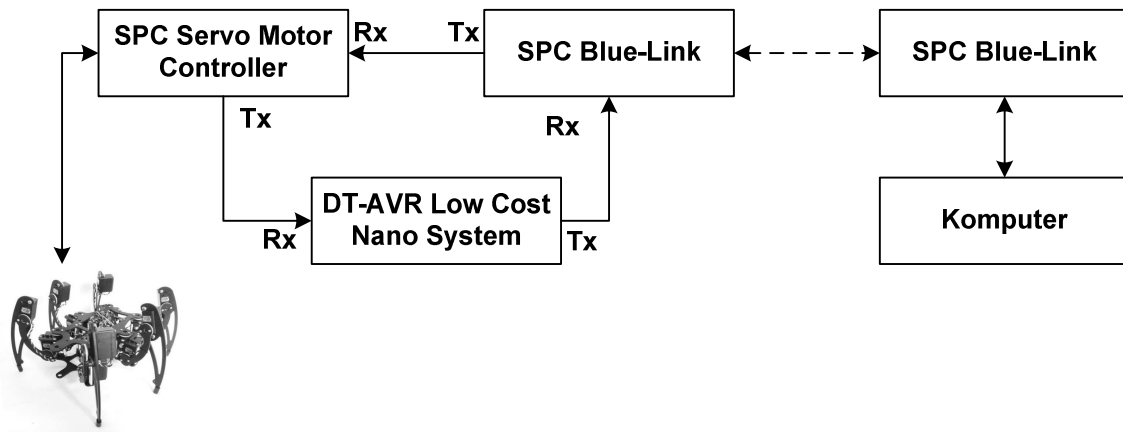


Hexapod robot merupakan *mobile* robot berkaki yang memiliki 6 buah kaki dengan 3 DOF pada setiap kakinya dan digerakan dengan bantuan motor servo. Rangka robot yang digunakan pada aplikasi ini adalah rangka hexapod robot MSR-H01 dari Micro Magic Systems yang memiliki 18 buah motor servo sebagai unit penggerak. Unit penggerak yang terdiri dari 18 motor servo dikendalikan oleh SPC Servo Motor Controller yang mampu mengendalikan 20 motor servo dan dilengkapi dengan antarmuka UART dan I2C. SPC Servo Controller dilengkapi kemampuan untuk menyimpan beberapa sekuens gerakan sehingga proses kendali gerakan hexapod robot menjadi sederhana, praktis, dan cukup mudah. Aplikasi ini dilengkapi dengan GUI (*Graphical User Interface*) yang dikembangkan dengan bahasa BASIC menggunakan IDE Microsoft Visual Basic 6.0 untuk mengendalikan hexapod robot secara *wireless* via SPC Blue-Link.

Modul yang dibutuhkan pada aplikasi ini adalah:

- 1 unit SPC Servo Controller.
- 1 unit MSR-H01 Hexapod Robot Kit (Kerangka robot hexapod yang dilengkapi dengan motor servo).
- 2 unit SPC Blue-Link.
- 1 unit DT-AVR Low Cost Nano System (ATtiny2313).
- 1 unit komputer dengan Sistem Operasi Microsoft Windows.

Blok diagram sistem secara keseluruhan sebagai berikut:



Gambar 1
Blok Diagram Sistem

Motor servo yang terpasang pada hexapod robot dihubungkan pada SPC Servo Motor Controller secara berurutan seperti ilustrasi berikut:



Gambar 2
Susunan Motor Servo pada Hexapod Robot

Pada setiap kaki hexapod robot, terdapat 3 buah motor servo. Hubungkan masing-masing motor servo SPC Servo Motor Controller sesuai dengan urutan yang diilustrasikan pada Gambar 2. Hubungkan motor servo nomor 1 pada konektor SERVO1 SPC Servo Motor Controller, motor servo nomor 2 pada konektor SERVO2 SPC Servo Motor Controller, dan seterusnya hingga seluruh motor servo terpasang pada modul SPC Servo Motor Controller. Hubungan antara SPC Servo Motor Controller dengan motor servo pada hexapod robot sebagai berikut:

Konektor SERVO1-SERVO20 (J3 – J20)	Pin Motor Servo Hexapod Robot (1 - 18)
Pin1 (SRV-GND)	Servo Ground (Hitam)
Pin2 (S-PWR)	Servo Power (Merah)
Pin3 (SERVOx)	Servo Data (Kuning)

Tabel 1
Hubungan antara Konektor SERVO SPC Servo Motor Controller dengan Pin Motor Servo Hexapod Robot

Pada aplikasi ini digunakan 2 buah modul SPC Blue-Link untuk proses kendali antara komputer dengan hexapod robot secara *wireless* menggunakan Bluetooth. Modul SPC Blue-Link yang terhubung dengan hexapod robot disebut SPC Blue-Link Slave dan modul SPC Blue-Link yang terhubung dengan komputer disebut SPC Blue-Link Master.

Pada bagian hexapod robot, SPC Blue-Link Slave terhubung dengan SPC Servo Motor Controller & DT-AVR LCNS melalui antarmuka UART TTL. Catu daya SPC Blue-Link Slave diperoleh dari *output* regulator 5 VDC (7805) DT-AVR LCNS. Oleh karena itu perlu dilakukan konfigurasi beberapa *jumper* pada ketiga modul tersebut sebagai berikut:

- J8 & J9 SPC Blue-Link Slave diatur pada posisi 2-3.

- J4 SPC Blue-Link Slave dibiarkan terbuka (tidak terhubung).
- J4 & J5 DT-AVR LCNS diatur pada posisi 2-3.

