

# Rancang Bangun Mobil Remote Control Android dengan Arduino

## *Prototype of Android RC Car by Arduino*

Andi Widiyanto\*<sup>1</sup>, Nuryanto<sup>2</sup>

<sup>1,2</sup>Teknik Informatika Universitas Muhammadiyah Magelang  
E-mail: \*[indypink77@gmail.com](mailto:indypink77@gmail.com), <sup>2</sup>[nuryantoummgl@yahoo.com](mailto:nuryantoummgl@yahoo.com)

### **Abstrak**

*Peralatan elektronik hampir tidak dapat lepas dari kehidupan manusia untuk meningkatkan kemudahan dan kenyamanan dalam pemenuhan kebutuhannya. Aktifitas sehari-hari banyak dilakukan melalui smartphone sebagai alat yang hampir selalu dalam genggam. Saat ini banyak alat elektronik yang dikendalikan hanya dengan menekan tombol remote. Perkembangan teknologi microcontroller seperti Arduino dapat diintegrasikan dengan alat yang lain, bukan hanya dengan robot saja. Penelitian ini membuat smartphone terhubung dengan mikrokontroler arduino yang digunakan sebagai pengendali alat elektronik. Obyek yang digunakan adalah mobil remote control mainan (RC Car). Metode penelitian yang dilakukan adalah studi literatur dan experiment. Arduino diprogram menggunakan bahasa C untuk menjalankan motor supaya bergerak sesuai dengan data masukan yang dikirimkan melalui Android. Penelitian ini menghasilkan prototype model sebuah alat elektronik yang dikendalikan dengan smartphone Android melalui koneksi bluetooth. Hasil pengujian menunjukkan bahwa kecepatan pairing Android-Arduino lebih dipengaruhi versi android dibandingkan hardware yang terpasang. Jarak antara Arduino-Android tidak berpengaruh besar terhadap kecepatan proses pairing.*

**Kata Kunci** — Smartphone, Microcontroller, Arduino, Bluetooth, RC Car

### **Abstract**

*Electronic equipment can hardly be separated from human life is being used to improve the ease and comfort in the fulfillment of their needs. Daily activities mostly done via smartphone as a device that is nearly always in hand. Today, many electronic devices are controlled simply by pressing the remote button. Technological developments such as the Arduino microcontroller can be integrated with other tools, not only with robots. The objective of this research is to make a smartphone connected to Arduino microcontroller as the electronic appliance controller. Object that was used is a car remote control toys (RC Car). The research method conducted was literature study and experiment. Arduino was programmed using C language to run the motor to move according to the input data that was sent through the android. This research resulted in prototype model of an electronic device that is controlled by android smartphone via a Bluetooth connection. Test results showed that the speed of pairing Android-Arduino was influenced more by the android version than the installed hardware. The distance between the Arduino-Android did not greatly affect the speed of the pairing process.*

**Keywords** — Smartphone, Microcontroller, Arduino, Bluetooth, RC Car.

---

## 1. PENDAHULUAN

Android adalah sistem operasi pada telepon seluler cerdas (*smartphone*) yang dikembangkan oleh *Google*. Banyak sensor dan alat yang ditanamkan pada *smartphone*, sehingga Android difungsikan menjadi banyak alat bantu.

Untuk mencari informasi pengguna tinggal mengatakan sebuah kata pada *smartphone*, *Google voice* akan menterjemahkan dan melakukan pencarian untuk menghasilkan informasi yang ditemukan. Bahkan *smartphone* dapat digunakan sebagai asisten pribadi misalnya aplikasi *Irish* yang dapat digunakan untuk mencari dan membacakan informasi terbaru, baca sms, email dan sebagainya.

GPS yang terdapat di *smartphone* dapat digunakan sebagai penunjuk arah kiblat, kompas, alat bantu saat tersesat atau mencari alamat dan sebagainya. Peralatan yang seperti kamera dapat digunakan sebagai *IP Camera* dengan memasang aplikasi pada *smartphone* selain digunakan sebagai fungsi kamera. *Infrared* dan *bluetooth* dapat digunakan sebagai *remote* peralatan rumah tangga [1] dan masih banyak lagi fitur yang dapat digunakan pada *smartphone android*.

Perkembangan teknologi mikrokontroler memberikan kemudahan dalam pemrogramannya seperti *arduino*. *Arduino* merupakan *design system* minimum mikrokontroler dengan modul mikrokontroler AVR, sehingga dapat digunakan untuk membangun sistem elektronika berukuran minimalis namun handal dan cepat [2].

Mikrokontroler dapat dihubungkan dengan alat elektronik bahkan dengan peralatan yang besar dan voltase besar melalui relay dengan mekanisme tertentu. Mobil mainan atau helikopter mainan sebagai gambarannya. Cukup menekan tombol tertentu maka mainan akan berjalan maju mundur, belok kanan kiri, melaju kencang atau pelan dan sebagainya.

*Smartphone* memiliki beberapa sensor dan perangkat yang ditanamkan oleh pabrik. Jika alat ini dapat berkomunikasi dengan mikrokontroler maka memungkinkan *smartphone* ini digunakan sebagai *controller* untuk alat-alat elektronik yang lain. Sehingga peralatan lain yang masih manual/mekanik dapat dikembangkan dalam bentuk elektronik dan dikontrol menggunakan *smartphone*.

