

## **BAB IV**

### **HASIL DAN UJI COBA**

Pada bab ini akan membahas hasil pengujian sistem, mulai dari pengujian permodul hingga pengujian sistem secara keseluruhan serta *monitoring* unjuk kerja dari sistem secara satu-persatu dan keseluruhan. Pengujian tersebut akan dilakukan dengan urutan sebagai berikut :

1. Pengujian rangkaian *adaptor*.
2. Pengujian papan Arduino dan GSM *Shield*.
3. Pengujian performa inisialisasi sistem.
4. Pengujian performa eksekusi perintah via SMS.
5. Pengujian aplikasi pengontrol sistem.

#### **IV.1. Pengujian Rangkaian *Adaptor***

Rangkaian adaptor sebagai pemasok utama daya untuk operasional sistem memiliki peranan penting. Oleh karena itu, rangkaian *adaptor* harus dapat bekerja sesuai dengan yang diharapkan. Untuk mengetahui apakah rangkaian *adaptor* bekerja dengan baik atau tidak dapat diketahui dengan menggunakan *volt meter*. Pada perancangan sistem ini terdapat dua *adaptor* dengan nilai voltase *output* masing-masing sebesar 9 *Volt DC* dan 8 *Volt DC*. Hasil pengujian rangkaian *adaptor* dapat dilihat pada tabel IV.1.

**Tabel IV.1 : Data Hasil Pengukuran Rangkaian Adaptor**

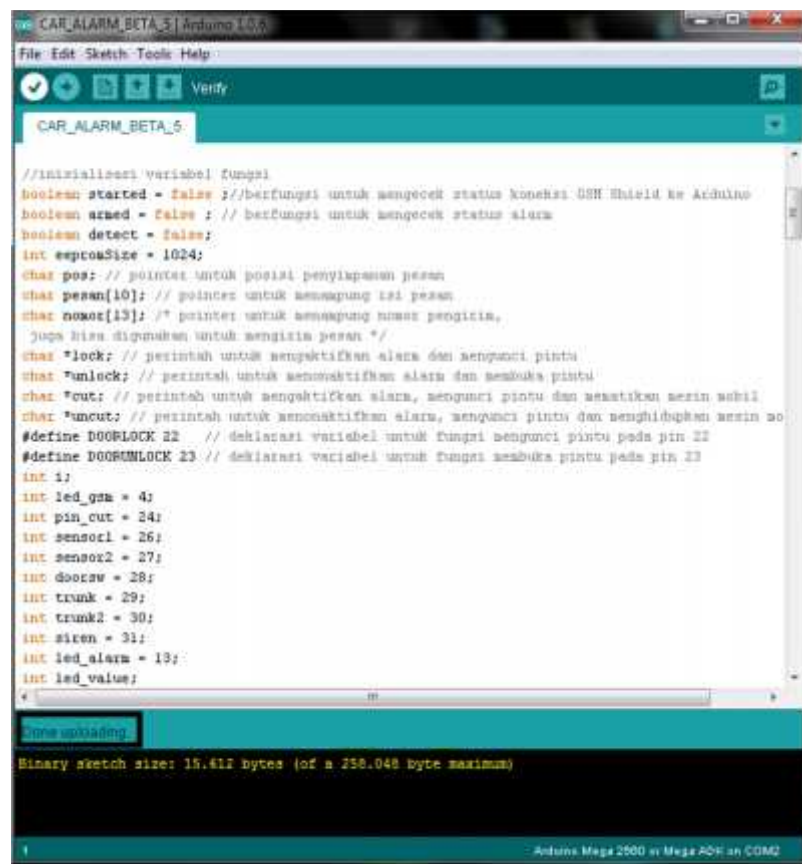
<b>Adaptor</b>	<b>Pengujian I</b>	<b>Pengujian II</b>	<b>Pengujian III</b>
<i>Input</i> = 12 V DC	12 V DC	12 V DC	12 V DC
<i>Output</i> = 9 V DC	9,01 V DC	9 V DC	9 V DC
<i>Input</i> = 12 V DC	12 V DC	12 V DC	12 V DC
<i>Output</i> = 8 V DC	7,57 V DC	7,56 V DC	7,56 V DC