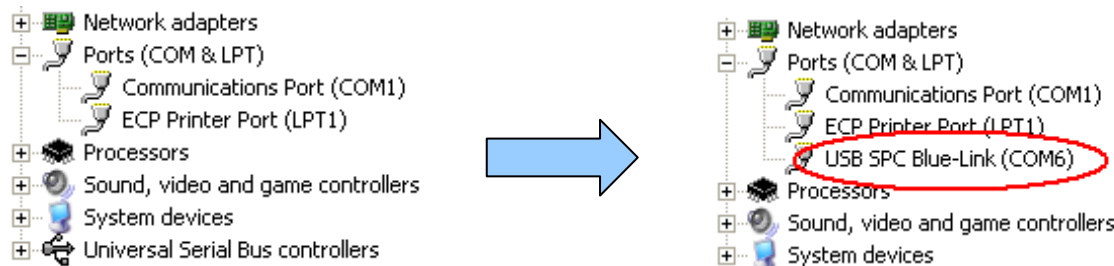
SPC Servo Motor Controller	SPC Blue-Link Slave	DT-AVR LCNS
J26 pin 3 (RX)	J3 pin 3 (TX)	-
-	J3 pin 2 (RX)	J8 pin 4 (TX/PD.1)
J26 pin 2 (TX)	-	J8 pin 3 (RX/PD.0)
J26 pin 1 (GND)	J3 pin 1 (GND)	J8 pin 1 (GND)

Tabel 2

Hubungan antara SPC Servo Motor Controller dengan SPC Blue-Link Slave dan DT-AVR LCNS

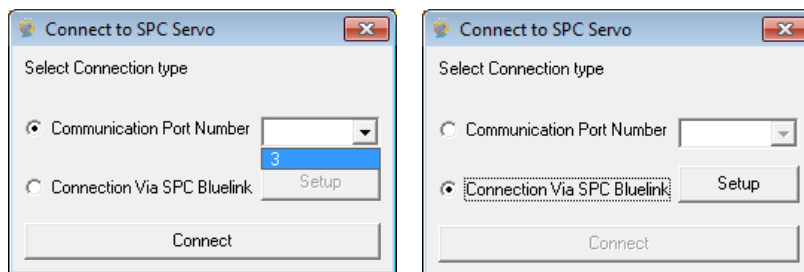
DT-AVR LCNS difungsikan sebagai jembatan komunikasi antara SPC Servo Motor Controller dengan SPC Blue-Link Slave khususnya dalam hal pengiriman data dari SPC Servo Motor Controller ke SPC Blue-Link Slave. Setiap *byte* data yang dikirimkan ke SPC Blue-Link memerlukan *delay* waktu, oleh karena itu program pada DT-AVR LCNS difungsikan untuk memberikan *delay* pada setiap *byte* data yang akan dikirimkan ke SPC Blue-Link.

SPC Blue-Link Master terhubung dengan komputer melalui antarmuka USB. Catu daya SPC Blue-Link Master diperoleh dari jalur USB (*bus power*) sehingga *jumper* J8 & J9 pada SPC Blue-Link Master perlu diatur pada posisi 1-2 dan *jumper* J4 pada posisi terpasang. Saat SPC Blue Link Master dihubungkan dengan *port* USB komputer maka akan muncul *Virtual COM Port (VCP)* pada komputer.



Gambar 3
Virtual COM Port (VCP) SPC Blue-Link pada Device Manager

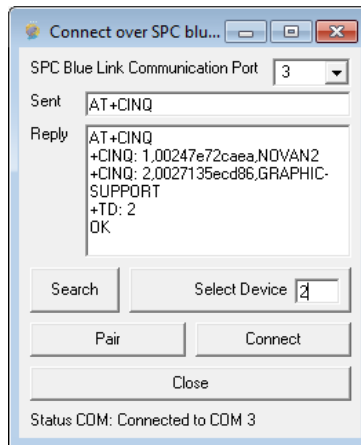
GUI pada aplikasi ini menyediakan 2 pilihan tipe koneksi yaitu menggunakan Bluetooth via SPC Blue-Link atau langsung menggunakan *port* serial yang ada. Pemilihan tipe koneksi yang akan digunakan dapat dipilih pada *form* Connect to SPC Servo seperti berikut ini:



Gambar 4
Form Koneksi ke SPC Servo Motor Controller

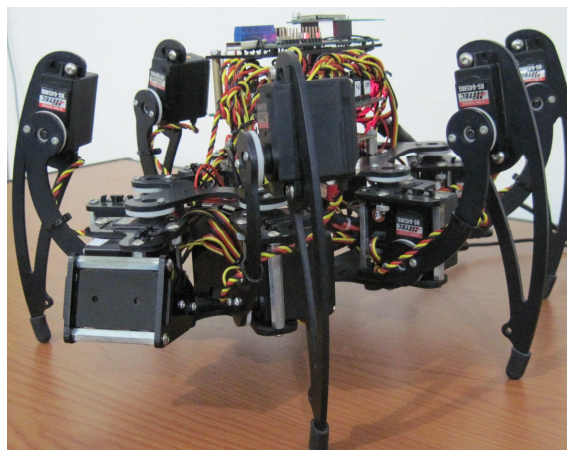
Port serial yang tersedia pada komputer akan ditampilkan pada *listbox* yang tersedia. Jika *user* memilih koneksi menggunakan SPC Blue-Link maka akan melewati beberapa langkah lagi seperti proses *searching* SPC Blue-Link lain, *pairing*, dan *connect* kemudian menuju *form* SPC_servo_control. Jika *user* memilih koneksi menggunakan *port* serial langsung maka proses *searching*, *pairing*, dan *connect* akan dilewati dan langsung menuju *form* SPC_servo_control. *Form* SPC_servo_control menyediakan pilihan untuk mengendalikan motor servo pada

hexapod robot secara manual maupun menggunakan sekuens gerakan yang telah ditentukan. Selain itu *user* juga dapat mendesain sendiri sekuens gerakan yang diinginkan kemudian menyimpannya pada *file* berekstensi *.seq*.