Rumusan masalah penelitian ini adalah bagaimana membuat *smartphone* terhubung dengan mikrokontroler *Arduino* sebagai pengendali alat elektronik. Penelitian ini akan menghasilkan sebuah *prototype* model alat pengendali peralatan elektronik melalui *Arduino* dan *android smartphone*.

*Arduino* adalah *physical computing* atau *single-board microcontroller* yang bersifat *open source*. *Arduino* dirancang untuk memudahkan penggunaan elektromekanik dalam berbagai kegiatan. *Microcontroller* yang digunakan pada *Arduino* berjenis *atmel AVR* dengan berbagai jenis lainnya. *Software Arduino IDE* dapat dijalankan pada sistem operasi *Windows*, *Macintosh OSx* dan *Linux*[3]. *Arduino* adalah sebuah produk *design system* minimum mikrokontroler yang di buka secara bebas. *Arduino* menggunakan bahasa pemrograman C yang telah dimodifikasi dan sudah ditanamkan *programmer bootloader* yang berfungsi untuk menjembatani antara *software compiler arduino* dengan mikrokontroler [4].

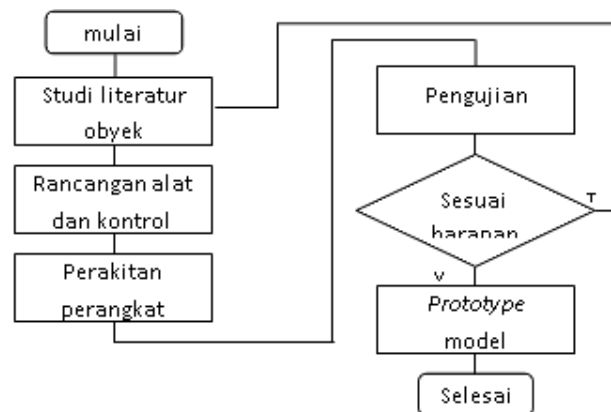
Kelebihan *Arduino* diantaranya adalah tidak perlu perangkat *chip programmer* karena didalamnya sudah ada *bootloader* yang akan menangani *upload* program dari komputer, *Arduino* sudah memiliki sarana komunikasi USB, sehingga pengguna laptop yang tidak memiliki *port serial/RS323* bisa menggunakannya [5].

Hasil penelitian Robot *Monomonic Fire Fighting* [6] menjadi model obyek penelitian, akan tetapi dengan obyek dan model yang diubah. Robot mobil pemadam api *Four Wheel Drive (4WD)* diganti dengan mobil RC mainan, walaupun sama-sama menggunakan 4 buah roda akan tetapi cara kerjanya berbeda. *4WD* menggunakan 4 buah motor DC, sedangkan mobil RC menggunakan 2 buah motor DC. Kontrol gerakan menggunakan android melalui *bluetooth* seperti penelitian saklar listrik menggunakan Android [7] akan tetapi perbedaannya adalah sistem kontrol dalam penelitian ini kontrolnya bersifat langsung dan terus-menerus bukan sekedar saklar berfungsi untuk *on* atau *off* saja.

## 2. METODE PENELITIAN

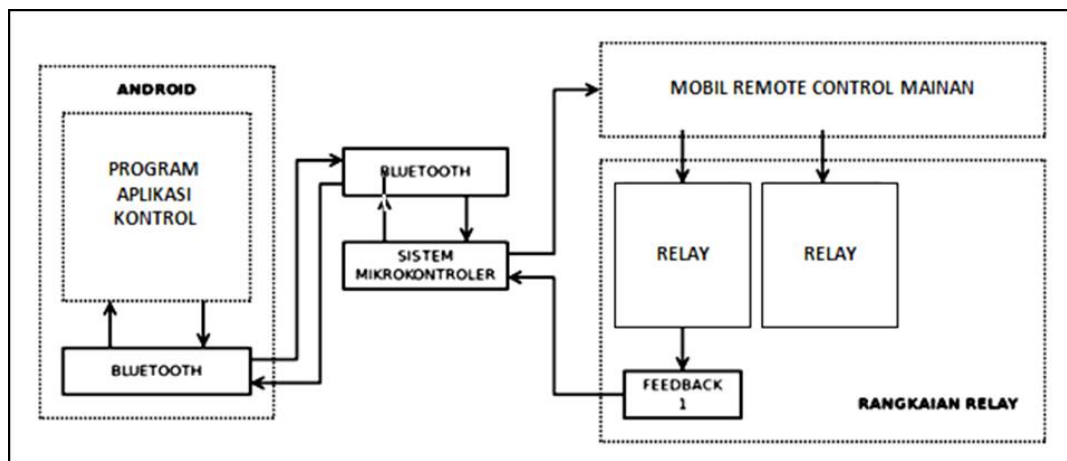
Pengendalian perangkat elektronik menggunakan *remote* sudah lazim saat ini. Makin banyak perangkat yang digunakan semakin banyak pula *remote* yang disiapkan. Alat elektronik yang tidak pernah jauh dengan kita adalah *smartphone*. Dengan teknologi yang ada maka *smartphone* contohnya android dapat digunakan untuk menggantikan fungsi *remote*. Apabila android dapat digunakan untuk mengontrol mikrokontroler maka dapat digunakan pula untuk mengendalikan perangkat elektronik yang terhubung. Seluruh alat elektronik dirumah seperti TV, radio, AC, mesin cuci, lampu dan sebagainya dapat dikendalikan melalui sebuah *smartphone*.

