

LAMPIRAN

Lampiran 1.

Instrumen Penelitian

Lampiran 1.1. Instrumen Angket Ahli Media

ANGKET PENILAIAN AHLI MATERI

Dalam rangka penelitian tugas akhir skripsi, saya mohon bantuan bapak/ibu/saudara untuk menjadi validator “**Quadruped Robot berbasis CM-510 sebagai Media Pembelajaran Mata Kuliah Praktik Robotika**” agar layak digunakan sebagai media pembelajaran.

Materi Pembelajaran : Praktik Robotika

Pembuat : Hilmi Musthafa Albasyir

Tanggal :

Prosedur Pengisian Instrumen Materi Pembelajaran

1. Dimohon kepada bapak/ibu/saudara untuk memberikan penilaian pada salah satu kolom di dalam angket ini berdasarkan kriteria berikut.
SS : Sangat Setuju
S : Setuju
TS : Tidak Setuju
STS : Sangat Tidak Setuju
2. Dimohon untuk memberikan tanda *check* (✓) pada salah satu pilihan jawaban dalam setiap pernyataan sesuai pendapat bapak/ibu/saudara.
3. Jika terdapat kekurangan pada media pembelajaran ini, mohon bapak/ibu/saudara memberikan saran dan masukan pada bagian yang telah disediakan.
4. Mohon untuk menuliskan nama sebagai validator (jika belum tersedia) pada bagian akhir lembar instrumen ini sebagai bukti bahwa instrumen ini dibuat sebenar-benarnya.
5. Atas bantuan bapak/ibu/saudara, kami mengucapkan terima kasih.

Pernyataan:

No	Pernyataan	Pilihan Jawaban			
		SS	S	TS	STS
1	Media pembelajaran sesuai dengan silabus.				
2	Materi yang diajarkan dalam media pembelajaran <i>quadruped</i> robot sesuai dengan capaian pembelajaran peserta didik pada praktik robotika.				
3	Materi yang diajarkan dalam media pembelajaran <i>quadruped</i> robot sesuai dengan bahan kajian praktik robotika.				
4	Materi dapat dipahami setelah menggunakan media pembelajaran <i>quadruped</i> robot.				
5	Media pembelajaran <i>quadruped</i> robot dapat mendukung materi yang terdapat dalam <i>labsheet</i> .				
6	Materi disajikan runtut/sistematis.				
7	Materi yang disajikan didukung dasar teori yang jelas.				
8	Petunjuk penggunaan perangkat keras pada media pembelajaran <i>quadruped</i> robot dijelaskan di dalam panduan pengoperasian dan <i>labsheet</i> .				
9	Petunjuk penggunaan perangkat lunak yang digunakan pada media pembelajaran <i>quadruped</i> robot dijelaskan dalam panduan pengoperasian dan <i>labsheet</i> .				
10	Cara menghubungkan motor servo Dynamixel MX-64 dan MX-28 pada media pembelajaran <i>quadruped</i> robot dijelaskan dalam panduan pengoperasian.				
11	Cara menghubungkan Stick PS2 pada media pembelajaran <i>quadruped</i> robot dijelaskan di dalam panduan pengoperasian.				

12	Cara membuat program yang digunakan dalam pemrograman <i>quadruped</i> robot dijelaskan di dalam panduan pengoperasian.			
13	Materi tentang <i>Stick PS2</i> disajikan dengan jelas di dalam <i>labsheet</i> .			
14	Materi tentang kontroler CM-510 disajikan dengan jelas di dalam <i>labsheet</i> .			
15	Materi tentang motor servo Dynamixel MX-64 dan MX-28 disajikan dengan jelas di dalam <i>labsheet</i> .			
16	Materi tentang <i>software RoboPlus</i> disajikan dengan jelas di dalam <i>labsheet</i> .			
17	Materi yang disajikan dapat meningkatkan motivasi belajar peserta didik dalam pembelajaran praktik robotika.			
18	Materi yang disajikan mendukung keterlibatan peserta didik dalam pembelajaran praktik robotika.			
19	Materi yang disajikan sesuai dengan perkembangan intelektual peserta didik.			
20	Materi yang disajikan sesuai dengan perkembangan emosional peserta didik.			
21	Tata bahasa pada panduan pengoperasian dan <i>labsheet</i> sesuai dengan kaidah Bahasa Indonesia			
22	Istilah-istilah pada panduan pengoperasian dan <i>labsheet</i> sudah baku.			

Kesimpulan:

Menurut saya, materi untuk Media Pembelajaran *Quadruped Robot* berbasis CM-510 sebagai Media Pembelajaran Mata Kuliah Praktik Robotika dinyatakan:

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

Saran dan Perbaikan:

.....
.....
.....
.....
.....
.....
.....
.....
.....

Yogyakarta,
Validator

(.....)

Lampiran 1.2. Instrumen Angket Ahli Materi

ANGKET PENILAIAN AHLI MEDIA

Dalam rangka penelitian tugas akhir skripsi, saya mohon bantuan bapak/ibu/saudara untuk menjadi validator **“Quadruped Robot berbasis CM-510 sebagai Media Pembelajaran Mata Kuliah Praktik Robotika”** agar layak digunakan sebagai media pembelajaran.

Materi Pembelajaran : Praktik Robotika

Pembuat : Hilmi Musthafa Albasyir

Tanggal :

Prosedur Pengisian Instrumen Media Pembelajaran

1. Dimohon kepada bapak/ibu/saudara untuk memberikan penilaian pada salah satu kolom di dalam angket ini berdasarkan kriteria berikut.

SS : Sangat Setuju
S : Setuju
TS : Tidak Setuju
STS : Sangat Tidak Setuju
2. Dimohon untuk memberikan tanda *check* (✓) pada salah satu pilihan jawaban dalam setiap pernyataan sesuai pendapat bapak/ibu/saudara.
3. Jika terdapat kekurangan pada media pembelajaran ini, mohon bapak/ibu/saudara memberikan saran dan masukan pada bagian yang telah disediakan.
4. Mohon untuk menuliskan nama sebagai validator (jika belum tersedia) pada bagian akhir lembar instrumen ini sebagai bukti bahwa instrumen ini dibuat sebenar-benarnya.
5. Atas bantuan bapak/ibu/saudara, kami mengucapkan terima kasih.

Pernyataan:

No	Pernyataan	Pilihan Jawaban			
		SS	S	TS	STS
1	Penggunaan media pembelajaran <i>quadruped</i> robot membantu proses pembelajaran praktik robotika.				
2	Media pembelajaran <i>quadruped</i> robot dapat digunakan pendidik untuk menjelaskan materi yang disampaikan.				
3	Media pembelajaran <i>quadruped</i> robot dapat digunakan pendidik untuk menarik perhatian peserta didik.				
4	Media pembelajaran <i>quadruped</i> robot dapat menambah variasi materi tentang pemrograman robot pada praktik robotika.				
5	Penggunaan CM-510 sebagai kontroler menambah variasi materi praktik robotika.				
6	Penggunaan motor servo MX-64 dan MX-28 sebagai aktuator menambah variasi materi praktik robotika.				
7	Media pembelajaran <i>quadruped</i> robot meningkatkan keaktifan peserta didik dalam pembelajaran.				
8	Penggunaan media pembelajaran <i>quadruped</i> robot menumbuhkan semangat belajar peserta didik.				
9	Media pembelajaran <i>quadruped</i> robot memiliki keterkaitan dengan mata kuliah lain.				
10	Media pembelajaran <i>quadruped</i> robot dapat mendukung pembelajaran pada mata kuliah lain.				

11	Tampilan GUI yang digunakan pada media pembelajaran menarik.			
12	Informasi pada tampilan GUI yang digunakan mudah dipahami.			
13	Bentuk media pembelajaran menarik.			
14	Peletakan komponen pada media pembelajaran tersusun dengan baik.			
15	Perangkat keras pada media pembelajaran <i>quadruped</i> robot berfungsi dengan baik.			
16	Perangkat lunak yang digunakan berfungsi dengan baik pada media pembelajaran <i>quadruped</i> robot.			
17	Panduan pengoperasian membantu penggunaan media pembelajaran dengan mudah.			
18	Instalasi <i>wiring</i> aktuator motor servo MX-64 dan MX-28 pada media pembelajaran <i>quadruped</i> robot dapat dilakukan dengan mudah.			
19	Perakitan media pembelajaran <i>quadruped</i> robot dapat dilakukan dengan mudah.			
20	Media pembelajaran menarik untuk digunakan.			
21	Media pembelajaran sesuai dengan karakteristik peserta didik.			
22	Media pembelajaran sesuai dengan jenjang pendidikan peserta didik.			

Kesimpulan:

Menurut saya, Media *Quadruped* Robot berbasis CM-510 sebagai Media Pembelajaran Mata Kuliah Praktik Robotika dinyatakan:

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

Saran dan Perbaikan:

.....
.....
.....
.....
.....
.....
.....

Yogyakarta,

Validator

(.....)

Lampiran 1.2. Instrumen Angket Pengguna

ANGKET PENILAIAN

QUADRUPED ROBOT BERBASIS CM-510

SEBAGAI MEDIA PEMBELAJARAN MATA KULIAH

PRAKTIK ROBOTIKA



IDENTITAS PESERTA DIDIK

Nama :

NIM :

**JURUSAN PENDIDIKAN TEKNIK ELEKTRO
FAKULTAS TEKNIK
UNIVERSITAS NEGERI YOGYAKARTA
2019**

Angket Penilaian Media

Hal : Pengisian Angket Penelitian

Kepada : Peserta didik Program Studi Pendidikan Teknik Mekatronika

Assalamu'alaikum wr. wb.

Dengan hormat,

Mohon kesediaan dan bantuan saudara untuk meluangkan waktu guna mengisi angket ini. Angket ini berguna untuk mengumpulkan data terkait dengan “*Quadruped* Robot berbasis CM-510 sebagai Media Pembelajaran Mata Kuliah Praktik Robotika”.

Angket ini bukan merupakan tes sehingga jawaban yang anda berikan tidak akan mempengaruhi nilai mata pelajaran. Jawaban yang baik adalah jawaban yang sesuai dengan kenyataan dan diisi berdasarkan hati nurani saudara, serta akan kami jamin kerahasiaannya. Kejujuran saudara dalam menjawab angket ini sangat diharapkan demi mendapatkan hasil penelitian yang maksimal.

Atas bantuan dan kerjasama dari saudara, saya ucapkan terimakasih.

Wassalamu'alaikum wr. wb.

Yogyakarta,

.....2019

Hormat Saya

Peneliti

A. Petunjuk Pengisian Angket

1. Angket ini dimaksudkan untuk mengetahui pendapat/penilaian anda sebagai pengguna media pembelajaran ***Quadruped Robot berbasis CM-510 sebagai Media Pembelajaran Mata Kuliah Praktik Robotika***.
2. Anda diharapkan memilih salah satu pilihan jawaban pada setiap pernyataan yang tersedia dengan memberikan TANDA SILANG (X) pada kolom jawaban.

Contoh:

NO	PERNYATAAN	JAWABAN			
1	Desain tata letak komponen pada <i>hardware</i> sudah rapi.	1	2	3	X

3. Jika anda ingin mengubah jawaban, maka anda memberikan tanda SAMA DENGAN (=) pada pilihan jawaban yang akan diganti dan memberikan TANDA SILANG pada kolom penggantinya.

NO	PERNYATAAN	JAWABAN			
1	Desain tata letak komponen pada <i>hardware</i> sudah rapi.	1	X	3	X

4. Keterangan jawaban:

1 = Sangat Tidak Setuju / Sangat Tidak Sesuai / Sangat Tidak Baik

2 = Tidak Setuju / Tidak Sesuai / Tidak Baik

3 = Setuju / Sesuai / Baik

4 = Sangat Setuju / Sangat Sesuai / Sangat Baik

5. Komentar atau saran anda mohon ditulis pada lebar yang telah disediakan.

Atas kesediaan anda untuk mengisi angket ini, saya ucapkan terima kasih.

B. Angket Penilaian

NO	PERNYATAAN	JAWABAN			
1	Media pembelajaran yang dibuat sesuai dengan materi pembelajaran yang diajarkan.	1	2	3	4
2	Materi yang ada dalam <i>labsheet</i> mudah dan menarik untuk dipahami.	1	2	3	4
3	Panduan pengoperasian media pembelajaran <i>quadruped</i> robot disajikan dengan lengkap.	1	2	3	4
4	Media pembelajaran <i>quadruped</i> robot dilengkapi dengan <i>labsheet</i> .	1	2	3	4
5	Materi yang disajikan sesuai dengan mata kuliah praktik robotika.	1	2	3	4
6	Materi yang disajikan berisi kompetensi yang dibutuhkan.	1	2	3	4
7	Gambar-gambar yang disajikan dalam panduan pengoperasian dan <i>labsheet</i> mempermudah dalam melaksanakan kegiatan praktikum.	1	2	3	4
8	Langkah kerja dalam panduan pengoperasian mudah untuk diikuti.	1	2	3	4
9	Bagian-bagian media pembelajaran <i>quadruped</i> robot tidak membingungkan.	1	2	3	4
10	Pengoperasian media pembelajaran <i>quadruped</i> robot dapat dilakukan dengan mudah.	1	2	3	4
11	GUI yang digunakan untuk mengoperasikan media pembelajaran <i>quadruped</i> robot menarik dan mudah digunakan.	1	2	3	4
12	Desain media pembelajaran menarik.	1	2	3	4
13	Media pembelajaran <i>quadruped</i> robot memberi tambahan pengetahuan dalam pemrograman sistem gerak robot pada mata kuliah praktik robotika.	1	2	3	4
14	Media pembelajaran <i>quadruped</i> robot memberi tambahan pengetahuan tentang sensor pada robot.	1	2	3	4

15	Media pembelajaran <i>quadruped</i> robot dapat membantu dalam memahami materi pada mata kuliah lain.	1	2	3	4
16	Media pembelajaran <i>quadruped</i> robot memberi kesempatan untuk mempelajari pemrograman Stick PS2 dan komunikasi Arduino dengan CM-510 pada mata kuliah praktik robotika.	1	2	3	4
17	Media pembelajaran <i>quadruped</i> robot memberi kesempatan untuk mempelajari pemrograman motor servo dynamixel pada mata kuliah praktik robotika.	1	2	3	4
18	Media pembelajaran <i>quadruped</i> robot dapat memotivasi untuk belajar pemrograman Stick PS2 dan komunikasi Arduino dengan CM-510 pada pembelajaran praktik robotika.	1	2	3	4
19	Media pembelajaran <i>quadruped</i> robot dapat memotivasi untuk belajar pemrograman motor servo dynamixel pada pembelajaran praktik robotika.	1	2	3	4
20	Media pembelajaran <i>quadruped</i> robot menambah kompetensi dalam pemrograman robot.	1	2	3	4
21	Media pembelajaran <i>quadruped</i> robot dapat meningkatkan keaktifan belajar saat pembelajaran praktik robotika.	1	2	3	4
22	Media pembelajaran <i>quadruped</i> robot memberi tambahan wawasan mengenai robot yang digunakan di dunia industri.	1	2	3	4

C. Komentar dan Saran Umum

.....
.....
.....
.....
.....
.....
.....

Yogyakarta,

(.....)

Lampiran 2.

Validasi Instrumen Penelitian

SURAT PERMOHONAN VALIDASI INSTRUMEN PENELITIAN

Hal : Permohonan Validasi Instrumen TAS
Lampiran : 1 Bendel

Kepada Yth,

Dr. phil. Nurhening Yuniarti

Dosen Jurusan Pendidikan Teknik Elektro
Di Fakultas Teknik UNY

Sehubungan dengan pelaksanaan Tugas Akhir Skripsi (TAS), dengan ini saya:

Nama : Vando Gusti Alhakim
NIM : 14518244011
Program Studi : Pendidikan Teknik Mekatronika
Judul TAS : Pengembangan *Trainer Kit* Lengan Robot Berbasis OpenCM 9.04
Menggunakan Sensor Jarak Inframerah Sharp GP2Y0A41SK0F
sebagai Media Pembelajaran Praktik Robotika

dengan hormat mohon Bapak/Ibu berkenan memberikan validasi terhadap instrumen penelitian TAS yang telah saya susun. Sebagai bahan pertimbangan, bersama ini saya lampirkan: (1) proposal TAS, (2) kisi-kisi instrumen penelitian TAS, dan (3) draf instrumen penelitian TAS.

Demikian permohonan saya, atas bantuan dan perhatian Bapak/Ibu diucapkan terima kasih.

Yogyakarta, 11 Januari 2018

Pemohon,



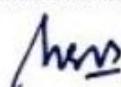
Vando Gusti Alhakim

NIM. 14518244011

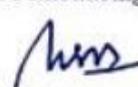
Mengetahui,

Kaprodi Pendidikan Teknik Mekatronika,

Dosen Pembimbing TAS,



Herlambang Sigit Pramono, S.T., M.Sc
NIP. 19650829 199903 1 001



Herlambang Sigit Pramono, S.T., M.Sc
NIP. 19650829 199903 1 001

**SURAT PERNYATAAN VALIDASI
INSTRUMEN PENELITIAN TUGAS AKHIR SKRIPSI**

Saya yang bertanda tangan di bawah ini:

Nama : Dr. phil. NURHENING YUNIARTI
NIP : 19750609 200212 2 002
Jurusan : P.T. ELEKTRO

menyatakan bahwa instrumen penelitian TAS atas nama mahasiswa:

Nama : Vando Gusti Alhakim

NIM : 14518244011

Program Studi : Pendidikan Teknik Mekatronika

Judul TAS : Pengembangan *Trainer Kit* Lengan Robot Berbasis OpenCM 9.04
Menggunakan Sensor Jarak Inframerah Sharp GP2Y0A41SK0F
sebagai Media Pembelajaran Praktik Robotika

Setelah dilakukan kajian atas instrumen penelitian TAS tersebut dapat dinyatakan:

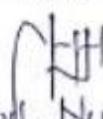
- Layak digunakan untuk penelitian
- Layak digunakan dengan perbaikan
- Tidak layak digunakan untuk penelitian yang bersangkutan

dengan catatan dan saran/perbaikan sebagaimana terlampir.

Demikian agar dapat digunakan sebagaimana semestinya.

Yogyakarta, 11 Januari 2018

Validator,


Dr. phil. Nurhening Y
NIP. 19750609 200212 2 002

Catatan:

- Beri tanda ✓

SURAT PERMOHONAN VALIDASI INSTRUMEN PENELITIAN

Hal : Permohonan Validasi Instrumen TAS
Lampiran : 1 Bendel

Kepada Yth,

Dr. Samsul Hadi, M.Pd., M.T

Dosen Jurusan Pendidikan Teknik Elektro
Di Fakultas Teknik UNY

Sehubungan dengan pelaksanaan Tugas Akhir Skripsi (TAS), dengan ini saya:

Nama : Vando Gusti Alhakim
NIM : 14518244011
Program Studi : Pendidikan Teknik Mekatronika
Judul TAS : Pengembangan *Trainer Kit* Lengan Robot Berbasis OpenCM 9.04
Menggunakan Sensor Jarak Inframerah Sharp GP2Y0A41SK0F
sebagai Media Pembelajaran Praktik Robotika

dengan hormat mohon Bapak/Ibu berkenan memberikan validasi terhadap instrumen penelitian TAS yang telah saya susun. Sebagai bahan pertimbangan, bersama ini saya lampirkan: (1) proposal TAS, (2) kisi-kisi instrumen penelitian TAS, dan (3) draf instrumen penelitian TAS.

Demikian permohonan saya, atas bantuan dan perhatian Bapak/Ibu diucapkan terima kasih.

Yogyakarta, 12 Januari 2018

Penitipan,

Vando Gusti Alhakim

NIM. 14518244011

Mengetahui,

Kaprodi Pendidikan Teknik Mekatronika,

Dosen Pembimbing TAS,

Herlambang Sigit Pramono, S.T., M.Sc
NIP. 19650829 199903 1 001

Herlambang Sigit Pramono, S.T., M.Sc
NIP. 19650829 199903 1 001

**SURAT PERNYATAAN VALIDASI
INSTRUMEN PENELITIAN TUGAS AKHIR SKRIPSI**

Saya yang bertanda tangan di bawah ini:

Nama : Dr. Samsul Hadi, M.Pd., M.T
NIP : 19600529 1984 03 1 003
Jurusan : Pendidikan Teknik Elektro

menyatakan bahwa instrumen penelitian TAS atas nama mahasiswa:

Nama : Vando Gusti Alhakim

NIM : 14518244011

Program Studi : Pendidikan Teknik Mekatronika

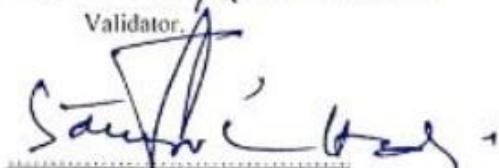
Judul TAS : Pengembangan *Trainer Kit* Lengan Robot Berbasis OpenCM 9.04
Menggunakan Sensor Jarak Inframerah Sharp GP2Y0A41SK0F
sebagai Media Pembelajaran Praktik Robotika

Setelah dilakukan kajian atas instrumen penelitian TAS tersebut dapat dinyatakan:

- Layak digunakan untuk penelitian
 Layak digunakan dengan perbaikan
 Tidak layak digunakan untuk penelitian yang bersangkutan

dengan catatan dan saran/perbaikan sebagaimana terlampir.

Demikian agar dapat digunakan sebagaimana semestinya.

Yogyakarta, 15/1 - 20/18
Validator, 
NIP.

Catatan:

- Beri tanda ✓

Lampiran 3.

Hasil Validasi Produk

Lampiran 3.1. Hasil Validasi Media

SURAT PERNYATAAN VALIDASI MEDIA PENELITIAN TUGAS AKHIR SKRIPSI

Saya yang bertanda tangan dibawah ini:

Nama : Amelia Fauziah Husna, S.Pd., M.Pd.

NIP : 199103012019032016

Jurusan : Pendidikan Teknik Elektro

Menyatakan bahwa media penelitian TAS tersebut atas nama mahasiswa:

Nama : Hilmi Musthafa Albasyir

Nim : 15518241029

Program Studi : Pendidikan Teknik Mekatronika

Judul TAS : *Quadruped Robot berbasis CM-510 sebagai Media Pembelajaran Mata Kuliah Praktik Robotika*

Setelah dilakukan kajian atas media penelitian TAS tersebut dapat dinyatakan:

- Layak digunakan tanpa perbaikan
- Layak digunakan dengan perbaikan
- Tidak layak digunakan untuk penelitian yang bersangkutan

Dengan saran / perbaikan sebagaimana terlampir,

Demikian agar dapat digunakan sebagaimana mestinya

Yogyakarta, 12 November 2019

Validator



Amelia Fauziah Husna, S.Pd., M.Pd.

NIP. 199103012019032016

Catatatan:

- Beri tanda ✓

ANGKET PENILAIAN AHLI MEDIA

Dalam rangka penelitian tugas akhir skripsi, saya mohon bantuan bapak/ibu/saudara untuk menjadi validator **“Quadruped Robot berbasis CM-510 sebagai Media Pembelajaran Mata Kuliah Praktik Robotika”** agar layak digunakan sebagai media pembelajaran.

Materi Pembelajaran : Praktik Robotika

Pembuat : Hilmi Musthafa Albasyir

Tanggal :

Prosedur Pengisian Instrumen Media Pembelajaran

1. Dimohon kepada bapak/ibu/saudara untuk memberikan penilaian pada salah satu kolom di dalam angket ini berdasarkan kriteria berikut.

SS : Sangat Setuju

S : Setuju

TS : Tidak Setuju

STS : Sangat Tidak Setuju

2. Dimohon untuk memberikan tanda *check* (✓) pada salah satu pilihan jawaban dalam setiap pernyataan sesuai pendapat bapak/ibu/saudara.
3. Jika terdapat kekurangan pada media pembelajaran ini, mohon bapak/ibu/saudara memberikan saran dan masukan pada bagian yang telah disediakan.
4. Mohon untuk menuliskan nama sebagai validator (jika belum tersedia) pada bagian akhir lembar instrumen ini sebagai bukti bahwa instrumen ini dibuat sebenar-benarnya.
5. Atas bantuan bapak/ibu/saudara, kami mengucapkan terima kasih.

Pernyataan:

No	Pernyataan	Pilihan Jawaban			
		SS	S	TS	STS
1	Penggunaan media pembelajaran <i>quadruped</i> robot membantu proses pembelajaran praktik robotika.	✓			
2	Media pembelajaran <i>quadruped</i> robot dapat digunakan pendidik untuk menjelaskan materi yang disampaikan.	✓			
3	Media pembelajaran <i>quadruped</i> robot dapat digunakan pendidik untuk menarik perhatian peserta didik.	✓			
4	Media pembelajaran <i>quadruped</i> robot dapat menambah variasi materi tentang pemrograman robot pada praktik robotika.	✓			
5	Penggunaan CM-510 sebagai kontroler menambah variasi materi praktik robotika.	✓			
6	Penggunaan motor servo MX-64 dan MX-28 sebagai aktuator menambah variasi materi praktik robotika.	✓			
7	Media pembelajaran <i>quadruped</i> robot meningkatkan keaktifan peserta didik dalam pembelajaran.		✓		
8	Penggunaan media pembelajaran <i>quadruped</i> robot menumbuhkan semangat belajar peserta didik.		✓		
9	Media pembelajaran <i>quadruped</i> robot memiliki keterkaitan dengan mata kuliah lain.	✓			
10	Media pembelajaran <i>quadruped</i> robot dapat mendukung pembelajaran pada mata kuliah lain.		✓		
11	Tampilan GUI yang digunakan pada media pembelajaran menarik.	✓			
12	Informasi pada tampilan GUI yang digunakan mudah dipahami.		✓		
13	Bentuk media pembelajaran menarik.	✓			
14	Peletakan komponen pada media pembelajaran tersusun dengan baik.		✓		

15	Perangkat keras pada media pembelajaran <i>quadruped</i> robot berfungsi dengan baik.	✓			
16	Perangkat lunak yang digunakan berfungsi dengan baik pada media pembelajaran <i>quadruped</i> robot.	✓			
17	Panduan pengoperasian membantu penggunaan media pembelajaran dengan mudah.		✓		
18	Instalasi <i>wiring</i> aktuator motor servo MX-64 dan MX-28 pada media pembelajaran <i>quadruped</i> robot dapat dilakukan dengan mudah.		✓		
19	Perakitan media pembelajaran <i>quadruped</i> robot dapat dilakukan dengan mudah.		✓		
20	Media pembelajaran menarik untuk digunakan.	✓			
21	Media pembelajaran sesuai dengan karakteristik peserta didik.	✓			
22	Media pembelajaran sesuai dengan jenjang pendidikan peserta didik.	✓			

Kesimpulan:

Menurut saya, Media *Quadruped* Robot berbasis CM-510 sebagai Media Pembelajaran Mata Kuliah Praktik Robotika dinyatakan:

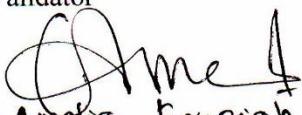
1. Layak digunakan tanpa revisi
2. Layak digunakan dengan revisi pada saran
3. Tidak layak digunakan

Saran dan Perbaikan:

- Pada lab sheet, langkah kerja pd awal lebih diperinci
- perbaiki pd sesuai dg arahan / ketuntukan pd lab sheet dan modul.

Yogyakarta,

Validator


(Amelia Fauziah Husna)

SURAT PERNYATAAN VALIDASI
MEDIA PENELITIAN TUGAS AKHIR SKRIPSI

Saya yang bertanda tangan dibawah ini:

Nama : Ir. Moh. Khairudin, M.T., Ph.D.

NIP : 197904122002121002

Jurusan : Pendidikan Teknik Elektro

Menyatakan bahwa media penelitian TAS tersebut atas nama mahasiswa:

Nama : Hilmi Musthafa Albasyir

Nim : 15518241029

Program Studi : Pendidikan Teknik Mekatronika

Judul TAS : *Quadruped Robot berbasis CM-510 sebagai Media Pembelajaran Mata Kuliah Praktik Robotika*

Setelah dilakukan kajian atas media penelitian TAS tersebut dapat dinyatakan:

Layak digunakan tanpa perbaikan

Layak digunakan dengan perbaikan

Tidak layak digunakan untuk penelitian yang bersangkutan

Dengan saran / perbaikan sebagaimana terlampir,

Demikian agar dapat digunakan sebagaimana mestinya

Yogyakarta, 12 November 2019

Validator



Ir. Moh. Khairudin, M.T., Ph.D.