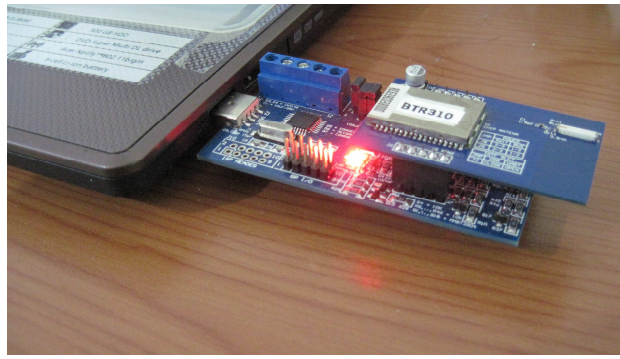
Gambar 5
Form Koneksi SPC Blue-Link

Pairing SPC Blue-Link dilakukan setelah SPC Blue-Link Slave terdeteksi oleh SPC Blue-Link Master. Proses *pairing* yang dimaksud adalah *pairing* antara SPC Blue-Link Master dengan SPC Blue-Link Slave. Sebelum melakukan *pairing*, isikan program *Tiny_bypass2.hex* ke dalam DT-AVR LCNS menggunakan DT-HiQ AVR USB ISP atau divais *programmer* lain yang kompatibel dan memiliki konektor ISP *standard* ATMEL. Kemudian hubungkan hexapod robot dengan catu daya yang sesuai, jika koneksi sudah benar maka LED *power* pada ketiga modul (DT-AVR LCNS, SPC Blue-Link, dan SPC Motor Controller) akan menyala dan hexapod robot berada pada posisi *standing up*. Catu daya untuk motor servo (VEXT) sebaiknya menggunakan catu daya yang mampu menyediakan kebutuhan arus dan tegangan yang cukup. Pada saat ujicoba digunakan catu daya sebesar 5V/8A untuk catu daya motor servo.



Gambar 6
Hexapod Robot pada Posisi Standing Up

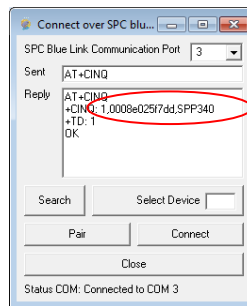
Kemudian hubungkan SPC Blue-Link Master ke *port* USB komputer. Jika pengaturan *jumper* pada SPC Blue-Link telah benar maka saat dihubungkan dengan *port* USB, LED indikator *power* pada SPC Blue-Link Master akan menyala. Pastikan SPC Blue-Link telah terdeteksi oleh komputer dan pada Device Manager telah muncul VCP (*Virtual COM Port*) SPC Blue-Link.



Gambar 7
SPC Blue-Link Master terhubung dengan Port USB Komputer

Prosedur *pairing* antar modul SPC Blue-Link sebagai berikut:

- Pilih “Connection via SPC Bluelink” pada *form* Connect to SPC Servo, lalu pilih setup. *Form* Connect Over SPC Blue-Link akan muncul seperti pada Gambar 8. Pilih *Communication port* yang terhubung ke SPC Blue-Link. Pilihan *port* serial yang tersedia ditampilkan pada *listbox*.
- Tekan tombol “Search”, hingga muncul *device* bernama SPP340.



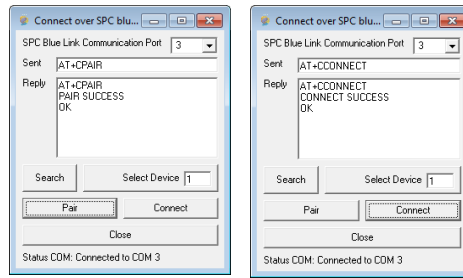
Gambar 8
Tampilan Form Connect Over SPC Blue-Link

- Setelah pencarian selesai, perhatikan pada hasil pencarian dengan *device name* SPP340. Karakter yang muncul setelah tulisan “+CINQ:” merupakan *index device* yang akan digunakan untuk melakukan *select device*.
- Masukkan *device index* dari SPP340 pada *text box* select device, lalu tekan select device, hingga muncul balasan seperti pada Gambar 9.



Gambar 9
Pemilihan Device Berdasarkan Device Index





- Setelah *device* terpilih, lakukan *pairing* dan *connecting* hingga muncul pesan sukses.



Gambar 10
Pesan Sukses Proses Pairing & Connect


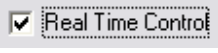
Setelah kedua modul SPC Blue-Link terkoneksi (muncul pesan CONNECT SUCCESS), kedua modul SPC Blue-Link telah keluar dari *command mode* dan masuk ke *data mode*. Hal ini berarti komputer telah terhubung langsung dengan SPC Servo Motor Controller secara serial via Bluetooth (*wireless*). Kemudian *form* SPC Hexapod Servo Manager akan ditampilkan.


