sebelum diberi filter

Pengukuran ketinggian setelah diberi low-pass filter dilakukan di lapangan udara kopasus batujajar Bandung 2 Febuari 2011. Pada pengujian ini didapat data selama 10 menit 23 detik sebanyak 2085 dataseperti pada Gambar 16.2 dibawah. Garis biru adalah ketinggian berdasarkan tabel ISA, garis yang berwarna merah menunjukkan pengukuran ketinggian dari perangkat GPS, dan garis yang berwarna hijau menunjukkan data pengukuran ketinggian dari perangkat sensor barometrik setelah diberi low-pass filter. Sesuai dengan yang diharapkan Hasil pengukuran perangkat sensor barometrik yang sudah mendekati data ketinggian pada tabel ISA dengan RMSE sebesar 68.20.

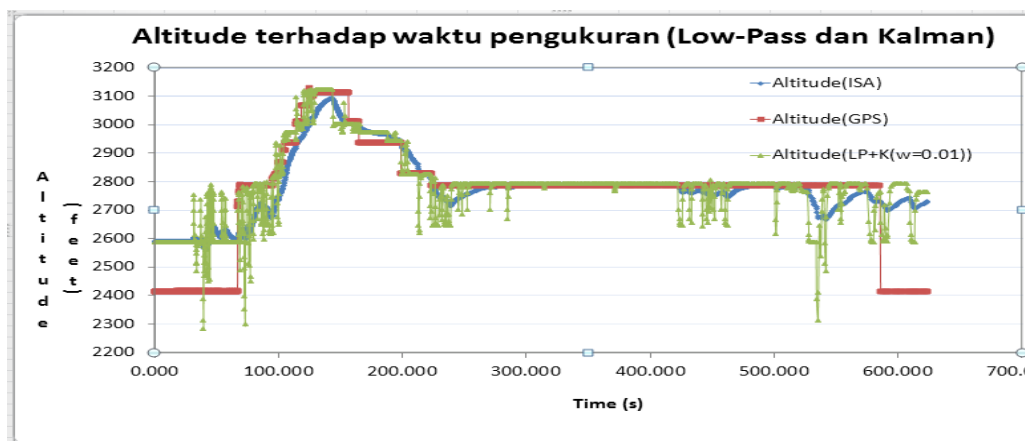


Gambar 16.2 Grafik Ketinggian (Low-pass) sebelum diberi kalman filter

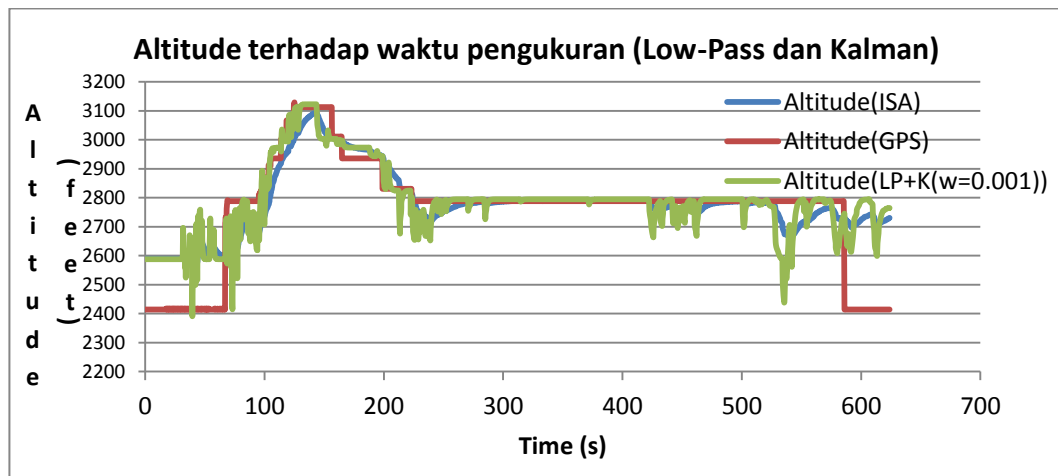
Setelah diberi kalman filter didapat grafik seperti pada gambar 17,18,19,20, dan 21 dibawah. Garis biru menunjukkan data ketinggian dari tabel ISA, garis yang berwarna merah menunjukkan pengukuran ketinggian dari perangkat GPS, garis berwarna hijau adalah pengukuran ketinggian setelah diberi low-pass filter juga kalman filter. Sesuai dengan yang diharapkan Hasil pengukuran perangkat sensor barometrik yang sudah diberi kalman dengan parameter kalman yang semakin kecil menunjukkan pengukuran semakin mendekati hasil yang sebenarnya seperti pada tabel ISA, sehingga berhimpit dengan garis ketinggian ISA, dengan RMSE = 64.11 pada nilai $w = 0.1$ (gambar 17), nilai RMSE = 56.47 pada nilai $w = 0.01$ (gambar 18), nilai RMSE = 42.87 pada nilai $w = 0.001$ (gambar 19), nilai RMSE = 26.05 pada nilai $w = 0.0001$ (gambar 20), nilai RMSE = 5.83 pada nilai $w = 0.00001$ (gambar 21).