NIP. 197904122002121002

Catatatan:

Beri tanda √

ANGKET PENILAIAN AHLI MEDIA

Dalam rangka penelitian tugas akhir skripsi, saya mohon bantuan bapak/ibu/saudara untuk menjadi validator **“Quadruped Robot berbasis CM-510 sebagai Media Pembelajaran Mata Kuliah Praktik Robotika”** agar layak digunakan sebagai media pembelajaran.

Materi Pembelajaran : Praktik Robotika

Pembuat : Hilmi Musthafa Albasyir

Tanggal :

Prosedur Pengisian Instrumen Media Pembelajaran

1. Dimohon kepada bapak/ibu.saudara untuk memberikan penilaian pada salah satu kolom di dalam angket ini berdasarkan kriteria berikut.
SS : Sangat Setuju
S : Setuju
TS : Tidak Setuju
STS : Sangat Tidak Setuju
2. Dimohon untuk memberikan tanda *check* (✓) pada salah satu pilihan jawaban dalam setiap pernyataan sesuai pendapat bapak/ibu/saudara.
3. Jika terdapat kekurangan pada media pembelajaran ini, mohon bapak/ibu/saudara memberikan saran dan masukan pada bagian yang telah disediakan.
4. Mohon untuk menuliskan nama sebagai validator (jika belum tersedia) pada bagian akhir lembar instrumen ini sebagai bukti bahwa instrumen ini dibuat sebenar-benarnya.
5. Atas bantuan bapak/ibu/saudara, kami mengucapkan terima kasih.

Pernyataan:

No	Pernyataan	Pilihan Jawaban			
		SS	S	TS	STS
1	Penggunaan media pembelajaran <i>quadruped</i> robot membantu proses pembelajaran praktik robotika.	✓			
2	Media pembelajaran <i>quadruped</i> robot dapat digunakan pendidik untuk menjelaskan materi yang disampaikan.	✓			
3	Media pembelajaran <i>quadruped</i> robot dapat digunakan pendidik untuk menarik perhatian peserta didik.	✓			
4	Media pembelajaran <i>quadruped</i> robot dapat menambah variasi materi tentang pemrograman robot pada praktik robotika.	✓			
5	Penggunaan CM-510 sebagai kontroler menambah variasi materi praktik robotika.	✓			
6	Penggunaan motor servo MX-64 dan MX-28 sebagai aktuator menambah variasi materi praktik robotika.	✓			
7	Media pembelajaran <i>quadruped</i> robot meningkatkan keaktifan peserta didik dalam pembelajaran.	✓			
8	Penggunaan media pembelajaran <i>quadruped</i> robot menumbuhkan semangat belajar peserta didik.	✓			
9	Media pembelajaran <i>quadruped</i> robot memiliki keterkaitan dengan mata kuliah lain.	✓			
10	Media pembelajaran <i>quadruped</i> robot dapat mendukung pembelajaran pada mata kuliah lain.	✓			
11	Tampilan GUI yang digunakan pada media pembelajaran menarik.		✓		
12	Informasi pada tampilan GUI yang digunakan mudah dipahami.		✓		
13	Bentuk media pembelajaran menarik		✓		
14	Peletakan komponen pada media pembelajaran tersusun dengan baik.		✓		

15	Perangkat keras pada media pembelajaran <i>quadruped</i> robot berfungsi dengan baik	✓			
16	Perangkat lunak yang digunakan berfungsi dengan baik pada media pembelajaran <i>quadruped</i> robot.	✓			
17	Panduan pengoperasian membantu penggunaan media pembelajaran dengan mudah.	✓			
18	Instalasi <i>wiring</i> aktuator motor servo MX-64 dan MX-28 pada media pembelajaran <i>quadruped</i> robot dapat dilakukan dengan mudah.	✓			
19	Perakitan media pembelajaran <i>quadruped</i> robot dapat dilakukan dengan mudah.	✓			
20	Media pembelajaran menarik untuk digunakan.	✓			
21	Media pembelajaran sesuai dengan karakteristik peserta didik.	✓			
22	Media pembelajaran sesuai dengan jenjang pendidikan peserta didik.	✓			

Kesimpulan:

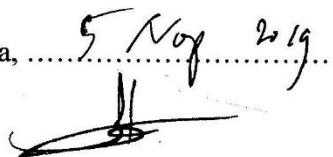
Menurut saya, Media *Quadruped* Robot berbasis CM-510 sebagai Media Pembelajaran Mata Kuliah Praktik Robotika dinyatakan:

- 4. Layak digunakan tanpa revisi
- 5. Layak digunakan dengan revisi pada saran
- 6. Tidak layak digunakan

Saran dan Perbaikan:

- *Entorical*
..... *gabungan RPS kesenian Komoditas, utama*
..... *pendeknya*
..... *Model* perl & layout ulang agar
..... *menarik*
.....
.....

Yogyakarta, 5 Nop 2019
Validator



(..... *Moh. Khairudin*)

Lampiran 3.1. Hasil Validasi Materi

SURAT PERNYATAAN VALIDASI
MEDIA PENELITIAN TUGAS AKHIR SKRIPSI

Saya yang bertanda tangan dibawah ini:

Nama : Ariadie Chandra Nugraha, ST., MT.

NIP : 197709132005011002

Jurusan : Pendidikan Teknik Elektro

Menyatakan bahwa media penelitian TAS tersebut atas nama mahasiswa:

Nama : Hilmi Musthafa Albasyir

Nim : 15518241029

Program Studi : Pendidikan Teknik Mekatronika

Judu TAS : *Quadruped Robot berbasis CM-510 sebagai Media Pembelajaran*
Mata Kuliah Praktik Robotika

Setelah dilakukan kajian atas media penelitian TAS tersebut dapat dinyatakan:

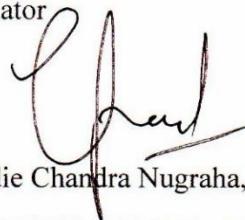
- Layak digunakan tanpa perbaikan
- Layak digunakan dengan perbaikan
- Tidak layak digunakan untuk penelitian yang bersangkutan

Dengan saran / perbaikan sebagaimana terlampir,

Demikian agar dapat digunakan sebagaimana mestinya

Yogyakarta, 04 Desember 2019

Validator



Ariadie Chandra Nugraha, ST., MT.

NIP. 197709132005011002

Catatatan:

- Beri tanda ✓

ANGKET PENILAIAN AHLI MATERI

Dalam rangka penelitian tugas akhir skripsi, saya mohon bantuan bapak/ibu/saudara untuk menjadi validator “**Quadruped Robot berbasis CM-510 sebagai Media Pembelajaran Mata Kuliah Praktik Robotika**” agar layak digunakan sebagai media pembelajaran.

Materi Pembelajaran : Praktik Robotika

Pembuat : Hilmi Musthafa Albasyir

Tanggal :

Prosedur Pengisian Instrumen Materi Pembelajaran

1. Dimohon kepada bapak/ibu/saudara untuk memberikan penilaian pada salah satu kolom di dalam angket ini berdasarkan kriteria berikut.
SS : Sangat Setuju
S : Setuju
TS : Tidak Setuju
STS : Sangat Tidak Setuju
2. Dimohon untuk memberikan tanda *check* (✓) pada salah satu pilihan jawaban dalam setiap pernyataan sesuai pendapat bapak/ibu/saudara.
3. Jika terdapat kekurangan pada media pembelajaran ini, mohon bapak/ibu/saudara memberikan saran dan masukan pada bagian yang telah disediakan.
4. Mohon untuk menuliskan nama sebagai validator (jika belum tersedia) pada bagian akhir lembar instrumen ini sebagai bukti bahwa instrumen ini dibuat sebenar-benarnya.
5. Atas bantuan bapak/ibu/saudara, kami mengucapkan terima kasih.

Pernyataan:

No	Pernyataan	Pilihan Jawaban			
		SS	S	TS	STS
1	Media pembelajaran sesuai dengan silabus.	✓			
2	Materi yang diajarkan dalam media pembelajaran <i>quadruped</i> robot sesuai dengan capaian pembelajaran peserta didik pada praktik robotika.	✓			
3	Materi yang diajarkan dalam media pembelajaran <i>quadruped</i> robot sesuai dengan bahan kajian praktik robotika.	✓			
4	Materi dapat dipahami setelah menggunakan media pembelajaran <i>quadruped</i> robot.	✓			
5	Media pembelajaran <i>quadruped</i> robot dapat mendukung materi yang terdapat dalam <i>labsheet</i> .	✓			
6	Materi disajikan runtut/sistematis.	✓			
7	Materi yang disajikan didukung dasar teori yang jelas.	✓			
8	Petunjuk penggunaan perangkat keras pada media pembelajaran <i>quadruped</i> robot dijelaskan di dalam panduan pengoperasian dan <i>labsheet</i> .	✓			
9	Petunjuk penggunaan perangkat lunak yang digunakan pada media pembelajaran <i>quadruped</i> robot dijelaskan dalam panduan pengoperasian dan <i>labsheet</i> .	✓			
10	Cara menghubungkan motor servo Dynamixel MX-64 dan MX-28 pada media pembelajaran <i>quadruped</i> robot dijelaskan dalam panduan pengoperasian.	✓			
11	Cara menghubungkan Stick PS2 pada media pembelajaran <i>quadruped</i> robot dijelaskan di dalam panduan pengoperasian.	✓			
12	Cara membuat program yang digunakan dalam pemrograman <i>quadruped</i> robot dijelaskan di dalam panduan pengoperasian.	✓			
13	Materi tentang Stick PS2 disajikan dengan jelas di dalam <i>labsheet</i> .		✓		

14	Materi tentang kontroler CM-510 disajikan dengan jelas di dalam <i>labsheet</i> .	✓			
15	Materi tentang motor servo Dynamixel MX-64 dan MX-28 disajikan dengan jelas di dalam <i>labsheet</i> .	✓			
16	Materi tentang <i>software</i> RoboPlus disajikan dengan jelas di dalam <i>labsheet</i> .	✓			
17	Materi yang disajikan dapat meningkatkan motivasi belajar peserta didik dalam pembelajaran praktik robotika.	✓			
18	Materi yang disajikan mendukung keterlibatan peserta didik dalam pembelajaran praktik robotika.	✓			
19	Materi yang disajikan sesuai dengan perkembangan intelektual peserta didik.		✓		
20	Materi yang disajikan sesuai dengan perkembangan emosional peserta didik.		✓		
21	Tata bahasa pada panduan pengoperasian dan <i>labsheet</i> sesuai dengan kaidah Bahasa Indonesia		✓		
22	Istilah-istilah pada panduan pengoperasian dan <i>labsheet</i> sudah baku.		✓		

Kesimpulan:

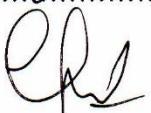
Menurut saya, materi untuk Media Pembelajaran *Quadruped* Robot berbasis CM-510 sebagai Media Pembelajaran Mata Kuliah Praktik Robotika dinyatakan:

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

Saran dan Perbaikan:

- Penulisan istilah dibuat konsisten dan sesuai penulisan bahasa, deskripsi → deskripsi
- Kode program ditulis diupayakan ditulis tdk sampai ganti baris
- Langkah-langkah dipertahankan contoh terhadap receive dr stick PS2

Yogyakarta, 6 Desember 2019
Validator


(...Ariadi Chandra N.....)

SURAT PERNYATAAN VALIDASI
MATERI PENELITIAN TUGAS AKHIR SKRIPSI

Saya yang bertanda tangan dibawah ini:

Nama : Sigit Yatmono, ST., M.T.

NIP : 197301251999031001

Jurusan : Pendidikan Teknik Elektro

Menyatakan bahwa media penelitian TAS tersebut atas nama mahasiswa:

Nama : Hilmi Musthafa Albasyir

Nim : 15518241029

Program Studi : Pendidikan Teknik Mekatronika

Judul TAS : *Quadruped Robot berbasis CM-510 sebagai Media Pembelajaran*
Mata Kuliah Praktik Robotika

Setelah dilakukan kajian atas media penelitian TAS tersebut dapat dinyatakan:

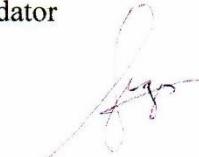
- Layak digunakan tanpa perbaikan
- Layak digunakan dengan perbaikan
- Tidak layak digunakan untuk penelitian yang bersangkutan

Dengan saran / perbaikan sebagaimana terlampir,

Demikian agar dapat digunakan sebagaimana mestinya

Yogyakarta, 1 November 2019

Validator



Sigit Yatmono, ST., M.T.

NIP. 197301251999031001

Catatatan:

- Beri tanda ✓

ANGKET PENILAIAN AHLI MATERI

Dalam rangka penelitian tugas akhir skripsi, saya mohon bantuan bapak/ibu/saudara untuk menjadi validator **“Quadruped Robot berbasis CM-510 sebagai Media Pembelajaran Mata Kuliah Praktik Robotika”** agar layak digunakan sebagai media pembelajaran.

Materi Pembelajaran : Praktik Robotika

Pembuat : Hilmi Musthafa Albasyir

Tanggal :

Prosedur Pengisian Instrumen Materi Pembelajaran

1. Dimohon kepada bapak/ibu/saudara untuk memberikan penilaian pada salah satu kolom di dalam angket ini berdasarkan kriteria berikut.
SS : Sangat Setuju
S : Setuju
TS : Tidak Setuju
STS : Sangat Tidak Setuju
2. Dimohon untuk memberikan tanda *check* (✓) pada salah satu pilihan jawaban dalam setiap pernyataan sesuai pendapat bapak/ibu/saudara.
3. Jika terdapat kekurangan pada media pembelajaran ini, mohon bapak/ibu/saudara memberikan saran dan masukan pada bagian yang telah disediakan.
4. Mohon untuk menuliskan nama sebagai validator (jika belum tersedia) pada bagian akhir lembar instrumen ini sebagai bukti bahwa instrumen ini dibuat sebenar-benarnya.
5. Atas bantuan bapak/ibu/saudara, kami mengucapkan terima kasih.

Pernyataan:

No	Pernyataan	Pilihan Jawaban			
		SS	S	TS	STS
1	Media pembelajaran sesuai dengan silabus.	✓			
2	Materi yang diajarkan dalam media pembelajaran <i>quadruped</i> robot sesuai dengan capaian pembelajaran peserta didik pada praktik robotika.	✓			
3	Materi yang diajarkan dalam media pembelajaran <i>quadruped</i> robot sesuai dengan bahan kajian praktik robotika.		✓		
4	Materi dapat dipahami setelah menggunakan media pembelajaran <i>quadruped</i> robot.	✓			
5	Media pembelajaran <i>quadruped</i> robot dapat mendukung materi yang terdapat dalam <i>labsheet</i> .	✓			
6	Materi disajikan runtut/sistematis.	✓			
7	Materi yang disajikan didukung dasar teori yang jelas.		✓		
8	Petunjuk penggunaan perangkat keras pada media pembelajaran <i>quadruped</i> robot dijelaskan di dalam panduan pengoperasian dan <i>labsheet</i> .		✓		
9	Petunjuk penggunaan perangkat lunak yang digunakan pada media pembelajaran <i>quadruped</i> robot dijelaskan dalam panduan pengoperasian dan <i>labsheet</i> .	✓			
10	Cara menghubungkan motor servo Dynamixel MX-64 dan MX-28 pada media pembelajaran <i>quadruped</i> robot dijelaskan dalam panduan pengoperasian.		✓		
11	Cara menghubungkan Stick PS2 pada media pembelajaran <i>quadruped</i> robot dijelaskan di dalam panduan pengoperasian.	✓			
12	Cara membuat program yang digunakan dalam pemrograman <i>quadruped</i> robot dijelaskan di dalam panduan pengoperasian.	✓			
13	Materi tentang Stick PS2 disajikan dengan jelas di dalam <i>labsheet</i> .		✓		

14	Materi tentang kontroler CM-510 disajikan dengan jelas di dalam <i>labsheet</i> .	✓			
15	Materi tentang motor servo Dynamixel MX-64 dan MX-28 disajikan dengan jelas di dalam <i>labsheet</i> .		✓		
16	Materi tentang <i>software</i> RoboPlus disajikan dengan jelas di dalam <i>labsheet</i> .	✓			
17	Materi yang disajikan dapat meningkatkan motivasi belajar peserta didik dalam pembelajaran praktik robotika.		✓		
18	Materi yang disajikan mendukung keterlibatan peserta didik dalam pembelajaran praktik robotika.	✓			
19	Materi yang disajikan sesuai dengan perkembangan intelektual peserta didik.	✓			
20	Materi yang disajikan sesuai dengan perkembangan emosional peserta didik.		✓		
21	Tata bahasa pada panduan pengoperasian dan <i>labsheet</i> sesuai dengan kaidah Bahasa Indonesia	✓			
22	Istilah-istilah pada panduan pengoperasian dan <i>labsheet</i> sudah baku.	✓			

Kesimpulan:

Menurut saya, materi untuk Media Pembelajaran *Quadruped* Robot berbasis CM-510 sebagai Media Pembelajaran Mata Kuliah Praktik Robotika dinyatakan:

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

Saran dan Perbaikan:

- * Ada beberapa bagian dalam latihan yg belum diulang gambar shg. jd kur
misal. latihan no 1 → langsung 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 129, 130, 131, 132, 133, 134, 135, 136, 137, 138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153, 154, 155, 156, 157, 158, 159, 160, 161, 162, 163, 164, 165, 166, 167, 168, 169, 170, 171, 172, 173, 174, 175, 176, 177, 178, 179, 180, 181, 182, 183, 184, 185, 186, 187, 188, 189, 190, 191, 192, 193, 194, 195, 196, 197, 198, 199, 200, 201, 202, 203, 204, 205, 206, 207, 208, 209, 210, 211, 212, 213, 214, 215, 216, 217, 218, 219, 220, 221, 222, 223, 224, 225, 226, 227, 228, 229, 230, 231, 232, 233, 234, 235, 236, 237, 238, 239, 240, 241, 242, 243, 244, 245, 246, 247, 248, 249, 250, 251, 252, 253, 254, 255, 256, 257, 258, 259, 260, 261, 262, 263, 264, 265, 266, 267, 268, 269, 270, 271, 272, 273, 274, 275, 276, 277, 278, 279, 280, 281, 282, 283, 284, 285, 286, 287, 288, 289, 290, 291, 292, 293, 294, 295, 296, 297, 298, 299, 300, 301, 302, 303, 304, 305, 306, 307, 308, 309, 310, 311, 312, 313, 314, 315, 316, 317, 318, 319, 320, 321, 322, 323, 324, 325, 326, 327, 328, 329, 330, 331, 332, 333, 334, 335, 336, 337, 338, 339, 340, 341, 342, 343, 344, 345, 346, 347, 348, 349, 350, 351, 352, 353, 354, 355, 356, 357, 358, 359, 360, 361, 362, 363, 364, 365, 366, 367, 368, 369, 370, 371, 372, 373, 374, 375, 376, 377, 378, 379, 380, 381, 382, 383, 384, 385, 386, 387, 388, 389, 390, 391, 392, 393, 394, 395, 396, 397, 398, 399, 400, 401, 402, 403, 404, 405, 406, 407, 408, 409, 410, 411, 412, 413, 414, 415, 416, 417, 418, 419, 420, 421, 422, 423, 424, 425, 426, 427, 428, 429, 430, 431, 432, 433, 434, 435, 436, 437, 438, 439, 440, 441, 442, 443, 444, 445, 446, 447, 448, 449, 450, 451, 452, 453, 454, 455, 456, 457, 458, 459, 460, 461, 462, 463, 464, 465, 466, 467, 468, 469, 470, 471, 472, 473, 474, 475, 476, 477, 478, 479, 480, 481, 482, 483, 484, 485, 486, 487, 488, 489, 490, 491, 492, 493, 494, 495, 496, 497, 498, 499, 500, 501, 502, 503, 504, 505, 506, 507, 508, 509, 510, 511, 512, 513, 514, 515, 516, 517, 518, 519, 520, 521, 522, 523, 524, 525, 526, 527, 528, 529, 530, 531, 532, 533, 534, 535, 536, 537, 538, 539, 540, 541, 542, 543, 544, 545, 546, 547, 548, 549, 550, 551, 552, 553, 554, 555, 556, 557, 558, 559, 560, 561, 562, 563, 564, 565, 566, 567, 568, 569, 5610, 5611, 5612, 5613, 5614, 5615, 5616, 5617, 5618, 5619, 5620, 5621, 5622, 5623, 5624, 5625, 5626, 5627, 5628, 5629, 5630, 5631, 5632, 5633, 5634, 5635, 5636, 5637, 5638, 5639, 5640, 5641, 5642, 5643, 5644, 5645, 5646, 5647, 5648, 5649, 5650, 5651, 5652, 5653, 5654, 5655, 5656, 5657, 5658, 5659, 5660, 5661, 5662, 5663, 5664, 5665, 5666, 5667, 5668, 5669, 56610, 56611, 56612, 56613, 56614, 56615, 56616, 56617, 56618, 56619, 56620, 56621, 56622, 56623, 56624, 56625, 56626, 56627, 56628, 56629, 56630, 56631, 56632, 56633, 56634, 56635, 56636, 56637, 56638, 56639, 56640, 56641, 56642, 56643, 56644, 56645, 56646, 56647, 56648, 56649, 56650, 56651, 56652, 56653, 56654, 56655, 56656, 56657, 56658, 56659, 56660, 56661, 56662, 56663, 56664, 56665, 56666, 56667, 56668, 56669, 566610, 566611, 566612, 566613, 566614, 566615, 566616, 566617, 566618, 566619, 566620, 566621, 566622, 566623, 566624, 566625, 566626, 566627, 566628, 566629, 566630, 566631, 566632, 566633, 566634, 566635, 566636, 566637, 566638, 566639, 566640, 566641, 566642, 566643, 566644, 566645, 566646, 566647, 566648, 566649, 566650, 566651, 566652, 566653, 566654, 566655, 566656, 566657, 566658, 566659, 566660, 566661, 566662, 566663, 566664, 566665, 566666, 566667, 566668, 566669, 5666610, 5666611, 5666612, 5666613, 5666614, 5666615, 5666616, 5666617, 5666618, 5666619, 5666620, 5666621, 5666622, 5666623, 5666624, 5666625, 5666626, 5666627, 5666628, 5666629, 5666630, 5666631, 5666632, 5666633, 5666634, 5666635, 5666636, 5666637, 5666638, 5666639, 5666640, 5666641, 5666642, 5666643, 5666644, 5666645, 5666646, 5666647, 5666648, 5666649, 5666650, 5666651, 5666652, 5666653, 5666654, 5666655, 5666656, 5666657, 5666658, 5666659, 5666660, 5666661, 5666662, 5666663, 5666664, 5666665, 5666666, 5666667, 5666668, 5666669, 56666610, 56666611, 56666612, 56666613, 56666614, 56666615, 56666616, 56666617, 56666618, 56666619, 56666620, 56666621, 56666622, 56666623, 56666624, 56666625, 56666626, 56666627, 56666628, 56666629, 56666630, 56666631, 56666632, 56666633, 56666634, 56666635, 56666636, 56666637, 56666638, 56666639, 56666640, 56666641, 56666642, 56666643, 56666644, 56666645, 56666646, 56666647, 56666648, 56666649, 56666650, 56666651, 56666652, 56666653, 56666654, 56666655, 56666656, 56666657, 56666658, 56666659, 56666660, 56666661, 56666662, 56666663, 56666664, 56666665, 56666666, 56666667, 56666668, 56666669, 566666610, 566666611, 566666612, 566666613, 566666614, 566666615, 566666616, 566666617, 566666618, 566666619, 566666620, 566666621, 566666622, 566666623, 566666624, 566666625, 566666626, 566666627, 566666628, 566666629, 566666630, 566666631, 566666632, 566666633, 566666634, 566666635, 566666636, 566666637, 566666638, 566666639, 566666640, 566666641, 566666642, 566666643, 566666644, 566666645, 566666646, 566666647, 566666648, 566666649, 566666650, 566666651, 566666652, 566666653, 566666654, 566666655, 566666656, 566666657, 566666658, 566666659, 566666660, 566666661, 566666662, 566666663, 566666664, 566666665, 566666666, 566666667, 566666668, 566666669, 5666666610, 5666666611, 5666666612, 5666666613, 5666666614, 5666666615, 5666666616, 5666666617, 5666666618, 5666666619, 5666666620, 5666666621, 5666666622, 5666666623, 5666666624, 5666666625, 5666666626, 5666666627, 5666666628, 5666666629, 5666666630, 5666666631, 5666666632, 5666666633, 5666666634, 5666666635, 5666666636, 5666666637, 5666666638, 5666666639, 5666666640, 5666666641, 5666666642, 5666666643, 5666666644, 5666666645, 5666666646, 5666666647, 5666666648, 5666666649, 5666666650, 5666666651, 5666666652, 5666666653, 5666666654, 5666666655, 5666666656, 5666666657, 5666666658, 5666666659, 5666666660, 5666666661, 5666666662, 5666666663, 5666666664, 5666666665, 5666666666, 5666666667, 5666666668, 5666666669, 56666666610, 56666666611, 56666666612, 56666666613, 56666666614, 56666666615, 56666666616, 56666666617, 56666666618, 56666666619, 56666666620, 56666666621, 56666666622, 56666666623, 56666666624, 56666666625, 56666666626, 56666666627, 56666666628, 56666666629, 56666666630, 56666666631, 56666666632, 56666666633, 56666666634, 56666666635, 56666666636, 56666666637, 56666666638, 56666666639, 56666666640, 56666666641, 56666666642, 56666666643, 56666666644, 56666666645, 56666666646, 56666666647, 56666666648, 56666666649, 56666666650, 56666666651, 56666666652, 56666666653, 56666666654, 56666666655, 56666666656, 56666666657, 56666666658, 56666666659, 56666666660, 56666666661, 56666666662, 56666666663, 56666666664, 56666666665, 56666666666, 56666666667, 56666666668, 56666666669, 566666666610, 566666666611, 566666666612, 566666666613, 566666666614, 566666666615, 566666666616, 566666666617, 566666666618, 566666666619, 566666666620, 566666666621, 566666666622, 566666666623, 566666666624, 566666666625, 566666666626, 566666666627, 566666666628, 566666666629, 566666666630, 566666666631, 566666666632, 566666666633, 566666666634, 566666666635, 566666666636, 566666666637, 566666666638, 566666666639, 566666666640, 566666666641, 566666666642, 566666666643, 566666666644, 566666666645, 566666666646, 566666666647, 566666666648, 566666666649, 566666666650, 566666666651, 566666666652, 566666666653, 566666666654, 566666666655, 566666666656, 566666666657, 566666666658, 566666666659, 566666666660, 566666666661, 566666666662, 566666666663, 566666666664, 566666666665, 566666666666, 566666666667, 566666666668, 566666666669, 5666666666610, 5666666666611, 5666666666612, 5666666666613, 5666666666614, 5666666666615, 5666666666616, 5666666666617, 5666666666618, 5666666666619, 5666666666620, 5666666666621, 5666666666622, 5666666666623, 5666666666624, 5666666666625, 5666666666626, 5666666666627, 5666666666628, 5666666666629, 5666666666630, 5666666666631, 5666666666632, 5666666666633, 5666666666634, 5666666666635, 5666666666636, 5666666666637, 5666666666638, 5666666666639, 5666666666640, 5666666666641, 5666666666642, 5666666666643, 5666666666644, 5666666666645, 5666666666646, 5666666666647, 5666666666648, 5666666666649, 5666666666650, 5666666666651, 5666666666652, 5666666666653, 5666666666654, 5666666666655, 5666666666656, 5666666666657, 5666666666658, 5666666666659, 5666666666660, 5666666666661, 5666666666662, 5666666666663, 5666666666664, 5666666666665, 5666666666666, 5666666666667, 5666666666668, 5666666666669, 56666666666610, 56666666666611, 56666666666612, 56666666666613, 56666666666614, 56666666666615, 56666666666616, 56666666666617, 56666666666618, 56666666666619, 56666666666620, 56666666666621, 56666666666622, 56666666666623, 56666666666624, 56666666666625, 56666666666626, 56666666666627, 56666666666628, 56666666666629, 56666666666630, 56666666666631, 56666666666632, 56666666666633, 56666666666634, 56666666666635, 56666666666636, 56666666666637, 56666666666638, 56666666666639, 56666666666640, 56666666666641, 56666666666642, 56666666666643, 56666666666644, 56666666666645, 56666666666646, 56666666666647, 56666666666648, 56666666666649, 56666666666650, 56666666666651, 56666666666652, 56666666666653, 56666666666654, 56666666666655, 56666666666656, 56666666666657, 56666666666658, 56666666666659, 56666666666660, 56666666666661, 56666666666662, 56666666666663, 56666666666664, 56666666666665, 56666666666666, 56666666666667, 56666666666668, 56666666666669, 566666666666610, 566666666666611, 566666666666612, 566666666666613, 566666666666614, 566666666666615, 566666666666616, 566666666666617, 566666666666618, 566666666666619, 566666666666620, 566666666666621, 566666666666622, 566666666666623, 566666666666624, 566666666666625, 566666666666626, 566666666666627, 566666666666628, 566666666666629, 566666666666630, 566666666666631, 566666666666632, 566666666666633, 566666666666634, 566666666666635, 566666666666636, 566666666666637, 566666666666638, 566666666666639, 566666666666640, 566666666666641, 566666666666642, 566666666666643, 566666666666644, 566666666666645, 566666666666646, 566666666666647, 566666666666648, 566666666666649, 566666666666650, 566666666666651, 566666666666652, 566666666666653, 566666666666654, 566666666666655, 566666666666656, 566666666666657, 566666666666658, 566666666666659, 566666666666660, 566666666666661, 566666666666662, 566666666666663, 566666666666664, 566666666666665, 566666666666666, 566666666666667, 566666666666668, 566666666666669, 5666666666666610, 5666666666666611, 5666666

Lampiran 4.

