

Aktuator Berkaki pada Robot Pencari Korban Bencana Tsunami

Sub judul : Algoritma Reinforcement Learning

Adnan Rachmat Anom Besari, Dr.Ir. Son Kuswadi
Jurusan Elektronika- Politeknik Elektronika Negeri Surabaya
Institut Teknologi Sepuluh Nopember
Kampus ITS Keputih Sukolilo Surabaya 60111 Telp. 031-5947280, Fax 031-5946114

ABSTRAK

. Penelitian SAR (Search and Rescue) Robot di seluruh dunia semakin marak, tetapi belum pernah dicetuskan pengembangan SAR Robot untuk menangani korban Tsunami, bencana yang menjadi langganan negara kepulauan. Pengembangan SAR Robot yang paling banyak dikembangkan sekarang adalah jenis robot beroda, karenanya penelitian ini mencoba untuk mengembagkan aktuator alternatif yang lain, yaitu aktuator berkaki. Perkembangan robot berkaki sangat pesat, hal ini bisa dilihat dari bermunculan ide-ide baru tentang sistem gerak robot yang mengambil ide dari alam (biology inspired) tersebut. Robot jenis ini sekarang mulai banyak menggantikan robot dengan aktuator beroda dengan berbagai fungsi dan kemampuan manuvernya yang sangat menarik.

Untuk mewujudkan sistem kontrol cerdas pada robot yang tidak bergantung pada model, penelitian ini menggunakan algoritma kontrol behavior based. Kemampuan belajar robot dalam melakukan navigasi didapatkan dengan menggunakan metode reinforcement learning(RL). Untuk mencapai tujuan ini, digunakan algoritma tryal and error. Proses pembelajaran dilakukan melalui interaksi sistem dengan lingkungannya berdasarkan suatu aturan reward and punishment. Lingkungan disini artinya segala sesuatu di luar robot. Oleh karena itu, manusia juga juga dapat disebut lingkungan oleh robot. Akan sangat menarik jika manusia dan robot dapat saling berinteraksi. Apa yang manusia inginkan akan coba dimengerti dan dipahami oleh robot, dan kemudian dilaksanakan. Itulah reinforcement learning.

Kata kunci : reinforcement learning, reward and punishment.

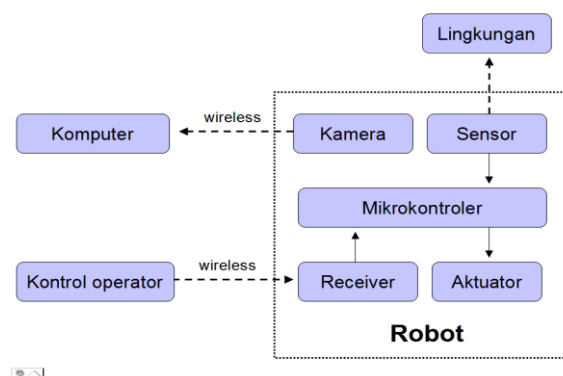
1. PENDAHULUAN

Proyek akhir ini mengambil bencana tsunami atau gempa bumi sebagai isu sentralnya, karena bencana ini merupakan bencana yang sering menimpa Indonesia sebagai negara yang banyak dilewati oleh jalur pegunungan. Fungsi robot disini menggantikan tim SAR (Search and Rescue) yang mempunyai tugas melewati daerah bencana yang kondisinya tidak tentu (walk on uncertain field), mendeteksi kemungkinan adanya korban (search), dan memberitahukan kepada tim SAR untuk mengevakuasi korban (rescue).

Proyek akhir ini merancang sebuah robot dengan aktuator berkaki yang mempunyai kemampuan melewati daerah bencana. Daerah bencana sendiri digantikan dengan sebuah medan (field) yang menggambarkan kondisi daerah akibat bencana tsunami. Sedangkan korban manusia digantikan dengan boneka yang menyerupai manusia dalam berbagai keadaan (meninggal atau masih hidup). Dalam proyek akhir nanti, robot diharuskan mencari dan medeteksi adanya korban bencana, lalu memberitahukan kepada tim SAR bahwa ada korban yang harus diselamatkan.

