

TEMPAT PENYIMPANAN DIES BERBASIS Arduino Mega 2560

Yumitra F Br Ginting¹, Rahmat Dinur²

^{1,2}Program Studi Teknik Elektro, Universitas Efarina

E-mail: ¹Yumitraginting@gmail.com

Abstrak

Dies merupakan komponen yang penting dalam suatu proses produksi. Penulis mengajukan rancangan rak untuk menyimpan dies secara aman. Rak otomatis ini menggunakan servo motor sebagai pengunci dan dikendalikan melalui keypad sebagai input PIC (Personal Input Code) dan kode dies. Alat ini dirancang dengan memanfaatkan Arduino Mega 2560 sebagai pengendali dan PC dengan menggunakan visual basic sebagai penampil aplikasi. Rak ini bekerja ketika ada perintah dari pengguna yang berupa masukan kode PIC melalui keypad. Keypad berfungsi sebagai kata kunci utama pada penggunaan alat ini jika PIC benar, maka Arduino akan memberikan input high pada relay untuk mengaktifkan servo motor. Dari hasil pengujian yang telah dilakukan dapat dibuktikan bahwa rak otomatis ini dapat bekerja dengan baik. Software yang digunakan Arduino Mega 2560 dan visual Basic dapat bekerja sama mendeteksi input keypad dengan baik. Fitur dalam penggunaan database dapat mencatat pengguna dies dengan baik. Dies hanya dapat diakses oleh pengguna dengan otorisasi tertentu, pengguna dies dan rentang waktu penggunaan tercatat dengan baik pada suatu database.

Keyword: Arduino Mega 2560, Visual Basic, Rak, PIC, Servo Motor.

PENDAHULUAN

Seiring majunya perkembangan jaman, inovasi peralatan baru banyak diciptakan guna mempermudah aktivitas manusia. Teknologi memegang peran penting di era modernisasi seperti saat ini, dimana teknologi menjadi bagian yang tidak dapat dipisahkan dari perkembangan kehidupan manusia. Sistem keamanan menjadi lebih baik melalui penggunaan teknologi. Dalam penelitian ini, rak adalah hal yang disorot dalam ruang lingkup sistem keamanan. Penulis mengajukan sistem keamanan laci (drawer) pada rak. Laci (drawer) menjadi dibuka dan ditutup secara otomatis melalui suatu program. Sistem ini juga dilengkapi dengan visual basic jika terjadi kesalahan prosedur penggunaan alat. Penggunaan password untuk akses masuk merupakan pokok permasalahan dari sistem keamanan ini. Untuk menanggulangi kebocoran password yang terjadi maka password tersebut dapat diganti sewaktu-waktu sesuai keinginan, sehingga kerahasiaannya dapat terus terjaga. Dengan adanya sistem keamanan ini pada rak diharapkan keamanan dapat terpantau lebih baik lagi dan dapat dipastikan hanya orang yang berhak saja yang dapat mengakses isi rak.