Hasil Uji Pengguna

ANGKET PENILAIAN

QUADRUPED ROBOT BERBASIS CM-510

SEBAGAI MEDIA PEMBELAJARAN MATA KULIAH

PRAKTIK ROBOTIKA



IDENTITAS PESERTA DIDIK

Nama : M. Kukuh Budi M.....
NIM : 175102A1023.....

JURUSAN PENDIDIKAN TEKNIK ELEKTRO

FAKULTAS TEKNIK

UNIVERSITAS NEGERI YOGYAKARTA

2019

Angket Penilaian Media

Hal : Pengisian Angket Penelitian

Kepada : Peserta didik Program Studi Pendidikan Teknik Mekatronika

Assalamu'alaikum wr. wb.

Dengan hormat,

Mohon kesediaan dan bantuan saudara untuk meluangkan waktu guna mengisi angket ini. Angket ini berguna untuk mengumpulkan data terkait dengan “*Quadruped Robot* berbasis CM-510 sebagai Media Pembelajaran Mata Kuliah Praktik Robotika”.

Angket ini bukan merupakan tes sehingga jawaban yang anda berikan tidak akan mempengaruhi nilai mata pelajaran. Jawaban yang baik adalah jawaban yang sesuai dengan kenyataan dan diisi berdasarkan hati nurani saudara, serta akan kami jamin kerahasiaannya. Kejujuran saudara dalam menjawab angket ini sangat diharapkan demi mendapatkan hasil penelitian yang maksimal.

Atas bantuan dan kerjasama dari saudara, saya ucapkan terimakasih.

Wassalamu'alaikum wr. wb.

Yogyakarta,
.....2019
Hormat Saya

Peneliti

A. Petunjuk Pengisian Angket

1. Angket ini dimaksudkan untuk mengetahui pendapat/penilaian anda sebagai pengguna media pembelajaran ***Quadruped Robot berbasis CM-510 sebagai Media Pembelajaran Mata Kuliah Praktik Robotika.***
2. Anda diharapkan memilih salah satu pilihan jawaban pada setiap pernyataan yang tersedia dengan memberikan TANDA SILANG (X) pada kolom jawaban.

Contoh:

NO	PERNYATAAN	JAWABAN			
1	Desain tata letak komponen pada <i>hardware</i> sudah rapi.	1	2	3	X

3. Jika anda ingin mengubah jawaban, maka anda memberikan tanda SAMA DENGAN (=) pada pilihan jawaban yang akan diganti dan memberikan TANDA SILANG pada kolom pengantinya.

NO	PERNYATAAN	JAWABAN			
1	Desain tata letak komponen pada <i>hardware</i> sudah rapi.	1	X	3	X

4. Keterangan jawaban:

1 = Sangat Tidak Setuju / Sangat Tidak Sesuai / Sangat Tidak Baik

2 = Tidak Setuju / Tidak Sesuai / Tidak Baik

3 = Setuju / Sesuai / Baik

4 = Sangat Setuju / Sangat Sesuai / Sangat Baik

5. Komentar atau saran anda mohon ditulis pada lebar yang telah disediakan.

Atas kesediaan anda untuk mengisi angket ini, saya ucapkan terima kasih.

B. Angket Penilaian

NO	PERNYATAAN	JAWABAN			
1	Media pembelajaran yang dibuat sesuai dengan materi pembelajaran yang diajarkan.	1	2	X	4
2	Materi yang ada dalam <i>labsheet</i> mudah dan menarik untuk dipahami.	1	X	3	4
3	Panduan pengoperasian media pembelajaran <i>quadruped</i> robot disajikan dengan lengkap.	1	2	3	X
4	Media pembelajaran <i>quadruped</i> robot dilengkapi dengan <i>labsheet</i> .	1	2	X	4
5	Materi yang disajikan sesuai dengan mata kuliah praktik robotika.	1	2	X	4
6	Materi yang disajikan berisi kompetensi yang dibutuhkan.	1	X	3	4
7	Gambar-gambar yang disajikan dalam panduan pengoperasian dan <i>labsheet</i> mempermudah dalam melaksanakan kegiatan praktikum.	1	2	3	X
8	Langkah kerja dalam panduan pengoperasian mudah untuk diikuti.	1	2	X	4
9	Bagian-bagian media pembelajaran <i>quadruped</i> robot tidak membingungkan.	1	2	X	4
10	Pengoperasian media pembelajaran <i>quadruped</i> robot dapat dilakukan dengan mudah.	1	2	X	4
11	GUI yang digunakan untuk mengoperasikan media pembelajaran <i>quadruped</i> robot menarik dan mudah digunakan.	1	2	X	4
12	Desain media pembelajaran menarik.	1	2	3	X
13	Media pembelajaran <i>quadruped</i> robot memberi tambahan pengetahuan dalam pemrograman sistem penyelaras gerak robot pada mata kuliah praktik robotika.	1	2	3	X

14	Media pembelajaran <i>quadruped</i> robot memberi tambahan pengetahuan tentang sensor pada robot.	1	2	3	<input checked="" type="checkbox"/>
15	Media pembelajaran <i>quadruped</i> robot dapat membantu dalam memahami materi pada mata kuliah lain.	1	2	<input checked="" type="checkbox"/>	4
16	Media pembelajaran <i>quadruped</i> robot memberi kesempatan untuk mempelajari pemrograman <i>Stick PS2</i> dan komunikasi Arduino dengan CM-510 pada mata kuliah praktik robotika.	1	2	<input checked="" type="checkbox"/>	4
17	Media pembelajaran <i>quadruped</i> robot memberi kesempatan untuk mempelajari pemrograman motor servo dynamixel pada mata kuliah praktik robotika.	1	2	3	<input checked="" type="checkbox"/>
18	Media pembelajaran <i>quadruped</i> robot dapat memotivasi untuk belajar pemrograman <i>Stick PS2</i> dan komunikasi Arduino dengan CM-510 pada pembelajaran praktik robotika.	1	2	<input checked="" type="checkbox"/>	4
19	Media pembelajaran <i>quadruped</i> robot dapat memotivasi untuk belajar pemrograman motor servo dynamixel pada pembelajaran praktik robotika.	1	2	3	<input checked="" type="checkbox"/>
20	Media pembelajaran <i>quadruped</i> robot menambah kompetensi dalam pemrograman robot.	1	2	<input checked="" type="checkbox"/>	4
21	Media pembelajaran <i>quadruped</i> robot dapat meningkatkan keaktifan belajar saat pembelajaran praktik robotika.	1	2	<input checked="" type="checkbox"/>	4
22	Media pembelajaran <i>quadruped</i> robot memberi tambahan wawasan mengenai robot yang digunakan di dunia industri.	1	2	<input checked="" type="checkbox"/>	4

C. Komentar dan Saran Umum

Desain dan Bentuk Robot menarik.

Yogyakarta,


(M. Kukuh Budi Martono)

Lampiran 5.

Hasil Analisis Data

Lampiran 5.1. Analisis Data Ahli Media

No	Validator	Aspek																	
		Kemanfaatan Media				Ketenggrahan Perangkat Media				Kemudahan Penggunaan Media				Jumlah					
1 1. Ir. Moh. Khairudin, M.T., Ph.D. 2. Amelia Faiziah Husna, M.Pd	Jumlah	Jumlah		%		Jumlah		%		Jumlah		%		Jumlah		%			
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16		
		4	4	4	4	4	4	4	4	4	4	40	3	3	3	3	4	4	
		4	4	4	4	4	4	4	4	4	4	37	4	3	4	3	4	4	
		Rata-rata		37,00		SL		38,50		SL		42,00		SL		21,00		SL	
		Persentase		96,25%		87,50%		96,25%		87,50%		93,75%		93,75%		93,18%			
		Jumlah Butir		10		6		6		6		6		6		22			
		Skor Maksimal		40		24		24		24		24		24		88			
		Skor Minimal		10		6		6		6		6		6		22			
		Rerata Ideal		25,00		15,00		15,00		15,00		15,00		15,00		55,00			
		Simpangan Baku		5,00		3,00		3,00		3,00		3,00		3,00		11,00			

Kriteria Penilaian	Interval Kemanfaatan Media	Interval Kelengkapan Perangkat Media	Interval Kemudahan Media	Keseluruhan	Kategori
Sangat Layak	$X > 34$	$X > 20,4$	$X > 20,4$	$X > 74,8$	SL
Layak	$28 < X \leq 34$	$16,8 < X \leq 20,4$	$16,8 < X \leq 20,4$	$61,6 < X \leq 74,8$	L
Cukup Layak	$22 < X \leq 28$	$13,2 < X \leq 16,8$	$13,2 < X \leq 16,8$	$48,4 < X \leq 61,6$	CL
Kurang Layak	$16 < X \leq 22$	$9,6 < X \leq 13,2$	$9,6 < X \leq 13,2$	$35,2 < X \leq 48,4$	KL
Tidak Layak	$X \leq 16$	$X \leq 9,6$	$X \leq 9,6$	$X \leq 35,2$	TL

Lampiran 5.2. Analisis Data Ahli Materi

No	Validator	Aspek										Total	Kategori Persentase																		
		Relevansi Materi dengan Tujuan Pembelajaran			Penyajian				Keterpaduan			Jumlah																			
Jumlah		Kategori		Penyajian		Kategori		Keterpaduan				Jumlah																			
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	Kategori	Persentase												
1	Sigit Yatmono, S.T., M.T.	4	4	3	4	4	19	SL	4	3	3	4	3	4	3	4	4	46	SL	4	3	4	4	15	SL	80	SL	90,91%			
2	Ariadie Chandra N., M.T.	4	4	4	4	4	20	SL	4	4	4	4	3	4	4	4	4	51	SL	3	3	3	3	12	L	83	SL	94,32%			
Jumlah		39																97						27		163					
Rata-rata		19,50														48,50		SL						13,50		L		81,50		SL	
Persentase		97,50%														93,27%								84,38%				92,61%			
Jumlah Butir		5														13								4		22					
Skor Maksimal		20														52								16		88					
Skor Minimal		5														13								4		22					
Rerata ideal		12,50														32,50								10,00		55,00					
Simpangan Baku		2,50														6,50								2,00		11,00					

Kriteria Penilaian	Interval Relevansi Materi dengan Tujuan Pembelajaran	Interval Penyajian	Interval Bahasa	Kesesuaian	Kategori
Sangat Layak	X > 17	X > 44,2	X > 13,6	X > 74,8	SL
Layak	14 < X ≤ 17	36,4 < X ≤ 44,2	11,2 < X ≤ 24,6	61,6 < X ≤ 74,8	L
Cukup Layak	11 < X ≤ 14	28,6 < X ≤ 36,4	8,8 < X ≤ 11,2	48,4 < X ≤ 61,6	CL
Kurang Layak	8 < X ≤ 11	20,8 < X ≤ 28,6	6,4 < X ≤ 8,8	35,2 < X ≤ 48,4	KL
Tidak Layak	X ≤ 8	X ≤ 20,8	X ≤ 6,4	X ≤ 35,2	TL

Lampiran 5.3. Analisis Data Uji Pengguna

Kriteria Penilaian	Interval Kualitas Isi dan Tujian	Interval Kualitas Pembelajaran	Interval Penggunaan	Keseluruhan	Kategori
Sangat Layak	X > 23,8	X > 34	X > 13,6	X > 74,8	SL
Layak	19,6 < X ≤ 23,8	28 < X ≤ 34	11,2 < X ≤ 24,6	61,6 < X ≤ 74,8	L
Cukup Layak	15,4 < X ≤ 19,6	22 < X ≤ 28	8,8 < X ≤ 11,2	48,4 < X ≤ 61,6	CL
Kurang Layak	11,2 < X ≤ 15,4	16 < X ≤ 22	6,4 < X ≤ 9,8	35,2 < X ≤ 48,4	KL
Tidak Layak	X ≤ 11,2	X ≤ 16	X ≤ 6,4	X ≤ 35,2	TL

Lampiran 5.4. Analisis Data Reliabilitas Instrumen

1	Lutfiana Bikhiswati Yusuf	3	3	3	4	3	3	3	4	4	4	4	3	3	3	3	3	3	3	3	3	3	3	3	
2	Neneng Thoyyibah	4	4	3	4	4	3	3	4	4	4	3	4	4	3	4	4	4	4	4	3	3	3	4	4
3	Ika Ruyanah	4	3	4	4	4	3	4	4	4	3	3	3	3	4	4	4	4	3	3	3	2	3	3	
4	Ikhsan Sahida	4	4	4	4	4	4	4	4	4	4	4	4	4	3	4	4	4	4	3	4	3	4	4	
5	Muhammad Fani Alfarizi	4	4	4	4	3	3	4	4	4	4	4	4	3	4	3	4	4	4	3	4	4	4	4	
6	Patriya Rusdi Pratama	4	4	4	4	4	4	4	4	3	3	4	4	4	3	4	4	4	4	4	4	4	4	3	
7	Amy Ayub A	4	4	4	4	4	4	4	4	4	4	4	3	4	4	4	4	4	4	4	4	4	4	4	
8	Satria Muhammad Azis	4	3	4	4	4	3	4	4	4	4	4	4	3	4	4	4	4	3	3	3	4	4		
9	Muhammad Nur Fauzan	4	3	4	4	4	4	3	4	4	4	4	4	4	4	4	4	4	4	3	3	4	4		
10	Okta Afif Rakhmawan	4	3	4	4	4	4	3	4	4	4	4	4	4	4	4	4	4	3	4	3	3	3	4	
11	Bondan Raharjo	4	3	3	4	4	3	2	4	4	3	3	4	4	4	2	3	2	4	3	3	4	3		
12	Hasan Rahmat Kamil	3	3	3	3	3	4	3	3	3	3	3	3	2	3	3	3	2	3	3	3	4	3		
13	Devi Nur Latifah	4	4	3	4	4	4	3	4	4	4	3	3	4	4	4	4	4	3	4	4	4	4		
14	Wisnu Alfianta Wibawa	4	3	4	3	3	4	4	3	3	4	4	4	4	3	3	3	3	4	3	4	3	4		
15	Nur Milati	4	3	4	3	3	4	4	3	3	4	4	4	4	3	3	3	3	4	3	4	3	4		
16	M Kukuh Budi M	3	2	4	3	3	2	3	4	4	4	4	3	3	4	3	4	3	3	3	3	3	3		
17	Sukoco	4	3	3	3	3	3	4	3	3	4	3	3	3	3	3	4	2	3	2	3	3	2		
18	Wirda Nur Alfianti	4	4	4	3	3	3	4	4	4	4	4	2	4	3	3	3	3	2	4	4	3	4		
19	Julizar Handi Wijaya	3	4	4	4	4	4	4	3	4	4	3	4	4	3	4	4	3	3	4	4	4	3		
20	Dimas Satrio P	4	4	4	4	4	4	4	4	4	4	4	3	3	3	3	3	4	4	4	3	3	4		

Scale: ALL VARIABLES

Case Processing Summary

Cases			%
	N		
Valid	8	25.0	
Excluded ^a	24	75.0	
Total	32	100.0	

- a. Listwise deletion based on all variables in the procedure.

Reliability Statistics

Cronbach's Alpha		
	N of Items	
.807	22	

Item-Total Statistics

	Scale Mean if Item Deleted	Scale Variance if Item Deleted	Corrected Item-Total Correlation	Cronbach's Alpha if Item Deleted
Item_1	73.25	28.500	-.029	.816
Item_2	73.63	23.696	.597	.783
Item_3	73.25	27.929	.088	.811
Item_4	73.63	24.554	.731	.780
Item_5	73.63	24.554	.731	.780
Item_6	73.50	22.571	.756	.771
Item_8	73.25	28.786	-.086	.818
Item_7	73.00	28.571	.000	.809
Item_13	73.38	26.839	.273	.803
Item_14	73.00	28.571	.000	.809
Item_15	73.88	27.839	.048	.818
Item_16	73.50	27.429	.153	.809
Item_17	73.38	26.839	.273	.803
Item_18	73.88	26.982	.399	.799
Item_19	73.63	27.411	.165	.809
Item_20	73.50	27.143	.205	.807
Item_21	73.88	23.554	.741	.775
Item_22	73.63	25.125	.388	.799
Item_9	73.88	25.554	.402	.797
Item_10	73.38	25.696	.497	.792
Item_11	73.75	25.929	.515	.793
Item_12	73.25	23.929	.599	.784

Lampiran 6.

Berkas Penelitian

Lampiran 6.1. Surat Keputusan Pelaksanaan Tugas Akhir Skripsi

**KEPUTUSAN DEKAN FAKULTAS TEKNIK
UNIVERSITAS NEGERI YOGYAKARTA
NOMOR : 26/PMEK/PB/VIII/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	:	Hilmi Musthafa Albasyir
NIM	:	15518241029
Prodi Studi	:	Pend. Teknik Mekatronika - S1
Judul Skripsi/TA	:	Pengembangan Quadruped Robot berbasis CM510 sebagai Media Pembelajaran Praktik Robotika

- 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 21 Agustus 2019.

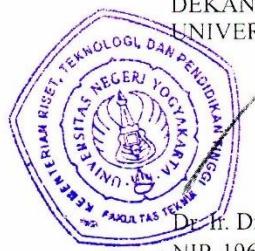
Tembusan Keputusan Dekan ini disampaikan kepada :

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 : 21 Agustus 2019

DEKAN FAKULTAS TEKNIK
UNIVERSITAS NEGERI YOGYAKARTA,



Drs. WIDARTO, M.Pd.
NIP. 19631230 198812 1 001

Lampiran 7.

Program Quadruped Robot

Lampiran 7.1. Program Menjalankan Motion

```
1 START PROGRAM
2 {
3     Motion Index Number = 1
4     WAIT WHILE ( Motion Status == TRUE )
5     ENDLESS LOOP
6     {
7         IF ( Button == U )
8         {
9             ENDLESS LOOP
10            {
11                Motion Index Number = 3
12                WAIT WHILE ( Motion Status == TRUE )
13                Motion Index Number = 4
14                WAIT WHILE ( Motion Status == TRUE )
15            }
16        }
17    }
18 }
19 }
```

Lampiran 7.2. Program Bergerak Menggunakan Stick

```
1  START PROGRAM
2  {
3      Motion Index Number = 1
4      WAIT WHILE ( Motion Status == TRUE )
5      ENDLESS LOOP
6      {
7          IF ( Remocon Data Received == TRUE )
8          {
9              data_In = Remocon RXD
10             IF ( data_In == U )
11             {
12                 Motion Index Number = 3
13                 WAIT WHILE ( Motion Status == TRUE )
14                 Motion Index Number = 4
15                 WAIT WHILE ( Motion Status == TRUE )
16             }
17         }
18     }
19 }
20 }
```

Lampiran 7.3. Program Stick PS2 Arduino

```
1. #include <PS2X_lib.h>
2.
3. #define PS2_DAT    11
4. #define PS2_CMD    12
5. #define PS2_SEL    10
6. #define PS2_CLK    13
7. #define pressures  false
8. #define rumble    false
9.
10. #define Cek_Stick false
11.
12. int Stick = 0, LX, LY, RX, RY;
13. byte type = 0;
14. byte vibrate = 0;
15.
16. PS2X ps2x;
17.
18. void setup(){
19. //repair:
20. if(Cek_Stick == true)
21. {Serial.begin(9600);}
22. else {Serial.begin(1900);}
23.
24. repair:
25.
26. delay(300); //tunggu pairing wirless
27.
28. //setup pins and settings: GamePad (clock, command, attention, data, Pressu
res?, Rumble?) check for error
29.
30. Stick = ps2x.config_gamepad (PS2_CLK, PS2_CMD, PS2_SEL, PS2_DAT
, pressures, rumble);
31.
32. if (Stick == 0){
33. Serial.print("Found Controller");
34. else {
35. Serial.print("Not Found Controller");
36. goto repair;}
37. }
38.
39. void loop() {
40. // Digunakan untuk memprogram pengiriman data ke kontroler CM-510
41. }
42.
```

```

43. void dualShock()
44. {
45. ps2x.read_gamepad(true, vibrate);
46. LY=(ps2x.Analog(PSS_LY));
47. LX=(ps2x.Analog(PSS_LX));
48. RY=(ps2x.Analog(PSS_RY));
49. RX=(ps2x.Analog(PSS_RX));
50.
51. if(ps2x.Button(PSB_START)) {Serial.println("start"); } else
52. if(ps2x.Button(PSB_SELECT)) {Serial.println("Select"); } else
53. if(ps2x.Button(PSB_PAD_UP)) {Serial.println("UP"); } else
54. if(ps2x.Button(PSB_PAD_RIGHT)) {Serial.println("Right"); } else
55. if(ps2x.Button(PSB_PAD_LEFT)) {Serial.println("Left"); } else
56. if(ps2x.Button(PSB_PAD_DOWN)) {Serial.println("Down"); } else
57. if(ps2x.Button(PSB_L1)) {Serial.println("L1"); } else
58. if(ps2x.Button(PSB_L2)) {Serial.println("L2"); } else
59. if(ps2x.Button(PSB_R1)) {Serial.println("R1"); } else
60. if(ps2x.Button(PSB_R2)) {Serial.println("R2"); } else
61. if(ps2x.Button(PSB_L3)) {Serial.println("L3"); } else
62. if(ps2x.Button(PSB_R3)) {Serial.println("R3"); } else
63. if(ps2x.Button(PSB_L2)) {Serial.println("L2"); } else
64. if(ps2x.Button(PSB_R2)) {Serial.println("R2"); } else
65. if(ps2x.Button(PSB_TRIANGLE)) {Serial.println("Segitiga"); } else
66. if(ps2x.Button(PSB_CIRCLE)) {Serial.println("O"); } else
67. if(ps2x.Button(PSB_CROSS)) {Serial.println("X"); } else
68. if(ps2x.Button(PSB_SQUARE)) {Serial.println("Kotak"); } else
69. if(LY==0) {Serial.println("L naik"); } else
70. if(LY==255) {Serial.println("L Turun"); } else
71. if(LX==255) {Serial.println("L kanan"); } else
72. if(LX==0) {Serial.println("L kiri"); } else
73. if(RY==0) {Serial.println("R naik"); } else
74. if(RY==255) {Serial.println("R Turun"); } else
75. if(RX==255) {Serial.println("R kanan"); } else
76. if(RX==0) {Serial.println("R kiri"); } else
77. }

```

Lampiran 7.3. Program Pengiriman data Arduino ke CM-510

```

1. void loop() {
2. if(Stick == 1) //repairing terus kalau gagal pair
3.     return;
4. else
5. { //DualShock Controller
6.

```

```
7.  if(Cek_Stick == true)
8.  { dualShock();}
9.  else
10. {
11.  ps2x.read_gamepad(true, vibrate);
12.  LY=(ps2x.Analog(PSS_LY));
13.  LX=(ps2x.Analog(PSS_LX));
14.  RY=(ps2x.Analog(PSS_RY));
15.  RX=(ps2x.Analog(PSS_RX));
16.
17.  if(LY==0)
18.  {
19.    Serial.write(moveForward_ptr, 3);
20.    delay(10);
21.    Serial.write(moveForward_ptr+3, 3);
22.  }
23.  else if(ps2x.Button(PSB_PAD_UP))
24.  {
25.    Serial.write(moveForward_ptr, 3);
26.    delay(10);
27.    Serial.write(moveForward_ptr+3, 3);
28.  }
29.  }
30. }
31. delay(10);
32. }
```

Lampiran 8.

Dokumentasi



Lampiran 9.

Modul Pembelajaran

Lampiran 9.1. Panduan Pengoperasian

Lampiran 9.2. *Jobsheet*

PANDUAN PENGOPERASIAN QUADRUPED ROBOT



Disusun Oleh:
Hilmi Musthafa Albasyir
NIM. 15518241029

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

KATA PENGANTAR

Assalamu'alaikum Warohmatullahi Wabarakatuh.

Segala Puji bagi Allah Tuhan semesta alam yang telah memberkahi penulis hingga dapat menyelesaikan modul pembelajaran *quadruped* robot dengan sebaik baiknya. Modul ini merupakan pelengkap bagi pembelajaran pemrograman robot pada mata kuliah Praktik Robotika. Teknologi *hardware* maupun *software* yang semakin maju menuntut proses pembelajaran pemrograman robot untuk semakin berkembang. Inovasi alat praktik selalu muncul guna menunjang proses pembelajaran, salah satunya adalah *quadruped* robot yang penulis kembangkan dalam penelitian dan pengembangan yang berjudul *Quadruped Robot Berbasis CM-510 sebagai Media Pembelajaran Mata Kuliah Praktik Robotika*.

Modul pembelajaran ini terselesaikan atas bantuan dari berbagai pihak, diantaranya Bapak Herlambang Sigit Pramono, S.T., M.Cs yang selalu membimbing penulisan dalam proses penelitian, serta adik-adik tim robot UNY, dosen JPTE, sahabat dan keluarga yang banyak membantu penulis dalam menyelesaikan penelitian ini khususnya pada saran dan penyelesaian *hardware* *quadruped* robot ini.

Penyusunan modul pembelajaran ini masih jauh dari kata sempurna, saran dan masukan yang membangun dari pembaca tentunya sangat bermanfaat bagi pembaca dan dapat digunakan sebagai referensi dalam pemrograman robot berbantuan aplikasi RoboPlus

Wassalamu'alaikum warohmatullahi Wabarakatuh.