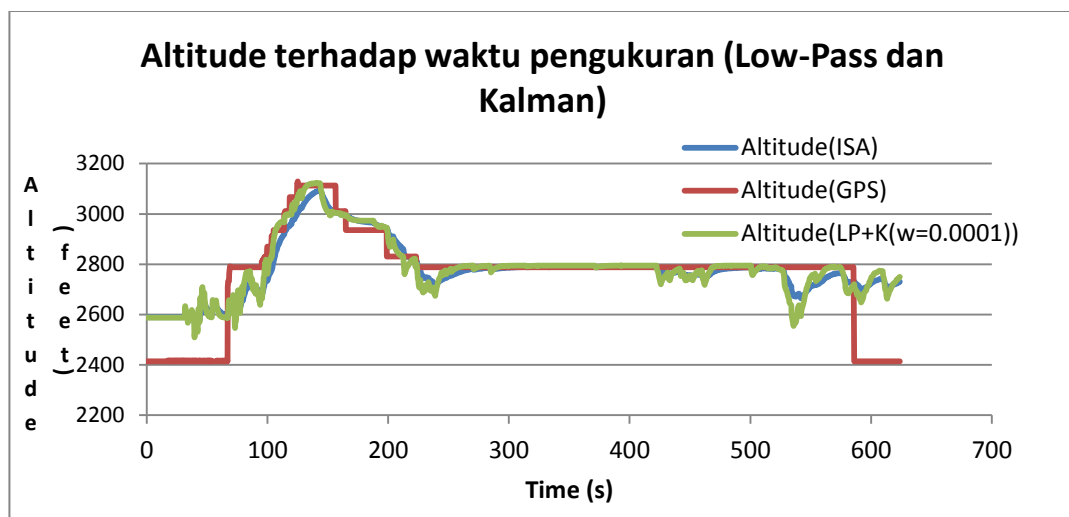
Gambar 17 Grafik Ketinggian setelah diberi kalman filter ($w=0.1$)



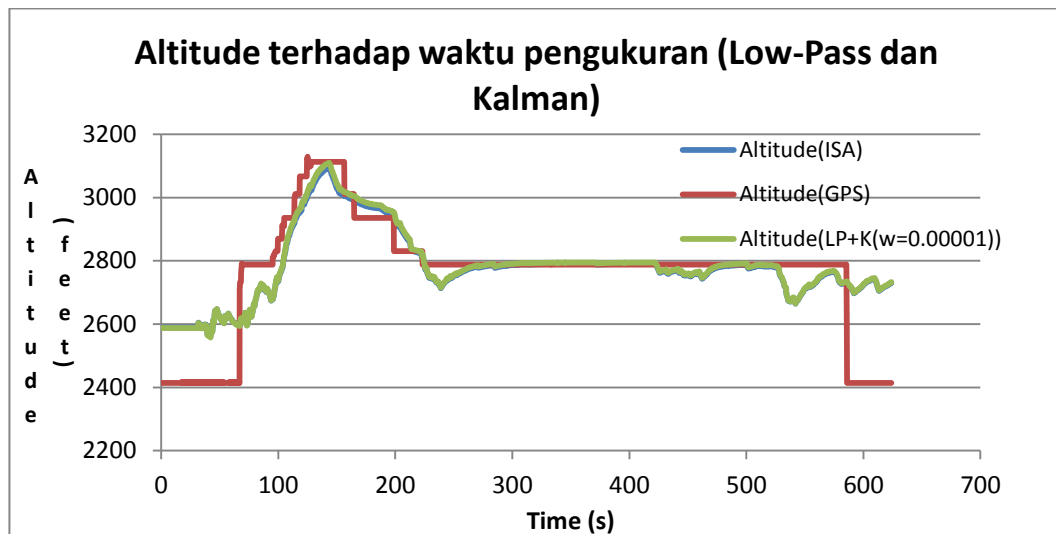
Gambar 18 Grafik Ketinggian setelah diberi kalman filter ($w=0.01$)



Gambar 19. Grafik Ketinggian setelah diberi kalman filter ($w=0.001$)



Gambar 20. Grafik Ketinggian setelah diberi kalman filter ($w=0.0001$)



Gambar 21. Grafik Ketinggian setelah diberi kalman filter ($w=0.00001$)

Nilai w adalah parameter tuning dari kalman filter, dengan Q (variabilitas Ketinggian sebenarnya) dan R (noise pengukuran). Q mewakili berapa banyak *variabilitas* data ketinggian PUNA yang sebenarnya, dan R mewakili seberapa banyak noise sensor bisa diharapkan untuk mengetahui ketinggian PUNA dari satu pengukuran ke pengukuran yang berikutnya dan t adalah waktu pengukuran.

$$\lim_{t \rightarrow \infty} \frac{Q}{R} = w$$

R dapat dihitung dengan membagi Q dengan w . Nilai w dapat digunakan sebagai parameter *tuning* untuk mengontrol seberapa cepat filter akan merespon perubahan pengukuran dalam menghitung keluaran Kalman filter. Semakin besar rasio, semakin lama waktu yang diperlukan untuk melakukan perubahan pada pengukuran ketinggian untuk dimasukkan ke dalam keluaran filter. Semakin kecil rasio, akan lebih rentan bagi filter untuk menerima perubahan besar dalam pengukuran ketinggian. Pemilihan nilai w telah dilakukan secara empiris. Penggunaan teknik varian rasio memungkinkan proses pemilihan Q dan R terbaik dengan perumpamaan nilai konstan untuk w .