2. DISAIN SISTEM

Secara garis besar sistem yang dibangun ditunjukkan pada blok diagram sebagai berikut :



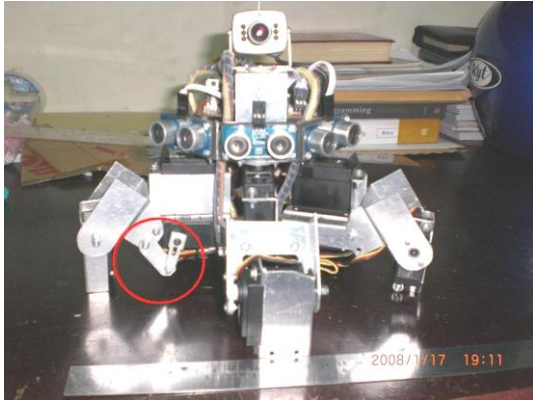
Gambar 1. Blok diagram rancangan sistem

Pembuatan sistem dalam proyek akhir ini dibagi dalam beberapa bagian sebagai berikut :

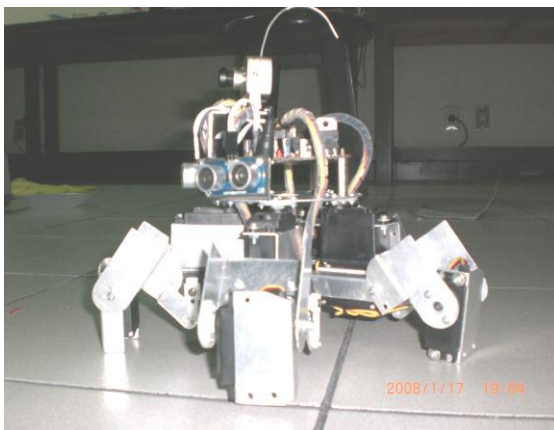
1. Pemilihan aktuator dan sensor
2. Pembuatan mekanik robot
3. Pembuatan perangkat keras
4. Pembuatan perangkat lunak

2.1 Perancangan Mekanik

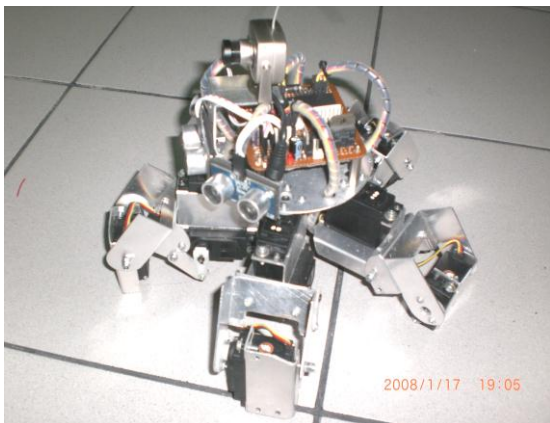
Berikut ini beberapa gambar desain mekanik robot dilihat dari beberapa sudut pandang.



Gambar 2. Robot tampak depan



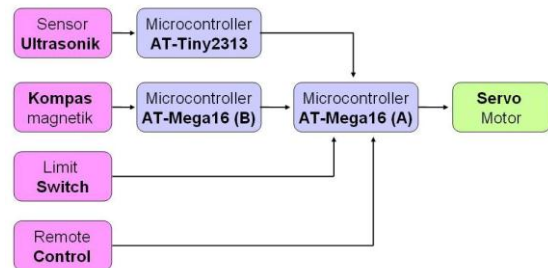
Gambar 3. Robot tampak samping



Gambar 4. Robot tampak atas

2.2 Pembuatan Perangkat Keras

Perangkat keras yang kami gunakan menggunakan dua jenis mikrokontroler, yaitu **ATtiny2313** dan **ATmega16** dua buah (A&B) dengan pembagian tugas seperti pada blok diagram dibawah, hal ini bertujuan agar mengurangi kinerja dari mikrokontroler utama dan mempermudah dalam pembagian kerja agar didapatkan hasil yang *real time*.



Gambar 5.