Pada *starting form* SPC Hexapod Servo Manager, tab pengendalian manual akan ditampilkan. *User* dapat mengumpulkan informasi tentang posisi pulsa saat ini, mengirimkan pulsa yang diinginkan, maupun mengendalikan pulsa secara *realtime*. Selain itu juga disediakan terminal untuk mengirimkan *command* secara manual. Seluruh perintah yang terkirim secara otomatis dapat dimonitor melalui terminal tersebut. Pengendalian hexapod robot secara manual dapat dilakukan sebagai berikut:


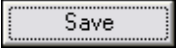

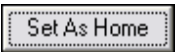
- Klik tombol  untuk mengembalikan posisi motor servo pada *home set*.
- Pilih tombol  untuk mendapatkan informasi nilai pulsa dari seluruh motor servo. Pulsa-pulsa tersebut akan ditampilkan pada *slider* dan *text box* tab Manual Control.
- Posisi *horn* motor servo (sudut) saat ini dapat diubah melalui *slider* atau angka pada *text box* kontrol motor servo yang bersangkutan, lalu tekan tombol .
- Robot hexapod akan bergerak berdasarkan pulsa servo yang diinputkan.
- Pengaturan ini bersifat *single execution*. Servo akan bergerak pada saat tombol  ditekan.

GUI ini juga dilengkapi dengan *real time servo control* untuk mengatur sudut/posisi *horn* motor servo secara *realtime*.

Proses kendali motor servo secara *realtime* dapat dilakukan sebagai berikut:

- Baca nilai pulsa motor servo saat ini dengan menekan tombol .
- Beri tanda centang pada pilihan Real Time Control. 
- Ubah nilai salah satu *slider* pengendali motor servo.
- Setiap perubahan pada *slider* maupun angka pada *text box* akan berpengaruh langsung pada motor servo hexapod robot.

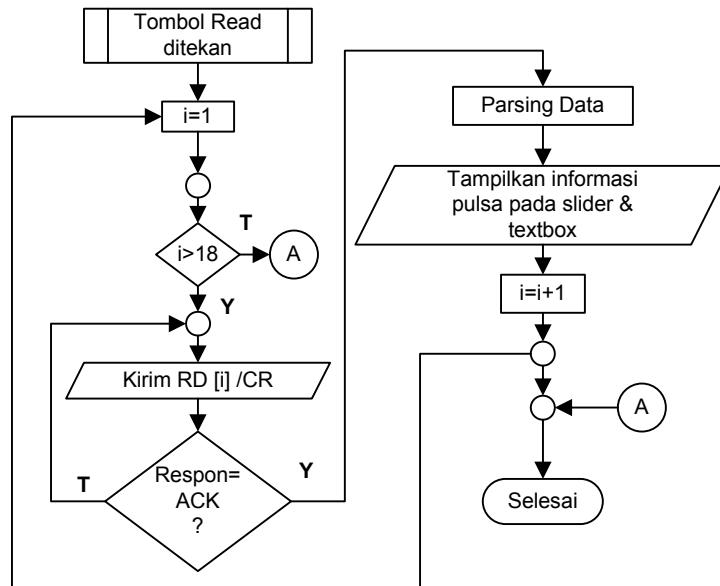
Lakukan prosedur tersebut pada posisi terminal aktif dengan menekan tombol , *command set* yang terkirim akan ditampilkan. Pada *manual control*, konfigurasi dari seluruh motor servo dapat disimpan dalam sebuah *file*, melalui prosedur berikut ini:

- Baca nilai pulsa motor servo saat ini dengan menekan tombol .
- Pilih tombol , data yang terbaca dari SPC Servo Motor Controller akan disimpan dalam *file* berformat “.ang”
- Pilih tombol  untuk membuka *file* konfigurasi dan memasukan datanya pada *slider* dan *textbox* tab *manual control*.
- Pilih tombol  untuk menyimpan konfigurasi motor servo pada *home position EEPROM* SPC

Servo Motor Controller.

- Konfigurasi tersebut akan dieksekusi saat tombol  ditekan, atau SPC servo Controller mengalami *restart*.

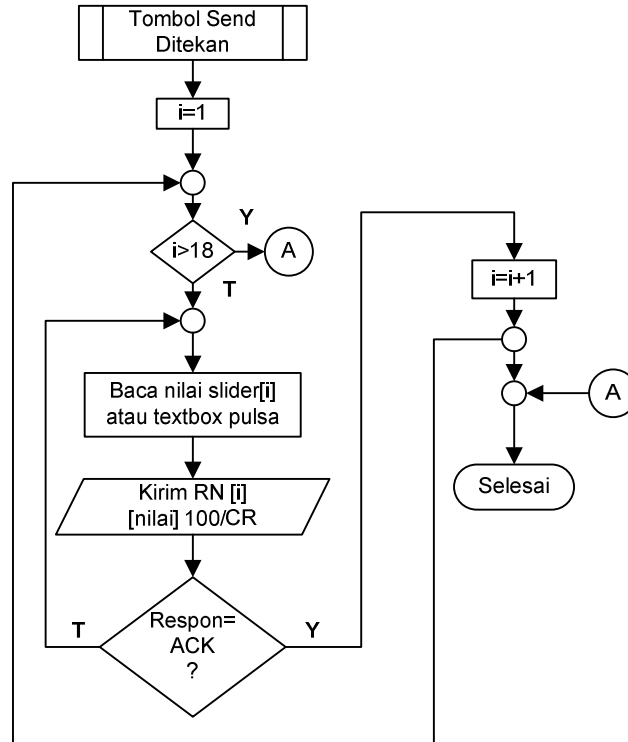
Flowchart rutin pembacaan pulsa motor servo (Read) sebagai berikut:



Gambar 11
Flowchart Penekanan Tombol Read

Secara umum saat tombol Read ditekan maka program akan mengirimkan perintah RD yang diikuti nomor motor servo (1-18) untuk membaca status & posisi motor servo. Selanjutnya program akan menunggu respon dari SPC Servo Motor Controller, jika respon yang diterima terdapat ACK maka program akan melakukan *parsing* data. Jika respon yang diterima bukan ACK maka program akan mengirim kembali perintah RD dengan nomor motor servo yang bersesuaian. Informasi posisi motor servo (yang diwakili nilai pulsa) dari hasil *parsing* data selanjutnya ditampilkan pada *slider* dan *textbox* nilai pulsa.

