

---

**USULAN TUGAS AKHIR**

**1. IDENTITAS PENGUSUL**

**NAMA** : Arya Cipta Graha Sinaga  
**NRP** : 5109100006  
**DOSEN WALI** : Isye Ariesianti, S.Kom, M. Phil.

**2. JUDUL TUGAS AKHIR**

**Rancang Bangun Pengendali Robot Kapal Perang Berbasis Suara.**  
*Design and Implementation of War Ship Controller Robot with Voice Recognition.*

**3. URAIAN SINGKAT**

Pertahanan dan keamanan adalah hal yang sangat penting bagi suatu negara, oleh karena itu diperlukan suatu strategi dalam menyusun formasi baik untuk menyerang maupun untuk bertahan. Bisa dibayangkan jika dalam menyusun strategi, alat atau kapal yang digunakan adalah kapal sungguhan, tentunya akan memakan waktu dan biaya, belum lagi kerahasiaan formasi yang dibangun.

Pada saat ini simulasi yang ada sudah menggunakan robot kapal perang, dan proses pengendaliannya menggunakan komputer yang terhubung dengan *server* melalui kabel RJ45, kemudian data dari *server* akan diteruskan ke *transmitter* melalui kabel USB untuk dikirim ke *client*/kapal. Setelah dikirim, melalui *transmitter*, data akan diterima melalui *receiver* dan digunakan untuk menggerakkan motor-motor yang ada pada robot.

Melihat pada kasus di atas, robot simulasi yang akan dibuat akan lebih mudah dioperasikan karena dikendalikan menggunakan teknologi pengenalan suara, data masukkan oleh pengguna adalah suara, dengan menggunakan perangkat *mobile* berbasis android dan perangkat lunak pengenalan suara, input suara dari pengguna akan diterjemahkan ke dalam bentuk teks, setelah itu data dalam bentuk teks akan dikirimkan ke *server* menggunakan Bluetooth, disisi *server* data akan diproses untuk menentukan kapal mana yang akan dikendalikan, setelah itu data akan diteruskan ke *transmitter*, kemudian *transmitter* akan mem-broadcast pesan, yang kemudian *receiver* akan menangkap data tersebut yang kemudian data akan diproses untuk menggerakkan motor-motor sesuai dengan program yang telah ditanam pada robot. Sehingga proses pengendalian dan pembelajaran robot akan lebih mudah serta dapat mempersingkat waktu.

**4. PENDAHULUAN**

**4.1. LATAR BELAKANG**

Perkembangan robot dari tahun ke tahun berkembang dengan pesat, mulai dari bentuk, fungsi, teknologi, sampai cara penggunaannya. Cara mengontrol robot jaman sekarang tidak lagi terbatas dengan menggunakan alat pengontrol, namun bisa juga dengan suara, bahkan dengan pikiran. Sehingga cara komunikasi manusia dan robot menjadi semakin mudah [1].

Penggunaan robot dewasa ini banyak digunakan untuk mempermudah pekerjaan manusia. Robot seringkali menjadi alternatif pengganti karena lebih murah dan aman, contohnya : dalam hal kesehatan, robot digunakan untuk membantu proses operasi, dalam hal bencana, robot digunakan untuk memadamkan kebakaran, dalam hal terorisme, robot digunakan untuk menjinakkan bom, dalam hal eksplorasi sebuah robot digunakan sebagai penjelajah ke luar angkasa. Kemudian bagaimana dengan bidang pertahanan dan keamanan? Dalam bidang pertahanan dan keamanan sebuah robot dapat digunakan sebagai sarana simulasi perang, guna membangun strategi untuk menyerang maupun bertahan. Robot dipilih sebagai sarana simulasi perang karena jauh lebih murah dibandingkan dengan alat sebenarnya. Bisa dibayangkan harga sebuah kapal perang asli, jika dibandingkan dengan robot tiruan kapal perang. Tentunya perbedaan harga akan sangat terlihat, selain itu juga sebuah robot akan lebih hemat tempat jika sebuah organisasi hendak melakukan praktek, belum lagi masalah kerahasiaan saat melakukan simulasi.

Oleh karena itu, penulis mencoba membuat sebuah robot yang bisa membantu proses simulasi yang dilengkapi dengan teknologi pengenalan suara pada perangkat *mobile* berbasis Android untuk semakin mempermudah pengguna mengendalikan robotnya.