Blok diagram rancangan sistem hardware

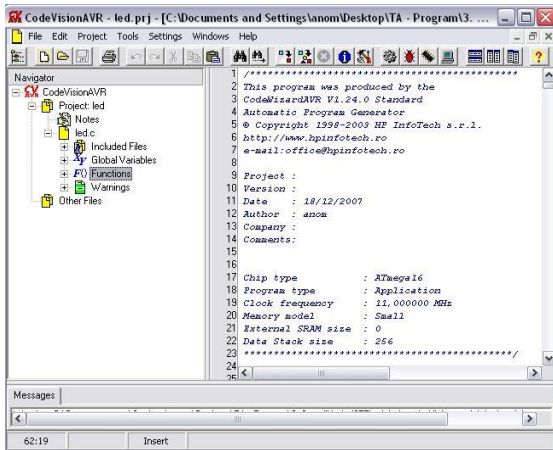
- **ATmega16 (A)** sebagai mikrokontroler utama menangani Atmega16 (B), ATtiny2313, remote control dan limit switch. Sebagai outputnya, ATmega16 (A) menggerakkan motor servo.
- **ATmega16 (B)** sebagai mikrokontroler pembantu menangani kompas magnetik dan mengirimkan data secara paralel ke ATmega16 (A).
- **ATtiny2313** digunakan untuk mengendalikan *ultrasonic sensor*. dan mengirimkan data secara serial (USART) ke ATmega16 (A).

2.3 Pembuatan Perangkat Lunak

Perencanaan dan pembuatan perangkat lunak ini menggunakan bahasa pemrograman C. Selain dikenal sebagai bahasa pemrograman yang mudah dimengerti, bahasa C dikenal juga dengan tingkat kecepatan pemrosesannya yang tinggi. Compiler yang digunakan adalah "**CodeVisionAVR C Compiler**" versi 1.24.0.

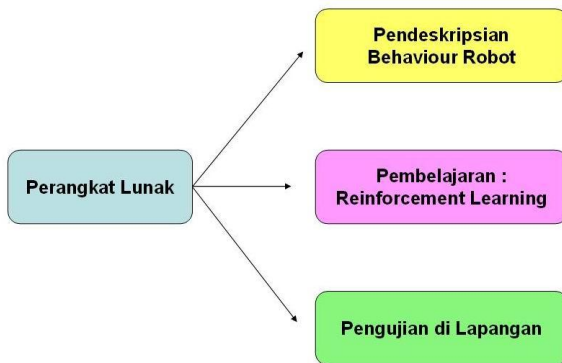
Compiler ini dipilih karena terdapat banyak kelebihan dibanding dengan aplikasi compiler lainnya, seperti :

- Merupakan C Cross Compiler dengan bahasa C sebagai bahasa pemrogramannya (disamping Assembly).
- Memiliki In-System Programmer Software yang didesain untuk banyak sistem (AVRISP, Kanda System, dsb.)
- Memiliki Code WizardAVR Automatic Program Generator untuk memudahkan programmer melakukan setting beberapa fungsi tanpa menyetikkan program satu-persatu.



Gambar 6. Tampilan CodeVisionAVR C Compiler

Program yang dibuat secara umum dibagi dalam beberapa bagian seperti terlihat pada bagan dibawah ini :



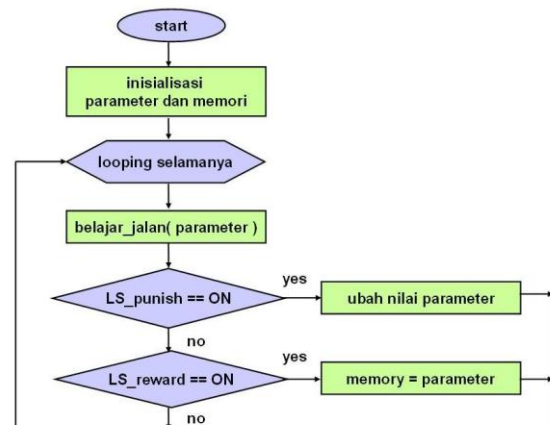
Gambar 7. Pembagian sistem kerja software

Sebelum sebuah program dibuat, terlebih dulu disusun algoritma yang menggambarkan secara singkat dan tepat pola pikir program. Secara garis besar penjelasan pembagian perangkat lunak yang dibuat seperti di bawah ini. :

- **Pendeskrisian Behaviour** : meliputi penentuan gerakan-gerakan apa saja yang dapat dilakukan oleh robot.
- **Pembelajaran (Reinforcement Learning)** : meliputi pengujian tiap-tiap behaviour dengan fungsi pembelajaran agar didapatkan hasil yang optimal pada masing-masing behaviour.
- **Pengujian di Lapangan** : meliputi pengujian hasil pembelajaran terhadap tiap-tiap behaviour pada lapangan dengan beberapa kondisi yang berbeda-beda.