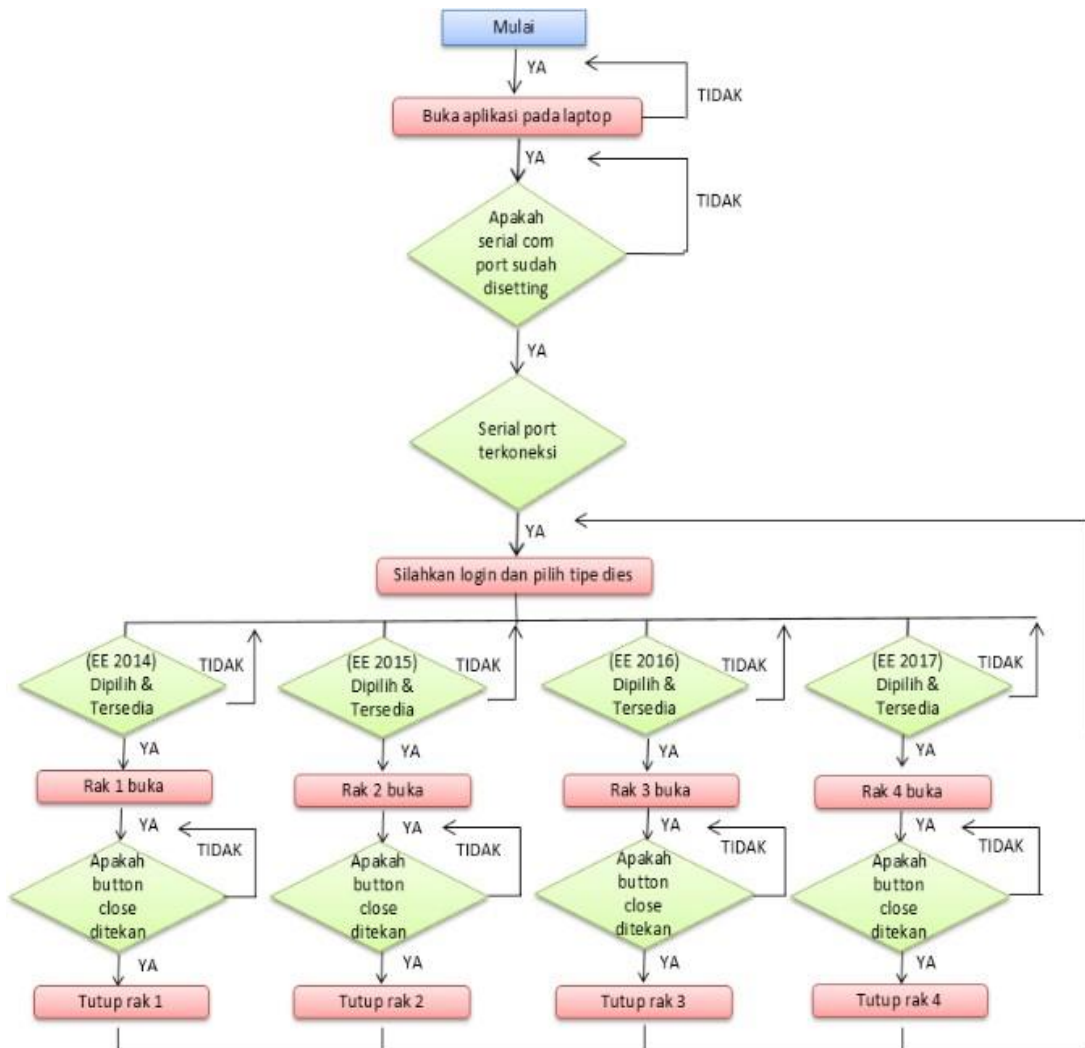
METODE PENELITIAN

Secara keseluruhan alat ini tersusun atas bagian-bagian penting yang saling berhubungan satu sama lain yaitu perangkat keras (hardware) dan perangkat lunak (software). Kedua bagian ini harus saling sinkron satu sama lain agar maksud dan tujuan dari pembuatan alat ini tercapai dan sesuai dengan yang diharapkan. Bagian hardware terdiri dari servo motor, rangkaian LED, rangkaian relay, rangkaian LCD dan Arduino. Bagian software terdiri dari program berbasis bahasa C yang disuntikan ke dalam mikrokontroler arduino mega 2560, untuk mengatur dan mengaktifkan servo motor yang nantinya untuk pengontrol keluar masuknya laci

(drawer), mengaktifkan LED sebagai indikasi laci (drawer) yang sudah berhasil di input oleh user, mengaktifkan push button sebagai kontrol manual alat jika terjadi kegagalan pada sistem

Setelah pembuatan dan perancangan *hardware* selesai, langkah selanjutnya yang di lakukan adalah pembuatan dan perancangan *software*. Pada perancangan *software* ini meliputi pemrograman dengan menggunakan arduino mega untuk *control* alat kerja dan *visual basic* untuk pembuatan aplikasi juga bisa mengontrol *database* dan *report*, agar perancangan *software* sistem mudah dilakukan dengan cepat. Maka terlebih dahulu membuat sebuah diagram *flowchart* atau diagram alir untuk setiap program yang menggambarkan jalannya program secara keseluruhan. Bisa dilihat pada Gambar 3.22 *Flowchart* sistem kerja prototipe.