Hasil reduksi *error* untuk hasil diatas adalah untuk $w = 0.1$ mereduksi *error* sebesar 16.96%, untuk $w = 0.01$ reduksi *error* sebesar = 26.85%, untuk $w =$

0.001 mereduksi *error* sebesar 44.47%, untuk $w = 0.0001$ mereduksi *error* sebesar 66.26%, untuk $w = 0.00001$ mereduksi *error* sebesar 92.45%. Hasil ini menunjukkan bahwa dengan semakin kecil nilai w maka semakin bagus dalam mereduksi *error* pengukuran. Low-pass filter ditambah Kalman filter dengan $W = 0.00001$ adalah yang paling optimal untuk mereduksi *error* pengukuran. Hal ini terjadi karena semakin besar nilai w yang diberikan maka respon filter terhadap perubahan pengukuran akan semakin cepat, sehingga akan semakin sedikit *error* yang direduksi, sebaliknya semakin kecil nilai w yang diberikan maka respon filter terhadap perubahan pengukuran akan semakin lambat, sehingga akan semakin banyak *error* yang direduksi.

C. Identifikasi Pengukuran

3.1 Kesalahan Rata-rata Pengukuran (*Root Mean Square Error*)

RMSE (*Root Mean Square Error*) merupakan indikator dari rata-rata kesalahan sebuah pengukuran dengan melihat perbedaan antara nilai estimasi dan nilai pengamatan sebenarnya. Meskipun pengukuran ketinggian ini telah diberi kalman filter, tidak menutup kemungkinan kesalahan tetap terjadi. Walaupun sudah memberikan parameter tuning w yang terbaik, kesalahan masih ada pada pengukuran namun dengan RMSE yang lebih kecil dibandingkan dengan pengukuran sebelum diberi kalman filter.

Semakin kecil nilai RMSE atau semakin mendekati nol, kesalahan pengukuran menjadi semakin kecil, maka pengukuran ketinggian dikatakan mendekati data ketinggian ISA atau sudah mendekati pengukuran ketinggian sebenarnya. Hal ini dapat dilihat dari hasil pengukuran yang sudah diberi *variabilitas* nilai dari parameter *tuning* kalman, yang menunjukkan semakin kecil nilai parameter tuning pada kalman maka hasilnya lebih optimal (mendekati nilai sebenarnya). Perhitungan besar RMSE adalah seperti pada persamaan dibawah :

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (x_{1,i} - x_{2,i})^2}{n}}$$

Dari perubahan nilai w yang digunakan, menghasilkan nilai RMSE yang berbeda, seiring dengan menurunnya nilai w maka nilai RMSE yang dihasilkan semakin mendekati nol, seperti terlihat pada tabel 2.

Tabel 1 RMSE

RMSE							
GPS	Tanpa Low-pass	Low-Pass	Low-Pass + Kalman				
			($w=0.1$)	($w=0.01$)	($w=0.001$)	($w=0.0001$)	($w=0.00001$)
118.51	77.19	68.20	64.11	56.47	42.87	26.05	5.83

3.2 Reduksi Kesalahan (*Error Reduction*)

Reduksi kesalahan dihitung dengan membandingkan rata hasil error yang terjadi pada setiap pengukuran. Low-pass filter mereduksi error terhadap pengukuran sebelumnya sebesar 11.66%. reduksi error semakin lebih baik ketika setelah diberi low-pass dan ditambahkan kalman filter. Terlihat dari hasil pada tabel 2 penambahan low-pass hanya dapat mereduksi *error* sebesar 11.66% dari pengukuran sebelum diberi low-pass. Tetapi pada saat ditambahkan low-pass dan kalman filter secara bersamaan dapat mereduksi *error* sampai dengan 92.45%.

Kalman filter mereduksi kesalahan pengukuran yang terjadi, dengan berbeda – beda parameter w yang diberikan memberikan hasil yang berbeda pula pada tingkat reduksi kesalahan yang dilakukan oleh Kalman filter. Seperti yang terlihat pada tabel 2 reduksi kesalahan mencapai 96.48% pada pengukuran dengan parameter w sebesar 0.00001. Hal ini terjadi karena semakin besar nilai w yang diberikan maka respon filter terhadap perubahan pengukuran akan semakin cepat, sehingga akan semakin sedikit error yang direduksi. Begitu juga sebaliknya semakin kecil nilai w yang diberikan akan semakin lambat eksekusi kalman filter terhadap perubahan pengukuran, sehingga semakin banyak *error* yang bisa direduksi.

Tabel2reduksi *error*

<i>Error Reduction (%)</i>					
Low-Pass	Low-Pass + Kalman				
	(w=0.1)	(w=0.01)	(w=0.001)	(w=0.0001)	(w=0.00001)
11.66	16.96	26.85	44.47	66.26	92.45

3.3 Tingkat Keakuratan Pengukuran

Tujuan awal dari penggunaan low-pass filter dan Kalman filter ini adalah untuk meningkatkan keakurasian atau ketepatan pengukuran, walaupun terjadi banyak gangguan pada pengukuran. Pada pengukuran ini didapatkan pengukuran yang paling optimal setelah diberi low-pass filter dan ditambahkan pula kalman filter (w=0.00001) dengan RMSE paling kecil/optimal sebesar 5.83, dengan reduksi error paling optimal dari sebelumnya yaitu sebesar 92.45% error yang direduksi, dengan keakurasian pengukuran sebesar 99.86%. Seperti terlihat pada tabel 3 untuk tingkat keakurasian pengukuran.