Yogyakarta, Januari 2020

Penulis,

DAFTAR ISI

SAMPUL.....	1
KATA PENGANTAR	2
DAFTAR ISI.....	3
PANDUAN PENGOPERASIAN QUADRUPED ROBOT	5
A. AKTUATOR DYNAMIXER MX-28 DAN MX-64	7
1. Pengantar.....	7
2. Langkah Menghubungkan Aktuator Dynamixel MX-28 dan MX-64 pada Quadruped Robot.....	8
B. ARDUINO NANO DAN STICK KE KONTROLER CM-510	9
1. Pengantar.....	9
2. Program Stick PS2 pada Arduino.....	10
3. Program Arduino ke kontroler CM-510.....	12
C. SOFTWARE ROBOPLUS MANAGER	14
1. Pengantar RoboPlus Manager	14
a. Pengenalan	14
b. Deskripsi Menu	15
2. Konsep dasar Firmware pada RoboPlus Manager	17
3. Percobaan	18
a. <i>Connecting</i> Kontroler CM-510 ke RoboPlus Manager.....	18
b. Memperbarui <i>Firmware</i> Kontroler	20
c. Menguji Tabel Kontrol.....	22
d. Pemulihan <i>Firmware</i>	23
D. SOFTWARE ROBOPLUS MOTION	26
1. Pengantar RoboPlus Motion.....	26
a. Pengenalan	26
b. Deskripsi Menu	27
2. Percobaan	29
a. Membuka dan Menyimpan <i>Project</i>	29
b. Menyambungkan Robot.....	30
c. Membuat <i>Motion</i> Baru	30
E. SOFTWARE ROBOPLUS TASK	32
1. Pengantar RoboPlus Task.....	32
a. Pengenalan	32
b. Deskripsi Menu	32
2. Percobaan	33
a. Menjalankan Program RoboPlus Task.....	33
b. Awal Programming	34
3. Pemrograman RoboPlus Task	35
a. Start Program	35
b. End Program.....	35
c. Start/End dari Block/Section.....	36

d.	Calculate.....	37
e.	Load.....	38
f.	Label/Jump.....	39
g.	If/Else if/Else.....	41
h.	Endles Loop	44
i.	Loop While.....	44
j.	Loop For.....	45
k.	Break Loop.....	46
l.	Wait While	47
m.	Return Function.....	47
n.	Callback Function	50
	Referensi	52

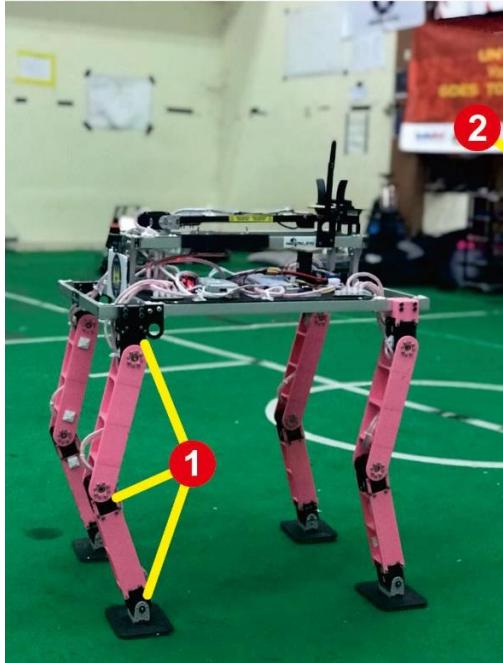
PANDUAN PENGOPERASIAN QUADRUPED ROBOT



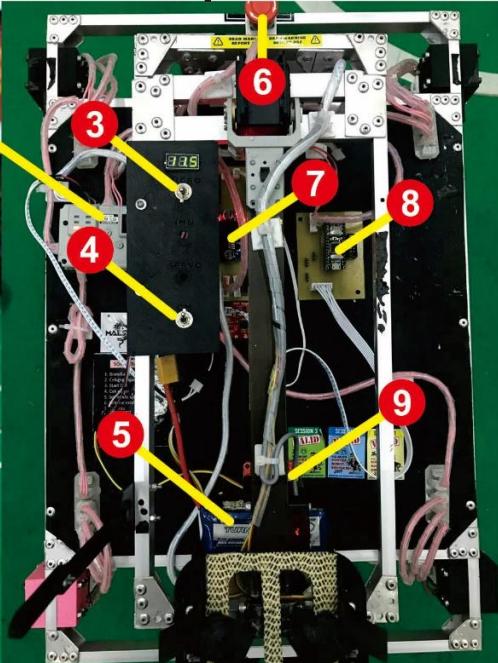
Gambar 1. Quadruped Robot

Quadruped robot merupakan robot yang dapat digunakan untuk melakukan eksperimen pemrograman robot berbantuan aplikasi Roboplus pada mata kuliah praktik robotika. *Quadruped* robot tersusun atas komponen kontroler CM-510, arduino nano, dan aktuator berupa Dynamixel MX-28 serta MX-64. Materi-materi dalam menggunakan *quadruped* robot sebagai alat praktik disesuaikan dengan silabus praktik robotika, yang bertujuan untuk melatih kompetensi pemrograman peserta didik.

Isometric view



Top view



Gambar 2. Bagian-bagian Quadruped Robot

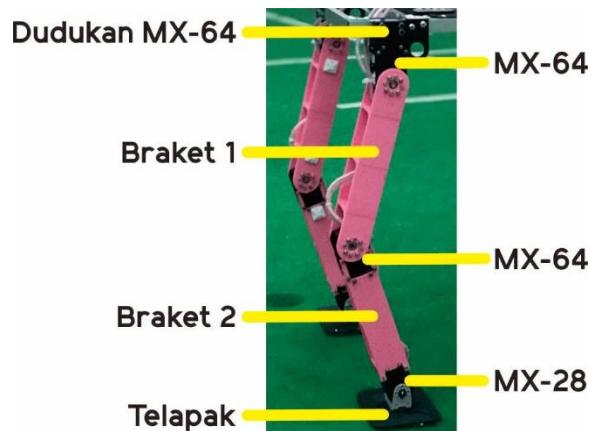
Keterangan:

1. Aktuator Servo MX-64 dan MX-28
2. Kontroler CM-510
3. Saklar Mikro kontroler
4. Saklar CM-510
5. Batrai Li-Po 3Cell 12VDC
6. *Emergency Stop*
7. Regulator tegangan
8. Arduino Nano Mikro kontroler
9. *Wireless Receiver PS2*

A. AKTUATOR DYNAMIXER MX-28 DAN MX-64

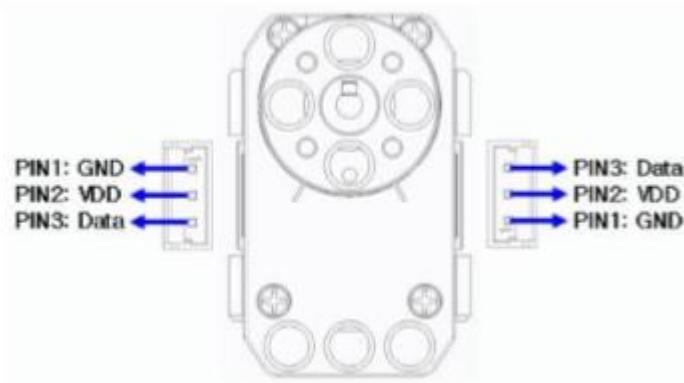
1. Pengantar

Dynamixel MX-28 dan MX-64 merupakan motor servo yang berfungsi sebagai aktuator dalam *quadruped* robot. *Quadruped* robot tersusun atas 8 buah motor servo MX-64 serta 4 buah motor servo MX-28. Setiap servo dihubungkan dengan bracket yang terbuat dari bahan PLA+ melalui proses 3D Print.



Gambar A.1. Sistem Kaki pada Quadruped Robot

Kedua servo ini memiliki 3 jenis pin berurutan yang merupakan pin VDD (sebagai pin masuknya arus listrik), GND (sebagai pin *ground*), dan Data (sebagai pin masukan dan keluaran dari atau ke motor servo). Susunan pin yang terdapat pada motor servo Dynamixel MX-28 dan MX-64 dilihat pada Gambar A.2. Sambungan antar servo Dynamixel menggunakan kabel 3poling seperti yang ditunjukkan pada Gambar A.3.



Gambar A.2. Susunan Pin pada Dynamixel MX-28 dan MX-64.
(Sumber: robotshop.com)

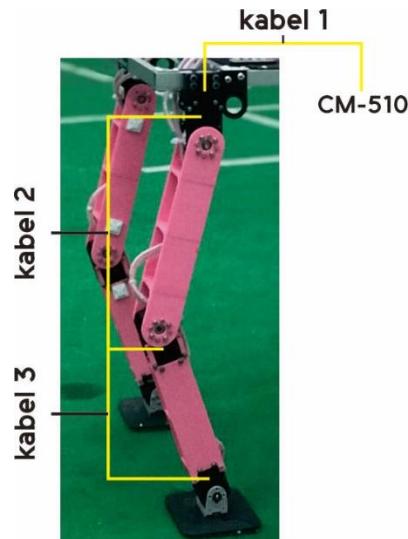


Gambar A.3. Kabel 3poling Dynamixel
(Sumber: robotis-shop.com)

2. Langkah Menghubungkan Aktuator Dynamixel MX-28 dan MX-64 pada Quadruped Robot

1. Siapkan *Quadruped* Robot dan 12 buah kabel 3poling Dynamixel.
2. Hubungkan sistem setiap kaki dari kontroler CM-510 ke setiap aktuator menggunakan kabel 3poling Dynamixel.

Pastikan Anda merangkai dengan benar sesuai dengan Gambar A.4 berikut.



Gambar A.4. Instalasi Kabel 3poling Sistem Aktuator *Quadruped* Robot

B. ARDUINO NANO DAN STICK KE KONTROLER CM-510

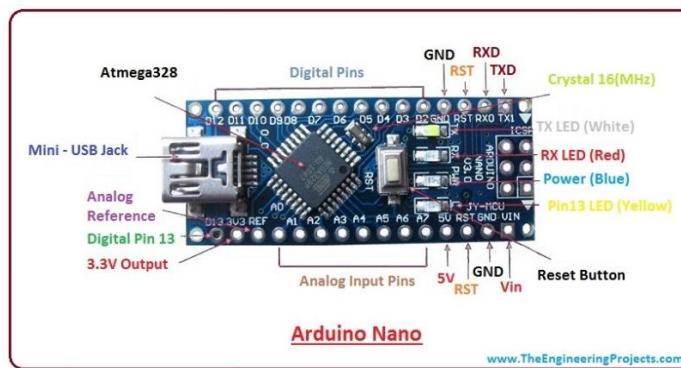
1. Pengantar

Arduino nano merupakan kontroler berbasis ATmega yang dapat digunakan untuk menerima input dari *stick wireless PlayStation 2*. *Stick wireless* terdapat dua bagian yaitu bagian *PS2 Wireless controller* dan *Wireless Receiver* yang dapat dilihat pada Gambar B.1.

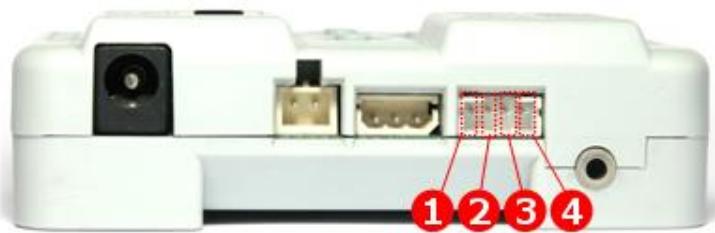


Gambar B.1. PS2 Wireless Controller dan Wireless Receiver
(Sumber: amazon.com)

Stick dapat digunakan sebagai input pada kontroler CM-510 dengan cara melewatkannya terlebih dahulu *stick* pada kontroler arduino. Selanjutnya kontroler arduino mengirimkan data serial ke kontroler CM-510. Pengiriman serial dari arduino ke CM-510 melalui fitur RX dan TX pada kedua kontroler. Susunan pin yang terdapat pada arduino dan CM-510 secara urut dapat dilihat pada Gambar B.2 dan B.3.



Gambar B.2. Susunan Pin pada Arduino
(Sumber: theengineeringprojects.com)



1. GND : Ground Level (0v)
2. VDD : Supply Voltage (2.7~3.6V)
3. RXD : Receive Signal Terminal
4. TXD : Transmit Signal Terminal

Gambar B.3. Susunan Pin Komunikasi Serial pada CM-510
(Sumber: emanual.robotis.com)

2. Program Stick PS2 pada Arduino

Pemrograman *stick* PS2 pada arduino menggunakan *library* tambahan yaitu “PS2X_lib” yang dapat di *download* dari internet. Setelah meng-*install* *library* tersebut maka pada project arduino dapat diprogram sebagai berikut.

```
1. #include <PS2X_lib.h>

2. #define PS2_DAT      11
3. #define PS2_CMD      12
4. #define PS2_SEL      10
5. #define PS2_CLK      13
6. #define pressures  false
7. #define rumble    false

8. #define Cek_Stick false

9. int Stick = 0, LX, LY, RX, RY;
10. byte type = 0;
11. byte vibrate = 0;

12. PS2X ps2x;

13. void setup(){
14. //repair:
15. if(Cek_Stick == true)
```

```

16. {Serial.begin(9600);}
17. else {Serial.begin(1900);}

18. repair:

19. delay(300); //tunggu pairing wirless

20. //setup pins and settings: GamePad (clock, command, attention, data,
Pressures?, Rumble?) check for error

21. Stick = ps2x.config_gamepad (PS2_CLK, PS2_CMD, PS2_SEL,
PS2_DAT, pressures, rumble);

22. if (Stick == 0){
23. Serial.print("Found Controller");
24. else {
25. Serial.print("Not Found Controller");
26. goto repair;
27. }

28. void loop() {

29. }

30. void dualShock()
31. {
32. ps2x.read_gamepad(true, vibrate);
33. LY=(ps2x.Analog(PSS_LY));
34. LX=(ps2x.Analog(PSS_LX));
35. RY=(ps2x.Analog(PSS_RY));
36. RX=(ps2x.Analog(PSS_RX));

37. if(ps2x.Button(PSB_START)) {Serial.println("start"); } else
38. if(ps2x.Button(PSB_SELECT)) {Serial.println("Select"); } else
39. if(ps2x.Button(PSB_PAD_UP)) {Serial.println("UP"); } else
40. if(ps2x.Button(PSB_PAD_RIGHT)) {Serial.println("Right"); } else
41. if(ps2x.Button(PSB_PAD_LEFT)) {Serial.println("Left"); }else
42. if(ps2x.Button(PSB_PAD_DOWN)) {Serial.println("Down"); }else
43. if(ps2x.Button(PSB_L1)) {Serial.println("L1"); } else
44. if(ps2x.Button(PSB_L2)) {Serial.println("L2"); } else
45. if(ps2x.Button(PSB_R1)) {Serial.println("R1"); } else

```

```

46. if(ps2x.Button(PSB_R2))      { Serial.println("R2");      } else
47. if(ps2x.Button(PSB_L3))      { Serial.println("L3");      } else
48. if(ps2x.Button(PSB_R3))      { Serial.println("R3");      } else
49. if(ps2x.Button(PSB_L2))      { Serial.println("L2");      } else
50. if(ps2x.Button(PSB_R2))      { Serial.println("R2");      } else
51. if(ps2x.Button(PSB_TRIANGLE)) {Serial.println("Segitiga"); } else
52. if(ps2x.Button(PSB_CIRCLE))  {Serial.println("O");       } else
53. if(ps2x.Button(PSB_CROSS))   {Serial.println("X");       } else
54. if(ps2x.Button(PSB_SQUARE))  {Serial.println("Kotak");    } else
55. if(LY==0)                   {Serial.println("L naik");    } else
56. if(LY==255)                 {Serial.println("L Turun");   } else
57. if(LX==255)                 {Serial.println("L kanan");  } else
58. if(LX==0)                   {Serial.println("L kiri");   } else
59. if(RY==0)                   {Serial.println("R naik");   } else
60. if(RY==255)                 {Serial.println("R Turun");  } else
61. if(RX==255)                 {Serial.println("R kanan"); } else
62. if(RX==0)                   {Serial.println("R kiri");  }
}

```

3. Program Arduino ke kontroler CM-510

Mengirim data pada kontroler arduino tidak dapat langsung dilakukan dengan mengirimkan data serial berupa data *integer* maupun *character*. Data yang dikirimkan agar dapat terbaca oleh kontroler CM-510 maupun kontroler-kontroler lain dari Robotis diperlukan pengiriman serial menggunakan pointer. Pointer adalah penunjuk suatu variabel. Karena menunjuk suatu variabel, maka pointer wajib memiliki alamat dari variabel yang ditunjuknya. Kadangkala dalam program yang besar, penghematan memori wajib untuk dilakukan. Dengan mekanisme *copy* dan *paste* nilai variabel satu kedalam variabel lain, akan sangat memboroskan memori. Dengan mekanisme pointer, suatu variabel dalam suatu fungsi dapat diakses oleh fungsi yang lain. Pointer yang dapat digunakan dalam arduino untuk mengirim kondisi ke kontroler CM-510 dapat dilihat pada tabel berikut ini.

Tabel B.1 Tabel Pointer Arduino ke CM-510

No.	Alamat Pointer CM-510 pada Arduino	Pembacaan pada CM-510
1.	{0xAA, 0x01, 0x0E, 0xAA, 0x00, 0x1E};	2 U
2.	{0xAA, 0x04, 0x5E, 0xAA, 0x00, 0x1E};	2 D
3.	{0xAA, 0x08, 0x1E, 0xAA, 0x00, 0x1E};	2 L
4.	{0xAA, 0x02, 0x7E, 0xAA, 0x00, 0x1E};	2 R
5.	{0xAA, 0x24, 0x5E, 0xAA, 0x00, 0x1E};	2 L+2
6.	{0xAA, 0x28, 0x1E, 0xAA, 0x00, 0x1E};	2 R+2
7.	{0xAA, 0x40, 0x1E, 0xAA, 0x00, 0x1E};	2 3
8.	{0xAA, 0x10, 0x1E, 0xAA, 0x00, 0x1E};	2 1
9.	{0xAA, 0x00, 0x0F, 0xAA, 0x00, 0x1E};	2 4

Penggunaan dari alamat pointer CM-510 tersebut dalam arduino perlu diawali dengan pendeklarasian variabel. Dilanjutkan dengan mengubah alamat tersebut menjadi pointer. Perhatikan contoh berikut ini.

1. const uint8_t moveFoward[6] = {0xAA, 0x01, 0x0E, 0xAA, 0x00, 0x1E};
2. const uint8_t* moveFoward_ptr = moveFoward;

Pada program tersebut alamat yang berisi untuk mengirim data pada kontroler CM-510 perintah  diawali dengan pendeklarasian variabel bertipe *const uint8_t* dengan nama “moveForward[6]”. Baris berikutnya mengubah variabel yang dideklarasikan sebelumnya menjadi variabel pointer yang akan dikirim dengan nama “moveFoward_ptr”. Pengiriman pointer ke kontroler CM-510 memiliki *baudrate* khusus yaitu 1900. Perhatikan program dibawah ini digunakan untuk mengirim data pointer ke kontroler CM-510.

1. Serial.write(moveFoward_ptr, 3);
2. delay(10);
3. Serial.write(moveFoward_ptr+3, 3);

C. SOFTWARE ROBOPLUS MANAGER

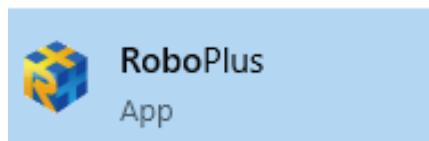
1. Pengantar RoboPlus Manager

a. Pengenalan

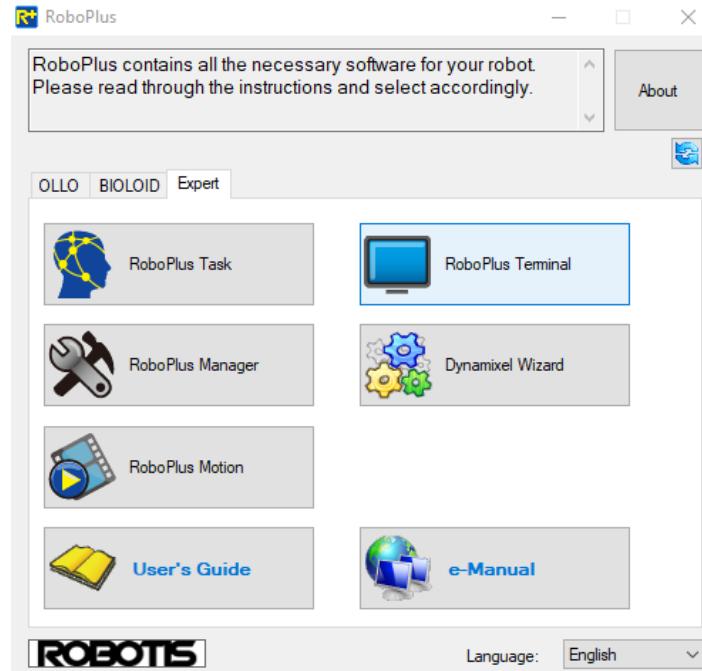
Roboplus Manager merupakan *software* keluaran ROBOTIS. *Software* ini memiliki fungsi untuk mengelola kontroler & aktuator servo dynamixel yang terdapat pada robot, memperbarui *firmware* (F/W) produk ROBOTIS ke versi yang terbaru, serta mengecek *feedback* berupa suhu, tegangan input, dan lain lain dari kontroler maupun servo dynamixel.

Software Roboplus Manager kompatibel dengan sistem operasi Windows dan OS X. Spesifikasi sistem operasi minimal yang diperlukan untuk menjalankan RoboPlus adalah Windows XP SP2 32-bit ataupun 64-bit. Sedangkan pada OS X sistem operasi minimal adalah Mac OS 10.5.

RoboPlus Manager dapat dijalankan dengan membuka aplikasi seperti yang ditampilkan pada Gambar C.1. Tampilan awal RoboPlus memberikan berbagai pilihan antara lain: 1) RoboPlus Task; 2) RoboPlus Manager; 3) Roboplus Motion; 4) RoboPlus Terminal; 5) Dynamixel Wizard; 6) User's Guide; dan 7) e-Manual. Perhatikan Gambar C.2. Produk kontroler yang dapat diakses menggunakan *software* ini antara lain: 1) CM-100; 2) CM-100A; 3) CM-150 4); CM-200; 5) CM-5; 5) CM-510; 7) CM-530; 8) CM-700; dan 9) OpenCM9.04.



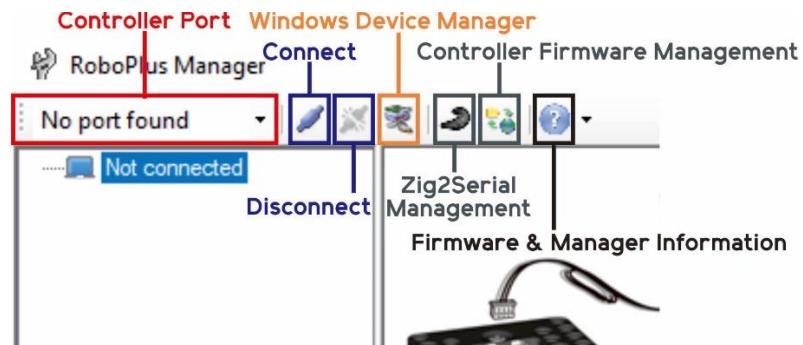
Gambar C.1. Aplikasi RoboPlus



Gambar C.2 Tampilan Awal Aplikasi RoboPlus

b. Deskripsi Menu

Terdapat beberapa Menu yang ada pada *software* RoboPlus Manager, yakni *Controller Port*, *Connect*, *Disconnect*, *Windows Device Manager*, *Zig2Serial Management*, *Controller Firmware Management*, dan *Firmware & Manager Information*. Gambar C.3 menunjukkan menu yang terdapat pada *software* RoboPlus Manager.



Gambar C.3. Menu RoboPlus Manager

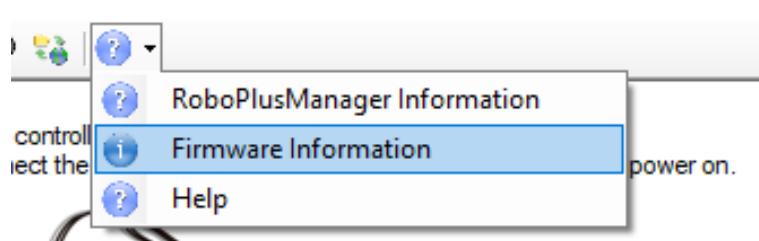
Fungsi dari Menu pada RoboPlus Manager dijelaskan pada Tabel C.1

Tabel C.1. Fungsi-fungsi Menu pada Roboplus

Menu	Fungsi
<i>Controller Port</i>	Tool yang digunakan untuk memilih lokasi <i>Port</i> dari kontroler
<i>Connect</i>	Menghubungkan RoboPlus Manager dengan kontroler sesuai dengan port yang dipilih.
<i>Disconnect</i>	Memutuskan hubungan RoboPlus Manager dengan kontroler
<i>Windows Device Manager</i>	Membuka <i>Windows Device Manager</i> untuk melihat perangkat yang tersambung pada komputer
<i>Zig2Serial Management</i>	Untuk Memanajemen Zig-100 / Zigbee
<i>Controller Firmware Management</i>	Untuk Mengupdate Firmware dari Kontroler
<i>Firmware & Manager Information</i>	Berisi informasi mengenai Firmware, Manajer, dan bantuan

Pengecekan informasi dari *firmware* dari produk dapat dengan menggunakan cara sebagai berikut.

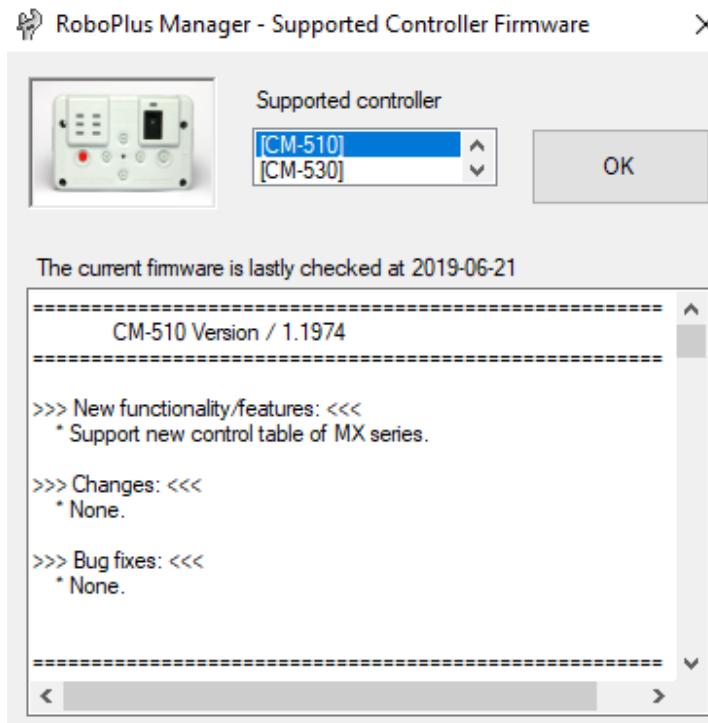
- Pada Menu pilih bagian *expand* pada *Firmware & Manager Information*
- Klik pada “*Firmware Information*” Perhatikan Gambar C.4.



Gambar C.4 Tombol *Firmware Information*

- Pilih produk yang akan dicek *firmware*-nya.

- d) Maka akan muncul versi *firmware*, tanggal perilisan, dan catatan untuk *update* dari produk yang dipilih. Perhatikan Gambar C.5 berikut.



Gambar C.5. Tampilan Informasi *Firmware*

2. Konsep dasar *Firmware* pada RoboPlus Manager

Firmware merupakan perangkat lunak (*software*) dasar yang terpasang pada suatu perangkat untuk mendapat akses perangkat keras (*hardware*). *Firmware* berisikan protokol komunikasi perangkat PC, *smartphone*, dan perangkat lainnya. Sehingga, sangat direkomendasikan untuk selalu memperbarui *firmware* dari suatu *hardware*.

Segala jenis perangkat robotis seperti kontroler, dynamixel pada dasarnya memiliki *firmware* yang sudah terpasang dari pabrik, akan tetapi belum menggunakan *firmware* terbaru sehingga perlu diperbarui. *Firmware* terbaru berguna untuk menambah beberapa fungsi ataupun memperbaiki *bug/error*. Berikut merupakan peran *firmware* pada kontroler dan dynamixel dijelaskan pada Tabel C.2.

Tabel C.2. Peran *Firmware* Pada Perangkat ROBOTIS

Perangkat	Peran
Kontroler	<ol style="list-style-type: none"> 1. Mengoperasikan program yang dibuat pada RoboPlus Task. 2. Menafsirkan data <i>motion</i> yang dibuat pada RoboPlus Motion. 3. Mengaktifkan fungsi paket komunikasi dengan PC, <i>smartphone</i>, dan dynamixel.
Dynamixel	<ol style="list-style-type: none"> 1. Mengaktifkan fungsi komunikasi dengan kontroler ketika menjalankan program maupun <i>motion</i> pada kontroler. 2. Mengaktifkan fungsi komunikasi dengan PC, <i>smartphone</i>, dan dynamixel lainnya.

Pentingnya memperbarui *firmware* dikarenakan terdapat *update firmware* yang dirilis oleh ROBOTIS. Alasan-alasan tersebut antara lain karena:

- a. Terdapat fungsi tambahan maupun modifikasi dari fungsi sebelumnya.
- b. Terdapat *software* baru, sehingga perlu kesesuaian dengan *software* tersebut.
- c. Terdapat perbaikan *bug / error*.
- d. Jika produk tidak dapat beroperasi dengan bagik atau tidak stabilnya komunikasi pada hardware.

3. Percobaan

a. *Connecting Kontroler CM-510 ke RoboPlus Manager*

Berikut ini merupakan langkah-langkah untuk menghubungkan kontroler dengan RoboPlus Manager.