3.1.1. Diagram Alir (*flowchart*)



Gambar 3.22 Flowchart sistem kerja prototipe

3.1.2. Program Software

Arduino IDE digunakan sebagai perangkat lunak untuk pembuatan program yang menunjang perangkat keras mikrokontroler Arduino.

3.4.2.1. Program Arduino Mega 2560

1. Program Arduino (1)

Penjelasan program arduino 1 pada Gambar 3.23 dibawah ini:

1. Mendefinisikan *library servo* ke program.
2. Menginisial jumlah *servo motor* yang akan digunakan.
3. Mendefinisikan *library* untuk komunikasi dengan LCD dengan model *i2c (inter integrate circuit)*.
4. Inisial *type* LCD yang akan digunakan.
5. Alamat *byte* untuk komunikasi dengan LCD.
6. Fungsi *show* untuk menampilkan karakter ke LCD, fungsi *pc, mc* adalah subrutin inisial program.
7. Pemesanan *pin input* atau *output*.

```
led_matrial_system_rev02 | Arduino 1.8.8
File Edit Sketch Tools Help
led_matrial_system_rev02 $
1
2 #include <Servo.h>
3 Servo myservo1; // create servo object to control a servo
4 Servo myservo2; // create servo object to control a servo
5 Servo myservo3; // create servo object to control a servo
6 Servo myservo4; // create servo object to control a servo
7 int potpin = 0; // analog pin used to connect the potentiometer
8 int val; // variable to read the value from the analog pin
9 #include <Wire.h>
10 // #include <LiquidCrystal_PCF8574.h>
11 //LiquidCrystal_PCF8574 lcd(0x27); // set the LCD address to 0x27 for a 16 chars and 2 line display
12 int show,pc,mc;
13 int rak1=0,rak2=0,rak3=0,rak4=0;
14 int L1=A8,L2=A9,L3=A10,L4=A11,L5=A12;
15 int s1=A0,s2=A1,s3=A2,s4=A3;
```

Gambar 3.23 Program arduino (1)

2. Program Arduino (2)

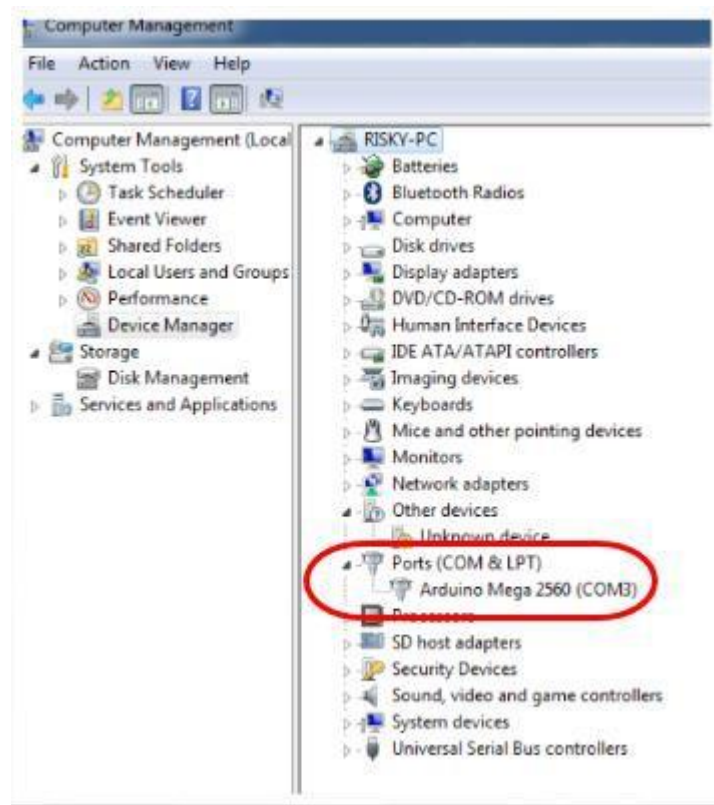
Penjelasan program arduino 2 pada Gambar 3.24 dibawah ini:

1. Berpungsi untuk menghapus *text* di LCD.
2. Mengaktifkan komunikasi dengan LCD.
3. Menentukan *address* komunikasi dengan LCD.
4. Nilai komunikasi serial *port*.
5. Mendefinisikan pin di mikrokontroler sebagai *input* atau *output*.
6. Subrutin atau *comand* untuk LCD.
7. Mendefinisikan *servo motor* ke pin mikrokontroler.
8. Mengatur kecerahan LCD.

```
led_matrial_system_rev02 | Arduino 1.8.8
File Edit Sketch Tools Help
led_matrial_system_rev02
16 //=====setup=====
17 void setup() {
18   show = 0;
19   Wire.begin();
20   Wire.beginTransmission(0x27);
21   Serial.begin(9600);
22   pinMode(s1, INPUT); pinMode(s2, INPUT); pinMode(s3, INPUT); pinMode(s4, INPUT);
23   digitalWrite(s1,1); digitalWrite(s2,1); digitalWrite(s3,1); digitalWrite(s4,1);
24   pinMode(L1, OUTPUT); pinMode(L2, OUTPUT); pinMode(L3, OUTPUT); pinMode(L4, OUTPUT); pinMode(L5, OUTPUT);
25   digitalWrite(L1,1); digitalWrite(L2,1); digitalWrite(L3,1); digitalWrite(L4,1); digitalWrite(L5,1);
26   lcd.begin(20,4); lcd.clear(); lcd.noCursor(); delay(100);
27   myservo1.attach(6); // attaches the servo on pin 9 to the servo object
28   myservo2.attach(7); // attaches the servo on pin 9 to the servo object
29   myservo3.attach(8); // attaches the servo on pin 9 to the servo object
30   myservo4.attach(9); // attaches the servo on pin 9 to the servo object
31   Serial.print("serial port initial ok....!");
32   myservo1.write(0); myservo2.write(0); myservo3.write(0); myservo4.write(0);
33   lcd.setCursor(0, 0); lcd.print("Penyimpanan Dies");
34   // lcd.setCursor(0, 1);
35   // lcd.print("Data:");
36   lcd.setBacklight(255);
37
38
```