Tabel3keakurasian pengukuran

Akurasi (%)							
GPS	Tanpa Low-pass	Low-Pass	Low-Pass + Kalman				
			(w=0.1)	(w=0.01)	(w=0.001)	(w=0.0001)	(w=0.00001)
95.32	95.28	99.78	99.79	99.79	99.79	99.81	99.86

Nilai keakuratan didapat dengan merata-ratakan hasil pengukuran dari 585 data yang terukur dari perangkat sebelum diberi low-pass, dan 2085 data dari perangkat yang sudah diberi low-pass maupun kalman filter. Dengan membandingkan hasil pengukuran baik setelah diberi filter maupun sebelum diberi filter terhadap data ketinggian ISA, maka akan terlihat pengukuran mana yang memiliki keakuratan pengukuran paling mendekati ketinggian sebenarnya.

Seperti pada tabel, terlihat pengukuran parameter w dengan nilai 0.00001 memiliki keakuratan tertinggi yaitu 99.86%. Sedangkan pengukuran yang sebelum diberi low-pass maupun kalman mendapatkan nilai akurasi terkecil dengan akurasi sebesar 95.283%.

BAB V

KESIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan hasil penelitian dan analisa data yang telah diuraikan pada bab sebelumnya, maka dapat ditarik beberapa kesimpulan sebagai berikut :

1. Low-pass filter sedikit berpengaruh kepada perangkat pengukuran, karena dapat mereduksi *error* pengukuran sebesar 11.66%.
2. Pengintegrasian low-pass filter dan kalman filter secara bersamaan pada perangkat pengukuran dapat mengurangi error pengukuran lebih baik dibanding hanya menggunakan low-pass filter saja, maupun tanpa low-pass filter.
3. Pengukuran ketinggian paling mendekati tabel ISA pada penelitian ini adalah setelah perangkat diberi low-pass filter dan juga kalman filter ($w=0.00001$) dengan RMSE (*Root Mean Square Error*) sebesar 5.83 yang merupakan hasil reduksi *error* sebesar 92.45%, dan keakurasian pengukuran tertinggi sebesar 99.86%.
4. Kinerja kalman filter bergantung pada parameter w . semakin besar parameter, kalman bekerja semakin cepat, namun banyak error yang tidak direduksi, begitupun sebaliknya.

B. Saran

Untuk penelitian selanjutnya, perlu diperhatikan beberapa hal berikut :

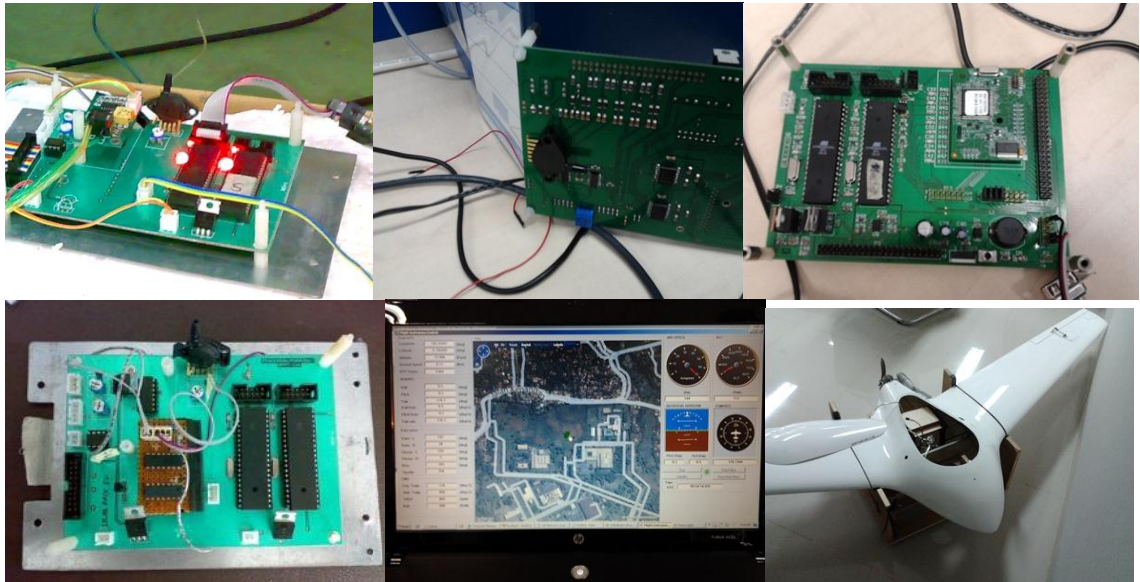
1. Penggunaan jaringan saraf tiruan (*Artificial Neural Network*) dalam mencari parameter w , dapat mempermudah dan mempercepat eksekusi kalman filter.
2. Pengukuran ketinggian PUNA sebaiknya dilakukan dengan variasi waktu. Missal pagi, siang, dan malam. Karena kondisi suhu dan kelembaban berbeda pada pagi,siang, dan sore.