2.3.1 Algoritma Reinforcement Learning

Pada tulisan ini dibahas perancangan metode *reinforcement learning* pada robot untuk menentukan gerakan optimal pada tiap-tiap behaviour. Behaviour-behaviour tersebut dideskripsikan oleh fungsi yang berbeda-beda. Pada masing-masing fungsi terdapat terdapat parameter yang sudah ditetapkan terlebih dahulu. Parameter inilah yang menjadi gerakan awal untuk dinilai. Parameter ini berubah nilai agar didapatkan parameter pembelajaran yang tepat. Penilaiannya dilakukan dengan pemberian “reward” bila gerakan yang dihasilkan robot telah sesuai dengan keinginan sang guru (master) atau dengan pemberian “punishment” bila gerakan yang dihasilkan robot tidak sesuai. Jika yang diterima adalah “punishment” berarti nilai parameter berubah dan jika yang diterima adalah “reward” berarti nilai parameter disimpan di memori. Secara umum algoritma *reinforcement learning* yang dirancang seperti di bawah ini.

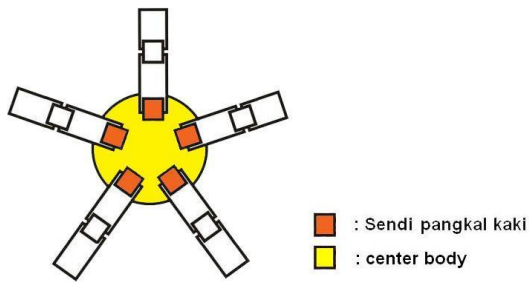


Gambar 8. Algoritma Reinforcement Learning

Algoritma diatas merupakan algoritma Reinforcement Learning yang secara umum dipakai pada proyek akhir ini. Berikut ini dibahas beberapa parameter yang berubah sesuai dengan proses pembelajaran Reinforcement Learning pada robot.

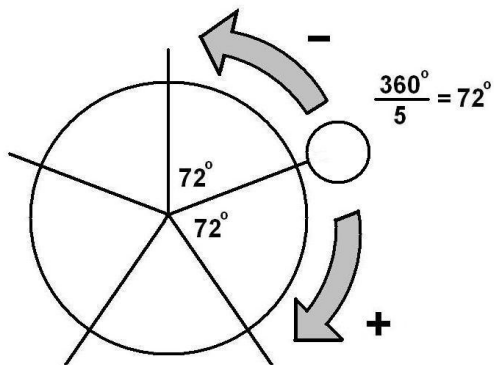
2.3.2 Parameter : Sudut pada Sendi Pangkal Kaki

Sudut pada sendi pangkal kaki yang dimaksud adalah sudut yang dapat dijangkau oleh pergerakan sendi kaki yang berada paling dekat tubuh robot (*center body*).



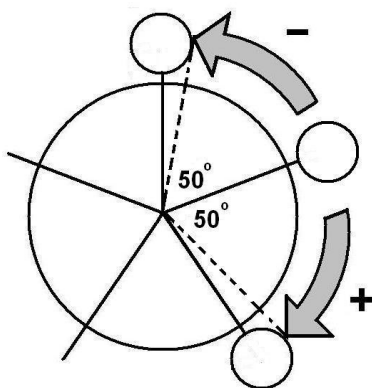
Gambar 9. Ilustrasi sendi pangkal kaki

Karena ada 5 kaki maka antara kaki satu dengan kaki lainnya berjarak 72° . Sehingga setiap kaki dapat bergerak $+72^\circ$ (searah jarum jam) dan -72° (berlawanan jarum jam).



Gambar 10. Sudut yang dibentuk sendi pangkal kaki

Tetapi pada kenyataannya robot hanya dapat menangkau -50° sampai $+50^\circ$ derajat saja dikarenakan mekanik kaki robot yang memerlukan tempat.



