



**RANCANG BANGUN KANDANG PINTAR UNTUK MELATIH BURUNG  
KICAU BERBASIS INTERNET OF THINGS DAN ANDROID**

**“Rancang Bangun Kandang Pintar Untuk Melatih Burung Kicau Berbasis  
Internet Of Things”**

**TUGAS AKHIR**

**FAZIHAN TACHTA FIRDAUS  
1803332058**

**PROGRAM STUDI TELEKOMUNIKASI  
JURUSAN TEKNIK ELEKTRO  
POLITEKNIK NEGERI JAKARTA  
2021**

HALAMAN JUDUL



**RANCANG BANGUN KANDANG PINTAR UNTUK MELATIH BURUNG  
KICAU BERBASIS INTERNET OF THINGS DAN ANDROID**

**“Rancang Bangun Kandang Pintar Untuk Melatih Burung Kicau Berbasis  
Internet of Things”**

**TUGAS AKHIR**

**Diajukan Diajukan sebagai salah satu syarat untuk memperoleh gelar  
Diploma Tiga**

**POLITEKNIK  
NEGERI  
JAKARTA**

**FAZIHAN TACHTA FIRDAUS  
1803332058**

**PROGRAM STUDI TELEKOMUNIKASI  
JURUSAN TEKNIK ELEKTRO  
POLITEKNIK NEGERI JAKARTA  
2021**

© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta





## HALAMAN PERNYATAAN ORISINALITAS

Tugas akhir ini adalah hasil karya sendiri dan semua sumber baik yang dikutip maupun dirujuk telah saya nyatakan dengan benar.

Nama : Fazihan Tachta Firdaus

NIM : 1803332012

Tanda Tangan :

Tanggal : 29 Juli 2021

POLITEKNIK  
NEGERI  
JAKARTA

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkannya dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

## HALAMAN PENGESAHAN TUGAS AKHIR

### Tugas akhir diajukan oleh:

Nama Mahasiswa I : Fazihan Tachta Firdaus  
NIM : 1803332058  
Program Studi : Teknik Telekomunikasi  
Judul Tugas Akhir : Rancang Bangun Sistem Kandang Untuk  
Melatih Burung Kicau Berkualitas Berbasis  
Internet of Things dan Android  
Sub Judul : Rancang Bangun Kandang Pintar Untuk Melatih  
Burung Kicau Berbasis Internet of Things

Telah diuji oleh tim penguji dalam Sidang Tugas Akhir pada Hari Kamis dan  
Tanggal 29 Juli 2021 dan dinyatakan **LULUS**.

Pembimbing : Rifqi Fuadi Hasani, S.T., M.T.

NIP. 199208182019031015

()

Depok, .....

Disahkan oleh

Ketua Jurusan Teknik Elektro



Ir. Sri Danaryani, M.T.

NIP. 1963 0503 99103 2 001



**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkannya dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

## KATA PENGANTAR

Puji syukur saya panjatkan kepada Tuhan Yang Maha Esa, karena atas berkat dan rahmat-Nya, penulis dapat menyelesaikan Tugas Akhir yang berjudul Rancang Bangun Sistem Kandang Untuk Melatih Burung Kicau Berkualitas Berbasis Internet of Things dan Android “*Rancang Bangun Kandang Pintar Untuk Melatih Burung Kicau Berbasis Internet of Things*”. Penulisan Tugas Akhir ini dilakukan dalam rangka memenuhi salah satu syarat untuk mencapai gelar Diploma Tiga Politeknik.

Penulis menyadari bahwa, tanpa bantuan dan bimbingan dari berbagai pihak, dari masa perkuliahan sampai pada penyusunan tugas akhir ini, sangatlah sulit bagi penulis untuk menyelesaikan tugas akhir ini. Oleh karena itu, penulis mengucapkan terima kasih kepada:

1. Rifqi Fuadi Hasani, ST.,MT selaku dosen pembimbing yang telah menyediakan waktu, tenaga, dan pikiran untuk mengarahkan penulis dalam penyusunan tugas akhir ini;
2. Seluruh Staf Pengajar dan Karyawan Jurusan Teknik Elektro Politeknik Negeri Jakarta, khususnya Program Studi Telekomunikasi;
3. Orang tua dan keluarga penulis yang telah memberikan bantuan dukungan material dan moral;
4. Tito Andi Baskoro selaku rekan Tugas Akhir dan juga teman teman di Program Studi Telekomunikasi Angkatan 2018 yang telah mendukung serta bekerja sama untuk menyelesaikan Tugas Akhir ini.

Akhir kata, penulis berharap Tuhan Yang Maha Esa berkenan membalas segala kebaikan semua pihak yang telah membantu. Semoga Tugas Akhir ini membawa manfaat bagi pengembangan ilmu.

Depok, Agustus 2021

Penulis

Fazihan Tachta Firdaus





Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

## RANCANG BANGUN KANDANG PINTAR UNTUK MELATIH BURUNG KICAU BERBASIS INTERNET OF THINGS DAN ANDROID

*“Rancang Bangun Kandang Pintar Untuk Melatih Burung Kicau Berbasis Internet of Things”*

### ABSTRAK

*Pada saat ini banyak masyarakat yang memiliki hobi memelihara burung kicau dan sebagian dari masyarakat memiliki kesibukan tertentu dalam melakukan kegiatan sehari-hari, sehingga burung kicau menjadi tidak terpelihara dengan baik, dan menghasilkan burung yang tidak sehat dan tidak berkualitas bahkan bisa menyebabkan kematian pada burung kicau yang dapat merugikan bagi yang memelihara burung karna memiliki nilai jual yang mahal. Pemeliharaan burung kicau cucak ranting merupakan hal yang cukup mudah akan tetapi harus telaten untuk mengurus burung tersebut. Perawatan burung kicau cucak ranting juga mudah dilatih agar menjadi burung yang aktif untuk berkicau. Harga burung cucak ranting kalangan pecinta burung kicau termasuk mahal dikarenakan warna bulu dan kicaunya yang indah dan merdu. Untuk perancangan alat ini dibuat sistem kandang pintar untuk melatih burung kicau dalam suatu sistem IoT menggunakan mikrokontroler Raspberry Pi diantaranya adalah sistem keamanan burung dengan RFID MFRC522, pemberi pangan otomatis dengan servo SG90S, dan sistem suara yang terdapat di kandang burung sebagai pemeliharaan dan pelatihan burung kicau agar menjadi sehat sehingga memiliki harga jual yang tinggi.*