DAFTAR PUSTAKA

- [1] Potter, Merle C, Ph.D.2009.*Fluid Mechanics Demystified*.ISBN: 978-0-07-162682-8. New York : The McGraw-Hill Companies, Inc.
- [2]Smith, Peter. 1950. *Dictionary of Applied Physics Volume 3*.New York : The McGraw-Hill Companies, Inc.
- [3] Prof. Onkar Singh.2009. *Applied Thermodynamics (third edition)*.ISBN (13) : 978-81-224-2916-9. New Delhi:New Age International (P) Ltd.
- [4] Zaloga, Steven J. 2008. *Unmanned Aerial Vehicles Robotic Air Warfare 1917–2007*. ISBN: 9781846032431. Osprey Publishing.
- [5] Pasahow,E. J. 1988. *Microprocessor Technology and Microcomputers*. New York:McGraw Hill.
- [6] Team PUNA BPPT. 2010. *Ground Control Station, GCS rev1.0, ES.WP.7.1.2*. Jakarta : PTIPK-BPPT.
- [7] Prajitno, Santoso Sugondo. Rancang Bangun Alat Pengukur Tekanan Gas. Jakarta : Puslitbang KIM LIPPI
- [8] Fatoni, Imas. Pengukuran Tinggi Permukaan Air Berbasis Gelombang Ultrasonik Menggunakan Kalman Filter. Jakarta : Fisika UNJ

LAMPIRAN

A. Gambar Alat



B. Hasil Pengukuran

2.1 Hasil Pengukuran Tanpa Filter

Tabel 4 Hasil Pengukuran Tanpa Filter

time(s)	Altitude(Feet)		
	Sensor	GPS	ISA
0.000	2243.52	2263.8	2345.078
0.297	2181.2	2263.8	2258.916
0.453	2243.52	2263.8	2373.799
0.609	2273.04	2263.8	2373.799
0.765	2273.04	2263.8	2373.799
0.906	2273.04	2263.8	2373.799
1.062	2243.52	2263.8	2373.799
1.250	2273.04	2263.8	2373.799
1.390	2243.52	2263.8	2373.799
1.547	2243.52	2263.8	2373.799
2.172	2335.36	2263.8	2373.799
2.437	2243.52	2263.8	2373.799
2.687	2243.52	2263.8	2373.799
2.937	2243.52	2263.8	2373.799

3.172	2243.52	2263.8	2373.799
3.437	2243.52	2263.8	2373.799
3.672	2243.52	2263.8	2373.799
3.953	2243.52	2263.8	2373.799
77.328	2364.88	2422.2	2546.123
77.687	2364.88	2422.2	2551.867
78.047	2500	2465.1	2557.611
78.547	2800	2471.7	2563.355
79.359	2519.04	2481.6	2569.099
79.937	2640	2494.8	2574.843
80.468	2380	2494.8	2580.588
80.875	2700	2494.8	2586.332
81.234	2600	2494.8	2592.076
81.547	2400	2494.8	2597.82
81.828	2620	2494.8	2603.564
82.156	2400	2517.9	2609.308
82.609	2680	2517.9	2615.052
83.406	2680	2524.5	2620.797
83.828	2364.88	2531.1	2626.541
...
234.984	2089.36	2089.36	2258.916
235.593	2089.36	2089.36	2258.916
236.109	2089.36	2089.36	2258.916
236.656	2089.36	2089.36	2258.916
237.312	2089.36	2089.36	2258.916
237.875	2089.36	2089.36	2258.916
238.531	2089.36	2089.36	2258.916
239.234	2089.36	2089.36	2258.916

2.2 Hasil Pengukuran dengan Filter

Tabel 5 Hasil Pengukuran dengan Filter

time(s)	press (kPa)	altitude (Feet)							
		LowPass	GPS	ISA	LP + K (W=0.1)	LP + K (W=0.01)	LP + K (W=0.001)	LP + K (W=0.0001)	LP + K (W=0.00001)
0.00	92.20	2587.68	2414.08	2589.20	2587.68	2587.68	2587.68	2587.68	2587.68
0.27	92.20	2587.68	2414.08	2589.20	2587.68	2587.68	2587.68	2587.68	2587.68
0.34	92.20	2587.68	2414.08	2589.20	2587.68	2587.68	2587.68	2587.68	2587.68
0.44	92.20	2587.68	2414.08	2589.20	2587.68	2587.68	2587.68	2587.68	2587.68
0.53	92.20	2587.68	2414.08	2589.20	2587.68	2587.68	2587.68	2587.68	2587.68
0.62	92.20	2587.68	2414.08	2589.20	2587.68	2587.68	2587.68	2587.68	2587.68