Gambar 11. Pengurangan sudut sendi pangkal kaki

2.3.3 Parameter : Waktu Tunda (Delay)

Kecepatan berjalan robot yang dimaksud disini adalah kecepatan yang dihasilkan dari *delay* (waktu tunda) antara gerakan sendi yang satu dengan gerakan sendi lainnya pada kaki robot. Sehingga parameternya adalah nilai yang diberikan pada fungsi *delay* yang ada pada program mikrokontroler. Prototipe fungsi ini ada pada "**delay.h**" sehingga harus menambahkan "**#include<delay.h>**" di awal program. Ada dua fungsi *delay* yang disediakan oleh **CodeVisionAVR**, yaitu :

- **void delay_ms** (unsigned int n)
Fungsi ini memberikan waktu *delay* (tundaan) dalam hitungan milisecond dengan parameter bertipe **unsigned integer**.
- **void delay_us** (unsigned int n)
Fungsi ini memberikan waktu *delay* (tundaan) dalam hitungan microsecond dengan parameter bertipe **unsigned integer**.

Sebelum memanggil fungsi ini, **interrupt** akan dimatikan, dan waktu tundaan akan lebih lama dari yang diperkirakan. Berikut ini contoh-contoh penulisan *delay* yang ada pada menu **Help CodeVisionAVR** :

```
void main(void)
{
    /* disable interrupts */
    #asm("cli")

    /* 100ms delay */
    delay_us(100);

    /* 10ms delay */
    delay_ms(10);

    /* enable interrupts */
    #asm("sei")
}
```

3. PENGUJIAN DAN ANALISA

Pengujian robot dilakukan dengan mengambil beberapa data perbandingan yang akan dijadikan bahan analisa, antara lain :

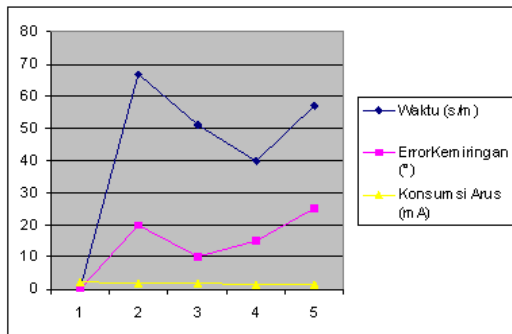
1. Kecepatan
2. Error kemiringan
3. Konsumsi arus

Berikut ini beberapa hasil pengujian

3.1 Jalan Rata (contoh : lantai kayu)

Tabel 1. Pengujian Robot di Jalan Rata

Coba Ke	Parameter Delay / Sudut	Waktu (s/m)	Error Kemiringan (°)	Konsumsi Arus (mA)
1	5 / 10°	-	-	2,1
2	10 / 30°	67	20	1,9
3	15 / 50°	51	10	1,7
4	20 / 70°	40	15	1,5
5	25 / 90°	57	25	1,6



Gambar 12.

Grafik perbandingan parameter di jalan rata

Analisa Waktu (Kecepatan):

- Waktu terkecil terdapat pada parameter 20/70°.
- Waktu terbesar terdapat pada parameter 5/10° (Pada 5/10° robot tidak berpindah tempat.)

Analisa Error Kemiringan:

- Error Kemiringan terkecil didapat pada parameter 15/50°.
- Error Kemiringan terbesar didapat pada parameter 25/90°.

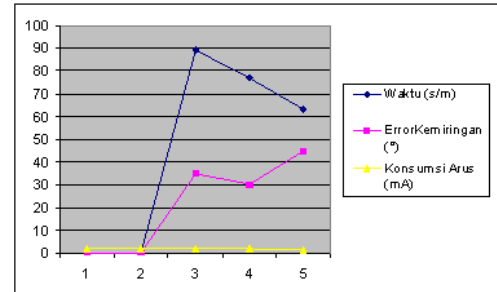
Analisa Konsumsi Arus :

- Konsumsi arus terkecil terdapat pada parameter 20/70°.
- Konsumsi arus terbesar terdapat pada parameter 5/10°

3.2 Jalan Tidak Rata (contoh : rumput)

Tabel 2. Pengujian Robot di Jalan Tidak Rata

Coba Ke	Parameter Delay / Sudut	Waktu (s/m)	Error Kemiringan (°)	Konsumsi Arus (mA)
1	5 / 10°	-	-	2,1
2	10 / 30°	-	-	1,8
3	15 / 50°	89	35	1,8
4	20 / 70°	77	30	1,7
5	25 / 90°	63	45	1,5



Gambar 13.

Grafik perbandingan parameter di jalan tidak rata

Analisa Waktu (Kecepatan):

- Waktu terkecil terdapat pada parameter 25/90°
- Waktu terbesar terdapat pada parameter 15/50° (Pada Parameter 5/10° & 10/30° robot tidak berpindah tempat.)

