

Lampiran 1. RPS Mata Kuliah Kendali dan Akuisisi Data



KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI  
UNIVERSITAS NEGERI YOGYAKARTA  
FAKULTAS TEKNIK

## RENCANA PEMBELAJARAN SEMESTER

NO.:RPS/MEK/6235/2014

SEM: VI

SKS: 2P

Revisi: 01

Tanggal 20 Agustus 2015



PROGRAM STUDI : PENDIDIKAN TEKNIK MEKATRONIKA

MATA KULIAH : PRAKTIK KENDALI DAN AKUISISI DATA

DOSEN PENGAMPU : TIM

### I. DESKRIPSI MATA KULIAH

Perkuliahan Praktik Kendali dan Akuisisi Data untuk mengembangkan kemampuan mahasiswa agar dapat melakukan kerja praktik kendali dan akuisisi data untuk mesin-mesin kendali dan atau peralatan elektronik/elektrik dan dapat mengimplementasikannya untuk berbagai proses pengendalian dengan mengutamakan prinsip-prinsip antarmuka data onwire dan wireless dan pengolahan data antar sistem kendali berbasis mikroprosessor dan atau mikrkontroler baik secara hardware, software maupun hardware-software. Perkuliahan dilaksanakan dengan pendekatan *student center learning*, dengan model pembelajaran problem based dan case based. Penilaian berbasis kompetensi dengan melibatkan unsur partisipasi aktif, tugas individu dan kelompok, ujian tengah maupun ujian akhir semester.

### II. CAPAIAN PEMBELAJARAN MATA KULIAH

1. Bertaqwa kepada Tuhan YME dan mampu menunjukkan sikap religius dan berkarakter yang diimplementasikan dalam pembelajaran dan hasil belajarnya,
2. Mahasiswa berpartisipasi aktif, bertanggungjawab, disiplin, mampu bekerjasama, dan memiliki motivasi mengembangkan diri,
3. Mahasiswa mampu menganalisis berbagai aplikasi sistem kendali dan akuisisi data agar proses operasi menjadi lebih efektif dan efisien,
4. Memiliki kemampuan bekerja secara efektif, berpikir analitis dan kritis untuk membuat keputusan yang tepat serta cepat dalam berbagai proses sistem kendali.

### **III. MATRIK RENCANA PEMBELAJARAN**

#### IV. BOBOT PENILAIAN<sup>\*)</sup>

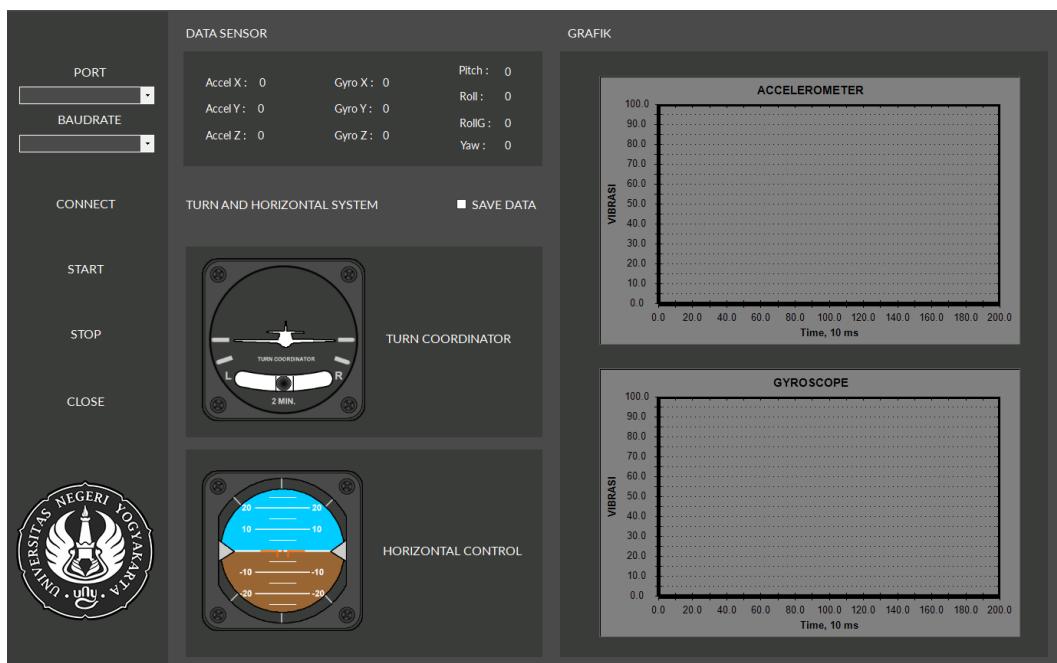
NO	ASPEK	JENIS TAGIHAN	NILAI MAKSIMAL	BOBOT
1	Kemampuan kognitif & Afektif	Tugas-tugas individu Tugas kelompok 2 sebelum UTS dan 2 setelah uts	Nilai berdasarkan akumulasi capaian skor setiap tagihan	40 %
		UTS <sup>*)</sup>	0-100	25 %
		UAS <sup>*)</sup>	0-100	25 %
2	Kehadiran	Hadir 100 %	100	10 %
		Tidak hadir satu kali	90	
		Tidak hadir dua kali	80	
		Tidak hadir tiga kali	70	
		Tidak hadir empat kali	60	

\*) Penilaian aspek, jenis penilaian dan pembobotan disesuaikan dengan capaian pembelajaran dan karakteristik mata kuliah

#### V. SUMBER BACAAN

1. Ghalnaraghi, F., and Kuo, B. 2010. *Automatic control systems*, USA: John Wesley Addison.
2. Houpis, C.H., & Lamont, G.B. (1992). *Digital control systems theory, hardware, software*. (2<sup>nd</sup> Ed.). New York: McGraw Hill, Inc.
3. Ogata (2006). *Automation control systems*. USA: Mc. Graw Hill.
4. Sukla, R.C. (2001). *Control Systems*. Delhi: Dhanpat Rai & Co. (Pvt.) Ltd.
5. Parck, J & Mackay, S (2003), *Practical Data Acquisition for Instrumentation and Control Systems*. British: Elsevier

## Lampiran 2. Foto Produk



### Lampiran 3. Validasi Ahli Media

ANGKET AHLI MEDIA

PENGEMBANGAN DRONE PEMANTAU GUNUNG BERAPI SEBAGAI MEDIA  
PEMBELAJARAN SENSOR ACCELEROMETER DAN GYROSCOPE PADA  
MATAKULIAH KENDALI DAN AKUISISI DATA

IDENTITAS RESPONDEN

NAMA RESPONDEN : Sigit Yatmono.....  
INSTANSI : .....



PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA  
JURUSAN PENDIDIKAN TEKNIK ELEKTRO  
FAKULTAS TEKNIK  
UNIVERSITAS NEGERI YOGYAKARTA

2019

## B. ANGKET

NO.	PERNYATAAN	JAWABAN			
		TS	KS	S	SS
1.	Penggunaan media pembelajaran membantu pembelajaran Kendali dan Akuisisi Data				✓
2.	Pengajar dapat menjelaskan materi dengan mudah menggunakan media pembelajaran sensor MPU6050			✓	
3.	Pengajar dapat menggunakan media pembelajaran sensor MPU6050 untuk menarik perhatian peserta didik			✓	
4.	Media pembelajaran sensor MPU6050 dapat menambah variasi materi tentang sensor				✓
5.	Keaktifan peserta didik meningkat dengan adanya media pembelajaran sensor MPU6050			✓	
6.	Penggunaan media pembelajaran sensor MPU6050 menumbuhkan semangat belajar peserta didik			✓	
7.	Media pembelajaran sensor MPU6050 berhubungan dengan materi mata kuliah lain				✓
8.	Media pembelajaran sensor MPU6050 dapat mendukung materi mata kuliah lain				✓
9.	Penggunaan STM32 sebagai controller menambah pengetahuan materi kendali dan akuisisi data			✓	
10.	Penggunaan drone sebagai actuator menambah pengetahuan materi kendali dan akuisisi data				✓
11.	Tampilan GUI pada media pembelajaran berfungsi dengan baik				✓
12.	Informasi pada tampilan GUI mudah dipahami.				✓
13.	Bentuk media pembelajaran menarik.				✓

14.	Peletakan komponen pada media pembelajaran tersusun dengan baik.			✓	
15.	STM32 pada media pembelajaran sensor MPU6050 mudah digunakan.			✓	
16.	Drone pada media pembelajaran sensor kamera mudah digunakan.			✓	
17.	MPU6050 yang digunakan pada media pembelajaran dapat bekerja dengan baik.				✓
18.	Semua komponen media pembelajaran berfungsi dengan baik.				✓
19.	Perakitan media pembelajaran dapat dilakukan dengan mudah.			✓	
20.	Modul membantu pengoperasian media pembelajaran lebih mudah.				✓
21.	Media pembelajaran menarik untuk digunakan.			✓	✓
22.	Media pembelajaran mudah digunakan.			✓	

### C. KOMENTAR DAN SARAN

\* Mohon dibuat panduan per langkah untuk menggunakan modul dimulai dari persiapan, pengaturan hardware maupun software.

### D. KESIMPULAN

Media pembelajaran Drone Pemantau Gunung Berapi Sebagai Media Pembelajaran Sensor Accelerometer Dan Gyroscope Pada Matakuliah Kendali dan Akuisisi Data ini dinyatakan:

1. Layak digunakan tanpa revisi
- 2 Layak digunakan dengan revisi sesuai saran
3. Tidak layak digunakan

(Mohon beri tanda lingkaran pada nomor sesuai dengan kesimpulan Bapak/Ibu)

Yogyakarta, 22-3-2019

Ahli Media

(.....Sigit.....)

ANGKET AHL. MEDIA

PENGEMBANGAN DRONE PEMANTAU GUNUNG BERAPI SEBAGAI MEDIA  
PEMBELAJARAN SENSOR ACCELEROMETER DAN GYROSCOPE PADA  
MATAKULIAH KENDALI DAN AKUISISI DATA

IDENTITAS RESPONDEN

NAMA RESPONDEN : M. Khairudin.....  
INSTANSI : UNY.....



PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA  
JURUSAN PENDIDIKAN TEKNIK ELEKTRO  
FAKULTAS TEKNIK  
UNIVERSITAS NEGERI YOGYAKARTA

2019

B. ANGKET

NO.	PERNYATAAN	JAWABAN			
		TS	KS	S	SS
1.	Penggunaan media pembelajaran membantu pembelajaran Kendali dan Akuisisi Data				✓
2.	Pengajar dapat menjelaskan materi dengan mudah menggunakan media pembelajaran sensor MPU6050				✓
3.	Pengajar dapat menggunakan media pembelajaran sensor MPU6050 untuk menarik perhatian peserta didik			✓	
4.	Media pembelajaran sensor MPU6050 dapat menambah variasi materi tentang sensor				✓
5.	Keaktifan peserta didik meningkat dengan adanya media pembelajaran sensor MPU6050				✓
6.	Penggunaan media pembelajaran sensor MPU6050 menumbuhkan semangat belajar peserta didik				✓
7.	Media pembelajaran sensor MPU6050 berhubungan dengan materi mata kuliah lain yg relevan			✓	
8.	Media pembelajaran sensor MPU6050 dapat mendukung materi mata kuliah lain yg relevan			✓	
9.	Penggunaan STM32 sebagai controller menambah pengetahuan materi kendali dan akuisisi data			✓	
10.	Penggunaan drone sebagai actuator menambah pengetahuan materi kendali dan akuisisi data			✓	
11.	Tampilan GUI pada media pembelajaran berfungsi dengan baik				✓
12.	Informasi pada tampilan GUI mudah dipahami.			✓	
13.	Bentuk media pembelajaran menarik			✓	

Tata letak

14.	Peletakan komponen pada media pembelajaran tersusun dengan baik.				✓
15.	STM32 pada media pembelajaran sensor MPU6050 mudah digunakan.				✓
16.	Drone pada media pembelajaran sensor kamera mudah digunakan.				✓
17.	MPU6050 yang digunakan pada media pembelajaran dapat bekerja dengan baik.				✓
18.	Semua komponen media pembelajaran berfungsi dengan baik.				✓
19.	Perakitan media pembelajaran dapat dilakukan dengan mudah.				✓
20.	Modul membantu pengoperasian media pembelajaran lebih mudah.				✓
21.	Media pembelajaran menarik untuk digunakan.				✓
22.	Media pembelajaran mudah digunakan.				✓

### C. KOMENTAR DAN SARAN

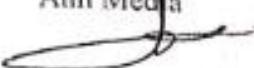
- 1) Sebaiknya desain perangkat sensor bisa secara plug & play untuk media pembelajaran
- 2) Untuk ke depan, data sensor accelerometer dan gyroscope dapat disimpan / direcord, tidak sekedar tampil
- 3) Editorial, lihat lampir

### D. KESIMPULAN

Media pembelajaran Drone Pemantau Gunung Berapi Sebagai Media Pembelajaran Sensor Accelerometer Dan Gyroscope Pada Mata Kuliah Kendali dan Akuisisi Data ini dinyatakan:

1. Layak digunakan tanpa revisi
2. Layak digunakan dengan revisi sesuai saran
3. Tidak layak digunakan

(Mohon beri tanda lingkaran pada nomor sesuai dengan kesimpulan Bapak/Ibu)

Yogyakarta, ..... 25/7/2014  
Ahli Media  
  
(.....)

#### Lampiran 4. Validasi Ahli Materi

**ANGKET UJI MATERI**

**PENGEMBANGAN DRONE PEMANTAU GUNUNG BERAPI SEBAGAI MEDIA  
PEMBELAJARAN SENSOR ACCELEROMETER DAN GYROSCOPE PADA  
MATAKULIAH KENDALI DAN AKUISISI DATA**

**IDENTITAS RESPONDEN**

NAMA RESPONDEN : *Sigit Yatmono*.....  
INSTANSI : .....



**PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA**

**JURUSAN PENDIDIKAN TEKNIK ELEKTRO**

**FAKULTAS TEKNIK**

**UNIVERSITAS NEGERI YOGYAKARTA**

**2019**

B. ANGKET

NO.	PERNYATAAN	JAWABAN			
		TS	KS	S	SS
1.	Materi pelajaran yang diajarkan menggunakan media pembelajaran sesuai dengan silabus			✓	
2.	Materi yang diajarkan menggunakan media sensor accelerometer dan gyroscope relevan dengan kompetensi peserta didik pada bidang Kendali dan Akuisisi Data				✓
3.	Materi pelajaran disajikan dengan jelas			✓	
4.	Petunjuk penggunaan media pembelajaran sensor accelerometer dan gyroscope dijelaskan di dalam modul			✓	
5.	Cara menghubungkan kontroler pada media pembelajaran sensor accelerometer dan gyroscope dijelaskan di dalam modul			✓	
6.	Cara menghubungkan drone pada media pembelajaran dijelaskan didalam modul				✓
7.	Fungsi program yang digunakan dalam pemrograman sensor accelerometer dan gyroscope dijelaskan didalam modul			✓	
8.	Materi tentang sensor accelerometer dan gyroscope disajikan dengan jelas				✓
9.	Materi tentang drone disajikan dengan jelas				✓
10.	Alur pengoperasian media pembelajaran disajikan dengan jelas			✓	
11.	Materi dapat dipahami setelah menggunakan media pembelajaran			✓	
12.	Media pembelajaran sensor accelerometer dan gyroscope dapat mendukung materi yang terdapat pada modul				✓
13.	Materi yang disajikan runtut				✓

14.	Terdapat penjelasan/keterangan pada setiap fungsi yang digunakan dalam program			✓	
15.	Meningkatkan motivasi belajar peserta didik			✓	
16.	Mengingkatkan keterlibatan peserta didik dalam pembelajaran Kendali dan Akuisisi Data				✓
17.	Materi yang disajikan sesuai dengan perkembangan intelektual peserta didik			✓	
18.	Materi yang disajikan sesuai dengan perkembangan emosional peserta didik			✓	
19.	Tata Bahasa yang digunakan pada modul sesuai dengan kaidah Bahasa Indonesia			✓	
20.	Istilah-istilah pada modul sudah baku				✓

#### C. KOMENTAR DAN SARAN

\* Materi GUI di modul ditambah dg paparan tentang bagian apa saja yg ada di tampilan GUI, cara pengaturan dan berbagaimana prosil data.  
 \* Untuk memenuhi karakteristik mata kuliah akuisisi data maka ditambah paparan/bagian modul yg cara mengumpulkan data drm sain file spreadsheet - excel dll.

#### D. KESIMPULAN

Media pembelajaran Drone Pemantau Gunung Berapi Sebagai Media Pembelajaran Sensor Accelerometer Dan Gyroscope Pada Matakuliah Kendali dan Akuisisi Data ini dinyatakan:

1. Layak digunakan tanpa revisi
- (2) Layak digunakan dengan revisi sesuai saran
3. Tidak layak digunakan

(Mohon beri tanda lingkaran pada nomor sesuai dengan kesimpulan Bapak/Ibu)

Yogyakarta, 22-7-2014.....

Ahli Media

(.....  
B1614 Y.....)

ANGKET UJI MATERI

PENGEMBANGAN DRONE PEMANTAU GUNUNG BERAPI SEBAGAI MEDIA  
PEMBELAJARAN SENSOR ACCELEROMETER DAN GYROSCOPE PADA  
MATAKULIAH KENDALI DAN AKUISISI DATA

IDENTITAS RESPONDEN

NAMA RESPONDEN

: .....Ariadi.....Chander.....N.....

INSTANSI

: .....DPTE.....UWY.....



PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA

JURUSAN PENDIDIKAN TEKNIK ELEKTRO

FAKULTAS TEKNIK

UNIVERSITAS NEGERI YOGYAKARTA

2019

B. ANGKET

NO.	PERNYATAAN	JAWABAN			
		TS	KS	S	SS
1.	Materi pelajaran yang diajarkan menggunakan media pembelajaran sesuai dengan silabus				✓
2.	Materi yang diajarkan menggunakan media sensor accelerometer dan gyroscope relevan dengan kompetensi peserta didik pada bidang Kendali dan Akuisisi Data				✓
3.	Materi pelajaran disajikan dengan jelas				✓
4.	Petunjuk penggunaan media pembelajaran sensor accelerometer dan gyroscope dijelaskan di dalam modul				✓
5.	Cara menghubungkan kontroler pada media pembelajaran sensor accelerometer dan gyroscope dijelaskan di dalam modul			✓	
6.	Cara menghubungkan drone pada media pembelajaran dijelaskan didalam modul			✓	
7.	Fungsi program yang digunakan dalam pemrograman sensor accelerometer dan gyroscope dijelaskan didalam modul				✓
8.	Materi tentang sensor accelerometer dan gyroscope disajikan dengan jelas				✓
9.	Materi tentang drone disajikan dengan jelas			✓	
10.	Alur pengoperasian media pembelajaran disajikan dengan jelas				✓
11.	Materi dapat dipahami setelah menggunakan media pembelajaran				✓
12.	Media pembelajaran sensor accelerometer dan gyroscope dapat mendukung materi yang terdapat pada modul				✓
13.	Materi yang disajikan runtut				✓

14.	Terdapat penjelasan/keterangan pada setiap fungsi yang digunakan dalam program			✓	
15.	Meningkatkan motivasi belajar peserta didik				✓
16.	Mengingkatkan keterlibatan peserta didik dalam pembelajaran Kendali dan Akuisisi Data				✓
17.	Materi yang disajikan sesuai dengan perkembangan intelektual peserta didik				✓
18.	Materi yang disajikan sesuai dengan perkembangan emosional peserta didik				✓
19.	Tata Bahasa yang digunakan pada modul sesuai dengan kaidah Bahasa Indonesia			✓	
20.	Istilah-istilah pada modul sudah baku			✓	

#### C. KOMENTAR DAN SARAN

1. Langkah-langkah pada labsheet diperjelas dengan ilustrasi atau minimal foto yang menunjukkan bagaimana menghubungkan drone dan komputer
2. Okurasi kurut kode program diperbesar
3. Informasi pemeriksaan port yang sesuai diberikan

#### D. KESIMPULAN

Media pembelajaran Drone Pemantau Gunung Berapi Sebagai Media Pembelajaran Sensor Accelerometer Dan Gyroscope Pada Mata Kuliah Kendali dan Akuisisi Data ini dinyatakan:

1. Layak digunakan tanpa revisi
2. Layak digunakan dengan revisi sesuai saran
3. Tidak layak digunakan

(Mohon beri tanda lingkaran pada nomor sesuai dengan kesimpulan Bapak/Ibu)

Yogyakarta, 18 Juli 2019

Khli Media  
  
 .....  
 (.....)

Ariadie Chandra

## Lampiran 5. Penilaian Pengguna

**ANGKET UJI PENGGUNA**

**PENGEMBANGAN DRONE PEMANTAU GUNUNG BERAPI SEBAGAI MEDIA  
PEMBELAJARAN SENSOR MPU6050 PADA MATAKULIAH KENDALI DAN  
AKUISISI DATA**

**IDENTITAS RESPONDEN**

**NAMA RESPONDEN** : Hermawati Galih M.....  
**INSTANSI** : UNY.....



**PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA  
JURUSAN PENDIDIKAN TEKNIK ELEKTRO  
FAKULTAS TEKNIK  
UNIVERSITAS NEGERI YOGYAKARTA  
2019**

**ANGKET**

NO	PERNYATAAN	JAWABAN			
		TS	KS	S	SS
1.	Saya mudah memahami kalimat yang ada dalam jobsheet			✓	
2.	Materi yang ada dalam jobsheet mudah untuk saya pahami				✓
3.	Contoh program pada jobsheet mudah untuk saya pahami			✓	
4.	Materi yang disajikan sesuai dengan materi mata kuliah Kendali dan Akuisisi Data			✓	
5.	Materi yang disajikan berisi kompetensi yang saya butuhkan			✓	
6.	Ilustrasi dalam <i>jobsheet</i> mempermudah saya dalam praktikum.				✓
7.	Langkah kerja dalam <i>jobsheet</i> mudah untuk saya ikuti			✓	
8.	Bagian-bagian modul sensor MPU6050 pada matakuliah kendali dan akuisisi data tidak membuat saya bingung.			✓	
9.	Saya dapat mengoperasikan modul sensor MPU6050 dengan mudah.				✓
10.	GUI untuk mengendalikan modul sensor MPU6050 berfungsi dengan baik.				✓
11.	Desain media pembelajaran menarik.			✓	
12.	Modul sensor MPU6050 memberi saya tambahan pengetahuan dalam pemrograman sensor MPU6050 pada mata kuliah Kendali dan Akuisisi Data			✓	
13.	Modul sensor MPU6050 memberi saya tambahan pengetahuan tentang sensor pada robot.				✓

14.	Media pembelajaran dapat membantu saya memahami materi pada mata kuliah lain.		<input checked="" type="checkbox"/>	
15.	Modul sensor MPU6050 membantu saya mempelajari pemrograman sensor MPU6050 pada mata kuliah praktik Kendali dan Akuisisi Data		<input checked="" type="checkbox"/>	
16.	Modul sensor MPU6050 memotivasi saya untuk belajar pemrograman sensor MPU6050 pada mata kuliah Kendali dan Akuisisi Data.			<input checked="" type="checkbox"/>
17.	Modul sensor MPU6050 menambah kompetensi saya dalam pemrograman Kendali dan Akuisisi Data			<input checked="" type="checkbox"/>
18.	Modul sensor MPU6050 dapat meningkatkan keaktifan saya saat pembelajaran Kendali dan Akuisisi Data		<input checked="" type="checkbox"/>	

KOMENTAR DAN SARAN

GUI menarik, kabel di drone dirapikan lagi

Yogyakarta, 23 Juli 2019

Peserta Didik

(Hermawan Galih M.....)

ANGKET UJI PENGGUNA

PENGEMBANGAN DRONE PEMANTAU GUNUNG BERAPI SEBAGAI MEDIA  
PEMBELAJARAN SENSOR MPU6050 PADA MATAKULIAH KENDALI DAN  
AKUISISI DATA

IDENTITAS RESPONDEN

NAMA RESPONDEN  
INSTANSI

: Nur Ichalish Mulya-  
: Universitas Negeri Yogyakarta -



PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA  
JURUSAN PENDIDIKAN TEKNIK ELEKTRO  
FAKULTAS TEKNIK  
UNIVERSITAS NEGERI YOGYAKARTA  
2019

**ANGKET**

NO	PERNYATAAN	JAWABAN			
		TS	KS	S	SS
1.	Saya mudah memahami kalimat yang ada dalam jobsheet				✓
2.	Materi yang ada dalam jobsheet mudah untuk saya pahami			✓	
3.	Contoh program pada jobsheet mudah untuk saya pahami				✓
4.	Materi yang disajikan sesuai dengan materi mata kuliah Kendali dan Akuisisi Data				✓
5.	Materi yang disajikan berisi kompetensi yang saya butuhkan			✓	
6.	Ilustrasi dalam <i>jobsheet</i> mempermudah saya dalam praktikum.				✓
7.	Langkah kerja dalam <i>jobsheet</i> mudah untuk saya ikuti			✓	
8.	Bagian-bagian modul sensor MPU6050 pada matakuliah kendali dan akuisisi data tidak membuat saya bingung.			✓	
9.	Saya dapat mengoperasikan modul sensor MPU6050 dengan mudah.				✓
10.	GUI untuk mengendalikan modul sensor MPU6050 berfungsi dengan baik.			✓	
11.	Desain media pembelajaran menarik.				✓
12.	Modul sensor MPU6050 memberi saya tambahan pengetahuan dalam pemrograman sensor MPU6050 pada mata kuliah Kendali dan Akuisisi Data				
13.	Modul sensor MPU6050 memberi saya tambahan pengetahuan tentang sensor pada robot.				✓

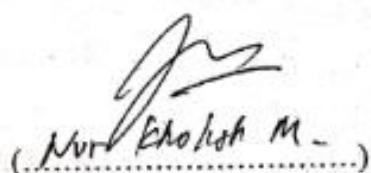
14.	Media pembelajaran dapat membantu saya memahami materi pada mata kuliah lain.				<input checked="" type="checkbox"/>
15.	Modul sensor MPU6050 membantu saya mempelajari pemrograman sensor MPU6050 pada mata kuliah praktik Kendali dan Akuisisi Data			<input checked="" type="checkbox"/>	
16.	Modul sensor MPU6050 memotivasi saya untuk belajar pemrograman sensor MPU6050 pada mata kuliah Kendali dan Akuisisi Data.			<input checked="" type="checkbox"/>	
17.	Modul sensor MPU6050 menambah kompetensi saya dalam pemrograman Kendali dan Akuisisi Data			<input checked="" type="checkbox"/>	
18.	Modul sensor MPU6050 dapat meningkatkan keaktifan saya saat pembelajaran Kendali dan Akuisisi Data			<input checked="" type="checkbox"/>	

KOMENTAR DAN SARAN

Modul sangat memudahkan dalam praktik karena sudah banyak gambar & ilustrasi.

Yogyakarta, 22 Juli 19

Peserta Didik



(Nur Eko Hoh M.)

Lampiran 6. Analisis Data

**Analisis Data Ahli Materi**

No	Validator	Aspek																		Total	Ktg								
		Kesesuaian media pembelajaran dengan materi												Jml	Ktg	Penyajian				Jml	Ktg								
		1	2	3	4	5	6	7	8	9	10	11	12			13	14	15	16										
1	Sigit Yatmono	3	4	3	3	3	4	3	4	4	3	3	4	41.0	SL	4	3	3	4	14.0	SL	3	3	3	4	13.0	L	68.0	SL
2	Ariadie Chandra N	4	4	4	4	3	3	4	4	3	4	4	4	45.0	SL	4	3	4	4	15.0	SL	4	4	3	3	14.0	SL	74.0	SL
		Jumlah												86.0					29.0					27.0		142.0			
		Rata-Rata												43.0	SL					14.5	SL				13.5	L	71.0	SL	
		Presentase												89.6	%					90.6	%				84.4	%	88.8	%	
		Jumlah Butir												12					4					4		20.0			
		Skor Maks												48					16					16		80.0			
		Skor Min												12					4					4		20.0			
		Rerata Ideal												30					10					10		50.0			
		Simpangan Ideal												6					2					2		10.0			

Kategori Penilaian	Interval Aspek Relevansi	Interval Aspek Penyajian	Interval Aspek Bahasa	Keseluruhan	Ket
Sangat Layak	40.8 ≤ X	13.6 ≤ X	13.6 ≤ X	68.0 ≤ X	SL
Layak	33.6 ≤ X < 40.8	11.2 ≤ X < 13.6	11.2 ≤ X < 13.6	56.0 ≤ X < 68.0	L
Cukup Layak	26.4 ≤ X < 33.6	8.8 ≤ X < 11.2	8.8 ≤ X < 11.2	44.0 ≤ X < 56.0	CL
Kurang Layak	19.2 ≤ X < 26.4	6.4 ≤ X < 8.8	6.4 ≤ X < 8.8	32.0 ≤ X < 44.0	KL
Sangat Kurang	X < 19.2	X < 6.4	X < 6.4	X < 32.0	SK

**Analisis Data Ahli Media**

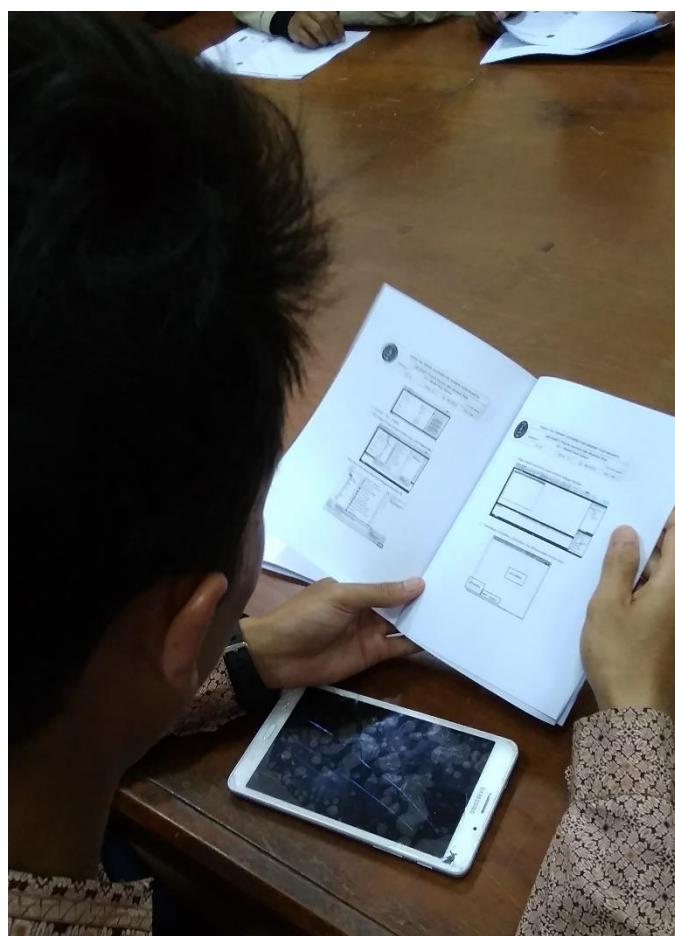
No	Validator	Aspek																Total	Ktg												
		Kemanfaatan								perangkat media								kemudahan													
		Jml	Ktg	9	10	11	12	13	14	15	16	17	18	Jml	Ktg	19	20	21	22												
1	Sigit Yatmono, M.T.	4	3	3	4	3	3	4	4	28.0	SL	3	4	4	4	3	3	3	4	36.0	SL	3	4	4	3		14.0	SL	78.0	SL	
2	M. Khoirudin, Ph.D.	4	4	3	4	4	4	3	3	29.0	SL	3	3	4	3	3	3	3	4	4	34.0	SL	3	4	3	4		14.0	SL	77.0	SL
		Jumlah								70.0								28.0								155.0					
		Rata-Rata								28.5								35.0								14.0					
		Presentase								89.1 %								87.5 %								87.5 %					
		Jumlah Butir								8								10								4					
		Skor Maks								32								40								16					
		Skor Min								8								10								4					
		Rerata Ideal								20								25								10					
		Simpangan Ideal								4								5								2					
																										11.0					

Kategori Penilaian	Interval Aspek Edukatif				Interval Aspek Teknik Pembuatan				Interval Aspek Estetika				Total				Ket				
	27.2	≤	X	<	34.0	≤	X	<	13.6	≤	X	<	74.8	≤	X	<	SL				
Sangat Layak	27.2	≤	X	<	34.0	≤	X	<	13.6	≤	X	<	74.8	≤	X	<	SL				
Layak	22.4	≤	X	<	27.2	28.0	≤	X	<	34.0	11.2	≤	X	<	13.6	61.6	≤	X	<	74.8	L
Cukup Layak	17.6	≤	X	<	22.4	22.0	≤	X	<	28.0	8.8	≤	X	<	11.2	48.4	≤	X	<	61.6	CL
Kurang Layak	12.8	≤	X	<	17.6	16.0	≤	X	<	22.0	6.4	≤	X	<	8.8	35.2	≤	X	<	48.4	KL
Sangat Kurang			X	<	12.8		X	<	16.0		X	<	6.4		X	<	35.2		X	<	SK