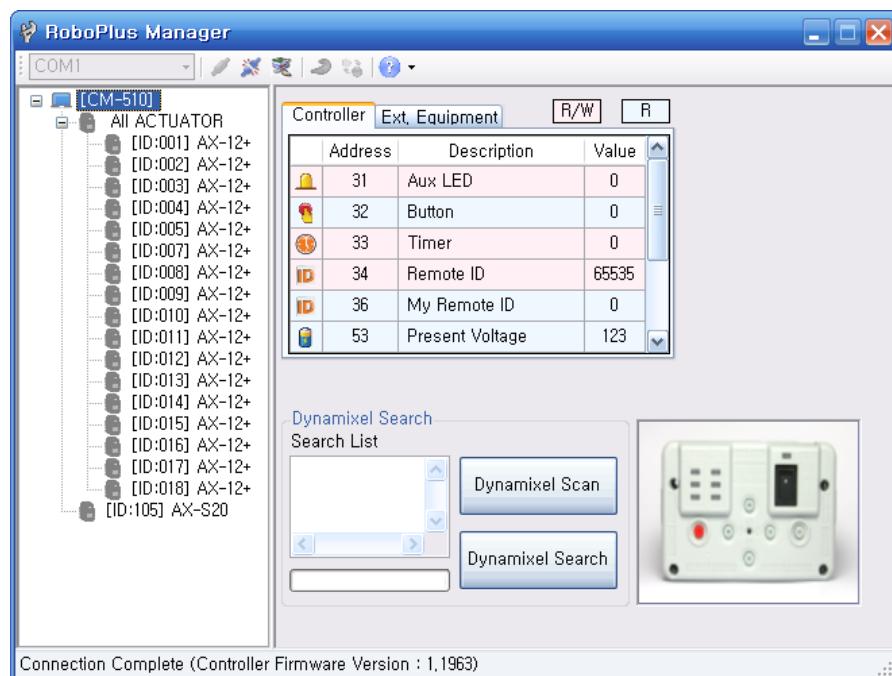
- 1) Sambungkan kontroler ke PC, lalu hidupkan kontroler.
- 2) Pilih *Controller Port* yang sesuai dengan *Port* terhubungnya kontroler dapat dengan melihat *Windows Device Manager* dengan menekan tool .
- 3) Klik tombol *Connect*  untuk menghubungkan RoboPlus Manager dengan kontroler.

- 4) Jika RoboPlus Manager tidak dapat menemukan kontroler, maka akan muncul pesan seperti pada Gambar C.6.



Gambar C.6. Tampilan Pesan RoboPlus Manager Gagal terhubung

- Cek apakah komputer terhubung dengan PC.
 - Cek apakah kontroler sudah dinyalakan.
 - Cek apakah komunikasi port yang dipilih benar.
 - Cek apakah program lain terhubung dengan port kontroler.
- 5) Jika kontroler sukses terhubung dengan RoboPlus Manager maka akan tampil seperti pada Gambar C.7.

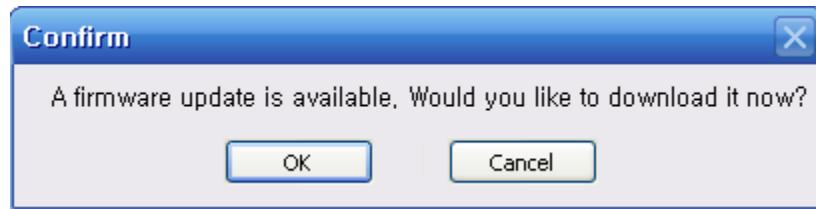


Gambar C.7. Tampilan RoboPlus Manager Terhubung dengan Kontroler

b. Memperbarui *Firmware* Kontroler

Berikut ini merupakan langkah-langkah untuk melakukan pembaruan *firmware* pada kontroler.

- 1) Ketika kontroler terhubung dengan RoboPlus Manager akan dicek versi dari *firmware*-nya. Jika terdapat *firmware* terbaru, maka akan diberikan pilihan untuk mengunduh *firmware* terbaru. Perhatikan Gambar C.8.



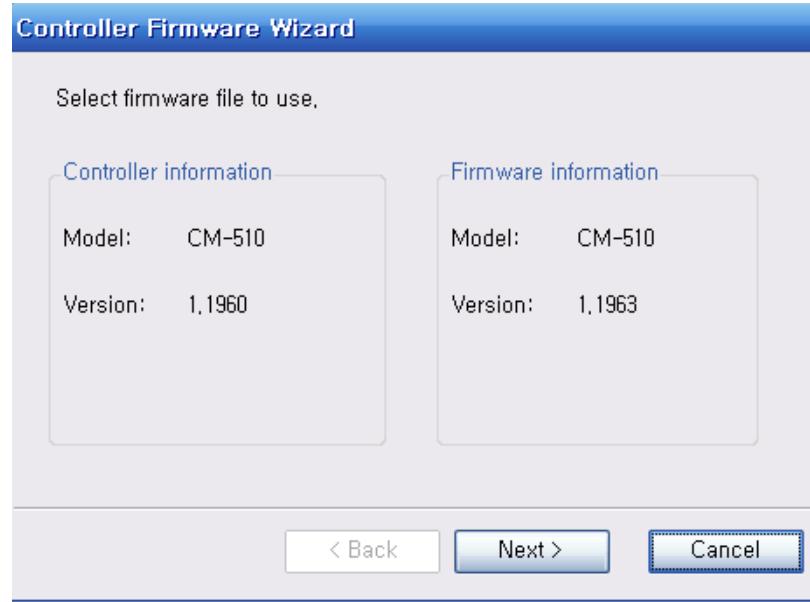
Gambar C.8. Pesan untuk Memperbarui *firmware*

- 2) Klik tombol “OK”, proses pembaruan *firmware* dimulai, lalu klik next. Perhatikan Gambar C.9.



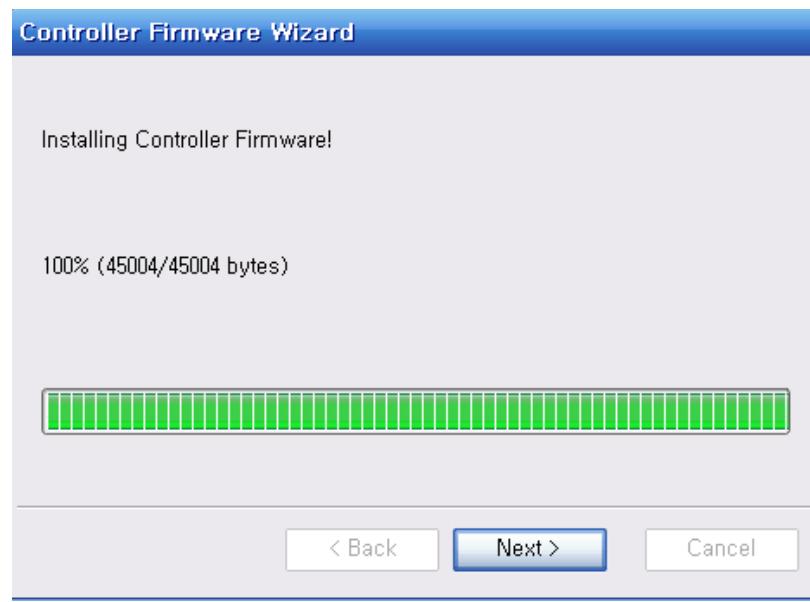
Gambar C.9. Tampilan awal *firmware update*

- 3) Selanjutnya dapat melihat versi *firmware* yang terdapat pada kontroler dan *firmware* yang terbaru seperti yang ditunjukkan pada Gambar C.10. Klik next untuk melanjutkan proses *update firmware*. Hati-hati, jangan matikan atau memutuskan hubungan kontroler ketika *firmware* sendang diperbarui.



Gambar C.10. Tampilan Versi *Firmware*

- 4) Proses *update firmware*, tunggu hingga proses selesai. Selanjutnya klik next. Perhatikan Gambar C.11.



Gambar C.11. Proses *Updating Fimware*

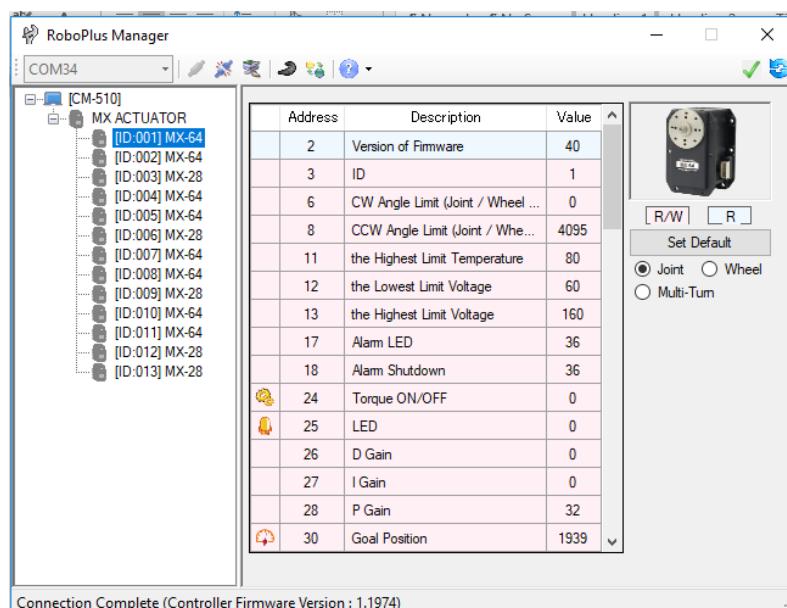
- 5) Proses pembaruan *firmware* berhasil, Klik *Finish* untuk kembali ke pengaturan kontroler



Gambar C.12. Tampilan Firmware Update Selesai

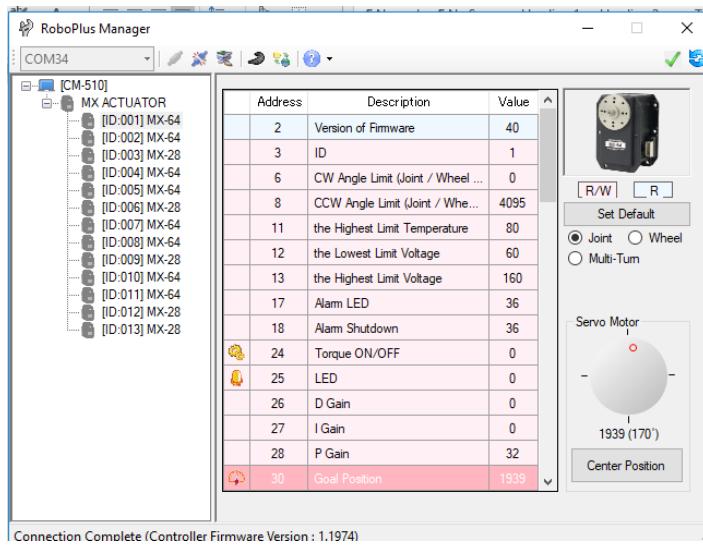
c. Menguji Tabel Kontrol

- 1) Kontroler terhubung pada RoboPlus Manager maka akan tampil kontroler dan *list* aktuator yang terhubung pada kontroler. Perhatikan Gambar C.13.
- 2) Jendela tabel kontrol mengkategorikan informasi aktuator berdasarkan urutan ID dan model perangkat. Pilihlah ID dan model perangkat yang akan di akses. Gambar C.13 menunjukkan contoh pemilihan perangkat MX-64.



Gambar C.13. Pemilihan Perangkat Aktuator MX-64

- 3) Pada daftar tabel kontrol, carilah dan pilih “Goal Position”.
- 4) Akses Goal Position (posisi) pada bagian kanan bawah dari jendela tabel kontrol untuk mengakses posisi motor servo MX-64 yang dipilih. Pada saat Anda merubah nilai data pada *software*, maka posisi motor servo MX-64 juga akan bergerak mengikuti.



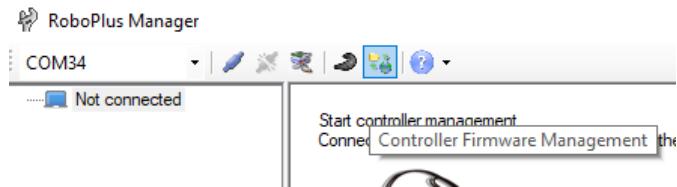
Gambar C.14. Menu “Goal Position”

- 5) Klik pada Tombol “Center Position”. Tombol ini berguna untuk menggerakkan sudut aktuator servo pada *center point*.

d. Pemulihan *Firmware*

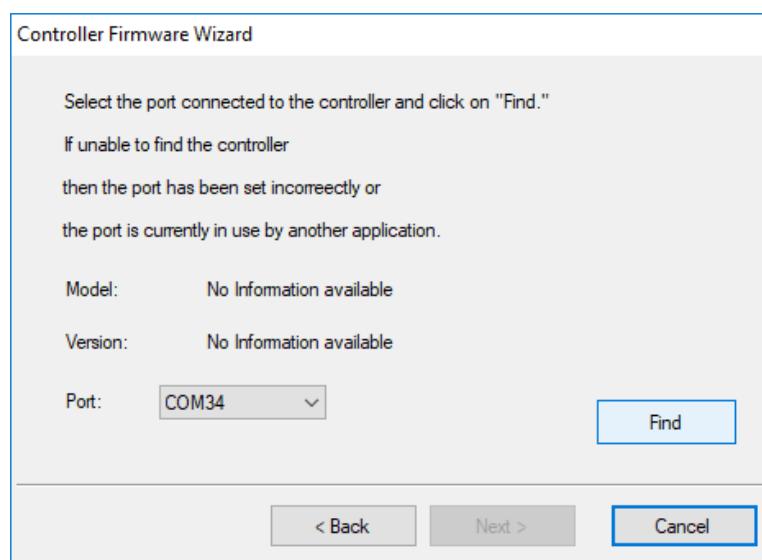
Jika terdapat masalah terhadap *firmware* kontroler, RoboPlus Manager dapat digunakan untuk *recover* (memperbaiki) *firmware*.

- 1) Hubungkan kontroler ke PC.
- 2) Jalankan aplikasi RoboPlus Manager dan buka *Controller Firmware Management Wizard*. Pada menu bar klik tombol *Firmware management*. Perhatikan Gambar C.15.



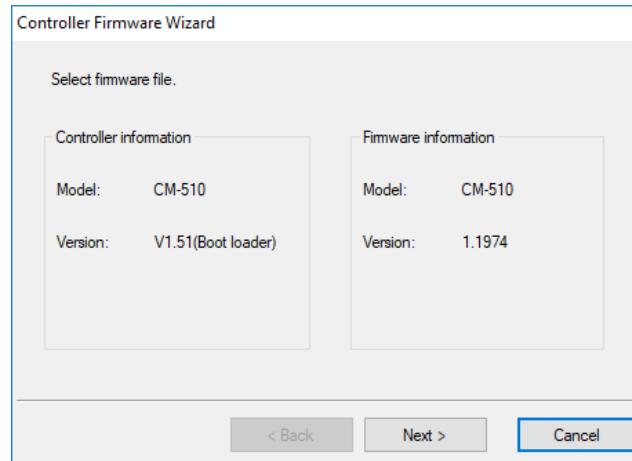
Gambar C.15. Menu *Controller Firmware Management*

- 3) Klik next pada *Controller Firmware Wizard*
- 4) Pilih nomer Port dari kontroler. Kerena *firmware error* sehingga kontroler tidak dapat terdeteksi secara otomatis. Pilih *port* yang sesuai. Perhatikan Gambar C.16.



Gambar C.16. Memilih Port Kontroler

- 5) Matikan lalu hidupkan kontroler jika *port* tidak terdeteksi
- 6) Cek informasi dari kontroler. Ketika kontroler terdeteksi, informasi dari *firmware* kontroler dan informasi *firmware recovery* akan ditampilkan. Perhatikan Gambar C.17. Klik pada tombol *next* untuk melanjutkan proses *recovery*.



Gambar C.17. Tampilan informasi *firmware*

- 7) *Firmware recovery* dimulai, tunggu hingga proses slesai.

D. SOFTWARE ROBOPLUS MOTION

1. Pengantar RoboPlus Motion

a. Pengenalan

RoboPlus Motion merupakan *software* keluaran ROBOTIS. *Software* ini memiliki fungsi untuk membuat *motion* pada kontroler. *Motion* merupakan rekaman beberapa posisi dan kecepatan dari kumpulan aktuator motor servo yang diterjemahkan sebagai pergerakan robot.

Software RoboPlus Motion kompatibel dengan sistem operasi Windows dan OS X. Spesifikasi sistem operasi minimal yang diperlukan untuk menjalankan RoboPlus adalah Windows XP SP2 32-bit ataupun 64-bit. Sedangkan pada OS X sistem operasi minimal adalah Mac OS 10.5.

Proses pembuatan *motion* meliputi beberapa tahap pada *software* dan harus didownloadkan ke kontroler robot. *File* ekstensi dari aplikasi RoboPlus Motion adalah “.mtn” (tanpa tanda petik). File *motion* memiliki hubungan dengan file yang dibuat pada aplikasi RoboPlus Task, dimana file ini merupakan kode program yang akan mengatur robot untuk bergerak dan membaca masukan sensor. Untuk dapat melakukan gerakan robot, program *task* akan memanggil data *motion* yang ada dalam file *motion* yang telah didownloadkan ke kontroler. Proses pembuatan *motion* diawali dengan mensetting *resolution* dari setiap aktuator motor servo langkah-langkah tersebut antara lain:

- 1) Membuat Project baru dan menghubungkan kontroler dengan komputer.
- 2) Mensetting resolusi dari aktuator servo yang digunakan.
- 3) Pada menu “Edit All Page” setting resolusi dari servo aktuator yang digunakan seperti yang ditampilkan pada Gambar D.1.

	Enable	Resolution
ID[0]	<input type="checkbox"/>	1024 (300')
ID[1]	<input checked="" type="checkbox"/>	1024 (300') ▼
ID[2]	<input checked="" type="checkbox"/>	1024 (300') 4096 (250')
ID[3]	<input checked="" type="checkbox"/>	4096 (360') ▼
ID[4]	<input checked="" type="checkbox"/>	1024 (300')
ID[5]	<input checked="" type="checkbox"/>	1024 (300')
ID[6]	<input checked="" type="checkbox"/>	1024 (300')

Gambar D.1 Mensetting Resolusi Atuator Servo

- 4) Dilanjutkan dengan membuat motion baru sesuai yang dibutuhkan.

b. Deskripsi Menu

RoboPlus Motion didesain untuk merekam posisi sudut pada robot yang dimasukkan pada setiap “step” yang akan dijalankan oleh kontroler. Menu pembuatan *motion* pada RoboPlus Motion dibagi menjadi tiga bagian yaitu: (1) *Pose Editing*; (2) *Step Editing*; dan (3) *Page Editing*.

a. *Pose Editing*

Pose adalah posisi robot pada suatu titik waktu. Pose merupakan kumpulan dari posisi sudut motor servo robot yang diperlukan untuk menghasilkan posisi yang diinginkan pada suatu waktu. Pada menu “Basic Pose Editing” terdapat dua tabel editor yaitu “Pose of Step” dan “Pose of Robot” yang ditunjukkan pada Gambar D.2.

Basic Pose Editor		Pose Utility	Edit All page																																																																					
<Pose of Step>		<Pose of Robot>																																																																						
<table border="1"> <thead> <tr> <th></th> <th>Value</th> <th></th> </tr> </thead> <tbody> <tr><td>ID[1]</td><td>1868</td><td><input checked="" type="checkbox"/></td></tr> <tr><td>ID[2]</td><td>1709</td><td><input checked="" type="checkbox"/></td></tr> <tr><td>ID[3]</td><td>1942</td><td><input checked="" type="checkbox"/></td></tr> <tr><td>ID[4]</td><td>2221</td><td><input checked="" type="checkbox"/></td></tr> <tr><td>ID[5]</td><td>2381</td><td><input checked="" type="checkbox"/></td></tr> <tr><td>ID[6]</td><td>2266</td><td><input checked="" type="checkbox"/></td></tr> <tr><td>ID[7]</td><td>2198</td><td><input checked="" type="checkbox"/></td></tr> <tr><td>ID[8]</td><td>2320</td><td><input checked="" type="checkbox"/></td></tr> <tr><td>ID[9]</td><td>2183</td><td><input checked="" type="checkbox"/></td></tr> <tr><td>ID[10]</td><td>1898</td><td><input checked="" type="checkbox"/></td></tr> <tr><td>ID[11]</td><td>1746</td><td><input checked="" type="checkbox"/></td></tr> <tr><td>ID[12]</td><td>1948</td><td><input checked="" type="checkbox"/></td></tr> <tr><td>ID[13]</td><td>2083</td><td><input checked="" type="checkbox"/></td></tr> </tbody> </table>			Value		ID[1]	1868	<input checked="" type="checkbox"/>	ID[2]	1709	<input checked="" type="checkbox"/>	ID[3]	1942	<input checked="" type="checkbox"/>	ID[4]	2221	<input checked="" type="checkbox"/>	ID[5]	2381	<input checked="" type="checkbox"/>	ID[6]	2266	<input checked="" type="checkbox"/>	ID[7]	2198	<input checked="" type="checkbox"/>	ID[8]	2320	<input checked="" type="checkbox"/>	ID[9]	2183	<input checked="" type="checkbox"/>	ID[10]	1898	<input checked="" type="checkbox"/>	ID[11]	1746	<input checked="" type="checkbox"/>	ID[12]	1948	<input checked="" type="checkbox"/>	ID[13]	2083	<input checked="" type="checkbox"/>	<table border="1"> <thead> <tr> <th></th> <th>Value</th> </tr> </thead> <tbody> <tr><td>ID[1]</td><td>OFF</td></tr> <tr><td>ID[2]</td><td>OFF</td></tr> <tr><td>ID[3]</td><td>OFF</td></tr> <tr><td>ID[4]</td><td>OFF</td></tr> <tr><td>ID[5]</td><td>OFF</td></tr> <tr><td>ID[6]</td><td>OFF</td></tr> <tr><td>ID[7]</td><td>OFF</td></tr> <tr><td>ID[8]</td><td>OFF</td></tr> <tr><td>ID[9]</td><td>OFF</td></tr> <tr><td>ID[10]</td><td>OFF</td></tr> <tr><td>ID[11]</td><td>OFF</td></tr> <tr><td>ID[12]</td><td>OFF</td></tr> <tr><td>ID[13]</td><td>OFF</td></tr> </tbody> </table>		Value	ID[1]	OFF	ID[2]	OFF	ID[3]	OFF	ID[4]	OFF	ID[5]	OFF	ID[6]	OFF	ID[7]	OFF	ID[8]	OFF	ID[9]	OFF	ID[10]	OFF	ID[11]	OFF	ID[12]	OFF	ID[13]	OFF
	Value																																																																							
ID[1]	1868	<input checked="" type="checkbox"/>																																																																						
ID[2]	1709	<input checked="" type="checkbox"/>																																																																						
ID[3]	1942	<input checked="" type="checkbox"/>																																																																						
ID[4]	2221	<input checked="" type="checkbox"/>																																																																						
ID[5]	2381	<input checked="" type="checkbox"/>																																																																						
ID[6]	2266	<input checked="" type="checkbox"/>																																																																						
ID[7]	2198	<input checked="" type="checkbox"/>																																																																						
ID[8]	2320	<input checked="" type="checkbox"/>																																																																						
ID[9]	2183	<input checked="" type="checkbox"/>																																																																						
ID[10]	1898	<input checked="" type="checkbox"/>																																																																						
ID[11]	1746	<input checked="" type="checkbox"/>																																																																						
ID[12]	1948	<input checked="" type="checkbox"/>																																																																						
ID[13]	2083	<input checked="" type="checkbox"/>																																																																						
	Value																																																																							
ID[1]	OFF																																																																							
ID[2]	OFF																																																																							
ID[3]	OFF																																																																							
ID[4]	OFF																																																																							
ID[5]	OFF																																																																							
ID[6]	OFF																																																																							
ID[7]	OFF																																																																							
ID[8]	OFF																																																																							
ID[9]	OFF																																																																							
ID[10]	OFF																																																																							
ID[11]	OFF																																																																							
ID[12]	OFF																																																																							
ID[13]	OFF																																																																							
																																																																								

Gambar D.2. Basic Pose Editor

Pose of Step merupakan data posisi sudut dari aktuator servo robot yang terekam. Sedangkan *Pose of Robot* adalah nilai posisi dari servo motor pada robot. Ketika posisi robot diubah maka nilai pada *Pose of Robot* akan berubah sesuai dengan posisi robot tersebut.

b. Step Editing

Sebuah “Motion Step” merupakan kumpulan dari *key-frame* yang dibutuhkan untuk membuat sebuah motion pada robot. *Key-frame* merupakan langkah-langkah dari *motion* robot. *Key-frame* terdiri atas: 1) posisi dan 2) kecepatan motor servo, yang dapat diterjemahkan sebagai Dimana (Posisi) selama Kapan (Waktu). Step Editing memiliki beberapa fungsi seperti *Add Step*, *Insert Step*, *Delete Step*, dan *Move Step*. Sedangkan pada setiap *Step* terdapat “Pause” dan “Time”. “Pause” adalah waktu antara akhir dari *Step* saat ini dengan awal dari *Step* selanjutnya.

	Pause	Time	
▶ STEP 0	0	1	          

Gambar D.3 Step Editing

c. *Page Editing*

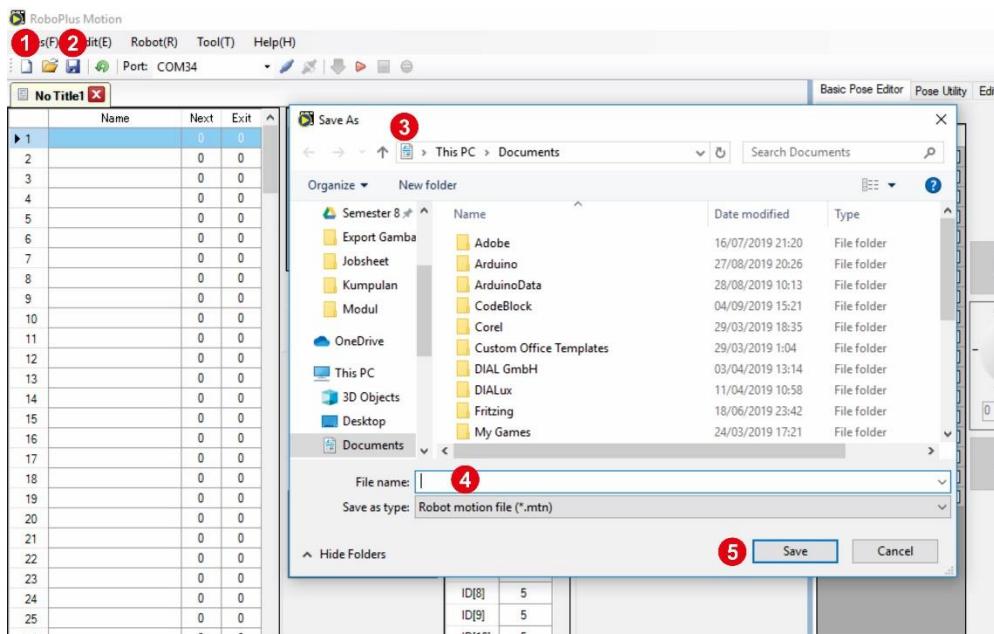
Page Editing merupakan tabel yang berisi *motion* yang tersimpan pada RoboPlus Motion. Data dari *motion* terdiri dari 255 *pages*. Namun pada beberapa kontroler hanya terbatas pada 127 *pages*.

2. Percobaan

a. Membuka dan Menyimpan Project

- Klik pada Aplikasi RoboPlus dengan icon seperti pada Gambar C.1.
- Klik pada *icon* RoboPlus Motion untuk menjalankan aplikasi.
- Klik pada *icon* “New”, selanjutnya klik *icon* “Save” untuk menyimpan project baru.
- Isikan “File name” sesuai dengan keinginan anda, lalu klik “Save”.

Perhatikan Gambar D.4



Gambar D.4. Membuat dan Menyimpan *Project Motion* Baru

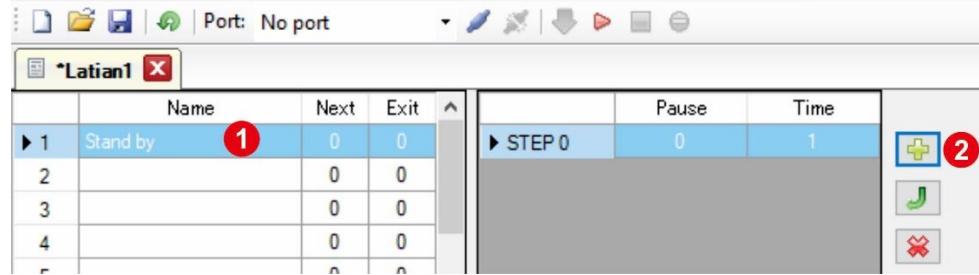
b. Menyambungkan Robot

- Hubungkanlah kontroler dengan komputer, lalu hidupkan kontroler.
- Klik pada bagian Port, pilihlah Port number yang sesuai dengan kontroler.
- Klik pada icon “Connect”, tunggu hingga proses singkronisasi selesai.

c. Membuat *Motion* Baru

- Double Klik* pada motion page nomer 1, lalu beri nama sesuai keinginan anda. Setelah selesai memberi nama tekan “Enter” pada keyboard.
- Klik pada icon “Add Step” untuk menambah *Step* pada *motion page*.