Flowchart rutin pengesetan pulsa motor servo (Send) sebagai berikut:

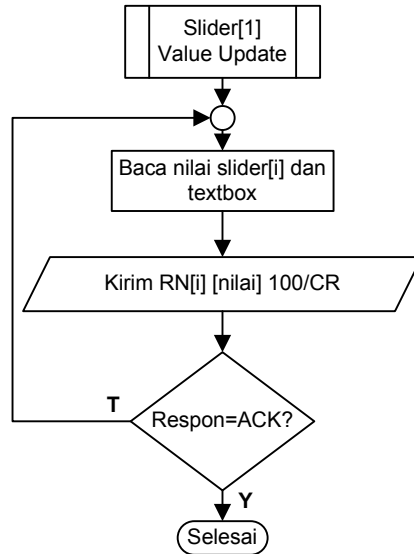


Gambar 12
Flowchart Penekanan Tombol Send

Secara umum saat tombol Send ditekan maka program akan mengirimkan perintah RN yang diikuti nomor motor servo (1-18), nilai pulsa, dan *servo rate*. Nilai pulsa diperoleh dari pembacaan nilai *slider* atau *textbox* nilai pulsa yang dilakukan sebelum perintah RN untuk masing-masing servo dikirim. Nilai *servo rate* ditetapkan pada nilai 100. Selanjutnya program akan menunggu respon dari SPC Servo Motor Controller, jika respon yang diterima bukan ACK maka program akan mengirim kembali perintah RN dengan nomor motor servo dan parameter yang bersesuaian.

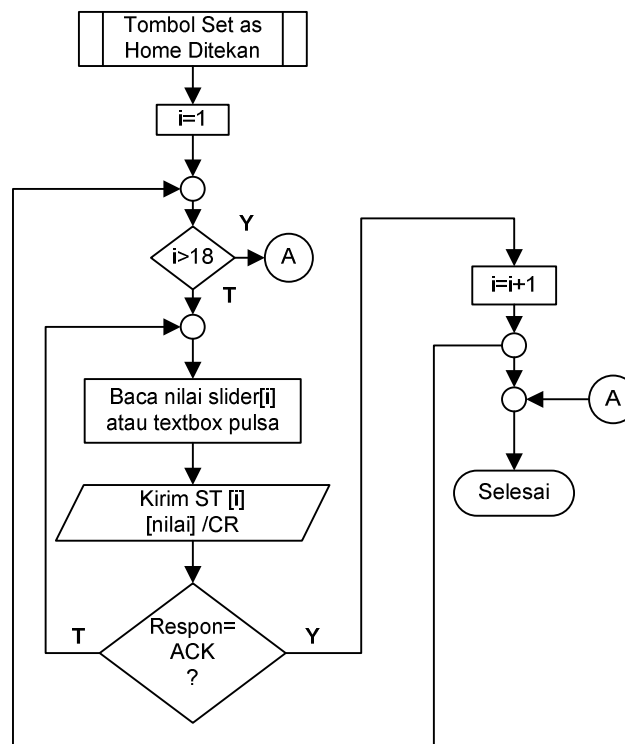
Saat mode *realtime* dipilih dengan memberikan tanda check (centang) maka setiap terjadi perubahan *slider* nilai pulsa program akan mengirimkan perintah RN yang diikuti nomor motor servo (1-18), nilai pulsa, dan *servo rate*. Nilai pulsa diperoleh dari pembacaan *slider* nilai pulsa. Nilai *servo rate* ditetapkan pada nilai 100. Selanjutnya program akan menunggu respon dari SPC Servo Motor Controller, jika respon yang diterima bukan ACK maka program akan mengirim kembali perintah RN dengan nomor motor servo dan parameter yang bersesuaian.

Flowchart program saat mode *realtime* dipilih sebagai berikut:



Gambar 13
Flowchart Update Slider

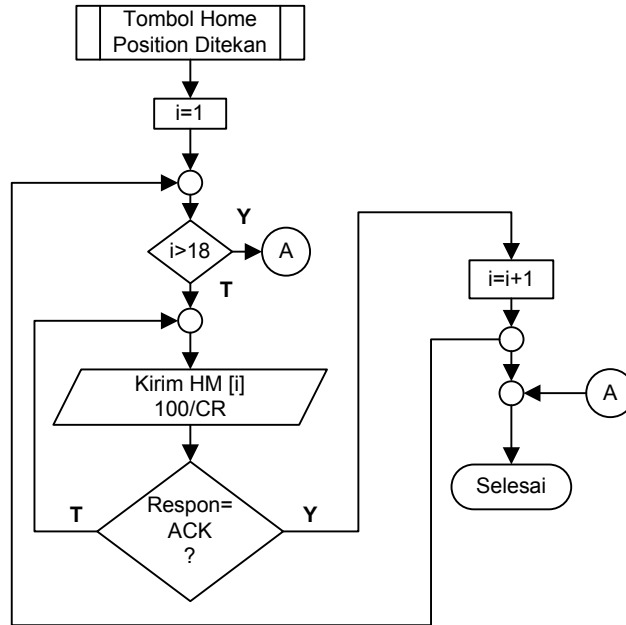
Flowchart penekanan tombol Set as Home sebagai berikut:



Gambar 14
Flowchart Penekanan Tombol Set as Home

Saat tombol Set as Home ditekan maka program akan mengirimkan perintah ST yang diikuti nomor motor servo (1-18) dan nilai pulsa yang mewakili nilai posisi home motor servo. Nilai pulsa diperoleh dari pembacaan nilai *slider* atau *textbox* nilai pulsa yang dilakukan sebelum perintah ST untuk masing-masing servo dikirim. Selanjutnya program akan menunggu respon dari SPC Servo Motor Controller, jika respon yang diterima bukan ACK maka program akan mengirim kembali perintah ST dengan nomor motor servo dan parameter yang bersesuaian. Nilai posisi *home* ini selanjutnya akan disimpan di EEPROM SPC Servo Motor Controller dan menjadi posisi *default* motor servo saat dinyalakan.

Flowchart pemanggilan posisi *home* (Home Position) sebagai berikut:

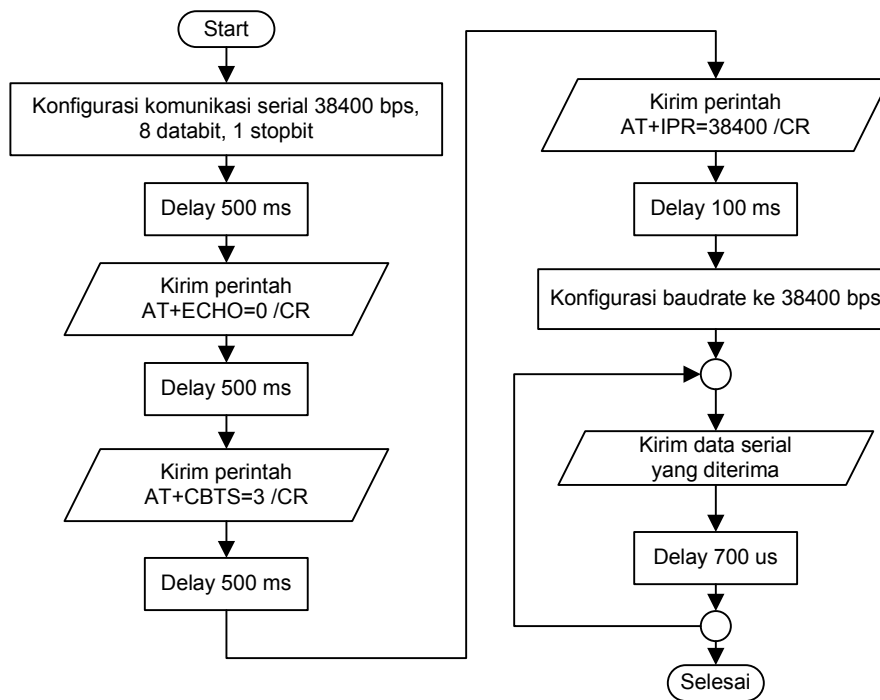


Gambar 15
Flowchart Penekanan Tombol Home Position

Secara umum saat tombol Home Position ditekan maka program akan mengirimkan perintah HM yang diikuti nomor motor servo (1-18) dan *servo rate*. Nilai *servo rate* ditetapkan pada nilai 100. Selanjutnya program akan menunggu respon dari SPC Servo Motor Controller, jika respon yang diterima bukan ACK maka program akan mengirim kembali perintah RN dengan nomor motor servo dan parameter yang bersesuaian. Penekanan tombol ini akan menyebabkan seluruh motor servo pada hexapod robot berada pada posisi *home/default*.

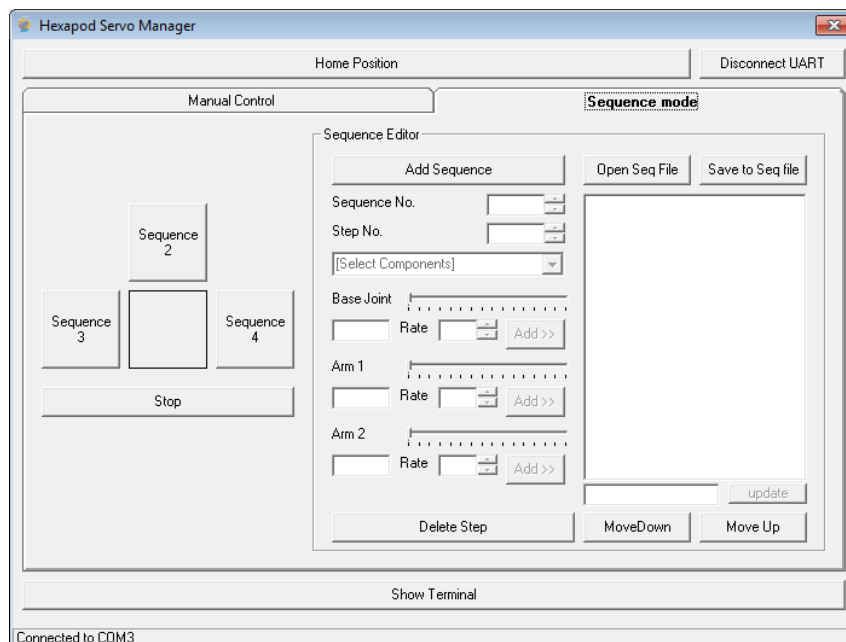
Program pada DT-AVR LCMS difungsikan untuk melakukan inisialisasi awal SPC Blue-Link yang terhubung dengan hexapod robot. Fungsi utama program tersebut adalah mem-*forward*-kan data yang diterima dari SPC Servo Motor Controller dengan penambahan *delay* sehingga data dapat diterima oleh SPC Blue-Link kemudian ditransmisikan. Program pada DT-AVR LCMS ditulis dalam bahasa C menggunakan IDE CodeVisionAVR versi Evaluasi.