**Kata Kunci:** *Raspberry Pi, RFID MFRC522, Servo SG90S*

### Abstract

*At this time, many people have a hobby of keeping chirping birds and some of the people have certain activities in carrying out daily activities, so that chirping birds are not well cared for, and produce unhealthy and quality birds that can even cause death in chirping birds. can be detrimental to those who keep birds because it has an expensive selling value. Maintenance of the nagging cucak chirping bird is quite easy but care must be taken to take care of the bird. Cucak ranting bird care is also easy to use to become an active bird for chirping. The price of the cucak ranting bird among chirping bird lovers is expensive because of the beautiful and melodious color of the feathers and the chirping. To design this tool, a smart cage system was created to train chirping birds in an IoT system using a Raspberry Pi microcontroller including a bird security system using RFID MFRC522, automatic feeding using the SG90S servo, and a sound system in the bird cage as maintenance and training. chirping birds to be healthy and of high quality so that they have a high selling price.*

**Keywords:** *Raspberry Pi, RFID MFRC522, Servo SG90S*



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

DAFTAR ISI

	Halaman
HALAMAN SAMBUNG.....	ii
HALAMAN JUDUL .....	ii
HALAMAN PERNYATAAN ORISINALITAS .....	iii
HALAMAN PENGESAHAN.....	iv
KATA PENGANTAR.....	iv
ABSTRAK .....	vi
DAFTAR ISI.....	vii
DAFTAR GAMBAR.....	lix
DAFTAR TABEL .....	x
DAFTAR LAMPIRAN .....	xi
<b>BAB I PENDAHULUAN.....</b>	<b>1</b>
1.1. Latar Belakang.....	1
1.2. Permasalahan .....	2
1.3. Tujuan .....	2
1.4. Luaran .....	2
<b>BAB II TINJAUAN PUSTAKA.....</b>	<b>3</b>
2.1. State of Art.....	3
2.2. Pengertian Internet of Things (IoT).....	4
2.2.1. Unsur-Unsur Internet of Things.....	4
2.2.2. Cara Kerja Internet of Things .....	5
2.2. Burung Cucak Ranting .....	6
2.3. Raspberry .....	7
2.3.1. Raspberry Pi 3 .....	9
2.3.2 GPIO Raspberry Pi 3 .....	10
2.4. Python.....	10
2.5. RFID .....	11
2.6. MFRC522 .....	14
2.7. MySQL .....	15
2.8. Speaker MP3.....	16
2.9. LCD (Liquid Crystal Display) 16x2.....	16
2.10. Relay .....	17
2.11. Solenoid Door Lock.....	18
2.12. Servo .....	19
<b>BAB III RANCANGAN DAN REALISASI .....</b>	<b>20</b>
3.1. Rancangan Alat.....	20
3.1.1. Deskripsi Alat .....	20
3.1.2. Cara Kerja Alat .....	21
3.1.4. Diagram Blok.....	26
3.2. Perancangan Alat.....	27
3.2.1. Perancangan Sistem Keamanan Kandang Pintar .....	27
3.2.2. Perancangan Sistem Pemberi Makan Otomatis dengan Servo .....	29
3.2.3. Perancangan Speaker Kicau Burung dengan Audio Jack 3,5 mm. ....	30
3.2.4. Perancangan Catu Daya .....	31
3.3. Realisasi Alat.....	32





**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

3.3.1. Realisasi Pembuatan sistem keamanan menggunakan RFID RC-522 dan solenoid doorlock pada Raspberry pi 3 b.....	32
3.3.2. Realisasi Pembuatan sistem pemberi makan burung menggunakan Servo pada Raspberry pi 3 b.....	34
3.3.3. Realisasi pembuatan sistem audio masteran kicau burung menggunakan output audio jack 3,5 mm pada Raspberry Pi 3.....	36
<b>BAB IV PEMBAHASAN.....</b>	<b>41</b>
4.1. Pengujian Sistem Keamanan .....	41
4.1.1. Deskripsi Pengujian .....	41
4.1.2. Prosedur Pengujian .....	41
4.1.3. Data Hasil Pengujian .....	42
4.2. Pengujian Jumlah Pakan Otomatis Menggunakan Servo .....	44
4.2.1. Deskripsi Pengujian .....	44
4.2.3. Data Pengujian.....	44
4.3. Pengujian Catu Daya .....	45
4.3.1. Deskripsi Pengujian.....	45
4.3.2. Prosedur Pengujian .....	45
4.3.3. Data Hasil Pengujian .....	46
4.4. Analisa Data/Evaluasi.....	47
<b>BAB V PENUTUP .....</b>	<b>48</b>
5.1. Simpulan.....	45
5.2. Saran .....	48
<b>DAFTAR PUSTAKA .....</b>	<b>49</b>
<b>DAFTAR RIWAYAT HIDUP .....</b>	<b>50</b>
<b>LAMPIRAN.....</b>	<b>51</b>

**POLITEKNIK  
NEGERI  
JAKARTA**





Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

## DAFTAR GAMBAR

	Halaman
Gambar 2.1. Cara kerja Internet of Thing .....	5
Gambar 2.2. Burung Cucak Ranting .....	6
Gambar 2.3. Dekstop Rasberry Pi 3 .....	8
Gambar 2.4. Raspberry pi 3 .....	9
Gambar 2.5. Pinout Rasperry Pi 3 GPIO .....	10
Gambar 2.6. RFIF <i>Tag Keychain</i> .....	13
Gambar 2.7 RFID <i>Card Tag</i> .....	14
Gambar 2.8. MFRC 522 .....	14
Gambar 2.9. LCD I2C 16x2 .....	17
Gambar 2.10. Relay .....	18
Gambar 2.11. Selonoid <i>Doorlock</i> .....	18
Gambar 2.12. Servo DC SG90S .....	19
Gambar 3.1. Ilustrasi Sistem .....	23
Gambar 3.2. Flowchart Sistem Kandang Pintar .....	24
Gambar 3.3. Diagram Blok Kandang Pintar .....	28
Gambar 3.4. Gambar Schematic dari Alat Kandang Pintar .....	29
Gambar 3.5. Gambar <i>schematic</i> keamanan RFID .....	29
Gambar 3.6. Gambar <i>schematic</i> Servo .....	31
Gambar 3.7 Gambar Schematic Speaker dengan Rasperry pi 3 .....	32
Gambar 3.8 Diagram <i>schematic power supply</i> .....	33
Gambar 4. 1 Hasil Pengukuran Tegangan RFID MFRC-522 .....	39
Gambar 4.2. Hasil Pengukuran Tegangan <i>selonoid doorock</i> .....	40
Gambar 4.3. Hasil Pengukuran Tegangan .....	41
Gambar 4.4. Keluaran dari catu daya 5V .....	42
Gambar 4. 5 Keluaran dari catu daya 12V .....	43