Perhatikan pada Gambar D.5



Gambar D.5 Membuat *Motion* Baru

- c. Klik kanan pada “Pose of Robot”, lalu klik “Select All”. Selanjutnya matikan *Torque* dari aktuator motor servo dengan menekan *icon* .
- d. Posisikan aktuator servo pada robot sesuai dengan kebutuhan anda. Setelah memposisikan aktuator servo anda dapat menghidupkan *Torque* pada aktuator secara bersamaan maupun satu persatu dengan cara memilih nomor ID servo pada “Pose of Robot”.
- e. Tekan *icon* tanda panah hijau yang mengarah dari “Pose of Robot” ke “Pose of Step” untuk menyimpan data posisi sudut aktuator ke *Step* yang dipilih.
- f. Dalam membuat sebuah motion dapat terdiri dari lebih dari satu *Step*.

E. SOFTWARE ROBOPLUS TASK

1. Pengantar RoboPlus Task

a. Pengenalan

RoboPlus Task merupakan produk keluaran ROBOTIS. *Software* ini memiliki fungsi untuk membuat kode program algoritma dan logika yang dapat disebut *task*, serta memanggil *motion* yang telah dibuat pada aplikasi RoboPlus Motion. Dalam ilmu komputer *task* diartikan sebagai aktivitas yang berhubungan untuk menyelesaikan suatu pekerjaan. Sedangkan dalam RoboPlus, kode *task* dimaknai sebagai kode program yang digunakan robot untuk menyelesaikan pergerakan robot.

Software RoboPlus Task kompatibel dengan sistem operasi Windows dan OS X. Spesifikasi sistem operasi minimal yang diperlukan untuk menjalankan RoboPlus adalah Windows XP SP2 32-bit ataupun 64-bit. Sedangkan pada OS X sistem operasi minimal adalah Mac OS 10.5. Robot yang diprogram menggunakan RoboPlus Task akan beroperasi berdasarkan kode *task* yang telah dibuat oleh pengguna. *File* ekstensi dari *file task* adalah “.tsk” (tanpa tanda petik).

b. Deskripsi Menu

RoboPlus Task memiliki *tools* untuk menunjang *programming*. Terdapat *tools* yaitu: *New*, *Open*, *Save*, *Print*, *Cut*, *Copy*, *Paste*, *Find Name*, *Check Rule*, *Download Program*, *Change Controller*, dan *Port Controller*. Bagian-bagian tersebut masing-masing memiliki fungsi yang mendukung dalam *programming*. Fungsi dari bagian-bagian tersebut dijelaskan pada Tabel E.1

Tabel E.1 Fungsi-fungsi *Tools* pada RoboPlus Task

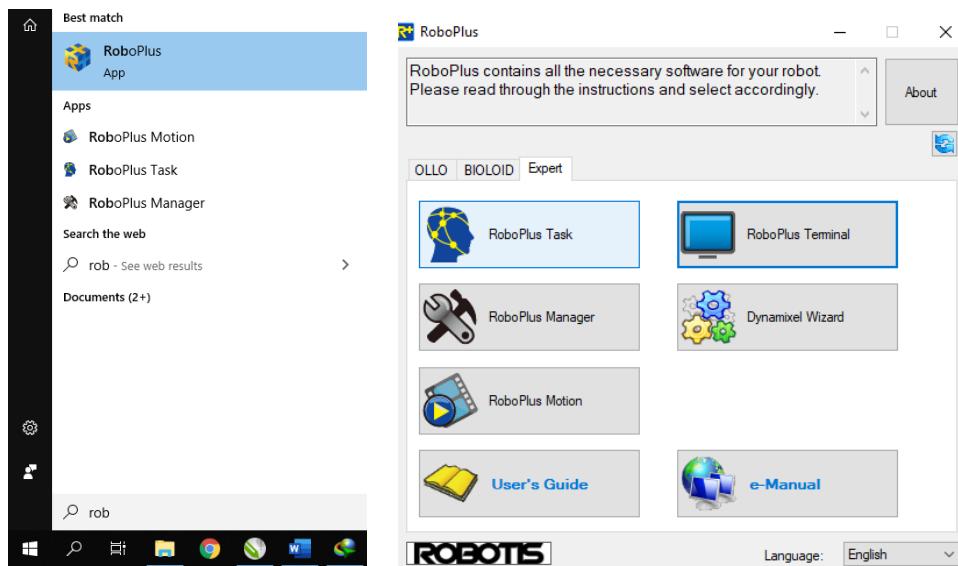
Tools	Fungsi
<i>New</i>	Membuat Program baru
<i>Open</i>	Membuka Program yang tersimpan
<i>Save</i>	Menyimpan Program
<i>Print</i>	Mencetak program
<i>Cut</i>	Memotong/memindahkan kode program

<i>Copy</i>	Menyalin kode program
<i>Paste</i>	Menampilkan hasil dari <i>copy/cut</i>
<i>Find Name</i>	Mencari kata
<i>Check Rule</i>	Mengecek program apakah terdapat <i>error</i>
<i>Download Program</i>	Memasukkan program ke kontroler
<i>Change Controller</i>	Mengubah seri kontroler
<i>Port Controller</i>	Memilih <i>Port</i> dari kontroler

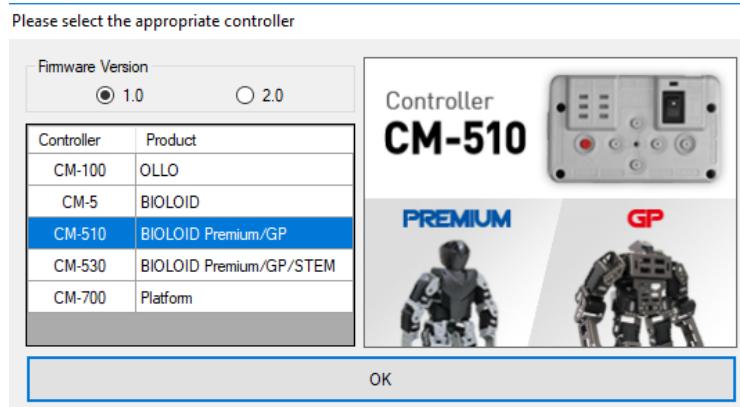
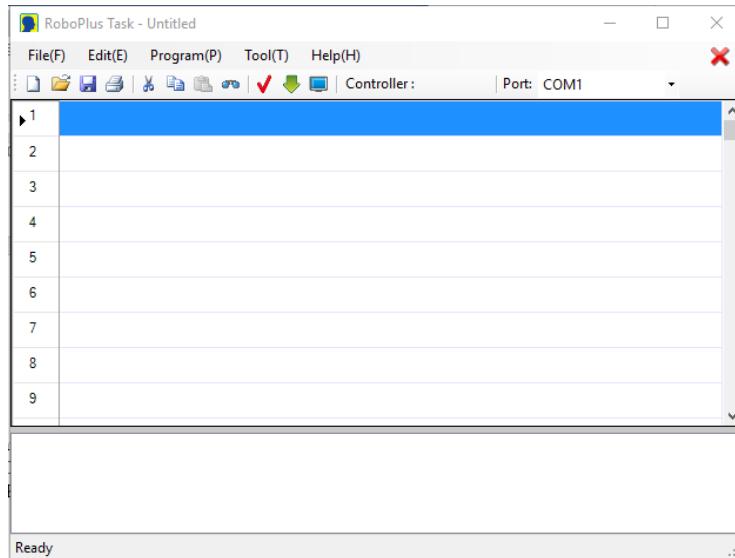
2. Percobaan

a. Menjalankan Program RoboPlus Task

- 1) Seperti yang anda lihat pada kambar di bawah ini, klik “Start” cari “RoboPlus” lalu pada *interface* awal RoboPlus klik RoboPlus Task.



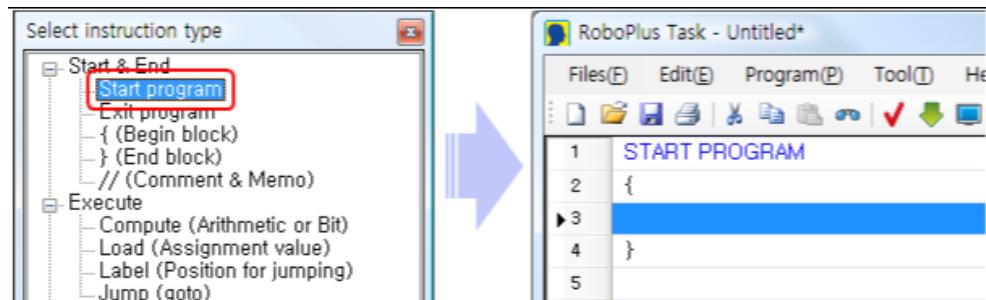
- 2) Tampilan awal RoboPlus Task dapat dilihat pada gambar dibawah. *Double click* pada baris kosong atau tekan “Enter”, pada <Select Controller> window, pilih kontroler yang anda gunakan. Klik tombok “OK”.



3) Klik icon Save untuk menyimpan dari RoboPlus Task.

b. Awal Programming

- 1) Setelah memilih kontroler pilih “Start program” dari <Select Instruction Type> window, “Start Program” akan secara otomatis dibuat pada RoboPlus Task.



- 2) RoboPlus Task siap diisi dengan program sesuai kebutuhan anda.

3. Pemrograman RoboPlus Task

a. Start Program

“Start Program” didesain untuk memulai sebuah program. Berdasarkan *line number*, program akan selalu mulai pada poin tersebut. “Start Program” pada pemrograman bahasa C dapat diartikan seperti “main” *function*.

1) Penggunaan

- “Start Program” dijalankan berdasarkan dengan *line number*-nya.
- Sebuah program tidak dapat memiliki perintah lebih dari satu “Start Program”
- Bagian dari perintah harus memiliki *bracket* yang berupa kurung kurawal.
- Program akan berakhir ketika mencapai akhir *bracket* “}”.

2) Contoh

```

1 START PROGRAM
2 {
3
4
5
6 }

```

b. End Program

Jika perintah ini dipanggil pada saat program berjalan, maka program akan secara langsung diberhentikan. Ada 2 cara untuk melakukan “End Program”

yaitu *Natural Close* dan *Force Exit*. Kegunaan dari perintah ini adalah untuk mengakhiri program sesuai dengan yang anda inginkan.

- When the end of "Start Program" is reached (Natural Close)

```
1  START PROGRAM
2  {
3
4
5
6 } ←
```

- When "End Program" is called (Forced Exit)

```
1  START PROGRAM
2  {
3
4  END PROGRAM ←
5
6 }
```

Contoh berikut ini program akan berhenti ketika *Touch* (Sensor Sentuh) yang terhubung pada PORT 3 disentuh.

```
IF ( PORT[3] == TRUE )
{
  END PROGRAM
}
```

c. Start/End dari Block/Section

Sebuah *block* atau *section* (diidentifikasi dengan “{“ dan “}”) adalah sebuah grup yang berisi perintah. Semua perintah pada *block* harus kesamaan. Konsep dari *block* pada RoboPlus Task sama seperti pada pemrograman bahasa C.

1) Penggunaan

- Setiap *block* diawali dengan kurung kurawal buka ({) dan diakhiri dengan kurung kurawal tutup (}).

1	START PROGRAM
2	{
3	ENDLESS LOOP
4	{
5	
6	
7	}

- Setiap *block* harus “dimiliki” oleh sebuah perintah. *Blocks* tidak dapat digunakan secara independen tanpa perintah. Berikut ini merupakan beberapa perintah yang sering digunakan dan diikuti dengan *block*, yaitu (1) Start Program; (2) If / Else if / Else; (3) Endless Loop; (4) Loop For; (5) Loop While; (6) Callback Function; dan (7) Function.
- Jika sebuah perintah dibuat hanya dengan satu baris, hal tersebut dapat mengabaikan *block bracket*.

1	
2	IF ( > 0)
3	 = 
4	

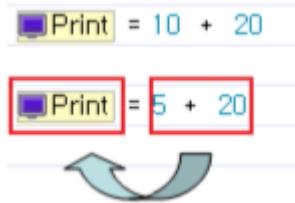
2) Contoh

Perintah “Start Program” dan “Endless Loop” harus diikuti dengan *blocks*, seperti yang ditunjukkan berikut.

1	START PROGRAM
2	{
3	ENDLESS LOOP
4	{
5	 = 10
6	}
7	}

d. Calculate

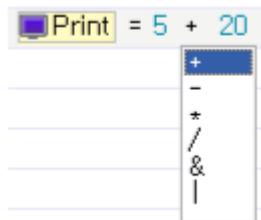
“Calculate” merupakan perintah yang digunakan untuk operasi aritmatika pada dua bilangan.



“Calculate” dapat melakukan operasi dasar seperti penjumlahan, pengurangan, perkalian, dan pembagian. “Calculate” juga dapat melakukan operasi *Bit* seperti operasi logika AND serta operasi logika OR.

1) Kegunaan

- Pada “Calculate” anda dapat memilih operator dengan *double click* atau dengan menekan tombol “Enter”.

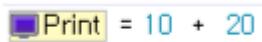


- Memilih 3 parameter yang disediakan (*result*, *operand1*, *operand2*) sesuai kebutuhan dari perintah



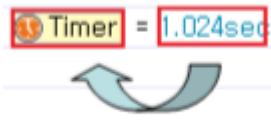
2) Contoh

Contoh ini menjelaskan bagaimana menjumlah 10 dan 20 serta menampilkan hasil pada layar.



e. Load

“Load” digunakan untuk “*to place into an appropriate device*”. Pada RoboPlus Task, “Load” digunakan untuk memberi nilai (*value*) pada *device*.



Pada perintah diatas “Load” digunakan untuk mengeksekusi fungsi pada *device* dan mengubah nilanya. Kegunaan dari perintah ini adalah untuk memilih 2 parameter (*destination, source*) sesuai dengan perintah yang diinginkan. Contohnya untuk mengeksekusi *device* dalam menentukan *timer* selama 1.024 detik maka dapat menggunakan perintah seperti yang ditunjukkan di bawah.

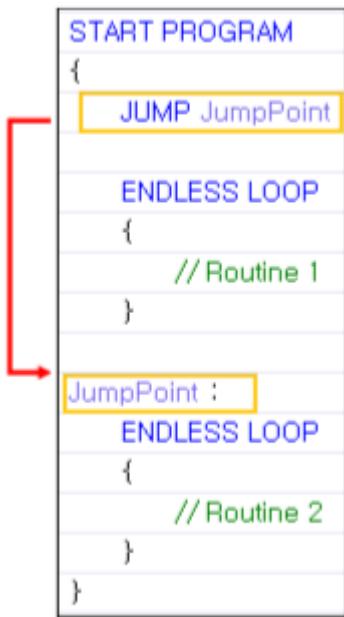
 **Timer** = 1.024sec

Sedangkan untuk menentukan nilai seperti memasukkan nilai 10 pada suatu variabel ditunjukkan di bawah.

 **Variable** = 10

f. Label/Jump

“Label” dan “Jump” digunakan untuk percabangan pada program. Percabangan digunakan untuk mengurutkan perintah pada program yang dijalankan. “Jump” digunakan untuk melakukan percabangan sedangkan “Label” digunakan untuk mengarahkan kemana percabangan tersebut. Hal tersebut sama seperti fungsi “Goto” pada pemrograman bahasa C.



1) Penggunaan

- Label harus dibuat berdasarkan peraturan berikut.
 - a. Tidak boleh terdapat nama label yang sama
 - b. Label harus berada dalam sebuah program atau sebuah fungsi.
 - c. Tidak mungkin melakukan “Jump” ke label pada fungsi lain.
 - d. Spasi dan karakter spesial (!, @, #, \$, dll.) tidak diperbolehkan pada label.
 - e. Label tidak dapat diawali dengan nomor.
- Ketika memasukkan nama Label, tekan “Esc” untuk *Cancel*.
- Disisilai, tekan “Enter” untuk *Save*.
- Ketika memilih label untuk “Jump”, tekan “Esc” untuk *Cancel*.
- Klik pada label yang tepat atau tekan enter ketika label ditampilkan untuk melakukan *Save*.

2) Contoh

Contoh program ini akan secara langsung melakukan loncat pada perintah “JumpPoint” saat program di jalankan dan menjalankan “Routine 2”.

```

START PROGRAM
{
    JUMP JumpPoint

    ENDLESS LOOP
    {
        // Routine 1
    }

    JumpPoint :
    ENDLESS LOOP
    {
        // Routine 2
    }
}

```

g. If/Else if/Else

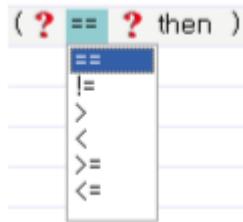
“If” akan menjalankan perintah ketika nilai dari syarat bernilai benar. “Else if” akan mengeksekusi program di dalamnya ketika syarat bernilai benar dan syarat sebelumnya (“If” atau “Else if”) bernilai salah. Sedangkan “Else” akan dieksekusi jika tidak ada kondisi sebelumnya yang bernilai benar. Perintah ini sama dengan “if”, “Else if” dan “Else” *statement* pada program bahasa C.

Conditional Clause (Kondisi Kebenaran)

Kondisi kebenaran adalah sebuah fitur untuk melakukan aksi yang berbeda berdasarkan bagaimana kondisi dari syarat yang diberikan yaitu benar atau salah. Kondisi kebenaran dibentuk dari 3 parameter yaitu: *parameter 1*, *relational operator*, dan *parameter 2*.



Terdapat 6 tipe dari *Relational Operator* yaitu:

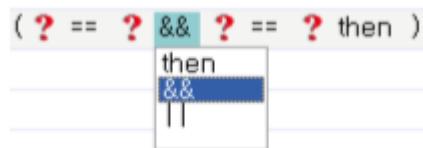


- a) == : Bernilai benar jika kedua parameter bernilai sama.
- b) != : Bernilai benar jika kedua parameter tidak bernilai sama.
- c) >= : Bernilai benar jika parameter 1 lebih besar atau sama dengan parameter 2.
- d) > : Bernilai benar jika parameter 1 lebih besar dari parameter 2.
- e) <= : Bernilai benar jika parameter 1 lebih kecil atau sama dengan parameter 2.
- f) < : Bernilai benar jika parameter 1 lebih kecil dari parameter 2.

Dalam penggunaannya kondisi kebenaran dapat dikombinasikan menjadi kondisi yang kompleks menggunakan *conditional operator*. Kondisi benar yang kompleks dibentuk dari 3 bagian yaitu: kondisi kebenaran 1, *conditional operator*, kondisi kebenaran 2.



Terdapat 3 tipe dari *conditional operator*



- a) **Then** : Tidak terhubung dengan kondisi kebenaran lain.
- b) **AND (&&)** : Bernilai benar jika kedua kondisi kebenaran bernilai benar.
- c) **OR(||)** : Bernilai benar jika salah satu atau kedua kondisi kebenaran bernilai benar.

Tidak ada batasan berapa banyak kondisi kebenaran dapat dikombinasikan menjadi kondisi kebenaran yang kompleks. Kondisi kebenaran akan dicek dan dijalankan sehingga pada akhir nilai yang diperoleh hanya diantara benar atau salah.

1) Penggunaan

- Sebuah perintah “If” selalu mendahului perintah “Else if” ataupun “Else”.
- Sebuah *block* selalu ditunjukkan dengan tanda kurung, dan diikuti dengan kondisi.

```
IF ( IR Center > 500 )
{
    Print with Line = 30
}

IF ( IR Center > 500 )
    Print with Line = 30
```

2) Contoh

Contoh ini menunjukkan bagaimana memprogram dengan syarat ketika variabel lebih besar atau sama dengan 90, ketika variabel lebih besar atau sama dengan 50 dan lebih kecil dari 90, dan ketika kondisi lain.

```
IF ( Variable >= 90 )
{
    // This will be executed, if conditions are satisfied.
}

ELSE IF ( Variable >= 50 )
{
    // This will be executed, if conditions are satisfied.
}

ELSE
{
    // Otherwise, this will be executed.
}
```

h. Endles Loop

Perintah ini merupakan perintah perulangan yang digunakan untuk mengulang baris perintah pada *block* secara terus menerus.

1) Penggunaan

- Selalu membutuhkan sebuah *block*.

```
ENDLESS LOOP
{
    Print = 30
}
```

- Gunakan perintah “Break Loop” untuk keluar dari perulangan.

2) Contoh

Program ini secara terus menerus mencetak “10” pada Program Output Monitor.

```
ENDLESS LOOP
{
    Print = 10
}

ENDLESS LOOP
Print = 10
```

i. Loop While

Perintah ini digunakan untuk melakukan perulangan pada baris perintah di dalam *block* ketika kondisi kebenaran dari syarat bernilai benar. Perintah ini sama dengan fungsu “while” pada pemrograman bahasa C.

1) Penggunaan

- Sebuah *block* program yang selalu ditandai dengan kurung kurawal dibutuhkan.

```
LOOP WHILE ( Timer > 0.000sec )
{
    Print = 10
}
```

```
LOOP WHILE ( Timer > 0.000sec )
    Print = 10
```

- Sama seperti halnya “Endless Loop” gunakan perintah “Break Loop” untuk keluar dari perulangan.

2) Contoh

Program ini akan secara terus menerus menampilkan nilai “Variable” pada Program Output Monitor hingga nilai “Variable” mencapai 30.

```
LOOP WHILE ( Variable <= 30 )
{
    Print = Variable
    Variable = Variable + 1
}
```

j. Loop For

“Loop For” merupakan perintah yang melakukan perulangan pada baris perintah dalam *block* dengan jumlah yang ditentukan. Memberikan nilai awal dan nilai akhir maka perulangan akan dilakukan dengan menambah variabel sebesar 1 setiap perulangan.

```
LOOP FOR ( LoopValue = 1 ~ 10 )
{
    Print = LoopValue
}
```

Nomor dari eksekusi dapat dihitung dengan rumus:

Nomor eksekusi = nilai yang ditentukan – nilai awal + 1

Perintah ini sama dengan “for” dari fungsi pemrograman bahasa C.

1) Penggunaan

- Tentukan 3 parameter dengan tepat (variabel, nilai awal, nilai akhir) dalam perintah.

```
LOOP FOR ( ? = ? ~ ? )
{
}
}
```

- Nilai awal yang ditentukan **harus** lebih kecil dari nilai akhir. **Jika nilai awal lebih besar dari nilai akhir maka perulangan tidak dieksekusi.**
- Sebuah *block* dibutuhkan dalam penggunaan “Loop For” ditandai dengan kurung kurawal.

```
LOOP FOR ( LoopValue = 1 ~ 10 )
{
    Print = LoopValue
}

LOOP FOR ( LoopValue = 1 ~ 10 )
    Print = LoopValue
```

- Gunakan perintah “Break Loop” untuk keluar dari perulangan.

2) Contoh

Pada program contoh ini program akan melakukan perulangan sebanyak suara yang terdeteksi.

```
LOOP FOR ( LoopValue = 1 ~ Sound count )
{
    // Repeation
}
```

k. Break Loop

Perintah ini digunakan untuk keluar dari perulangan yang sedang dijalankan. “Break Loop” sama dengan fungsi perintah “break” pada pemrograman bahasa C. Penggunaan “Break Loop” harus selalu dalam *block* perulangan.

```

ENDLESS LOOP
{
    Print = 10
    IF ( IR Center > 400 )
    {
        BREAK LOOP
    }
}
Print = 30

```



Contoh program secara terus menerus menampilkan angka “10” pada layar hingga nilai dari IR sensor lebih dari 400 dan saat keluar dari perulangan mencetak angka “30”.

```

ENDLESS LOOP
{
    Print = 10
    IF ( IR Center > 400 )
    {
        BREAK LOOP
    }
}
Print = 30

```

I. Wait While

“Wait While” merupakan perintah yang digunakan untuk menunda eksekusi ketika kondisi kebenarannya bernilai benar. Dalam penggunaan perintah ini tentukan kondisi dari nilai kebenaran tanpa menggunakan *block*.

```
WAIT WHILE ( Timer > 0.000sec )
```

m. Return Function

Saat perintah ini dieksekusi secara langsung akan mengakhiri operasi yang sedang berjalan walaupun belum semua baris perintah dijalankan dan akan menuju pada posisi fungsi yang memanggil. Penggunaan “Return Function” dapat digunakan hanya pada “general function” atau “callback function”.



```
1 START PROGRAM
2 {
3     ENDLESS LOOP
4     {
5         CALL UserFunction
6     }
7 }
8
9 FUNCTION UserFunction
10 {
11     PORT[3] = [ON,OFF]
12     Timer = 1.024sec
13     WAIT WHILE ( Timer > 0.000sec )
14
15     IF ( PORT[4] > 100 )
16         RETURN
17
18     PORT[3] = [OFF,ON]
19     Timer = 1.024sec
20     WAIT WHILE ( Timer > 0.000sec )
21 }
```

Pada contoh dibawah ini *UserFunction* dipanggil berulang kali. Karena perintah “Return” pada 3 baris akhir *UserFunction* tidak akan dieksekusi.

```

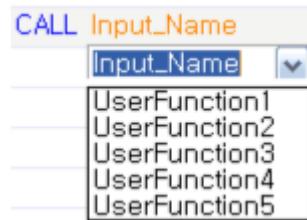
1  START PROGRAM
2  {
3      ENDLESS LOOP
4      {
5          CALL UserFunction
6      }
7  }
8
9  FUNCTION UserFunction
10 {
11     PORT[3] = [ON,OFF]
12     Timer = 1.024sec
13     WAIT WHILE ( Timer > 0.000sec )
14
15     IF ( PORT[4] > 100 )
16         RETURN
17
18     PORT[3] = [OFF,ON]
19     Timer = 1.024sec
20     WAIT WHILE ( Timer > 0.000sec )
21 }

```

1) Penggunaan

- Tidak boleh ada duplikasi nama.
- Sebuah fungsi harus berada di luar dari fungsi program lain atau program *body*.
- Spasi dan spesial karakter (!, @, #, \$, dll.) tidak diperbolehkan dalam penamaan fungsi.
- Nama fungsi tidak bisa diawali dengan angka.
- Saat memasukkan nama fungsi, tekan “Esk” untuk membatalkan dan tekan “Enter” untuk menyimpan.
- Ketika memilih fungsi yang akan dipanggil, tekan “Esc” untuk membatalkan.

- Klik pada fungsi atau tekan enter pada fungsi yang ingin dipanggil pada *highlight* untuk menyimpan.



- Sebuah fungsi tidak dapat memanggil fungsinya sendiri.

```
FUNCTION UserFunction1
{
    CALL UserFunction1
}
```

2) Contoh

Program akan memanggil secara terus menerus fungsi gerak *forward*, *backward*, *right*, dan *left*.

```
START PROGRAM
{
    ENDLESS LOOP
    {
        CALL Forward
        CALL Reverse
        CALL RightTurn
        CALL LeftTurn
    }
}
```

n. Callback Function

“Callback Function” merupakan fungsi yang berjalan secara independen dari program utama dan secara otomatis dijalankan pada interval yang sudah ditentukan. Terdapat “Callback Function” yang tidak dapat dimasukkan pada program yang membutuhkan banyak waktu. Penggunaan loops, variabel, pemanggilan fungsi dibatasi.

1) Penggunaan

- “Callback Function” tidak dapat berada di dalam fungsi lain atau di tubuh program.
- Hanya dapat memiliki satu “Callback Function”
- “Callback Function” tidak memiliki penamaan dan tidak dapat dipanggil.

2) Contoh

Contoh program ditampilkan bagaimana secara periode menerima *wireless* data dan menyimpan pada variable.

```
1  START PROGRAM
2  {
3      ENDLESS LOOP
4      {
5          IF ( ReceivedData > 100 )
6          {
7              // This will be executed, if conditions are satisfied.
8          }
9          ELSE
10         {
11             // Otherwise, this will be executed.
12         }
13     }
14 }
15
16 CALLBACK
17 {
18     IF ( Remocon Arrived == TRUE )
19     {
20         ReceivedData = Remocon RXD
21         IF ( ReceivedData == 0 )
22         {
23             // This will be executed, if conditions are satisfied.
24         }
25     }
26 }
```

Referensi

ROBOTIS. (2010). ROBOTIS e-manual v1.31.30. Diakses dari <http://support.robotis.com/en/> pada tanggal 2 September 2019, pukul 18.35 WIB.