**Analisis Data Uji Pengguna**

No	Nama Responden	Aspek																							
		Kualitas Isi & Tujuan						Penggunaan				Kualitas Pembelajaran								Jml	Kigr	Total	Kigr		
		1	2	3	4	5	6	15	16	17	18	7	8	9	10	11	12	13	14						
1	M. Bregas b	3	4	4	4	3	4	22	SL	3	4	3	4	14	SL	3	4	3	4	4	30	SL	66	SL	
2	rino prihantoro	3	3	4	4	3	2	19	L	3	4	3	4	14	SL	3	3	4	3	4	3	26	L	59	L
3	arie dwi kusumo	3	4	3	4	4	4	22	SL	3	3	3	3	12	L	4	4	4	4	4	3	29	SL	63	SL
4	dwi krisnaningrum	3	3	3	4	3	4	20	L	4	3	4	3	14	SL	4	3	3	3	4	3	28	SL	62	SL
5	m. ikhsani	4	3	3	4	3	3	20	L	4	4	4	4	16	SL	4	3	3	4	4	3	29	SL	65	SL
6	ema safitri	3	3	3	3	3	3	18	L	3	3	3	4	13	L	3	3	3	3	4	3	25	L	56	L
7	fany retnanigtyas	3	3	4	3	3	4	20	L	3	3	3	3	12	L	4	3	3	3	4	3	27	L	59	L
8	nur kholish mujib	4	3	4	4	3	4	22	SL	3	3	3	3	12	L	3	3	4	3	4	4	25	L	59	L
9	aqwa abdillah	4	3	3	4	4	3	21	SL	3	3	3	3	12	L	3	3	3	4	4	3	27	L	60	L
10	rafid zuhdi nugroho	4	4	4	4	4	4	24	SL	4	4	4	4	16	SL	4	4	4	3	4	4	31	SL	71	SL
11	ma'ruf wahyup	4	4	4	4	4	4	24	SL	4	4	4	4	16	SL	4	4	4	4	4	4	32	SL	72	SL
12	wilda fiqurilila	4	4	3	3	3	4	21	SL	4	3	3	4	14	SL	4	3	3	4	4	4	28	SL	63	SL
13	choirul abdul j. b.	4	3	4	4	4	3	22	SL	4	4	4	4	16	SL	3	3	3	4	4	3	28	SL	66	SL
14	hermawan galih m	3	4	3	3	3	4	20	L	3	4	4	3	14	SL	3	3	4	4	3	3	27	L	61	L
15	ikhsan sahida	3	3	4	3	3	4	20	L	4	3	3	3	13	L	3	3	3	3	3	3	24	L	57	L
16	krisnasiwi c. bayu	3	3	3	4	3	4	20	L	4	3	4	4	15	SL	3	3	3	4	4	3	27	L	62	SL
17	allamanda m. e. s.	3	3	3	3	3	3	18	L	3	2	3	2	10	CL	3	3	3	3	3	2	23	L	51	L
18							0	SK					0	SK							0	SK	0	SK	
19							0	SK					0	SK							0	SK	0	SK	
20							0	SK					0	SK							0	SK	0	SK	
		Jumlah		353						233											466		1052		
		Rata-Rata		17.7	L					11.7	L										23.3	L	52.6	L	
		Presentase		73.5 %						72.8 %											72.8 %		73.1 %		
		Jumlah Butir		6						4										8		18			
		Skor Maks		24						16										32		72			
		Skor Min		6						4										8		18			
		Rerata Ideal		15						10										20		45			

Lampiran 7. Dokumentasi



Lampiran 8. SK Dosen Pembimbing

**KEPUTUSAN DEKAN FAKULTAS TEKNIK  
UNIVERSITAS NEGERI YOGYAKARTA  
NOMOR : 15/PMEK/PB/IV/2019**

**TENTANG  
PENGANGKATAN DOSEN PEMBIMBING TUGAS AKHIR SKRIPSI (TAS) MAHASISWA  
FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA**

**DEKAN FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA**

- Menimbang : a. bahwa untuk kelancaran pelaksanaan kegiatan Tugas Akhir Skripsi (TAS) mahasiswa, dipandang perlu mengangkat dosen pembimbingnya;  
b. bahwa untuk keperluan sebagaimana dimaksud pada huruf a perlu menetapkan Keputusan Dekan Tentang Pengangkatan Dosen Pembimbing Tugas Akhir Skripsi (TAS) Fakultas Teknik Universitas Negeri Yogyakarta.
- Mengingat : 1. Undang-undang RI Nomor 20 Tahun 2003 Tentang Sistem Pendidikan Nasional (Lembaran Negara Tahun 2003 Nomor 78, Tambahan Lembaran Negara Nomor 4301);  
2. Peraturan Pemerintah Republik Indonesia Nomor 4 Tahun 2014 Tentang Penyelenggaraan Pendidikan Tinggi dan Pengelolaan Perguruan Tinggi (Lembaran Negara Tahun 2014 Nomor 16, Tambahan Lembaran Negara Republik Indonesia Nomor 5500);  
3. Keputusan Presiden Republik Indonesia Nomor 93 Tahun 1999 Tentang Perubahan Institut Keguruan dan Ilmu Pendidikan menjadi Universitas;  
4. Peraturan Mendiknas RI Nomor 23 Tahun 2011 Tentang Organisasi dan Tata Kerja Universitas Negeri Yogyakarta;  
5. Peraturan Mendiknas RI Nomor 34 Tahun 2011 Tentang Statuta Universitas Negeri Yogyakarta;  
6. Keputusan Menteri Pendidikan dan Kebudayaan RI Nomor 98/MPK.A4/KP/2013 Tentang Pengangkatan Rektor Universitas Negeri Yogyakarta;  
7. Peraturan Rektor Nomor 2 Tahun 2014 tentang Peraturan Akademik;  
8. Keputusan Rektor Nomor 800/UN.34/KP/2016 tahun 2016 tentang Pengangkatan Dekan Fakultas Teknik Universitas Negeri Yogyakarta.

**MEMUTUSKAN**

Menetapkan : **KEPUTUSAN DEKAN TENTANG PENGANGKATAN DOSEN PEMBIMBING TUGAS AKHIR SKRIPSI (TAS) FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA.**

PERTAMA : Mengangkat Saudara :

Nama	:	Herlambang Sigit Pramono, S.T., M.Cs
NIP	:	19650829 199903 1 001
Pangkat/Golongan	:	Penata Tk.I, III/d
Jabatan Akademik	:	Lektor

sebagai Dosen Pembimbing Untuk mahasiswa penyusun Tugas Akhir Skripsi (TAS) :

Nama	:	Rizky Ghozia Ulhaq
NIM	:	15518244007
Prodi Studi	:	Pend. Teknik Mekatronika - S1
Judul Skripsi/TA	:	PENGEMBANGAN DRONE PEMANTAU GUNUNG BERAPI SEBAGAI MEDIA PEMBELAJARAN SENSOR ACCELEROMETER DAN GYROSCOPE PADA MATAKULIAH KENDALI DAN AKUISISI DATA



- KEDUA : Dosen Pembimbing sebagaimana dimaksud dalam Diktum PERTAMA bertugas merencanakan, mempersiapkan, melaksanakan, dan mempertanggungjawabkan pelaksanaan kegiatan bimbingan terhadap mahasiswa sebagaimana dimaksud dalam Diktum PERTAMA sampai mahasiswa dimaksud dinyatakan lulus.
- KETIGA : Biaya yang diperlukan dengan adanya Keputusan ini dibebankan pada Anggaran DIPA Universitas Negeri Yogyakarta Tahun 2019.
- KEEMPAT : Keputusan ini berlaku sejak tanggal 16 April 2019.

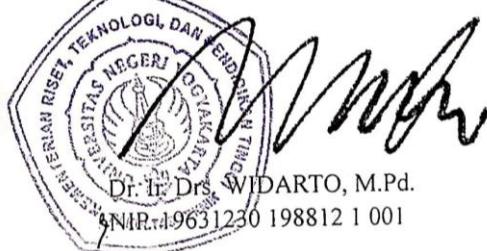
Tembusan Keputusan Dekan ini disampaikan kepada :

1. Para Wakil Dekan Fakultas Teknik;
2. Kepala Bagian Tata Usaha Fakultas Teknik;
3. Kepala Subbagian Keuangan dan Akuntansi Fakultas Teknik;
4. Kepala Subbagian Pendidikan Fakultas Teknik;
5. Ketua Jurusan Pendidikan Teknik Elektro Fakultas Teknik;
6. Mahasiswa yang bersangkutan;

Universitas Negeri Yogyakarta.

Ditetapkan di : Yogyakarta  
Pada tanggal : 16 April 2019

DEKAN FAKULTAS TEKNIK  
UNIVERSITAS NEGERI YOGYAKARTA,



### Lampiran 9. Program C# Arduino

```
#include "Wire.h"
#include "I2Cdev.h"
#include "MPU6050.h"
#if I2CDEV_IMPLEMENTATION == I2CDEV_ARDUINO_WIRE
    #include "Wire.h"
#endif
MPU6050 accelgyro;
int16_t ax, ay, az;
int16_t gx, gy, gz;
int16_t AccX, AccY, AccZ, GyX, GyY, GyZ;
int Ax, Ay, Az, Gx,Gy,Gz;
int val1;
int val2;
int val3;
int val4;
int val5;
int val6;
int prevVal1;
int prevVal2;
int prevVal3;
int prevVal4;
int prevVal5;
int prevVal6;

void setup()
{
    #if I2CDEV_IMPLEMENTATION == I2CDEV_ARDUINO_WIRE
        Wire.begin();
    #elif I2CDEV_IMPLEMENTATION == I2CDEV_BUILTIN_FASTWIRE
        Fastwire::setup(400, true);
    #endif

    accelgyro.initialize();
    Wire.begin();

    Serial.begin(57600);          //setup serial monitor
}

void loop()
{
    AccelGyro();
    copy_data();           // menampilkan data pengiriman di serial monitor
    kirim_data();
    delay(1000);
}

void AccelGyro()
{
    accelgyro.getMotion6(&ax, &ay, &az, &gx, &gy, &gz);
    Ax=((ax/257));
    Ay=((ay/257))-1;
```

```

Az=((az/257))-96;

Gx=(((gx/257))/4);
Gy=(((gy/257))/4);
Gz=(((gz/257))/4);

AccX = Ax;
AccY=Ay;
AccZ=Az;

GyX=Gx;
GyY=Gy;
GyZ=Gz;

// Program YPR
val1 = map(ax, -17000, 17000, -179, 179);
if (val1 != prevVal1)
{
    prevVal1 = val1;
}

val2 = map(ay, -17000, 17000, -45, 45);
if (val2 != prevVal2)
{
    prevVal2 = val2;
}

val3= map(gx, -17000, 17000, -60, 60);
if(val3 != prevVal3)
{
    prevVal3= val3;
}

val4= map(gz, -17000, 17000, -20, 20);
if(val4 != prevVal4)
{
    prevVal4= val4;
}

void copy_data()
{
    //pengiriman data Accel & Gyro
    Serial.print("<B0"); Serial.print(prevVal1);Serial.print("B0>");
    Serial.print("<B1"); Serial.print(prevVal2);Serial.print("B1>");
    Serial.print("<B2"); Serial.print(prevVal3);Serial.print("B2>");
    Serial.print("<B3"); Serial.print(prevVal4);Serial.print("B3>");
    Serial.print("<B4"); Serial.print(AccX);Serial.print("B4>");
    Serial.print("<B5"); Serial.print(AccY);Serial.print("B5>");
    Serial.print("<B6"); Serial.print(AccZ);Serial.print("B6>");
    Serial.print("<B7"); Serial.print(GyX);Serial.print("B7>");
    Serial.print("<B8"); Serial.print(GyY);Serial.print("B8>");
    Serial.print("<B9"); Serial.print(GyZ);Serial.print("B9>");
}

```

```
void kirim_data()
{
    //pengiriman data Accel & Gyro
    Serial2.print("<B0"); Serial2.print(prevVal1);Serial2.print("B0>");
    Serial2.print("<B1"); Serial2.print(prevVal2);Serial2.print("B1>");
    Serial2.print("<B2"); Serial2.print(prevVal3);Serial2.print("B2>");
    Serial2.print("<B3"); Serial2.print(prevVal4);Serial2.print("B3>");
    Serial2.print("<B4"); Serial2.print(AccX);Serial2.print("B4>");
    Serial2.print("<B5"); Serial2.print(AccY);Serial2.print("B5>");
    Serial2.print("<B6"); Serial2.print(AccZ);Serial2.print("B6>");
    Serial2.print("<B7"); Serial2.print(GyX);Serial2.print("B7>");
    Serial2.print("<B8"); Serial2.print(GyY);Serial2.print("B8>");
    Serial2.print("<B9"); Serial2.print(GyZ);Serial2.print("B9>");
}
```

## Lampiran 10. Program C# Visual Studio

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;
using System.IO.Ports;
using System.Collections;
using ZedGraph;
using System.Data.OleDb;

namespace AvionicsInstrumentControlDemo
{

    public partial class DemoWinow : Form
    {
        DataTable table = new DataTable();
        OleDbConnection akses_data = new
        OleDbConnection(@"Provider=Microsoft.Jet.OLEDB.4.0;Data
        Source=      'E:\Pelajaran\Semester 8\SKRIPSI 2019\3
        Drone\Database6.mdb'");

        string DataMasuk = "";
        int TickStartACC;
        int TickStartGYRO;
        string AccX, AccY, AccZ, GyroX, GyroY, GyroZ;
        String ACCX, ACCY, ACCZ, GYROX, GYROY, GYROZ, ROLL,
        PITCH, YAW;
        int Roll, Pitch, Yaw, Rollg;

        #region 2. ZEDGRAPH inisialisasi Curve
        //Accelero
        double timeAcc;
        LineItem curve_Acc1, curve_Acc2, curve_Acc3;
        RollingPointPairList list_Acc1 = new
        RollingPointPairList(60000);
        RollingPointPairList list_Acc2 = new
        RollingPointPairList(60000);
        RollingPointPairList list_Acc3 = new
        RollingPointPairList(60000);

        //Gyroscope
        double timeGyro;
        LineItem curve_Gyro1, curve_Gyro2, curve_gyro3;
        RollingPointPairList list_Gyro1 = new
        RollingPointPairList(60000);
```

```

        RollingPointPairList list_Gyro2 = new
RollingPointPairList(60000);
        RollingPointPairList list_Gyro3 = new
RollingPointPairList(60000);
        #endregion

        #region 3. ZEDGRAPH Inisisalisasi Vibrasi
private void graphVibrasiACC()
{
    GraphPane mypaneACC = zedGraphACC.GraphPane;
    //inisialisasi garis x,y,z
    curve_Acc1 = mypaneACC.AddCurve("X", list_Acc1,
Color.Red, SymbolType.None);
    curve_Acc1.Line.IsSmooth = true;
    curve_Acc1.Line.SmoothTension = 0.2F;

    curve_Acc2 = mypaneACC.AddCurve("Y", list_Acc2,
Color.Green, SymbolType.None);
    curve_Acc2.Line.IsSmooth = true;
    curve_Acc2.Line.SmoothTension = 0.2F;

    curve_Acc3 = mypaneACC.AddCurve("Z", list_Acc3,
Color.Blue, SymbolType.None);
    curve_Acc3.Line.IsSmooth = true;
    curve_Acc3.Line.SmoothTension = 0.2F;

    //aturan scale dari sumbu X
    mypaneACC.XAxis.Scale.Min = 0;
    mypaneACC.XAxis.Scale.Max = 100;
    mypaneACC.XAxis.Scale.MinorStep = 5;
    mypaneACC.XAxis.Scale.MajorStep = 10;
    mypaneACC.AxisChange();
//perubahan sumbu X realtime
    TickStartACC = Environment.TickCount + 200;
}

private void graphVibrasiGYRO()
{
    GraphPane mypaneGYRO = zedGraphGYRO.GraphPane;
    //inisialisasi garis x,y,z
    curve_Gyro1 = mypaneGYRO.AddCurve("X", list_Acc1,
Color.Red, SymbolType.None);
    curve_Gyro1.Line.IsSmooth = true;
    curve_Gyro1.Line.SmoothTension = 0.2F;

    curve_Gyro2 = mypaneGYRO.AddCurve("Y", list_Acc2,
Color.Green, SymbolType.None);
    curve_Gyro2.Line.IsSmooth = true;
    curve_Gyro2.Line.SmoothTension = 0.2F;
}

```

```

        curve_gyro3 = mypaneGYRO.AddCurve("Z", list_Acc3,
Color.Blue, SymbolType.None);
        curve_gyro3.Line.IsSmooth = true;
        curve_gyro3.Line.SmoothTension = 0.2F;

        //aturan scale dari sumbu X
        mypaneGYRO.XAxis.Scale.Min = 0;
        mypaneGYRO.XAxis.Scale.Max = 100;
        mypaneGYRO.XAxis.Scale.MinorStep = 5;
        mypaneGYRO.XAxis.Scale.MajorStep = 10;
        mypaneGYRO.AxisChange();
//perubahan sumbu X realtime
        TickStartGYRO = Environment.TickCount + 200;
    }

#endregion

#region 4. ZEDGRAPH Gambar Inputan
private void DrawVibrasiAcc(string setpoint, string
current1, string current2)
{
    //Make sure that the curvelist has at least one curve
    double intsetpoint;
    double intcurrent1;
    double intcurrent2;
    double.TryParse(setpoint, out intsetpoint);
    double.TryParse(current1, out intcurrent1);
    double.TryParse(current2, out intcurrent2);
    if (zedGraphACC.GraphPane.CurveList.Count <= 0)
        return;

    //get the first CurveItem in the graph
    LineItem curve = zedGraphACC.GraphPane.CurveList[0]
as LineItem;
    LineItem curve1 = zedGraphACC.GraphPane.CurveList[1]
as LineItem;
    LineItem curve2 = zedGraphACC.GraphPane.CurveList[2]
as LineItem;
    if (curve == null) return;
    if (curve1 == null) return;
    if (curve2 == null) return;

    //get the PointPairList
    IPointListEdit List = curve.Points as IPointListEdit;
    IPointListEdit List1 = curve1.Points as
IPointListEdit;
    IPointListEdit List2 = curve2.Points as
IPointListEdit;
}

```

```

        //if this is null, it means the reference at
curve.Points does not
        //support IPointListEdit, so we wont be able to
modify it
        if (List == null) return;
        if (List1 == null) return;
        if (List2 == null) return;

        //time is measured in second
        //double time = (Environment.TickCount -
TickStartGyro) / 100.0;
        timeAcc = (Environment.TickCount - TickStartGYRO) /
100.0;

        //3 seconds per cycle
        List.Add(timeAcc, intsetpoint);
        List1.Add(timeAcc, intcurrent1);
        List2.Add(timeAcc, intcurrent2);

        //keep the X scale at a rolling 30 second interval,
with one
        //major step between the max X value and the end of
the axis

        Scale xScale = zedGraphACC.GraphPane.XAxis.Scale;
        if (timeAcc > xScale.Max - xScale.MajorStep)
        {
            xScale.Max = timeAcc + xScale.MajorStep;
            xScale.Min = xScale.Max - 100.0;
        }

        //make sure the Y axis is rescaled to accommodate
actual data
        zedGraphACC.AxisChange();
        //force a redraw
        zedGraphACC.Invalidate();
    }

    private void DrawVibrasiGyro(string setpoint, string
current1, string current2)
{
    //Make sure that the curvelist has at least one curve
    double intsetpoint;
    double intcurrent1;
    double intcurrent2;
    double.TryParse(setpoint, out intsetpoint);
    double.TryParse(current1, out intcurrent1);
    double.TryParse(current2, out intcurrent2);
}

```

```

        if (zedGraphGYRO.GraphPane.CurveList.Count <= 0)
            return;

        //get the first CurveItem in the graph
        LineItem curve = zedGraphGYRO.GraphPane.CurveList[0]
as LineItem;
        LineItem curve1 = zedGraphGYRO.GraphPane.CurveList[1]
as LineItem;
        LineItem curve2 = zedGraphGYRO.GraphPane.CurveList[2]
as LineItem;
        if (curve == null) return;
        if (curve1 == null) return;
        if (curve2 == null) return;

        //get the PointPairList
        IPointListEdit List = curve.Points as IPointListEdit;
        IPointListEdit List1 = curve1.Points as
IPointListEdit;
        IPointListEdit List2 = curve2.Points as
IPointListEdit;

        //if this is null, it means the reference at
curve.Points does not
        //support IPointListEdit, so we wont be able to
modify it
        if (List == null) return;
        if (List1 == null) return;
        if (List2 == null) return;
        //time is measured in second
        //double time = (Environment.TickCount -
TickStartGyro) / 100.0;
        timeGyro = (Environment.TickCount - TickStartGYRO) /
100.0;

        //3 seconds per cycle
        List.Add(timeGyro, intsetpoint);
        List1.Add(timeGyro, intcurrent1);
        List2.Add(timeGyro, intcurrent2);

        //keep the X scale at a rolling 30 second interval,
with one
        //major step between the max X value and the end of
the axis

        Scale xScale = zedGraphGYRO.GraphPane.XAxis.Scale;
        if (timeGyro > xScale.Max - xScale.MajorStep)
{
            xScale.Max = timeGyro + xScale.MajorStep;
            xScale.Min = xScale.Max - 50.0;
}

```

```

        }

        //make sure the Y axis is rescaled to accommodate
actual data
        zedGraphGYRO.AxisChange();
        //force a redraw
        zedGraphGYRO.Invalidate();
    }

#endifregion

public static string GetStringBetweenCharacters(string
input, string charFrom, string charTo)
{
    int chrFrom = charFrom.Length;
    int chrTo = charTo.Length;
    int posFrom = input.IndexOf(charFrom);
    if (posFrom != -1) //if found char
    {
        int posto = input.IndexOf(charTo, posFrom + 1);
        if (posto != -1) //if found char
        {
            try { return input.Substring(posFrom +
chrFrom, posto - posFrom - chrTo); }
            catch { }
        }
    }
    return string.Empty;
}
public DemoWinow()
{
    InitializeComponent();
}

private void btnStop_Click(object sender, EventArgs e)
{
    timer1.Stop();
    timer2.Stop();
}

private void btnClose_Click(object sender, EventArgs e)
{
    this.Close();
}

private void timer2_Tick(object sender, EventArgs e)
{
}

```

```

        if(checkBox1.Checked==true)
        {
            akses_data.Open();
            try
            {
                OleDbCommand ghozia = new OleDbCommand();
                ghozia.Connection = akses_data;
                ghozia.CommandText = string.Format("insert
                into Table1(ACCX, ACCY, ACCZ, GYROX, GYROY,
                GYROZ, ROLL, PITCH, YAW) values
                ('{0}', '{1}', '{2}', '{3}', '{4}', '{5}', '{6}', '{7}',
                '{8}')", lbAccX.Text, lbAccY.Text,
                lbAccZ.Text, lbGyroX.Text, lbGyroY.Text,
                lbGyroZ.Text, lbl_Roll.Text, lbl_Pitch.Text,
                lbl_Yaw.Text);
                ghozia.ExecuteNonQuery();
            }
            catch (OleDbException rizky)
            {
                MessageBox.Show(rizky.Message);
            }
            finally { akses_data.Close(); }
        }

        private void DemoWinow_Load(object sender, EventArgs e)
        {
            // TODO: This line of code loads data into the
            'database4DataSet.Table1' table. You can move, or remove it, as
            needed.

            this.table1TableAdapter.Fill(this.database4DataSet.Table1);
            table.Columns.Add("ACCX",typeof(int));
            table.Columns.Add("ACCY", typeof(int));
            table.Columns.Add("ACCZ", typeof(int));
            table.Columns.Add("GYROX", typeof(int));
            table.Columns.Add("GYROY", typeof(int));
            table.Columns.Add("GYROZ", typeof(int));
            table.Columns.Add("ROLL", typeof(int));
            table.Columns.Add("PITCH", typeof(int));
            table.Columns.Add("YAW", typeof(int));

            //BAUDRATE
            cbBaudr.Items.Add("9600");
            cbBaudr.Items.Add("38400");
            cbBaudr.Items.Add("57600");
            cbBaudr.Items.Add("115200");
        }
    }
}

```

```

#region //1. ZEDGRAPH ACCELEROMETER GYROSCOPE

    //ZedGraph Accelero
    zedGraphACC.GraphPane.Chart.Fill = new
Fill(Color.Gray);
    zedGraphACC.GraphPane.Fill = new Fill(Color.Gray);
    zedGraphACC.GraphPane.YAxis.MajorGrid.Visible =
true;

    //Nama Grafik sumbu x dan y
    GraphPane mypaneACC = zedGraphACC.GraphPane;
    mypaneACC.Title.Text = "ACCELEROMETER";
    mypaneACC.YAxis.Title.Text = "VIBRASI";
    mypaneACC.XAxis.Title.Text = "Time, 10 ms";

    //Skala sumbu x dan y
    mypaneACC.XAxis.Scale.MajorStep = 10;
    mypaneACC.YAxis.Scale.Mag = 0;
    mypaneACC.XAxis.Scale.Max = 200;

    mypaneACC.YAxis.Scale.MajorStep = 20;
    mypaneACC.XAxis.Scale.Mag = 0;
    mypaneACC.YAxis.Scale.Max = 200;

    mypaneACC.YAxis.Scale.MajorStep = 5;
    mypaneACC.XAxis.Scale.Mag = 0;
    mypaneACC.YAxis.Scale.Max = 100;

    //Zedgraph Gyroscope
    zedGraphGYRO.GraphPane.Chart.Fill = new
Fill(Color.Gray);
    zedGraphGYRO.GraphPane.Fill = new Fill(Color.Gray);
    zedGraphGYRO.GraphPane.YAxis.MajorGrid.Visible =
true;

    //Nama Grafik sumbu x dan y
    GraphPane mypaneGYRO = zedGraphGYRO.GraphPane;
    mypaneGYRO.Title.Text = "GYROSCOPE";
    mypaneGYRO.YAxis.Title.Text = "VIBRASI";
    mypaneGYRO.XAxis.Title.Text = "Time, 10 ms";

    //Skala sumbu x dan y
    mypaneGYRO.XAxis.Scale.MajorStep = 10;
    mypaneGYRO.YAxis.Scale.Mag = 0;
    mypaneGYRO.XAxis.Scale.Max = 200;

    mypaneGYRO.YAxis.Scale.MajorStep = 20;
    mypaneGYRO.XAxis.Scale.Mag = 0;

```

```

mypaneGYRO.YAxis.Scale.Max = 200;

mypaneGYRO.YAxis.Scale.MajorStep = 5;
mypaneGYRO.XAxis.Scale.Mag = 0;
mypaneGYRO.YAxis.Scale.Max = 100;
#endregion
}

private void cbPort_DropDown(object sender, EventArgs e)
{
    cbPort.Items.Clear();

cbPort.Items.AddRange(System.IO.Ports.SerialPort.GetPortNames());
}

private void btStart_Click(object sender, EventArgs e)
{
    serialPort1.Write("a");
    timer1.Start();
    btnStop.Enabled = true;
    timer2.Enabled = true;
    // timer2.Start();

    graphVibrasiACC();
    graphVibrasiGYRO();
}

private void btnCon_Click(object sender, EventArgs e)
{
    if (btnCon.Text == "CONNECT") ;
    {
        if (serialPort1.IsOpen == false)
        {
            serialPort1.PortName = cbPort.Text;
            serialPort1.BaudRate =
int.Parse(cbBaudr.Text);
            try
            {
                serialPort1.Open();
                btnCon.Text = "DISCONNECT";

            }
            catch (Exception Ghozia)
//memunculkan kalok ada error
            {
                MessageBox.Show("Serial Error" + Ghozia);
            }
        }
    else
}

```

```

        {
            serialPort1.Close();
            btnCon.Text = "CONNECT";
        }
    }

    private void timer1_Tick(object sender, EventArgs e)
    {
        //Menerima data dari arduino dan ngubah label di GUI
        DataMasuk = serialPort1.ReadExisting();
        if (GetStringBetweenCharacters(DataMasuk, "<B4",
"B4>") != "")
            AccX = GetStringBetweenCharacters(DataMasuk,
"<B4", "B4>");
            lbAccX.Text = AccX;
            if (GetStringBetweenCharacters(DataMasuk, "<B5",
"B5>") != "")
                AccY = GetStringBetweenCharacters(DataMasuk,
"<B5", "B5>");
                lbAccY.Text = AccY;
                if (GetStringBetweenCharacters(DataMasuk, "<B6",
"B6>") != "")
                    AccZ = GetStringBetweenCharacters(DataMasuk,
"<B6", "B6>");
                    lbAccZ.Text = AccZ;
                    if (GetStringBetweenCharacters(DataMasuk, "<B7",
"B7>") != "")
                        GyroX = GetStringBetweenCharacters(DataMasuk,
"<B7", "B7>");
                        lbGyroX.Text = GyroX;
                        if (GetStringBetweenCharacters(DataMasuk, "<B8",
"B8>") != "")
                            GyroY = GetStringBetweenCharacters(DataMasuk,
"<B8", "B8>");
                            lbGyroY.Text = GyroY;
                            if (GetStringBetweenCharacters(DataMasuk, "<B9",
"B9>") != "")
                                GyroZ = GetStringBetweenCharacters(DataMasuk,
"<B9", "B9>");
                                lbGyroZ.Text = GyroZ;

                                if (GetStringBetweenCharacters(DataMasuk, "<B0",
"B0>") != "")
                                    lbl_Pitch.Text =
GetStringBetweenCharacters(DataMasuk, "<B0", "B0>");
                                    if (GetStringBetweenCharacters(DataMasuk, "<B1",
"B1>") != "")

```

```

        lbl_Roll.Text =
GetStringBetweenCharacters(DataMasuk, "<B1", "B1");
        if (GetStringBetweenCharacters(DataMasuk, "<B2",
"B2") != ""))
            lbl_Rollg.Text =
GetStringBetweenCharacters(DataMasuk, "<B2", "B2");
            if (GetStringBetweenCharacters(DataMasuk, "<B3",
"B3") != ""))
                lbl_Yaw.Text =
GetStringBetweenCharacters(DataMasuk, "<B3", "B3");

        Pitch = Convert.ToInt32(lbl_Pitch.Text); //ax
        Roll = Convert.ToInt32(lbl_Roll.Text); //ay
        Rollg = Convert.ToInt32(lbl_Rollg.Text); //gx
        Yaw = Convert.ToInt32(lbl_Yaw.Text); //gy

        //menjalankan void ==> buat garis di graph
        DrawVibrasiAcc(AccX, AccY, AccZ);
        DrawVibrasiGyro(GyroX, GyroY, GyroZ);

horizonInstrumentControl1.SetAttitudeIndicatorParameters(Pitch,Ro
ll);
horizonInstrumentControl1.SetAttitudeIndicatorParameters(Pitch,
Roll);

turnCoordinatorInstrumentControl1.SetTurnCoordinatorParameters((R
ollg / 10), Yaw);
turnCoordinatorInstrumentControl1.SetTurnCoordinatorParameters((R
ollg / 10), Yaw);

        table.Rows.Add(ACCX, ACCY, ACCZ, GYROX, GYROY,GYROZ,
ROLL, PITCH, YAW);
        dataGridView1.DataSource = table;
        if (dataGridView1.Rows.Count >= 20)
        {
            table.Rows.Clear();
            dataGridView1.DataSource = table;
        }