**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

**DAFTAR TABEL**

	<b>Halaman</b>
Tabel 2. 1 Perbedaan kartu tag aktif dan kartu tag pasif.....	12
Tabel 3.1 Spesifikasi <i>RFID MFRC-522</i> .....	23
Tabel 3. 2 Pin Spesifikasi <i>selonoid doorlock</i> .....	23
Tabel 3. 3 Spesifikasi <i>speaker</i> .....	24
Tabel 3. 4 Spesifikasi servo motor SG90S.....	24
Tabel 3. 5 Spesifikasi LCD 16×2 (Liquid Crystal Display) .....	25
Tabel 3. 6 Spesifikasi relay .....	25
Tabel 3. 7 Spesifikasi Raspberry Pi 3 b .....	26
Tabel 3. 8 Pin RFID MFRC-522 dengan Pin Raspberry Pi.....	28
Tabel 3. 9 Pin LCD I2C 16x2 dengan Pin Raspberry Pi.....	28
Tabel 3. 10 Pin Relay dengan Pin Raspberry Pi .....	29
Tabel 3. 11 Pin Servo dengan Pin Raspberry Pi .....	30
Tabel 4.1. Data Nomer RFID yang dapat diakses.....	42
Tabel 4.2. Tegangan MFRC-522 .....	42
Tabel 4.3. Tegangan <i>Selonoid Doorlock</i> .....	43
Tabel 4.4. Pengujian Tegangan Relay.....	44
Tabel 4.5. Pengujian Jumlah Makan Otomatis .....	45
Tabel 4.6. Tegangan Servo SG90S .....	43

**POLITEKNIK  
NEGERI  
JAKARTA**





**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

**DAFTAR LAMPIRAN**

	<b>Halaman</b>
Lampiran 1. Skematik Rangkaian Catu Daya .....	52
Lampiran 2. Diagram Skematik Keseluruhan Sistem .....	53
Lampiran 3. Gambar Kandang Pintar .....	54
Lampiran 4-Datasheet Raspberry .....	55
Lampiran 5-Datasheet RFID MFRC-522.....	56
Lampiran 6-Datasheet LCD .....	57
Lampiran 7 Datasheet LCD I2C .....	58
Lampiran 8-Datasheet Servo SG90.....	59
Lampiran 9-Datasheet <i>Selonoid Doorlock</i> .....	60





Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

## BAB I PENDAHULUAN

### 1.1. Latar Belakang

*Internet of Things* merupakan sebuah konsep yang bertujuan untuk memperluas manfaat dari konektivitas internet yang tersambung secara terus-menerus. Perkembangan internet pada revolusi industri 4.0 ini sudah mengalami kemajuan yang sangat pesat dan menjadi suatu hal yang sangat bermanfaat dan penting untuk masyarakat. Internet pada saat ini banyak digunakan untuk mencari suatu informasi dan digunakan untuk berkomunikasi. Seiring dengan pesatnya teknologi di zaman seperti sekarang ini, telepon genggam telah dilengkapi dengan sistem layaknya seperti komputer, yang disebut *smartphone*. Aplikasi mobile dalam hal ini adalah Android yang menjadi salah satu media elektronik yang sangat strategis. Android merupakan salah satu sistem operasi mobile yang sangat populer dan pengoperasiannya cukup mudah.

Penggunaan aplikasi android dalam *Internet of Things* pada era industri 4.0 semakin banyak diterapkan dalam kehidupan sehari-hari. Adapun penggunaannya seperti berbagi data, *remote control*, dan penerimaan sensor, termasuk juga pada benda. *Internet Of Things* yang berperan penting dalam pemeliharaan burung, menggunakan berbagai sensor dan sistem kebersihan, keamanan, pangan, dan pelatihan kicauan burung. Pada era sekarang banyak masyarakat yang memiliki hobi memelihara burung kicau dan sebagian dari masyarakat memiliki kesibukan tertentu dalam melakukan kegiatan sehari-hari, sehingga burung kicau menjadi tidak terpelihara dengan baik, dan menghasilkan burung yang tidak sehat dan tidak berkualitas bahkan bisa menyebabkan kematian pada burung kicau yang dapat merugikan bagi yang memelihara burung karena memiliki nilai jual yang mahal.

Pada sistem pakan otomatis dilengkapi sistem keamanan pada kandang dan *speaker* yang berbunyi di waktu tertentu untuk melatih kicauan burung, oleh karena itu akan dibuat suatu alat dengan judul “*Rancang Bangun Kandang Pintar Untuk Melatih Burung Kicau Berbasis Internet of Things*”

”




**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

**1.2**
**Permasalahan**

Berdasarkan latar belakang yang diuraikan di atas, maka permasalahan yang akan dibahas dalam tugas akhir ini adalah sebagai berikut :

1. Bagaimana kandang pintar dapat melatih burung berkicau?
2. Bagaimana kandang pintar dapat memberi makan otomatis?
3. Bagaimana sistem keamanan pada kandang pintar ini?

**1.3**
**Tujuan**

1. Dapat merancang kandang pintar untuk melatih burung kicau berbasis *Internet of Things*.
2. Mampu melakukan pengujian alat agar sesuai spesifikasi yang diharapkan.
3. Mampu menghubungkan kandang pintar ke database menggunakan Raspberry Pi 3b.

**1.4**
**Luaran**

Adapun luaran dari Tugas Akhir ini adalah:

1. Menghasilkan alat kandang pintar untuk melatih burung kicau berbasis *Internet of Things* yang dapat membantu masyarakat yang memiliki hobi memelihara burung.
2. Menghasilkan buku laporan Tugas Akhir yang dapat digunakan sebagai sumber refrensi bagi masyarakat dan mahasiswa berkaitan dengan kandang pintar berbasis *Internet of Things*.

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

## BAB V PENUTUP

### 5.1. Simpulan

Berdasarkan data hasil pembuatan Tugas Akhir "Rancang Bangun Sistem Kandang Untuk Melatih Penghasil Burung Kicau Berkualitas berbasis Internet of Things dan Android" dapat disimpulkan bahwa:

1. Pembuatan kandang pintar dapat melatih burung kicau dengan menggunakan *speaker* yang mengeluarkan suara masteran burung cucak ranting dan terhubung dengan Raspberry dan dapat diatur untuk memutar suaranya.
2. Pada kandang pintar terdapat beberapa fitur seperti pemberian makan otomatis menggunakan servo memudahkan untuk memberi makan burung. Makanan burung yang keluar secara otomatis dapat memenuhi makan burung dalam waktu sekali makan. Sistem lainnya adalah sistem keamanan pada kandang burung menggunakan RFID sebagai akses kunci dan *selonoid doorlock* sebagai slot kunci pada pintu.
3. Pembuatan kandang pintar untuk melatih burung kicau dapat dihubungkan ke *database* MySQL yang terdapat di internet agar dapat dihubungkan dengan aplikasi android.

### 5.2. Saran

Dalam perawatan burung kicau cucak ranting melalui aplikasi android harus diperhatikan pemberian makanan pada kandang apakah masuh ada sisa atau tidak demi kebersihan kandang burung dikarenakan kesehatan dan kebersihan kandang adalah hal yang utama.