LABSHEET

PRAKTIK QUADRUPED ROBOT



Disusun Oleh:
Hilmi Musthafa Albasyir
NIM. 15518241029

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



**JURUSAN PENDIDIKAN TEKNIK ELEKTRO
FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA**

LABSHEET PRAKTIK ROBOTIKA

No. 01

RoboPlus Manager dan Motion

A. KOMPETENSI

1. Dapat memahami *software* RoboPlus Manager

B. SUB KOMPETENSI

Setelah melakukan praktikum mahasiswa diharapkan dapat:

1. Memahami fitur-fitur yang terdapat pada *software* RoboPlus Manager.
2. Merubah nomor ID perangkat aktuator servo menggunakan *software* RoboPlus Manager.
3. Mengakses tabel kontrol dari perangkat kontroler CM-510 dan aktuator servo menggunakan *software* RoboPlus Manager.

C. DASAR TEORI

1. CM-510

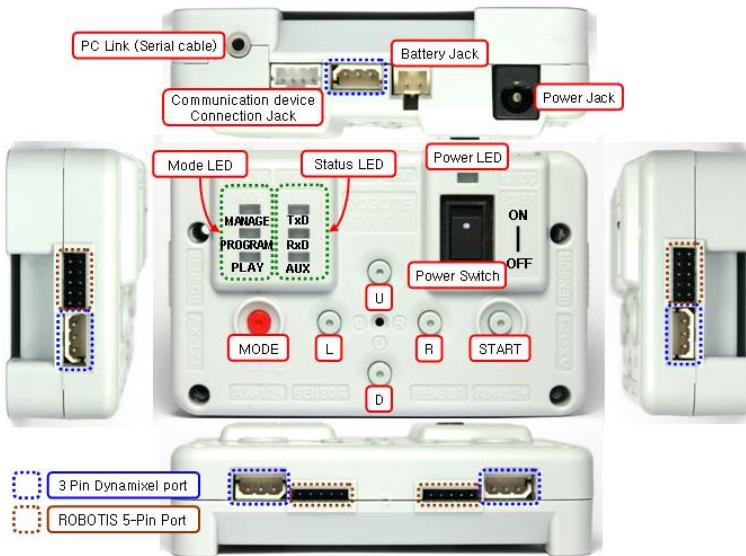
CM-510 merupakan kontroler keluaran perusahaan ROBOTIS. Kontroler CM-510 memiliki port yang identik dengan kontroler. Kontroler CM-510 berbasis mikrokontroler ATMega2461 sedangkan kontroler CM-530 menggunakan mikrokontroler berbasis ARM CPU (STM32F103RE). Kontroler CM-510 dibekali dengan *flash* memori sebesar 256KB dan 8KB untuk SRAM. *Supply* daya pada kontroler ini didesain mampu menerima masukan tegangan dengan rentang antara 6.5Volt hingga 15Volt. Rekomendasi tegangan masukan untuk kontroler ini adalah 11.1V atau dapat digunakan batrei Lippo 3 Cell. CM-510 didesain dapat beroperasi pada rentang suhu antara -5°C sampai dengan 70°C. Terdapat masukan (*input*) dan keluaran (*Output*) yang disediakan kontroler CM-510 pada Gambar 1. Masukan (*input*) dan keluaran (*Output*) pada CM-510 memiliki kegunaan masing-masing yang dijelaskan pada Tabel 1.



JURUSAN PENDIDIKAN TEKNIK ELEKTRO
FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA
LABSHEET PRAKTIK ROBOTIKA

No. 01

RoboPlus Manager dan Motion



Gambar 1. Input dan Output Port CM-510

(Sumber: ROBOTIS)

Tabel 2. Input dan Output beserta kegunaan pada kontroler CM-510.

No	Input dan Output	Kegunaan
1.	PC Link (<i>Serial Cable</i>)	Menghubungkan CM-510 dengan PC menggunakan serial port. Berguna untuk mendownload kode program atau berkomunikasi dengan PC.
2.	Communication Device Connection Jack	Komunikasi Wireless dengan ZIG-110, IR Receiver atau <i>boards</i> lain.
3.	Battery Jack	Masukan tegangan dari batrai.
4.	Power Jack	Masukan tegangan menggunakan SMPS Power Supply.
5.	Power LED	Status <i>Power</i> dari kontroler,
6.	Powe Switch	Menghidupkan dan mematikan kontroler.
7.	MODE Button	Mengubah mode operasi.
8.	START Button	Menjalankan mode yang ditentukan.
9.	U / L / D / R Button	Input yang digunakan ketika program berjalan. Tombol dapat digunakan untuk memberikan perintah pada kontroler.
10.	AX/MX Series Bus Port	Memasang Aktuator berupa AX/MX Dynamixel Servo series.
11.	Peripheral Device Connection Port	Memasang Sensor jarak, Sensor sentuh, IR Sensor dan <i>Peripheral devices</i> .
12.	Mode Display LED	Menampilkan mode operasi yang terpilih dari CM-510.



JURUSAN PENDIDIKAN TEKNIK ELEKTRO
FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA
LABSHEET PRAKTIK ROBOTIKA

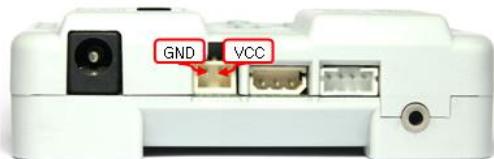
No. 01

RoboPlus Manager dan Motion

Gambar 1 dan Tabel 1 ditunjukkan posisi dari setiap input dan output dari beberapa input dan output tersebut terdapat *Pinout* yang perlu diketahui agar dalam penggunaan kontroler tidak terjadi kesalahan. *Pinout* yang perlu diketahui pada kontroler CM-510 yaitu:

- a. Power

Komposisi pin pada Power ditunjukkan pada Gambar 2.

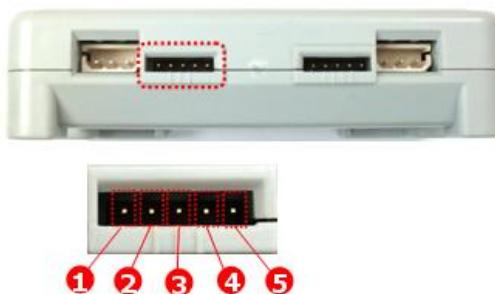


Gambar 2. Power Pinout Composition

(Sumber: ROBOTIS)

- b. External 5-Pin Port

Pin ini digunakan untuk memasang *Peripheral devices*. Komposisi pin pada External 5-Pin Port ditunjukkan pada Gambar 3.



Gambar 3. External 5-Pin Port Pinout Composition

(Sumber: ROBOTIS)

Pada Gambar 3 ditunjukkan angka yang memiliki kegunaan masing-masing yaitu: (1) OUT : 5 Volt dengan arus maksimal 0.9 A, (2) VCC : 5 Volt, (3) ADC : Sinyal analog dari sensor yang akan dibaca, (4) GND, dan (5) NC : tidak terpakai.

- c. Communication Device Connection Port

Pengiriman data pada kontroler CM-510 dapat melalui soket ini yang menggunakan pengiriman data berupa serial. Komposisi pin pada communication device connection port ditunjukkan pada Gambar 4.



**JURUSAN PENDIDIKAN TEKNIK ELEKTRO
FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA**

LABSHEET PRAKTIK ROBOTIKA

No. 01

RoboPlus Manager dan Motion



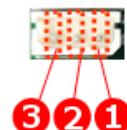
Gambar 4. Communication Device Connection Port Pinout Composition

(Sumber: ROBOTIS)

Gambar 4 yang ditunjuk dengan nomor berwarna merah yaitu: (1) GND: *Ground Level* (0 Volt), (2) VCC: *Supply Voltage* (2.7 ~ 3.6 Volt), (3) RXD: *Receive Signal Terminal*, dan (4) TXD: *Transmit Signal Terminal*.

d. 3-Pin Connector Port

Aktuator berupa MX ataupun AX Servo Dynamixel dihubungkan pada pin ini. Komposisi pin pada 3-Pin Connector Port ditunjukkan pada Gambar 5.



Gambar 5. 3-Pin Connector Port Pinout Composition

(Sumber: ROBOTIS)

Pada setiap nomor pin memiliki fungsi yaitu: (1) GND: *Ground Level* (0 Volt), (2) VDD: *Supply Voltage*, dan (3) DATA: Pin Transmisi Data.

2. Motor Servo Dyanamixel MX-28 dan MX-64

Dynamixel MX merupakan salah satu jenis servo produksi dari perusahaan ROBOTIS. Servo ini termasuk jenis *smart actuator*, karena servo terintegrasi penuh dengan motor DC penggerak, kontroler, *driver*, dan jaringan dalam satu modul. *Smart actuator* ini dapat memberikan *feedback* (umpan balik) berupa nilai posisi, suhu, beban, tegangan input, torsi, dan lain-lain. Komunikasi antara servo dynamixel dengan kontroler utama dalam menyampaikan data menggunakan antarmuka UART TTL *half duplex*. Sama dengan servo dynamixel lainnya, instalasi fisik antar servo menggunakan sambungan rantai *daisy*. Servo Dynamixel MX-28 ditunjukkan pada Gambar 6 dan Servo Dynamixel MX-64 ditunjukkan pada Gambar 7



**JURUSAN PENDIDIKAN TEKNIK ELEKTRO
FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA**

LABSHEET PRAKTIK ROBOTIKA

No. 01

RoboPlus Manager dan Motion



Gambar 6. Servo Dynamixel MX-28

(Sumber: ROBOTIS)



Gambar 7. Servo Dynamixel MX-64

(Sumber: ROBOTIS)

3. *Software RoboPlus Manager*

Roboplus Manager merupakan software keluaran ROBOTIS. *Software* ini memiliki fungsi untuk mengelola kontroler & aktuator servo dynamixel yang terdapat pada robot, memperbarui *firmware* (F/W) produk ROBOTIS ke versi yang terbaru, serta mengecek *feedback* berupa suhu, tegangan input, dan lain lain dari kontroler maupun servo dynamixel.

Software Roboplus Manager kompatibel dengan sistem operasi Windows dan OS X. Spesifikasi sistem operasi minimal yang diperlukan untuk menjalankan RoboPlus adalah Windows XP SP2 32-bit ataupun 64-bit. Sedangkan pada OS X sistem operasi minimal adalah Mac OS 10.5.

Tampilan awal RoboPlus memberikan berbagai pilihan antara lain: 1) RoboPlus Task; 2) RoboPlus Manager; 3) Roboplus Motion; 4) RoboPlus Terminal; 5) Dynamixel Wizard; 6) User's Guide; dan 7) e-Manual. Perhatikan Gambar C.2.



**JURUSAN PENDIDIKAN TEKNIK ELEKTRO
FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA**

LABSHEET PRAKTIK ROBOTIKA

No. 01

RoboPlus Manager dan Motion

Produk kontroller yang dapat diakses menggunakan *software* ini antara lain: 1) CM-100; 2) CM-100A; 3) CM-150 4); CM-200; 5) CM-5; 5) CM-510; 7) CM-530; 8) CM-700; dan 9) OpenCM9.04

4. *Software RoboPlus Motion*

RoboPlus Motion merupakan *software* keluaran ROBOTIS. *Software* ini memiliki fungsi untuk membuat *motion* pada kontroler. *Motion* merupakan rekaman beberapa posisi dan kecepatan dari kumpulan aktuator motor servo yang diterjemahkan sebagai pergerakan robot.

RoboPlus Motion didesain untuk merekam posisi sudut pada robot yang dimasukkan pada setiap “step” yang akan dijalankan oleh kontroler. Menu pembuatan *motion* pada RoboPlus Motion dibagi menjadi tiga bagian yaitu: (1) *Pose Editing*; (2) *Step Editing*; dan (3) *Page Editing*.

g. *Pose Editing*

Pose adalah posisi robot pada suatu titik waktu. *Pose* merupakan kumpulan dari posisi sudut motor servo robot yang diperlukan untuk menghasilkan posisi yang diinginkan pada suatu waktu. Pada menu “Basic Pose Editing” terdapat dua tabel editor yaitu “Pose of Step” dan “Pose of Robot” yang ditunjukkan pada Gambar 8.

The screenshot shows the 'Basic Pose Editor' interface with two tables. The left table, titled '<Pose of Step>', contains 13 rows of servo ID and value pairs. The right table, titled '<Pose of Robot>', contains 13 rows of servo ID and value pairs. Between the tables is a central control panel with a circular dial and two green arrows pointing left and right, and a small digital display below it.

ID[1]	1868	<input checked="" type="checkbox"/>
ID[2]	1709	<input checked="" type="checkbox"/>
ID[3]	1942	<input checked="" type="checkbox"/>
ID[4]	2221	<input checked="" type="checkbox"/>
ID[5]	2381	<input checked="" type="checkbox"/>
ID[6]	2266	<input checked="" type="checkbox"/>
ID[7]	2198	<input checked="" type="checkbox"/>
ID[8]	2320	<input checked="" type="checkbox"/>
ID[9]	2183	<input checked="" type="checkbox"/>
ID[10]	1898	<input checked="" type="checkbox"/>
ID[11]	1746	<input checked="" type="checkbox"/>
ID[12]	1948	<input checked="" type="checkbox"/>
ID[13]	2083	<input checked="" type="checkbox"/>

ID[1]	OFF
ID[2]	OFF
ID[3]	OFF
ID[4]	OFF
ID[5]	OFF
ID[6]	OFF
ID[7]	OFF
ID[8]	OFF
ID[9]	OFF
ID[10]	OFF
ID[11]	OFF
ID[12]	OFF
ID[13]	OFF

Gambar 8. Basic Pose Editor

Pose of Step merupakan data posisi sudut dari aktuator servo robot yang terekam. Sedangkan *Pose of Robot* adalah nilai posisi dari servo motor pada robot. Ketika posisi robot diubah maka nilai pada *Pose of Robot* akan berubah sesuai dengan posisi robot tersebut.



**JURUSAN PENDIDIKAN TEKNIK ELEKTRO
FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA**

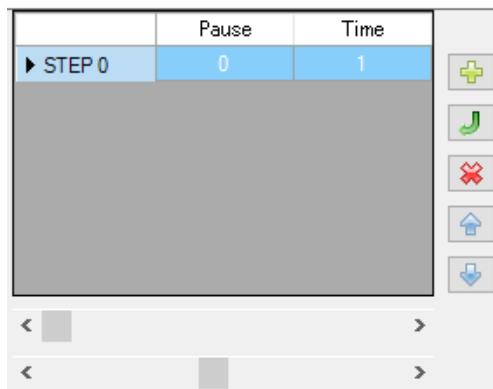
LABSHEET PRAKTIK ROBOTIKA

No. 01

RoboPlus Manager dan Motion

h. *Step Editing*

Sebuah “Motion Step” merupakan kumpulan dari *key-frame* yang dibutuhkan untuk membuat sebuah motion pada robot. *Key-frame* merupakan langkah-langkah dari *motion* robot. *Key-frame* terdiri atas: 1) posisi dan 2) kecepatan motor servo, yang dapat diterjemahkan sebagai Dimana (Posisi) selama Kapan (Waktu). Step Editing memiliki beberapa fungsi seperti *Add Step*, *Insert Step*, *Delete Step*, dan *Move Step*. Sedangkan pada setiap *Step* terdapat “Pause” dan “Time”. “Pause” adalah waktu antara akhir dari *Step* saat ini dengan awal dari *Step* selanjutnya.



Gambar 9. *Step Editing*

i. *Page Editing*

Page Editing merupakan tabel yang berisi *motion* yang tersimpan pada RoboPlus Motion. Data dari *motion* terdiri dari 255 *pages*. Namun pada beberapa kontroler hanya terbatas pada 127 *pages*.

D. ALAT DAN BAHAN

1. *Quadruped* Robot
2. Kabel downloader USB RS232 to Audio Stereo 2.5
3. *Power supply* 12 V DC
4. Komputer / PC yang sudah ter-*install software* RoboPlus Manager & Motion



**JURUSAN PENDIDIKAN TEKNIK ELEKTRO
FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA**

LABSHEET PRAKTIK ROBOTIKA

No. 01

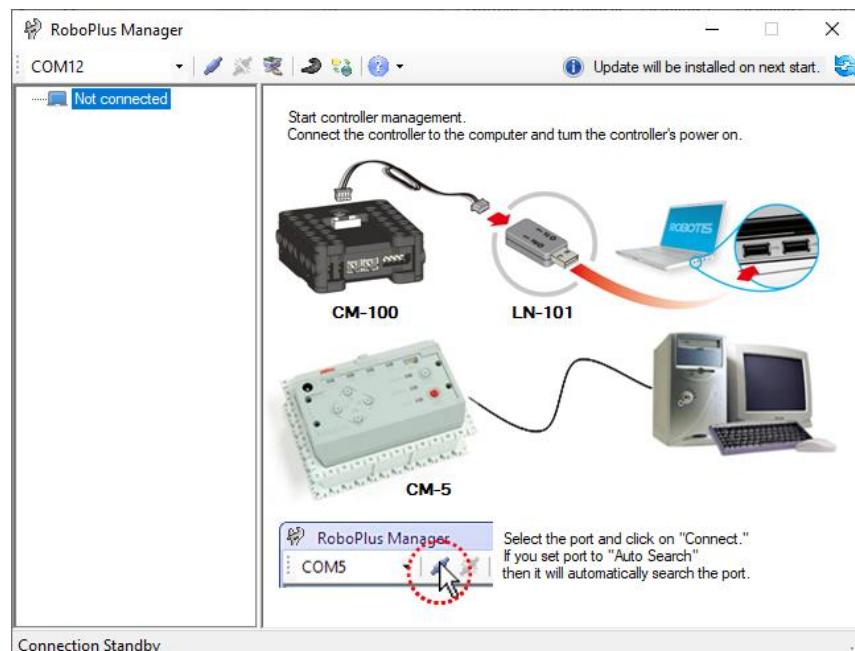
RoboPlus Manager dan Motion

E. KESELAMATAN KERJA

1. Bekerjalah dengan keadaan tanpa tegangan saat memasang komponen pada *quadruped* robot.
2. Periksalah tegangan *output power supply*. Tegangan output yang diizinkan pada *quadruped* robot adalah 12VDC.
3. Pastikan semua komponen telah terpasang dengan benar sebelum menghubungkan *quadruped* robot dengan sumber tegangan.
4. Gunakan peralatan sesuai dengan fungsinya.
5. Duduklah di kursi komputer dengan baik dan benar, dengan minimal jarak mata dan layar komputer +30 cm. Disarankan menggunakan kacamata anti radiasi.
6. Jauhkan peralatan yang tidak digunakan dari meja kerja.
7. Bekerjalah sesuai dengan langkah kerja yang telah ditentukan.

F. LANGKAH KERJA

1. Buka *software Roboplus Manager* untuk mengakses kontroler CM-510 dan servo aktuator yang terhubung pada kontroler. Tampilan antarmuka RoboPlus Manager seperti yang ditunjukkan pada Gambar 10.



Gambar 10. Roboplus Manager

2. Hubungkan kabel downloader ke PC dan kontroler. Klik “Controller Port” pilih nomor *port* yang sesuai dengan kontroler pada *Windows Device Manager*.



JURUSAN PENDIDIKAN TEKNIK ELEKTRO
FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA
LABSHEET PRAKTIK ROBOTIKA

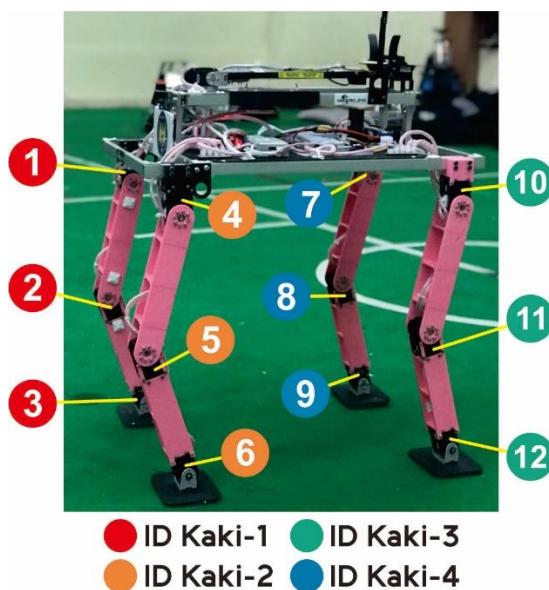
No. 01

RoboPlus Manager dan Motion

3. Hidupkan kontroler CM-510 pada *Quadruped* robot.
4. Klik tombol “Connect” pada RoboPlus Manager untuk menghubungkan kontroler. Pastikan nomor *port* kontroler benar.
5. Apabila semua prosedur telah dilakukan dengan benar, maka pada RoboPlus Manager akan tampil seri kontroler serta daftar aktuator motor servo yang diurutkan sesuai ID.
6. Lakukan pengamatan pada tabel kontrol untuk CM-510 dan catat hasil pengamatan pada tabel di bawah ini.

No	Deskripsi	Nilai
1	<i>Model Number</i>	
2	<i>Version of Firmware</i>	
3	ID	

7. Apakah yang akan terjadi apabila nilai dari “Aux LED” pada tabel kontrol CM-510 diubah menjadi 1?
8. Selanjutnya lakukan pengamatan pada servo MX-28 dan MX 64 pastikan nomor ID setiap motor servo benar. Perhatikan gambar 11.



Gambar 11. ID Servo pada *Quadruped* Robot

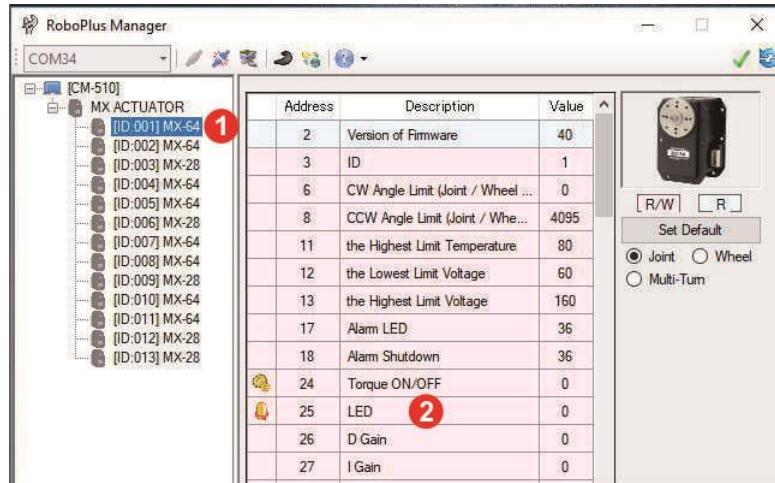
9. Untuk memastikan kesesuaian nomor ID adalah dengan mengakses ID servo yang terdeteksi dan menyalakan LED-nya. Perhatikan Gambar 12 untuk menyalakan LED pada servo. 1) Klik pada ID servo yang akan diuji; 2) Klik pada *Address* nomor 25 yang memiliki deskripsi LED; 3) Ubah *value*-nya menjadi 1.



**JURUSAN PENDIDIKAN TEKNIK ELEKTRO
FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA
LABSHEET PRAKTIK ROBOTIKA**

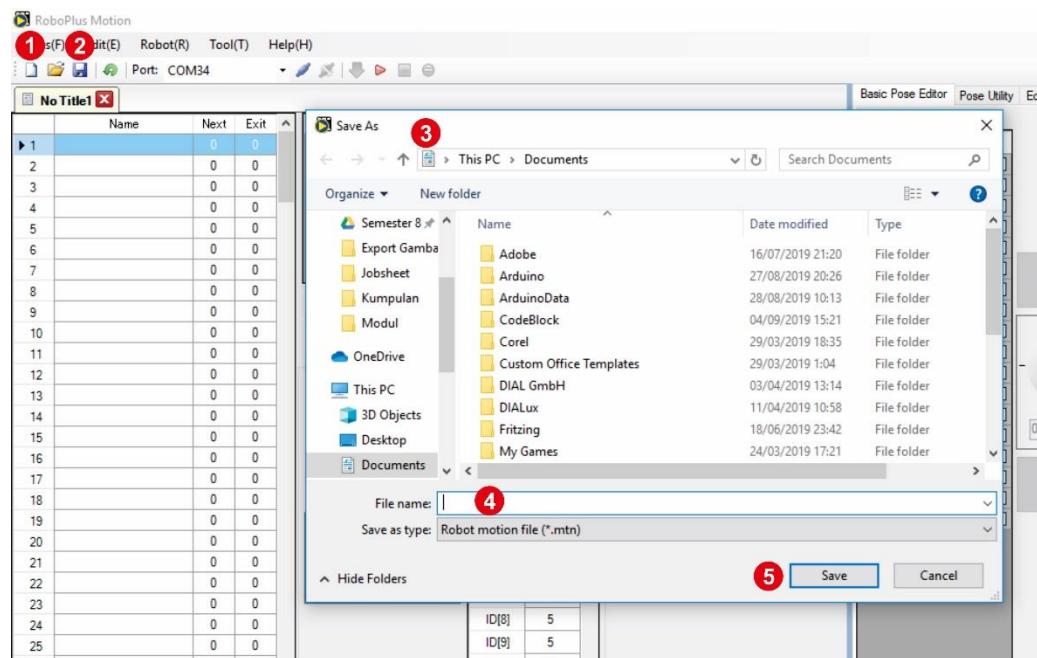
No. 01

RoboPlus Manager dan Motion



Gambar 12. Setting LED

10. Jika seluruh ID servo sudah benar, bukalah *software* RoboPlus Motion untuk membuat *motion quadruped* robot.
11. Hubungkan kontroler CM-510 pada *quadruped* robot ke PC. Pilihlah nomor *port* yang sesuai lalu klik *icon* “Connect”. Tunggu hingga *quadruped* robot tersambung.
12. Langkah pertama yang dilakukan adalah membuat *project* baru, klik “File” lalu “New”. Simpanlah *project* dengan nama sesuai kebutuhan anda. Perhatikan Gambar 13.



Gambar 13. Membuat *Project motion* baru



**JURUSAN PENDIDIKAN TEKNIK ELEKTRO
FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA**

LABSHEET PRAKTIK ROBOTIKA

No. 01

RoboPlus Manager dan Motion

13. *Setting* resolusi dari aktuator servo yang digunakan. Pada menu “Edit All Page” *check list* ID servo dari 1 sampai dengan 12 dan ubah resolusi pada setiap ID servo yang ter-*check list* menjadi 4096 (360°) seperti yang ditunjukkan pada Gambar 14.

	Enable	Resolution
ID[0]	<input type="checkbox"/>	1024 (300°)
ID[1]	<input checked="" type="checkbox"/>	1024 (300°) 2
ID[2]	<input checked="" type="checkbox"/>	1024 (300°) 4096 (250°)
ID[3]	<input checked="" type="checkbox"/>	4096 (360°) 3
ID[4]	<input checked="" type="checkbox"/>	1024 (300°)
ID[5]	<input checked="" type="checkbox"/>	1024 (300°)
ID[6]	<input checked="" type="checkbox"/>	1024 (300°)

Gambar 14. *Setting* Resolusi Servo

14. Selanjutnya buatlah *motion* baru untuk membuat gerakan berjalan pada *quadruped* robot. Perhatikan tabel berikut ini.

No	Motion	Step	Image
1.	Stand by	Step 0	
2.	Motion 1	Step 0	



**JURUSAN PENDIDIKAN TEKNIK ELEKTRO
FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA
LABSHEET PRAKTIK ROBOTIKA**

No. 01

RoboPlus Manager dan Motion

		Step 1	
		Step 2	
3. Motion 2		Step 0	
		Step 1	



**JURUSAN PENDIDIKAN TEKNIK ELEKTRO
FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA
LABSHEET PRAKTIK ROBOTIKA**

No. 01

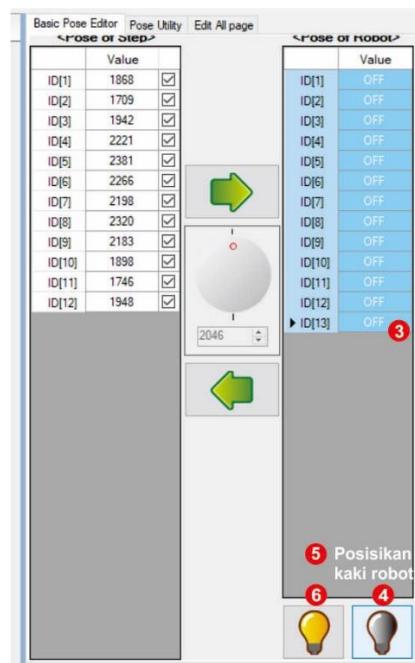
RoboPlus Manager dan Motion

	Step 2	
--	--------	--

Keterangan:

Motion Stanby digunakan untuk menentukan posisi awal dari robot. *Motion 1* untuk langkah pertama dari *quadruped* robot. Sedangkan *motion 2* digunakan untuk langkah kedua pada robot.