    }
}
}

```



# MODUL

## NAVIGASI DRONE GUNUNG BERAPI DENGAN ACCELEROMETER DAN GYROSCOPE

DISUSUN OLEH:

RIZKY GHOZIA ULHAQ  
15518244007

DOSEN PEMBIMBING:  
HERLAMBANG SIGIT PURNOMO, M.SC.



## **MODUL**

### **NAVIGASI DRONE DENGAN ACCELEROMETER DAN GYROSCOPE**

## KATA PENGANTAR

Puji syukur kami ucapkan ke hadirat Allah SWT yang telah melimpahkan karunia, rahmat, serta petunjuk-Nya, sehingga penyusun dapat menyelesaikan modul praktikum mengenai navigasi drone dengan accelerometer dan gyroscope dari Pengembangan Drone Pemantau Gunung Berapi Sebagai Media Pembelajaran Sensor Accelerometer Dan Gyroscope Pada Matakuliah Kendali Dan Akuisisi Data. Modul ini disusun agar dapat membantu pendidik dan peserta didik dalam kegiatan pembelajaran praktik Kendali dan Akuisisi data.

Modul praktikum ini memuat materi pembelajaran praktik kendali dan akuisisi data untuk Drone Pemantau Gunung Berapi Sebagai Media Pembelajaran Sensor Accelerometer Dan Gyroscope dibahas dalam modul ini. Komponen tersebut adalah komponen-komponen utama yang ada pada system seperti STM32, MPU6050, dan GUI. Materi yang dibahas termasuk pembahasan mendasar tentang komponen maupun cara memrogram dan pengendaliannya.

Tentu tidak ada hal yang sempurna di dunia ini, semoga modul praktikum ini bermanfaat bagi dunia pendidikan untuk meningkatkan kompetensi peserta didik khususnya di lingkungan Universitas Negeri Yogyakarta. Modul praktikum ini masih banyak kekurangan, saya mengharapkan saran dan kritik dari pengguna modul praktikum ini untuk perbaikan di masa yang akan datang. Penyusun mengucapkan terima kasih kepada semua pihak yang turut membantu dalam penyusunan modul praktikum ini.

Penyusun

## DAFTAR ISI

<b>KATA PENGANTAR.....</b>	<b>2</b>
<b>BAB I DRONE .....</b>	<b>4</b>
A. DRONE .....	4
B. JENIS .....	5
C. KOMPONEN.....	12
<b>BAB II GRAPHICAL USER INTERFACE (GUI) .....</b>	<b>19</b>
A. GUI .....	19
<b>BAB III SISTEM MIKROKONTROLER DAN PEMROGRAMAN .....</b>	<b>20</b>
A. STM32F103C8.....	20
D. PROTOKOL KOMUNIKASI ARDUINO .....	23
E. PEMROGRAMAN DASAR ARDUINO .....	27
<b>BAB IV MPU6050 .....</b>	<b>33</b>
A. MODUL MPU6050 .....	33
B. ACCELEROMETER .....	34
<b>BAB V NAVIGASI DRONE PEMANTAU GUNUNG BERAPI .....</b>	<b>37</b>
A. GERAKAN UAV.....	38
B. DASAR INSTRUMEN PENERBANGAN.....	41
<b>DAFTAR PUSTAKA .....</b>	<b>45</b>

## **BAB I**

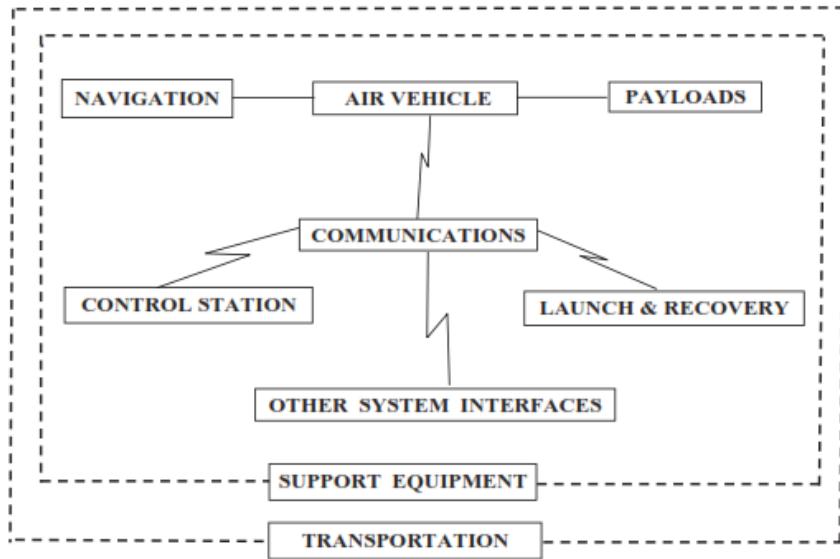
### **DRONE**

#### **A. DRONE**

*Unmanned Aerial Vehicle* (UAV) atau *Unmanned Aerial System* (UAS) atau yang lebih terkenal dimasyarakat dengan sebutan drone merupakan pesawat yang terbang tanpa adanya pilot didalamnya. Pesawat ini dikendalikan dari jarak jauh dan dapat beroperasi secara otomatis dan atau menggunakan kendali dari pilot. Drone sendiri memiliki arti lebah betina karena awal mulanya UAV memiliki suara propeller yang berisik seperti lebah. Sebenarnya drone memiliki arti teknologi yang dimana wahana beroperasi secara otomatis berdasarkan data input, contohnya seperti mengikuti jalur yang telah dimasukan dalam controller. Jadi drone tidak hanya mencangkup pesawat terbang saja tetapi juga wahana lain seperti kapal, mobil, dan sebagainya. (Caesar Wiratama, 2016).

Pada dasarnya pilot/awak dalam UAV dipindah dan digantikan dengan system computer dan radio-link. Kenyataannya dalam pembuatannya lebih kompleks dan UAV harus didesain sedemikian rupa dari awal tanpa adanya awak/crew, akomodasi, dan sebagainya didalamnya. Kerangka pesawat hanyalah salah satu bagian penting dari sistem total. Keuntungan dari keseluruhan sistem dilihat dari desain sejak awal sebagai sistem yang lengkap seperti gambar, yang secara singkat terdiri dari:

1. *Ground Control System* (GCS) yang menampung operator system dan tampilan yang menjembatani antara operator dengan system.
2. UAV bermuatan yang memiliki beberapa tipe
3. System komunikasi antara GCS dengan UAV menggunakan transmitter dan receiver
4. Peralatan pendukung yang mungkin termasuk alat perawatan dan transportasi.



Gambar 1. Struktur fungsi UAV system

Pesawat terbang tanpa awak dapat digunakan untuk menggantikan tugas manusia dalam melaksanakan tugas/misi yang berbahaya, kasar, kotor maupun membosankan. Pengaplikasian seperti pemetaan hutan atau lahan, monitoring area yang luas, pencarian korban bencana, dan sebagainya. Pada mulanya UAV memang digunakan pertama kali oleh militer namun seiring berkembangnya jaman penggunaannya semakin luas.

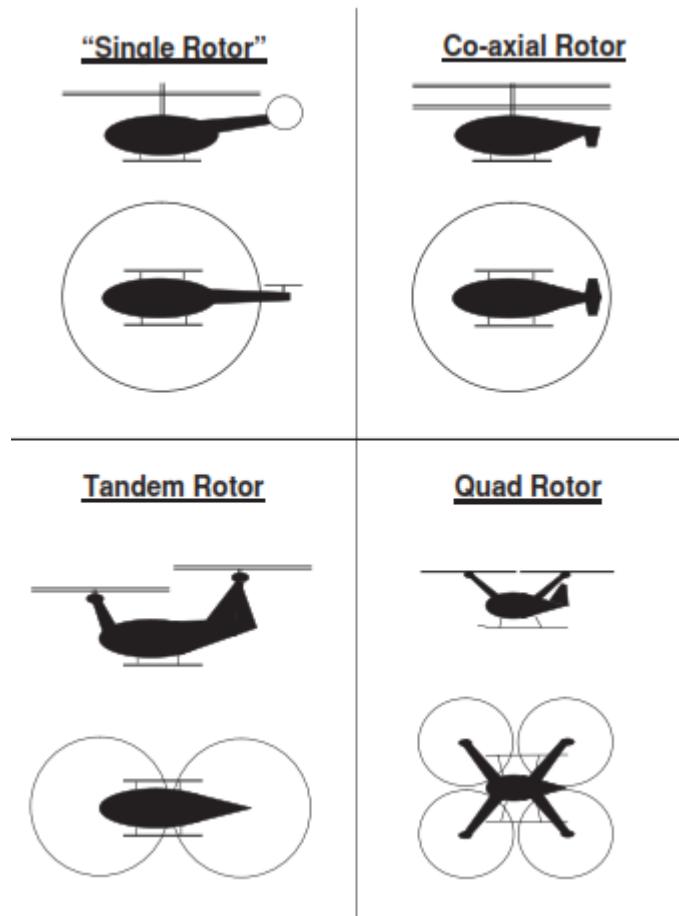
## B. JENIS

Jumlah jenis pesawat UAV sama beragamnya dengan pesawat komersial yang memiliki kru didalamnya, selain itu resiko komersial menyebabkan kurangnya dukungan unruk produsen UAV karena masih dianggap tidak lazim. Hal ini juga disebabkan oleh bentuk dan ukuran dari pesawat tanpa awak yang relative kecil. Lebih mudah untuk mengelompokan jenis pesawat tanpa awak menjadi dua jenis secara garis besar berdasarkan pada bentuk dan penggeraknya yaitu Vertical Take Off and Landing (VTOL) dan Horizontal Take Off and Landing (HTOL).

### 1. Vertical Take Off and Landing (VTOL)

Seperti namanya VTOL merupakan jenis pesawat tanpa awak yang dapat terbang secara vertical ketika landing dan juga take off.

VTOL didalam masyarakat sering dikenal dengan nama drone. Terdapat beberapa jenis VTOL yang digolongkan berdasarkan pada jumlah motor atau penggerak yang digunakan.



Gambar 2. Jenis-jenis VTOL

a. Single main rotor

Pada VTOL jenis ini menggunakan satu motor sebagai penggerak utama. Torsi dari motor bergerak membalik badan dari pesawat sehingga perlu dinetralkan dengan rotor yang ada pada ekor untuk mengontrol arah. Penggunaan VTOL jenis ini sangat asimetris sehingga menambah sulit dalam control dan kompleksitas algoritma dari system control penerbangannya. Selain itu rotor yang terpasang diekor relative rentan dan rapuh terhadap seraangan

benda yang terdapat didarat, terutama yang berukuran lebih kecil dari mesin.

b. Co-axial rotor

Konfigurasi ini, terutama dari pabrikan Rusia Mil., terbatas digunakan untuk awak kapal aplikasi. Jenis ini tidak lebih populer secara umum karena tingginya yang lebih tinggi dibandingkan dengan jenis yang lain. Hal ini dapat menyebabkan kerugian dalam pemeliharaan dan penyimpanannya. Tetapi untuk pengaplikasian pada UAV, maka jauh lebih ringan dan lebih kecil, sehingga bukan hal yang merugikan lagi. Keuntungan dari jenis ini memiliki bentuk yang simetri aerodinamis hampir sempurna, kekompakan tanpa rotor ekor yang rentan, efisiensi daya dan keserbagunaan menyediakan desain bodi alternatif untuk kegunaan yang berbeda, tetapi masing-masing bagian menggunakan sub-sistem unit daya, transmisi, dan kontrol yang sama. Karenanya algoritme sistem kontrol penerbangan otomatis (AFCS) tidak lebih kompleks dari pada yang tipikal pesawat HTOL. Selain itu, sebagian besar karena simetri, responsnya terhadap turbulensi udara yang terendah dari semua konfigurasi helicopter dan dapat bernilai nol di sebagian besar mode.

c. Tandem rotor

Ada efek skala yang kuat pada ukuran rotor helikopter sehingga rasio massa rotor untuk mengangkat meningkatkuat dengan ukuran rotor yang lebih besar yang dibutuhkan oleh pesawat yang lebih berat. Oleh karena itu lebih efisien untuk disesuaikan dua rotor yang lebih kecil dari satu besar untuk pesawat di atas AUM tertentu. Poin 'cross-over' naik sebagai teknologi, terutama bahan, meningkat. Ini telah meningkat dari nilai sekitar 10.000 kg di 1960-an menjadi sekitar 15.000 kg hari ini. Sampai VTOL UAV dari AUM ini dipertimbangkan konfigurasi rotor tandem tidak sesuai untuk UAV, meskipun konfigurasi lebih

simetris dalam kontrol daripada 'rotor tunggal' dan lebih efisien daya. Volume payload yang kecil dari pesawat AUM rendah tidak memerlukan badan pesawat yang panjang sehingga rotor harus dipasang di tiang diperpanjang. Itu tidak efisien secara struktural. Untuk alasan ini, penulis tidak mengetahui adanya contoh UAV tandem rotor saat ini.

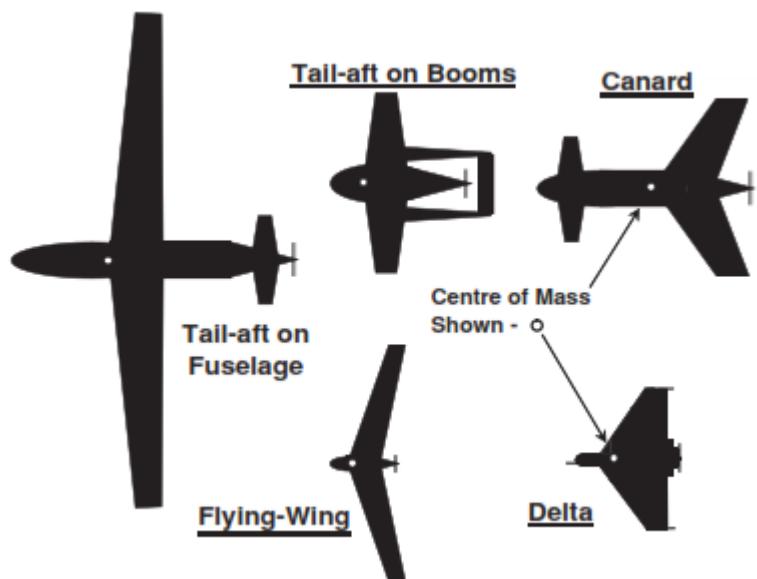
d. Quad rotor

Dibandingkan dari semua konfigurasi yang dibahas sebelumnya, jenis ini menggunakan sistem kontrol rotor-head yang menerapkan keduanya perubahan pitch siklik dan kolektif pada baling-baling rotor sebagai alat kontrol pesawat. Tujuan desain dari bentuk quad-rotor adalah untuk menghilangkan komplikasi ini, dan juga untuk menghilangkan kebutuhan mekanik sistem transmisi. Idenya adalah untuk memiliki semua bilah rotor tetap di pitch dan untuk mencapai perubahan dorong pada setiap rotor dengan mengubah kecepatan putarannya. Setiap rotor digerakkan secara individual oleh motor listrik dipasang di kepala rotor. Jadi, misalnya, untuk pesawat bergerak maju kecepatan rotasi dua rotor belakang akan dinaikkan untuk mengarahkan pesawat ke bawah dan mengarahkan dorong yang dihasilkan vektor ke depan. Pada saat yang sama total dorong harus ditingkatkan untuk mencegah kehilangan ketinggian dan, sekali ditetapkan dalam penerbangan maju, kecepatan rotor harus diselaraskan kembali.

2. Horizontal Take Off and Landing (HTOL).

Pesawat jenis ini merupakan pesawat dengan sayap tetap atau sering disebut dengan fixed wing. Setelah bertahun-tahun pengembangan dalam aplikasi kru, ini telah berkurang menjadi tiga jenis mendasar, sebagian besar ditentukan dengan cara mereka mengangkat / keseimbangan massa dan dengan stabilitas dan kontrol. Mereka adalah plane tailplane tipe aft, tailplane forward atau tailless, ditampilkan dalam bentuk garis besar pada gambar 3. Semua konfigurasi, dengan

pengecualian hanya Phoenix yang diketahui, memiliki powerplant di bagian belakang badan pesawat. Ini untuk membebaskan bagian depan pesawat untuk pemasangan muatan untuk memiliki pandangan ke depan yang tidak terhalang. Dari sudut pandang aerodinamis, jika baling-baling digunakan, kecepatan udara terinduksi di depan rear-mounted baling-baling tidak menambah gesekan gesekan badan pesawat seperti halnya slipstream dari baling-baling traktor yang dipasang di depan.



Gambar 3. Jenis-jenis HTOL

a. Main wing forward

Ini diterima sebagai pengaturan konvensional dan sejauh ini merupakan yang paling umum. Pusat pesawat massa maju dari pusat sayap lift dan ini diimbangi dengan down-load pada tailplane, sehingga memberikan kecepatan aerodinamis dan stabilitas sikap di bidang horizontal. Sebuah jaring vertikal menyediakan Stabilitas weathercock di menguap dengan sayap dihedral memberikan stabilitas dalam gulungan. Konfigurasi ini didirikan adalah datum yang dibandingkan dengan konfigurasi lainnya. Perbedaan dalam kategori harus dibedakan dari satu sama lain hanya dengan

bagaimana permukaan ekor dilakukan - yaitu boom ekor tunggal atau kembar buntut ekor dan dengan jumlah mesin yang digunakan. HALE dan MALE saat ini, mis. UAV jarak jauh, semuanya memiliki permukaan ekor yang terbawa di belakang badan pesawat. Ini mungkin karena volume pesawat yang panjang diperlukan untuk membawa sejumlah besar peralatan dan beban bahan bakar yang dibutuhkan pada jenis operasinya. Indikasi volume yang diminta ditunjukkan pada Gambar 6.8 (Bab 6) oleh tampilan artis yang meledak dari UAV jarak jauh yang khas. Pengaturan twin-boom populer untuk UAV menengah dan jarak dekat karena ini memungkinkan mesin harus dipasang sebagai sistem pendorong hanya buritan sayap, sekali lagi membebaskan pesawat depan untuk muatan instalasi. Ini juga memberikan tingkat perlindungan untuk dan dari mesin dan baling-baling. Ada juga beberapa keuntungan aerodinamis yang bisa diperoleh dengan konfigurasi ini. Baling-baling pendorong dan mesin erat di belakang pusat pesawat massa mengurangi inersia pesawat di lapangan dan mengoleng. Kedekatan relatif baling - baling dengan empennage meningkatkan daya kontrol melalui slipstream melewati elevator dan kemudi dan, dengan inersia yang lebih rendah, memberikan pesawat yang lebih banyak responsif terhadap kontrol pitch dan yaw. Kualitas-kualitas ini menjelaskan popularitas konfigurasi ini. Untuk contoh lihat Hunter dan Seeker UAV pada Gambar 4.11.

b. Canard

Konfigurasi canard memiliki stabilizer horizontal, atau foreplane, yang dipasang di depan sayap. Itu pusat massa pesawat terbang juga maju dari sayap dan keseimbangan dicapai dengan garis depan menghasilkan daya angkat positif. Stabilitas aerodinamis pada bidang horizontal adalah hasilnya. Keuntungan dari sistem canard adalah karena kedua pesawat menghasilkan daya angkat positif, aerodinamis lebih efisien daripada konfigurasi ekor

belakang. Ini juga memiliki keuntungan bahwa, karena diatur pada sudut pandang kejadian yang lebih tinggi daripada sayap utama, sayap depan berdiri di depan sayap utama. Ini hanya menghasilkan sedikit kehilangan daya angkat dan lembut gerakan pitching hidung ke bawah menuju pemulihan dengan sedikit kehilangan tinggi dibandingkan dengan yang terjadi setelahnya kandang konfigurasi ekor belakang.

c. Flying wing

Ini termasuk pesawat bersayap delta yang, seperti di atas, memiliki 'ekor' yang efektif. Sayap memiliki 'Sweep-back' dan tip aerofoils memiliki insiden yang sangat berkurang dibandingkan dengan aerofoils dari sayap bagian dalam. Ini memastikan bahwa, ketika hidung pesawat naik, pusat pengangkatan sayap bergerak ke belakang, sehingga mengembalikan pesawat ke sikap semula. Pesawat-pesawat ini menderita dengan cara yang mirip dengan canard karena memiliki lengan ekor yang berkurang efektif pada keduanya pitch dan yaw kapak, meskipun sapuan sayap ke belakang memang menambah stabilitas arah. Argumen yang umumnya ditawarkan untuk konfigurasi ini adalah bahwa menghilangkan horizontal stabilizer menyimpan profil drag permukaan itu. Lawan akan menunjuk ke distribusi angkat yang lebih buruk dari sayap terbang yang dapat mengakibatkan pengangkatan negatif di bagian ujung dan menghasilkan hambatan yang diinduksi tinggi.

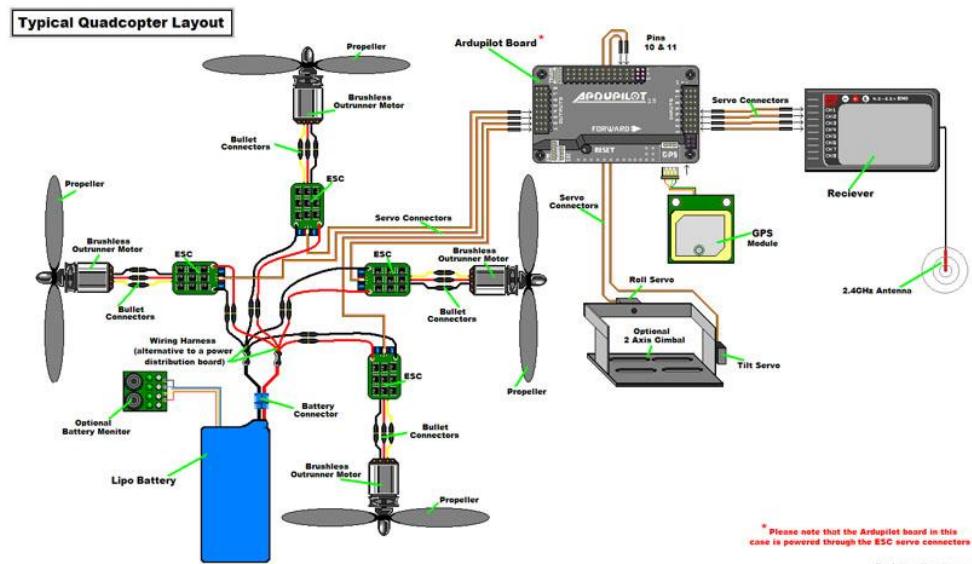
d. Delta wing

Konfigurasi sayap-delta, seperti pada Observer UAV memberikan badan pesawat yang kokoh untuk pendaratan skid atau parasut, tanpa ekor lebih ringan dan lebih rentan. Ini memiliki respons hembusan yang lebih rendah, karena rasio aspeknya lebih rendah, daripada pesawat HTOL lainnya. Namun, itu berbagi dengan kritik sayap terbang distribusi lift yang buruk,

menghasilkan tarikan yang diinduksi lebih tinggi diperburuk oleh pemuatan bentang yang lebih tinggi. Sistem propulsive yang paling umum digunakan adalah, seperti pada canard dan sayap terbang, oleh engine yang dipasang di belakang dalam bentuk turbo-jet atau baling-baling.

### C. KOMPONEN

Pada UAV terdapat komponen-komponen yang diperlukan akan wahana dapat beroperasi. Komponen yang digunakan yaitu motor brushless, Electronic Speed Control (ESC), Transmitter Receiver, GPS, sensor Accelerometer Gyroscope, Ardupilot, dan radio control.



Gambar 4. Skema diagram Quadcopter

#### 1. Motor brushless

Motor Brushless direct current (BLDC) adalah salah satu dari sekian jenis-jenis motor yang umum digunakan. Sesuai dengan namanya, BLDC motor tidak menggunakan sikat atau brush untuk kontak pergantian magnet (komunikasi) tetapi dilakukan secara komutasi elektronis. Penggunaan motor BLDC memiliki banyak keuntungan dibandingkan motor DC dan motor induksi lainnya. Keuntungannya antara lain:

- a. Kecepatan yang lebih baik untuk melawan karakteristik tenaga putaran
- b. Tanggapan Dinamis tinggi
- c. Efisiensi tinggi
- d. Tahan Lama
- e. Nyaris tanpa suara
- f. Speed range yang lebih luas

Sebagai tambahan perbandingan tenaga putaran lebih besar dibandingkan dengan ukuran motornya. Dengan ukuran motor yang relatif kecil dapat menghasilkan torsi cukup besar. Jadi penggunaan motor BLDC sangat bermanfaat bila digunakan pada aplikasi kritis terhadap beban dan tempat pemasangan.

Motor BLDC adalah suatu jenis motor sinkron yang artinya medan magnet yang dihasilkan oleh strator dan medan magnet yang dihasilkan oleh rotor berputar di frekuensi yang sama. Motor BLDC tidak mengalami slip, tidak seperti yang terjadi pada motor induksi biasa. Motor BLDC adalah motor AC tiga fasa yang ter-supply pada setiap saat sementara satu fasa lainnya tak tersupply. Fenomena ini mengakibatkan motor ini seperti motor DC, karena arus yang mengalir pada akumparan staror mirip dengan motor DC meskipun motor ini sebenarnya tercatu dengan arus tiga fasa.

Spesifikasi dari motor BLDC RCTimer1000Kv yang ada pada gambar 10 adalah sebagai berikut :

- a. Battery: 2~4 Cell /7.4~14.8V
- b. RPM: 1000kv
- c. Max current: 38A
- d. No load current: 2.4A
- e. Max power: 665W
- f. Internal resistance: 0.031 ohm
- g. Weight: 130g (including connectors)
- h. Diameter of shaft: 5mm

- i. Dimensions: 35 x 42m
- j. Prop size: 7.4V/11x7 14.8V/10x5
- k. Max thrust: 1260g



Gambar 5. Motor Brushless

## 2. Electronic Speed Control (ESC)

ESC adalah salah satu komponen elektronik yang digunakan untuk mengatur kecepatan motor.



Gambar 6. ESC Turnigy UBEC

Berikut adalah spesifikasi dari turnigy UBEC.

- a. Output : 5v/3A atau 6v/3A (dapat dipilih melalui jumper)
- b. Noise : <50mVp-p (@ 2A/12v)
- c. Input : 5.5V-23v (2-5S pak Lipo, 5-15cells NiMH)
- d. Ukuran: 41.6x16.6x7.0mm
- e. Berat : 7.5g

## 3. GPS



Gambar 7. GPS Module

Global Positioning System (GPS) adalah sistem radio navigasi dan penentuan posisi menggunakan satelit, dengan nama resminya NAVSTAR GPS (Navigation Satellite Timing and Ranging Global Positioning System). GPS dikembangkan pertama kali oleh Departemen Pertahanan Amerika Serikat pada tahun 1978 dan secara resmi GPS dinyatakan operasional 3 pada tahun 1994. Pada awalnya GPS digunakan hanya untuk kepentingan militer Amerika Serikat, tetapi kemudian dapat dimanfaatkan juga untuk kepentingan sipil.

Ada beberapa karakteristik yang menjadikan GPS menarik untuk digunakan yaitu dapat digunakan setiap saat tanpa tergantung waktu dan cuaca, posisi yang dihasilkan mengacu pada suatu datum global, pengoperasian alat receiver relatif mudah, relatif tidak terpengaruh dengan kondisi topografis, dan ketelitian yang dihasilkan dapat diandalkan (Abidin,H.Z, 2007).

Kompas adalah alat navigasi untuk menentukan arah berupa sebuah panah penunjuk magnetis yang bebas menyelaraskan dirinya dengan medan magnet bumi secara akurat. Kompas memberikan rujukan arah tertentu, sehingga sangat membantu dalam bidang navigasi. Arah mata yang ditunjuknya adalah utara, selatan, timur, dan barat. Apabila digunakan bersama-sama dengan jam dan sekstan, maka kompas akan lebih akurat dalam menunjukkan arah. Alat ini membantu perkembangan perdagangan maritime dengan membuat perjalanan jauh

lebih aman dan efisien dibandingkan saat manusia masih berpedoman pada kedudukan bintang untuk menentukan arah.

Kompas dibedakan menjadi dua jenis, yaitu kompas analog dan kompas digital. Kompas analog adalah kompas yang biasa kita lihat dalam kehidupan sehari-hari. Misalnya saja kompas yang dipakai ketika acara pramuka. Sedangkan kompas digital merupakan kompas yang telah menggunakan proses digitalisasi. Dengan kata lain, cara kerja kompas ini menggunakan komputerisasi.

#### 4. Sensor Accelerometer Gyroscope

Accelerometer adalah alat yang digunakan untuk mengukur percepatan, mendekripsi dan mengukur getaran (vibrasi), dan mengukur percepatan akibat gravitasi. Sensor accelerometer mengukur percepatan dari 3 sumbu gerakan akibat gerakan benda yang melekat padanya.

Percepatan merupakan suatu keadaan berubahnya kecepatan terhadap waktu. Bertambahnya suatu kecepatan dalam suatu rentang waktu disebut juga percepatan (acceleration). Jika kecepatan semakin berkurang daripada kecepatan sebelumnya, disebut deceleration. Percepatan juga bergantung pada arah/orientasi karena merupakan penurunan kecepatan yang merupakan besaran vektor. Berubahnya arah pergerakan suatu benda akan menimbulkan percepatan pula.

*Gyroscope* adalah suatu alat berupa sensor gyro untuk menentukan orientasi gerak dengan bertumpu pada roda atau cakram yang berotasi dengan cepat pada sumbu. *Gyroscope* memiliki output yang peka terhadap kecepatan sudut dari arah sumbu x yang nantinya akan menjadi sudut phi (*Roll*), dari sumbu y nantinya menjadi sudut theta (*Pitch*), dan sumbu z nantinya menjadi sudut psi (*Yaw*).

#### 5. Ardupilot

Ardupilot adalah proyek autopilot berbasis platform open-source Arduino. Terdapat pilihan Ardupilot Mega (APM) untuk perangkat keras elektroniknya dan versi perangkat lunak untuk kendaraan yang berbeda. Perangkat keras APM akan tetap sama untuk semua tipe

pesawat dan dibedakan berdasarkan firmware yang ditanamkan sesuai dengan tipe pesawat yang digunakan menggunakan via Mission Planner. Untuk meningkatkan performa pesawat secara spesifik, dapat ditambahkan sensor lain seperti sensor airspeed atau magnetometer untuk multicopter. Fitur yang terdapat didalam APM antara lain:

- a. Firmware open source gratis yang mendukung pengendalian pesawat, multicopter (quad, hex, oct, dll) helicopter dan ground rover.
- b. Proses setup yang mudah. Apabila ingin merubah kode program di dalamnya dapat digunakan software Arduino.
- c. Dapat membuat script misi terbang yang diinginkan.
- d. Dapat mendukung ratusan waypoint secara 3D
- e. Telemetry dua arah dan perintah in-flight menggunakan protocol MAVLink
- f. Pilihan Ground Station gratis digunakan
- g. Dapat melakukan autonomous take off, landing dan beberapa perintah khusus seperti kontrol kamera dan video
- h. Mendukung simulasi “hardware in the loop” menggunakan Xplane dan Flight Gear.



Gambar 8. APM Ardupilot

## 6. Radio control

Untuk mengendalikan sebuah benda dari jarak yang cukup jauh dibutuhkan sebuah alat bantu. Turnigy 9X merupakan salah satu produk pengendalian secara jarak jauh. Radio control ini merupakan media

untuk melakukan kontrol jarak jauh guna mengendalikan hardware yang jaraknya jauh dari pengguna.

Radio control Turnigy 9X merupakan salah satu radio transmitter & Receiver yang bekerja pada frekuensi 2,4Ghz, pemilihan rekuensi ini dikarenakan frekuensi ini bebas digunakan. Radio receiver merupakan perangkat yang digunakan untuk menerima sinyal dari radio transmitter yang dikendalikan oleh pilot.

Radio transmitter mengirimkan sinyal-sinyal sesuai dengan posisi-posisi dari setiap kanal. Keluaran radio receiver dapat langsung digunakan untuk mengendalikan servo dan ESC karena sinyalnya merupakan sinyal standar dalam dunia RC. Keluaran RC receiver dihubungkan dengan perangkat modul mikrokontroler agar dapat digunakan untuk mengatur gerakan-gerakan *Quadcopter* pada saat terbang secara manual. Turnigy 9X dilengkapi dengan 8ch ppm/9ch pcm dan display 128X64 mm LCD. Bentuk dari Turnigy 9X ditunjukkan pada gambar 8.



Gambar 9. Radio Control

## **BAB II**

### **GRAPHICAL USER INTERFACE (GUI)**

#### **A. GUI**

*Graphic User Interface* (GUI) merupakan antarmuka atau tampilan elektronik berupa grafis suatu aplikasi yang dilihat dan sebagai media komunikasi dengan pengguna. GUI yang baik dapat mempermudah pengguna dengan memberikan tampilan yang konsisten dan kontrol yang intuitif seperti tombol, slider, menu, list boxes, dan sejenisnya. Sehingga pengguna dapat berkomunikasi dengan mesin dengan baik dan benar. Menurut Mauladi dkk (2016: 64) GUI merupakan tipe antarmuka yang digunakan pengguna untuk berinteraksi dengan sistem operasi melalui grafik, ikon, menu dan perangkat petunjuk seperti mouse atau track ball.

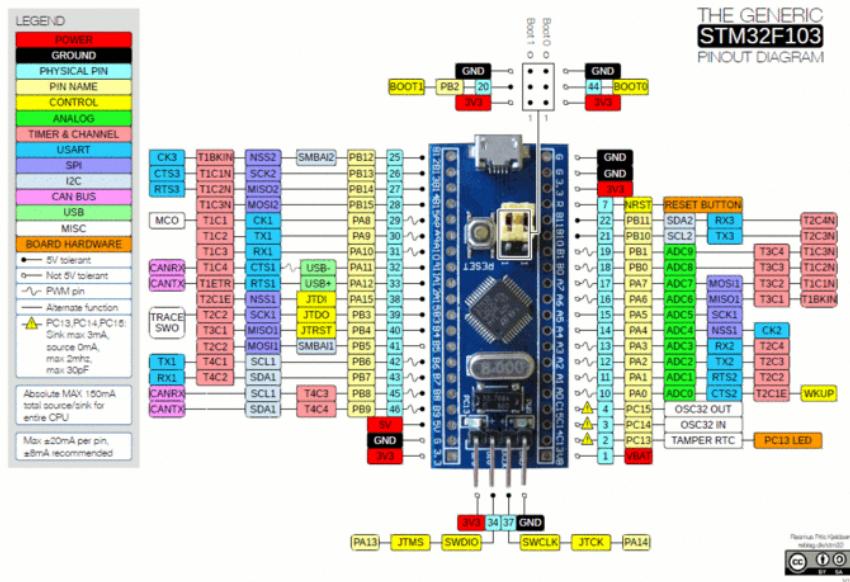
Pemrograman berbasis grafis akan menggunakan tool GUI editor yang telah memiliki *Integrated Development Environment* (IDE) sehingga pemrograman grafis relatif mudah. Pembuatan GUI dengan tool yang sudah IDE lebih mudah karena dapat dilihat secara visual dan melinatkan unsur desain pada pemrogramannya. Selain itu penggunaan berbagai komponen siap pakai yang bersifat visual maupun non-visual. Penggunaannya melainkan hanya *drag* dan *drop* lalu diprogram sedemikian rupa. Salah satu tool yang biasanya digunakan programmer untuk mengembangkan GUI yaitu Visual Studio.

## BAB III

### SISTEM MIKROKONTROLER DAN PEMROGRAMAN

#### A. STM32F103C8

STM32F103C8 atau Blue Pill merupakan papan mikrokontroller berbasis STM32 yang dikembangkan oleh STMicroelectronics. Papan ini memiliki pin GPIO sebanyak 32 pin diantaranya (12-bit ADC), I2C, USART, dan SPI. Perbedaan signifikan STM32 dengan arduino adalah pada frekuensi clock sebesar 72 Mhz, sedangkan arduino hanya 16 Mhz dengan kata lain STM32 4,5 kali lebih cepat dari arduino sehingga STM32 sangat cocok digunakan pada sensor maupun module yang membutuhkan pembacaan yang sangat cepat salah satunya adalah sensor MPU6050. STM32 juga dilengkapi dengan USB type B yang umumnya di temukan pada smartphone. Port USB ini bisa di manfaatkan sebagai *power supplay* bahkan bisa digunakan untuk memprogram dengan metode bootloader. Berikut adalah pin GPIO STM32:



Gambar 10. Pin STM32

Metode upload program pada STM32F103C8 memiliki 3 metode yaitu:

- Metode bootloader

Metode bootloader melibatkan aplikasi ketiga yaitu aplikasi Demostrasi GUI, dan hardware USB to ttl atau USB to serial.

- Metode serial

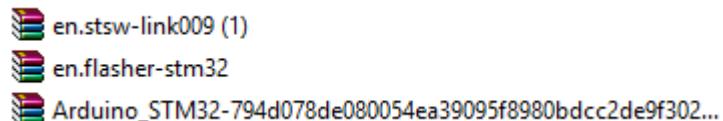
Seperti halnya bootloader, metode serial menggunakan hardware tambahan yaitu USB to serial tanpa melibatkan software tambahan.

c. Metode ST-Link

Metode ini paling umum digunakan dan sangat di rekomendasikan bagi para pemula STM32. Metode ST-Link menggunakan hardware tambahan yang bernama ST-Link v.2 yang di hubungkan ke mikrokontroller.

Penginstalan STM32F103C8 untuk digunakan menggunakan Arduino IDE dapat menggunakan salah satu cara diatas. Berikut salah satu cara menggunakan Bootloader sebagai berikut:

a. Download file installer berikut ini



b. Install satu per satu, untuk install Arduino\_STM32 lakukan langkah berikut:

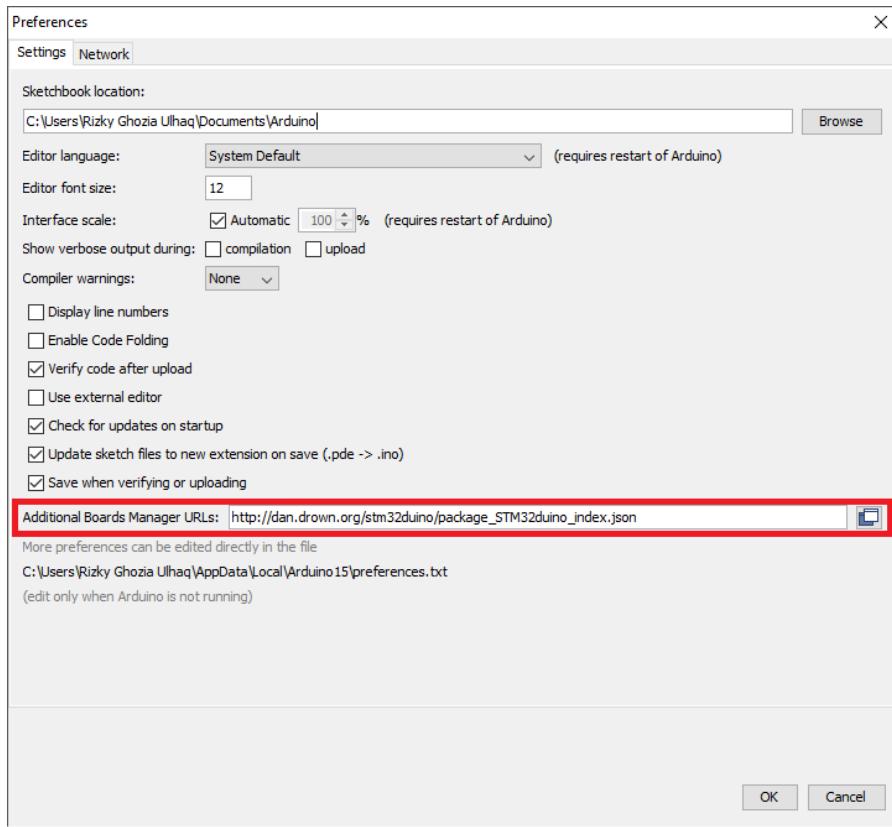
Arduino\_STM32-794d078de080054ea39095f8980bdcc2de9f3020

(1)\Arduino\_STM32-

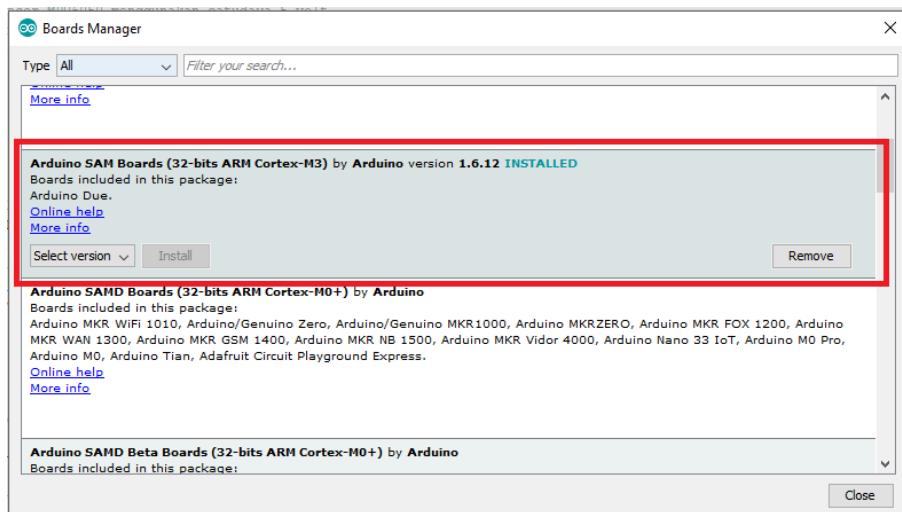
794d078de080054ea39095f8980bdcc2de9f3020\drivers\win

c. Setelah install driver berikut lanjutkan install dua file lainnya

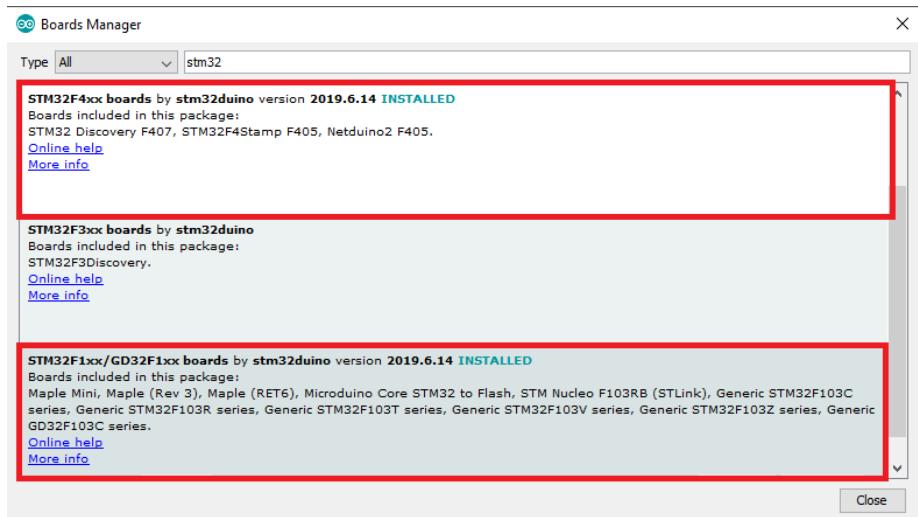
d. Lalu buka Arduino IDE → Preference → ketikan alamat [http://dan.drown.org/stm32duino/package\\_STM32duino\\_index.json](http://dan.drown.org/stm32duino/package_STM32duino_index.json) pada kolom Additional Board seperti gambar berikut:



- e. Setelah itu buka Tool → Board → Board Manager → Arduino SAM → Instal



- f. Cari STM32 lalu install sesuai yang diberi kotak merah pada gambar berikut:



- g. Setelah itu pilih Close dan cek pada Board apakah STM32 sudah tersedia.

## D. PROTOKOL KOMUNIKASI ARDUINO

### 1. Analog to Digital Converter (ADC)

Data analog merupakan data berupa sinyal berbentuk gelombang yang kontinyu. Gelombang ini pada umumnya berbentuk gelombang sinus dengan tiga variable dasar, yaitu amplitude, frekuensi, dan phase. Analog to Digital Converter (ADC) berfungsi sebagai pengubah sinyal analog menjadi digital. Prinsip kerjanya yaitu dengan mengkonversi sinyal analog kedalam bentuk besaran yang merupakan rasio perbandingan sinyal input dan tegangan referensi. Sinyal digital dapat berupa decimal ataupun biner.

Penggunaan data analog terkhusus pada pembacaan ADC lebih banyak digunakan pada perangkat berupa sensor yang memiliki kondisi lebih dari dua keadaan standar. Komponen yang menggunakan data ADC pada umumnya berupa sensor seperti sensor suhu LM35, cahaya photodiode, gaas, dan sebagainya. Pada pemrogramannya di Arduino dengan menggunakan perintahh “analogRead (pin);”.

### 2. Komunikasi serial USART

Komunikasi Serial merupakan sala satu alternatif yang relatif lebih murah untuk menggantikan komunikasi paralel, karena transfer data paralel menggunakan 8 jalur kondukt/kawat untuk menstransfer 8bit sekaligus.

Jika menggunakan komunikasi serial, maka hal tersebut dapat dilakukan dengan hanya menggunakan satu kawat konduktor saja, tetapi ditransfer bit demi bit sebanyak 8 kali untuk dapat menyelesaikan transfer satu byte data. Selain relatif lebih murah, komunikasi serial juga memberikan jangkauan transmisi yang lebih panjang dari komunikasi paralel. Bandingkan dengan komunikasi paralel yang hanya bisa 1 hingga dua meter saja, dengan komunikasi serial maka jangkauan tersebut bisa dilakukan berlipat-lipat ganda. Misalnya dengan menggunakan standar komunikasi serial EIA RS232 yang lebih dikenal dengan standard RS-232, dapat melakukan transmisi sejauh kurang lebih 50 feet dengan baud rate 9600. Transmisi yang lebih jauh dapat dilakukan pada data rate yang lebih rendah, dan jarak transmisi menjadi lebih pendek pada data rate yang lebih besar dari 9600 baud. Device yang melakukan konversi paralel ke serial dan konversi serial ke paralel salah satunya disebut universal asynchronous receiver-transmitter (UART). Selain itu terdapat pula yang disebut universal synchronous asynchronous receiver-transmitter (USART), di mana selain dapat melakukan komunikasi serial secara asinkron juga dapat dengan cara sinkron. Baik cara sinkron maupun asinkron keduanya sama-sama mempunyai keuntungan dan kelemahannya masing-masing.

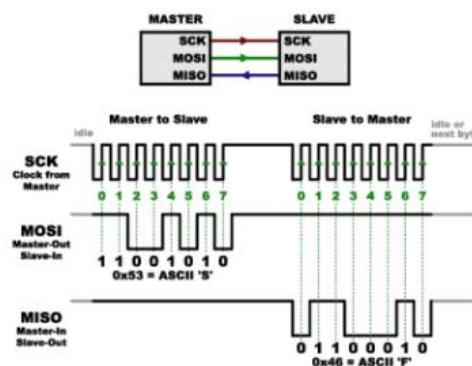
*Universal Synchronous And Asynchronous Receiver Transmitter* (USART) merupakan gabungan dari teknik pengiriman tak sinkron dan sinkron dan memiliki komunikasi serial dua arah. USART salah satu perangkat komunikasi serial yang mempunyai tingkat fleksibilitas komunikasi yang sangat tinggi. Syarat komponen untuk berkomunikasi dengan mikrokontroler USART adalah komponen atau modul yang memiliki pin RX (receiver) dan TX (transmitter). Fitur-fiturnya mencakup:

- a. Operasi full duplex
- b. Dapat beroperasi pada mode sinkron dan tak sinkron
- c. Dapat bekerja dengan resolusi baudrate yang tinggi
- d. Mendukung serial frames dengan data bit 5, 6, 7, 8, atau 9 data bit dan dilengkapi dengan 1 atau 2 stop bit.

- e. Dilengkapi dengan fasilitas parity check dan dapat bekerja pada parity Odd atau Even.
- f. Dapat beroperasi sebagai Master atau Slave Clock Synchronous.
- g. Dilengkapi dengan fitur data Over Run Detection.
- h. Terdapat fasilitas Frame Error Detection.
- i. Dilengkapi dengan filter untuk menyaring noise yaitu digital low pass filter.
- j. Mempunyai tiga interupsi yang terpisah pata Tx Complete, Data Register Empty Rx complete.
- k. Mendukung komunikasi multiprosessor.
- l. Bekerja pada mode komunikasi Double Speed Asynchronous.

### 3. Serial Peripheral Interface (SPI)

Serial peripheral interface (SPI) merupakan komunikasi serial sinkron karena menggunakan jalur terpisah antara data dan clock. Jalur data dan clock yang terpisah membuat jalur komunikasi SPI menjadi lebih cepat dan lebih akurat. Pada umumnya komunikasi ini memiliki beberapa jalur yaitu Vcc, Ground, Serial Clock (SCK), Master Output Slave Input (MOSI), Master Input Slave Output (MISO), Slave-Select (SS) dan terkadang terdapat pin Reset (R). namun pada dasarnya hanya 3 pin/jalur yang digunakan yaitu SCK, MOSI, dan MISO.

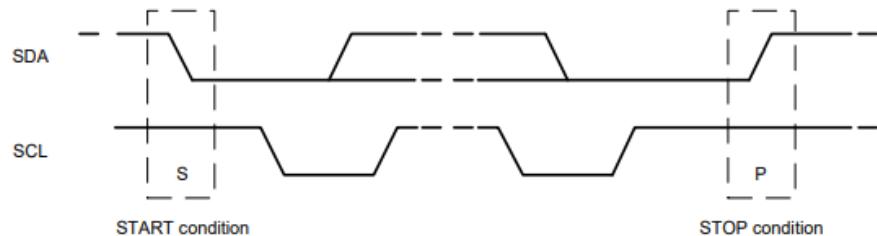


Gambar 11. Komunikasi SPI

Komponen yang dapat digunakan menggunkana komunikasi SPI harus memiliki jalur komunikasi MOSI, MISO, dan SCK. Beberapa komponen yang memiliki jalur data tersebut yaitu Bluetooth, RFID, SD Card, dan sebagainya. Pada pemrogramnnya di Arduino dengan menggunakan library “SPI.h”.

#### 4. Inter Intergrated Circuit (I2C)

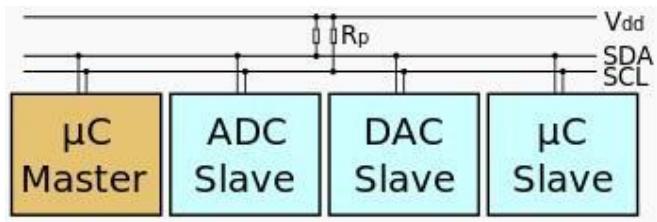
Inter integrated circuit (I2C) merupakan jenis komunikasi menggunakan dua saluran yang didesain khusus untuk mengirim dan menerima data. System I2C sendiri memiliki jalur Serial Clock (SCL) dan Serial Data (SDA) yang bekerja sebagai pembawa informasi data. Komponen yang dihubungkan dengan I2C dioperasikan sebagai Master dan Slave. Master merupakan komponen yang mulai transfer/pengirim data pada I2C dengan membentuk sinyal Start, mengakhiri transfer dengan sinyal Stop, dan membangkitkan sinyal clock. Sedangkan slave berperan sebagai penerima data yang dikirim oleh master melalui I2C.



Gambar 12. Kondisi Start dan Stop

Kecepatan transfer data dari protocol I2C ditentukan oleh besarnya clock speed yang digunakan pada jalur SCL. Kecepatan standar yang digunakan pada jalur SCL sebesar 100 kHz. Jumlah pengalamatan pada protocol I2C sebanyak 7bit alamat, sehingga meskipun hanya memiliki dua pin komunikasi yaitu SDA dan SCL, I2C memiliki kemampuan untuk digunakan multislave. Jadi setiap jalur I2C dapat digunakan untuk perangkat Slave sebanyak 2 buah dengan kisaran alamat antara 0 sampai dengan 127. Salah satu komponen yang dapat menggunakan komunikasi ini

yaitu MPU6050. Pada pemrogramannya di Arduino dengan menggunakan library “Wire.h”.



Gambar 13. Skema Komunikasi multislv I2C

#### E. PEMROGRAMAN DASAR ARDUINO

Arduino IDE merupakan salah satu aplikasi yang digunakan untuk memrogram mikrokontroler menggunakan bahasa C. Ketika membuka aplikasi Arduino Terdapat default program yaitu “void setup ();” dan “void loop ();”. Kode program yang dijalankan sekali diletakan di dalam void setup sedangkan program yang akan dijalankan berulang-ulang diletakan di dalam void loop. Setup memiliki arti sebagai pengaturan pada program, contohnya untuk komunikasi dengan komputer maka menggunakan “Serial.begin (nilai Baudrate);”.