0.75	92.20	2587.68	2414.08	2589.20	2587.68	2587.68	2587.68	2587.68	2587.68
0.87	92.20	2587.68	2414.08	2589.20	2587.68	2587.68	2587.68	2587.68	2587.68
0.97	92.20	2587.68	2414.08	2589.20	2587.68	2587.68	2587.68	2587.68	2587.68
1.08	92.20	2587.68	2414.08	2589.20	2587.68	2587.68	2587.68	2587.68	2587.68
1.19	92.20	2587.68	2414.08	2589.20	2587.68	2587.68	2587.68	2587.68	2587.68
1.31	92.20	2587.68	2414.08	2589.20	2587.68	2587.68	2587.68	2587.68	2587.68
1.39	92.20	2587.68	2414.08	2589.20	2587.68	2587.68	2587.68	2587.68	2587.68
1.50	92.20	2587.68	2414.08	2589.20	2587.68	2587.68	2587.68	2587.68	2587.68
1.62	92.20	2587.68	2414.08	2589.20	2587.68	2587.68	2587.68	2587.68	2587.68
.....									
309.88	91.50	2794.62	2788.00	2788.47	2794.62	2794.62	2794.62	2794.56	2792.78
310.49	91.50	2794.62	2788.00	2788.52	2794.62	2794.62	2794.62	2794.57	2792.84
311.05	91.50	2794.62	2788.00	2788.58	2794.62	2794.62	2794.62	2794.57	2792.89
311.24	91.50	2794.62	2788.00	2788.63	2794.62	2794.62	2794.62	2794.57	2792.95
311.44	91.50	2794.62	2788.00	2788.68	2794.62	2794.62	2794.62	2794.58	2793.00
312.00	91.50	2794.62	2788.00	2788.73	2794.62	2794.62	2794.62	2794.58	2793.05
312.22	91.50	2794.62	2788.00	2788.78	2794.62	2794.62	2794.62	2794.59	2793.10
312.73	91.50	2794.62	2788.00	2788.82	2794.62	2794.62	2794.62	2794.59	2793.15
313.37	91.50	2794.62	2788.00	2788.87	2794.62	2794.62	2794.62	2794.59	2793.19
313.37	91.50	2794.62	2788.00	2788.91	2794.62	2794.62	2794.62	2794.59	2793.24
314.00	91.60	2764.97	2788.00	2788.95	2794.62	2794.62	2794.62	2794.60	2793.28
314.57	91.50	2794.62	2788.00	2788.10	2767.46	2776.30	2786.61	2791.78	2792.40
315.12	91.50	2794.62	2788.00	2788.16	2792.34	2787.62	2788.77	2792.05	2792.47
315.12	91.50	2794.62	2788.00	2788.23	2794.42	2791.94	2790.35	2792.29	2792.53
315.82	91.50	2794.62	2788.00	2788.29	2794.60	2793.59	2791.50	2792.51	2792.60
316.46	91.50	2794.62	2788.00	2788.35	2794.61	2794.23	2792.34	2792.71	2792.66
316.46	91.50	2794.62	2788.00	2788.41	2794.62	2794.47	2792.96	2792.89	2792.72
316.74	91.50	2794.62	2788.00	2788.47	2794.62	2794.56	2793.41	2793.06	2792.78
317.37	91.50	2794.62	2788.00	2788.52	2794.62	2794.59	2793.73	2793.21	2792.84
317.91	91.50	2794.62	2788.00	2788.58	2794.62	2794.61	2793.97	2793.34	2792.89
318.21	91.50	2794.62	2788.00	2788.63	2794.62	2794.61	2794.15	2793.46	2792.95
319.02	91.50	2794.62	2788.00	2788.68	2794.62	2794.61	2794.27	2793.57	2793.00
319.55	91.50	2794.62	2788.00	2788.73	2794.62	2794.62	2794.37	2793.67	2793.05
320.28	91.50	2794.62	2788.00	2788.78	2794.62	2794.62	2794.43	2793.76	2793.10
320.53	91.50	2794.62	2788.00	2788.82	2794.62	2794.62	2794.48	2793.84	2793.15
321.28	91.50	2794.62	2788.00	2788.87	2794.62	2794.62	2794.52	2793.92	2793.19
321.53	91.50	2794.62	2788.00	2788.91	2794.62	2794.62	2794.54	2793.98	2793.24
322.11	91.50	2794.62	2788.00	2788.95	2794.62	2794.62	2794.56	2794.04	2793.28
322.11	91.50	2794.62	2788.00	2788.99	2794.62	2794.62	2794.58	2794.10	2793.32
322.36	91.50	2794.62	2788.00	2789.03	2794.62	2794.62	2794.59	2794.15	2793.36
323.00	91.50	2794.62	2788.00	2789.07	2794.62	2794.62	2794.60	2794.19	2793.40

323.56	91.50	2794.62	2788.00	2789.11	2794.62	2794.62	2794.60	2794.23	2793.44
324.31	91.50	2794.62	2788.00	2789.14	2794.62	2794.62	2794.61	2794.27	2793.47
324.31	91.50	2794.62	2788.00	2789.18	2794.62	2794.62	2794.61	2794.30	2793.51
324.67	91.50	2794.62	2788.00	2789.21	2794.62	2794.62	2794.61	2794.33	2793.54
325.34	91.50	2794.62	2788.00	2789.24	2794.62	2794.62	2794.61	2794.36	2793.58
325.60	91.50	2794.62	2788.00	2789.27	2794.62	2794.62	2794.61	2794.38	2793.61
.....									
616.95	91.60	2764.97	2414.08	2714.53	2764.97	2764.42	2739.85	2708.85	2716.65
617.50	91.60	2764.97	2414.08	2715.99	2764.97	2764.76	2746.64	2714.19	2718.16
618.10	91.60	2764.97	2414.08	2717.41	2764.97	2764.89	2751.59	2719.02	2719.62
618.43	91.60	2764.97	2414.08	2718.78	2764.97	2764.94	2755.21	2723.39	2721.03
619.80	91.60	2764.97	2414.08	2720.11	2764.97	2764.96	2757.85	2727.35	2722.40
620.35	91.60	2764.97	2414.08	2721.40	2764.97	2764.97	2759.77	2730.93	2723.72
620.87	91.60	2764.97	2414.08	2722.65	2764.97	2764.97	2761.18	2734.17	2725.01
621.38	91.60	2764.97	2414.08	2723.86	2764.97	2764.97	2762.20	2737.10	2726.25
621.38	91.60	2764.97	2414.08	2725.03	2764.97	2764.97	2762.95	2739.75	2727.45
622.32	91.60	2764.97	2414.08	2726.17	2764.97	2764.97	2763.50	2742.15	2728.62
622.83	91.60	2764.97	2414.08	2727.27	2764.97	2764.97	2763.90	2744.32	2729.75
623.35	91.60	2764.97	2414.08	2728.34	2764.97	2764.97	2764.19	2746.28	2730.85
623.35	91.60	2764.97	2414.08	2729.37	2764.97	2764.97	2764.40	2748.06	2731.91
623.85	91.60	2764.97	2414.08	2730.37	2764.97	2764.97	2764.56	2749.67	2732.94