Analisa Error Kemiringan:

- Error Kemiringan terkecil didapat pada parameter 20/70°
- Error Kemiringan terbesar didapat pada parameter 25/90°

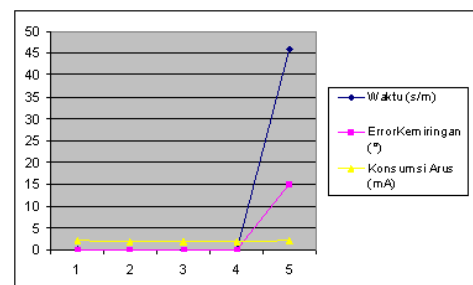
Analisa Konsumsi Arus :

- Konsumsi arus terkecil terdapat pada parameter 25/90°
- Konsumsi arus terbesar terdapat pada parameter 5/10°

3.3 Jalan Miring (contoh : bidang miring 10°)

Tabel 3. Pengujian Robot di Jalan Miring

Coba Ke	Parameter Delay / Sudut	Waktu (s/m)	Error Kemiringan (°)	Konsumsi Arus (mA)
1	5 / 10°	-	-	2,1
2	10 / 30°	-	-	2,0
3	15 / 50°	-	-	1,9
4	20 / 70°	-	-	1,9
5	25 / 90°	46	15	2,2



Gambar 14.

Grafik Perbandingan Parameter di Jalan Miring

Analisa Waktu (Kecepatan):

- Waktu yang dapat dicatat hanya terdapat pada parameter 25 / 90°.
(Pada parameter 5/10°,10/30°,15/50°,20/70° robot tidak berpindah tempat)

Analisa Error Kemiringan:

- Error kemiringan yang dapat dicatat terdapat pada parameter 25 / 90°.
(Pada parameter 5/10°,10/30°,15/50°,20/70° robot tidak berpindah tempat)

Analisa Konsumsi Arus :

- Konsumsi arus terkecil terdapat pada parameter 15/50° & 20 / 70°.
- Konsumsi arus terbesar terdapat pada parameter 25/90°.

4. KESIMPULAN

Setelah dilakukan pengujian dan analisisnya, dapat diambil beberapa kesimpulan antara lain :

1. Pada jalan rata, performa terbaik didapatkan jika robot berjalan pada **delay 20ms** dan **sudut 70 derajat** (tidak terlalu cepat)
2. Pada jalan tidak rata, performa terbaik didapatkan jika robot berjalan pada **delay 40ms** dan **sudut 80 derajat** (lambat)
3. Pada jalan miring, performa terbaik didapatkan jika robot berjalan pada **delay 50ms** dan **sudut 90 derajat** (sangat lambat)
4. Besarnya **kecepatan** pada jalannya robot disebabkan oleh Waktu tunda (delay)
(semakin besar waktu tunda, kecepatan berkurang)
5. Besarnya **error kemiringan** jalannya robot disebabkan oleh:
 1. Kecepatan gerak motor
(semakin cepat gerak motor, error semakin besar)
 2. Sudut sendi pangkal kaki
(semakin kecil sudut yang dibentuk, error semakin besar)
6. Besarnya **konsumsi arus** disebabkan oleh :
 1. Kecepatan gerak motor
(semakin cepat gerak motor, konsumsi arus semakin besar)
 2. Beban yang diterima oleh motor
(semakin beban yang diterima oleh motor, konsumsi arus semakin besar)

5. DAFTAR PUSTAKA

- [1] Wikipedia Indonesia (2006) : **Sejarah dan Penyebab Terjadinya Tsunami**, <http://id.wikipedia.org/wiki/Tsunami>.
- [2] CRASAR (Center for Robot Assisted Search & Rescue) (2002) : **History of Rescue Robot**, <http://crasar.csee.usf.edu>.
- [3] R.Siegwart & I. Nourbakhsh (2003) : **Autonomous Mobile Robot** : Chapter 2, The MIT Press. Massachusetts Institute of Technology Cambridge,Massachusetts.
- [4] Brook, Rodney. (1992) : **The Past and The Future Behavior Robotic**, MIT Press, Cambridge, Massachusetts.
- [5] Sutton, R. and Barto, A. (1998). **Reinforcement Learning: An Introduction**. MIT Press, Cambridge, MA.
- [6] Bekey, George A. (2005) : **Autonomous Robot** , MIT Press, Cambridge, Massachusetts.