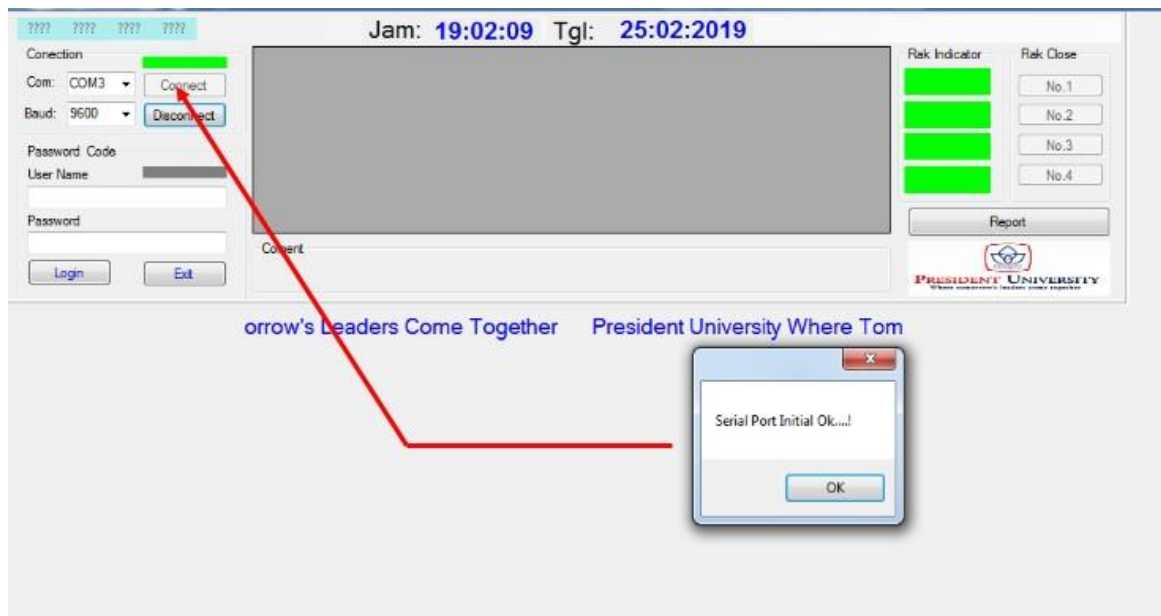
1. HASIL DAN PEMBAHASAN

Port yang digunakan Arduino Mega 2560 berada pada (com3). Sebelum pengujian aplikasi berjalan *visual basic* koneksi dengan arduino harus dipastikan dulu berada pada *port* (com) berapa arduino mega 2560 agar aplikasi dapat berjalan, dapat dilihat dari Gambar 4.1 untuk mengetahui *port* (com).



Gambar 4.1 Menentukan port pada device manager

Setelah mengetahui port com arduino mega 2560 dengan benar maka akan muncul komentar (Serial Port Initial Ok...!) maka aplikasi *visual basic* sudah bisa digunakan dan *visual basic* akan terhubung dengan program Arduino [8]. Pada tahapan pertama setiap user harus memasukkan nama dan password sesuai dengan yang dimiliki atau yang sudah diatur sebelumnya jika login sudah berhasil tahapan berikutnya user memilih *type dies* apa yang ingin di ambil. Bisa kita lihat pada Gambar 4.2 dibawah ini.

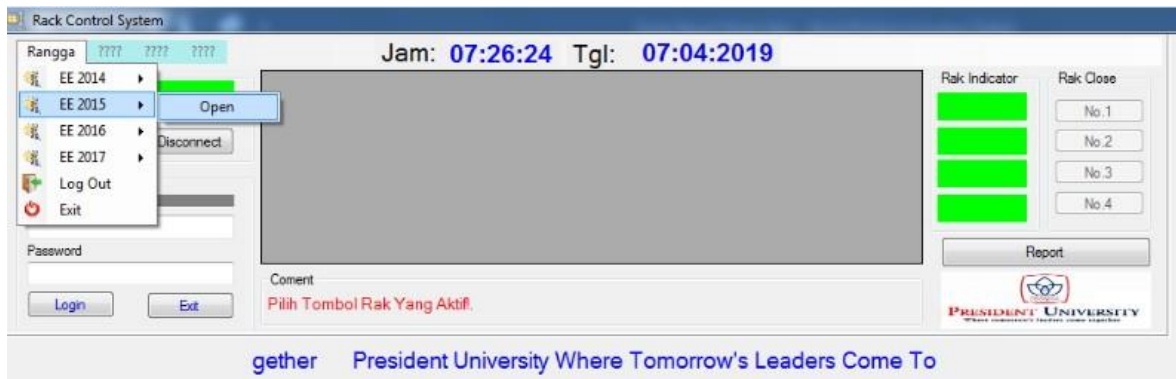


Gambar 4.2 Aplikasi visual basic terhubung dengan arduino

4.1.1. Pengujian Kontrol Rak Sistem Setiap User

a. Pengujian Kontrol Rak Sistem Oleh Rangga

Pada Gambar 4.3 menampilkan aplikasi *visual basic* pengambilan *dies* yang dilakukan oleh **rangga** pada *type dies* (EE 2015) yang berada pada rak nomer 2.