## DAFTAR PUSTAKA

- Asfihan, Akbar, 2021. Mysql adalah: Cara Kerja, Sejarah, Manfaat, Kelebihan dan Kekurangannya. <https://adalah.co.id/mysql/>[diakses pada 17 Februari 2021]
- Adm, Zanoor, 2020. Pengertian Speaker: Fungsi, Jenis, Perbedaan Speaker Aktif dan Pasif. <https://www.zanoor.com/pengertian-speaker/> [diakses pada tanggal 23 Maret 2021]
- Cahyono, G. H. (2017), “Internet of Things ( Sejarah, Teknologi, dan Penerapannya)”, Forum Teknologi .
- E. Rakhman, F. Candrasyah, dan F. D. Sutera, Raspberry Pi–mikrokontroler mungil yang serba bisa. Yogyakarta: Andi, 2014
- Hewanesia, Redaksi (2018) Info Lengkap Harga Cucak Rante Terbaru. [hewanesia.com/harga-cucak-rante/](http://hewanesia.com/harga-cucak-rante/) [1 agustus 2021]
- Kadir, Abdul. (2017). Dasar Raspberry Pi. Yogyakarta: CV Andi Offset
- Kho, Dickson, 2021. Pengertian Relay dan Fungsinya. <https://teknikelektronika.com/pengertian-relay-fungsi-relay/> [diakses pada 18 Februari 2021]
- Munandar Aris, 2012. Liquid Crystal Display (LCD) 16 x 2. [http://www.leselektronika.com/2012/06/liquid-crystal-display-lcd-16-x-2.html#:~:text=LCD%20\(Liquid%20Crystal%20Display\)%20adalah,kalkulator%2C%20atau%20pun%20layar%20komputer.](http://www.leselektronika.com/2012/06/liquid-crystal-display-lcd-16-x-2.html#:~:text=LCD%20(Liquid%20Crystal%20Display)%20adalah,kalkulator%2C%20atau%20pun%20layar%20komputer.) [diakses pada 23 Maret 2021]
- M. Sirait dan K. Tanjung, “Perancangan sistem keamanan akses buka pintu menggunakan RFID (Radio Frequency Identification) dan pengiriman informasi ke ponsel,” Jurnal SINGUDA ENSIKOM, vol.13, no.37, hal.129 – 133, 2015.
- Nasution, Anwar Kholidi. 2015. Rancang Bangun Alat Pemberi Pakan dan Pengatur Suhu Otomatis untuk Ayam Pedaging Berbasis Programmable Logic Controller pada Kandang Tertutup. Skripsi. Universitas Lampung.
- Pudin, Saripudin, Kendali Motor Servo dengan Pulse Width Modulation (PWM) pada Mikrokontroler AVR, (Online), (<http://pudintekel.com/2011/09/kendali-motor-servo-dengan-pulse-width.html>, diakses tanggal 18 Juli 2021 jam 08.22 WIB)
- S. Winardi, Firmansyah, dan W. A.Kristiana, “Rancang bangun sistem pengaman pintu rumah menggunakan android berbasis Arduino Uno,” e-Jurnal NARODROID, vol. 2, no.1, hal. 98 – 104, 2016

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

## DAFTAR RIWAYAT HIDUP



Fazihan Tachta Firdaus lahir di Jakarta, 16 Juli 1999. Memulai pendidikan formal di SDN Gedong 09 Pagi Jakarta tahun 2005 hingga lulus tahun 2011. Setelah itu melanjutkan pendidikan SMPN 223 Jakarta, lulus pada tahun 2014. Lalu melanjutkan pendidikan di SMA Negeri 62 Jakarta, lulus pada tahun 2017. Setelah lulus dari SMA, penulis melanjutkan pendidikan ke jenjang diploma III di jurusan teknik elektro program studi telekomunikasi politeknik negeri Jakarta.



POLITEKNIK  
NEGERI  
JAKARTA





# LAMPIRAN

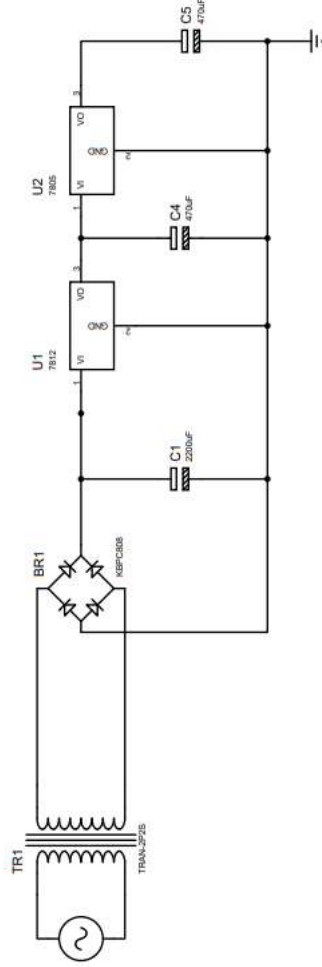
## © Hak Cipta milik Politeknik Negeri Jakarta



### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Lampiran 1. Skematik Rangkaian Catu Daya