Berdasarkan data pada tabel IV.1, hasil pengukuran pada rangkaian adaptor 9 V dan 8 V yang dilakukan dengan *volt meter* memiliki nilai hasil *output* yang sesuai untuk kebutuhan operasional sistem. Karena adaptor 9 V digunakan untuk menghidupkan sistem minimum Arduino dan *GSM Shield*, yang dimana rangkaian minimum sistem membutuhkan voltase antara 7-12 Volt untuk bekerja secara normal. Sedangkan rangkaian adaptor 8 Volt digunakan untuk menghidupkan modul *relay*, karena modul *relay* membutuhkan tegangan sekitar 6-8 Volt. Jadi, rangkaian adaptor tersebut sudah dikatakan bekerja dengan baik dan sesuai dengan yang diharapkan.

#### **IV.2. Pengujian Papan Arduino dan *GSM Shield***

Arduino dan *GSM Shield* sebagai alat utama kontrol sistem pastinya harus dapat bekerja dengan baik sesuai dengan kode program. Maka dari pada itu, penulis melakukan ujicoba terhadap papan Arduino dan *GSM Shield*. Pengujian ini dapat dilakukan dengan meng-*upload* kode program ke papan Arduino, serta menumpuk papan *GSM Shield* diatas Arduino. Kemudian, memasang lampu LED

sebagai indikator bahwa *GSM Shield* terkoneksi dengan Arduino. Kode program diketik pada program Arduino IDE dengan menggunakan bahasa “C++”. Kemudian setelah kode program selesai, tinggal menghubungkan papan Arduino dan *GSM Shield* ke komputer melalui kabel USB, kemudian melakukan proses *upload sketch* (mengunggah kode program ke papan Arduino). Hasil dari pengujian tersebut dapat dilihat pada gambar IV.1 dan IV.2.



```

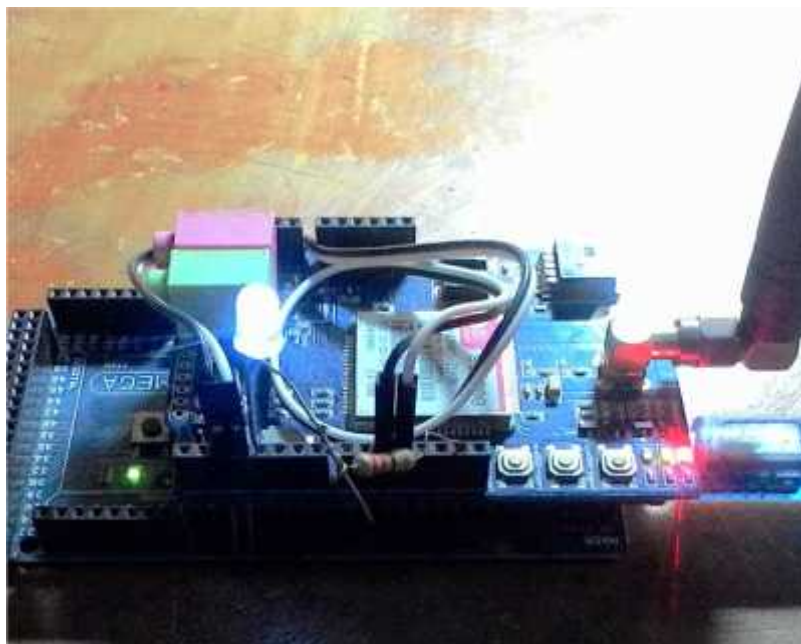
CAR_ALARM_BETA_5 | Arduino IDE
File Edit Sketch Tools Help
Verify
CAR_ALARM_BETA_5

//inisialisasi variabel fungsi
boolean started = false ;//berfungsi untuk mengecek status koneksi USB Shield ke Arduino
boolean armed = false ; // berfungsi untuk mengecek status alarm
boolean detect = false;
int espromSize = 1024;
char pos; // pointer untuk posisi penyisipan pesan
char pesan[10]; // pointer untuk menampung isi pesan
char nomor[13]; /* pointer untuk menampung nomor pengirim,
juga bisa digunakan untuk mengisi pesan */
char *lock; // perintah untuk mengaktifkan alarm dan membuka pintu
char *unlock; // perintah untuk menonaktifkan alarm dan membuka pintu
char *cut; // perintah untuk mengaktifkan alarm, mengunci pintu dan mematikan mesin mobil
char *uncut; // perintah untuk menonaktifkan alarm, mengunci pintu dan menghidupkan mesin mobil
#define DOORLOCK 22 // deklarasi variabel untuk fungsi mengunci pintu pada pin 22
#define DOORUNLOCK 23 // deklarasi variabel untuk fungsi membuka pintu pada pin 23
int i;
int led_gsm = 4;
int pin_cut = 24;
int sensori = 26;
int sensor2 = 27;
int dooraw = 28;
int trunk = 29;
int trunk2 = 30;
int siren = 31;
int led_alarm = 13;
int led_value;

Time uploading
Binary sketch size: 15.612 bytes (of a 258.048 byte maximum)
Arduino Mega 2560 w/ Mega ADK on COM2