Pada awal program *Tiny_bypass2.c*, program akan melakukan konfigurasi komunikasi serial pada 9600 bps, 8 databit, dan 1 stopbit untuk komunikasi dengan SPC Blue-Link. Hal ini dikarenakan setelah *power-up* nilai *baudrate* komunikasi serial SPC Blue-Link diatur pada nilai 9600 bps. Selanjutnya program akan mengirimkan perintah AT+ECHO=0 untuk mematikan fitur echo SPC Blue-Link dan dilanjutkan pengiriman perintah AT+CBTS=3 untuk mengubah status modul agar dapat dihubungi oleh modul Bluetooth lainnya. Program kemudian mengirimkan perintah AT+IPR=38400 untuk mengubah nilai *baudrate* komunikasi serial SPC Blue-Link ke nilai 38400 bps sehingga komunikasi antara SPC Blue-Link dengan SPC Servo Motor Controller dapat berjalan dengan baik. Selanjutnya program juga akan mengubah konfigurasi serial DT-AVR LCMS ke nilai 38400 bps untuk keperluan komunikasi serial antara SPC Blue-Link dengan SPC Servo Motor Controller via DT-AVR LCMS.



Gambar 16
Flowchart Program Tiny_bypass2.c


Pada *sequence mode*, tersedia *tools* untuk merancang sekuens pergerakan hexapod robot. Setiap sekuens gerakan terdiri dari beberapa step gerakan motor servo bagian base joint, arm1, dan arm2. Pada setiap step, *user* dapat mengatur nilai sudut *horn* motor servo melalui pemberian nilai pulsa dan kecepatan motor servo (*rate*) untuk masing-masing bagian (base joint/arm1/arm2). Jika step gerakan untuk sekuens tertentu dirasa sudah sesuai dengan kebutuhan maka sekuens tersebut dapat disimpan untuk keperluan pergerakan robot.




Gambar 17
Menu Sequence mode

Di bagian kiri tab *Sequence mode*, terdapat tombol-tombol aktivasi sekuens gerakan hexapod robot. Tombol tersebut dapat digunakan untuk menjalankan suatu sekuens yang telah tertulis pada *EEPROM SPC Servo Controller*. Bagian sekuens *editor* digunakan untuk merancang sekuens pergerakan hexapod robot. Simpan sekuens yang telah dirancang ke dalam file berekstensi *.seq*, lalu tuliskan sekuens tersebut pada *EEPROM SPC*


Servo Motor Controller menggunakan aplikasi SPC servoCon.exe.

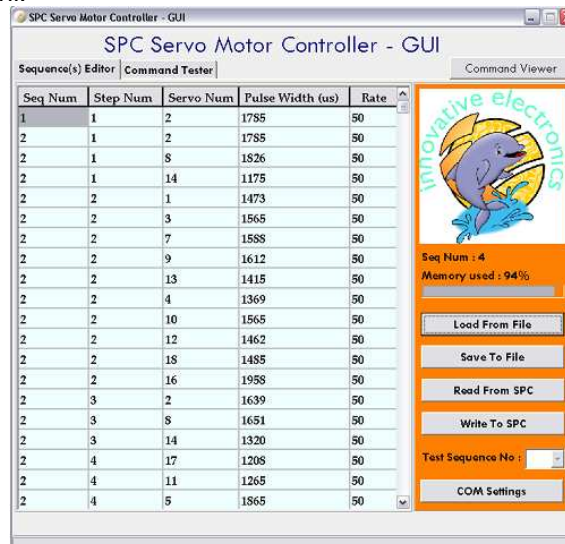
Jika ingin merancang sebuah sekuens gerakan sekuens, tekan tombol . Kemudian *textbox* sequence dan step akan aktif. Kemudian ikuti prosedur berikut ini:

- Tuliskan nomor sekuens & step pada *textbox* sequence no dan step no sesuai dengan yang diinginkan. Pada satu sekuens, step yang sama akan dijalankan bersamaan sedangkan step yang berurutan dijalankan secara berurutan pula.
- Setelah mengisi nomor sequence dan step, pilih komponen yang akan digerakkan pada sekuens tersebut.
- Selanjutnya gunakan *slider* maupun *textbox* untuk mengubah posisi komponen yang dikehendaki. Pergerakan akan teraktualisasi secara realtime pada hexapod robot.
- Kemudian tekan add di sebelah kanan *slider* untuk mendaftarkan pergerakan tersebut pada sekuens gerakan.

Setelah sekuens gerakan terdaftar, simpan sebagai *file* berekstensi .seq untuk dikirim ke dalam EEPROM SPC Servo Motor Controller melalui aplikasi servoCon.exe. Sebagai contoh, sequence14.seq akan dikirimkan ke EEPROM SPC Servo Controller. Tutup aplikasi SPC4hexapod.exe atau tekan , kemudian jalankan aplikasi servoCon.exe.