Metode penelitian yang dilakukan studi literatur dan eksperimen laboratorium. Penelitian yang dilakukan akan terfokus pada komunikasi antara perangkat *smartphone* dengan mikrokontroler Arduino melalui *Bluetooth* untuk mengendalikan perangkat elektronik seperti pada *flowchart* Gambar 1:



Gambar 1. Sistematika penelitian

Alat yang menjadi obyek penelitian adalah sebuah mobil *Remote Control* (RC) yang selama ini dikontrol melalui *remote* dengan gelombang radio, akan dikontrol menggunakan Android melalui *bluetooth*. Rangkaian elektronik akan digantikan dengan Arduino yang akan mengontrol gerakan motor DC. Kecepatan putaran dan arah putaran motor dikendalikan oleh arduino. Hubungan antara Android dengan mobil RC dirancang dengan skema peralatan seperti pada Gambar 2.



Gambar 2. Skema Rancangan Alat Penelitian

Komunikasi antara Android dengan dengan mobil RC melalui koneksi *bluetooth*. Proses awal adalah *pairing* untuk menghubungkan Android dengan Arduino pada mobil RC dengan otentikasi *password* untuk pertama kali supaya hanya *smartphone* tertentu saja yang dapat digunakan. Setelah *pairing* berhasil, Arduino siap untuk menerima perintah dari Android melalui *bluetooth*. Arduino akan menerima data dari Android dan menterjemahkannya untuk menggerakkan motor DC, sehingga mobil RC dapat dikendalikan dengan Android dari jarak jauh. Proses kendali akan putus jika komunikasi dengan Android putus baik diputus atau karena diluar jangkauan.

Mobil RC memiliki 2 motor DC dengan fungsi yang berbeda yaitu motor penggerak/belakang untuk menggerakkan mobil maju-mundur dan motor *steering*/depan untuk belok kanan-kiri. Ukuran motor depan biasanya lebih kecil dibanding yang belakang karena fungsi putarannya hanya untuk memutar roda depan untuk belok kanan atau belok kiri. Motor belakang mempunyai tugas yang lebih berat yaitu menggerakkan roda belakang untuk maju dan mundur.

Pada saat Android mengirimkan data, mikrokontroler akan memprosesnya kemudian menggerakkan motor melalui relay. Konsep kendali seperti mobil yang sebenarnya yaitu untuk menggerakkan maju atau mundur dengan menginjak pedal gas dan stir diputar ke kanan atau kekiri. Penerapannya pedal gas menggerakkan motor belakang dan stir menggerakkan motor depan. Rancangan gerakan motor berdasarkan data kode dari android seperti pada Tabel 1.

Tabel 1. Tabel Rancangan Gerakan Motor Berdasarkan Kode Kiriman

| Kode | Motor          |                | Keterangan      |
|------|----------------|----------------|-----------------|
|      | Depan          | Belakang       |                 |
| F    | Mati           | Putar ke kanan | Bergerak maju   |
| B    | Mati           | Putar ke kiri  | Bergerak mundur |
| L    | Putar ke kiri  | Idle           | Belok kiri      |
| R    | Putar ke kanan | Idle           | Belok kanan     |