# RANGKAIAN SKEMATIK POWER SUPPLY

PROGRAM STUDI TELEKOMUNIKASI

BURUSAN TEKNIK ELEKTRO – POLITEKNIK NEGERI

JAKARTA

Digambar

Fazihan Tachta Firdaus

Diperiksa

Rifqi Fuadi Hasani, ST.,MT

Tanggal

Agustus 2021

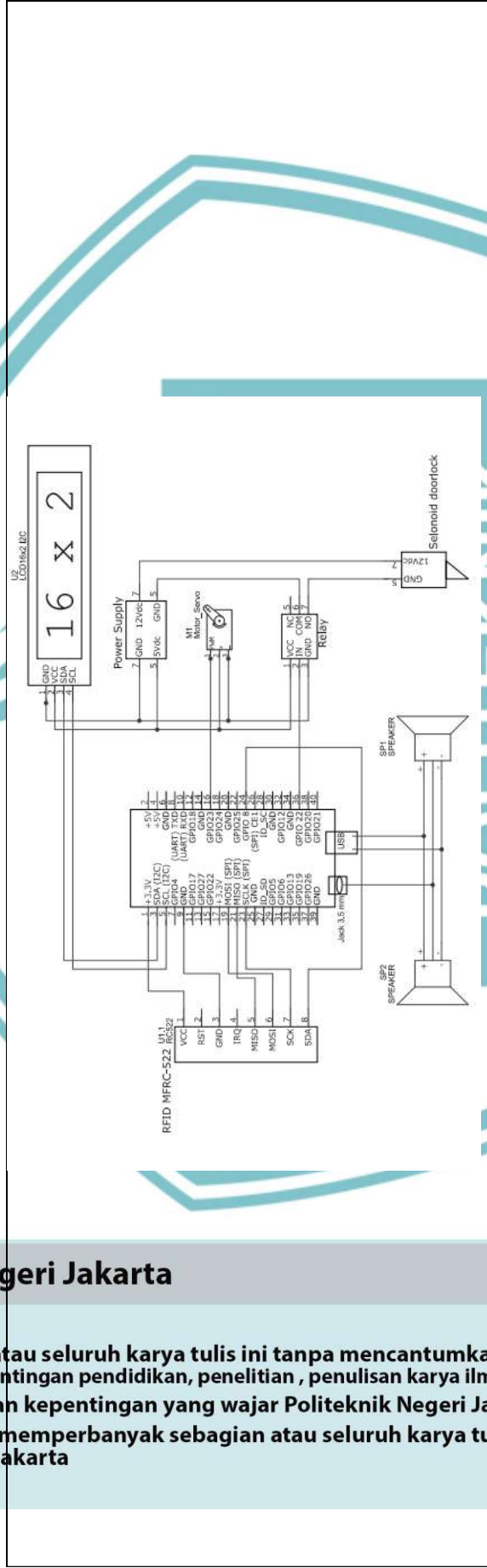


Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan umum
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Lampiran 2. Diagram Skematik Keseluruhan Sistem



# 02 **DIAGRAM SKEMATIK KESELURUHAN SISTEM**

<b>PROGRAM STUDI TELEKOMUNIKASI</b>		<i>Digambar</i>	Fazihan Tachta Firdaus
<b>PURUSAN TEKNIK ELEKTRO – POLITEKNIK NEGERI</b>		<i>Diperiksa</i>	Rifqi Fuadi Hasani, ST.,MT
<b>JAKARTA</b>		<i>Tanggal</i>	Agustus 2021



Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan umum
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Lampiran 3. Gambar Kandang Pintar

	<p style="text-align: center;"><b>GAMBAR KANDANG PINTAR</b></p>	Fazihan Tachta Firdaus
Digambar		Rifqi Fuadi Hasani, ST., MT
Diperiksa		Tanggal
<p style="text-align: center;"><b>03</b></p>	<p style="text-align: center;"><b>PROGRAM STUDI TELEKOMUNIKASI</b> <b>JURUSAN TEKNIK ELEKTRO – POLITEKNIK NEGERI</b> <b>JAKARTA</b></p> 	Agustus 2021

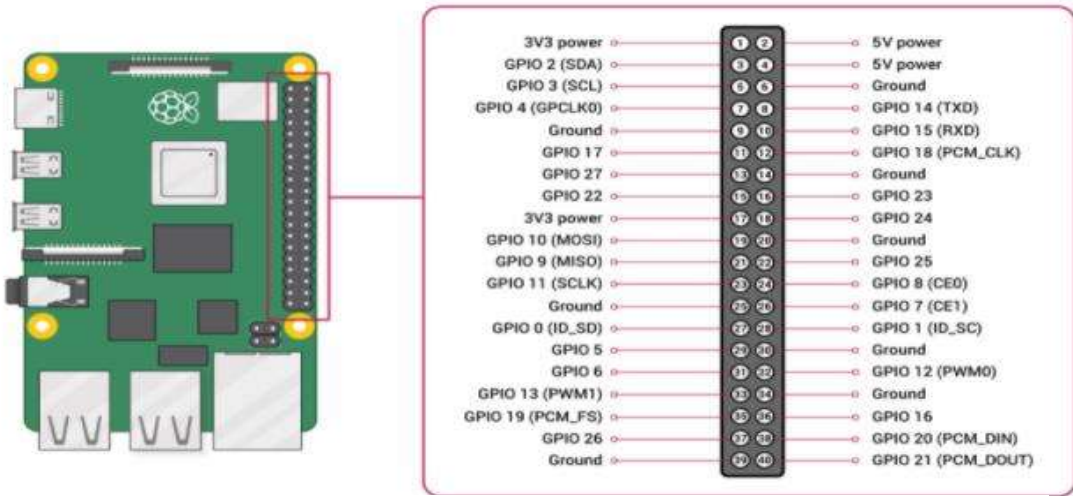
Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan umum
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



## Lampiran 4-Datasheet Raspberry



The Raspberry Pi Compute Module 3+ (CM3+) is a range of DDR2-SODIMM-mechanically-compatible System on Modules (SoMs) containing processor, memory, eMMC Flash (on non-Lite variants) and supporting power circuitry. These modules allow a designer to leverage the Raspberry Pi hardware and software stack in their own custom systems and form factors. In addition these modules have extra IO interfaces over and above what is available on the Raspberry Pi model A/B boards, opening up more options for the designer.

The CM3+ contains a BCM2837B0 processor (as used on the Raspberry Pi 3B+), 1Gbyte LPDDR2 RAM and eMMC Flash. The CM3+ is currently available in 4 variants, CM3+/8GB, CM3+/16GB, CM3+/32GB and CM3+ Lite, which have 8, 16 and 32 Gigabytes of eMMC Flash, or no eMMC Flash, respectively.

The CM3+ Lite product is the same as CM3+ except the eMMC Flash is not fitted, and the SD/eMMC interface pins are available for the user to connect their own SD/eMMC device.

Note that the CM3+ is electrically identical and, with the exception of higher CPU z-height, physically identical to the legacy CM3 products.

CM3+ modules require a software/firmware image dated November 2018 or newer to function correctly.

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



## MFRC522

### Contactless Reader IC

Rev. 3.2 — 22 May 2007  
112132

Product data sheet  
PUBLIC INFORMATION

### 1. Introduction

This document describes the functionality of the contactless reader/writer MFRC522. It includes the functional and electrical specifications.

### 2. General description

The MFRC522 is a highly integrated reader/writer for contactless communication at 13.56 MHz. The MFRC522 reader supports ISO 14443A / MIFARE® mode.

The MFRC522's internal transmitter part is able to drive a reader/writer antenna designed to communicate with ISO/IEC 14443A/MIFARE® cards and transponders without additional active circuitry. The receiver part provides a robust and efficient implementation of a demodulation and decoding circuitry for signals from ISO/IEC 14443A/MIFARE® compatible cards and transponders. The digital part handles the complete ISO/IEC 14443A framing and error detection (Parity & CRC). The MFRC522 supports MIFARE® Classic (e.g. MIFARE® Standard) products. The MFRC522 supports contactless communication using MIFARE® higher transfer speeds up to 848 kbit/s in both directions.

Various host interfaces are implemented:

- SPI interface
- serial UART (similar to RS232 with voltage levels according pad voltage supply)
- I<sup>2</sup>C interface.