15. Buatlah *motion* baru dengan mengikuti instruksi berikut. 1) Beri nama “Stand by” pada bagian *name motion*; 2) klik *insert Step*; 3) Pilih “STEP 0” pada motion “Stand by”; 4) Matikan torsi dari semua servo dengan cara *select* semua ID pada menu “Pose of Robot” lalu tekan gambar lampu padam. 5) Selanjutnya posisikan kaki robot dalam kondisi default; 6) Hidupkan torsi dari semua servo pada robot lalu tekan tombol panah berwarna hijau yang menyarah ke “Pose of Step”. Pastikan seluruh ID pada menu “Pose of Step” terceklis.



Gambar 15. Membuat *motion*



JURUSAN PENDIDIKAN TEKNIK ELEKTRO
FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA
LABSHEET PRAKTIK ROBOTIKA

No. 01

RoboPlus Manager dan Motion

16. Setelah selesai membuat *motion* Stanby, selanjutnya buatlah *motion* sesuai dengan gambar pada tabel diatas. Tulislah data posisi sudut dari setiap servo pada tabel yang tersedia.

ID	Stand by	Motion 1			Motion 2			Motion 3		
		S0	S1	S2	S0	S1	S2	S0	S1	S2
1										
2										
3										
4										
5										
6										
7										
8										
9										
10										
11										
12										

17. Cobalah motion yang telah dibuat dengan cara memilih motion tersebut lalu klik icon “play motion” lalu isikan respon pada tabel berikut.

No.	Motion	Respon
1.	Stand by	
2.	Motion 1	
3.	Motion 2	

G. TUGAS

1. Sesuai dengan konsep pada *motion* 1 dan 2 buatlah *motion* agar robot bergerak belok ke **kanan**. (Dalam hal ini robot akan bergerak ke arah kanan jika besar langkah robot sisi kanan lebih kecil dari langkah robot sisi kiri.)
2. Buatlah *motion* untuk robot bergerak ke **keduanya**.



**JURUSAN PENDIDIKAN TEKNIK ELEKTRO
FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA
LABSHEET PRAKTIK ROBOTIKA**

No. 02

RoboPlus Task

A. KOMPETENSI

1. Dapat menggunakan *software* RoboPlus Task

B. SUB KOMPETENSI

Setelah melakukan praktikum mahasiswa diharapkan dapat:

1. Membuat program berinstruksi pada *software* RoboPlus Task
2. Membuat program berparameter pada *software* RoboPlus Task

C. DASAR TEORI

RoboPlus Task merupakan produk keluaran ROBOTIS. *Software* ini memiliki fungsi untuk membuat kode program algoritma dan logika yang dapat disebut *task*, serta memanggil *motion* yang telah dibuat pada aplikasi RoboPlus Motion. Dalam ilmu komputer *task* diartikan sebagai aktivitas yang berhubungan untuk menyelesaikan suatu pekerjaan. Sedangkan dalam RoboPlus, kode *task* dimaknai sebagai kode program yang digunakan robot untuk menyelesaikan pergerakan robot.

Software RoboPlus Task kompatibel dengan sistem operasi Windows dan OS X. Spesifikasi sistem operasi minimal yang diperlukan untuk menjalankan RoboPlus adalah Windows XP SP2 32-bit ataupun 64-bit. Sedangkan pada OS X sistem operasi minimal adalah Mac OS 10.5. Robot yang diprogram menggunakan RoboPlus Task akan beroperasi berdasarkan kode *task* yang telah dibuat oleh pengguna. File ekstensi dari file *task* adalah “.tsk” (tanpa tanda petik).

RoboPlus Task memiliki *tools* untuk menunjang programming. Terdapat *tools* yaitu: *New*, *Open*, *Save*, *Print*, *Cut*, *Copy*, *Paste*, *Find Name*, *Check Rule*, *Download Program*, *Change Controller*, dan *Port Controller*. Bagian-bagian tersebut masing-masing memiliki fungsi yang mendukung dalam programming. Fungsi dari bagian-bagian tersebut dijelaskan pada Tabel E.1



**JURUSAN PENDIDIKAN TEKNIK ELEKTRO
FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA
LABSHEET PRAKTIK ROBOTIKA**

No. 02

RoboPlus Task

Tabel 1. Fungsi-fungsi *Tools* pada RoboPlus Task

Tools	Fungsi
<i>New</i>	Membuat Program baru
<i>Open</i>	Membuka Program yang tersimpan
<i>Save</i>	Menyimpan Program
<i>Print</i>	Mencetak program
<i>Cut</i>	Memotong/memindahkan kode program
<i>Copy</i>	Menyalin kode program
<i>Paste</i>	Menampilkan hasil dari <i>copy/cut</i>
<i>Find Name</i>	Mencari kata
<i>Check Rule</i>	Mengecek program apakah terdapat <i>error</i>
<i>Download Program</i>	Memasukkan program ke kontroler
<i>Change Controller</i>	Mengubah seri kontroler
<i>Port Controller</i>	Memilih <i>port</i> dari kontroler

D. ALAT DAN BAHAN

1. *Quadruped* Robot
2. Kabel *downloader* USB RS232 to Audio Stereo 2.5
3. *Power supply* 12 V DC
4. Komputer / PC yang sudah ter-*install software* RoboPlus Manager & Motion

E. KESELAMATAN KERJA

1. Bekerjalah dengan keadaan tanpa tegangan saat memasang komponen pada *quadruped* robot.
2. Periksalah tegangan *output power supply*. Tegangan output yang diizinkan pada *quadruped* robot adalah 12VDC.
3. Pastikan semua komponen telah terpasang dengan benar sebelum menghubungkan *quadruped* robot dengan sumber tegangan.
4. Gunakan peralatan sesuai dengan fungsinya.
5. Duduklah di kursi komputer dengan baik dan benar, dengan minimal jarak mata dan layar komputer +30 cm. Disarankan menggunakan kacamata anti radiasi.
6. Jauhkan peralatan yang tidak digunakan dari meja kerja.
7. Bekerjalah sesuai dengan langkah kerja yang telah ditentukan.



**JURUSAN PENDIDIKAN TEKNIK ELEKTRO
FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA
LABSHEET PRAKTIK ROBOTIKA**

No. 02

RoboPlus Task

F. LANGKAH KERJA

1. Bukalah *software* RoboPlus Task untuk program *task quadruped* robot. Tampilan antarmuka RoboPlus Task seperti di bawah ini.
2. Langkah pertama dalam membuat program *task* adalah dengan membuat *file* program *task*. *Double click* pada baris kosong atau tekan “Enter”, pada <Select Controller> window, pilih *radio button* “1.0” dan klik kontroler CM-510. Klik tombok “OK”.
3. Klik *icon* Save untuk menyimpan *task* baru.
4. Sebelum memulai program pastikan *motion* “job2.mtn” atau *motion* yang telah dibuat pada labsheet sebelumnya telah di-*download* ke kontroler *quadruped* robot. Jika belum maka *download* terlebih dahulu *file motion* tersebut pada kontroler CM-510.
5. Buatlah program untuk robot berjalan kedepan terus menerus ketika di tekan tombol “U” pada kontroler. Untuk posisi awal dapat menggunakan *file motion index* 1 sedangkan untuk berjalan ke depan dapat menggunakan *file motion index* 3 dan 4. Perhatikan gambar berikut.

```
1 START PROGRAM
2 {
3     Motion Index Number = 1
4     WAIT WHILE ( Motion Status == TRUE )
5     ENDLESS LOOP
6     {
7         IF ( Button == U )
8         {
9             ENDLESS LOOP
10            {
11                Motion Index Number = 3
12                WAIT WHILE ( Motion Status == TRUE )
13                Motion Index Number = 4
14                WAIT WHILE ( Motion Status == TRUE )
15            }
16        }
17    }
18 }
19 }
```



**JURUSAN PENDIDIKAN TEKNIK ELEKTRO
FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA
LABSHEET PRAKTIK ROBOTIKA**

No. 02

RoboPlus Task

Keterangan:

- a. Untuk memasukkan perintah (*command*), baca panduan Awal *Programming* pada modul.
- b. Untuk mengatur input dan syarat, baca panduan Pemrograman RoboPlus Task.
- c. Untuk mengetahui berbagai perintah pada RoboPlus Task, baca panduan Pemrograman RoboPlus Task
6. Selanjutnya tekan tombol ON pada kontroler CM-510 *quadruped* robot.
7. Hubungkan kontroler CM-510 *quadruped* robot pada PC. Pada RoboPlus Task pilih nomor *port* kontroler tersebut.
8. *Download*-kan program *task* yang telah dibuat dengan menekan tombol *download* pada menu. Jika sudah selesai, lepas sambungan kabel *downloader* USB RS232 to Audio Stereo 2.5 dari *quadruped* robot dan PC.
9. Simulasikan hasil program Anda dengan menjalankan *quadruped* robot. Jalankan program dengan memilih mode “PLAY” lalu tekan tombol start. Jika *task* dan *motion* benar maka *quadruped* robot akan bergerak melakukan posisi awal. Amati yang terjadi! Apakah sudah sesuai dengan rancangan *task* dan *motion* Anda?
10. Selanjutnya jalankan robot dengan menekan tombol “U” pada kontroler. Amati gerakan robot! Apakah *motion* berjalan sesuai *task* dan robot dapat berjalan ke depan?
11. Untuk mematikan robot saat bergerak dapat menekan tombol “STOP” pada kontroler CM-510.

G. TUGAS

1. Buatlah program *quadruped* robot berjalan dengan ketentuan:
 - 1) Robot bergerak ketika ditekan tombol “R” pada kontroler.
 - 2) Gerak robot berjalan berbelok ke arah kanan, untuk memudahkan gunakan *file motion index* 6 dan 7.
 - 3) Ketika robot sudah pergerak 6 langkah ke kanan maka akan berhenti dan mempertahankan posisi langkah terakhir.
2. Buatlah program *quadruped* robot berjalan dengan ketentuan:
 - 1) Robot bergerak ketika ditekan tombol “L” pada kontroler.



**JURUSAN PENDIDIKAN TEKNIK ELEKTRO
FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA
LABSHEET PRAKTIK ROBOTIKA**

No. 02

RoboPlus Task

- 2) Gerak robot berjalan berbelok ke arah kiri, untuk memudahkan gunakan *file motion index* 9 dan 10.
- 3) Ketika robot sudah bergerak 4 langkah ke kiri maka robot akan berjalan lurus (*file motion index* 3 dan 4) terus menerus.

H. BAHAN DISKUSI

1. Jelaskan perbedaan pemrograman pada RoboPlus Task dengan pemrograman pada aplikasi lain yang anda ketahui.
2. Jelaskan bagaimana hubungan antara *file task* dan *motion*.
3. Buatlah kesimpulan dari hasil praktikum Anda!

	JURUSAN PENDIDIKAN TEKNIK ELEKTRO FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA LABSHEET PRAKTIK ROBOTIKA	
No. 03	Mengendalikan <i>Quadruped</i> Robot dengan <i>Stick PS2</i>	

A. KOMPETENSI

1. Membuat program pengendalian *quadruped* robot dengan *stick PS2*.

B. SUB KOMPETENSI

Setelah melakukan praktikum mahasiswa diharapkan dapat:

1. Membuat Program Roboplus Task untuk menerima data.
2. Membuat Program Arduino dan *stick PS2*.
3. Membuat program mengirim data dari Arduino ke kontroler CM-510.

C. DASAR TEORI

Arduino nano merupakan kontroler berbasis ATmega yang dapat digunakan untuk menerima input dari *stick wireless PlayStation 2*. *Stick wireless* terdapat dua bagian yaitu bagian PS2 *Wireless controller* dan *Wireless Receiver* yang dapat dilihat pada Gambar 1.



Gambar 1. PS2 *Wireless Controller* dan *Wireless Receiver*
(Sumber: amazon.com)

Stick dapat digunakan sebagai input pada kontroler CM-510 dengan cara melewatkannya terlebih dahulu *stick* pada kontroler arduino. Selanjutnya kontroler arduino mengirimkan data serial ke kontroler CM-510. Pengiriman serial dari arduino ke CM-510 melalui fitur RX dan TX pada kedua kontroler. Susunan pin yang terdapat pada arduino dan CM-510 secara urut dapat dilihat pada Gambar 2 dan 3.

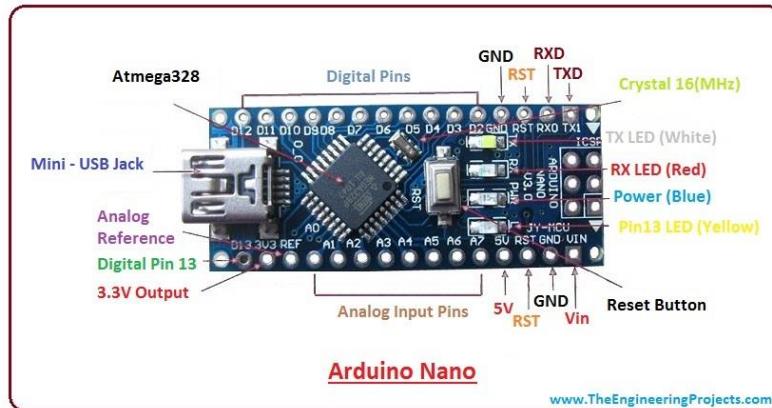


**JURUSAN PENDIDIKAN TEKNIK ELEKTRO
FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA**

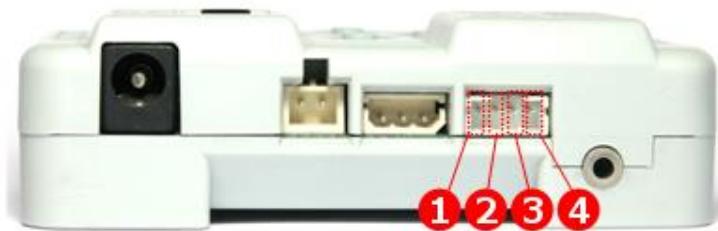
No. 03

LABSHEET PRAKTIK ROBOTIKA

Mengendalikan *Quadruped* Robot
dengan *Stick PS2*



Gambar 2. Susunan Pin pada Arduino
(Sumber: theengineeringprojects.com)



1. GND : Ground Level (0v)
2. VDD : Supply Voltage (2.7~3.6V)
3. RXD : Receive Signal Terminal
4. TXD : Transmit Signal Terminal

Gambar 3. Susunan Pin Komunikasi Serial pada CM-510
(Sumber: emanual.robotis.com)

Mengirim data pada kontroler arduino tidak dapat langsung dilakukan dengan mengirimkan data serial berupa data *integer* maupun *character*. Data yang dikirimkan agar dapat terbaca oleh kontroler CM-510 maupun kontroler-kontroler lain dari Robotis diperlukan pengiriman serial menggunakan pointer. Pointer adalah penunjuk suatu variabel. Karena menunjuk suatu variabel, maka pointer wajib memiliki alamat dari variabel yang ditunjuknya. Kadangkala dalam program yang besar, penghematan memori wajib untuk dilakukan. Dengan mekanisme *copy* dan *paste* nilai variabel satu kedalam variabel lain, akan sangat memboroskan memori. Dengan mekanisme pointer, suatu variabel dalam suatu fungsi dapat diakses oleh fungsi yang lain. Pointer yang dapat digunakan dalam arduino untuk mengirim kondisi ke kontroler CM-510 dapat dilihat pada tabel berikut ini.



**JURUSAN PENDIDIKAN TEKNIK ELEKTRO
FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA**

LABSHEET PRAKTIK ROBOTIKA

No. 03

Mengendalikan *Quadruped* Robot
dengan *Stick PS2*

Tabel 1. Tabel Pointer Arduino ke CM-510

No.	Alamat Pointer CM-510 pada Arduino	Pembacaan pada CM-510
1.	{0xAA, 0x01, 0x0E, 0xAA, 0x00, 0x1E};	U
2.	{0xAA, 0x04, 0x5E, 0xAA, 0x00, 0x1E};	D
3.	{0xAA, 0x08, 0x1E, 0xAA, 0x00, 0x1E};	L
4.	{0xAA, 0x02, 0x7E, 0xAA, 0x00, 0x1E};	R
5.	{0xAA, 0x24, 0x5E, 0xAA, 0x00, 0x1E};	L+2
6.	{0xAA, 0x28, 0x1E, 0xAA, 0x00, 0x1E};	R+2
7.	{0xAA, 0x40, 0x1E, 0xAA, 0x00, 0x1E};	3
8.	{0xAA, 0x10, 0x1E, 0xAA, 0x00, 0x1E};	1
9.	{0xAA, 0x00, 0x0F, 0xAA, 0x00, 0x1E};	4

Penggunaan dari alamat pointer CM-510 tersebut dalam arduino perlu diawali dengan pendeklarasian variabel. Dilanjutkan dengan mengubah alamat tersebut menjadi pointer. Perhatikan contoh berikut ini.

```
1. const uint8_t moveForward[6] = {0xAA, 0x01, 0x0E, 0xAA, 0x00, 0x1E};  
2. const uint8_t* moveFoward_ptr = moveForward;
```

Pada program tersebut alamat yang berisi untuk mengirim data pada kontroler CM-510 perintah diawali dengan pendeklarasian variabel bertipe *const uint8_t* dengan nama “moveForward[6]”. Baris berikutnya mengubah variabel yang dideklarasikan sebelumnya menjadi variabel pointer yang akan dikirim dengan nama “moveFoward_ptr”. Pengiriman pointer ke kontroler CM-510 memiliki *baudrate* khusus yaitu 1900. Perhatikan program dibawah ini digunakan untuk mengirim data pointer ke kontroler CM-510.

```
1. Serial.write(moveFoward_ptr, 3);  
2. delay(10);  
3. Serial.write(moveFoward_ptr+3, 3);
```

	JURUSAN PENDIDIKAN TEKNIK ELEKTRO FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA LABSHEET PRAKTIK ROBOTIKA	
No. 03	Mengendalikan <i>Quadruped</i> Robot dengan <i>Stick PS2</i>	

D. ALAT DAN BAHAN

1. *Quadruped* Robot
2. Kabel downloader USB RS232 to Audio Stereo 2.5
3. Kabel USB to Micro-USB
4. *Power supply* 12 V DC
5. Komputer / PC yang sudah ter-*install software* RoboPlus Manager, Motion, Task dan Arduino IDE

E. KESELAMATAN KERJA

1. Bekerjalah dengan keadaan tanpa tegangan saat memasang komponen pada *quadruped* robot.
2. Periksalah tegangan *output power supply*. Tegangan output yang diizinkan pada *quadruped* robot adalah 12VDC.
3. Pastikan semua komponen telah terpasang dengan benar sebelum menghubungkan *quadruped* robot dengan sumber tegangan.
4. Gunakan peralatan sesuai dengan fungsinya.
5. Duduklah di kursi komputer dengan baik dan benar, dengan minimal jarak mata dan layar komputer +30 cm. Disarankan menggunakan kacamata anti radiasi.
6. Jauhkan peralatan yang tidak digunakan dari meja kerja.
7. Bekerjalah sesuai dengan langkah kerja yang telah ditentukan.

F. LANGKAH KERJA

1. Hubungkan *downloader* dari komputer ke kontroler CM-510 pada *quadruped* robot.
2. Bukalah aplikasi Roboplus Motion, lalu klik *connect* untuk menghubungkan aplikasi dengan kontroler.
3. Selanjutnya Open file motion “job3.mtn” atau menggunakan file yang telah dibuat pada jobsheet 3.
4. *Download*-kan file tersebut pada kontroler *quadruped* robot.
5. Bukalah *software* Roboplus Task. Klik dua kali pada line program, lalu pilih kontroler CM-510.
6. Buatlah fungsi *Start Program* dilanjutkan dengan fungsi *Endless Loop* didalamnya.



**JURUSAN PENDIDIKAN TEKNIK ELEKTRO
FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA**

LABSHEET PRAKTIK ROBOTIKA

No. 03

Mengendalikan *Quadruped* Robot
dengan *Stick PS2*

7. Buatlah program seperti pada contoh di bawah ini untuk membuat robot bergerak saat menerima input dari arduino.

```
1  START PROGRAM
2  {
3      Motion Index Number = 1
4      WAIT WHILE ( Motion Status == TRUE )
5      ENDLESS LOOP
6      {
7          IF ( Remocon Data Received == TRUE )
8          {
9              data_In = Remocon RXD
10             IF ( data_In == U )
11             {
12                 Motion Index Number = 3
13                 WAIT WHILE ( Motion Status == TRUE )
14                 Motion Index Number = 4
15                 WAIT WHILE ( Motion Status == TRUE )
16             }
17         }
18     }
19 }
20 }
```

Keterangan:

Pada program tersebut saat robot menerima data serial maka robot akan menjalankan rangkaian motion untuk bergerak maju.

8. Simpan dan *upload* program task yang sudah dibuat.
9. Buka aplikasi Arduino IDE dan buatlah project baru. Simpan project dengan nama sesuai yang diinginkan.
10. *Install library* PS2X_lib pada Arduino IDE.
11. Buatlah program *stick PS2* seperti pada contoh program berikut ini.

```
78. #include <PS2X_lib.h>
79.
80. #define PS2_DAT      11
81. #define PS2_CMD      12
82. #define PS2_SEL      10
83. #define PS2_CLK      13
84. #define pressures  false
```



JURUSAN PENDIDIKAN TEKNIK ELEKTRO
FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA

LABSHEET PRAKTIK ROBOTIKA

No. 03

Mengendalikan *Quadruped* Robot
dengan *Stick PS2*

```
85. #define rumble      false
86.
87. #define Cek_Stick false
88.
89. int Stick = 0, LX, LY, RX, RY;
90. byte type = 0;
91. byte vibrate = 0;
92.
93. PS2X ps2x;
94.
95. void setup(){
96. //repair:
97. if(Cek_Stick == true)
98. {Serial.begin(9600);}
99. else {Serial.begin(1900);}
100.
101. repair:
102.
103. delay(300); //tunggu pairing wirless
104.
105. //setup pins and settings: GamePad (clock, command, attention,
   data, Pressures?, Rumble?) check for error
106.
107. Stick = ps2x.config_gamepad (PS2_CLK, PS2_CMD, PS2_SEL, PS2_DAT
   , pressures, rumble);
108.
109. if (Stick == 0){
110. Serial.print("Found Controller");
111. else {
112. Serial.print("Not Found Controller");
113. goto repair;}
114. }
115.
116. void loop() {
117. // Digunakan untuk memprogram pengiriman data ke kontroler CM-
   510
118. }
119.
120. void dualShock()
121. {
122. ps2x.read_gamepad(true, vibrate);
123. LY=(ps2x.Analog(PSS_LY));
124. LX=(ps2x.Analog(PSS_LX));
125. RY=(ps2x.Analog(PSS_RY));
126. RX=(ps2x.Analog(PSS_RX));
127.
128. if(ps2x.Button(PSB_START))      {Serial.println("start");}
   else
129. if(ps2x.Button(PSB_SELECT))    {Serial.println("Select");}
   else
130. if(ps2x.Button(PSB_PAD_UP))   {Serial.println("UP");}
   else
131. if(ps2x.Button(PSB_PAD_RIGHT)) {Serial.println("Right");}
   else
```



**JURUSAN PENDIDIKAN TEKNIK ELEKTRO
FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA**

LABSHEET PRAKTIK ROBOTIKA

No. 03

Mengendalikan *Quadruped* Robot
dengan *Stick PS2*

```
132. if(ps2x.Button(PSB_PAD_LEFT)) {Serial.println("Left"); }  
else  
133. if(ps2x.Button(PSB_PAD_DOWN)) {Serial.println("Down"); }  
else  
134. if(ps2x.Button(PSB_L1)) {Serial.println("L1"); }  
else  
135. if(ps2x.Button(PSB_L2)) {Serial.println("L2"); }  
else  
136. if(ps2x.Button(PSB_R1)) {Serial.println("R1"); }  
else  
137. if(ps2x.Button(PSB_R2)) {Serial.println("R2"); }  
else  
138. if(ps2x.Button(PSB_L3)) {Serial.println("L3"); }  
else  
139. if(ps2x.Button(PSB_R3)) {Serial.println("R3"); }  
else  
140. if(ps2x.Button(PSB_L2)) {Serial.println("L2"); }  
else  
141. if(ps2x.Button(PSB_R2)) {Serial.println("R2"); }  
else  
142. if(ps2x.Button(PSB_TRIANGLE)) {Serial.println("Segitiga"); }  
else  
143. if(ps2x.Button(PSB_CIRCLE)) {Serial.println("O"); }  
else  
144. if(ps2x.Button(PSB_CROSS)) {Serial.println("X"); }  
else  
145. if(ps2x.Button(PSB_SQUARE)) {Serial.println("Kotak"); }  
else  
146. if(LY==0) {Serial.println("L naik"); } else  
147. if(LY==255) {Serial.println("L Turun"); } else  
148. if(LX==255) {Serial.println("L kanan"); } else  
149. if(LX==0) {Serial.println("L kiri"); } else  
150. if(RY==0) {Serial.println("R naik"); } else  
151. if(RY==255) {Serial.println("R Turun"); } else  
152. if(RX==255) {Serial.println("R kanan"); } else  
153. if(RX==0) {Serial.println("R kiri"); }  
154. }
```

Keterangan:

Program arduino tersebut digunakan untuk menggunakan PS2 sebagai input pada arduino.

13. Selanjutnya pada buatlah program yang akan mengirim komunikasi serial pada kontroler CM-510 ketika diberi input berupa tombol pada *stick* ditekan. Data serial yang dikirim menggunakan fungsi pointer seperti yang dijelaskan pada kajian teori maupun pada modul. Tambahkan program pada bagian deklarasi variabel seperti pada contoh berikut ini.



JURUSAN PENDIDIKAN TEKNIK ELEKTRO
FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA
LABSHEET PRAKTIK ROBOTIKA

No. 03

Mengendalikan *Quadruped* Robot
dengan *Stick PS2*

```
1. const uint8_t moveForward[6] = {0xAA, 0x01, 0x0E, 0xAA, 0x00,  
    0x1E}  
2. const uint8_t* moveForward_ptr = moveForward;
```

Keterangan:

Program definisi tersebut digunakan untuk pointer dalam mengirim data serial ke arduino untuk menggerakkan robot maju.

14. Pada bagian fungsi *looping* program digunakan untuk mengirim serial yang telah di deklarasikan sebelumnya. Perhatikan program berikut ini.

```
33. void loop() {  
34.     if(Stick == 1) //repairing terus kalau gagal pair  
35.         return;  
36.     else  
37.     { //DualShock Controller  
38.  
39.         if(Cek_Stick == true)  
40.         { dualShock();}  
41.         else  
42.         {  
43.             ps2x.read_gamepad(true, vibrate);  
44.             LY=(ps2x.Analog(PSS_LY));  
45.             LX=(ps2x.Analog(PSS_LX));  
46.             RY=(ps2x.Analog(PSS_RY));  
47.             RX=(ps2x.Analog(PSS_RX));  
48.  
49.             if(LY==0)  
50.             {  
51.                 Serial.write(moveForward_ptr, 3);  
52.                 delay(10);  
53.                 Serial.write(moveForward_ptr+3, 3);  
54.             }  
55.             else if(ps2x.Button(PSB_PAD_UP))  
56.             {  
57.                 Serial.write(moveForward_ptr, 3);  
58.                 delay(10);  
59.                 Serial.write(moveForward_ptr+3, 3);  
60.             }  
61.         }  
62.     }  
63.     delay(10);  
64. }
```

Keterangan:

Program tersebut digunakan jika button maju ditekan atau analog kiri di arahkan ke depan maka program akan mengirim data pointer melalui serial ke kontroler CM510 dan di definisikan sebagai data *gamepad* tombol U.



JURUSAN PENDIDIKAN TEKNIK ELEKTRO FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA		
LABSHEET PRAKTIK ROBOTIKA		
No. 03	Mengendalikan <i>Quadruped</i> Robot dengan <i>Stick PS2</i>	

15. Hubungkan kabel downloader dari komputer ke arduino pada robot. *Download*-kan program tersebut. Setelah program selesai lakukan pengujian menekan tombol pada *stick* dan amati respon dari *quadruped* robot.

G. TUGAS

1. Buatlah program lanjutan jika *stick* ditekan tombol ke arah kanan atau pada *analog stick* diarahkan ke kanan maka robot akan bergerak berbelok ke kanan.
2. Buatlah program lanjutan jika *stick* ditekan tombol ke arah kiri atau pada *analog stick* diarahkan ke kiri maka robot akan bergerak berbelok ke kiri.

H. BAHAN DISKUSI

1. Jelaskan bagaimana respon dari *quadruped* robot ketika tombol pada *stick* ditekan.
2. Jelaskan hubungan dan fungsi Arduino terhadap kontroler CM-510 dan *Stick PS2*.