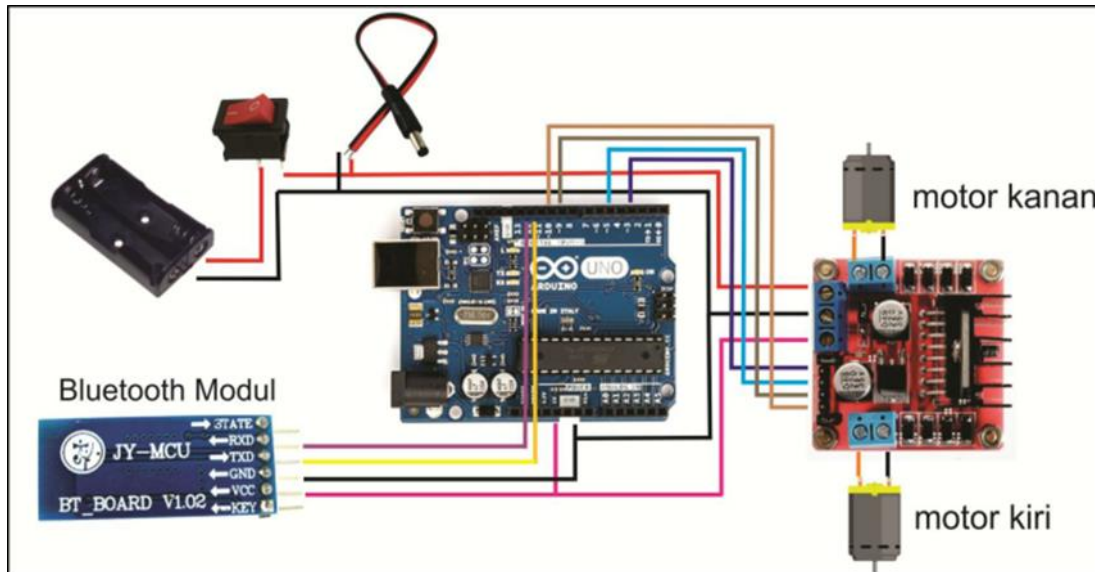
### 3. HASIL DAN PEMBAHASAN

#### 3.1. Implementasi Perakitan Alat

Implementasi penelitian memerlukan hardware yang akan dirakit dan software yang digunakan untuk memprogram mikrokontroler. Software yang digunakan adalah Arduino IDE versi 1.6.3, sedangkan hardware yang digunakan sebagai berikut:

- 1) Smartphone Android
- 2) Mikrokontroler Arduino Uno R3
- 3) HC-05 bluetooth
- 4) L298N driver modul/relay
- 5) Project board mini 400 point
- 6) Mobil RC mainan bekas

Skema perakitan peralatan mengacu pada panduan *Arduino Bluetooth RC* ([www.jogjarobotika.com](http://www.jogjarobotika.com)) untuk perakitan mobil RC 2WD dengan penggerak 2 motor di belakang (Gambar 3) hanya saja perubahan pada perakitan motor yang disesuaikan dengan mobil RC. Kode perintah android diterima melalui *HC-05bluetooth*, kemudian Arduino Uno R3 akan menggerakkan motor melalui L298N. Project board mini digunakan sebagai sambungan kabel antar alat supaya mudah untuk lepas dan pasang tanpa harus menggunakan *solder*.



Gambar 3. Skema Perakitan Peralatan

Arduino Uno terhubung dengan bluetooth HC-05 dan L298N driver modul melalui pin digital. Untuk bagian input Pin Tx dan Rx bluetooth HC-05 terhubung dengan pin digital 12 dan 13. Untuk mengontrol gerakan motor, L298N driver modul terhubung melalui pin digital 3, 5, 9 dan 10.

Data input dari HC-05 supaya dapat digunakan untuk mengontrol motor maka mikrokontroler diprogram dengan bahasa C pada Arduino IDE versi 1.6.3. Komunikasi data antara HC-05 dengan Arduino uno menggunakan serial sehingga membutuhkan *library unitSoftwareSerial.h* supaya perintahnya dapat dimengerti *compiler* bahasa C seperti Gambar 4.

```
#Include <softwareserial.h>
SoftwareSerial mySerial(12, 13); // RX, TX
#define m1 3
#define m2 5
#define m3 9
#define m4 10
Void setup ()
{
  pinMode (m1, OUTPUT);
  pinMode (m2, OUTPUT);
  pinMode (m3, OUTPUT);
  pinMode (m4, OUTPUT);
  mySerial.begin(9600);
}
```

Gambar 4. Source Code Setting pin Komunikasi Arduino

Untuk mengontrol gerakan implementasi dari rancangan Tabel 1 dibuat sebuah fungsi motorOut seperti Gambar 5, untuk mempermudah pemanggilan program. Fungsi motorOut membutuhkan parameter yaitu kecepatan motor depan (lpwm) dan belakang(rpwm) serta arah putaran motor (arrow). Perubahan arah putaran motor dengan mengubah arus kutup positif (+) dan negatif (-) pada motor, misalnya saat kabel merah dialiri arus positif dan kabel hitam dialiri arus negatif motor berputar maju, dengan mengubah arus negatif pada kabel merah dan arus positif pada kabel hitam maka motor akan berputar mundur. Kecepatan motor disetting melalui android dimana kecepatan dalam skala 0 – 255 yang akan diterjemahkan dalam bentuk kuat arus yang akan dialirkan ke motor.

```
Void motorOut(unsigned char lpwm, unsigned char rpwm, boolean
arrow) {
//arrow=false=maju; arrow=true=mundur;
//ar=0=stop; ar=1=kanan      ar=2=kiri
If (arrow==false)
{
    Digitalwrite (m3, HIGH);
    Digitalwrite (m1, LOW);
    Analogwrite (m4, 255-lpwm);
    Analogwrite (m2, rpwm);
}
Else
{
    Digitalwrite (m3, LOW);
    Digitalwrite (m1, HIGH);
    Analogwrite (m4, lpwm);
    Analogwrite (m2, 255-rpwm);
}
}
```