```

**Gambar IV.1 : Proses Upload berhasil**



**Gambar IV.2 : Hasil Upload ke Papan Arduino**

Pada gambar IV.2, dapat terlihat lampu indikator LED berwarna putih menyala, ini menandakan bahwa *GSM Shield* dapat terkoneksi ke Arduino dengan benar, dengan kata lain papan Arduino dan *GSM Shield* siap digunakan.

#### **IV.2. Pengujian Performa Inisialisasi Sistem**

Sebelum sistem dapat digunakan sepenuhnya, sistem harus di inisialisasi terlebih dahulu untuk memastikan bahwa modul utama kontrol sistem bekerja dengan benar dan dapat menerima perintah yang dikirimkan lewat SMS. Inisialisasi juga dapat digunakan sebagai indikator bahwa *GSM Shield* dapat terhubung dengan Arduino. Hasil dari ujicoba inisialisasi sistem dapat dilihat pada tabel IV.2.

**Tabel IV.2 : Data Hasil Pengujian Performa Inisialisasi Sistem**

<b>Parameter</b>	<b>Pengujian I</b>	<b>Pengujian II</b>	<b>Pengujian III</b>
Inisialisasi Arduino	1,08 detik	0,89 detik	1,04 detik
GSM <i>Shield</i> terkoneksi ke jaringan GSM	13,78 detik	13,73 detik	16,17 detik
GSM <i>Shield</i> terkoneksi ke Arduino	16,49 detik	16,46 detik	17,61 detik
<b>Total waktu</b>	<b>31,35 detik</b>	<b>31,08 detik</b>	<b>34,82 detik</b>

Berdasarkan data pada tabel IV.2, dapat diketahui waktu rata-rata inisialisasi sistem dengan menggunakan rumus perhitungan, yaitu :

$$\frac{31,35 + 31,08 + 34,82}{3} = 32 \text{ detik}$$

Maka, sistem membutuhkan waktu rata-rata sekitar 32 detik untuk inisialisasi awal sistem sebelum bisa menerima perintah yang dikirimkan melalui SMS. Berdasarkan fakta tersebut, dapat dikatakan bahwa penyalaan sistem stabil berdasarkan waktu yang dibutuhkan.

#### **IV.4. Pengujian Performa Eksekusi Perintah SMS**

Bagian yang penting dari sistem ini adalah eksekusi perintah yang dikirimkan melalui SMS, karena operasional sistem bergantung kepada SMS perintah yang masuk. Untuk mengetahui tingkat unjuk kerja sistem dalam mengeksekusi perintah SMS yang masuk, maka perlu dilakukan pengujian performa. Dalam pengujian ini, penulis menggunakan parameter jarak dan waktu

dalam pengujian performa sistem dalam mengeksekusi perintah SMS. Hasil dari ujicoba dapat dilihat pada tabel IV.3.