```

/*
Blink
Turns on an LED on for one second, then off for one second, repeatedly.

Most Arduinos have an on-board LED you can control. On the UNO, MEGA and ZERO |
it is attached to digital pin 13, on MKR1000 on pin 6. LED_BUILTIN takes care
of use the correct LED pin whatever is the board used.
If you want to know what pin the on-board LED is connected to on your Arduino model,
check the Technical Specs of your board at https://www.arduino.cc/en/Main/Products

This example code is in the public domain.

modified 8 May 2014
by Scott Fitzgerald

modified 2 Sep 2016
by Arturo Guadalupi
*/
// the setup function runs once when you press reset or power the board
void setup() {
    // initialize digital pin LED_BUILTIN as an output.
    pinMode(LED_BUILTIN, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
    digitalWrite(LED_BUILTIN, HIGH);      // turn the LED on (HIGH is the voltage level)
    delay(1000);                         // wait for a second
    digitalWrite(LED_BUILTIN, LOW);        // turn the LED off by making the voltage LOW
    delay(1000);                         // wait for a second
}

```

#### 1. Titik koma (;

Pada setiap baris kode pada program harus diakhiri dengan titik koma (;).

#### 2. Kurung kurawal {}

Sebelum menulis program akan diberi ruang pembatas awalan dan akhiran dari suatu program yaitu dengan menggunakan kurung kurawal.

#### 3. Garis miring

Garis miring dapat digunakan untuk memberi catatan atau comment pada satu baris yang tidak akan dibaca sebagai program yaitu dengan memberi tanda dua garis miring (//). Jika menginginkan satu paragraph catatan atau comment maka menggunakan bintang garis miring (\*/).

#### 4. Variable

Variable merupakan kode program yang digunakan untuk menyimpan uatu nilai pada suatu nama dan memiliki beberapa tipe data. Setiap tipe data memiki ukuran dan jangkauan nilai yang berbeda. Berikut ini tipe-tipe data yang sering digunakan pada pemrograman Arduino.

Tipe	Ukuran (bit)	Jangkauan
Bool	8	0 dan 1
Char	8	-128 sampai 127
Unsigned char	8	0 sampai 255
Signed char	8	-128 sampai 127
Byte	8	0 sampai 255
Int	16	-32768 sampai 32767
Short int	16	-32768 sampai 32767
Unsigned int	16	0 sampai 65535
Signed int	16	-32768 sampai 32767
Long int	32	-2147483648 sampai 2147483647
Unsigned long int	32	0 sampai 4294967295
Signed long int	32	-2147483648 sampai 2147483647
Float	32	+/- 1.145e-38 sampai 3.402e38
Double	32	+/- 1.145e-38 sampai 3.402e38

## 5. Operator matematika

Operasi matematika digunakan untuk memanipulasi nilai dengan perhitungan matematika sederhana seperti penjumlahan, pengurangan, sama dengan, dan sebagainya.

=	Sama dengan $X=10*2$ artinya x bernilai 200
%	Persentase 2%10 artinya 2
+	Penambahan
-	Pengurangan
*	Perkalian
/	pembagian

## 6. Operator perbandingan kondisi

Perbandingan kondisi secara logika menggunakan operator berikut ini

==	Sama dengan 15==10 FALSE 15==15 TRUE
!=	Tidak sama dengan 15 != 10 TRUE 15 != 15 FALSE
<	Lebih kecil dari
>	Lebih besar dari

## 7. Instruksi pengendali

Instruksi pengendali digunakan sebagai instruksi yang akan berjalan ketika kondisi tertentu. Berikut ini instruksi Bahasa C yang sering digunakan:

No	Fungsi	Penulisan
1	Syarat	if (kondisi) { ..... (aksi yang dikerjakan) };
2	Percabangan	if (kondisi) { ..... (aksi yang dikerjakan) } else if (kondisi) { ..... (aksi yang dikerjakan) }

		<pre> ..... ..... else { ..... (aksi yang dikerjakan) }; </pre>
3	Percabangan	<pre> switch (variable) {     case nilai_variabel_ke-1     {         ..... (aksi yang dikerjakan)     }     case nilai_variabel_ke-2     {         ..... (aksi yang dikerjakan)     }     .....     .....     default:     {         ..... (aksi yang dikerjakan)     } } </pre>
4	Melompat	<pre> goto alamat_tujuan; ..... ..... alamat_tujuan: </pre>

		..... .....
5	Melompat keluar dari perulangan	break;
6	Perulangan	while (kondisi) { ..... (aksi yang dikerjakan) }
7	Perulangan	Do { ..... (aksi yang dikerjakan) } while (kondisi);
8	Perulangan	for (nilai_awal,syarat,operasi++) { ..... (aksi yang dikerjakan) };

## BAB IV

### MPU6050

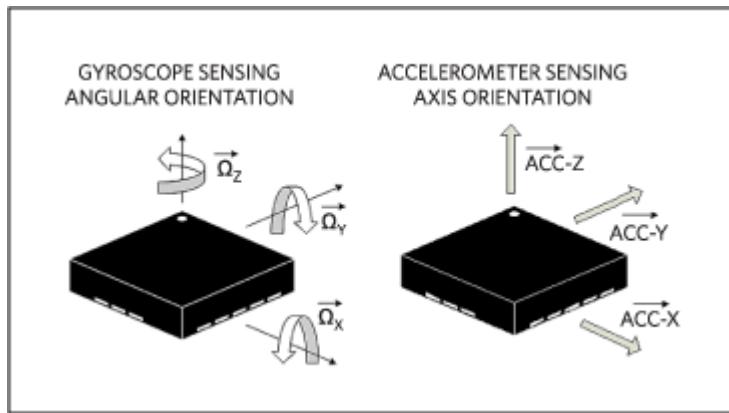
#### A. MODUL MPU6050

Berdasarkan datasheet sensor MPU6050 adalah perangkat sensor yang terdapat 3-axis accelerometer (sensor percepatan), 3-axis gyroscope (sensor keseimbangan) atau yang dikenal dengan 6DOF (*Degrees of Freedom*), suhu, dan magnetometer. Sensor MPU6050 merupakan salah satu jenis alat elektronik yang digunakan sebagai pengukur inersia atau yang disebut dengan Inertial Measurement Unit (IMU) yang dapat mengukur kecepatan, orientasi, dan gaya gravitasi. Nilai yang dihasilkan sensor didapat dari gerakan tiga sumbu yaitu x, y, dan z. Sensor ini mampu membaca kemiringan sudut berdasarkan data sensor yang termuat pada modul MPU6050. Akses modul sensor ini menggunakan jalur data I2C.

Part/ Item	MPU6050	MPU6000
VDD	2.375V-3.46V	2.375V-3.46V
VLOGIC	1.71V to VDD	n/a
Serial Interface Supported	I2C	I2C, SPI
Pin 8	VLOGIC	/CS
Pin 9	AD0	AD0/ SD0
Pin 23	SCL	SCL/ SCLK
Pin 24	SDA	SDA/ SDI



Gambar 14. Modul MPU6050

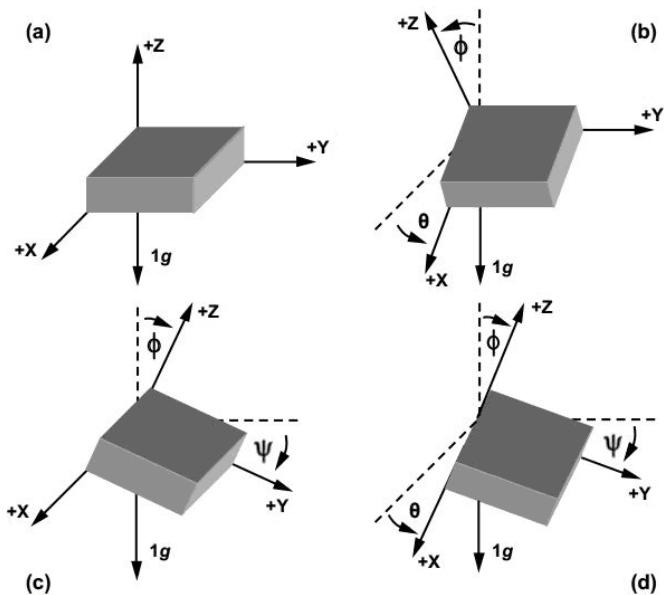


Gambar 15. Perbedaan Accelerometer dan Gyroscope

## B. ACCELEROMETER

Dasar dari accelerometer yaitu Hukum Newton II yang mengatakan bahwa percepatan yang dialami suatu benda berbanding lurus terhadap gaya dan berbanding terbalik dengan massa benda yang bekerja. Ketika kecepatan suatu benda berubah maka benda tersebut mengalami akselerasi. Akselerasi atau yang sering disebut percepatan merupakan laju perubahan kecepatan terhadap waktu. Sama seperti kecepatan, percepatan juga terbagi menjadi dua yaitu percepatan rata-rata dan sesaat. Percepatan merupakan hasil dari turunan kecepatan terhadap waktu dan kecepatan didapatkan dari hasil turunan dari perpindahan terhadap waktu.

MPU6050 terdiri dari Accelerometer 3-sumbu dengan teknologi Micro Electro Mechanical (MEMs). Accelerometer digunakan untuk mengukur sudut kemiringan atau kemiringan sepanjang sumbu X, Y dan Z dikarenakan adanya perubahan kecepatan yaitu percepatan, mendekripsi dan mengukur getaran. seperti yang ditunjukkan pada gambar 16.



Gambar 16. Accelerometer

Fitur sensor accelerometer pada MPU6050 sebagai berikut:

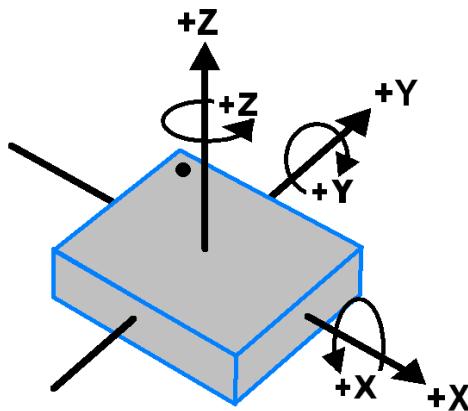
- Digital output dengan 3 axis dengan jangkauan  $\pm 2g$ ,  $\pm 4g$ ,  $\pm 8g$ , dan  $\pm 16g$ .
- Memiliki satuan berupa g (gaya gravitasi)
- Terintegrasi ADC 16bit yang digunakan untuk mendapatkan output digital
- Ketika diletakan pada permukaan datar maka nilai X dan Y Axis yaitu 0g sedangkan Z axis bernilai +1g
- Memerlukan arus 500 micro ampere
- Orientation detection dan signaling
- Tap detection
- User programmable interrupts
- High-G interrupt
- User self-test

### C. GYROSCOPE

MPU6050 terdiri dari Giroskop 3-sumbu dengan teknologi Micro Electro Mechanical System (MEMS). Ini digunakan untuk mendeteksi kecepatan rotasi/sudut sepanjang sumbu X, Y, Z dalam satuan derajat per detik seperti yang ditunjukkan pada gambar 7. Ketika gyro berputar dan

menyentuh sumbu maka Coriolis Effect akan menyebabkan getaran yang terdeteki oleh MEM didalam MPU6050. Sinyal yang dihasilkan akan diperkuat, didemodulasi, dan disaring untuk menghasilkan tegangan yang sebanding dengan rentang kecepatan sudut.

Gyroscope memiliki output yang eka terhadap kecepatan sudut dari arah sumbu x yang nantinya akan menjadi sudut phi (roll), arah sumbu y akan menghasilkan sudut theta (pitch) dan sumbu z akan menjadi sudut psi (yaw). Data yang masuk kedalam mikrokontroler berupa rad/s dan akan dikonversi menjadi deg/s. terdapat banyak versi dari persamaan yang digunakan dalam pengolahan data gyroscope.



Gambar 17. MPU6050 Orientation and polarity of Rotation

Fitur sensor gyroscope pada MPU6050 sebagai berikut:

- Digital output X-, Y-, dan Z- axis sudut kemiringan
- Rentang user-programmable  $\pm 250, \pm 500, \pm 1000$ , dan  $\pm 2000$  derajat per detik
- Terintegrasi ADC 16bit
- Operating current 3.6mA
- User self-test
- Digitally-programmable low-pass filter
- Improved low-frequency noise performance

## **BAB V**

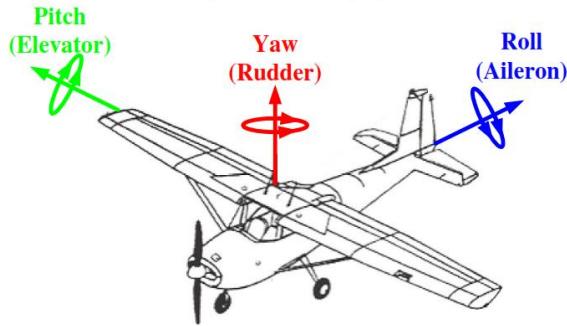
### **NAVIGASI DRONE PEMANTAU GUNUNG BERAPI**

Navigasi atau panduan arah merupakan teknik untuk membaca kedudukan (posisi) dan arah benda terhadap konsidi sekitarnya. Untuk melihat posisi dari suatu benda kita dapat menggunakan posisi dari benda yang lainnya. UAV atau kendaraan udara tanpa awak merupakan perkembangan teknologi yang memiliki potensi sangat besar untuk berbagai macam keperluan, baik untuk militer maupun untuk kepentingan sipil. Keuntungan dari penggunaan UAV adalah dapat digunakan untuk berbagaimacam misi-misi dengan resiko bahaya besar sehingga dapat mengurangi resiko pilot/manusia.

Pada permulaannya UAV (Unmanned Aerial Vehicle) dikembangkan oleh militer untuk berbagai macam operasi rahasia dan berbahaya. Namun seiring perkembangan teknologi dan perkembangan ilmu pengetahuan menuntut peneliti turut serta menggunakan UAV untuk berbagaimacam bidang penelitian yang mereka butuhkan sesuai dengan fungsi. Beberapa aplikasi yang merupakan implementasi dari UAV adalah untuk kebutuhan fotografi, survey, monitoring, pencarian sesuai, pemantauan lahan, dan keperluan lainnya. UAV sering digunakan untuk melakukan pekerjaan yang memiliki resiko tinggi.

Seperti halnya kendaraan pada umumnya UAV juga membutuhkan pembacaan kondisi dan posisi dari lingkungan sekitarnya. Oleh karena itu UAV membutuhkan sistem navigasi untuk mengetahui kondisi sekitar dengan berbagaimacam sensor. Penggunaan sistem navigasi pad UAV tidak diletakkan pada bagian dalam UAV melainkan dikirimkan melalui pusat kendali dari UAV tersebut.

UAV pada umumnya digunakan untuk berbagai macam operasi berbahaya yang apabila dilakukan oleh pesawat berawak akan membahayakan penumpang maupun pilot pesawat tersebut. Beberapa UAV sudah dilengkapi dengan berbagai senjata yang digunakan untuk kawasan perang, seperti Predator RQ-1 yang telah dapat melakukan serangan ke target-target di darat, bahkan kendaraan ini mampu melakukan serangan yang tertuju pada pesawat lain di udara.

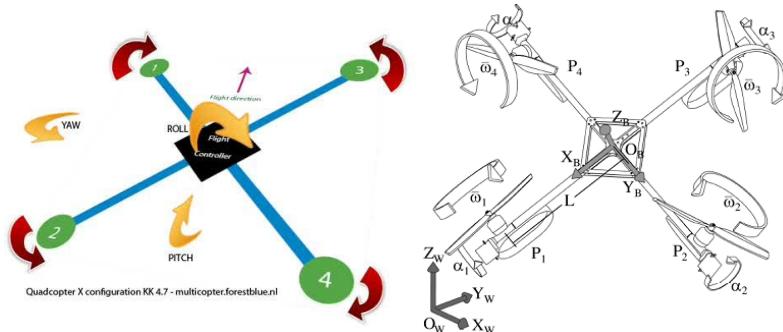


Gambar 18. Sudut kebebasan

### A. GERAKAN UAV

*Quadcopter* merupakan pesawat multirotor/multiwing yang memiliki rotor sejumlah 4 buah yang memiliki gerakan lebih leluasa dibandingkan dengan helicopter dengan 2 buah rotor. Terdapat 4 gerakan dasar pada *Quadcopter* yaitu gerakan altitude (throttle), gerakan sudut (*Roll*), gerakan sudut *Pitch*, dan gerakan sudut *Yaw*.

Gerakan throttle merupakan gerak translasi *Quadcopter* sepanjang sumbu-z, Gerakan ini dipengaruhi oleh perubahan kecepatan keempat rotor dengan nilai kecepatan putaran yang sama. Dengan pergerakan Throttle, *Quadcopter* akan dapat melakukan aksesori naik atau turun sesuai dengan tingkat kecepatan rotor. Gerakan *Roll* merupakan gerakan rotasi pada sumbu-x, gerakan ini dipengaruhi oleh perubahan kecepatan rotor kanan dan kiri. Gerakan *Pitch* merupakan gerakan rotasi pada sumbu-y, gerakan ini dipengaruhi oleh perubahan kecepatan pada motor depan dan belakang. Gerakan sudut *Yaw* merupakan gerakan rotasi pada sumbu z.



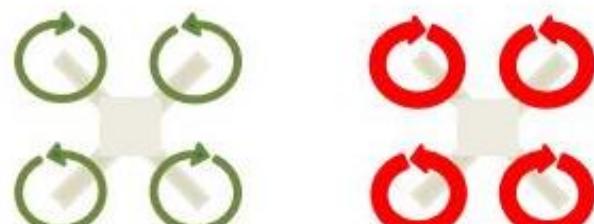
Gambar 19. Gerakan sudut Quadcpter

Pola Terbang *Quadcopter* untuk manuver atau bergerak mempunyai aturan khusus berdasarkan pergerakan rotor pada keempat sisinya.

### 1. Throttle

Gerak menaikan atau menurunkan ketinggian dengan mengatur kecepatan keempat rotor sama besar. *Quadcopter* akan bergerak searah sumbu-z dengan mengacu pada koordinat *Quadcopter*. Arah putar masing-masing rotor seperti pada gambar 11.

#### Throttle control



Move down

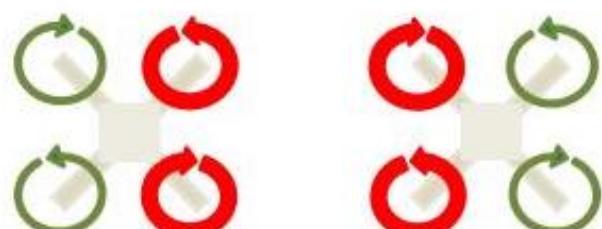
Move up

Gambar 20. Gerak Throttle

### 2. Roll

Gerak manuver kesamping dengan cara meningkatkan kecepatan putar bagian kiri atau kanan *Quadcopter* dengan bersamaan. Ilustrasi gerakan *Roll* pada gerak rotasi *Quadcopter* seperti gambar 12.

#### Roll control



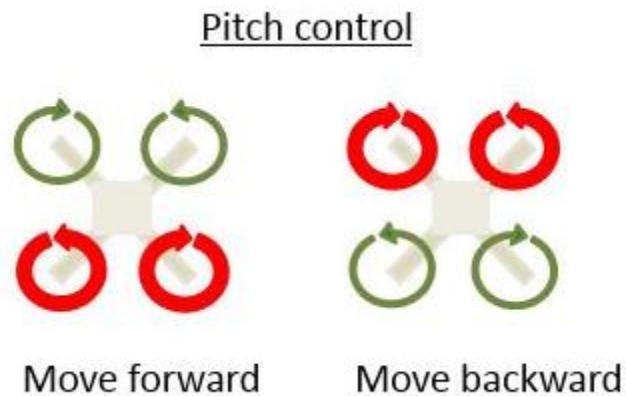
Bend left

Bend Right

Gambar 21. Gerak *Roll*

### 3. *Pitch*

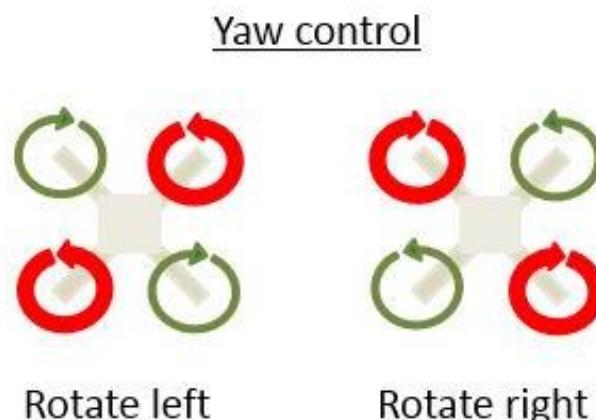
Gerak *Pitch* hampir sama dengan gerak *Roll*. Penerapan *Pitch* pada manuver *Quadcopter* adalah dengan meningkatkan atau menurunkan kecepatan putar rotor bagian depan atau belakang secara bersama dan kecepatan sama.



Gambar 22. Gerak *Pitch*

### 4. *Yaw*

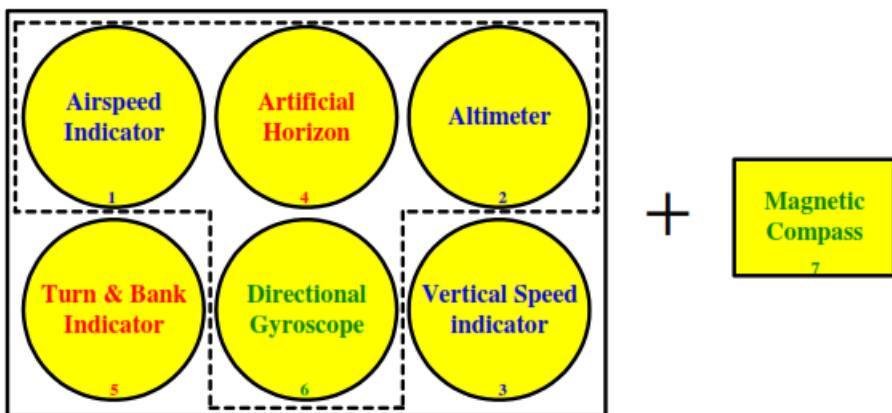
Gerakan *Yaw* dilakukan dengan cara meningkatkan kecepatan putar rotor dengan posisi bersilangan secara bersama-sama. Tujuan gerak ini adalah mendapatkan manuver *Quadcopter* berputar searah atau berlawanan jarum jam. Ilustrasi untuk gerak *Yaw* pada *Quadcopter* dapat dilihat pada gambar 14.



Gambar 23. Gerak *Yaw*

## B. DASAR INSTRUMEN PENERBANGAN

Tujuan dari adanya instrument pada system UAV yaitu untuk membantu pilot dengan memberikan informasi-informasi mengenai posisi real dari wahana ketika diudara yang dapat membantu menjaga keselamatan dan keefektifan wahana dari bahaya.



Gambar 24. Basic Flight Instrumen

Berdasarkan gambar 11 dasar instrumen penerbangan digolongkan sebagai berikut:

1. Pilot static instrument: 1, 2, dan 3
2. Altitude instrument: 4 dan 5
3. Heading instrument: 6 dan 7

Ground Control Station (GCS) sebagai pendukung wahana terbang akan membantu pilot dengan mengawasi melalui Grafic User Interface (GUI) yang menampilkan informasi mengenai wahana UAV yang sedang beroperasi. GUI akan menampilkan informasi seperti Basic Flight Instrumen. Berikut penjelasan secara singkat mengenai dasar instrument penerbangan.

### 1. Pilot static instrument

Pilot static instrument memiliki fungsi sebagai petunjuk dari wahana UAV yang beroperasi dikendalikan oleh tekanan static dan tekanan pitot dari tabung pitot static. Pada bagian ini menggunakan perhitungan Bernoulli.

a. Airspeed Indicator (AI)

Indikator kecepatan didapatkan melalui pengukuran perbedaan antara pitot dan tekanan statis, sehingga memperoleh tekanan dinamis.

b. Altimeter

Barometer akan mengukur perubahan ketinggian tekanan dan menampilkan tinggi ketinggian UAV. Altimeter dapat diatur ke mode tekanan berbeda untuk pembacaan ketinggian tertentu. Altitude digunakan untuk menghindari kecelakaan dan menjaga keamanan jalur udara, dimana pengaturan altimeter diatur berdasarkan wilayah penerbangan.

c. Vertical Speed Indicator (VSI)

VSI akan menampilkan hasil perhitungan dari tekanan altitude dan kecepatan terbang secara vertical. Jika wahana terbang lurus, maka tekanan akan konstan oleh karena itu VSI akan diatur bernilai 0.

## 2. Altitude instrument

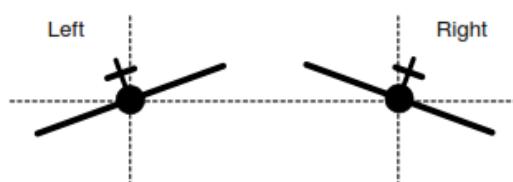
Pada bagian Altitude didasarkan dengan penggunaan gyroscope untuk mengetahui nilai roll dan pitch dari wahana.

a. Turn and bank indicator

Kinerja dari gyroscope didasarkan oleh gerak dari penggerak UAV seperti turbine atau motor elektrik. Hal ini menyebabkan kemungkinan UAV dapat terbang secara auto tanpa adanya pilot. Instrument ini terdiri dari dua mekanisme yang terpisah.