### 3. Features

- Highly integrated analog circuitry to demodulate and decode responses
- Buffered output drivers to connect an antenna with minimum number of external components
- Supports ISO/IEC 14443A / MIFARE®
- Typical operating distance in Reader/Writer mode for communication to a ISO/IEC 14443A / MIFARE® up to 50 mm depending on the antenna size and tuning
- Supports MIFARE® Classic encryption in Reader/Writer mode
- Supports ISO/IEC 14443A higher transfer speed communication up to 848 kbit/s
- Support of the MFIN / MFOUT
- Additional power supply to directly supply the smart card IC connected via MFIN / MFOUT
- Supported host interfaces

#### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

## Lampiran 6-Datasheet LCD

This is I2C interface 16x2 LCD display module, a high-quality 2 line 16 character LCD module with on-board contrast control adjustment, backlight and I2C communication interface. For Arduino beginners, no more cumbersome and complex LCD driver circuit connection. The real significance advantages of this I2C Serial LCD module will simplify the circuit connection, save some I/O pins on Arduino board, simplified firmware development with widely available Arduino library.



SKU: [DSP-1182](#)

**Brief Data:**

- Compatible with Arduino Board or other controller board with I2C bus.
- Display Type: Negative white on Blue backlight.
- I2C Address:0x38-0x3F (0x3F default)
- Supply voltage: 5V
- Interface: I2C to 4bits LCD data and control lines.
- Contrast Adjustment: built-in Potentiometer.
- Backlight Control: Firmware or jumper wire.
- Board Size: 80x36 mm.

**Setting Up:**

Hitachi's HD44780 based character LCD are very cheap and widely available, and is an essential part for any project that displays information. Using the LCD piggy-back board, desired data can be displayed on the LCD through the I2C bus. In principle, such backpacks are built around PCF8574 (from NXP) which is a general purpose bidirectional 8 bit I/O port expander that uses the I2C protocol. The PCF8574 is a silicon CMOS circuit provides general purpose remote I/O expansion (an 8-bit quasi-bidirectional) for most microcontroller families via the two-line bidirectional bus (I2C-bus). Note that most piggy-back modules are centered around PCF8574T (SO16 package of PCF8574 in DIP16 package) with a default slave address of 0x27. If your piggy-back board holds a PCF8574AT chip, then the default slave address will change to 0x3F. In short, if the piggy-back board is based on PCF8574T and the address connections (A0-A1-A2) are not bridged with solder it will have the slave address 0x27.



Address selection pads in the I2C-to-LCD piggy-back board.

## Lampiran 7 Datasheet LCD I2C

Table 5. PCF8574A address map

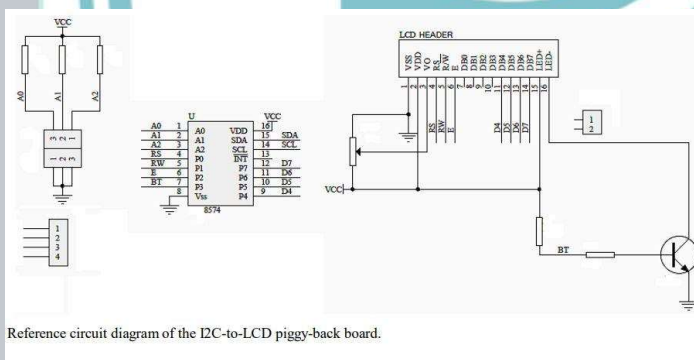
Pin connectivity			Address of PCF8574A							Address byte value		7-bit hexadecimal address without R/W	
A2	A1	A0	A6	A5	A4	A3	A2	A1	A0	R/W	Write		Read
V <sub>SS</sub>	V <sub>SS</sub>	V <sub>SS</sub>	0	1	1	1	0	0	0	-	70h	71h	38h
V <sub>SS</sub>	V <sub>SS</sub>	V <sub>DD</sub>	0	1	1	1	0	0	1	-	72h	73h	39h
V <sub>SS</sub>	V <sub>DD</sub>	V <sub>SS</sub>	0	1	1	1	0	1	0	-	74h	75h	3Ah
V <sub>SS</sub>	V <sub>DD</sub>	V <sub>DD</sub>	0	1	1	1	0	1	1	-	76h	77h	3Bh
V <sub>DD</sub>	V <sub>SS</sub>	V <sub>SS</sub>	0	1	1	1	1	0	0	-	78h	79h	3Ch
V <sub>DD</sub>	V <sub>SS</sub>	V <sub>DD</sub>	0	1	1	1	1	0	1	-	7Ah	7Bh	3Dh
V <sub>DD</sub>	V <sub>DD</sub>	V <sub>SS</sub>	0	1	1	1	1	1	0	-	7Ch	7Dh	3Eh
V <sub>DD</sub>	V <sub>DD</sub>	V <sub>DD</sub>	0	1	1	1	1	1	1	-	7Eh	7Fh	3Fh

Address Setting of PCD8574A (extract from PCF8574A data specs).

*Note: When the pad A0~A2 is open, the pin is pull up to VDD. When the pin is solder shorted, it is pull down to VSS.*

*The default setting of this module is A0~A2 all open, so is pull up to VDD. The address is 3Fh in this case.*

Reference circuit diagram of an Arduino-compatible LCD backpack is shown below. What follows next is information on how to use one of these inexpensive backpacks to interface with a microcontroller in ways it was exactly intended.



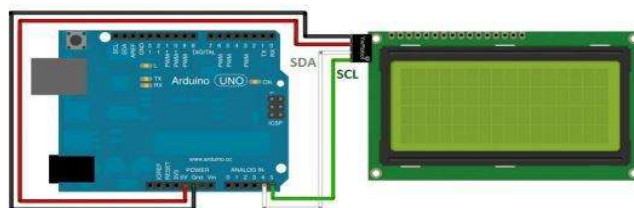
Reference circuit diagram of the I2C-to-LCD piggy-back board.

### I2C LCD Display.

At first you need to solder the I2C-to-LCD piggy-back board to the 16-pins LCD module. Ensure that the I2C-to-LCD piggy-back board pins are straight and fit in the LCD module, then solder in the first pin while keeping the I2C-to-LCD piggy-back board in the same plane with the LCD module. Once you have finished the soldering work, get four jumper wires and connect the LCD module to your Arduino as per the instruction given below.



LCD display to Arduino wiring.



- Hak Cipta :**
- Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
    - Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
    - Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

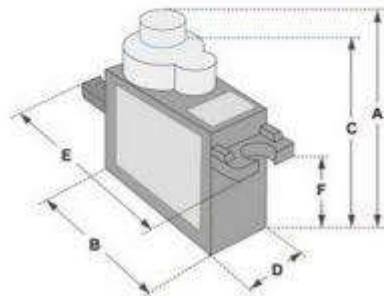
**Lampiran 8-Datasheet Servo SG90**

**SERVO MOTOR SG90**

**DATA SHEET**



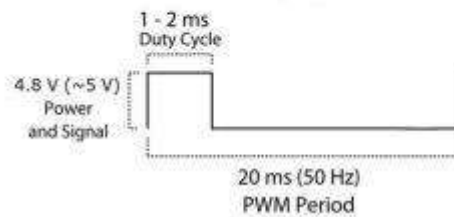
Tiny and lightweight with high output power. Servo can rotate approximately 180 degrees (90 in each direction), and works just like the standard kinds but smaller. You can use any servo code, hardware or library to control these servos. Good for beginners who want to make stuff move without building a motor controller with feedback & gear box, especially since it will fit in small places. It comes with a 3 horns (arms) and hardware.