**Tabel IV.3 : Tabel Hasil Pengujian Performa Eksekusi Perintah SMS**

Item pengujian	Jarak Pengiriman Perintah			Nilai rata-rata
	10 meter	20 meter	30 meter	
Fungsi <i>Lock</i>	8,32 detik	23 detik	10 detik	13 detik
Fungsi <i>Unlock</i>	10 detik	11 detik	10 detik	10,33 detik
Fungsi <i>Cut</i>	12 detik	11 detik	10 detik	11 detik
Fungsi <i>Uncut</i>	12 detik	10 detik	9,79 detik	10 detik
Fungsi <i>Alert!</i>	9 detik	10 detik	10 detik	9,6 detik
<b>Total</b>	<b>51,32 detik</b>	<b>65 detik</b>	<b>49,79 detik</b>	<b>10,7 detik</b>

Berdasarkan data pada tabel IV.3, dapat diketahui total rata-rata waktu pemrosesan sebesar 10,7 detik. Berdasarkan hasil perhitungan tersebut, maka dapat dikatakan bahwa sistem membutuhkan waktu rata-rata sebesar 10,7 detik untuk menerima dan mengeksekusi perintah. Maka hasil kesimpulannya adalah bahwa sistem yang dirancang cukup stabil untuk menerima dan mengeksekusi perintah. Walaupun pada dasarnya nilai tersebut dapat bersifat fluktuatif karena tergantung kepada ketersediaan jaringan GSM operator tertentu. Namun secara keseluruhan, sistem yang dirancang cukup layak untuk digunakan.

## **IV.5. Pengujian Aplikasi Pengontrol Sistem**

Aplikasi untuk pengontrol sistem alarm dirancang dengan menggunakan software *AppInventor 2*, software untuk sistem alarm ini diberi nama CAR ALARM. Software ini berguna untuk mengontrol sistem alarm secara penuh menggunakan tampilan dan *command button*, sehingga pengguna tidak perlu berulang kali mengetikkan SMS untuk mengontrol sistem alarm. Aplikasi ini dapat digunakan pada *smartphone* dengan sistem operasi *Android*. Sedangkan versi minimal sistem operasi *Android* yang dibutuhkan untuk menjalankan aplikasi ini adalah versi 2.3 (*Gingerbread*). Elemen tampilan serta bagian-bagian dari aplikasi akan dijelaskan pada bagian berikut ini :

### **IV.5.1. Tampilan *Login* Program**

Sebelum masuk ke tampilan utama program, dibutuhkan suatu otentifikasi untuk memastikan hanya pengguna yang dapat masuk ke program. Oleh karena itu, penulis merancang suatu mekanisme *login* untuk masuk ke program utama. Hasil pengujian menu login dapat dilihat pada tabel IV.4 :








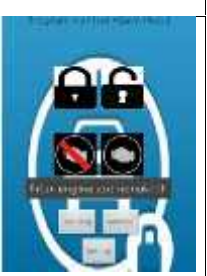
Tabel IV.4 : Tabel Pengujian Login Program

No	Skenario	Test Case	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
1	Memasukkan password yang salah		Muncul pesan peringatan bahwa password yang di input salah		Valid
2	Memasukkan password yang benar		Akan langsung masuk ke program utama		Valid

#### IV.5.2. Tampilan Program Utama

Setelah pengguna memasukkan *password* dengan benar, maka program akan menampilkan tampilan utama program. Di sinilah algoritma utama program berada, dimana bagian tampilan ini dapat mengontrol sistem alarm yang telah dirancang sebelumnya via SMS. Hasil pengujian program utama dapat dilihat pada tabel IV.5.

Tabel IV.5 : Tabel Pengujian Program Utama







No	Skenario	Test Case	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
1	Aktivasi fitur <i>Lock</i>		Muncul pesan bahwa sistem alarm aktif		Valid
2	Aktivasi fitur <i>Unlock</i>		Muncul pesan bahwa sistem alarm tidak aktif		Valid
3	Aktivasi fitur <i>Cut</i>		Muncul pesan bahwa fitur engine cut aktif		Valid
4	Aktivasi fitur <i>Uncut</i>		Muncul pesan bahwa fitur engine cut tidak aktif		Valid

5	Memicu sistem alarm		Akan muncul kotak dialog untuk memilih mengaktifkan /tidak mengaktifkan fitur <i>engine cut</i>		Valid
---	---------------------	---	---	---	-------

#### IV.5.3. Tampilan Program *Setting*

Program *setting* berisi pilihan pengaturan untuk nomor tujuan penerima SMS perintah serta pengaturan *password* untuk *login* ke program utama. Untuk menyimpan data nomor tujuan serta data *password*, penulis menggunakan database *TinyDB* yang ada didalam *AppInventor*. Hasil pengujian program *setting* dapat dilihat pada tabel IV.6.