## **4.2. RUMUSAN MASALAH**

Rumusan masalah yang diangkat dalam Tugas Akhir ini dapat dipaparkan sebagai berikut:

1. Bagaimana proses pengiriman perintah dari perangkat *mobile* ke *server*?
2. Bagaimana mekanisme pengiriman data dari pengendali ke robot melalui *transmitter*?
3. Bagaimana proses pembacaan perintah oleh robot?
4. Bagaimana perintah dapat menggerakkan motor-motor yang ada pada robot?

## **4.3. BATASAN MASALAH**

Permasalahan yang dibahas dalam tugas akhir ini memiliki beberapa batasan, di antaranya sebagai berikut:

1. Perintah suara sudah ditentukan dari awal.
2. Pengenalan suara menggunakan perangkat *mobile* berbasis Android.
3. *Transmitter* yang digunakan adalah Easy Radio (ER400).
4. Aplikasi menggunakan bahasa java dan C/C++.

## **4.4. TUJUAN DAN MANFAAT TUGAS AKHIR**

Tujuan dari Tugas Akhir ini adalah sebagai berikut:

1. Membantu proses simulasi kapal perang.
2. Mempermudah pengguna untuk mengendalikan kapal.

Manfaat tugas akhir ini adalah untuk membantu suatu organisasi untuk menentukan formasi kapal perang yang strategis saat menyerang maupun bertahan, dengan menggunakan robot yang lebih mudah dikendalikan serta lebih murah dan lebih aman jika dibandingkan dengan kapal perang yang sebenarnya.

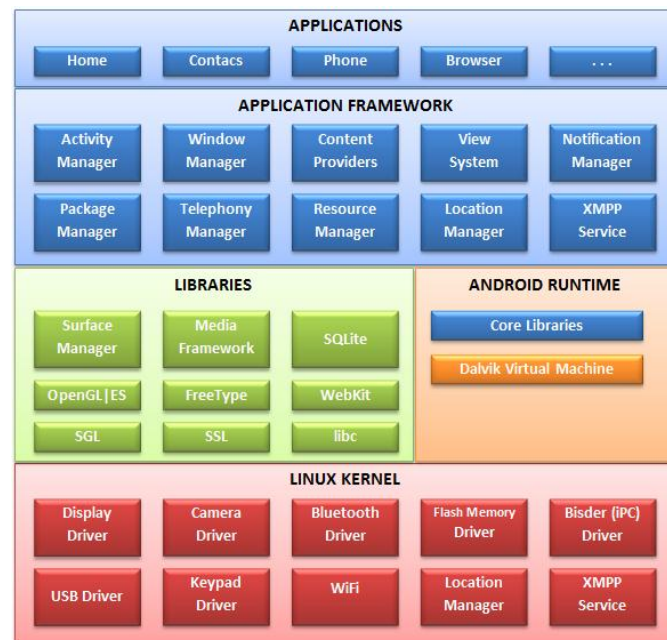
## **5. TINJAUAN PUSTAKA**

### **5.1. Android**

Merupakan sistem operasi *open source* yang dikembangkan oleh Google untuk perangkat *mobile*. Aplikasi Android dapat dilakukan pengembangan melalui Android SDK (*Software*

*Development Kit*) dengan menggunakan bahasa Java dan Google menyediakan banyak pustaka untuk memanjakan pengembang mengembangkan program pada Android.

Perangkat Android sendiri terdiri dari sistem operasi, *middleware*, dan *key application*. Pengembangan perangkat lunak Android menggunakan bahasa pemrograman Java dan aplikasi Android tidak berjalan langsung diatas *kernel* sistem operasi namun aplikasi tersebut berjalan diatas Dalvik, *virtual machine* yang khusus untuk perangkat *mobile* [2].



**Gambar 1. Lapisan Sistem Perangkat Lunak Android**

Pada Gambar 1 digambarkan lapisan perangkat lunak Android, dimana pada masing-masing lapisan mempunyai fungsi, yaitu :

1. Lapisan *Application*.  
Lapisan *Application* adalah suatu lapisan dimana aplikasi tersebut berjalan seperti *email client*, kalender, peta, dll.
2. Lapisan *Framework*.  
Lapisan *Framework* adalah lapisan yang berisikan *framework API* yang digunakan oleh pengembang.
3. Lapisan *Libraries*.  
Berisi pustaka yang digunakan oleh berbagai komponen dalam sistem Android.
4. Lapisan *Runtime*.  
Berisi berbagai inti pustaka yang menyediakan sebagian besar fungsionalitas yang serupa dengan API pada pemrograman Java.
5. Lapisan Linux Kernel.  
Merupakan lapisan inti dari sistem operasi Android. Pada lapisan ini terdapat file-file sistem yang mengatur proses sistem, memori, *resource*, *drivers*, dan sistem Android lainnya [3].