Gambar 4.3 Menentukan dies yang ingin diambil oleh rangga

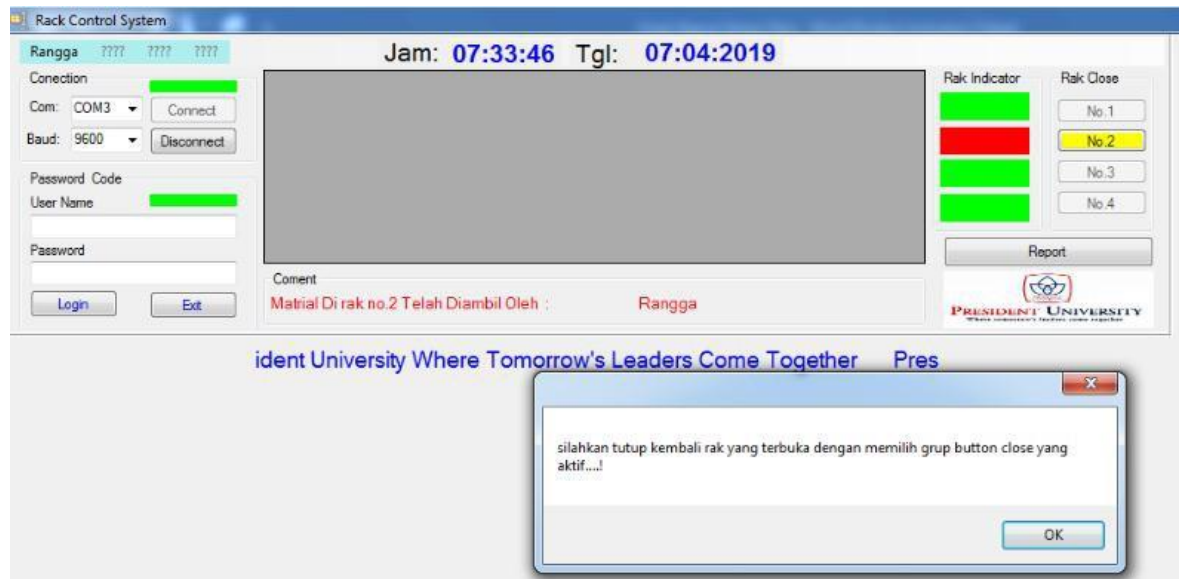
Pada gambar 4.4 menampilkan rak 2 terbuka, yang sebelumnya sudah ditentukan oleh **rangga** untuk pengambilan *type dies*.



Gambar 4.4 Menampilkan rak 2 terbuka

Pada Gambar 4.5 menampilkan indikator yang sebelumnya berwarna hijau berubah menjadi merah

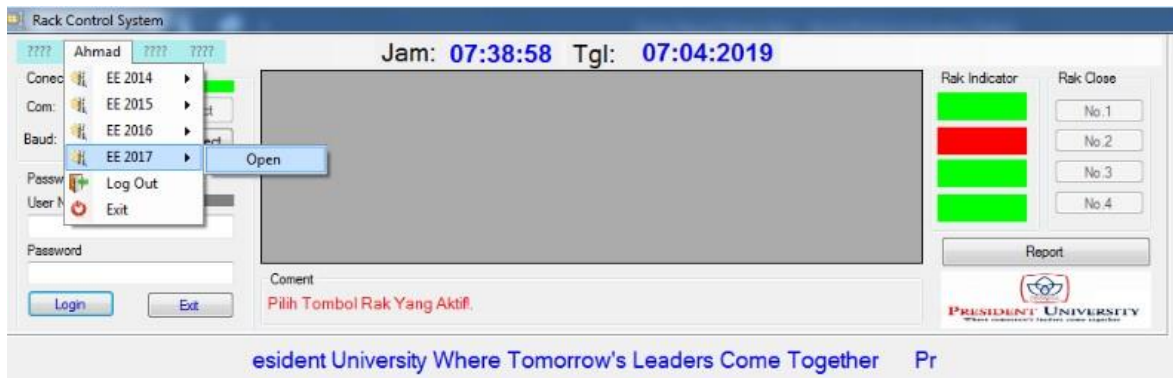
setelah *dies* sudah diambil oleh **rangka** pada rak 2. Ketika rak ingin ditutup, setiap *user* harus menekan tombol rak *close* agar rak yang terbuka bisa tertutup kembali.