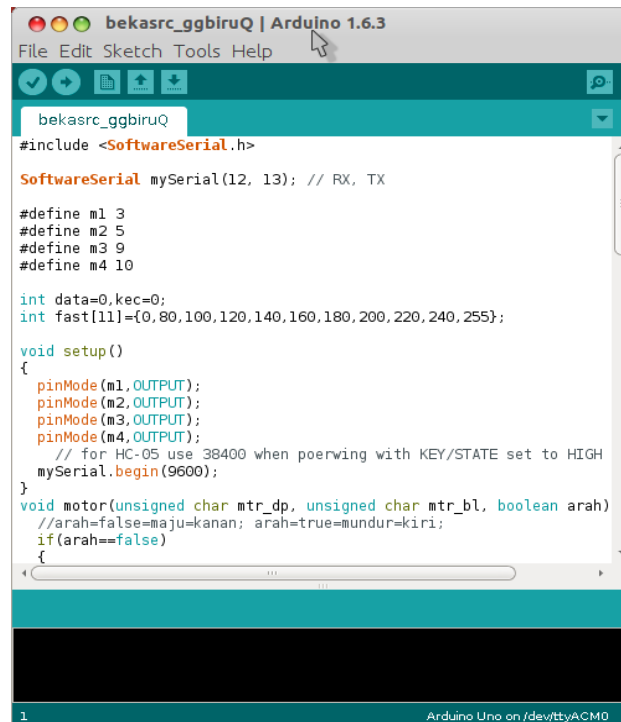
Gambar 5. Source Code Fungsi untuk Menggerakkan Motor

Pergerakan mobil RC dikendalikan melalui kode-kode yang dikirimkan Android. Untuk menggerakkan maju maka motor depan mati dan motor belakang berputan ke kanan/maju, sehingga pemanggilan fungsi menggunakan *parameter* lpwm=0, rpwm=kecepatan dan arrow=true. Untuk gerakan mundur hanya mengubah isi *parameter* arrow menjadi false. Penerapan lengkap *Source code* program untuk mengatur gerakan seperti Gambar 6.

```
//maju
If (data=='F' )
{
    motorOut (0, fast [kec], true);
    delay (2);
}
//mundur
If (data=='B' )
{
    Motorout (0, fast [kec], false);
    Delay (2);
}
//kanan
If (data=='R' )
{
    motorOut (fast [kec], 0, true);
    delay (2)
}
//kiri
If (data=='L' )
{motorOut (fast [kec], 0, false);
    Delay (2)
}
```

Gambar 6. Source Code Pengontrol Gerakan

Program diketik menggunakan Arduino IDE versi 1.6.3. pada laptop lalu dihubungkan dengan Arduino uno R3 melalui kabel USB. Laptop yang digunakan pada penelitian ini menggunakan *Operating System linux Ubuntu 12.04 LTS*, sehingga *port* komunikasi otomatis terdeteksi dengan tepat baik port maupun jenis mikrokontrolernya lalu diupload ke mikrokontroler seperti Gambar 7.

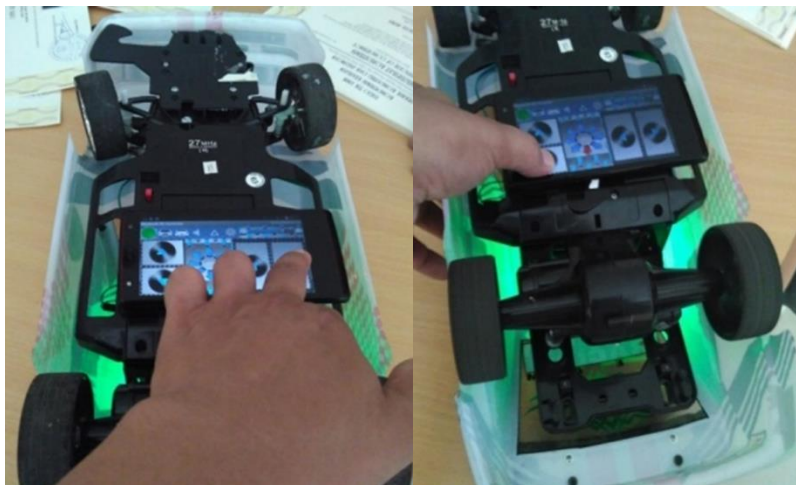


```
File Edit Sketch Tools Help
bekasrc_ggbiruQ | Arduino 1.6.3
File Edit Sketch Tools Help
bekasrc_ggbiruQ
#include <SoftwareSerial.h>
SoftwareSerial mySerial(12, 13); // RX, TX
#define m1 3
#define m2 5
#define m3 9
#define m4 10
int data=0, kec=0;
int fast[11]={0,80,100,120,140,160,180,200,220,240,255};
void setup()
{
  pinMode(m1, OUTPUT);
  pinMode(m2, OUTPUT);
  pinMode(m3, OUTPUT);
  pinMode(m4, OUTPUT);
  // for HC-05 use 38400 when poerwing with KEY/STATE set to HIGH
  mySerial.begin(9600);
}
void motor(unsigned char mtr_dp, unsigned char mtr_bl, boolean arah)
//arah=false=maju=kanan; arah=true=mundur=kiri;
if(arah==false)
{
```