## 5.2. Android SDK

Android SDK (*Software Development Kit*) adalah paket yang digunakan untuk membangun aplikasi Android. Paket tersebut berisi *debugger*, emulator, *library*, dokumentasi, contoh *source code*, dan tutorial untuk sistem operasi Android [4].

### 5.3. Speech Input API

Speech Input API merupakan sebuah layanan disediakan oleh Google yang membantu aplikasi untuk menerima gelombang suara yang terdengar dari sekitar perangkat dan menerjemahkannya ke dalam sebuah teks.

Layanan ini sudah tersedia pada perangkat bergerak berbasis Android. Android yang bersifat *open platform* ini memudahkan bagi para pengembang aplikasi dalam mengakses Google Speech Recognizer.

Penggunaan API ini dilakukan dengan mengalirkan gelombang suara ke *server* audio Google. *Server* kemudian melakukan proses pengenalan suara yang dilanjutkan dengan pengiriman respon kepada klien berupa kemungkinan-kemungkinan kalimat yang paling mirip dengan suara masukan [5].

### 5.4. Bluetooth

Berdasarkan standar IEEE 802.15.1, Bluetooth termasuk teknologi standar nirkabel untuk pertukaran data jangka pendek, dengan menggunakan gelombang transmisi radio pada pita ISM (*industrial, scientific and medical radio bands*) 2400-2480MHz dengan membangun PAN (*personal area network*) dengan sekuritas tingkat tinggi. Pada awalnya dianggap sebagai nirkabel pengganti untuk kabel data RS-232, dan dapat menghubungkan beberapa perangkat dan mengatasi masalah sinkronisasi [6].

### 5.5. Easy Radio '02'

Easy Radio adalah perangkat keras yang menyediakan *transmitter, receiver* dan *transceiver*. Teknologi Easy Radio menyediakan performa tingkat tinggi yang mudah digunakan dan dapat melakukan proses pertukaran data hingga jarak 250m. Selain itu teknologi pada Easy Radio memungkinkan pengguna untuk mengatur frekuensi, *data rate* dan daya keluaran sesuai dengan keinginan pengguna. Modul ER tersedia dalam dua versi frekuensi: seri ER400 (433-434MHz) dan seri ER900 (869.85MHz & 902-928MHz).

Tabel 1. Easy Radio Configuration Command Set

Command	Function	ER400	ER900	Notes
<b>RS232 Communication Settings</b>				
ER_CMD#U0	Custom BAUD rate	300	300	Programmable via ER Windows Software (ii)
ER_CMD#U1	UART Data Rate	2400	2400	
ER_CMD#U2		4800	4800	
ER_CMD#U3		9600	9600	
ER_CMD#U4		19200	19200	
ER_CMD#U5		38400	38400	
ER_CMD#U?	Get UART Value			The module replies echos with the UART value. Eg: ER_CMD#U2 No ACK is required.

ER_CMD#H1	Handshaking ON	OFF	OFF	Only effects RTS Pin.
ER_CMD#H2	Handshaking OFF			
ER_CMD#A70	PARITY DISABLE	DISABLED BY DEFAULT When enabled data = 1 Start, 8 Data, 1 Parity, 1 Stop		
ER_CMD#A71	EVEN PARITY			
ER_CMD#A72	ODD PARITY			
ER_CMD#I7	FAST ACK Enable	OFF	OFF	(Upper case i) See notes on “FAST ACK” below.
ER_CMD#I8	FAST ACK Disable			
RF POWER Settings				
		ER400Series	ER400Series	
ER_CMD#P0	RF Power Output Sets output power on a channel.  Warning! This level will be set to the default setting when the frequency is changed or reset via a Channel command.	1mW	0.0625mW (TS)	0.0625mW (TRS)
ER_CMD#P1		2mW	0.125mW (TS)	0.125mW (TRS)
ER_CMD#P2		3mW	0.25mW (TS)	0.25mW (TRS)
ER_CMD#P3		4mW	0.5mW (TS)	0.5mW (TRS)
ER_CMD#P4		5mW	1.2mW (TS)	1mW (TRS)
ER_CMD#P5		6mW	1.5mW (TS)	1.2mW (TRS)
ER_CMD#P6		7mW	2mW (TS)	1.5mW (TRS)
ER_CMD#P7		8mW	3.1mW (TS)	2mW (TRS)
ER_CMD#P8		9mW	4mW (TS)	2.5mW (TRS)
ER_CMD#P9		10mW	5mW (TS)	3.1mW (TRS)
ER_CMD#P?	Get Power Value			The module replies with the power value. eg: ER_CMD#P9 No ACK is required.
ER_CMD#p0	Set <b>Default</b> RF Power Output. This allows the host to set each channel to a different default power setting. (Ideal for automatic selection between bands like 869 & 914 etc.)	1mW	0.0625mW	NOTE lower case ‘p’
ER_CMD#p1		2mW	0.125mW	
ER_CMD#p2		3mW	0.25mW	
ER_CMD#p3		4mW	0.5mW	
ER_CMD#p4		5mW	0mW	
ER_CMD#p5		6mW	1mW	
ER_CMD#p6		7mW	2mW	
ER_CMD#p7		8mW	3mW	
ER_CMD#p8		9mW	4mW	