- Gyro driven pointer.

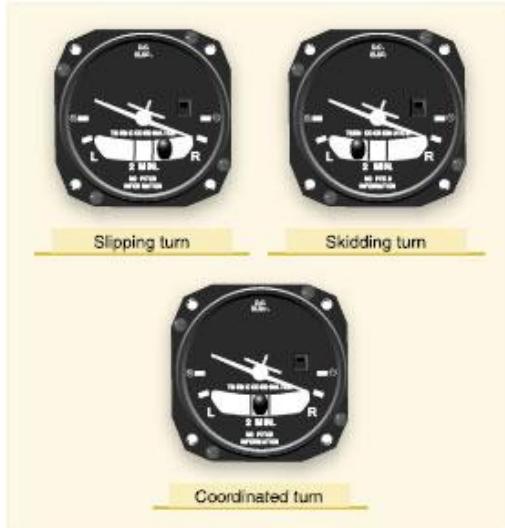
Bagian ini sebagai indicator petunjuk arah belok dari UAV dan kecepatannya.



Gambar 25. Gerakan belok UAV

- Detect slip dan skid in turn.

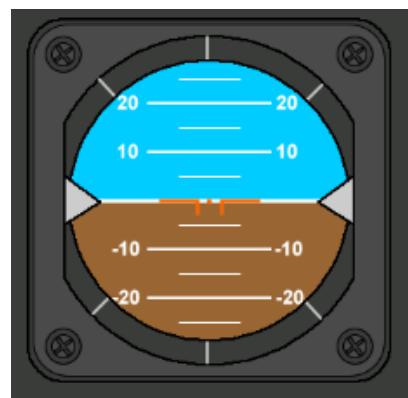
Slip terjadi apabila gerakan dari yaw terlalu terlambat jika dibandingkan dengan sudut gerak roll. Skid terjadi apabila gerakan yaw terlalu cepat dibandingkan sudut gerak roll



Gambar 26. Petunjuk turn indicator

#### b. Artificial horizon

Instrumen yang dioperasikan gyro yang menunjukkan sikap roll pada sumbu x dan pitch pada sumbu y pesawat terbang yang dihubungkan dengan garis lurus buatan sebagai referensi bumi. Gyro berputar pada sumbu z yaw dan gyro bergerak berdasarkan gerakan UAV yang diekndalikan oleh turbin udara vakum atau motor listrik.



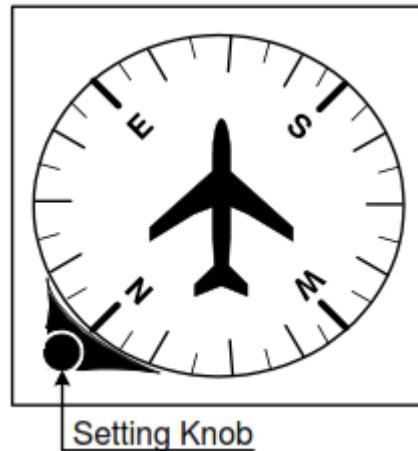
Gambar 27. Artificial Instrumen

#### 3. Heading instrument

a. Directional gyroscope (DG)

Instrument ini menampilkan gerakan UAV berdasarkan sudut yaw.

Perputaran gyro pada pitch sumbu y dan dikuatkan dengan sumbu Y dan Z dalam 2D.



Gambar 28. Indikator arah gyroscope

b. Magnetic Compass

Instrument ini menampilkan pesawat dalam posisi horizontal terhadap gravitasi bumi.

## **DAFTAR PUSTAKA**

- Austin, Reg. 2010. Unmanned Aircraft System: UAVS Design, Development and Deployment. United Kingdom. Wiley A John and Sons, Ltd., Publication.
- Abdulla, M., Svoboda, J. V., & Rodrigues, L. 2005. Avionics Made Simple. Montreal, Quebec, Canada.
- Wiratama, C. 27 Mei 2016. Unmaned Aerial Vehicle (UAV). Diunduh pada tanggal 8 Juli 2019 dari <http://aeroengineering.co.id/2016/05/unmaned-aerial-vehicle-uav/>
- STM32. 15 Maret 2019. Blue Pill. Diunduh pada tanggal 8 Juli 2019 dari [http://wiki.stm32duino.com/index.php?title=Blue\\_Pill](http://wiki.stm32duino.com/index.php?title=Blue_Pill)
- The Physics Hypertextbok. Acceleration. Diunduh pada tanggal 8 Juli 2019 dari <https://physics.info/acceleration/>
- Syam, R, Ph.D. 2013. Dasar-dasar Teknik Sensor. Jurusan Teknik Mesin Fakultas Teknik Universitas Hasanudin.
- Dadafshar, M. 2014. Accelerometer and Gyroscopes Sensors: Operation, Sensing, and Applications. Maxim Integrated.
- InvenSense Dokument number: RM-MPU-6000A-00. 2013 MPU-6000 and MPU-6050 Register Map and Descriptions Revision 4.2. USA: InvenSense.
- Rachmat C, A. & Wikan A, M. 2016. Konsep dan Impelemtnasi Pemrograman GUI. Universitas Kristen Duta Wacana.

# **BUKU PANDUAN**

## **NAVIGASI DRONE GUNUNG BERAPI DENGAN ACCELEROMETER DAN GYROSCOPE**

**DISUSUN OLEH:**

**RIZKY GHÖZIA ULHAQ  
15518244007**

**DOSEN PEMBIMBING:  
HERLAMBANG SIGIT PURNOMO, M.SC.**



**BUKU PANDUAN**  
**DRONE PEMANTAU GUNUNG BERAPI**  
**PRAKTIK KENDALI DAN AKUISISI DATA**

## PENGENALAN ALAT

### A. ALAT

#### 1. Nama Alat

Nama alat ini adalah Drone Pemantau Gunung Berapi. Drone ini digunakan sebagai media pembelajaran Praktik Kendali dan Akuisisi Data.



Gambar 1. Drone Pemantau Gunung Berapi

#### 2. Definisi Alat

Drone Pemantau Gunung Berapi merupakan pengembangan media pembelajaran pada mata kuliah praktik kendali dan akuisisi data. Drone ini dilengkapi dengan Sensor MPU6050 untuk mendekripsi sumbu x, y, dan z dengan menggunakan sensor accelerometer dan gyroscope, radio telemetry untuk mengirim data, serta mikrokontroler STM32F103C8 untuk memproses pengolahan dan pengiriman data. Media pembelajaran ini juga terhubung dengan *Graphical User Interface* (GUI) untuk menampilkan data sensor dan *database*. GUI pada media pembelajaran ini menggunakan aplikasi Visual Studio.

#### 3. Spesifikasi Alat

Drone Pemantau Gunung Berapi memiliki spesifikasi seperti terdapat pada Tabel 1.

Tabel 1. Spesifikasi Drone Pemantau Gunung Berapi

Penggerak	Drone MJX BUGS 6W
Dimensi	310 x 310 x 140 mm
Mikrokontroler	STM32F103C8
Sensor	DHT22 & BMP180

Komunikasi	Modul Telemetry 433 MHz
Interface	GUI dengan aplikasi LabVIEW 2017
Catu Daya	Baterai Lippo 2 Cell 1300 MAh



Gambar 2. Drone Pemantau Gunung Berapi tampak atas



Gambar 3. Drone Pemantau Gunung Berapi tampak bawah

## B. Komponen Alat

### 1. Komponen *Software*

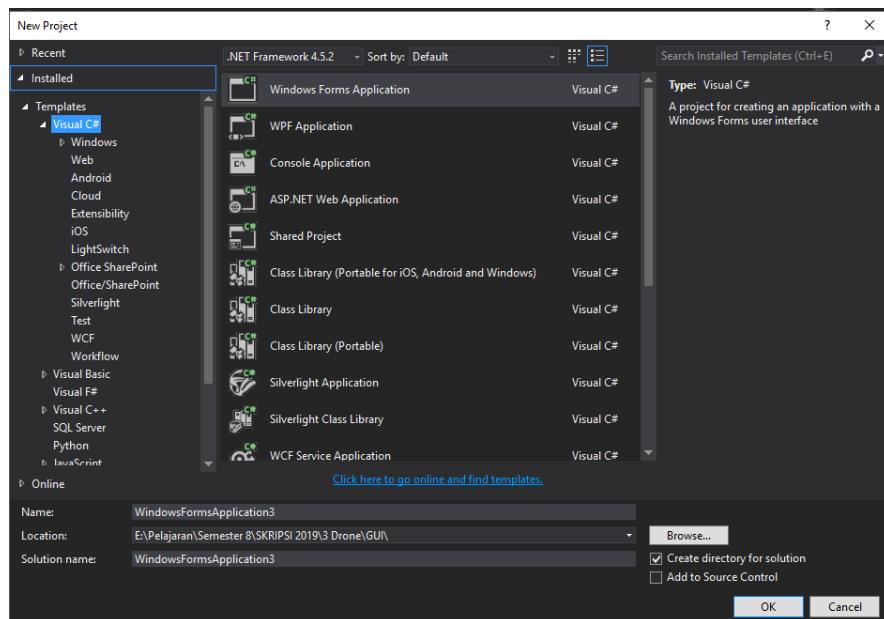
Perangkat lunak yang digunakan pada Drone Pemantau Gunung Berapi antara lain sebagai berikut :

#### a. Arduino IDE



Gambar 4. Tampilan lembar kerja Arduino IDE

#### b. Visual Studio 2015

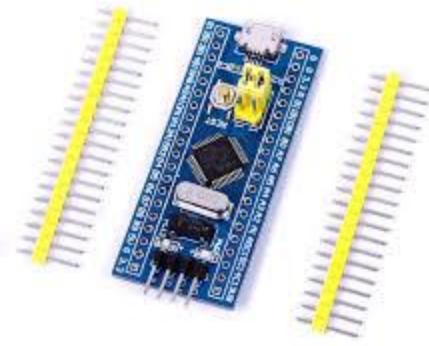


Gambar 5. Tampilan awal Visual Studio 2015

## 2. Komponen *Hardware*

Perangkat keras yang digunakan pada Drone Pemantau Gunung Berapi antara lain sebagai berikut :

- a. Mikrokontroler STM32F103C8



Gambar 6. STM32F103C8

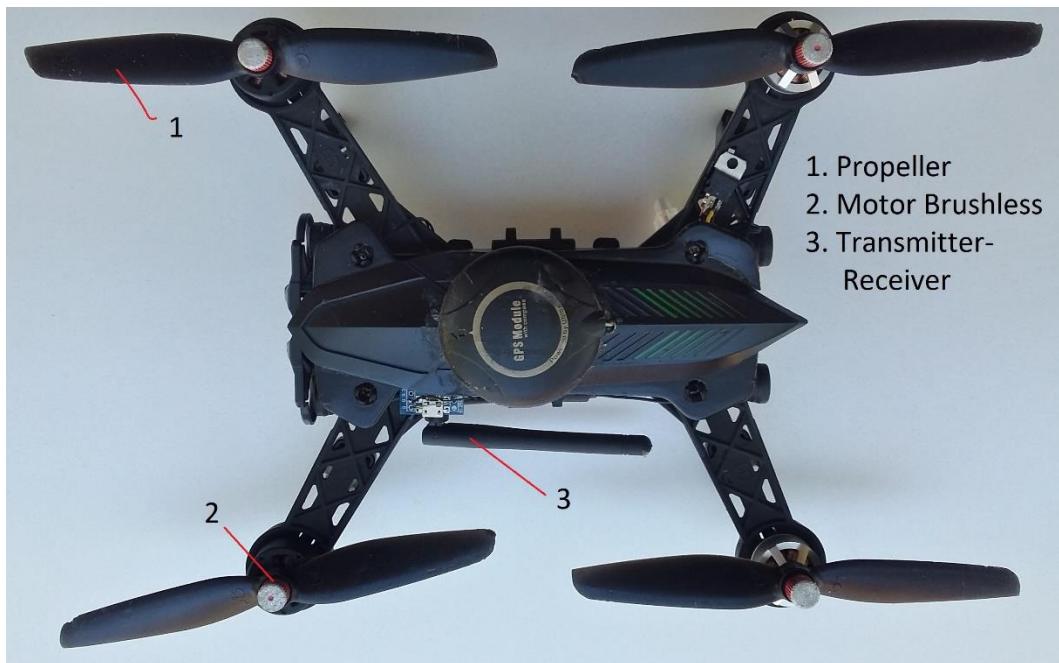
- b. Sensor MPU6050



Gambar 7. Sensor MPU6050

**PETUNJUK PENGGUNAAN DRONE PEMANTAU GUNUNG BERAPI DENGAN  
MPU6050 PADA GUI**

**A. BAGIAN-BAGIAN UNIT**



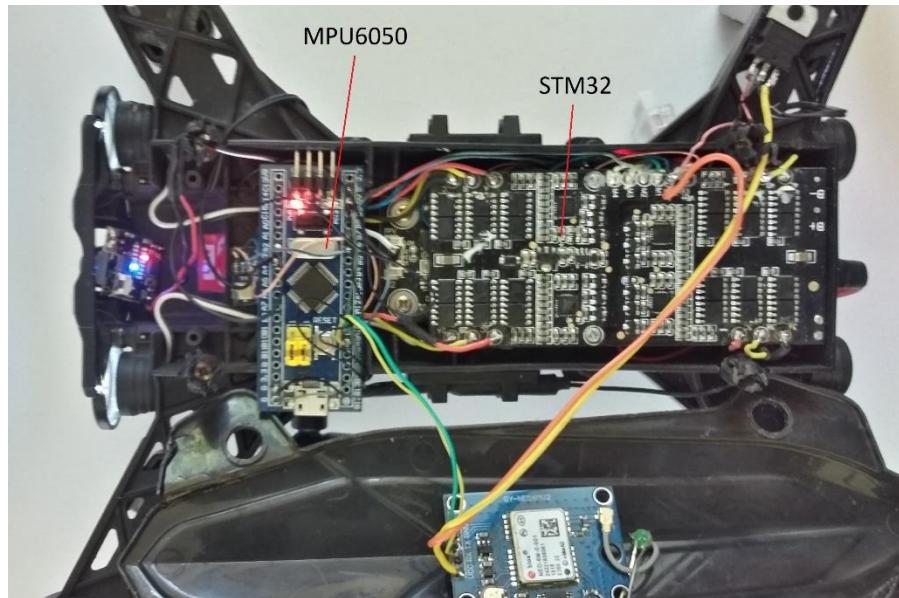
**Gambar 1. Main unit**



**Gambar 2. Remote Control**

NO.	KETERANGAN
-----	------------

1	kontrol kiri
2	Trime putar kanan/kiri
3	-
4	Lampu indicator power
5	Powr ON/OFF
6	kontrol kanan
7	Trim kanan/kiri
8	Trim trim maju/mundur
9	kontrol kecepatan
10	Menyalakan mesin
11	-
12	Fungsi flip



**Gambar 3. Sensor dan mikrokontroller**

## B. PERAKITAN

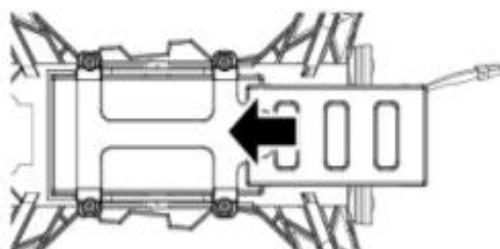
1. Perhatikan Keselamatan dan Kesehatan Kerja (K3) dalam mengoperasikan. Jangan pasang propeller jika belum siap.
2. Pasang propeller dengan arah yang sesuai

Sebelum memulai menerbangkan drone, perlu diperhatikan arah pemasangan dan putaran propeller drone pada motor. Arah putaran tiap propeller di bagi dua yaitu *clockwise* (searah jarumjam) dan *anticlockwise* (berlawanan arah jarumjam) hal ini bertujuan agar drone dapat terbang setabil dan sesuai dengan kendali pada remot kontrol. saat memasang propeller pastikan penguci dalam keadaan kuat agar propeller tidak lepas saat terbang. Berikut adalah arah putaran propeller pada drone:



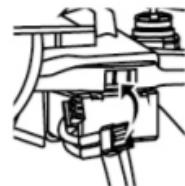
### 3. Pemasangan baterai

- Sebelum memasang baterai drone pastikan terlebih dahulu baterai dalam keadaan full dan tidak panas. Panas berlebih pada baterai bisa menurunkan kemampuan dan performa baterai bahkan dapat menyebabkan baterai meledak. Tipe baterai pada drone ini adalah lipo 2S dengan arus 1300Mah, sedangkan baterai pada remote kontrol drone hanya menggunakan baterai jenis AA sebanyak 4 buah dan berikut adalah cara pasang baterai pada drone dan remot kontrol:

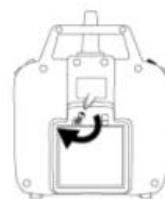
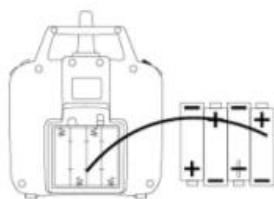
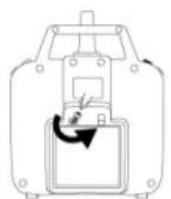


### Masukkan baterai

- b. Masukkan batrai pada bagian bawah drone yang sudah di sediakan, lalu sambungkan konektor baterai pada drone, lalu pada remot kontrol buka penutup baterai dan masukkan baterai jenis AA sebanyak 4 buah.



Menyambungkan konektor baterai

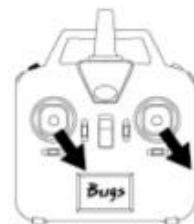


Baterai remot kontrol

## C. PENGOPERASIAN

### 1. Menjalankan Drone

- a. Untuk menerbangkan drone perlu dilakukan terlebih dahulu kalibrasi gyroscope, hal ini bertujuan agar drone dapat terbang stabil. Berikut adalah cara kalibrasi gyroscope.

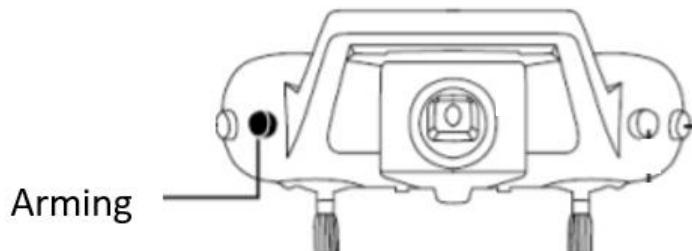


Kalibrasi gyroscope

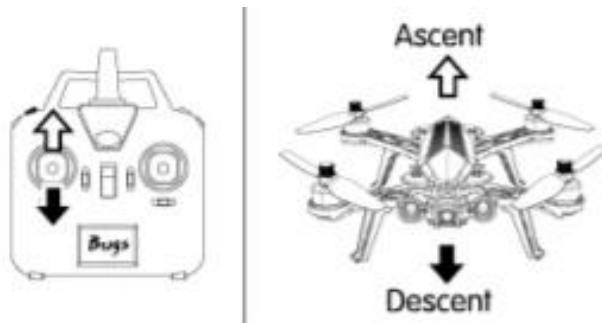
- b. Untuk melakukan kalibrasi tuas remote kiri maupun kanan di Tarik kea rah bawah kiri secara bersamaan, dan LED pada drone akan berkedip yang menunjukkan

bahwa proses kalibrasi berhasil dan drone siap terbang. jika tidak terjadi apaapa pada LED hubungi teknisi.

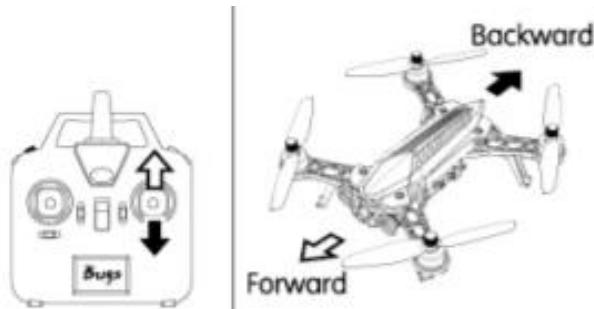
- c. Selanjutnya tunggu sampai GPS pada drone terkoneksi dengan satelit, tersambungnya GPS dengan satelit ditandai dengan LED GPS yang berkedip. Namun dalam proses ini membutuhkan waktu sekitar 2-3 menit dan GPS harus berada di luar ruangan. Jika LED GPS tidak bisa berkedip hubungi teknisi. Tekan tombol arming pada drone untuk memulai menyalakan mesin



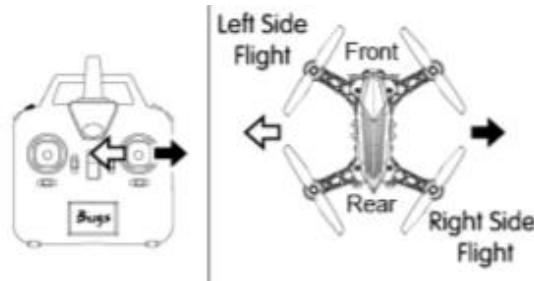
- d. Sebelum menekan tombol arming pastikan disekitar drone tidak ada benda yang bisa menghalangi putaran propeller drone dan disekitar drone tidak ada manusia maupun hewan. Jika mesin tidak menyala hubungi teknisi. Untuk menerbangkan drone geser tuas kiri kearah atas untuk naik dan arah bawah untuk turun.



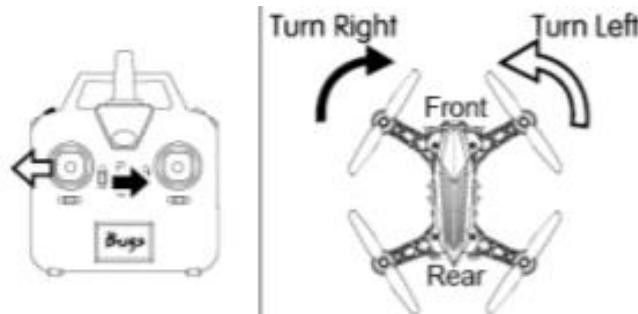
- e. Untuk mengontrol drone agar terbang kearah depan geser tuas kanan kearah atas untuk maju dan arah bawah untuk mundur.



- f. Untuk menerbangkan drone kearah kanan geser tuas kanan kearah kanan dan geser tuas kearah kiri agar drone dapat terbang kearah kiri.

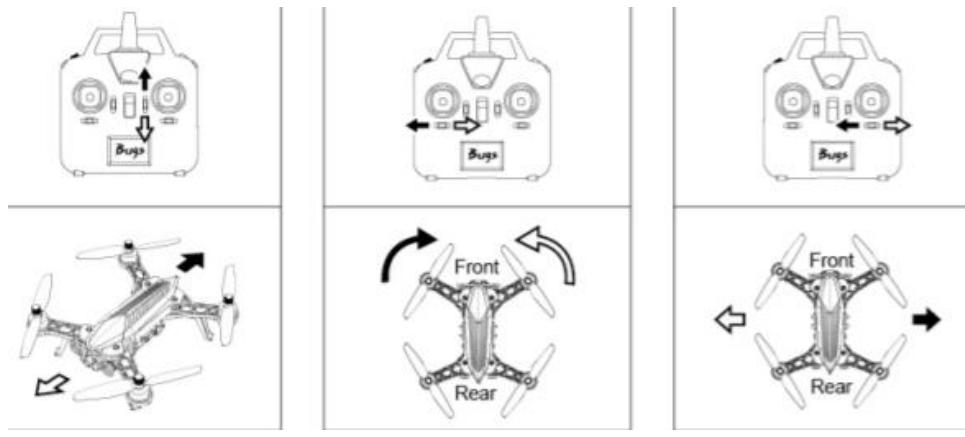


- g. Untuk memutar drone geser tuas kanan kearah kanan agar drone berputar kearah kanan dan kearah kiri agar drone berputar kearah kiri



- h. Apabila drone masih kurang stabil saat terbang maka perlu dilakukan proses trim.

Berikut cara trim pada drone:



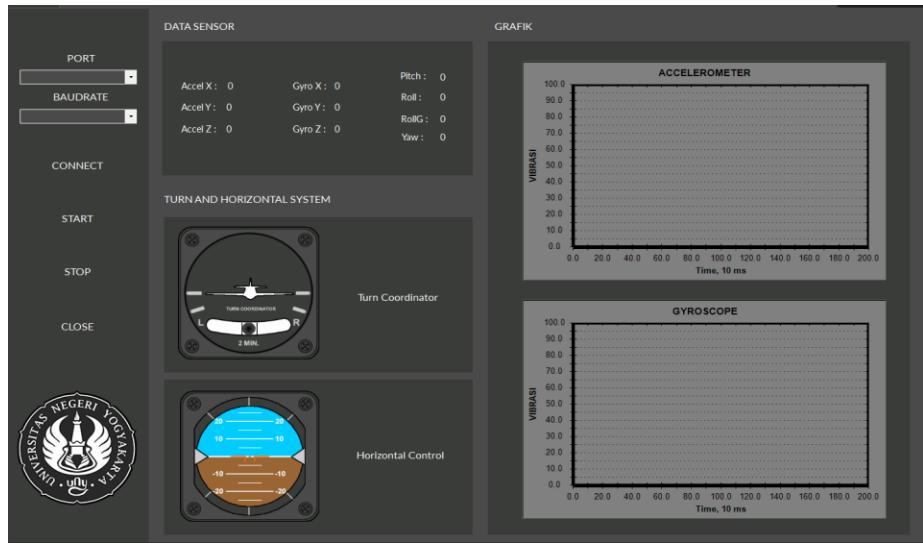
Trim maju/mundur

Trim putar kanan/kiri

Trim kanan/kiri

#### 4. Menjalankan GUI

- a. Buka GUI yang telah di install dan menampilkan tampilan sebagai berikut:



- b. Pilih PORT dan BAUDRATE yang digunakan hardware lalu tekan tombol CONNECT
- c. Setelah hardware siap terbang tekan tombol START dan terbangkan drone maka GUI akan menampilkan data Accelerometer dan Gyroscope.
- d. Apabila terjadi kesalahan ataupun masalah hubungi teknisi.

# **JOBSCHEET**

## **NAVIGASI DRONE GUNUNG BERAPI DENGAN ACCELEROMETER DAN GYROSCOPE**

**DISUSUN OLEH:**

**RIZKY GHÖZIA ULHAQ  
15518244007**

**DOSEN PEMBIMBING:  
HERLAMBANG SIGIT PURNOMO, M.SC.**





## FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA

### JOBSCHEET KENDALI DAN AKUISISI DATA

Semester 5

**JS 1: Accelerometer pada MPU6050  
dengan STM32**

4x60 menit

No.

Revisi: 02

Tgl: 5 Februari 2019

Hal 1 dari

### A. TUJUAN

Pada percobaan ini peserta didik diharapkan dapat mengetahui konfigurasi, dasar pemrograman, dan cara menampilkan nilai sensor accelerometer dengan STM32.

### B. ALAT DAN BAHAN

#### 1. Alat

- a. STM32
- b. Laptop/ Komputer
- c. Arduino IDE

#### 2. Bahan

- a. Sensor MPU6050

### C. KESELAMATAN KERJA

1. Sebelum praktik mahasiswa harus memahami tujuan dan kompetensi dari praktikum
2. Membaca dan memahami semua langkah kerja dari praktikum ini dengan cermat
3. Sebelum memberikan sumber listrik dan menyalakan semua unit praktikum harus diperiksa instruktur lebih dahulu
4. Pastikan dan lindungi diri anda dari efek kejutan listrik karena grounding listrik yang tidak sempurna
5. Perhatikan propeller dari drone ketika menyalakan drone

### D. MATERI

#### 1. STM32F103C8 atau Blue Pill

STM32F103C8 atau Blue Pill merupakan papan mikrokontroler berbasis STM32 yang dikembangkan oleh STMicroelectronics. Papan ini memiliki pin GPIO sebanyak 32 pin diantaranya (12-bit ADC), I2C, USART, dan SPI. Perbedaan signifikan STM32 dengan arduino adalah pada frekuensi clock sebesar 72 MHz, sedangkan arduino hanya 16 MHz

Dibuat oleh:

Dilarang memperbanyak sebagian atau seluruh isi dokumen  
tanpa ijin tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh:



# FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA

## **JOB SHEET KENDALI DAN AKUISISI DATA**

Semester 5

## **JS 1: Accelerometer pada MPU6050 dengan STM32**

4x60 menit

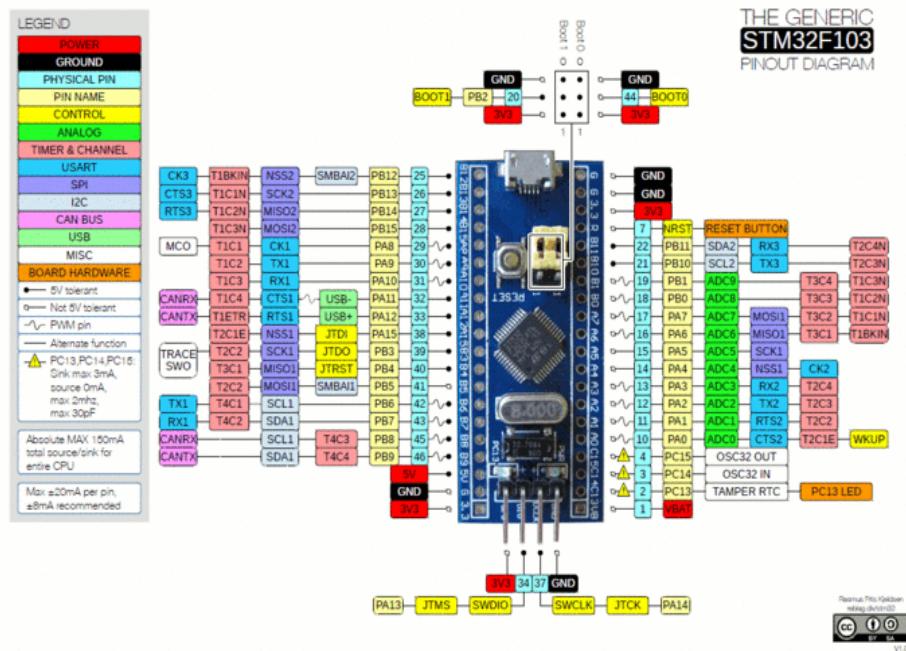
No

Revisi: 02

Tgl: 5 Februari 2019

Hal 1 dari

dengan kata lain STM32 4,5 kali lebih cepat dari arduino sehingga STM32 sangat cocok digunakan pada sensor maupun module yang membutuhkan pembacaan yang sangat cepat salah satunya adalah sensor MPU6050. STM32 juga dilengkapi dengan USB type B yang umumnya di temukan pada smarphone. Port USB ini bisa di manfaatkan sebagai *power supply* bahkan bisa digunakan untuk memprogram dengan metode bootloader. Berikut adalah pin GPIO STM32:



## 2. ACCELEROMETER

Dasar dari accelerometer yaitu Hukum Newton II yang mengatakan bahwa percepatan yang dialami suatu benda berbanding lurus terhadap gaya dan berbanding terbalik dengan massa benda yang bekerja. Ketika kecepatan suatu benda berubah maka benda tersebut mengalami akselerasi. Akselerasi atau yang sering disebut percepatan merupakan laju perubahan kecepatan terhadap waktu. Sama seperti kecepatan, percepatan juga terbagi menjadi dua yaitu percepatan rata-rata dan sesaat. Percepatan merupakan hasil dari turunan kecepatan terhadap waktu dan kecepatan didapatkan dari hasil turunan dari perpindahan terhadap waktu.

Dibuat oleh: Dilarang memperbanyak sebagian atau seluruh isi dokumen tanpa ijin tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta Diperiksa oleh:

**FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA****JOBSCHEET KENDALI DAN AKUISISI DATA**

Semester 5

**JS 1: Accelerometer pada MPU6050  
dengan STM32**

4x60 menit

No.

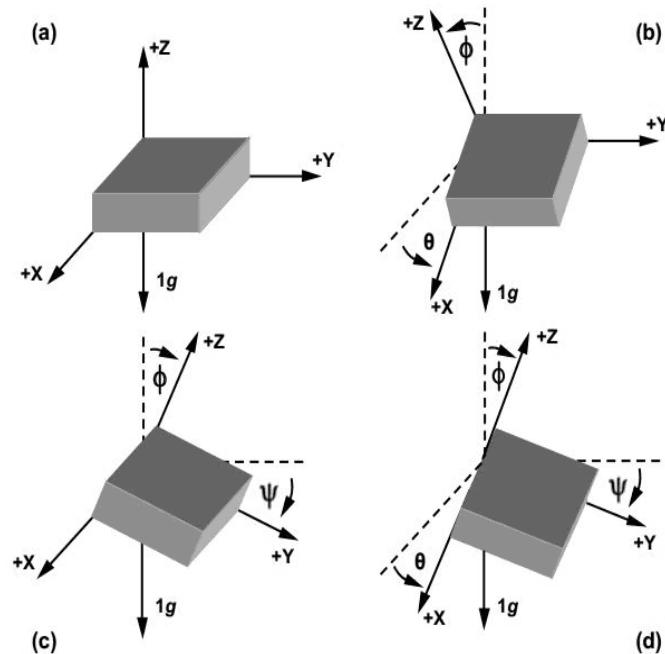
Revisi: 02

Tgl: 5 Februari 2019

Hal 1 dari

Fitur sensor accelerometer pada MPU6050 sebagai berikut:

- Digital output dengan 3 axis dengan jangkauan  $\pm 2g$ ,  $\pm 4g$ ,  $\pm 8g$ , dan  $\pm 16g$ .
- Memiliki satuan berupa g (gaya gravitasi)
- Terintegrasi ADC 16bit yang digunakan untuk mendapatkan output digital
- Ketika diletakan pada permukaan datar maka nilai X dan Y Axis yaitu 0g sedangkan Z axis bernilai +1g
- Memerlukan arus 500 micro ampere
- Orientation detection dan signaling
- Tap detection
- User programmable interrupts
- High-G interrupt
- User self-test



Dibuat oleh:

Dilarang memperbanyak sebagian atau seluruh isi dokumen  
tanpa ijin tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh:



# FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA

## **JOB SHEET KENDALI DAN AKUISISI DATA**

Semester 5

## JS 1: Accelerometer pada MPU6050 dengan STM32

4x60 menit

No

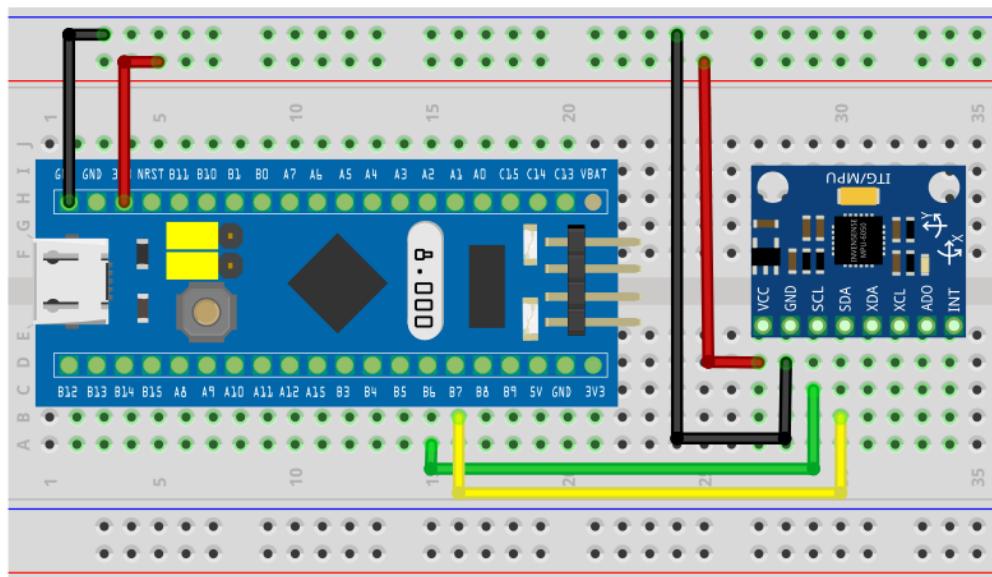
Revisi: 02

Tgl: 5 Februari 2019

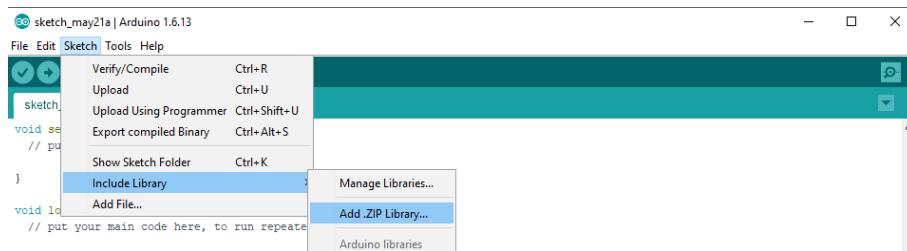
Hal 1 dari

## E. LANGKAH KERJA

1. Buat wiring hardware seperti gambar



2. Download library dari sensor MPU6050 versi Jrowberg
  3. Download library dari I2C
  4. Download Flash Loader Demostrator
  5. Setting STM32
    - a. Install aplikasi Flash Loader Demostrator
    - b. Hubungkan STM32 dengan USB ke laptop/computer
    - c. Pilih Port Name sesuai dengan COM USB dari STM32 lalu pilih Next dan tunggu beberapa saat
    - d. Flash telah berhasil dilakukan dan STM32 dapat digunakan seperti arduino
  6. Masukan library MPU6050 dan I2C dengan cara pilih **Sketch → Include Library → Add Zip Library** lalu **Open file MPU6050.Zip**



Dibuat oleh:	Dilarang memperbanyak sebagian atau seluruh isi dokumen tanpa ijin tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta	Diperiksa oleh:
--------------	--	-----------------

**FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA****JOBSCHEET KENDALI DAN AKUISISI DATA**

Semester 5

**JS 1: Accelerometer pada MPU6050  
dengan STM32**

4x60 menit

No. Revisi: 02 Tgl: 5 Februari 2019 Hal 1 dari

7. Hingga pada tampilan utama dari Arduino IDE muncul library yang telah dimasukan seperti gambar berikut