Gambar 7. Gambar Arduino 1.6.3

### 3.2. Pengujian

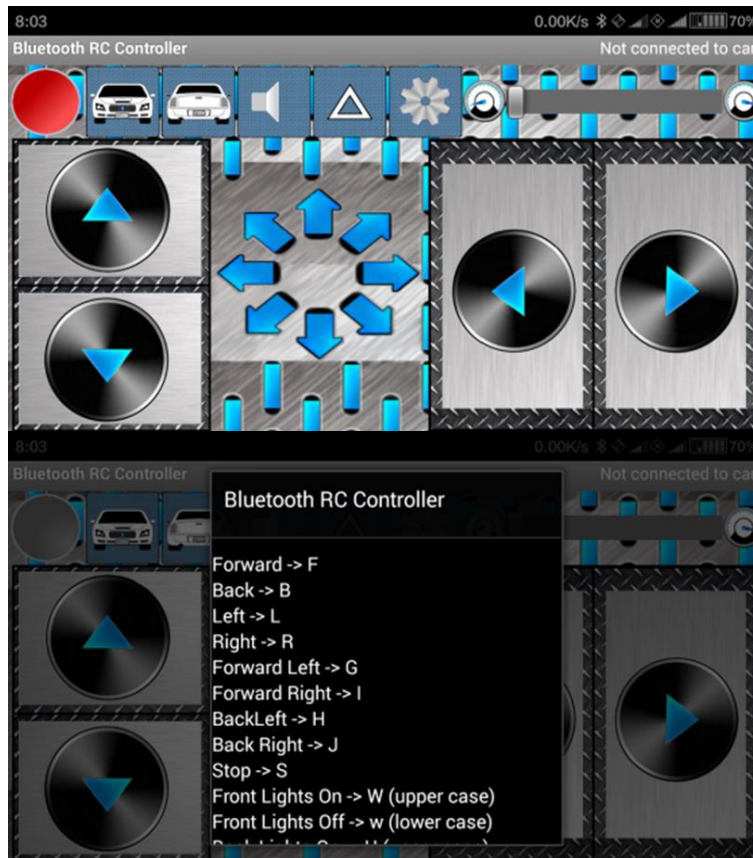
Untuk membuktikan bahwa android dapat digunakan untuk mengendalikan mobil RC, pada Android diinstall program *Bluetooth RC Controller* dari *Google Play*. Semua Android versi *gingerbread* ke atas dapat digunakan dan berjalan dengan baik untuk mengontrol mobil RC seperti Gambar 8. Android yang telah terpasang aplikasi Bluetooth RC Controller, setelah berhasil *pairing* maka saat tombol ditekan, mobil RC akan merespon seperti yang diinginkan seperti belok kanan-kiri, maju-mundur. Android versi 2.1 (*Eclair*) tidak dapat diinstall aplikasinya sehingga tidak dapat digunakan untuk pengujian.



Gambar 8. Pengujian Kontrol Mobil RC

Pengujian pertama adalah pengujian kontrol Arduino terhadap mobil RC menggunakan beberapa perangkat Android dengan spesifikasi yang berbeda dengan menggunakan aplikasi


Bluetooth RC Controller 1.6.yang diinstall dari *Google Play*. Pada info aplikasi terdapat keterangan kode yang dikirimkan saat terjadi penekanan tombol kontrol seperti Gambar 9.




Gambar 9.Screenshot Aplikasi Bluetooth RC Controller

Spesifikasi smartphone yang disampaikan vendor tidak persis sama dengan kenyataannya misalnya Redmi 2 CPU quad core 1,2 GHz RAM 1GB setelah diuji CPU 1,21 GHz dan RAM 898 MB. Jika diuji dengan aplikasi yang lain seperti antutu, cell phone info dan sebagainya hasilnya juga berbeda, sehingga aplikasi yang digunakan untuk mengetahui informasi spesifikasi *smartphone* menggunakan aplikasi CPU-Z for android versi 1.10.dari *Google Play* seperti Gambar 10.





| SOC   |  | SYSTEM   |  | BATTERY                    |  | THERMAL                                 |  |
|---|--|--|--|----------------------------|--|---|--|
|  |  | <b>Qualcomm Snapdragon 400/410</b><br>1,21 GHz |  | Model                      |  | 2014811 (2014811)                       |  |
| Cores   |  | 4  |  | Manufacturer               |  | Xiaomi                                  |  |
| Architecture  |  | 4x ARM Cortex-A53 @ 1,21 GHz                   |  | Board                      |  | msm8916                                 |  |
| Revision  |  | r0p0   |  | Bootloader                 |  | unknown                                 |  |
| Process   |  | 28 nm  |  | Build ID                   |  | KTU84P                                  |  |
| Clock Speed   |  | 200 MHz - 1,21 GHz                             |  | Hardware                   |  | qcom                                    |  |
| CPU 0   |  | 800 MHz  |  | Android Version            |  | 4.4.4                                   |  |
| CPU 1   |  | stopped  |  | API Level                  |  | 19                                      |  |
| CPU 2   |  | stopped  |  | Java VM                    |  | Dalvik 1.6.0                            |  |
| CPU 3   |  | stopped  |  | OpenGL ES Version          |  | 3.0                                     |  |
| GPU Vendor  |  | Qualcomm                                       |  | Kernel Architecture        |  | armv7l                                  |  |
| GPU Renderer  |  | Adreno (TM) 306 @ 400 MHz                      |  | Kernel Version             |  | 3.10.28-g093959c<br>(V16.2.8.0.KHJCNBK) |  |
| GPU Clock Speed   |  | 310 MHz  |  | Screen Size                |  | 5,11 inches                             |  |
| CPU Load  |  | 29 %   |  | Screen Resolution (pixels) |  | 720 x 1280                              |  |
| Scaling Governor  |  | interactive                                    |  | Screen Density             |  | 320 dpi                                 |  |
|   |  |  |  | Total RAM                  |  | 898 MB                                  |  |
|   |  |  |  | Available RAM              |  | 181 MB (20%)                            |  |
|   |  |  |  | Internal Storage           |  | 5.25 GB                                 |  |