ER_CMD#p9		10mW	5mW	
<b>RF Channel Settings</b>				
ER_CMD#C0	Channel 0	433.23 MHz	869.9MHz	All channels can now be chosen in software. Custom frequencies can only be set using software available from LPRS.
ER_CMD#C1	Channel 1	433.30 MHz	914.65MHz	
ER_CMD#C2	Channel 2	433.45 MHz	Not Set	
ER_CMD#C3	Channel 3	433.55 MHz	Not Set	
ER_CMD#C4	Channel 4	433.68 MHz	869.85MHz	
ER_CMD#C5	Channel 5	433.83 MHz	Not Set	
ER_CMD#C6	Channel 6	433.88 MHz	Not Set	
ER_CMD#C7	Channel 7	434.00 MHz	Not Set	
ER_CMD#C8	Channel 8	434.15 MHz	Not Set	
ER_CMD#C9	Channel 9	434.35 MHz	Not Set	
ER_CMD#C?	Get Channel Value			The module replies echos with the power value. Eg: ER_CMD#C9 No ACK is required.
<b>MISCELLANEOUS COMMANDS</b>				
ER_CMD#R1	Reset to Default Settings	U4, P9, C7	U4, P9, C0	Factory Default
ER_CMD#L0	Sleep	This command shuts down the transceiver completely by halting the oscillator and dropping the microprocessor down to a slow clock speed. Toggle the RTS pin to wake up. (Allow start-up Time)		
ER_CMD#A00	DCS OFF (default)	Recommended ON for new designs where back compatibility to older devices is not required		
ER_CMD#A01	DCS ON			
ER_CMD#A10	Encryption OFF (default)	Encryption algorithm is created and owned solely by LPRS. It uses a 16-bit seed that can be set by the developer.		
ER_CMD#A11	Encryption ON			
ER_CMD#A20	CRC16 OFF (default)	The CRC16 routines are more efficient and secure than the old CRC8. For new applications it is recommended.		
ER_CMD#A21	CRC16 ON			
ER_CMD#A30	Repeater Mode OFF (default)	When enabled, this mode will simply echo EVERY message it hears. Therefore no more than 1 repeater should be used UNLESS specific rules are followed. An application note will be available shortly to explain this further.		
ER_CMD#A31	Repeater Mode ON			

ER_CMD#Fs	FREQUENCY SWAP 'd' = destination 's' = source	This feature allows you to transmit on a different channel frequency to the receive frequency. e.g. ER_CMD#F07 After reselecting channel 0, the module will listen on CH0 but transmit on CH7. To reset the module to its default, send the same source channel number to the destination: e.g. ER_CMD#F00 <b>Warning DCS should be disabled to use this feature.</b>		
TEST MODES				
ER_CMD#T0	Upper FSK Carrier			Test Mode 0
ER_CMD#T1	Modulated Carrier			Test Mode 1
ER_CMD#T2	Lower FSK Carrier			Test Mode 2
ER_CMD#T3	Get Firmware Revision			Returns Firmware String: eg ER400TRS-02V2.01.5
ER_CMD#T4	RAW Data Test			Out of CTS pin

Tabel 1 merupakan, contoh-contoh perintah yang dapat dilakukan oleh pengguna, aturan-aturan pada Tabel 1 dikirim melalui perangkat lunak ke mikrokontroler untuk menggerakkan robot. Baris yang berwarna abu-abu hanya berlaku pada Easy Radio '02' [7].