C. List Program

3.1 List Program pada Mikrokontroler

```

/*****
Chip type      : ATmega8535
Program type   : Application
Clock frequency : 8.000000 MHz
Memory model   : Small
External RAM size : 0
Data Stack size : 128
*****/

#include <mega8535.h>

// Standard Input/Output functions
#include <stdio.h>
#include <delay.h>
#define ADC_VREF_TYPE 0x00

```

```

// Read the AD conversion result
unsigned int read_adc(unsigned char adc_input)
{
  ADMUX=adc_input | (ADC_VREF_TYPE & 0xff);
  // Delay needed for the stabilization of the ADC input voltage
  delay_us(10);
  // Start the AD conversion
  ADCSRA|=0x40;
  // Wait for the AD conversion to complete
  while ((ADCSRA & 0x10)==0);
  ADCSRA|=0x10;
  return ADCW;
}

// Declare your global variables here
Int adc_bar, avr[9],press,KPA,altbar;
void main(void)
{
  // Declare your local variables here

  // Input/Output Ports initialization
  // Port A initialization
  // Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In
  Func0=In
  // State7=T State6=T State5=T State4=T State3=T State2=T State1=T
  State0=T
  PORTA=0x00;
  DDRA=0x00;

  // Port B initialization
  // Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In
  Func0=In
  // State7=T State6=T State5=T State4=T State3=T State2=T State1=T
  State0=T
  PORTB=0x00;
  DDRB=0x00;

  // Port C initialization
  // Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In
  Func0=In
  // State7=T State6=T State5=T State4=T State3=T State2=T State1=T
  State0=T
  PORTC=0x00;
  DDRC=0x00;

  // Port D initialization

```

```
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In
Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T
State0=T
PORTD=0x00;
DDRD=0x00;
// Timer/Counter 0 initialization
// Clock source: System Clock
// Clock value: Timer 0 Stopped
// Mode: Normal top=FFh
// OC0 output: Disconnected
TCCR0=0x00;
TCNT0=0x00;
OCR0=0x00;

// Timer/Counter 1 initialization
// Clock source: System Clock
// Clock value: Timer 1 Stopped
// Mode: Normal top=FFFFh
// OC1A output: Discon.
// OC1B output: Discon.
// Noise Canceler: Off
// Input Capture on Falling Edge
// Timer 1 Overflow Interrupt: Off
// Input Capture Interrupt: Off
// Compare A Match Interrupt: Off
// Compare B Match Interrupt: Off
TCCR1A=0x00;
TCCR1B=0x00;
TCNT1H=0x00;
TCNT1L=0x00;
ICR1H=0x00;
ICR1L=0x00;
OCR1AH=0x00;
OCR1AL=0x00;
OCR1BH=0x00;
OCR1BL=0x00;

// Timer/Counter 2 initialization
// Clock source: System Clock
// Clock value: Timer 2 Stopped
// Mode: Normal top=FFh
// OC2 output: Disconnected
ASSR=0x00;
TCCR2=0x00;
TCNT2=0x00;
```

```
OCR2=0x00;

// External Interrupt(s) initialization
// INT0: Off
// INT1: Off
// INT2: Off
MCUCR=0x00;
MCUCSR=0x00;

// Timer(s)/Counter(s) Interrupt(s) initialization
TIMSK=0x00;

// USART initialization
// Communication Parameters: 8 Data, 1 Stop, No Parity
// USART Receiver: On
// USART Transmitter: On
// USART Mode: Asynchronous
// USART Baud Rate: 9600
UCSRA=0x00;
UCSRB=0x18;
UCSRC=0x86;
UBRRH=0x00;
UBRRL=0x33;

// Analog Comparator initialization
// Analog Comparator: Off
// Analog Comparator Input Capture by Timer/Counter 1: Off
ACSR=0x80;
SFIOR=0x00;

// ADC initialization
// ADC Clock frequency: 1000.000 kHz
// ADC Voltage Reference: AREF pin
// ADC High Speed Mode: Off
// ADC Auto Trigger Source: None
ADMUX=ADC_VREF_TYPE & 0xff;
ADCSRA=0x83;
SFIOR&=0xEF;

while (1)
{

    adc_bar = read_adc(0); // ADC Barometrik
    press = (adc_bar*4.835)/40.3; // Pressure
    KPA = press;
```

```

altbar = (((273.15+Thsm)/0.0065) * (pow(KPA/101.325,-0.1902)-
1))*3.28; //Altitude Barometrik in Feet

printf(“%5.2f”,altbar);
};
}