Gambar 4.5 Menampilkan indikator rak 2 ketika *dies* sudah diambil

b. Pengujian Kontrol Rak Sistem Oleh Ahmad

Pada Gambar 4.6 menampilkan aplikasi *visual basic* pengambilan *dies* yang dilakukan oleh **ahmad** pada *type dies* (EE 2017) yang berada pada rak nomer 4.



Gambar 4.6 Menentukan *dies* yang ingin diambil oleh ahmad

Pada Gambar 4.7 menampilkan rak 4 terbuka, yang sebelumnya sudah ditentukan oleh **ahmad** untuk pengambilan *type dies*.



Gambar 4.7 Menampilkan rak 4 terbuka

2. KESIMPULAN DAN SARAN

Berdasarkan hasil pengujian dan hasil perancangan alat penyimpanan dies ini dapat diambil kesimpulan:

1. Terwujudnya alat ini ada beberapa pemrograman yang dilibatkan, diantaranya Arduino Programming, Visual Basic Grafik Programming, Visual Basic Coding Programming, dan MS. Access sebagai database, dari pemrograman tersebut bisa mengendalikan rangkaian “sistem penyimpanan dies”.

2. Perancangan “sistem penyimpanan dies” yang telah dibuat, diharapkan servo motor dapat menggerakkan laci (drawer). Berdasarkan pengujian dan analisa yang dilakukan, dapat disimpulkan bahwa perancangan dan pembuatan alat pengendali servo motor DC standart berbasis mikrokontroller Arduino Mega 2560 berfungsi dengan baik, namun masih terdapat beberapa kekurangan yang diantaranya ketepatan perputaran pada motor. Untuk putaran 90 derajat yang diperlukan meggerakkan keluarnya laci pada alat ini penulis hanya memberikan 43 derajat pada program Arduino dikarenakan tidak sinkronya pada alat.

Dan di saran penulis bisa menyimpulkan

1. Kedepannya diharapkan mampu dikembangkan lagi, dengan pengambilan dan penyimpanan menggunakan robot yang berfungsi mempermudah pengambilan dan penyimpanan dies.

2. Di kembangkan dan di kolaborasikan dengan sistem monitoring berbasis web agar kinerja alat lebih optimal dan mudah dalam monitoring dimanapun kita berada, karena alat yang penulis buat masih terbatas pada ruang lingkup tertentu.

DAFTAR PUSTAKA

- [1] Sutabri, Tata. 2000. Analisis Sistem Informasi, Andi Offset, Yogyakarta.
- [2] Nurfiana dan Danang Ade Muktiawan, Jurnal “Rancang Bangun Monitoring Alat Penyimpanan Berbasis Mikrokontroler”.
- [3] Gunterus, Frans. 1994. Falsafah Dasar. Sistem Pengendalian Proses. PT.Elex Komputindo. Jakarta.
- [4] Yohanes C. Saghoa, Sheerwin R.U.A Shompie, dan Novi M. Tulung, Jurnal “Kotak Penyimpanan Uang Bebas Mikrokontroler Arduino Uno”.
- [5] Feri Siswoyo Hadisanatoso, dan Afrezi Tamsir, Jurnal “Perancangan Sistem Penyimpanan dan Pengambilan Barang Otomatis Menggunakan Arduino Berbasis Web Database”.
- [6] Anonim. 2011. Pengertian Catu Daya atau Power Supply. Diakses pada tanggal 20 Juni 2014.
- [7] George H Martin, 1984. Kinematika dan Dinamika Teknik. Penerbit Erlangga, 1994.
- [8] Komunikasi Serial Arduino dan Visual Basic : <https://elmechtechnology.com>
- [9] Menghubungkan Database Microsoft Access Dengan Visual Basic : <https://www.kaskus.co.id>