Tabel IV.6 : Tabel Pengujian Program Setting

No	Skenario	Test Case	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
1	Menginput nomor tujuan		Jika klik tombol simpan, maka akan muncul pesan bahwa nomor tujuan telah disimpan		Valid
2	Mengganti password, namun dengan nilai yang berbeda pada kolom input password		Muncul peringatan bahwa nilai kedua <i>password</i> berbeda pada kolom inputan <i>password</i>		Valid
3	Mengganti password, dengan nilai yang sama		Muncul peringatan bahwa <i>password</i>		Valid

	pada kolom input password		telah tersimpan		
--	---------------------------------	--	--------------------	--	--

#### IV.6. Kelebihan dan Kekurangan

Pada perancangan alarm mobil berbasis mikrokontroler dan android ini masih berupa purwarupa (*prototype*), oleh karena itu masih kurang sempurna. Sistem ini memiliki beberapa kelebihan dan kekurangan, diantaranya :

##### a. Kelebihan

Adapun beberapa kelebihan yang dimiliki oleh sistem alarm mobil berbasis mikrokontroler dan android ini, antara lain :

- 1) Sistem ini dapat dikontrol dari jarak yang jauh, selama masih berada dalam jangkauan jaringan GSM. Ini melebihi kemampuan alarm mobil konvensional yang hanya dapat dioperasikan dengan jarak maksimal 10 meter.
- 2) Sistem ini dapat memberitahukan jika ada sesuatu yang memicu sistem alarm secara langsung kepada pengguna melalui *smartphone*. Ini juga melebihi kemampuan alarm mobil konvensional yang hanya memberi peringatan kepada pengguna hanya melalui suara sirine.
- 3) Sistem ini dapat dikontrol dengan menggunakan aplikasi untuk sistem operasi Android. Dengan kata lain, apabila pengguna memiliki *smartphone* dengan sistem operasi Android dapat menggunakan sistem ini dengan lebih mudah daripada harus mengetikkan SMS untuk mengirimkan perintah.

- 4) Sistem ini juga dapat mengontrol mesin mobil secara jarak jauh, jadi apabila ada orang yang tidak bertanggung jawab yang membawa lari mobil pengguna yang terpasang dengan sistem ini, maka pengguna dapat secara langsung menonaktifkan mesin mobil dari jarak jauh. Ini sangat efektif untuk pencegahan kejahatan pencurian mobil.

**b. Kekurangan**

Meskipun memiliki beberapa kelebihan seperti yang telah dijelaskan sebelumnya, sistem ini juga tak luput dari beberapa kelemahan yang melekat padanya, antara lain :

- 1) Kelemahan utama pada sistem ini adalah bergantung pada ketersediaan jaringan GSM operator tertentu, apabila mobil berada pada daerah yang tidak terjangkau oleh jaringan GSM, maka alat tidak akan berfungsi secara optimal. Namun kelemahan ini masih dapat diatasi dengan menggunakan kunci manual mobil hanya untuk fungsi mengunci dan membuka pintu mobil, namun fungsi yang lain tidak dapat digunakan.
- 2) *Smartphone* Android memiliki energi baterai yang terbatas, jadi apabila baterai *smartphone* Android telah habis, maka alat tidak dapat dikontrol secara langsung. Namun kelemahan ini masih dapat diatasi dengan mengirimkan SMS secara manual dari *handphone* lain.
- 3) Karena menggunakan SMS, maka harus mengeluarkan biaya untuk operasional sistem alarm ini dengan menggunakan pulsa Prabayar.
- 4) Karena sistem ini menggunakan daya listrik yang berasal dari aki utama mobil, maka apabila aki utama mobil dalam kondisi yang tidak baik (rusak

atau *soak*), maka akan menyebabkan sistem tidak dapat berfungsi secara optimal.