Dimensions & Specifications	
A (mm) :	32
B (mm) :	23
C (mm) :	28.5
D (mm) :	12
E (mm) :	32
F (mm) :	19.5
Speed (sec) :	0.1
Torque (kg-cm) :	2.5
Weight (g) :	14.7
Voltage :	4.8 - 6

Position "0" (1.5 ms pulse) is middle, "90" (~2ms pulse) is middle, is all the way to the right, "-90" (~1ms pulse) is all the way to the left.

PWM=Orange (⌋⌋)  
 Vcc = Red (+)  
 Ground=Brown (-)



**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



**DESCRIPTION**

Solenoids are basically electromagnets; they are made of a big coil of copper wire with an armature (a slug of metal) in the middle. When the coil is energized, the slug is pulled into the center of the coil. This makes the solenoid able to pull from one end.

This solenoid in particular is nice and strong, and has a slug with a slanted cut and a good mounting bracket. It's basically an electronic lock, designed for a basic cabinet or safe or door. Normally the lock is active so you can't open the door because the solenoid slug is in the way. [Downloaded from Arrow.com](#)

does not use any power in this state. When 9-12VDC is applied, the slug pulls in so it doesn't stick out anymore and the door can be opened.

The solenoids come with the slanted slug as shown above, but you can open it with the two Phillips-head screws and turn it around so its rotated 90, 180 or 270 degrees so that it matches the door you want to use it with.

To drive a solenoid you will a power transistor and a diode, check this diagram for how to wire it to an Arduino or other microcontroller. You will need a fairly good power supply to drive a solenoid, as a lot of current will rush into the solenoid to charge up the electro-magnet, about 500mA, so don't try to power it with a 9V battery!



Lock-style Solenoid - 12VDC (8:17)

**TECHNICAL DETAILS**

- 12VDC (you can use 9-12 DC volts, but lower voltage results in weaker/slower operation)
- Draws 650mA at 12V, 500 mA at 9V when activated
- Designed for 1-10 seconds long activation time
- Max Dimensions: 41.85mm / 1.64" x 53.57mm / 2.1" x 27.59mm / 1.08"
- Dimensions: 23.57mm / 0.92" x 67.47mm / 2.65" x 27.59mm / 1.08"
- Wire length: 222.25mm / 8.75"
- Weight: 147.71g
- [Diagram](#)

**Lampiran 9-Datasheet Solenoid Doorlock**







```
import vlc
import threading
import os
import signal
import RPi.GPIO as GPIO
import lcdDriver
import time
import yaml
from dotmap import DotMap
from mfrc522 import SimpleMFRC522
import lcdDriver
import requests

exit_event = threading.Event()

#__location__ = Untuk membaca folder yang digunakan
sekarang
__location__ =
os.path.realpath(os.path.join(os.getcwd(),
os.path.dirname(__file__)))
os.chdir(__location__) #chdir (Change Directory) :
Kita pindah directory / folder ke __location__ saat
ini
# __location__ /home/pi/workspace/kicau_burung

#Dengan membuka file config.yml, simpan sebagai
variable config, lalu load menggunakan yaml.safe_load
dan simpen kedalam variable cfg
with open("config.yml", 'r') as config: cfg =
yaml.safe_load(config)

#DotMap itu adalah untuk konversi dari dictionary /
[''] ke dot (.)
#Misal config['Delay']['OpenDoor'] Menjadi
config.DELAY.OpenDoor
config = DotMap(cfg)

#Load file audio/kicau ke variable media menggunakan
vlc
media =
vlc.MediaPlayer(f"{__location__}/audio/kicau.mp3")

GPIO.setmode(GPIO.BCM)

LCD = None
btnHandling = None
# Stop Musik > 11 (GPIO 17)
```

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
# Stop Musik > 11 (GPIO 17)
# Play Musik > 12 (GPIO 27)
# Beri Makan > 37 (GPIO 26)
# (BUTTON) Beri Makan > 16 (GPIO 23)

#Class adalah kumpulan dari function yang memiliki 1
bagian yg sama

class databaseHandling:
    def __init__(self, host): #Inisialisasi class
        self.host = host

    def getStatus(self):
        try: #Jika terjadi error, maka program tidak
langsung keluar, karena sudah ada yang handle
error tersebut
            resp =
requests.get(f'{self.host}/getStatus').json()
            # {playMusic : 1, stopMusic : 1, feedBirds
: 1}
            if resp['feedBirds'] == 1:
btnHandling.feedBird()
            if resp['playMusic'] == 1:
btnHandling.playMusic()
            if resp['stopMusic'] == 1:
btnHandling.stopMusic()
        except Exception as e:
            print("Gagal mendapatkan status dari
Database")

#startListener
def startListener(self):
    #while variable_yang_benar:
    #    maka ulangi
    #while self.active:
    #    maka ulangi
    #while True:
    #    maka ulangi
    while True: #Ulangi loop secara terus menerus
        self.getStatus() #Jalankan getStatus
        time.sleep(config.DELAY.GetStatus)
#delay selama Delay.GetStatus yang ada di config.yml

#Class LCDHandling
class lcdHandling:
    #Inisialiasi LCD Handling
    def __init__(self):
        self.activate = True #flag activate = true
        trv:
```





### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
#Class LCDHandling
class lcdHandling:
    #Inisialiasi LCD Handling
    def __init__(self):
        self.activate = True #flag activate = true
        try:
            self.lcd = lcdDriver.lcd() # LCD Gagal
Load
        except Exception as e:
            self.activate = False #Ubah flag active
            print("Gagal terhubung ke LCD")

    def print(self, text, line):
        if not self.activate: return
        try:
            lcd.lcd_display_string(text, line)
        except Exception as e:
            print("Gagal menulis ke LCD")

#rfidHandling
class rfidHandling:
    #Inisialisasi, butuh parameter doorSolenoid (Pin
DoorSolenoid)
    def __init__(self, doorSolenoid):
        self.doorSolenoid = doorSolenoid
        GPIO.setup(self.doorSolenoid, GPIO.OUT) #GPIO
tetapkan bahwa pin doorSolenoid ini adalah OUTPUT
        GPIO.output(self.doorSolenoid, GPIO.LOW)
#Berikan tegangan LOW untuk pin doorSolenoid
        self.scanner = SimpleMFRC522() #self.scanner
= Inisiliasi class untuk membaca MFC522 menggunakan
library SimpleMFC522
        self.active = True # bikin variable
self.active = True
        self.id = '' #Inisialisasi variable self.id
untuk menampung hasil baca dari RFID

    def openDoor(self, id):
        # print("Opening Solenoid")
        LCD.print("Pintu Terbuka ", 1) #Print "Pintu
Terbuka" di Line 1
        LCD.print(f"{id:13} OK", 2) #{id:13} print
dengan format, id diberikan kuota sebanyak 13
character

        GPIO.output(self.doorSolenoid,GPIO.HIGH)
#Beri signal GPIO untuk membuka doorSolenoid

    def closeDoor(self, id):
```



### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
LCD.print(f"{id:13} XX", 2)
GPIO.output(self.doorSolenoid,GPIO.LOW)

def stopReader(self):
    print("Stop Reader")
    self.active = False
    None

def startReader(self):
    print("Start Reader")
    self.active = True
    while self.active:
        try:
            print("Reading ...")
            id = ''
            id = self.scanner.read_id()
            print("TAG Kartu")
            print(f"ID: {id}")
            if id in config.ACCEPTED_RFID: #Jika
id terdapat di array config.ACCEPTED_RFID
                print("RFID Terdaftar")
                self.openDoor(id)

time.sleep(config.DELAY.OpenDoor)
                self.closeDoor(id)
            else: #Jika if sebelumnya tidak benar
(id tidak terdapat di config.ACCEPTED_RFID)
                print("RFID tidak Terdaftar")
                self.closeDoor(id)
                time.sleep(0.5)
                print()
        except Exception as e:
            print(e)
            os._exit()

#buttonHandling(config.PIN.BUTTON.playMusic,
config.PIN.BUTTON.stopMusic,
config.PIN.BUTTON.feedBirds, config.PIN.SERVO)
class buttonHandling:
    def __init__(self, playBtn, stopBtn, feedBtn,
feedServo):
        self.playBtn = playBtn
        self.stopBtn = stopBtn
        self.feedBtn = feedBtn

        self.feedServo = feedServo
        self.pwm = None
        self.initializeGPIO()
```



### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
def stopMusic(self):
    print("Stop Music")
    media.stop()

def feedBird(self):
    print("Feed the birds")
    self.pwm.ChangeDutyCycle(7.5)
    time.sleep(0.5)
    self.pwm.ChangeDutyCycle(2.5)

def cb_playMusic(self, channel):
    self.playMusic()

def cb_stopMusic(self, channel):
    self.stopMusic()

def cb_feedBird(self, channel):
    self.feedBird()

def initializeGPIO(self):
    btn = [self.playBtn, self.stopBtn,
self.feedBtn] #Daftarkan semua button kedalam
variable btn
    for x in btn: #setiap variable di dalam array
Btn, simpan sebagai x dan ulangi
        GPIO.setup(x, GPIO.IN,
pull_up_down=GPIO.PUD_UP) #Tetapkan pin dari x (btn)
sebagai INPUT, DAN Aktifkan pull up dari raspberry
        GPIO.setup(self.feedServo, GPIO.OUT)
#Tetapkan pin feedServo sebagai OUTPUT
        GPIO.output(self.feedServo, False) #Tidak
memberikan signal apapun

        GPIO.add_event_detect(self.playBtn,
GPIO.FALLING, callback=self.cb_playMusic,
bouncetime=200) #Bikin Event / Kondisi jika pin
playBtn JATUH (FALLING), maka panggil perintah
self.cb_playMusic,
        GPIO.add_event_detect(self.stopBtn,
GPIO.FALLING, callback=self.cb_stopMusic,
bouncetime=200)
        GPIO.add_event_detect(self.feedBtn,
GPIO.FALLING, callback=self.cb_feedBird,
bouncetime=200)

        self.pwm = GPIO.PWM(self.feedServo, 50) #
GPIO 17 for PWM with 50Hz #Pin feedServo adalah PWM,
dengan jarak gelombang 50Mhz
        self.pwm.start(2.5) # Initialization
```

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
def stopMusic(self):
    print("Stop Music")
    media.stop()

def feedBird(self):
    print("Feed the birds")
    self.pwm.ChangeDutyCycle(7.5)
    time.sleep(0.5)
    self.pwm.ChangeDutyCycle(2.5)

def cb_playMusic(self, channel):
    self.playMusic()

def cb_stopMusic(self, channel):
    self.stopMusic()

def cb_feedBird(self, channel):
    self.feedBird()

def initializeGPIO(self):
    btn = [self.playBtn, self.stopBtn,
self.feedBtn] #Daftarkan semua button kedalam
variable btn
    for x in btn: #setiap variable di dalam array
Btn, simpan sebagai x dan ulangi
        GPIO.setup(x, GPIO.IN,
pull_up_down=GPIO.PUD_UP) #Tetapkan pin dari x (btn)
sebagai INPUT, DAN Aktifkan pull up dari raspberry
        GPIO.setup(self.feedServo, GPIO.OUT)
#Tetapkan pin feedServo sebagai OUTPUT
        GPIO.output(self.feedServo, False) #Tidak
memberikan signal apapun

        GPIO.add_event_detect(self.playBtn,
GPIO.FALLING, callback=self.cb_playMusic,
bouncetime=200) #Bikin Event / Kondisi jika pin
playBtn JATUH (FALLING), maka panggil perintah
self.cb_playMusic,
        GPIO.add_event_detect(self.stopBtn,
GPIO.FALLING, callback=self.cb_stopMusic,
bouncetime=200)
        GPIO.add_event_detect(self.feedBtn,
GPIO.FALLING, callback=self.cb_feedBird,
bouncetime=200)

        self.pwm = GPIO.PWM(self.feedServo, 50) #
GPIO 17 for PWM with 50Hz #Pin feedServo adalah PWM,
dengan jarak gelombang 50Mhz
        self.pwm.start(2.5) # Initialization
```





### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
def signal_handler(signum, frame):
    exit_event.set()
    GPIO.cleanup() #GPIO Cleanup artinya membersihkan
    semua GPIO
    print("Exit . . . .")
    os._exit(0) #keluar dari aplikasi dengan return
    code 0

def main():
    global LCD, btnHandling
    LCD = lcdHandling()
    dbHandling = databaseHandling(config.HOST)
    btnHandling =
    buttonHandling(config.PIN.BUTTON.playMusic,
    config.PIN.BUTTON.stopMusic,
    config.PIN.BUTTON.feedBirds, config.PIN.SERVO)

    threading.Thread(target=dbHandling.startListener).st
    art()
    rfidHandling(config.PIN.SELENOID).startReader()

    if __name__ == '__main__':
        signal.signal(signal.SIGINT, signal_handler)
        #Untuk menhandle jika sistem memberikan signal
        INTERUPT # SIGNAL.INTERUMPT ( Ctrl + C )
        try:
            main()
        finally:
            GPIO.cleanup()
            print("Cleaning GPIO")
```

NEGERI  
JAKARTA