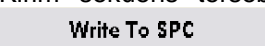
Pada tab sequence editor, terdapat antarmuka untuk membuat sekuens dengan tampilan tabel. Prosedur untuk menuliskan sequence14.seq pada EEPROM SPC Servo Motor Controller sebagai berikut:

- Tekan tombol  buka *file* sequence14.seq, maka susunan sekuensial akan tampil pada tabel seperti berikut ini:



Seq Num	Step Num	Servo Num	Pulse Width (us)	Rate
1	1	2	1785	50
2	1	2	1785	50
2	1	8	1826	50
2	1	14	1175	50
2	2	1	1473	50
2	2	3	1565	50
2	2	7	1588	50
2	2	9	1612	50
2	2	13	1415	50
2	2	4	1369	50
2	2	10	1565	50
2	2	12	1462	50
2	2	18	1485	50
2	2	16	1958	50
2	3	2	1639	50
2	3	8	1651	50
2	3	14	1320	50
2	4	17	1208	50
2	4	11	1265	50
2	4	5	1865	50

Gambar 18
Tampilan SPC Servo Motor Controller GUI

- Kirim sekuens tersebut ke dalam EEPROM SPC Servo Motor Controller dengan menekan tombol , maka akan muncul pesan sebagai berikut:



Gambar 19
Progress Bar Pengisian EEPROM SPC Servo Motor Controller

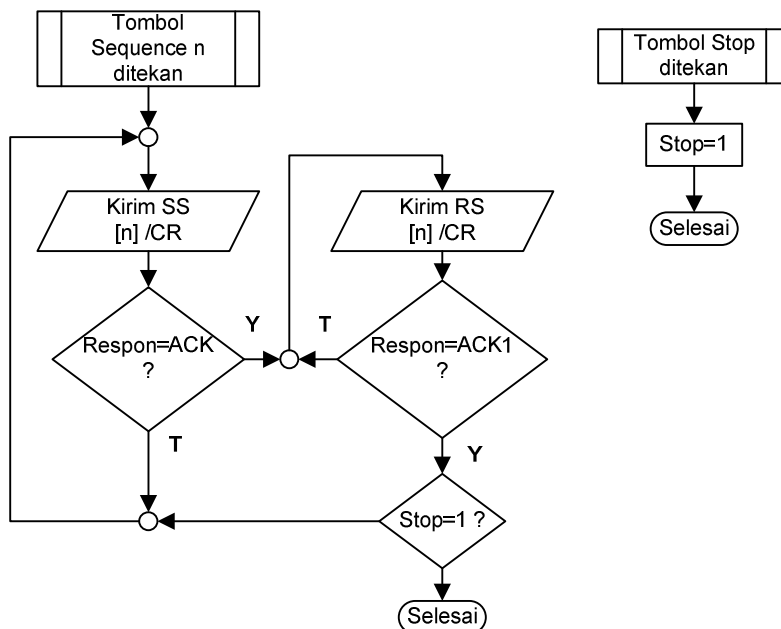
- Tunggu hingga *progress bar* terisi penuh dan proses penulisan hingga 100%.

Setelah proses pengisian selesai, maka hexapod robot telah memiliki sekuens gerakan yang telah dirancang sebelumnya di dalam EEPROM SPC Servo Motor Controller. Sekuens gerakan tersebut dapat dipanggil sewaktu-

waktu dengan mengirimkan perintah perintah “SS” diikuti indeks sekuens yang diinginkan. Kombinasi sekuens tersebut akan menghasilkan gerakan-gerakan hexapod robot.

Program SPC4hexapod.exe bagian tab sequence mode menyediakan *trigger* untuk menjalankan sekuens gerakan yang telah tersimpan pada EEPROM SPC Servo Motor Controller. Saat tombol sekuens ditekan maka program akan mengirimkan perintah aktivasi sekuens (“SS”) secara berulang-ulang hingga tombol Stop ditekan. Aktifnya beberapa kombinasi sekuens secara berurutan akan membentuk gerakan-gerakan yang beraturan hingga robot bergerak sesuai urutan gerakan yang diinginkan.

Flowchart penekanan tombol run sekuens sebagai berikut:



Gambar 20
Flowchart Penekanan Tombol Run Sequence & Stop

Jika salah satu tombol sekuens gerakan ditekan maka program akan mengirimkan perintah SS diikuti parameter nomor sekuens gerakan yang ingin dijalankan. Sekali tombol sekuens tertentu ditekan maka program akan terus menerus mengirimkan perintah SS (*Start Sequence*) hingga terjadi penekanan tombol Stop.

Perhatian:

- Pastikan catu daya yang digunakan untuk motor servo hexapod robot mampu menyediakan kebutuhan tegangan dan arus seluruh motor servo.
- Jika menggunakan Sistem Operasi Windows 7, jalankan program spc4hexapod.exe dengan **Run as Administrator**.
- Jika sebelumnya belum pernah melakukan instalasi Microsoft Visual Basic 6.0, *copy*-kan file TABCTL32.OCX dan TABCTL32.OCA ke direktori Windows | System32 atau secara *default* berada pada lokasi C:\Windows\System32.

Listing program terdapat pada **AN181.zip**

Video demo hexapod robot dapat di-*download* dari *link* berikut ini <http://www.youtube.com/watch?v=Mx-PpQdr4nY>

Selamat berinovasi !

All trademarks, trade names, company names, and product names are the property of their respective owners. All softwares are copyright by their respective software publishers and/or creators.