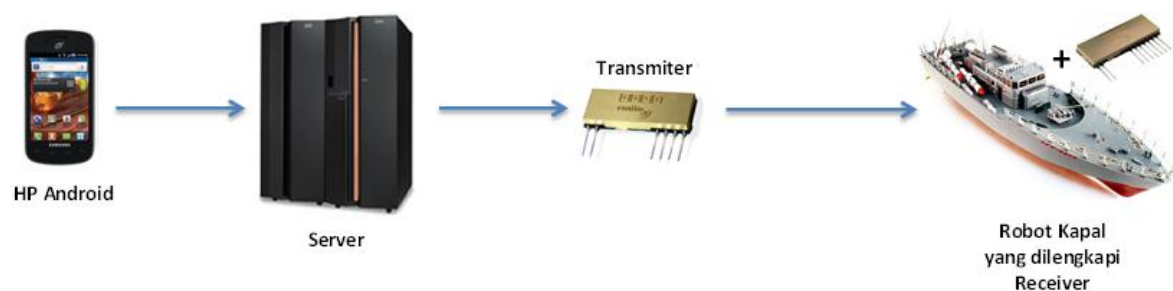
## 5.6. Mikrokontroler

Mikrokontroler adalah suatu alat elektronika digital yang mengerjakan instruksi-instruksi yang diberikan kepadanya. Artinya, bagian terpenting dan utama dari suatu sistem terkomputerisasi adalah program itu sendiri yang dibuat oleh seorang *programmer*. Program ini menginstruksikan komputer untuk melakukan jalinan yang panjang dari aksi-aksi sederhana untuk melakukan tugas yang lebih kompleks yang diinginkan oleh pengembang [8].

## 6. METODOLOGI

Robot simulasi adalah robot yang digunakan sebagai sarana simulasi kapal perang guna mempersiapkan formasi kapal saat menyerang maupun bertahan dengan menggunakan teknologi pengenalan suara.

Gambar 2 adalah gambaran dari alur arsitektur sistem.



**Gambar 2. Arsitektur Sistem**

- a. Proses pengolahan data suara.  
Pada proses ini data masukan dari pengguna yang berupa suara akan diubah menjadi data teks dengan menggunakan Google Speech API, yang kemudian data akan di kirimkan ke *server* dengan menggunakan Bluetooth.
- b. Proses penerjemahan data pada *server*.  
Pada proses ini, data teks dari perangkat *mobile* akan diubah sesuai dengan standar data Easy Radio, yang kemudian data akan di kirimkan ke transmitter dengan menggunakan kabel USB.
- c. Proses pengiriman data oleh *transmitter*.  
Pada proses ini data akan dikirimkan ke *receiver* (ER400RS-02) melalui *transmitter* (ER400TS-02), melalui gelombang radio.
- d. Proses penggerakkan robot.  
Pada proses ini data yang diterima oleh *receiver* (ER400RS-02) akan digunakan untuk menggerakkan motor-motor pada robot, sesuai dengan program yang ada pada mikrokontroler.

## 7. JADWAL KEGIATAN

**Tabel 2. Tahap Pengerjaan Tugas Akhir**

Tahapan	Maret				April				Mei				Juni			
Analisa Kebutuhan dan Studi Literatur																
Perancangan Sistem																
Implementasi																
Uji Coba dan Evaluasi																
Penyusunan Buku																

## 8. Daftar Pustaka

- [1] Tomasz Kubik and Masanori Sugisaka, *SICE 2001. Proceedings of the 40th SICE Annual Conference. International Session Papers*, pp. 106 - 111, 2001.
- [2] N Safaat, *Android : Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Android*, Pertama ed. Bandung, Indonesia: Informatika, 2011.
- [3] Herdi Naufal. (2012, September) Mengenal Arsitektur Android OS - TWOH's Engineering. [Online]. <http://www.twoh.web.id/2012/09/mengenal-arsitektur-sistem-operasi-android/>
- [4] Cory Janssen. What is Android SDK? - Definition from Technopedia. [Online]. <http://www.techopedia.com/definition/4220/android-sdk>
- [5] Alex Gruenstein. (2010) Speech Input API For Android. [Online]. [http://static.googleusercontent.com/external\\_content/untrusted\\_dlcp/www.google.co.jp/ja/jp/events/developerday/2010/tokyo/pdf/tt1-gruenstein.pdf](http://static.googleusercontent.com/external_content/untrusted_dlcp/www.google.co.jp/ja/jp/events/developerday/2010/tokyo/pdf/tt1-gruenstein.pdf)



- [6] Bluetooth Radio Interface, Modulation & Channel. [Online]. <http://www.radio-electronics.com/info/wireless/bluetooth/radio-interface-modulation.php>
- [7] (2005, Maret) LPRS Data Sheet. [Online]. <http://www.robot-electronics.co.uk/datasheets/ER400-900TRS-v2.pdf>
- [8] Hermawan Sutanto. (1998, Januari) Konsep Mikrokontroler. [Online]. <http://mikrokontroler.tripod.com/6805/bab1.htm>