Gambar 9. Screenshot Aplikasi Bluetooth RC Controller

Perangkat yang digunakan pada pengujian diambil secara acak untuk mewakili berbagai variasi hardware seperti *single core*, *dual core* hingga *quad core*. Untuk versi Android paling rendah adalah *gingerbread* karena aplikasi hanya dapat berjalan pada versi tersebut ke atas, dan yang paling tinggi adalah versi *kitkat*. Versi *smartphone* terbaru saat ini adalah *lollipop* pada saat penelitian ini dilakukan masih jarang ditemukan, sehingga *smartphone* yang digunakan dan spesifikasi yang berasal dari CPU-Z yang digunakan untuk pengujian seperti pada Tabel 2.

Tabel 2. Smartphone yang Diuji

| HP | TYPE             | CPU  | RAM    | OS              |
|----|------------------|--|--------|-----------------|
| A  | SAMSUNG GALAXY 5 | Qualcomm MSM7 600 MHz<br>(1x ARM 1136)<br>Clock Speed 600 MHz              | 183 MB | 2.3.3<br>API 10 |
| B  | TABULET TS-201   | Snapdragon 200 1,01 GHz<br>(2 x ARM Cortex-A5)<br>Clock Speed 1008 MHz     | 390 MB | 4.0.4<br>API 15 |
| C  | EVERCOSS A12     | MediaTek MT6572 1,00 GHz<br>(2 x ARM Cortex-A7)<br>Clock Speed 1001 MHz    | 220 MB | 4.2.2<br>API 17 |
| D  | REDMI 1S         | Snapdragon 400 1,59 GHz<br>(4 x ARM Cortex-A7)<br>Clock Speed 1593 MHz     | 871 MB | 4.3<br>API 18   |
| E  | REDMI 2          | Snapdragon 400/410 1,21 GHz<br>(4 x ARM Cortex-A53)<br>Clock Speed 800 MHz | 898 MB | 4.4.4<br>API 19 |

Hasil pengamatan semua perangkat android dapat digunakan untuk mengendalikan mobil RC dengan baik selama terhubung dengan Arduino. Tidak ada perbedaan yang dirasakan antara smartphone satu dengan yang lain saat digunakan untuk mengontrol pergerakan mobil RC sehingga pengujian antar smartphone sebagai alat kontrol mobil RC tidak dibandingkan. Perbedaan yang dirasakan adalah waktu yang dibutuhkan dalam *pairing* dengan Arduino.

Pengujian selanjutnya dilakukan untuk mengetahui waktu yang dibutuhkan dalam proses *pairing*, dilakukan dengan mengubah jarak antara Android dengan mobil RC. Masing-masing *smartphone* dilakukan proses *pairing* beberapa kali dan dicatat waktunya dengan *stopwath*, kemudian dihitung rata-rata waktu koneksinya, hasilnya pada Tabel 3.

Tabel 3. Pengujian Pairing Android-Arduino

| HP | AVERAGE PAIRING TIME |         |         |         |         |         |
|----|----------------------|---------|---------|---------|---------|---------|
|    | <1M                  | 1M      | 2M      | 3M      | 4M      | 5M      |
| A  | 0:04:48              | 0:05:04 | 0:05:04 | 0:04:58 | 0:05:10 | 0:05:16 |
| B  | 0:02:30              | 0:02:28 | 0:03:15 | 0:03:34 | 0:03:28 | 0:03:25 |
| C  | 0:02:18              | 0:02:14 | 0:03:11 | 0:03:15 | 0:03:18 | 0:02:58 |
| D  | 0:02:07              | 0:01:49 | 0:02:55 | 0:02:22 | 0:02:08 | 0:02:13 |
| E  | 0:01:34              | 0:01:43 | 0:01:56 | 0:01:51 | 0:01:51 | 0:01:46 |

Hasil pengamatan pengujian menunjukkan bahwa jarak antara Android dengan mobil RC (Arduino) tidak mempengaruhi rata-rata waktu *pairing* untuk semua *smartphone*. Semakin jauh jarak antara *Android-Arduino* tidak ada perangkat yang menunjukkan pola makin besar atau makin kecil rata-rata waktu *pairing*.

*Smartphone* yang diuji terdiri dari *single core*, *dual core* dan *quad core* dengan ukuran RAM yang bervariasi. Semakin tinggi *hardware* semakin cepat pula proses *pairing time*. Smartphone dengan CPU *single core* (Galaxy 5) kalah jauh dengan *smartphone dual core* maupun *quad core*, akan tetapi tidak berlaku untuk spesifikasi *hardware* yang hampir sebanding. Evercross A-12 bahkan beberapa kali pengujian lebih cepat dibanding TS-201 yang memiliki *clock speed* CPU dan RAM yang lebih tinggi. Redmi 2 yang memiliki *clock speed* lebih rendah dari Redmi 1S juga lebih cepat rata-rata *pairing time* nya.