```
#include "I2Cdev.h" |  
#include "MPU6050_6Axis_MotionApps20.h"  
#if I2CDEV_IMPLEMENTATION == I2CDEV_ARDUINO_WIRE  
    #include "Wire.h"  
#endif
```

8. Pada void setup tuliskan program berikut

```
void setup() {  
    // put your setup code here, to run once:  
    Wire.begin();  
    Serial.begin(57600);  
    mpu.initialize();  
}
```

9. Pada void loop tuliskan program berikut

```
void loop() {  
    // put your main code here, to run repeatedly:  
    copy_data(); // menampilkan data pengiriman di serial monitor  
    AccelGyro();  
    delay(100);  
}
```

10. Buat void baru yaitu void AccelGyro dan ketikan program berikut didalamnya. Pada bagian ini berisi program perhitungan dari accelerometer.

```
void AccelGyro()  
{  
    mpu.getMotion6(&ax, &ay, &az);  
    //Program Accel & Gyro  
    Ax=((ax/257));  
    Ay=((ay/257))-1;  
    Az=((az/257))-96;  
  
    AccX = Ax;  
    AccY=Ay;  
    AccZ=Az;  
}
```

11. Buat void baru yaitu void copy data dan ketikan program berikut didalamnya. Bagian ini digunakan untuk menampilkan data pada Serial Monitor.

Dibuat oleh:

Dilarang memperbanyak sebagian atau seluruh isi dokumen  
tanpa ijin tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh:

**FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA****JOBSCHEET KENDALI DAN AKUISISI DATA**

Semester 5

**JS 1: Accelerometer pada MPU6050  
dengan STM32**

4x60 menit

No. Revisi: 02 Tgl: 5 Februari 2019 Hal 1 dari

```
void copy_data()
{
    Serial.print("<B4"); Serial.print(AccX);Serial.print("B4>");
    Serial.print("<B5"); Serial.print(AccY);Serial.print("B5>");
    Serial.print("<B6"); Serial.print(AccZ);Serial.print("B6>");
}
```

12. Setelah selesai lakukan Verify dan Run lalu lihat datanya pada serial monitor
13. Hubungkan rangkaian hardware dengan computer lalu setting PORT. Setelah itu verify program jika tidak ada error klik Run. Setelah selesai upload klik Serial Monitor dan atur Baudrate sesuai serial yang ditulis pada program.

**F. TUGAS**

Buatlah laporan dari pembelajaran job 1!

Dibuat oleh:

Dilarang memperbanyak sebagian atau seluruh isi dokumen  
tanpa ijin tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh:

	<b>FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA</b> <b>JOBSCHEET KENDALI DAN AKUISISI DATA</b>		
Semester 5	<b>JS 2: Gyroscope pada MPU6050 dengan STM32</b>		4x60 menit
No.	Revisi: 02	Tgl: 5 Februari 2019	Hal 1 dari

## A. TUJUAN

Pada percobaan ini peserta didik diharapkan dapat mengetahui konfigurasi, dasar pemrograman, dan cara menampilkan nilai sensor gyroscope dengan STM32.

## B. ALAT DAN BAHAN

1. Alat
  - a. STM32
  - b. Laptop/ Komputer
  - c. Arduino IDE
2. Bahan
  - a. Sensor MPU6050

## C. KESELAMATAN KERJA

1. Sebelum praktik mahasiswa harus memahami tujuan dan kompetensi dari praktikum
2. Membaca dan memahami semua langkah kerja dari praktikum ini dengan cermat
3. Sebelum memberikan sumber listrik dan menyalakan semua unit praktikum harus diperiksa instruktur telebih dahulu
4. Pastikan dan lindungi diri anda dari efek kejutan listrik karena grounding listrik yang tidak sempurna
5. Perhatikan propeller dari drone ketika menyalakan drone

## D. MATERI

### GYROSCOPE

Gyroscope memiliki output yang ekuivalen terhadap kecepatan sudut dari arah sumbu x yang nantinya akan menjadi sudut phi (roll), arah sumbu y akan menghasilkan sudut theta (pitch) dan sumbu z akan menjadi sudut psi (yaw). Data yang masuk kedalam mikrokontroler berupa rad/s dan akan dikonversi menjadi deg/s. Terdapat banyak versi dari persamaan yang digunakan dalam pengolahan data gyroscope.

Dibuat oleh:	Dilarang memperbanyak sebagian atau seluruh isi dokumen tanpa ijin tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta	Diperiksa oleh:
--------------	--	-----------------



## FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA

### JOBSCHEET KENDALI DAN AKUISISI DATA

Semester 5

JS 2: Gyroscope pada MPU6050  
dengan STM32

4x60 menit

No.

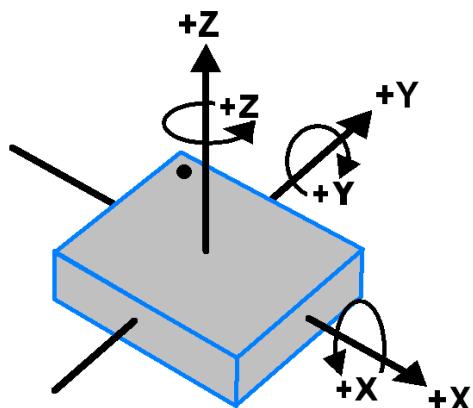
Revisi: 02

Tgl: 5 Februari 2019

Hal 1 dari

Fitur sensor gyroscope pada MPU6050 sebagai berikut:

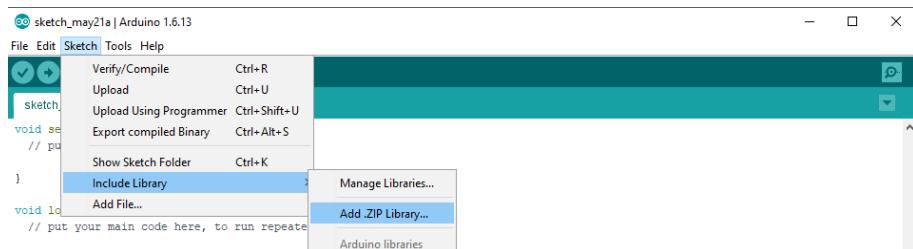
- Digital output X-, Y-, dan Z- axis sudut kemiringan
- Rentang user-programmable  $\pm 250, \pm 500, \pm 1000$ , dan  $\pm 2000$  derajat per detik
- Terintegrasi ADC 16bit
- Operating current 3.6mA
- User self-test
- Digitally-programmable low-pass filter
- Improved low-frequency noise performance



## E. LANGKAH KERJA

### 1. Pemrograman Gyroscope

- a. Buka aplikasi Arduino IDE
- b. Masukan library dari MPU6050 dan I2C dengan cara pilih Sketch → Include Library → Add Zip Library lalu Open file **MPU6050.Zip**



Dibuat oleh:

Dilarang memperbanyak sebagian atau seluruh isi dokumen tanpa ijin tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh:

**FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA****JOBSCHEET KENDALI DAN AKUISISI DATA**

Semester 5

**JS 2: Gyroscope pada MPU6050  
dengan STM32**

4x60 menit

No.

Revisi: 02

Tgl: 5 Februari 2019

Hal 1 dari

- c. Hingga pada tampilan utama dari Arduino IDE muncul library yang telah dimasukan seperti gambar berikut

```
#include "I2Cdev.h" |  
#include "MPU6050_6Axis_MotionApps20.h"  
#if I2CDEV_IMPLEMENTATION == I2CDEV_ARDUINO_WIRE  
    #include "Wire.h"  
#endif
```

- d. Buat inisialisasi MPU6050

```
MPU6050 mpu;  
int16_t gx, gy, gz;  
int GyX, GyY, GyZ;  
int Gx,Gy,Gz;
```

- e. Pada void setup tuliskan program berikut

```
void setup() {  
    // put your setup code here, to run once:  
    Wire.begin();  
    Serial.begin(57600);  
    mpu.initialize();  
}
```

- f. Pada void loop tuliskan program berikut

```
void loop() {  
    // put your main code here, to run repeatedly:  
    copy_data();      // menampilkan data pengiriman di serial monitor  
    AccelGyro();  
    delay(100);  
}
```

- g. Buat void baru yaitu void AccelGyro dan ketikan program berikut didalamnya. Pada bagian ini berisi program perhitungan dari gyroscope.

Dibuat oleh:

Dilarang memperbanyak sebagian atau seluruh isi dokumen tanpa ijin tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh:



## FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA

### JOBSCHEET KENDALI DAN AKUISISI DATA

Semester 5

JS 2: Gyroscope pada MPU6050  
dengan STM32

4x60 menit

No.

Revisi: 02

Tgl: 5 Februari 2019

Hal 1 dari

```
void AccelGyro()
{
    mpu.getMotion6(&ax, &ay, &az);
    //Program Accel & Gyro
    Gx=(((gx/257))/4);
    Gy=(((gy/257))/4);
    Gz=(((gz/257))/4);

    GyX=Gx;
    GyY=Gy;
    GyZ=Gz;
}
```

- h. Buat void baru yaitu void copy data dan ketikan program berikut didalamnya. Bagian ini digunakan untuk menampilkan data pada Serial Monitor.

```
void copy_data()
{
    Serial.print("<B7"); Serial.print(GyX);Serial.print("B7>");
    Serial.print("<B8"); Serial.print(GyY);Serial.print("B8>");
    Serial.print("<B9"); Serial.print(GyZ);Serial.print("B9>");
}
```

- i. Setelah selesai lakukan Verify dan Run lalu lihat datanya pada serial monitor.

## F. TUGAS

Buatlah laporan dari jobsheet yang telah dilakukan!

Dibuat oleh:

Dilarang memperbanyak sebagian atau seluruh isi dokumen  
tanpa ijin tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh:



## FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA

### JOBSCHEET KENDALI DAN AKUISISI DATA

Semester 5	<b>JS 3: Raw Pitch Yaw pada MPU6050 dengan STM32 dan GUI Visual Studio</b>	4x60 menit	
No.	Revisi: 02	Tgl: 5 Februari 2019	Hal 1 dari

#### A. TUJUAN

Pada percobaan ini peserta didik diharapkan dapat mengetahui konfigurasi, dasar pemrograman, dan cara menampilkan nilai sensor accelerometer dan gyroscope yang diolah menjadi Raw Pitch Yaw dengan STM32 lalu menampilkan pada GUI yang telah dibuat.

#### B. ALAT DAN BAHAN

1. Alat
  - a. STM32
  - b. Laptop/ Komputer
  - c. Visual Studio
  - d. Arduino IDE
2. Bahan
  - a. Sensor MPU6050

#### C. KESELAMATAN KERJA

1. Sebelum praktik mahasiswa harus memahami tujuan dan kompetensi dari praktikum
2. Membaca dan memahami semua langkah kerja dari praktikum ini dengan cermat
3. Sebelum memberikan sumber listrik dan menyalaikan semua unit praktikum harus diperiksa instruktur telebih dahulu
4. Pastikan dan lindungi diri anda dari efek kejutan listrik karena grounding listrik yang tidak sempurna
5. Perhatikan propeller dari drone ketika menyalaikan drone

#### D. MATERI

##### 1. ROLL YAW PITCH

Pola Terbang *Quadcopter* untuk manuver atau bergerak mempunyai aturan khusus berdasarkan pergerakan rotor pada keempat sisinya.

Dibuat oleh:

Dilarang memperbanyak sebagian atau seluruh isi dokumen tanpa ijin tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh:



## FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA

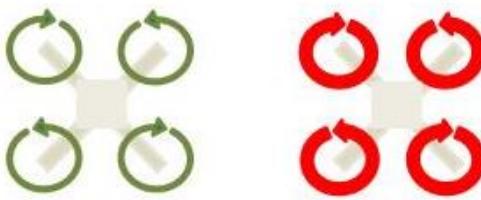
### JOBSCHEET KENDALI DAN AKUISISI DATA

Semester 5	JS 3: Raw Pitch Yaw pada MPU6050 dengan STM32 dan GUI Visual Studio	4x60 menit	
No.	Revisi: 02	Tgl: 5 Februari 2019	Hal 1 dari

#### a. Throttle

Gerak menaikan atau menurunkan ketinggian dengan mengatur kecepatan keempat rotor sama besar. *Quadcopter* akan bergerak searah sumbu-z dengan mengacu pada koordinat *Quadcopter*. Arah putar masing-masing rotor seperti pada gambar 1.

#### Throttle control

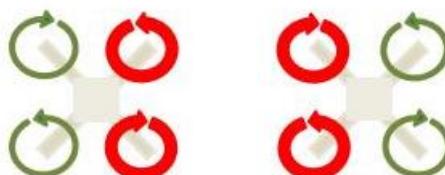


Move down                  Move up  
Gambar 1. Gerak Throttle

#### b. Roll

Gerak manuver kesamping dengan cara meningkatkan kecepatan putar bagian kiri atau kanan *Quadcopter* dengan bersamaan. Ilustrasi gerakan *Roll* pada gerak rotasi *Quadcopter* seperti gambar 2.

#### Roll control



Bend left                  Bend Right  
Gambar 2. Gerak Roll

#### c. Pitch

Gerak *Pitch* hampir sama dengan gerak *Roll*. Penerapan *Pitch* pada manuver *Quadcopter* adalah dengan meningkatkan atau menurunkan kecepatan putar rotor bagian depan atau belakang secara bersama dan kecepatan sama.

Dibuat oleh:

Dilarang memperbanyak sebagian atau seluruh isi dokumen tanpa ijin tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh:



## FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA

### JOBSCHEET KENDALI DAN AKUISISI DATA

Semester 5

JS 3: Raw Pitch Yaw pada MPU6050  
dengan STM32 dan GUI Visual Studio

4x60 menit

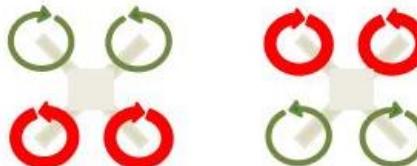
No.

Revisi: 02

Tgl: 5 Februari 2019

Hal 1 dari

#### Pitch control



Move forward

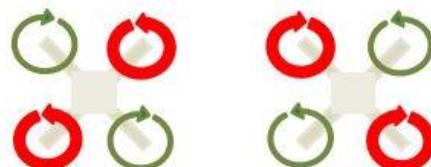
Move backward

Gambar 13. Gerak *Pitch*

#### d. *Yaw*

Gerakan *Yaw* dilakukan dengan cara meningkatkan kecepatan putar rotor dengan posisi bersilangan secara bersama-sama. Tujuan gerak ini adalah mendapatkan manuver *Quadcopter* berputar searah atau berlawanan jarum jam. Ilustrasi untuk gerak *Yaw* pada *Quadcopter* dapat dilihat pada gambar 14.

#### Yaw control



Rotate left

Rotate right

Gambar 14. Gerak *Yaw*

## 2. GUI

*Graphic User Interface* (GUI) merupakan antarmuka atau tampilan elektronik berupa grafis suatu aplikasi yang dilihat dan sebagai media komunikasi dengan pengguna. GUI yang baik dapat mempermudah pengguna dengan memberikan tampilan yang konsisten dan kontrol yang intuitif seperti tombol, slider, menu, list boxes, dan sejenisnya. Sehingga pengguna dapat berkomunikasi dengan mesin dengan baik dan benar. Menurut Mauladi dkk (2016: 64) GUI

Dibuat oleh:

Dilarang memperbanyak sebagian atau seluruh isi dokumen tanpa ijin tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh:



## FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA

### JOBSCHEET KENDALI DAN AKUISISI DATA

Semester 5	JS 3: Raw Pitch Yaw pada MPU6050 dengan STM32 dan GUI Visual Studio	4x60 menit	
No.	Revisi: 02	Tgl: 5 Februari 2019	Hal 1 dari

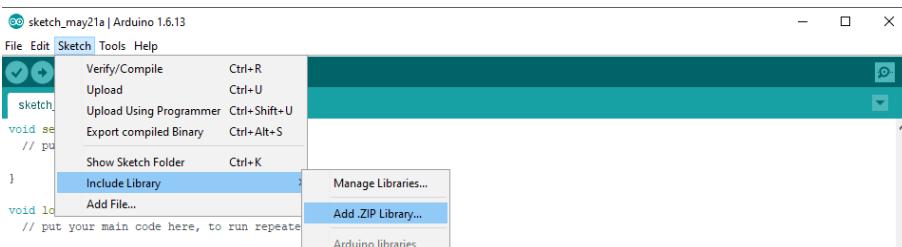
merupakan tipe antarmuka yang digunakan pengguna untuk berinteraksi dengan sistem operasi melalui grafik, ikon, menu dan perangkat petunjuk seperti mouse atau track ball.

Pemrograman berbasis grafis akan menggunakan tool GUI editor yang telah memiliki *Integrated Development Environment* (IDE) sehingga pemrograman grafis relatif mudah. Pembuatan GUI dengan tool yang sudah IDE lebih mudah karena dapat dilihat secara visual dan melinatkan unsur desain pada pemrogramannya. Selain itu penggunaan berbagai komponen siap pakai yang bersifat visual maupun non-visual. Penggunaannya melainkan hanya *drag* dan *drop* lalu diprogram sedemikian rupa. Salah satu tool yang biasanya digunakan programmer untuk mengembangkan GUI yaitu Visual Studio.

## E. LANGKAH KERJA

### 1. Pemrograman Menampilkan Raw Pitch Yaw

- Buka aplikasi Arduino IDE
- Masukan library dari MPU6050 dan I2C dengan cara pilih **Sketch→Include Library→Add Zip Library** lalu **Open file MPU6050.Zip**



- Hingga pada tampilan utama dari Arduino IDE muncul library yang telah dimasukan seperti gambar berikut

Dibuat oleh:

Dilarang memperbanyak sebagian atau seluruh isi dokumen tanpa ijin tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh:



## FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA

### JOBSCHEET KENDALI DAN AKUISISI DATA

Semester 5	<b>JS 3: Raw Pitch Yaw pada MPU6050 dengan STM32 dan GUI Visual Studio</b>	4x60 menit	
No.	Revisi: 02	Tgl: 5 Februari 2019	Hal 1 dari

```
#include "I2Cdev.h" |
#include "MPU6050_6Axis_MotionApps20.h"
#if I2CDEV_IMPLEMENTATION == I2CDEV_ARDUINO_WIRE
    #include "Wire.h"
#endif
```

d. Buat inisialisasi MPU6050

```
MPU6050 accelgyro;
int16_t ax, ay, az;
int16_t gx, gy, gz;
int16_t AccX, AccY, AccZ, GyX, GyY, GyZ;
int Ax, Ay, Az, Gx, Gy, Gz;

int vall;
int val2;
int val3;
int val4;
int val5;
int val6;
int prevVall;
int prevVal2;
int prevVal3;
int prevVal4;
int prevVal5;
int prevVal6;
```

e. Pada void setup tuliskan program berikut

```
void setup()
{
    #if I2CDEV_IMPLEMENTATION == I2CDEV_ARDUINO_WIRE
        Wire.begin();
    #elif I2CDEV_IMPLEMENTATION == I2CDEV_BUILTIN_FASTWIRE
        Fastwire::setup(400, true);
    #endif

    accelgyro.initialize();
    Wire.begin();
    Wire.beginTransmission(address); // I2C Stuff
    error = Wire.endTransmission();
    Serial.begin(57600); //setup serial monitor
    Serial2.begin(57600); //setup serial telemetry
}
```

f. Pada void loop tuliskan program berikut

```
void loop()
{
    AccelGyro();
    copy_data(); // menampilkan data pengiriman di serial monitor
    kirim_data();

    delay(1000);
}
```

Dibuat oleh:

Dilarang memperbanyak sebagian atau seluruh isi dokumen  
tanpa ijin tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh:

**FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA****JOBSCHEET KENDALI DAN AKUISISI DATA**

Semester 5	<b>JS 3: Raw Pitch Yaw pada MPU6050 dengan STM32 dan GUI Visual Studio</b>	4x60 menit	
No.	Revisi: 02	Tgl: 5 Februari 2019	Hal 1 dari

- g. Buat void baru yaitu void AccelGyro dan ketikan program berikut didalamnya. Pada bagian ini berisi program perhitungan dari raw pitch yaw.