```

3.2 List Program pada GCS PUNA dengan Kalman Filter

```

//program Utama di dalam MainForm.cs
public double ALT;
public void data_received(object sender, EventArgs e)    {
    String data = serialPort1.ReadExisting();

    if (data.Length >= 153)
    {
        PTIPKUAV = data.Trim();
    }
    this.Invoke(new EventHandler(DisplayText));
    timer11.Enabled = true;
    this.Invoke(new EventHandler(olahdata));
}

private void DisplayText(object sender, EventArgs e)
{
    richTextBox1.AppendText(data);
}

public string[] s;
private void olahdata(object sender, EventArgs e)
{
    s = PTIPKUAV.Split(new string[] { " ", ",", "\n\r" }, 32,
        StringSplitOptions.RemoveEmptyEntries);
    id = PTIPKUAV.Substring(0, 5);
    if (id == "SRITI")
    {
        roll = (s[(1)]);
        pitch = (s[(2)]);
        yaw = (s[(3)]);
        Ax = (s[(4)]);
        Ay = (s[(5)]);
        Az = (s[(6)]);
        P = ((s[(7)]));
    }
}

```

```

    Q = ((s[8]));
    R = (s[9]);
    status = Convert.ToDouble(s[(16)]);
    if (s[(16)] == "0") { validitas = " Not Valid"; }
    else { validitas = "Valid"; }
    Altitude = (s[(17)]);
    gspeed = ((s[18]));
    CHTL = (s[(19)]);
    ALT = Convert.ToDouble(s[(20)]);
    batt = (s[(21)]);
    CHTR = (s[(22)]);
    airs = (Convert.ToDouble((s[(23)])));
    rpm = (s[(24)]);
    barometrik = ((s[31]));
}
}

```

```

private void GetKalman_class(object sender, EventArgs e)
{

```

```

    // kalman filter roll w = 0.0001
    KalmanSuper kalmanroll = new KalmanSuper(1, 1000,
    Convert.ToDouble(roll), z);
    z = kalmanroll.z;
    //kalman filter pitch w = 0.0001
    KalmanSuper kalmanpitch = new KalmanSuper(1, 1000,
    Convert.ToDouble(pitch), zp);
    zp = kalmanpitch.z;
    //kalman filter heading w = 0.0001
    KalmanSuper kalmanhead = new KalmanSuper(1, 1000,
    Convert.ToDouble(yaw), zy);
    zy = kalmanhead.z;
    //kalman filter altitude w = 0.0001
    KalmanSuper kalmanalt = new KalmanSuper(1, 1000, ALT, Kalt);
    Kalt = kalmanalt.z;
    //kalman filter airs w = 0.0001
    KalmanSuper kalmanairs = new KalmanSuper(1, 1000, airs, Kairs);
    Kairs = kalmanairs.z;

```

```

}

```

```

private void ShowResult(object sender, EventArgs e)
{

```

```

    This.Altimeter1_view.value = Kalt;
    This.Roll_view.value = z;
    This.Pitch_view.value = zp;
    This.Yaw_view.value = zy;
    This.Airspeed_view.value = Kairs;

```

```

}

Private void StoredToDatabase(object sender, EventArgs e)
{
    OnClassStoredDB.Stored.IntoTable(data);
}

//Kalman Filter class di dalam file KalmanSuper.cs

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace coba_puna_sriti_3_layar
{
    public class KalmanSuper
    {
        public double P0r, Pr, Pr2, Kr, xr;
        public double QQ { get; private set; }
        public double RR { get; private set; }
        public double yy { get; private set; }
        public double z { get; private set; }

        public KalmanSuper(double Qparameter, double Rparameter, double ynilai,
double hasil)
        {
            QQ = Qparameter;
            RR = Rparameter;
            yy = ynilai;
            z = hasil;
            double ti = 1;
            double P0r = 1;
            if (ti == 1) { xr = yy; Pr = P0r + QQ; }
            if (ti > 1)
            {
                xr = z; Pr = Pr2 + QQ;
            }
            Kr = Pr * (1 / (Pr + RR));
            z = xr + Kr * yy;
            Pr2 = (1 - Kr) * Pr;
        }

        public void Predict()

```



```
{
  X0 = F*State + (B*U);
  P0 = F*Covariance*F.Transpose() + Q;
}

public void hasilKalman(double z)
{
  double ti = 1;
  double P0r = 1;
  if (ti == 1) { xr = yy; Pr = P0r + QQ; }
  if (ti > 1)
  {
    xr = z; Pr = Pr2 + QQ;
  }
  Kr = Pr * (1 / (Pr + RR));
  z = xr + Kr * yy;
  Pr2 = (1 - Kr) * Pr;
}
}
}
```

RIWAYAT HIDUP



Muhammad Erwin, lahir di Bogor pada tanggal 28 Nopember 1988. Anak kelima dari 5 bersaudara dari pasangan Usman.B dan Allah.

Penulis menempuh pendidikan Sekolah Dasar di SDN Bangka 4 Bogor lulus tahun 2000. Pada tahun yang sama masuk SMPN 2 Bogor dan lulus tahun 2003, kemudian melanjutkan pendidikan ke SMAN 6 Bogor pada tahun yang sama dan lulus pada tahun 2006. Pada tahun yang sama masuk Universitas Negeri Jakarta Jurusan Fisika melalui seleksi nasional perguruan tinggi negeri (SNPTN). Selama perkuliahan penulis tidak aktif dalam organisasi kampus seperti BEM, tetapi penulis lebih aktif dalam pembelajaran perkuliahan, seperti menjadi asisten dosen, dan asisten laboratorium.