Jika Tabel 3 dihitung lagi rata-rata per smartphone dari rata-rata waktu *pairing* per jarak pada Galaxy 5 sebagai berikut ini:

$$\begin{aligned} \text{Rata-rata} &= \frac{\text{avg}T_{<1m} + \text{avg}T_{1m} + \text{avg}T_{2m} + \text{avg}T_{3m} + \text{avg}T_{4m} + \text{avg}T_{5m}}{6} \\ &= \frac{0:04:48 + 0:05:04 + 0:05:04 + 0:04:58 + 0:05:10 + 0:05:16}{6} \\ &= 0:05:03 \end{aligned}$$

Perhitungan yang sama dilakukan pada *smartphone* yang lainnya hasilnya sebagai berikut:

|                |           |
|----------------|-----------|
| GALAXY 5       | : 0:05:03 |
| TABULET TS-201 | : 0:03:06 |
| EVERCOSS A12   | : 0:02:52 |
| REDMI 1S       | : 0:02:16 |
| REDMI 2        | : 0:01:47 |

Hasil perhitungan tersebut menunjukkan urutan versi android, Galaxy 5 paling rendah versi OS yaitu 2.3.3 dan Redmi 2 versi OS 4.4.4 yang paling tinggi. Data hasil perhitungan rata-rata menunjukkan pola bahwa semakin tinggi versi Android maka waktu *pairing* semakin kecil atau semakin cepat.

Dalam proses pengontrolan mobil RC ada hal yang berbeda dengan motor DC yang digunakan untuk robot 4WD maupun 2WD. Pada saat mobil RC dikirim sinyal untuk maju terus menerus kecepatan motor akan semakin cepat walaupun sinyal sudah dihentikan mobil RC akan tetap melaju sehingga tabrakan tidak bisa dihindarkan. Perlu dibahas dan dikaji lebih jauh yang menyebabkan hal ini yaitu karakteristik motor DC atau mekanik mobil RC yang berbeda dengan motor DC.

#### 4. KESIMPULAN

Penelitian ini menghasilkan *prototype* model sebuah alat elektronik (mobil RC) yang dikendalikan dengan smartphone Android dan ditarik kesimpulan sebagai berikut:

- 1) Mobil RC Arduino dapat dikendalikan dengan baik menggunakan *smartphone* Android versi minimal *Gingerbread* (2.3.x) melalui koneksi bluetooth.
- 2) Jarak antara Arduino dengan Android tidak berpengaruh besar dengan kecepatan proses *pairing*.
- 3) Kecepatan proses *pairing* lebih dipengaruhi oleh versi Android dibanding dengan *hardware* yang terpasang.

#### 5. SARAN

Perlu penelitian lebih lanjut dengan menambahkan sensor jarak untuk menghindari tabrakan saat melaju dengan kecepatan tinggi dan pengaruh kecepatan dan jarak untuk menentukan pembelokan yang tepat. Alat penelitian dikembangkan dengan alat yang lain dan media transmisi data menggunakan Wi-Fi atau menggunakan jaringan internet sehingga jangkauan menjadi lebih jauh.

#### 6. UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada Fakultas Teknik Universitas Muhammadiyah Magelang yang telah memberi dukungan financial dan fasilitas terhadap penelitian ini.

#### DAFTAR PUSTAKA

- [1] Rahmiyati, P., Firdau, G., Fathorrahman, N., 2014, Implementasi Sistem Bluetooth Menggunakan Android dan Arduino untuk Kendali Peralatan Elektronik, *Jurnal ELKOMIKA*, No.1, Vol. 2, Hal 1-14.
  - [2] Masinambow, V., Najoan, M.E.I, Lumenta, A.S.M., 2014, Pengendali Saklar Listrik Melalui Ponsel Pintar Android, <http://ejournal.unsrat.ac.id/index.php/elekdankom/article/view/3772>, diakses 7 April 2015.
  - [3] Yudhistira, A.F., 2014, Rancang Bangun Alat Bantu Parkir Mobil Menggunakan Sensor Jarak Ultrasonik Berbasis Arduino Uno, *Jurnal Tugas Akhir*, STT Telkom, Purwokerto.
  - [4] Masinambow V., Najoan, M.E.I., Lumenta, A.S.M., 2014, Pengendali Saklar Listrik Melalui Ponsel Pintar Android, *e-journal Teknik Elektro dan Komputer*, <http://ejournal.unsrat.ac.id/index.php/elekdankom/article/download/3772/3295>, diakses tanggal 7 April 2015.
  - [5] Silvia, A.F, Haritman, E., Muladi, Y., 2014, Rancang Bangun Akses Kontrol Pintu Gerbang Berbasis Arduino Dan Android, *Jurnal ELECTRANS*, Vol.13, No.1, Hal 1-10.
-

- 
- [6] Rahman, A., 2007, Monomonic Fire Fighting Mobile Robot Dengan Sistem Gerak Four Wheel Drive, *Jurnal Algoritma*, Vol. 3, No 1, Hal 17-26.
- [7] Rahmiati, P., Firdaus, G., Fathorrahman, N., 2014, Implementasi Sistem Bluetooth menggunakan Android dan Arduino untuk Kendali Peralatan Elektronik, *Jurnal ELKOMIKA*, No.1, Vol. 2, Hal 1-14.
-