```
void AccelGyro()
{
    accelgyro.getMotion6(&ax, &ay, &az, &gx, &gy, &gz);
    Ax=((ax/257));
    Ay=((ay/257))-1;
    Az=((az/257))-96;

    Gx=(((gx/257))/4);
    Gy=(((gy/257))/4);
    Gz=(((gz/257))/4);

    AccX = Ax;
    AccY=Ay;
    AccZ=Az;

    GyX=Gx;
    GyY=Gy;
    GyZ=Gz;

    // Program YPR
    val1 = map(ax, -17000, 17000, -179, 179);
    if (val1 != prevVal1)
    {
        prevVal1 = val1;
    }
    val2 = map(ay, -17000, 17000, -45, 45);
    if (val2 != prevVal2)
    {
        prevVal2 = val2;
    }
    val3= map(gx, -17000, 17000, -60, 60);
    if(val3 != prevVal3)
    {
        prevVal3= val3;
    }
    val4= map(gz, -17000, 17000, -20, 20);
    if(val4 != prevVal4)
    {
        prevVal4= val4;
    }
}
```

- h. Buat void baru yaitu void copy data dan ketikan program berikut didalamnya. Bagian ini digunakan untuk menampilkan data pada Serial Monitor.

Dibuat oleh:

Dilarang memperbanyak sebagian atau seluruh isi dokumen  
tanpa ijin tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh:



## FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA

### JOBSCHEET KENDALI DAN AKUISISI DATA

Semester 5

JS 3: Raw Pitch Yaw pada MPU6050  
dengan STM32 dan GUI Visual Studio

4x60 menit

No.

Revisi: 02

Tgl: 5 Februari 2019

Hal 1 dari

```
void copy_data()
{
    //pengiriman data Accel & Gyro
    Serial.print("<B0>"); Serial.print(prevVal1);Serial.print("B0>");
    Serial.print("<B1>"); Serial.print(prevVal2);Serial.print("B1>");
    Serial.print("<B2>"); Serial.print(prevVal3);Serial.print("B2>");
    Serial.print("<B3>"); Serial.print(prevVal4);Serial.print("B3>");
    Serial.print("<B4>"); Serial.print(AccX);Serial.print("B4>");
    Serial.print("<B5>"); Serial.print(AccY);Serial.print("B5>");
    Serial.print("<B6>"); Serial.print(AccZ);Serial.print("B6>");
    Serial.print("<B7>"); Serial.print(GyX);Serial.print("B7>");
    Serial.print("<B8>"); Serial.print(GyY);Serial.print("B8>");
    Serial.print("<B9>"); Serial.print(GyZ);Serial.print("B9>");|}
}
```

- i. Buat void baru yaitu void kirim\_data dan ketikan program berikut didalanya. Bagian ini digunakan untuk mengirim data dari STM32 untuk ditampilkan pada GUI.

```
void kirim_data()
{
    Serial2.print("<B0>"); Serial2.print(prevVal1);Serial2.print("B0>");
    Serial2.print("<B1>"); Serial2.print(prevVal2);Serial2.print("B1>");
    Serial2.print("<B2>"); Serial2.print(prevVal3);Serial2.print("B2>");
    Serial2.print("<B3>"); Serial2.print(prevVal4);Serial2.print("B3>");
    Serial2.print("<B4>"); Serial2.print(AccX);Serial2.print("B4>");
    Serial2.print("<B5>"); Serial2.print(AccY);Serial2.print("B5>");
    Serial2.print("<B6>"); Serial2.print(AccZ);Serial2.print("B6>");
    Serial2.print("<B7>"); Serial2.print(GyX);Serial2.print("B7>");
    Serial2.print("<B8>"); Serial2.print(GyY);Serial2.print("B8>");
    Serial2.print("<B9>"); Serial2.print(GyZ);Serial2.print("B9>");|}
}
```

- j. Setelah selesai lakukan Verify dan Run lalu lihat datanya pada serial monitor.

## 2. Pembuatan GUI dengan Visual Studio

Buatlah GUI yang digunakan untuk menggantikan serial monitor dalam menampilkan data yang diperoleh dari sensor MPU6050 yaitu data accelerometer, gyroscope, dan raw pitch yaw.

- a. Buka Visual Studio
- b. Pilih New Project → Pilih Windows Form Application Visual C# → ubah Name dan Location sesuai keinginan → OK

Dibuat oleh:

Dilarang memperbanyak sebagian atau seluruh isi dokumen  
tanpa ijin tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh:



## FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA

### JOBSCHEET KENDALI DAN AKUISISI DATA

Semester 5

JS 3: Raw Pitch Yaw pada MPU6050  
dengan STM32 dan GUI Visual Studio

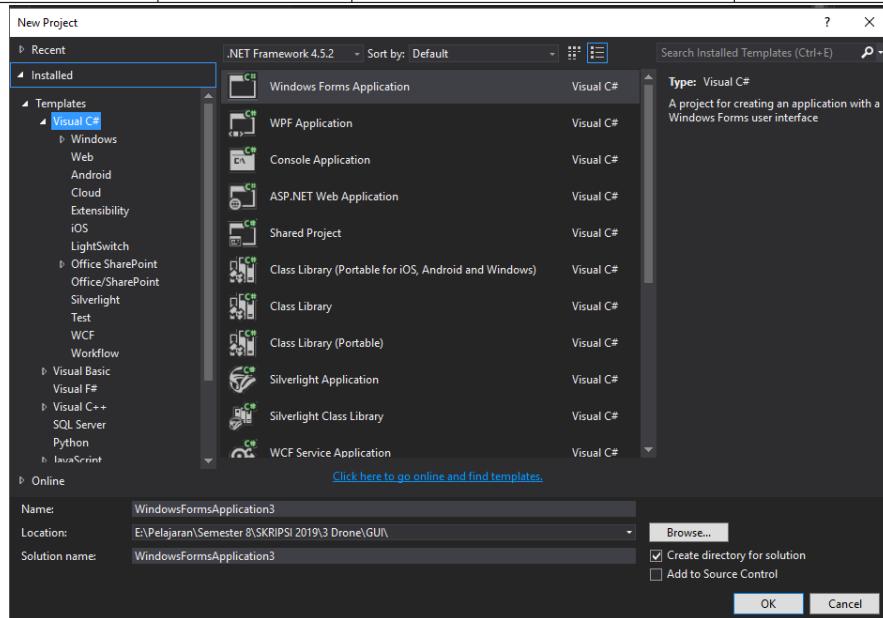
4x60 menit

No.

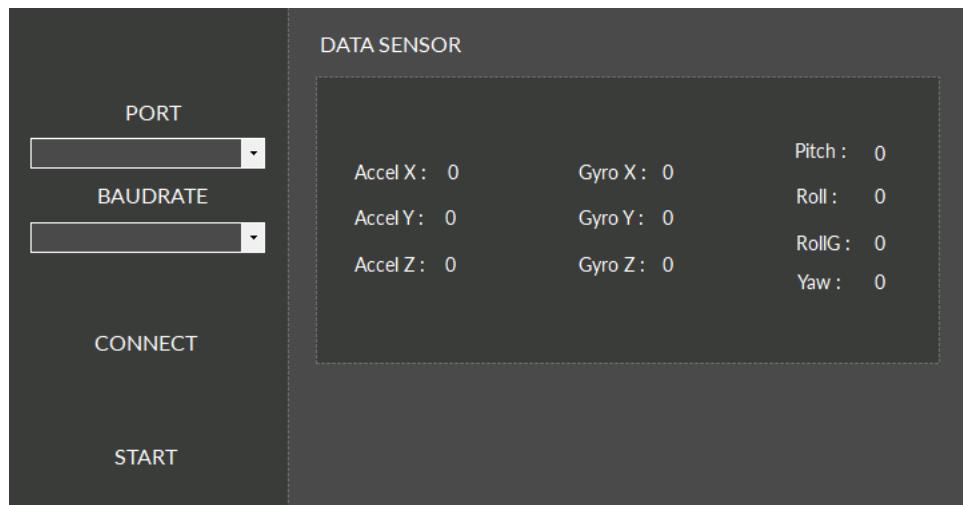
Revisi: 02

Tgl: 5 Februari 2019

Hal 1 dari



- c. Lalu buat tampilan seperti gambar berikut dengan menambahkan Item melalui Toolbox dengan drag item pada layer desain



- d. Atur layout, nama dan alamat setiap item melalui properties.

Dibuat oleh:

Dilarang memperbanyak sebagian atau seluruh isi dokumen  
tanpa ijin tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh:



## FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA

## JOBSHEET KENDALI DAN AKUISISI DATA

Semester 5

JS 3: Raw Pitch Yaw pada MPU6050  
dengan STM32 dan GUI Visual Studio

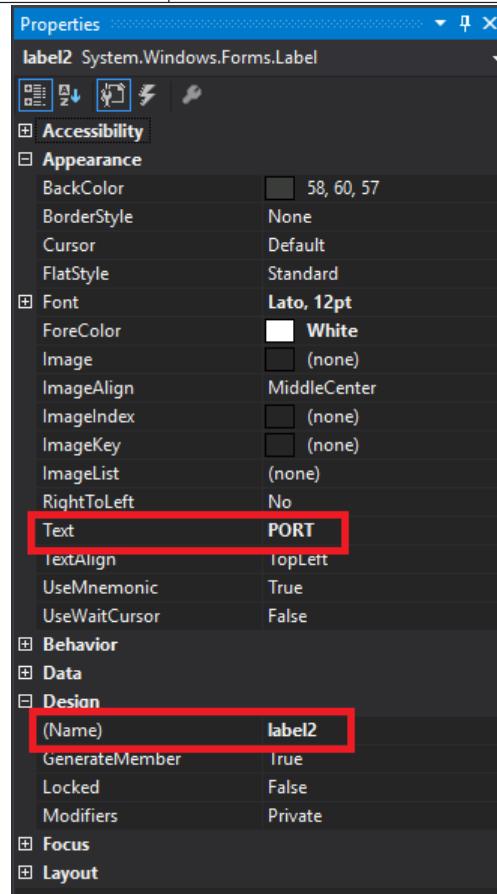
4x60 menit

No.

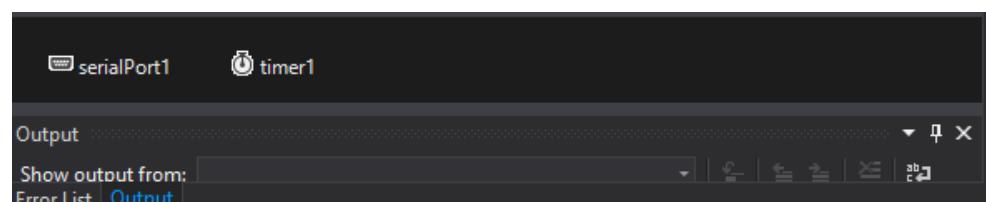
Revisi: 02

Tgl: 5 Februari 2019

Hal 1 dari



- e. Tambahkan SerialPort dan timer pada layer dengan cara drag, dan icon akan berpindah dibawah.



- f. Buat Public static string seperti gambar berikut

Dibuat oleh:	Dilarang memperbanyak sebagian atau seluruh isi dokumen tanpa ijin tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta	Diperiksa oleh:
--------------	---	-----------------



## FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA

## JOBSHEET KENDALI DAN AKUISISI DATA

Semester 5	JS 3: Raw Pitch Yaw pada MPU6050 dengan STM32 dan GUI Visual Studio	4x60 menit	
No.	Revisi: 02	Tgl: 5 Februari 2019	Hal 1 dari

```
public static string GetStringBetweenCharacters(string input, string charFrom, string charTo)
{
    int chrFrom = charFrom.Length;
    int chrTo = charTo.Length;
    int posFrom = input.IndexOf(charFrom);
    if (posFrom != -1) //if found char
    {
        int posTo = input.IndexOf(charTo, posFrom + 1);
        if (posTo != -1) //if found char
        {
            try { return input.Substring(posFrom + chrFrom, posTo - posFrom - chrTo); }
            catch { }
        }
        return string.Empty;
    }
    1 reference
    public DemoWinow()
    {
        InitializeComponent();
    }
}
```

- g. Setelah desain jadi lalu buka bagian program dengan cara klik dua kali pada layer desain hingga masuk tab baru. Lalu ketikan program berikut.

```
1 reference
public DemoWinow()
{
    InitializeComponent();
}
```

- h. Pada NamaFileMu\_Load tuliskan program berikut. Program ini untuk menampilkan pilihan Baudrate yang dapat digunakan.

```
1 reference
private void DemoWinow_Load(object sender, EventArgs e)
{
    //BAUDRATE
    cbBaudr.Items.Add("9600");
    cbBaudr.Items.Add("38400");
    cbBaudr.Items.Add("57600");
    cbBaudr.Items.Add("115200");
}
```

- i. Pada bagian desain clik 2x ComboBox PORT lalu ketikan program berikut

```
1 reference
private void cbPort_DropDown(object sender, EventArgs e)
{
    cbPort.Items.Clear();
    cbPort.Items.AddRange(System.IO.Ports.SerialPort.GetPortNames());
}
```

- j. Kembali ke halaman desain, klik 2x Button START lalu ketikan program berikut ini

Dibuat oleh:

Dilarang memperbanyak sebagian atau seluruh isi dokumen tanpa ijin tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh:



## FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA

### JOBSCHEET KENDALI DAN AKUISISI DATA

Semester 5

JS 3: Raw Pitch Yaw pada MPU6050  
dengan STM32 dan GUI Visual Studio

4x60 menit

No.

Revisi: 02

Tgl: 5 Februari 2019

Hal 1 dari

```
1 reference
private void btStart_Click(object sender, EventArgs e)
{
    serialPort1.WriteLine("a");
    timer1.Start();
    btnStop.Enabled = true;

    graphVibrasiACC();
    graphVibrasiGYRO();
}
```

- k. Kembali ke halaman desain, klik 2x Button CONNECT lalu ketikan program berikut ini

```
1 reference
private void btnCon_Click(object sender, EventArgs e)
{
    if (btnCon.Text == "CONNECT") {
        if (serialPort1.IsOpen == false)
        {
            serialPort1.PortName = cbPort.Text;
            serialPort1.BaudRate = int.Parse(cbBaudr.Text);      //mengubah nilai string jadi integer

            try
            {
                serialPort1.Open();
                btnCon.Text = "DISCONNECT";
            }
            catch (Exception Ghozia)                      //memunculkan kalok ada error
            {
                MessageBox.Show("Serial Error" + Ghozia);
            }
        }
        else
        {
            serialPort1.Close();
            btnCon.Text = "CONNECT";
        }
    }
}
```

- l. Kembali ke halaman desain, klik 2x Timer lalu ketikan proram berikut ini

Dibuat oleh:

Dilarang memperbanyak sebagian atau seluruh isi dokumen  
tanpa ijin tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh:



## FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA

### JOBSCHEET KENDALI DAN AKUISISI DATA

Semester 5	JS 3: Raw Pitch Yaw pada MPU6050 dengan STM32 dan GUI Visual Studio	4x60 menit	
No.	Revisi: 02	Tgl: 5 Februari 2019	Hal 1 dari

```
1 reference
private void timer1_Tick(object sender, EventArgs e)
{
    //Menerima data dari arduino dan ngubah label di GUI
    DataMasuk = serialPort1.ReadExisting();
    if (GetStringBetweenCharacters(DataMasuk, "<B4", "B4>") != "")
        AccX = GetStringBetweenCharacters(DataMasuk, "<B4", "B4>");
    lbAccX.Text = AccX;
    if (GetStringBetweenCharacters(DataMasuk, "<B5", "B5>") != "")
        AccY = GetStringBetweenCharacters(DataMasuk, "<B5", "B5>");
    lbAccY.Text = AccY;
    if (GetStringBetweenCharacters(DataMasuk, "<B6", "B6>") != "")
        AccZ = GetStringBetweenCharacters(DataMasuk, "<B6", "B6>");
    lbAccZ.Text = AccZ;
    if (GetStringBetweenCharacters(DataMasuk, "<B7", "B7>") != "")
        GyroX = GetStringBetweenCharacters(DataMasuk, "<B7", "B7>");
    lbGyroX.Text = GyroX;
    if (GetStringBetweenCharacters(DataMasuk, "<B8", "B8>") != "")
        GyroY = GetStringBetweenCharacters(DataMasuk, "<B8", "B8>");
    lbGyroY.Text = GyroY;
    if (GetStringBetweenCharacters(DataMasuk, "<B9", "B9>") != "")
        GyroZ = GetStringBetweenCharacters(DataMasuk, "<B9", "B9>");
    lbGyroZ.Text = GyroZ;

    if (GetStringBetweenCharacters(DataMasuk, "<B0", "B0>") != "")
        lbl_Pitch.Text = GetStringBetweenCharacters(DataMasuk, "<B0", "B0>");
    if (GetStringBetweenCharacters(DataMasuk, "<B1", "B1>") != "")
        lbl_Roll.Text = GetStringBetweenCharacters(DataMasuk, "<B1", "B1>");
    if (GetStringBetweenCharacters(DataMasuk, "<B2", "B2>") != "")
        lbl_Rollg.Text = GetStringBetweenCharacters(DataMasuk, "<B2", "B2>");
    if (GetStringBetweenCharacters(DataMasuk, "<B3", "B3>") != "")
        lbl_Yaw.Text = GetStringBetweenCharacters(DataMasuk, "<B3", "B3>");

    Pitch = Convert.ToInt32(lbl_Pitch.Text); //ax
    Roll = Convert.ToInt32(lbl_Roll.Text); //ay
    Rollg = Convert.ToInt32(lbl_Rollg.Text); //gx
    Yaw = Convert.ToInt32(lbl_Yaw.Text); //gy
```

- m. Setelah semua komponen diprogram selanjutnya klik compile jika berhasil maka silahkan di Run dan dijalankan.

## F. TUGAS

Buatlah laporan dari jobsheet yang telah dilakukan!

Dibuat oleh:

Dilarang memperbanyak sebagian atau seluruh isi dokumen  
tanpa ijin tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh:



## FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA

### JOBSHEET KENDALI DAN AKUISISI DATA

Semester 5

**JS 4: MPU6050 menggunakan STM32 dan GUI**  
Visual Studio

4x60 menit

No.

Revisi: 02

Tgl: 5 Februari 2019

Hal 1 dari

#### A. TUJUAN

Pada percobaan ini peserta didik diharapkan dapat mengetahui konfigurasi, dasar pemrograman, dan cara menampilkan nilai MPU6050 menggunakan STM32 lalu menampilkan pada GUI yang telah dibuat.

#### B. ALAT DAN BAHAN

##### 1. Alat

- a. Drone
- b. Remote Control
- c. STM32
- d. Laptop/ Komputer
- e. Visual Studio
- f. Arduino IDE

##### 2. Bahan

- a. Sensor MPU6050

#### C. KESELAMATAN KERJA

1. Sebelum praktik mahasiswa harus memahami tujuan dan kompetensi dari praktikum
2. Membaca dan memahami semua langkah kerja dari praktikum ini dengan cermat
3. Sebelum memberikan sumber listrik dan menyalaikan semua unit praktikum harus diperiksa instruktur telebih dahulu
4. Pastikan dan lindungi diri anda dari efek kejutan listrik karena *grounding* listrik yang tidak sempurna
5. Perhatikan propeller dari drone ketika menyalaikan drone

#### D. MATERI

##### MPU6050

Berdasarkan datasheet sensor MPU6050 adalah perangkat sensor yang terdapat 3-axis accelerometer (sensor percepatan), 3-axis gyroscope (sensor keseimbangan) atau yang dikenal dengan 6DOF (*Degrees of Freedom*), suhu, dan magnetometer. Sensor MPU6050 merupakan salah

Dibuat oleh:

Dilarang memperbanyak sebagian atau seluruh isi dokumen tanpa ijin tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh:

**FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA****JOBSHEET KENDALI DAN AKUISISI DATA**

Semester 5

**JS 4: MPU6050 menggunakan STM32 dan GUI  
Visual Studio**

4x60 menit

No.

Revisi: 02

Tgl: 5 Februari 2019

Hal 1 dari

satu jenis alat elektronik yang digunakan sebagai pengukur inersia atau yang disebut dengan Inertial Measurement Unit (IMU) yang dapat mengukur kecepatan, orientasi, dan gaya gravitasi. Nilai yang dihasilkan sensor didapat dari gerakan tiga sumbu yaitu x, y, dan z. Sensor ini mampu membaca kemiringan sudut berdasarkan data sensor yang termuat pada modul MPU6050. Akses modul sensor ini menggunakan jalur data I2C.

Part/ Item	MPU6050	MPU6000
VDD	2.375V-3.46V	2.375V-3.46V
VLOGIC	1.71V to VDD	n/a
Serial Interface Supported	I2C	I2C, SPI
Pin 8	VLOGIC	/CS
Pin 9	AD0	AD0/ SD0
Pin 23	SCL	SCL/ SCLK
Pin 24	SDA	SDA/ SDI



Gambar 9. Modul MPU6050

(Sumber: <https://playgrond.arduino.cc>)

Dibuat oleh:

Dilarang memperbanyak sebagian atau seluruh isi dokumen  
tanpa ijin tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh:



## FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA

### JOBSHEET KENDALI DAN AKUISISI DATA

Semester 5

JS 4: MPU6050 menggunakan STM32 dan GUI  
Visual Studio

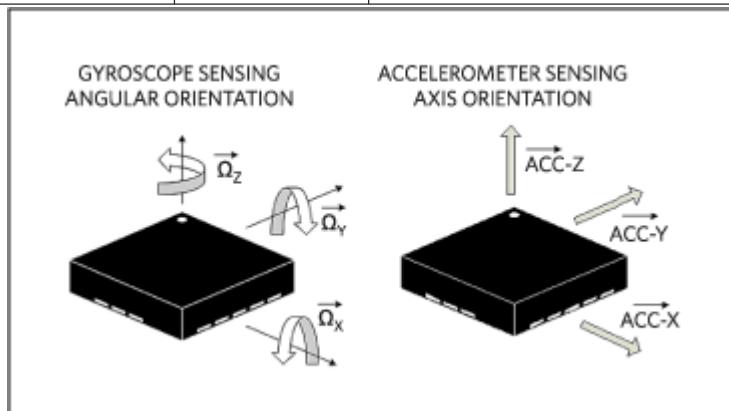
4x60 menit

No.

Revisi: 02

Tgl: 5 Februari 2019

Hal 1 dari



Perbedaan Accelerometer dan Gyroscope

### E. LANGKAH KERJA

1. Lakukan persiapan seperti pada jobsheet sebelumnya ketika akan menggunakan sensor MPU6050.
  - a. Wiring hardware
  - b. Masukan library yang dibutuhkan
  - c. Setting STM32 jika perlu
2. Download GUI dari Avionic Instrument melalui website <https://www.codeproject.com/Articles/27411/C-Avionic-Instrument-Controls>
3. Download library dari Zedgraph
4. Pembuatan Program Arduino
  - a. Buka aplikasi Arduino IDE
  - b. Buat inisialisasi dari sensor MPU6050 yang digunakan yaitu seperti berikut

Dibuat oleh:

Dilarang memperbanyak sebagian atau seluruh isi dokumen tanpa ijin tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh:



## FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA

### JOBSCHEET KENDALI DAN AKUISISI DATA

Semester 5

JS 4: MPU6050 menggunakan STM32 dan GUI  
Visual Studio

4x60 menit

No.

Revisi: 02

Tgl: 5 Februari 2019

Hal 1 dari

```
MPU6050 mpu;
int16_t ax, ay, az;
int16_t gx, gy, gz;
int AccX, AccY, AccZ, GyX, GyY, GyZ;
int Ax, Ay, Az, Gx,Gy,Gz;
int val1;
int val2;
int val3;
int val4;
int val5;
int val6;
int prevVal1;
int prevVal2;
int prevVal3;
int prevVal4;
int prevVal5;
int prevVal6;
```

- c. Ketikan pada void setup seperti berikut

```
void setup() {
    // put your setup code here, to run once:
    Wire.begin();
    Serial.begin(57600);
    mpu.initialize();
}
```

#### Keterangan:

- **Wire.begin();** memiliki fungsi sebagai penghubung antara Arduino dengan aplikasi Arduino IDE
- **Serial.begin(57600);** memiliki fungsi sebagai pe
- **mpu.initiakize();** memiliki fungsi sebagai penginisialisian dari sensor MPU6050

- d. Ketikan pada bagian void loop seperti berikut

```
void loop() {
    // put your main code here, to run repeatedly:
    AccelGyro();
    copy_data();
    kirim_data();

    delay(100);
}
```

Dibuat oleh:

Dilarang memperbanyak sebagian atau seluruh isi dokumen  
tanpa ijin tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh:



## FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA

### JOBSCHEET KENDALI DAN AKUISISI DATA

Semester 5

**JS 4: MPU6050 menggunakan STM32 dan GUI**  
**Visual Studio**

4x60 menit

No.

Revisi: 02

Tgl: 5 Februari 2019

Hal 1 dari

- e. Buatlah void baru yaitu void AccelGyro seperti inisialisasi pada void loop lalu ketik program seperti berikut

```
void AccelGyro()
{
    mpu.getMotion6(&ax, &ay, &az, &gx, &gy, &gz);
    //Program Accel & Gyro
    Ax=((ax/257));
    Ay=((ay/257))-1;
    Az=((az/257))-96;

    Gx=(( (gx/257))/4);
    Gy=(( (gy/257))/4);
    Gz=(( (gz/257))/4);

    AccX = Ax;
    AccY=Ay;
    AccZ=Az;

    GyX=Gx;
    GyY=Gy;
    GyZ=Gz;

    // Program YPR
    val1 = map(ax, -17000, 17000, -179, 179);
    if (val1 != prevVal1)
    {
        prevVal1 = val1;
    }
    val2 = map(ay, -17000, 17000, -45, 45);
    if (val2 != prevVal2)
    {
        prevVal2 = val2;
    }
    val3= map(gx, -17000, 17000, -60, 60);
    if(val3 != prevVal3)
    {
        prevVal3= val3;
    }
    val4= map(gz, -17000, 17000, -20, 20);
    if(val4 != prevVal4)
    {
        prevVal4= val4;
    }
}
```

- f. Buatlah void baru yaitu void copy\_data dan ketikan program didalamnya seperti berikut

Dibuat oleh:

Dilarang memperbanyak sebagian atau seluruh isi dokumen  
tanpa ijin tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh:



## FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA

### JOBSCHEET KENDALI DAN AKUISISI DATA

Semester 5

**JS 4: MPU6050 menggunakan STM32 dan GUI  
Visual Studio**

4x60 menit

No.

Revisi: 02

Tgl: 5 Februari 2019

Hal 1 dari

```
void copy_data()
{
    //pengiriman data Accel & Gyro
    Serial.print("<B0"); Serial.print(prevVal1);Serial.print("B0>");
    Serial.print("<B1"); Serial.print(prevVal2);Serial.print("B1>");
    Serial.print("<B2"); Serial.print(prevVal3);Serial.print("B2>");
    Serial.print("<B3"); Serial.print(prevVal4);Serial.print("B3>");
    Serial.print("<B4"); Serial.print(AccX);Serial.print("B4>");
    Serial.print("<B5"); Serial.print(AccY);Serial.print("B5>");
    Serial.print("<B6"); Serial.print(AccZ);Serial.print("B6>");
    Serial.print("<B7"); Serial.print(GyX);Serial.print("B7>");
    Serial.print("<B8"); Serial.print(GyY);Serial.print("B8>");
    Serial.print("<B9"); Serial.print(GyZ);Serial.print("B9>");
}
```

- g. Buatlah void baru yaitu void kirim\_data dan ketikan program didalmnya seperti berikut

```
void kirim_data()
{
    //pengiriman data Accel & Gyro
    Serial2.print("<B0"); Serial2.print(prevVal1);Serial2.print("B0>");
    Serial2.print("<B1"); Serial2.print(prevVal2);Serial2.print("B1>");
    Serial2.print("<B2"); Serial2.print(prevVal3);Serial2.print("B2>");
    Serial2.print("<B3"); Serial2.print(prevVal4);Serial2.print("B3>");
    Serial2.print("<B4"); Serial2.print(AccX);Serial2.print("B4>");
    Serial2.print("<B5"); Serial2.print(AccY);Serial2.print("B5>");
    Serial2.print("<B6"); Serial2.print(AccZ);Serial2.print("B6>");
    Serial2.print("<B7"); Serial2.print(GyX);Serial2.print("B7>");
    Serial2.print("<B8"); Serial2.print(GyY);Serial2.print("B8>");
    Serial2.print("<B9"); Serial2.print(GyZ);Serial2.print("B9>");
```

- h. Hubungkan antara Arduino yang telah terpasang sensor MPU6050 ke laptop/computer.

- i. Pilih Arduino dengan mengklik Tools → Board → STM32

Dibuat oleh:

Dilarang memperbanyak sebagian atau seluruh isi dokumen tanpa ijin tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh:



## FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA

### JOBSCHEET KENDALI DAN AKUISISI DATA

Semester 5

JS 4: MPU6050 menggunakan STM32 dan GUI  
Visual Studio

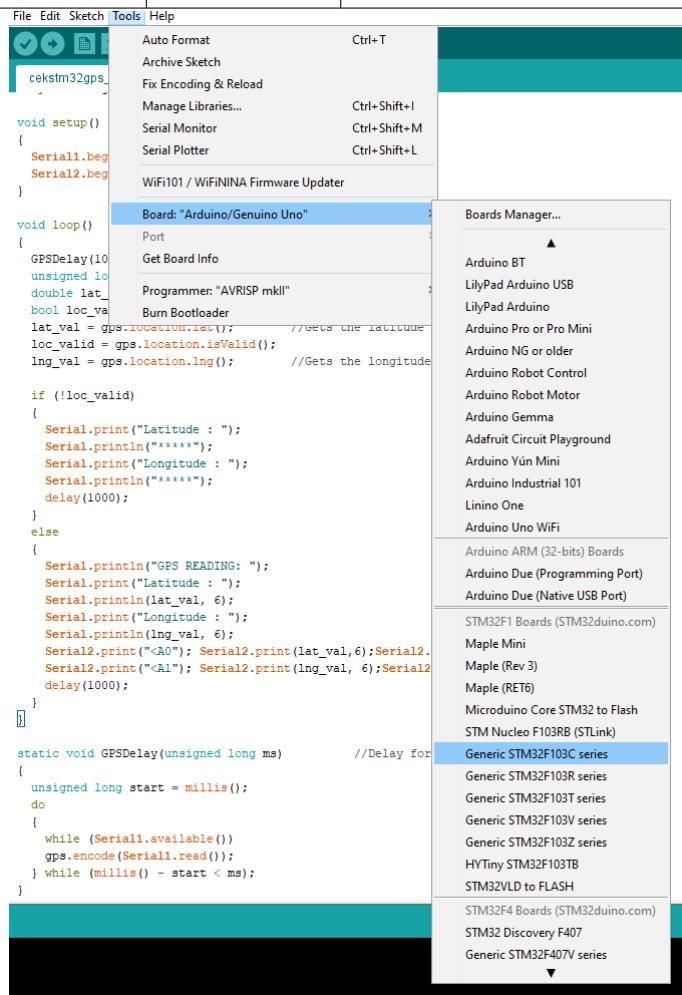
4x60 menit

No.

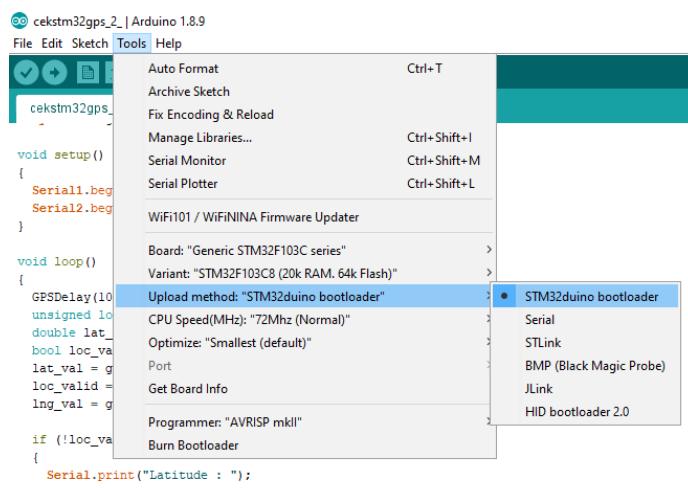
Revisi: 02

Tgl: 5 Februari 2019

Hal 1 dari



- j. Pilih metode upload pada menu Tools → Upload Method → Pilih STM32duino Bootloader



Dibuat oleh:

Dilarang memperbanyak sebagian atau seluruh isi dokumen tanpa ijin tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh:



## FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA

### JOBSHEET KENDALI DAN AKUISISI DATA

Semester 5

**JS 4: MPU6050 menggunakan STM32 dan GUI  
Visual Studio**

4x60 menit

No.

Revisi: 02

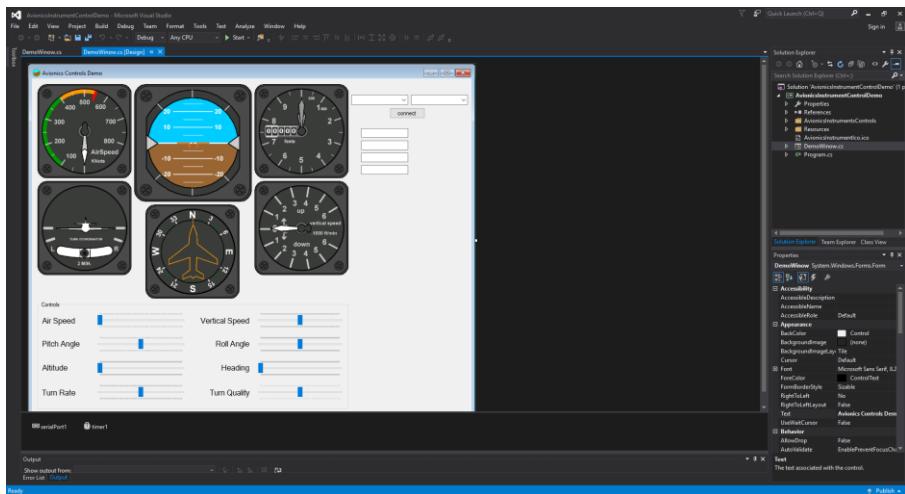
Tgl: 5 Februari 2019

Hal 1 dari

- k. Compile program yang telah diketik
- l. Upload dan simpan file yang telah selesai. Buka serial monitor untuk melihat kinerja dari program dan sensor

#### 5. Pembuatan GUI di Visual Studio

- a. Buka file GUI Avionic Instrument yang telah didownload



- b. Tambahkan library dari Zedgraph dengan cara klik Toolbox → Klik kanan pada list komponen → Pilih Choose Items... → Browse → Pilih Zedgraph.dll → OK

Dibuat oleh:

Dilarang memperbanyak sebagian atau seluruh isi dokumen tanpa ijin tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh:



## FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA

### JOBSCHEET KENDALI DAN AKUISISI DATA

Semester 5

JS 4: MPU6050 menggunakan STM32 dan GUI  
Visual Studio

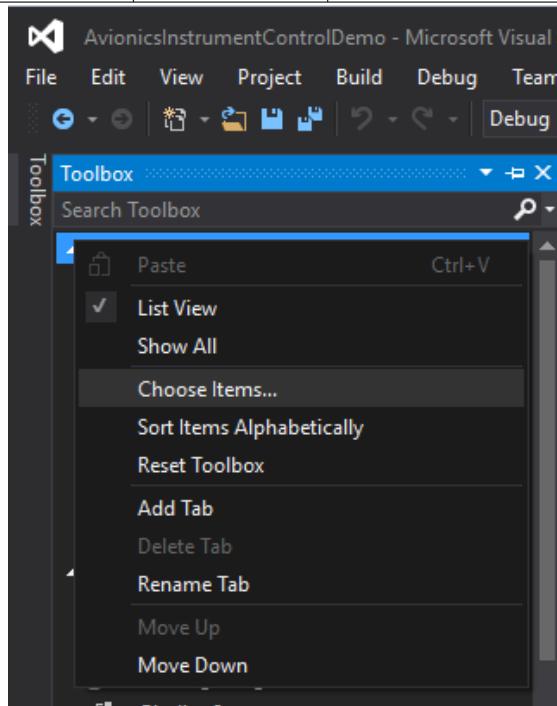
4x60 menit

No.

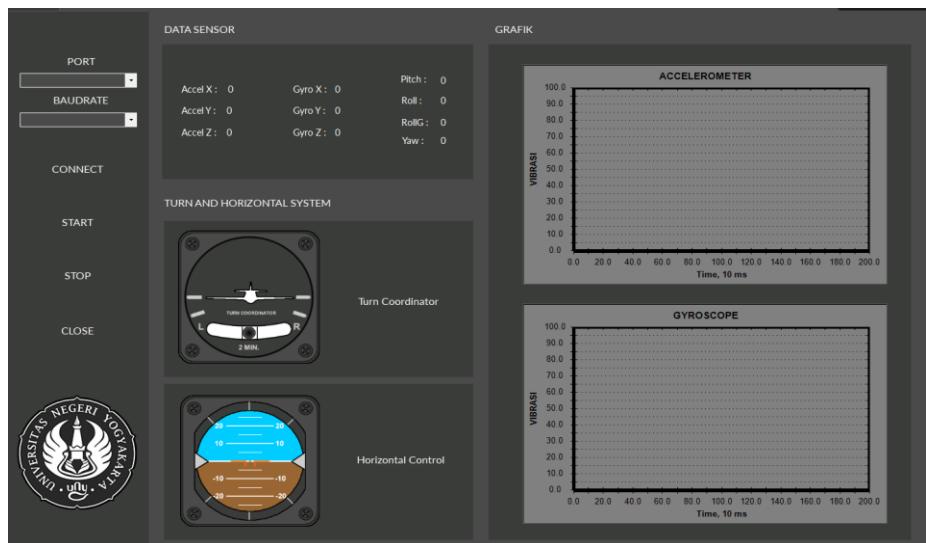
Revisi: 02

Tgl: 5 Februari 2019

Hal 1 dari



- c. Hapus beberapa item sisakan Horizontal and Turn Indicator pada desainer. Buat tampilan seperti gambar dibawah dengan menambahkan komponen melalui Toolbox.



- d. Beri nama setiap komponen dengan nama yang mudah seperti contoh yaitu tombol CONNECT berikut

Dibuat oleh:

Dilarang memperbanyak sebagian atau seluruh isi dokumen tanpa ijin tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh:



## FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA

## JOBSHEET KENDALI DAN AKUISISI DATA

Semester 5

JS 4: MPU6050 menggunakan STM32 dan GUI  
Visual Studio

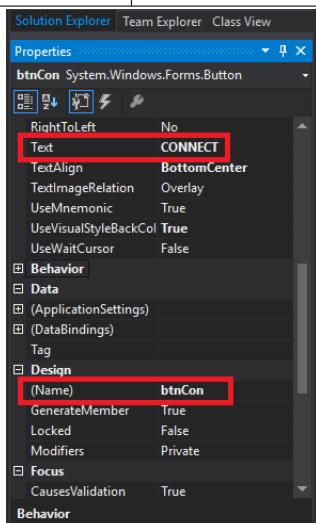
4x60 menit

No.

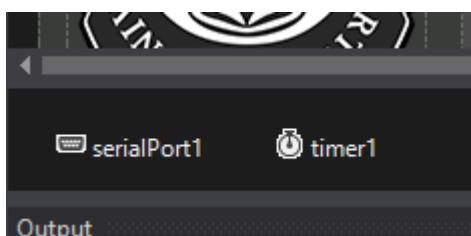
Revisi: 02

Tgl: 5 Februari 2019

Hal 1 dari



- e. Tambahkan komponen Serial Port dan Timer pada Form



- f. Setelah tampilan desain sudah jadi maka langkah selanjutnya memrogram GUI. Klik dua kali pada Form maka akan masuk pada tampilan Program.cs
- g. Tambahkan library Zedgraph dengan mengetik ‘using Zedgraph;’ seperti gambar berikut

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;
using System.IO.Ports;
using System.Collections;
using ZedGraph;
```

- h. Tambahkan inisialisasi komponen pada Public Form seperti gambar berikut

Dibuat oleh:

Dilarang memperbanyak sebagian atau seluruh isi dokumen  
tanpa ijin tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh:



## FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA

### JOBSHEET KENDALI DAN AKUISISI DATA

Semester 5

**JS 4: MPU6050 menggunakan STM32 dan GUI  
Visual Studio**

4x60 menit

No.

Revisi: 02

Tgl: 5 Februari 2019

Hal 1 dari

```
1 reference
public DemoWinow()
{
    InitializeComponent();
}
```

- i. Membuat data Baudrate dengan mengetik pada Form Load seperti gambar berikut

```
1 reference
private void DemoWinow_Load(object sender, EventArgs e)
{
    //BAUDRATE
    cbBaudr.Items.Add("9600");
    cbBaudr.Items.Add("38400");
    cbBaudr.Items.Add("57600");
    cbBaudr.Items.Add("115200");
```

- j. Lalu atur Combobox dari PORT dengan klik dua kali pada item cbPort, lalu ketikan program seperti berikut

```
1 reference
private void cbPort_DropDown(object sender, EventArgs e)
{
    cbPort.Items.Clear();
    cbPort.Items.AddRange(System.IO.Ports.SerialPort.GetPortNames());
```

- k. Program tombol Connect dengan klik dua kali button Connect dan ketikan program berikut

Dibuat oleh:

Dilarang memperbanyak sebagian atau seluruh isi dokumen tanpa ijin tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh:



## FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA

## JOBSHEET KENDALI DAN AKUISISI DATA

Semester 5

JS 4: MPU6050 menggunakan STM32 dan GUI  
Visual Studio

4x60 menit

No.

Revisi: 02

Tgl: 5 Februari 2019

Hal 1 dari

```
1 reference
private void btnCon_Click(object sender, EventArgs e)
{
    if (btnCon.Text == "CONNECT")
    {
        if (serialPort1.IsOpen == false)
        {
            serialPort1.PortName = cbPort.Text;
            serialPort1.BaudRate = int.Parse(cbBaudr.Text);

            try
            {
                serialPort1.Open();
                btnCon.Text = "DISCONNECT";
            }
            catch (Exception Ghozia)
            {
                MessageBox.Show("Serial Error" + Ghozia);
            }
        }
        else
        {
            serialPort1.Close();
            btnCon.Text = "CONNECT";
        }
    }
}
```

1. Program tombol Start dengan klik dua kali button Start dan ketikan program berikut

```
1 reference
private void btStart_Click(object sender, EventArgs e)
{
    serialPort1.Write("a");
    timer1.Start();
    btnStop.Enabled = true;
```

- m. Program tombol Stop dengan klik dua kali pada Button Stop dan ketikan program berikut ini

```
private void btnStop_Click(object sender, EventArgs e)
{
    timer1.Stop();
}
```

- n. Selanjutnya membuat program untuk mengubah jenis data input, buat public baru didalam **Public Partial Class**. Ketikan seperti gambar berikut

Dibuat oleh:

Dilarang memperbanyak sebagian atau seluruh isi dokumen tanpa ijin tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh:



## FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA

### JOBSCHEET KENDALI DAN AKUISISI DATA

Semester 5

JS 4: MPU6050 menggunakan STM32 dan GUI  
Visual Studio

4x60 menit

No.

Revisi: 02

Tgl: 5 Februari 2019

Hal 1 dari

```
20 references
public static string GetStringBetweenCharacters(string input, string charFrom, string charTo)
{
    int chrFrom = charFrom.Length;
    int chrTo = charTo.Length;
    int posFrom = input.IndexOf(charFrom);
    if (posFrom != -1) //if found char
    {
        int posTo = input.IndexOf(charTo, posFrom + 1);
        if (posTo != -1) //if found char
        {
            try { return input.Substring(posFrom + chrFrom, posTo - posFrom - chrTo); }
            catch { }
        }
    }
    return string.Empty;
}
1 reference
```

- o. Buat inisialisasi pada Public partial Class seperti berikut

```
public partial class DemoWinow : Form
{
    string DataMasuk = "";
    int TickStartACC;
    int TickStartGYRO;
    string AccX, AccY, AccZ, GyroX, GyroY, GyroZ;
    int Roll, Pitch, Yaw, Rollg;
```

- p. Ketikan program berikut didalam Timer untuk menampilkan data sensor pada label

```
private void timer1_Tick(object sender, EventArgs e)
{
    //Menerima data dari arduino dan ngubah label di GUI
    DataMasuk = serialPort1.ReadExisting();
    if (GetStringBetweenCharacters(DataMasuk, "<B4", "B4>") != "")
        AccX = GetStringBetweenCharacters(DataMasuk, "<B4", "B4>");
    lbAccX.Text = AccX;
    if (GetStringBetweenCharacters(DataMasuk, "<B5", "B5>") != "")
        AccY = GetStringBetweenCharacters(DataMasuk, "<B5", "B5>");
    lbAccY.Text = AccY;
    if (GetStringBetweenCharacters(DataMasuk, "<B6", "B6>") != "")
        AccZ = GetStringBetweenCharacters(DataMasuk, "<B6", "B6>");
    lbAccZ.Text = AccZ;
    if (GetStringBetweenCharacters(DataMasuk, "<B7", "B7>") != "")
        GyroX = GetStringBetweenCharacters(DataMasuk, "<B7", "B7>");
    lbGyroX.Text = GyroX;
    if (GetStringBetweenCharacters(DataMasuk, "<B8", "B8>") != "")
        GyroY = GetStringBetweenCharacters(DataMasuk, "<B8", "B8>");
    lbGyroY.Text = GyroY;
    if (GetStringBetweenCharacters(DataMasuk, "<B9", "B9>") != "")
        GyroZ = GetStringBetweenCharacters(DataMasuk, "<B9", "B9>");
    lbGyroZ.Text = GyroZ;

    if (GetStringBetweenCharacters(DataMasuk, "<B0", "B0>") != "")
        lbl_Pitch.Text = GetStringBetweenCharacters(DataMasuk, "<B0", "B0>");
    if (GetStringBetweenCharacters(DataMasuk, "<B1", "B1>") != "")
        lbl_Roll.Text = GetStringBetweenCharacters(DataMasuk, "<B1", "B1>");
    if (GetStringBetweenCharacters(DataMasuk, "<B2", "B2>") != "")
        lbl_Rollg.Text = GetStringBetweenCharacters(DataMasuk, "<B2", "B2>");
    if (GetStringBetweenCharacters(DataMasuk, "<B3", "B3>") != "")
        lbl_Yaw.Text = GetStringBetweenCharacters(DataMasuk, "<B3", "B3>");

    Pitch = Convert.ToInt32(lbl_Pitch.Text); //ax
    Roll = Convert.ToInt32(lbl_Roll.Text); //ay
    Rollg = Convert.ToInt32(lbl_Rollg.Text); //gx
    Yaw = Convert.ToInt32(lbl_Yaw.Text); //gy
```

Dibuat oleh:

Dilarang memperbanyak sebagian atau seluruh isi dokumen  
tanpa ijin tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh:



## FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA

### JOBSHEET KENDALI DAN AKUISISI DATA

Semester 5

**JS 4: MPU6050 menggunakan STM32 dan GUI  
Visual Studio**

4x60 menit

No.

Revisi: 02

Tgl: 5 Februari 2019

Hal 1 dari

- q. Program untuk menampilkan data sensor menjadi grafik yaitu dengan mengatur tampilan dari grafik seperti berikut ketikan didalam DemoWinow\_Load

```
//ZedGraph Accelero
zedGraphACC.GraphPane.Chart.Fill = new Fill(Color.Gray);
zedGraphACC.GraphPane.Fill = new Fill(Color.Gray);
zedGraphACC.GraphPane.YAxis.Visible = true;

//Nama Grafik sumbu x dan y
GraphPane mypaneACC = zedGraphACC.GraphPane;
mypaneACC.Title.Text = "ACCELEROMETER";
mypaneACC.YAxis.Title.Text = "VIBRASI";
mypaneACC.XAxis.Title.Text = "Time, 10 ms";

//Skala sumbu x dan y
mypaneACC.XAxis.Scale.MajorStep = 10;
mypaneACC.YAxis.Scale.Mag = 0;
mypaneACC.XAxis.Scale.Max = 200;

mypaneACC.YAxis.Scale.MajorStep = 20;
mypaneACC.XAxis.Scale.Mag = 0;
mypaneACC.YAxis.Scale.Max = 200;

mypaneACC.YAxis.Scale.MajorStep = 5;
mypaneACC.XAxis.Scale.Mag = 0;
mypaneACC.YAxis.Scale.Max = 100;

//Zedgraph Gyroscope
zedGraphGYRO.GraphPane.Chart.Fill = new Fill(Color.Gray);
zedGraphGYRO.GraphPane.Fill = new Fill(Color.Gray);
zedGraphGYRO.GraphPane.YAxis.Visible = true;

//Nama Grafik sumbu x dan y
GraphPane mypaneGYRO = zedGraphGYRO.GraphPane;
mypaneGYRO.Title.Text = "GYROSCOPE";
mypaneGYRO.YAxis.Title.Text = "VIBRASI";
mypaneGYRO.XAxis.Title.Text = "Time, 10 ms";

//Skala sumbu x dan y
mypaneGYRO.XAxis.Scale.MajorStep = 10;
mypaneGYRO.YAxis.Scale.Mag = 0;
mypaneGYRO.XAxis.Scale.Max = 200;

mypaneGYRO.YAxis.Scale.MajorStep = 20;
mypaneGYRO.XAxis.Scale.Mag = 0;
mypaneGYRO.YAxis.Scale.Max = 200;

mypaneGYRO.YAxis.Scale.MajorStep = 5;
mypaneGYRO.XAxis.Scale.Mag = 0;
mypaneGYRO.YAxis.Scale.Max = 100;
```

- r. Lalu mengatur inisialisasi garis data pada Public Partial Class

Dibuat oleh:

Dilarang memperbanyak sebagian atau seluruh isi dokumen  
tanpa ijin tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh:



## FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA

### JOBSHEET KENDALI DAN AKUISISI DATA

Semester 5

**JS 4: MPU6050 menggunakan STM32 dan GUI  
Visual Studio**

4x60 menit

No.

Revisi: 02

Tgl: 5 Februari 2019

Hal 1 dari

```
//Accelero
double timeAcc;
LineItem curve_Acc1, curve_Acc2, curve_Acc3;
RollingPointPairList list_Acc1 = new RollingPointPairList(60000);
RollingPointPairList list_Acc2 = new RollingPointPairList(60000);
RollingPointPairList list_Acc3 = new RollingPointPairList(60000);

//Gyroscope
double timeGyro;
LineItem curve_Gyro1, curve_Gyro2, curve_gyro3;
RollingPointPairList list_Gyro1 = new RollingPointPairList(60000);
RollingPointPairList list_Gyro2 = new RollingPointPairList(60000);
RollingPointPairList list_Gyro3 = new RollingPointPairList(60000);
```

- s. Lalu mengatur inisialisasi vibrasi untuk grafik gyroscope dan accelerometer data pada Public Partial Class

Dibuat oleh:

Dilarang memperbanyak sebagian atau seluruh isi dokumen  
tanpa ijin tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh:



## FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA

### JOBSCHEET KENDALI DAN AKUISISI DATA

Semester 5

JS 4: MPU6050 menggunakan STM32 dan GUI  
Visual Studio

4x60 menit

No.

Revisi: 02

Tgl: 5 Februari 2019

Hal 1 dari

```
private void graphVibrasiACC()
{
    GraphPane mypaneACC = zedGraphACC.GraphPane;
    //inisialisasi garis x,y,z
    curve_Acc1 = mypaneACC.AddCurve("X", list_Acc1, Color.Red, SymbolType.None);
    curve_Acc1.Line.Smooth = true;
    curve_Acc1.Line.SmoothTension = 0.2F;

    curve_Acc2 = mypaneACC.AddCurve("Y", list_Acc2, Color.Green, SymbolType.None);
    curve_Acc2.Line.Smooth = true;
    curve_Acc2.Line.SmoothTension = 0.2F;

    curve_Acc3 = mypaneACC.AddCurve("Z", list_Acc3, Color.Blue, SymbolType.None);
    curve_Acc3.Line.Smooth = true;
    curve_Acc3.Line.SmoothTension = 0.2F;

    //aturan scale dari sumbu X
    mypaneACC.XAxis.Scale.Min = 0;
    mypaneACC.XAxis.Scale.Max = 100;
    mypaneACC.XAxis.Scale.MinorStep = 5;
    mypaneACC.XAxis.Scale.MajorStep = 10;
    mypaneACC.AxisChange();                                     //perubahan sumbu X realtime
    TickStartACC = Environment.TickCount + 200;
}

reference
private void graphVibrasiGYRO()
{
    GraphPane mypaneGYRO = zedGraphGYRO.GraphPane;
    //inisialisasi garis x,y,z
    curve_Gyro1 = mypaneGYRO.AddCurve("X", list_Acc1, Color.Red, SymbolType.None);
    curve_Gyro1.Line.Smooth = true;
    curve_Gyro1.Line.SmoothTension = 0.2F;

    curve_Gyro2 = mypaneGYRO.AddCurve("Y", list_Acc2, Color.Green, SymbolType.None);
    curve_Gyro2.Line.Smooth = true;
    curve_Gyro2.Line.SmoothTension = 0.2F;

    curve_gyro3 = mypaneGYRO.AddCurve("Z", list_Acc3, Color.Blue, SymbolType.None);
    curve_gyro3.Line.Smooth = true;
    curve_gyro3.Line.SmoothTension = 0.2F;

    //aturan scale dari sumbu X
    mypaneGYRO.XAxis.Scale.Min = 0;
    mypaneGYRO.XAxis.Scale.Max = 100;
    mypaneGYRO.XAxis.Scale.MinorStep = 5;
    mypaneGYRO.XAxis.Scale.MajorStep = 10;
    mypaneGYRO.AxisChange();                                     //perubahan sumbu X realtime
    TickStartGYRO = Environment.TickCount + 200;
}
```

- t. Lalu mengatur munculnya gambar dari input data sensor pada grafik dengan membuat Private void baru didalam Public Partial Class. Buatlah dua private void untuk accelerometer dan gyroscope.

Dibuat oleh:

Dilarang memperbanyak sebagian atau seluruh isi dokumen tanpa ijin tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh:



## FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA

### JOBSCHEET KENDALI DAN AKUISISI DATA

Semester 5      JS 4: MPU6050 menggunakan STM32 dan GUI  
Visual Studio      4x60 menit

No.      Revisi: 02      Tgl: 5 Februari 2019      Hal 1 dari

```
private void DrawVibrasiAcc(string setpoint, string current1, string current2)
{
    //Make sure that the curvelist has at least one curve
    double intsetpoint;
    double intcurrent1;
    double intcurrent2;
    double.TryParse(setpoint, out intsetpoint);
    double.TryParse(current1, out intcurrent1);
    double.TryParse(current2, out intcurrent2);
    if (zedGraphACC.GraphPane.CurveList.Count <= 0)
        return;

    //get the first CurveItem in the graph
    LineItem curve = zedGraphACC.GraphPane.CurveList[0] as LineItem;
    LineItem curve1 = zedGraphACC.GraphPane.CurveList[1] as LineItem;
    LineItem curve2 = zedGraphACC.GraphPane.CurveList[2] as LineItem;
    if (curve == null) return;
    if (curve1 == null) return;
    if (curve2 == null) return;

    //get the PointPairList
    IPointListEdit List = curve.Points as IPointListEdit;
    IPointListEdit List1 = curve1.Points as IPointListEdit;
    IPointListEdit List2 = curve2.Points as IPointListEdit;

    //if this is null, it means the reference at curve.Points does not
    //support IPointListEdit, so we wont be able to modify it
    if (List == null) return;
    if (List1 == null) return;
    if (List2 == null) return;

    //time is measured in second
    //double time = (Environment.TickCount - TickStartGyro) / 100.0;
    timeAcc = (Environment.TickCount - TickStartGYRO) / 100.0;
    //3 seconds per cycle
    List.Add(timeAcc, intsetpoint);
    List1.Add(timeAcc, intcurrent1);
    List2.Add(timeAcc, intcurrent2);
    //keep the X scale at a rolling 30 second interval, with one
    //major step between the max X value and the end of the axis
    Scale xScale = zedGraphACC.GraphPane.XAxis.Scale;
    if (timeAcc > xScale.Max - xScale.MajorStep)
    {
        xScale.Max = timeAcc + xScale.MajorStep;
        xScale.Min = xScale.Max - 100.0;
    }
    //make sure the Y axis is rescaled to accommodate actual data
    zedGraphACC.AxisChange();
    //force a redraw
    zedGraphACC.Invalidate();
}
```

Dibuat oleh:

Dilarang memperbanyak sebagian atau seluruh isi dokumen  
tanpa ijin tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh:



## FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA

## JOBSHEET KENDALI DAN AKUISISI DATA

Semester 5 JS 4: MPU6050 menggunakan STM32 dan GUI Visual Studio 4x60 menit

No. Revisi: 02 Tgl: 5 Februari 2019 Hal 1 dari

```
private void DrawVibrasiGyro(string setpoint, string current1, string current2)
{
    //Make sure that the curvelist has at least one curve
    double intsetpoint;
    double intcurrent1;
    double intcurrent2;
    double.TryParse(setpoint, out intsetpoint);
    double.TryParse(current1, out intcurrent1);
    double.TryParse(current2, out intcurrent2);
    if (zedGraphGYRO.GraphPane.CurveList.Count <= 0)
        return;
    //get the first CurveItem in the graph
    LineItem curve = zedGraphGYRO.GraphPane.CurveList[0] as LineItem;
    LineItem curve1 = zedGraphGYRO.GraphPane.CurveList[1] as LineItem;
    LineItem curve2 = zedGraphGYRO.GraphPane.CurveList[2] as LineItem;
    if (curve == null) return;
    if (curve1 == null) return;
    if (curve2 == null) return;
    //get the PointPairList
    IPointListEdit List = curve.Points as IPointListEdit;
    IPointListEdit List1 = curve1.Points as IPointListEdit;
    IPointListEdit List2 = curve2.Points as IPointListEdit;
    //if this is null, it means the reference at curve.Points does not
    //support IPointListEdit, so we wont be able to modify it
    if (List == null) return;
    if (List1 == null) return;
    if (List2 == null) return;
    //time is measured in second
    //double time = (Environment.TickCount - TickStartGYRO) / 100.0;
    timeGyro = (Environment.TickCount - TickStartGYRO) / 100.0;
    //3 seconds per cycle
    List.Add(timeGyro, intsetpoint);
    List1.Add(timeGyro, intcurrent1);
    List2.Add(timeGyro, intcurrent2);
    //keep the X scale at a rolling 30 second interval, with one
    //major step between the max X value and the end of the axis
    Scale xScale = zedGraphGYRO.GraphPane.XAxis.Scale;
    if (timeGyro > xScale.Max - xScale.MajorStep)
    {
        xScale.Max = timeGyro + xScale.MajorStep;
        xScale.Min = xScale.Max - 50.0;
    }
    //make sure the Y axis is rescaled to accommodate actual data
    zedGraphGYRO.AxisChange();
    //force a redraw
    zedGraphGYRO.Invalidate();
}
```

- u. Lalu panggil/ ketikan program berikut di dalam program button Start

```
private void btStart_Click(object sender, EventArgs e)
{
    serialPort1.Write("a");
    timer1.Start();
    btnStop.Enabled = true;

    graphVibrasiACC();
    graphVibrasiGYRO();
}
```

Dibuat oleh:

Dilarang memperbanyak sebagian atau seluruh isi dokumen  
tanpa ijin tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh:



## FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA

### JOBSHEET KENDALI DAN AKUISISI DATA

Semester 5

**JS 4: MPU6050 menggunakan STM32 dan GUI  
Visual Studio**

4x60 menit

No. Revisi: 02 Tgl: 5 Februari 2019 Hal 1 dari

- v. Panggil/ ketikan program berikut didalam Timer untuk menampilkan gambar garis pada grafik secara up to date setiap detik

```
//menjalankan void ==> buat garis di graph
DrawVibrasiAcc(AccX, AccY, AccZ);
DrawVibrasiGyro(GyroX, GyroY, GyroZ);
```

- w. Lalu menampilkan data sensor untuk menggerakan Horizontal dan Turn Indicator. Keikan program berikut didalam Timer

```
horizonInstrumentControl1.SetAttitudeIndicatorParameters(Pitch,Roll);
horizonInstrumentControl1.SetAttitudeIndicatorParameters(Pitch, Roll);

turnCoordinatorInstrumentControl1.SetTurnCoordinatorParameters((Rollg / 10), Yaw);
turnCoordinatorInstrumentControl1.SetTurnCoordinatorParameters((Rollg / 10), Yaw);
```

## F. TUGAS

Buatlah laporan dari jobsheet yang telah dilakukan!

Dibuat oleh:

Dilarang memperbanyak sebagian atau seluruh isi dokumen tanpa ijin tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh: