



TUGAS AKHIR - TE 141599

**OTOMATISASI PELAYANAN BINATU BERBASIS
RASPBERRY PI UNTUK MENINGKATKAN
EFEKTIVITAS DAN EFISIENSI KEGIATAN
OPERASIONAL DAN PELAYANAN BINATU**

AL HABSYI YESA
NRP 2212 100 197

Dosen Pembimbing
Dr. Muhammad Rivai, ST, MT.
Ir. Tasripan, M.T.

JURUSAN TEKNIK ELEKTRO
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember
Surabaya 2016



FINAL PROJECT - TE141599

**LAUNDRY SERVICE AUTOMATION BASED ON
RASPBERRY PI TO INCREASE EFFECTIVITY
AND EFFICIENCY OF OPERATIONAL ACTIVITY
AND LAUNDRY SERVICE**

**AL HABSUYI YESA
NRP 2212 100 197**

**Supervisor
Dr. Muhammad Rivai, ST, MT.
Ir. Tasripan, M.T.**

**DEPARTMENT OF ELECTRICAL ENGINEERING
Faculty of Industrial Technology
Sepuluh Nopember Institute of Technology
Surabaya 2016**

OTOMATISASI PELAYANAN BINATU BERBASIS RASPERRY PI UNTUK MENINGKATKAN EFEKTIVITAS DAN EFISIENSI KEGIATAN OPERASIONAL DAN PELAYANAN BINATU

Al Habsyi Yesa
2212100197

Dosen Pembimbing I : Dr. Muhammad Rivai, ST, MT.
Dosen Pembimbing II : Ir. Tasripan, M.T.

Abstrak:

Bisnis binatu (*laundry*) atau bisnis jasa cuci pakaian merupakan bisnis yang menggiurkan saat sekarang ini. Proses layanan binatu umumnya terdiri dari pendaftaran, pengolahan, dan pengambilan pakaian. Proses ini memiliki banyak kekurangan yaitu pendaftaran manual tidak efektif dan efisien, pengolahan pakaian hanya diketahui oleh penyedia layanan binatu, dan pelanggan tidak mengetahui pakaian telah selesai. Ditengah persaingan bisnis yang begitu ketat, pelayanan konsumen merupakan suatu hal yang sangat penting. Penyedia jasa dituntut untuk berinovasi untuk menyediakan layanan jasa yang efektif dan efisien. Oleh karena itu pada tugas akhir ini merancang dan merealisasikan otomatisasi pelayanan binatu berbasis *raspberry pi*. Sistem ini menggunakan timbangan digital berbasis *load cell*, teknologi *radio frequency identification* (RFID) sebagai masukan data otomatis, *Raspberry Pi* sebagai pusat dari pengolahan basis data (*database*), dan *webserver* sebagai pusat informasi bagi pelanggan. Hasil penelitian menghasilkan beberapa kesimpulan. Timbangan digital memiliki eror rata-rata 0,88 %, metode RFID dapat memasukan data pelanggan secara otomatis dengan metode pengambilan nomor identitas jenis *hex 8 digit*, *Raspberry Pi* dapat mengakomodasi aplikasi yang efektif dan efisien untuk menyimpan data pelanggan secara otomatis dengan pemakaian CPU rata-rata 5%, dan *webserver* yang digunakan dapat menyediakan informasi layanan binatu bagi pelanggan.

Kata kunci: *Binatu, Efektif, Efisien, Load Cell, RFID, Raspberry Pi, Webserver,*

***LAUNDRY SERVICE AUTOMATION BASED ON RASPBERRY
PI TO INCREASE EFFECTIVITY AND EFFICIENCY OF
OPERATIONAL ACTIVITY AND LAUNDRY SERVICE***

Al Habsyi Yesa
2212100197

Supervisor I : Dr. Muhammad Rivai, ST, MT.
Supervisor II : Ir. Tasripan, M.T.

Abstract:

Nowadays, laundry business is a very prospective business. Laundry service processes generally consist of registration, clothing processing, and pick up clothes. This process has many lacks, that are manual registration is not effective and efficient, clothing processing only known by the laundry service, and customer does not know what clothes has been completed. Due to business competition is so tight, service to customers is a very important thing. Laundry service must innovate to provide effective and efficient services. Therefore, the final project researched laundry automation based raspberry pi. This system is expected to improve the effectiveness and efficiency of the service. The system uses digital scales based load cell, radio frequency identification technology (RFID) as an automatic data input, raspberry pi as a center of processing database and webserver as an information center for customers. The test results produced some conclusions. Digital scales have an average error of 0.88%, RFID method can include customer data automatically with the method of taking the identity number 8 digit hex types, Raspberry Pi can accommodate applications that are effective and efficient to store customer data automatically with an average CPU utilization of 5%, and webserver used can provide laundry service for customer information.

Keyword: Effective, Efficient, Laundry, Load Cell, RFID, Raspberry Pi, Webserver

DAFTAR ISI

HALAMAN JUDUL	
LEMBAR PERNYATAAN	
LEMBAR PENGESAHAN	
ABSTRAK	i
<i>ABSTRACT</i>	iii
KATA PENGANTAR	v
DAFTAR ISI	vii
DAFTAR GAMBAR	ix
DAFTAR TABEL	xi
PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Perumusan Masalah	2
1.3 Tujuan Penelitian	2
1.4 Batasan Masalah	2
1.5 Metodologi Penelitian	3
1.6 Sistematika Penulisan	3
1.7 Relevansi	4
TEORI PENUNJANG	5
2.1 Bisnis Binatu (<i>Laundry</i>)	5
2.2 <i>Load Cell</i>	6
2.2.1 Prinsip Kerja Dan Jenis-Jenis <i>Load Cell</i>	6
2.2.2 <i>Load Cell</i> YZC-1B	8
2.3 Modul HX711	9
2.3.1 IC HX711	10
2.3.2 Masukan Analog HX711	10
2.3.3 <i>Power Supply</i> HX711	10
2.3.4 <i>Clock Source</i> pada HX711	12
2.3.5 Format dan Nilai Data <i>Output</i>	12
2.3.6 <i>Serial Interface</i> pada HX711	12
2.4 <i>Radio Frequency Identification (RFID)</i>	13
2.4.1 Prinsip Kerja RFID	13
2.4.2 Karakteristik Teknologi RFID	14
2.4.3 Jenis-jenis RFID	14
2.4.4 Modul RFID MFRC522	16
2.5 <i>Arduino Uno R3</i>	17
2.6 Komunikasi Serial dengan <i>USB Port</i>	17

2.7	<i>Raspberry Pi</i>	18
2.8	<i>Lazarus</i>	20
	2.8.1 <i>Interface Lazarus</i>	21
	2.8.2 <i>Jendela Object Inspector</i>	22
	2.8.3 <i>Jendela Form Designer</i>	23
	2.8.4 <i>Jendela Komponen Palette</i>	24
	2.8.5 <i>Jendela Source Code Editor</i>	24
	2.8.6 <i>Jendela Messages</i>	24
2.9	<i>File CSV (Comma Separated Value)</i>	25
2.10	<i>XAMPP (Apache, MySQL, PHP)</i>	26
	2.10.1 <i>Apache</i>	26
	2.10.2 <i>Personal Home Page (PHP)</i>	26
	2.10.3 <i>MySQL</i>	27
	2.10.4 <i>PHPMyAdmin</i>	27
	PERANCANGAN SISTEM	29
3.1	<i>Diagram Blok Sistem</i>	29
3.2	<i>Perancangan Perangkat Keras</i>	30
3.3	<i>Perancangan Elektrik</i>	32
3.4	<i>Perancangan Perangkat Lunak (Software)</i>	35
	3.4.1 <i>Program Arduino</i>	35
	3.4.2 <i>Komunikasi Serial Arduino dengan Raspberry Pi</i> ...	36
	3.4.3 <i>Program Raspberry Pi</i>	38
	PENGUJIAN DAN ANALISIS	43
4.1	<i>Pengujian Perangkat Keras</i>	43
	4.1.1 <i>Pengujian Timbangan Digital</i>	43
	4.1.2 <i>Pengujian Modul RFID MFRC522</i>	47
4.2	<i>Pengujian Perangkat Lunak</i>	50
	4.2.1 <i>Pengujian Komunikasi Serial</i>	50
	4.2.2 <i>Pengujian Input Data</i>	52
	4.2.3 <i>Pengujian Simpan Data Ke Database (Export)</i>	54
	4.2.4 <i>Pengujian Pemanggilan Database (Import)</i>	55
	4.2.5 <i>Pengujian Webserver</i>	56
	PENUTUP	59
5.1	<i>Kesimpulan</i>	59
5.2	<i>Saran</i>	59
	DAFTAR PUSTAKA	61
	BIODATA PENULIS	63

DAFTAR TABEL

Tabel 2. 1 Spesifikasi YZC-1B.....	8
Tabel 2. 2 Deskripsi Pin HX711	11
Tabel 2. 3 Pemilihan input dan penguatan pada HX711	12
Tabel 2. 4 Perbandingan fitur dari empat jenis RFID	15
Tabel 2. 5 Spesifikasi <i>Arduino Uno</i>	17
Tabel 2. 6 Keuntungan dan Kelemahan <i>USB Interfacing</i>	18
Tabel 2. 7 Spesifikasi <i>Raspberry Pi 3</i>	19
Tabel 2. 8 Konektor pada <i>Raspberry Pi 3</i>	20
Tabel 3. 1 Konfigurasi Pin <i>Arduino</i>	35
Tabel 3. 2 Konfigurasi Komunikasi Serial	37
Tabel 4. 1 Tabel Pengujian Beban	44
Tabel 4. 2 Tabel <i>Zero Factor</i> Dan <i>Calibration Factor</i>	46
Tabel 4. 3 Pengujian Timbangan Digital	46
Tabel 4. 4 Hasil Pengujian Jarak Optimum Pembacaan RFID	49
Tabel 4. 5 Data NUID.....	50
Tabel 4. 6 Konfigurasi Komunikasi Serial	51
Tabel 4. 7 Pengujian pemakaian CPU <i>Raspberry Pi</i>	54

DAFTAR GAMBAR

Gambar 2. 1 Pemasangan <i>Strain Gauge</i> pada <i>Load Cell</i>	6
Gambar 2. 2 Jembatan <i>Wheatstone</i>	7
Gambar 2. 3 <i>Canister</i> dan <i>Single Ended Beam Load Cell</i>	7
Gambar 2. 4 Dimensi YZC-1B.....	9
Gambar 2. 5 Diagram blok Modul HX711.....	9
Gambar 2. 6 Deskripsi Pin IC HX711.....	10
Gambar 2. 7 Proses <i>serial interface</i> pada HX711.....	13
Gambar 2. 8 Blok Diagram MFRC522.....	16
Gambar 2. 9 <i>Raspberry Pi</i> 3 Model B.....	18
Gambar 2. 10 Komponen pada <i>Raspberry Pi</i> 3.....	19
Gambar 2. 11 Tampilan <i>Lazarus</i>	22
Gambar 2. 12 Jendela <i>Object Inspector</i>	23
Gambar 2. 13 Contoh Jendela <i>Form Designer</i>	24
Gambar 2. 14 Jendela Komponen <i>Palette</i>	24
Gambar 2. 15 Jendela <i>Source Code Editor</i>	25
Gambar 2. 16 Jendela <i>Messages</i>	25
Gambar 2. 17 <i>File CSV</i> pada <i>Excell</i>	26
Gambar 3. 1 Diagram Blok Perangkat Keras Sistem.....	30
Gambar 3. 2 Pengaturan Titik Beban <i>Load Cell</i>	31
Gambar 3. 3 Kabel <i>Load Cell</i>	32
Gambar 3. 4 Konfigurasi modul HX711 dengan <i>Arduino</i>	33
Gambar 3. 5 Konfigurasi MFR522 dengan <i>Arduino</i>	33
Gambar 3. 6 Rancangan Konfigurasi <i>Arduino</i>	34
Gambar 3. 7 <i>Raspbery Pi</i> dan Kelengkapannya.....	35
Gambar 3. 8 Diagram Alir Program <i>Arduino</i>	36
Gambar 3. 9 Diagram Alir Komunikasi Serial.....	37
Gambar 3. 10 Diagram Alir Aplikasi.....	38
Gambar 3. 11 Diagram Alir <i>Input Data</i> ke Tabel.....	39
Gambar 3. 12 Diagram Alir Pemanggilan <i>Database</i>	40
Gambar 3. 13 Diagram Alir Menyimpan ke <i>Database</i>	41
Gambar 3. 14 Diagram Alir <i>Websserver</i>	42
Gambar 4. 1 Realisasi Timbangan Digital.....	43
Gambar 4. 2 Grafik Beban vs Tegangan Analog.....	44
Gambar 4. 3 Kalibrasi Nilai <i>Zero Factor</i>	45
Gambar 4. 4 Kalibrasi Nilai <i>Calibration Factor</i>	45

Gambar 4. 5 Beban vs eror pengukuran	47
Gambar 4. 6 Pengujian RFID	47
Gambar 4. 7 Hasil Pengujian MFRC522	48
Gambar 4. 8 Pengujian Jarak Pembacaan	48
Gambar 4. 9 Pengambilan Data NUID	49
Gambar 4. 10 Pengujian Komunikasi Serial	50
Gambar 4. 11 Pengujian USB <i>Arduino</i> pada <i>Raspberry Pi</i>	51
Gambar 4. 12 Tombol <i>Start</i>	52
Gambar 4. 13 Hasil Pengujian Komunikasi Serial pada <i>Raspberry Pi</i>	52
Gambar 4. 14 Pengujian <i>Input Data</i>	53
Gambar 4. 15 Hasil Pengujian <i>Input Data</i>	53
Gambar 4. 16 Hasil Pengujian Simpan Data ke <i>Database</i>	54
Gambar 4. 17 <i>Customer Data</i> dan <i>Database</i>	55
Gambar 4. 18 Pengujian Pemanggilan <i>Database (Import)</i>	55
Gambar 4. 19 Halaman Utama <i>Webserver</i>	56
Gambar 4. 20 Halaman Ketika Nama Pelanggan Setelah Dimasukkan	57

BAB I

PENDAHULUAN

1.1 Latar Belakang

Bisnis binatu (*laundry*) atau bisnis jasa cuci pakaian merupakan suatu aspek bisnis yang sangat menggiurkan saat sekarang ini. Proses layanan binatu dibagi beberapa tahap yaitu tahap pendaftaran cucian, pemrosesan pakaian, dan pengambilan pakaian. Umumnya pendaftaran cucian dilakukan dengan cara manual. Pelanggan mengukur kuantitas pakaian. Kemudian petugas mencatat data yang terdiri dari kuantitas pakaian dan nama pelanggan. Lalu pakaian akan diproses oleh penyedia jasa binatu. Proses tersebut berupa cuci, jemur, dan setrika menurut permintaan pelanggan. Lalu pengambilan pakaian akan dilakukan oleh pelanggan beberapa hari kemudian [1].

Terdapat beberapa kekurangan yang ditemukan dari sistem layanan binatu model lama. Pertama, sistem pendaftaran manual tidak efektif dan efisien bagi pelanggan dan penyedia jasa. Pelanggan akan kehilangan waktu yang relatif banyak. Mulai dari menyebutkan nama, pencatatan tanggal sampai pengambilan bukti (*receipt*). Bagi penyedia jasa, cara manual ini juga akan menyulitkan untuk melakukan pendataan dan pembukuan. Kedua, bagian pemrosesan pakaian hanya diketahui oleh penyedia jasa saja. Pelanggan tidak mengetahui bagaimana kondisi pakaian. Dengan demikian pelanggan tidak bisa memperkirakan waktu proses layanan binatu akan selesai. Ketiga, pelanggan tidak dapat mengetahui dengan pasti waktu selesainya proses binatu. Hal tersebut membuat pelanggan tidak bisa menentukan pengambilan pakaian dengan sendirinya.

Ditengah persaingan bisnis yang begitu ketat, pelayanan terhadap konsumen merupakan suatu hal yang sangat penting. Penyedia jasa dituntut untuk berinovasi untuk menyediakan layanan jasa yang efektif dan efisien. Oleh karena itu, diajukanlah proposal tugas akhir berupa otomatisasi pelayanan binatu berbasis *Raspberry Pi*. Sistem ini dilengkapi dengan timbangan digital berbasis sensor *load cell* dan teknologi *radio frequency identification* (RFID) untuk proses pendaftaran binatu. Sistem ini dilengkapi dengan aplikasi dan basis data (*database*) berbasis *Raspberry Pi*. Untuk penyedia informasi bagi pelanggan, sistem ini dilengkapi *webserver* yang dapat

diakses langsung oleh pelanggan. *Websserver* berfungsi sebagai penyedia informasi bagi pelanggan. Informasi tersebut berupa riwayat binatu yang dilakukan pelanggan. *Web* juga menyediakan informasi proses binatu (cuci, jemur, setrika, selesai) yang sedang dilakukan penyedia jasa binatu. Sistem ini diharapkan bisa meningkatkan efektivitas dan efisiensi pelayanan sehingga memberi kepuasan yang lebih terhadap konsumen.

1.2 Perumusan Masalah

Berdasarkan latar belakang di atas, dapat dirumuskan beberapa masalah, antara lain :

1. Bagaimana cara pengukuran kuantitas pakaian?
2. Bagaimana cara memasukkan data pelanggan secara otomatis?
3. Bagaimana mencatat dan menyimpan data secara otomatis?
4. Bagaimana membuat basis data (*database*) yang terintegrasi dengan aplikasi?
5. Bagaimana cara menampilkan informasi layanan binatu secara *online*?

1.3 Tujuan Penelitian

Penelitian pada tugas akhir ini bertujuan sebagai berikut :

1. Implementasi *load cell* pada timbangan digital untuk pengukuran kuantitas pakaian
2. Implementasi teknologi RFID untuk memasukkan data secara otomatis
3. Pembuatan aplikasi dengan *software Lazarus* untuk mencatat dan menyimpan data secara otomatis
4. Penggunaan basis data berbasis *comma separated value* (CSV) untuk mengintegrasikan data pada aplikasi dengan memori
5. Pembuatan *websserver* dengan software XAMPP untuk menampilkan informasi layanan binatu secara *online*.

1.4 Batasan Masalah

Batasan masalah dari tugas akhir ini adalah sebagai berikut:

1. Sensor *load cell* yang digunakan YZC-1B dengan beban maksimal 10 Kg.
2. RFID *reader* menggunakan MRCF522 dengan kartu RFID dengan frekuensi 13,56 MHz
3. Mikrokontroler pembantu menggunakan *Arduino Uno*

4. Jenis *Raspberry Pi* yang digunakan *Raspberry Pi 3*
5. *Software* yang digunakan adalah *Open Sorce Free Pascal Lazarus* versi 1.4.4
6. *Database* yang digunakan berformat *CSV(Comma Separated Value)*
7. *Websserver* menggunakan *software* XAMPP.

1.5 Metodologi Penelitian

Dalam penyelesaian tugas akhir ini digunakan metodologi sebagai berikut:

1. Studi literatur
Pada tahap ini dilakukan pengumpulan dasar teori yang menunjang dalam penulisan tugas akhir. Dasar teori ini dapat diambil dari buku-buku, jurnal, *proceeding*, dan artikel-artikel di internet.
2. Perancangan sistem
Setelah mempelajari literatur yang ada, selanjutnya akan dilakukan perancangan sistem. Perancangan sistem terbagi sebagai berikut:
 - a. Perancangan Perangkat Keras
 - b. Perancangan perangkat lunak
 - c. Pengujian sistem
 - d. Pengolahan data
 - e. Penulisan laporan Tugas Akhir
Tahap penulisan laporan Tugas Akhir dilakukan pada saat tahap pengujian sistem dimulai serta setelahnya.

1.6 Sistematika Penulisan

Laporan tugas akhir ini terdiri dari Lima Bab dengan sistematika penulisan sebagai berikut:

Bab 1 :PENDAHULUAN

Bab ini meliputi latar belakang, perumusan masalah, tujuan penelitian, batasan masalah, metodologi, sistematika penulisan, dan relevansi.

Bab 2 :DASAR TEORI

Bab ini menjelaskan tentang dasar-dasar teori yang dibutuhkan dalam pengerjaan tugas akhir ini, yang meliputi teori bisnis layanan binatu, *Load cell*, Modul HX711, *Radio Frequency Identification* (RFID), *Liquid Crystal Display* (LCD), *Arduino Uno*, Komunikasi Serial, *Raspberry Pi*, *Lazarus*, *File CSV*, dan XAMPP.

Bab 3: PERANCANGAN SISTEM

Bab ini menjelaskan tentang perencanaan, sistem elektrik, mekanik, serta perangkat lunak. Bab ini juga berisi menjelaskan tentang prosedur pengujian yang dilakukan dalam penelitian.

Bab 4 :PENGUKURAN DAN ANALISIS SISTEM

Bab ini menjelaskan tentang hasil yang didapat dari pengujian tiap blok sistem secara keseluruhan.

Bab 5 : PENUTUP

Bab ini menjelaskan tentang kesimpulan meliputi kekurangan-kekurangan pada kerja alat dari hasil analisa serta saran untuk pengembangan ke depan.

1.7 Relevansi

Hasil dari tugas akhir ini diharapkan dapat memberikan manfaat Sebagai inovasi teknologi bisnis layanan binatu yang dapat diterapkan pada kondisi sesungguhnya.

BAB II

TEORI PENUNJANG

2.1 Bisnis Binatu (*Laundry*)

Bisnis binatu merupakan usaha penyedia jasa layanan dalam menangani masalah pakaian kotor. Layanan binatu telah menjadi gaya hidup modern dari masyarakat kota. Hal ini menimbulkan usaha atau gerai binatu terus tumbuh setiap tahunnya. Dalam bahasa modern binatu lebih dikenal dengan istilah *laundry & dry clean*. Proses pakain dicuci dengan menggunakan mesin cuci. Pakaian dibersihkan dengan cairan kimia khusus yang bisa membersihkan dan merontokan kotoran di pakaian tanpa dicuci secara biasa.

Awal mulanya, jenis usaha ini hanya dilakukan secara rumahan atau hotel-hotel mewah untuk fasilitas tamunya. Sejak tahun 1990-an, sistem *franchise* (waralaba) bisnis binatu menjamur di luar negeri. Dalam kurun waktu tahun terakhir, bisnis binatu juga juga menjamur di Indonesia. Sistem yang dipakai menggunakan waralaba lokal dan sistem *agency*. Sistem ini dapat menyediakan layanan dengan harga terjangkau. Layanan yang tadinya hanya diperuntukkan bagi masyarakat kelas atas, sekarang bisa dinikmati masyarakat kelas menengah ke bawah. Saat sekarang ini kombinasi antara layanan murah dengan layanan binatu (cuci setrika) berkembang ke arah kreatif dengan munculnya binatu kiloan. Binatu kiloan yaitu binatu biasa dengan harga berdasarkan hitungan kilogram (bukan per potong pakaian) [1].

Umumnya proses binatu yang ada di Indonesia seperti berikut,

1. Pelanggan mendaftarkan pakaian kotor
2. Petugas mencatat data pelanggan yang terdiri dari nama pelanggan, berat, pilihan layanan, dan biaya
3. Setelah itu pakaian akan diproses sampai bersih dan rapi. Proses tersebut yaitu pencucian, pengeringan, setrika, dan pengemasan.
4. Pelanggan mengambil pakaian yang sudah selesai dibersihkan dan dirapikan.

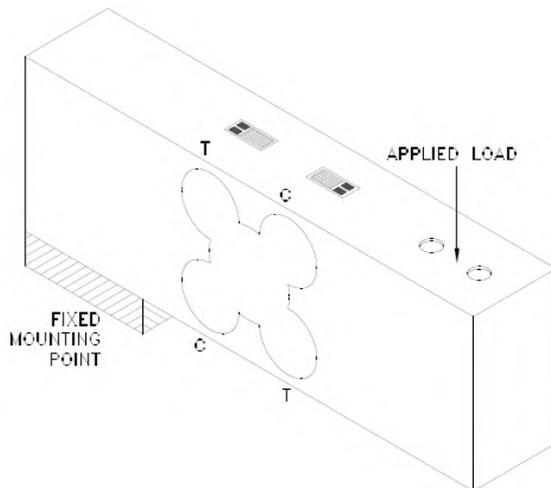
2.2 Load Cell

Load cell adalah sebuah sensor elektromekanik yang berfungsi untuk mengukur besarnya gaya statik maupun dinamik yang bekerja padanya.

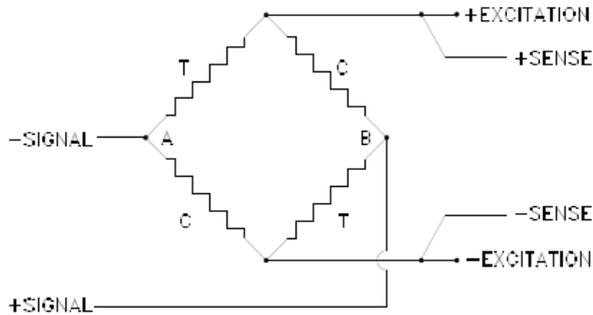
2.2.1 Prinsip Kerja Dan Jenis-Jenis Load Cell

Load Cell terdiri dari suatu bahan elastik yang akan mengalami deformasi sesuai dengan gaya yang diterimanya, besarnya deformasi ini sebanding dengan besarnya gaya. Untuk mengukur besarnya defleksi deformasi bahan tersebut, salah satu cara yang populer adalah menggunakan *strain gauge*.

Gambar 2.1 menunjukkan *load cell* yang menggunakan *strain gauge*. *Strain gauge* T dan C dipasang sejajar di bagian *load cell* yang akan mengalami deformasi. *Strain gauge* T dan C dihubungkan menjadi jembatan *wheatstone*.



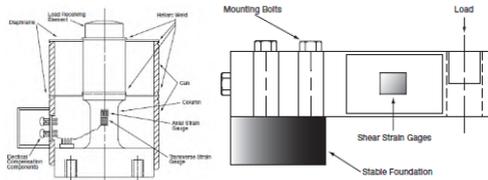
Gambar 2. 1 Pemasangan *Strain Gauge* pada *Load Cell*



Gambar 2. 2 Jembatan *Wheatstone*

Umumnya *load cell* diberi resistor tambahan untuk mengompensasi pengaruh temperatur pada kinerja *load cell*. Tegangan eksitasi di aplikasikan pada sudut jembatan *wheatstone* yang berlawanan dan sinyal *output* diukur melalui poin A dan B. Ketika tidak ada beban, semua *strain gauge* memiliki resistansi yang sama sehingga tidak ada perbedaan tegangan pada poin A dan B. Ketika beban ditambahkan, resistansi dari *strain gauge* akan berubah. Perubahan tersebut akan mengikuti rumus $C/T = T/C$. Hal ini akan menyebabkan ketidakseimbangan pada jembatan dan muncul perbedaan tegangan pada poin A dan B. Titik A dan B ini akan menjadi sinyal yang akan diolah mikrokontroller untuk mendapatkan nilai beban yang sebenarnya [2].

Load cell memiliki banyak tipe yaitu *S-beam load cell*, *Canister load cell*, *Compression disk load cell*, *Rocker column load cell*, *Double-ended shear beam*, *single ended beam load cell*, dan *single point load cell* [3].



Gambar 2. 3 *Canister* dan *Single Ended Beam Load Cell*

2.2.2 Load Cell YZC-1B

Load cell YZC-1B merupakan sensor yang dipakai untuk timbangan digital. *Load cell* ini bertipe *single point* dan biasanya digunakan untuk pengukuran dengan komputer. Dimensi dan spesifikasi dari *load cell* tersebut dijelaskan pada gambar 2.4 dan tabel 2.1 [4].

Tabel 2.1 Spesifikasi YZC-1B

Aplikasi	Price computing scales, bench scale etc.
Model	YZC-1B
Kapasitas (Kg)	2,3,5,6,8,10,30,35,40,50
Akurasi	C2
Resistansi masukan (Ω)	402 \pm 6
Resistansi keluaran (Ω)	350 \pm 3
Error toral (%F.S)	$\leq \pm 0.030$
Resistansi insulasi (M Ω)	5000
Nilai keluaran (MV/V)	2.0 \pm 0.15
Tegangan Eksitasi (V)	10~15
Kisaran suhu operasi	-35~+80
Efek temperatur saat beban 0 (%F.S/)	0.003
<i>Creep</i> (%F.S/)	30min 0.03
Keluaran nol (%F.S)	± 1.0
Overload yang aman (%F.S)	150
Material <i>load cell</i>	Aluminum
Kabel penghubung	4.2*350mm
Metode menghubungkan kabel	Merah (<i>input +</i>),Hitam (<i>input -</i>), Hijau(<i>output +</i>),Putih (<i>output -</i>)

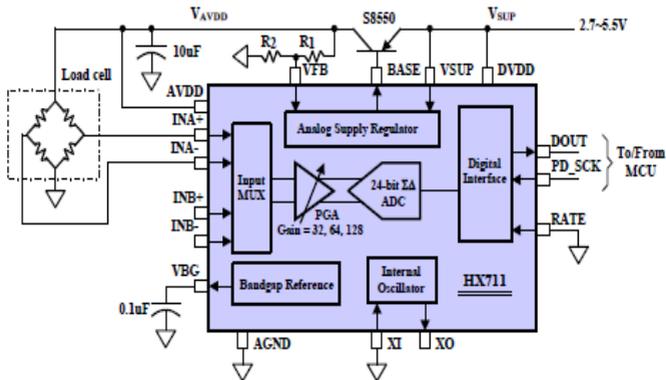


Gambar 2. 4 Dimensi YZC-1B

2.3 Modul HX711

Modul HX711 merupakan modul *analog-to-digital converter* (ADC) 24 bit yang digunakan untuk timbangan berat untuk dihubungkan langsung dengan sensor *bridge*. Dibandingkan dengan cip lainnya, modul HX711 memiliki banyak kelebihan yaitu ketahanan, respon cepat, integrasi tinggi, dan harga murah. Modul ini dapat meningkatkan performa dan reliabilitas timbangan berat [5].

Diagram blok dari modul HX711 untuk timbangan berat dijelaskan pada gambar 2.5 [5],



Gambar 2. 5 Diagram blok Modul HX711

2.3.1 IC HX711

Modul ini memiliki IC utama HX711, terdapat beberapa karakteristik khusus dari IC HX711. Deskripsi pin HX711 dijelaskan pada gambar 2.6 dan tabel 2.2.

2.3.2 Masukan Analog HX711

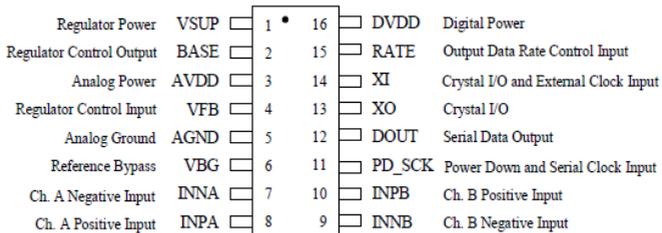
Masukan differensial *channel* A dirancang untuk dihubungkan langsung dengan sebuah keluaran differensial dari sensor jembatan *wheatstone*. Penguatan besar dibutuhkan untuk mengakomodasi sinyal keluaran yang kecil dari sensor. Ketika *supply* 5V digunakan untuk pin AVDD, penguatan tersebut akan menyesuaikan dengan tegangan masukan differensial *full-scale* sebesar $\pm 20\text{mV}$ atau $\pm 40\text{mV}$.

Masukan differensial *channel* B mempunyai penguatan (*gain*) tetap sebesar 32. Tegangan masukan *full-scale* berkisar $\pm 80\text{mV}$ ketika *supply* 5V digunakan pada pin AVDD [5].

2.3.3 Power Supply HX711

Digital power supply (DVDD) harus sama dengan *power supply* MCU (*Main Controll Unit*). Ketika menggunakan regulator *supply* analog internal, tegangan *drop* akan bergantung pada transistor eksternal yang digunakan. Tegangan keluaran yaitu $V_{\text{ADD}} = V_{\text{BG}} \cdot (R_1 + R_2) / R_1$. Tegangan ini harus dirancang dengan sebuah tegangan minimum VSUP dibawah 100mV.

Jika regulator *supply* analog tidak digunakan, pin VSUP harus dihubungkan antara AVDD atau DVDD tergantung pada tegangan mana yang lebih tinggi. Pin VFB harus dihubungkan dengan *Ground* dan pin *Base* menjadi NC [5].



Gambar 2. 6 Deskripsi Pin IC HX711[5]

Tabel 2. 2 Deskripsi Pin HX711 [5]

Pin #	Nama	Fungsi	Deskripsi
1	VSUP	<i>Power</i>	Regulator <i>supply</i> : 2.7 ~ 5.5V
2	BASE	<i>Analog Output</i>	Regulator <i>control output</i> (NC tidak digunakan)
3	AVDD	<i>Power</i>	<i>Supply</i> analog: 2.6 ~ 5.5V
4	VFB	<i>Analog Input</i>	Regulator <i>control input</i> (AGND tidak digunakan)
5	AGND	<i>Ground</i>	<i>Ground</i> analog
6	VBG	<i>Analog Output</i>	Referensi <i>bypass output</i>
7	INA-	<i>Analog Input</i>	Masukan negatif A
8	INA+	<i>Analog Input</i>	Masukan positif A
9	INB-	<i>Analog Input</i>	Masukan negatif B
10	INB+	<i>Analog Input</i>	Masukan positif B
11	PD_SCK	<i>Digital Input</i>	<i>Power down control (high active)</i> dan <i>serial clock input</i>
12	DOUT	<i>Digital Output</i>	Keluaran data serial
13	XO	<i>Digital I/O</i>	Crystal I/O (NC tidak digunakan)
14	XI	<i>Digital Input</i>	Crystal I/O atau <i>clock</i> ekstrnal masukan, 0: gunakan <i>on-chip oscillator</i>
15	RATE	<i>Digital Input</i>	<i>Output data rate control</i> , 0: 10Hz;1:80 Hz
16	DVDD	<i>Power</i>	<i>supply</i> digital: 2.6 ~ 5.5V

2.3.4 Clock Source pada HX711

Dengan menghubungkan pin XI ke *Ground*, osilator diaktifkan. Sebuah kristal dapat langsung dihubungkan dengan pin XI dan XO. Sebuah *clock* eksternal akan dihubungkan dengan pin XI melalui sepasang kapasitor 20pF. *Clock* eksternal tidak dibutuhkan dalam bentuk kotak. *Clock* dapat langsung dari pin kristal keluaran pada cip MCU dengan amplitudo sebesar 150mV. Ketika menggunakan kristal atau *clock* eksternal, osilator internal akan otomatis mati [5].

2.3.5 Format dan Nilai Data Output

Ketika menggunakan osilator internal pada cip, nilai data keluaran yaitu 10 ($RATE=0$) atau 80SPS ($RATE=1$). Ketika menggunakan kristal, nilai data keluaran akan langsung proporsional mengikuti frekuensi kristal. Jika menggunakan kristal 11.0592MHz akan menghasilkan nilai data keluaran tepat pada 10 ($RATE=0$) atau 80SPS($RATE=1$) [5].

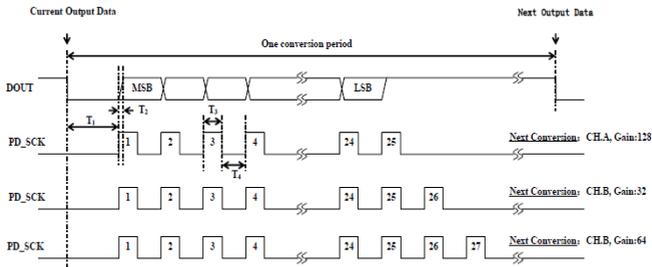
2.3.6 Serial Interface pada HX711

Pin PD_SCK dan DOUT digunakan untuk penerimaan data, pemilihan masukan, pemilihan penguatan dan pengaturan *power*.

Ketika data keluaran tidak siap diterima, pin digital keluaran DOUT adalah *high*. *Clock* serial pada PD_SCK harus *low*. Ketika DOUT *low*, data siap diterima. Dengan menambahkan 25-27 pulsa *clock* positif pada pin PD_SCK, data dipindahkan keluar dari pin DOUT. Setiap pulsa pada PD_SCK mengeluarkan satu bit dimulai dari MSB sampai 24 bit. Pulsa ke 25 pada PD_SCK akan membuat pin DOUT menjadi *high* kembali. Proses tersebut juga akan mengatur ulang kembali (*reset*) operasi pengiriman.

Tabel 2. 3 Pemilihan input dan penguatan pada HX711 [5]

Pulsa PD_SCK	<i>Input Channel</i>	Penguatan (<i>gain</i>)
25	A	128
26	B	32
27	A	64



Gambar 2. 7 Proses *serial interface* pada HX711 [5]

Pemilihan masukan dan penguatan dikendalikan oleh jumlah pulsa PD_SCK. Pulsa *clock* PD_SCK harus kurang dari 27 dan lebih besar dari 25 pada satu periode konversi [5].

2.4 Radio Frequency Identification (RFID)

RFID merupakan suatu teknologi yang memanfaatkan frekuensi radio sebagai identifikasi otomatis terhadap suatu objek. Teknologi RFID dapat digunakan secara elektronik untuk mengidentifikasi, melacak dan menyimpan informasi dalam tag RFID.

2.4.1 Prinsip Kerja RFID

RFID menggunakan teknologi identifikasi dengan gelombang radio. Untuk itu minimal dibutuhkan dua buah perangkat, yaitu yang disebut *Tag* dan *Reader*.

RFID *Tag* Adalah sebuah alat yang melekat pada obyek yang akan diidentifikasi oleh RFID *Reader*. RFID *Tag* dapat berupa perangkat pasif atau aktif. *Tag* pasif artinya tanpa baterai dan *Tag* aktif artinya menggunakan baterai. *Tag* pasif lebih banyak digunakan karena murah dan mempunyai ukuran lebih kecil. RFID *Tag* dapat berupa perangkat *read-only* yang berarti hanya dapat dibaca saja ataupun perangkat *read-write* yang berarti dapat dibaca dan ditulis ulang untuk *update*. RFID *Tag* mempunyai dua bagian penting, yaitu *Integrated Circuit* (IC) yang berfungsi menyimpan dan memproses informasi, modulasi dan demodulasi sinyal *radio frequency* (RF), mengambil tegangan DC yang dikirim dari RFID *Reader* melalui induksi, dan beberapa fungsi khusus lainnya. Bagian lainnya adalah

antena yang berfungsi menerima dan mengirim sinyal RF.

RFID *reader* adalah merupakan alat pembaca RFID *Tag*. Ada dua macam RFID *Reader* yaitu *Reader* Pasif (PRAT) dan *Reader* Aktif (ARPT). *Reader* Pasif memiliki sistem pembaca pasif yang hanya menerima sinyal radio dari RFID *Tag* Aktif (yang dioperasikan dengan baterai). Jangkauan penerima RFID Pasif bisa mencapai 600 meter. Hal ini memungkinkan aplikasi RFID untuk sistem perlindungan dan pengawasan aset. *Reader* Aktif memiliki sistem pembaca aktif yang memancarkan sinyal interogator ke *Tag* dan menerima balasan autentikasi dari *Tag*. Sinyal interogator ini juga menginduksi *Tag* dan akhirnya menjadi sinyal DC yang menjadi sumber daya *Tag* Pasif [6].

2.4.2 Karakteristik Teknologi RFID

Tidak seperti teknologi identifikasi otomatis lainnya (teknologi *bar code*, teknologi pengenalan optik, *voice recognition*, *biometrics*, kartu magnetik), RFID memiliki karakteristik [7]:

1. EEPROM merupakan *information carrier* yang dapat menyimpan data dengan kapasitas besar
2. *Read* dan *write* dengan kecepatan tinggi melalui komunikasi *wireless*
3. Identifikasi tanpa kontak langsung, pembacaan tidak dipengaruhi lokasi dan lingkungan
4. Memori mempunyai nomor serial yang unik untuk keperluan identifikasi
5. *Tag* bisa dibaca bersamaan dalam satu waktu.

2.4.3 Jenis-jenis RFID

Ada empat macam RFID tag yang sering digunakan bila dikategorikan berdasarkan frekuensi radio yaitu *Low frequency tag* (antara 30 ke 300 KHz), *High frequency tag* (3-30 MHz), *UHF tag* (100 sampai 968 MHz), dan *Microwave tag* (2.45-5.8 GHz) [7]. Perbandingan fitur dari 4 jenis RFID dijelaskan pada tabel 2.4.

Tabel 2. 4 Perbandingan fitur dari empat jenis RFID [7]

	<i>Low-frequency</i>	<i>High-frequency</i>	
Frekuensi	4-10 z	3-30 MHz	
Jarak Pembacaan	0-0.5m	ISO14443 <0.5m	ISO15693 <1.5m
Cip memori	64B-2KB	8KB-128KB	256-512B
<i>Transmit power</i>	72dB μ A/m	42 dB μ A/m	
Fitur	Tidak sensitif, baik untuk membaca data pada besi atau cairan, konsumsi daya yang rendah karena frekuensi clock yang rendah	Bisa digunakan pada lingkungan berair Frekuensi clock yang tinggi memungkinkan fungsi <i>password</i> , mikroprosesor. Menggunakan ISO14443A/B,ISO15693 2, ISO18000-3	

	<i>Ultra-high frequency</i>		<i>Micro-wave</i>	
Frekuensi	100-986 MHz		2.45-5.8 GHz	
Jarak Pembacaan	Aktif 3-10m	Pasif >10m	Aktif 3-10m	Pasif >10.
Cip memori	64-512B		16-64B	
<i>Transmit power</i>	10mW-4W		4W(USA)/500mW(Europe)	
Fitur	Tidak terbaca pada bahan metal dan lingkungan berair. Protokol <i>ISO18000-6 and EPC</i> . Pembacaan jarak jauh, kecepatan transfer tinggi, dan dapat membaca beberapa data secara simultan.		Tidak terbaca pada bahan metal dan lingkungan berair dikarenakan refleksi energi elektromagnetik. Ukuran kecil tapi harga mahal	

2.4.4 Modul RFID MFRC522

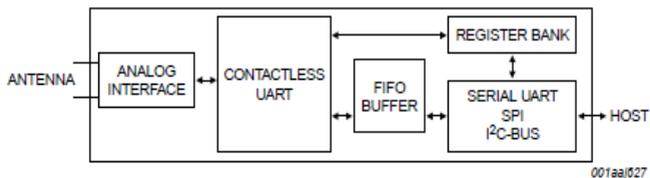
MFRC522 merupakan komponen terintegrasi dari IC *reader/writer* untuk komunikasi non kontak pada frekuensi 13.56MHz. MFRC522 reader mendukung ISO/IEC 14443 A/MIFARE dan NTAG

Pemancar internal MFRC522 mampu menjalankan antena *reader/writer* yang dirancang untuk berkomunikasi dengan kartu ISO/IEC 14443 A/MIFARE dan transponder tanpa sirkuit aktif tambahan. Modul penerima dapat memodulasi dan menguraikan sinyal dari ISO/IEC 14443 A/MIFARE dari kartu dan transponder yang kompatibel [8].

MFRC522 mendukung produk MF1xxS70 dan MF1xxS50. MFRC522 mendukung komunikasi non kontak. Kecepatan transfer MIFARE mencapai 848kBd pada dua arah. Berikut, host interface yang disediakan:

- *Serial Peripheral Interface (SPI)*
- Serial UART
- *I2C bus interface*

Blok diagram diri MFRC522 dijelaskan pada gambar 2.8. *Interface analog* menangani masalah modulasi dan demodulasi dari sinyal analog. *Contactless UART* menangani protokol yang dibutuhkan saat komunikasi dengan *host*. *FIFO buffer* memastikan pengiriman data yang cepat dan tepat ke dan dari *host* dan *contactless UART* dan *vice versa*. Berbagai *host interface* diimplementasikan untuk memenuhi kebutuhan pelanggan [8].



Gambar 2. 8 Blok Diagram MFRC522 [8]

2.5 *Arduino Uno R3*

Arduino Uno adalah sebuah *mini board* berbasis mikrokontroler Atmega328. *Arduino UNO* mempunyai 14 pin digital *input/output* (pin 0-13) yang terdiri dari 6 pin *input* analog (pin 0-5) yang biasa digunakan untuk membaca tegangan dari sensor dan mengkonversikannya menjadi nilai 0 dan 1023, 6 pin *output* analog (pin 3, 5, 6, 9, 10, 11) yang digunakan untuk pengaturan PWM (*Pulse Width Modulation*), sebuah osilator Kristal 16 MHz, sebuah koneksi USB, sebuah *power jack*, sebuah ICSP *header*, dan sebuah tombol *reset*. *Arduino UNO* dapat dioperasikan dengan menggunakan *port* USB komputer, USB *charger*, atau adaptor AC-DC dengan tegangan yang direkomendasikan 9 Volt [9]. Spesifikasi *Arduino Uno* dijelaskan pada tabel 2.5.

2.6 Komunikasi Serial dengan USB Port

Penggunaan serial dan paralel *port* sangat terbatas dalam hal transfer data, sementara perkembangan teknologi membutuhkan transfer data yang lebih cepat. *Universal Serial bus* (USB) adalah solusi untuk masalah ini. Dengan menggunakan USB, transfer data berlangsung jauh lebih cepat dibandingkan dengan penggunaan *port* tipe lainnya. Keuntungan dan kelemahan USB interfacing dijelaskan pada tabel 2.6 ,

Tabel 2. 5 Spesifikasi *Arduino Uno* [9]

Microcontroller	Atmega328
Operating Voltage	5 V
Input Voltage	7-12 V
Digital I/O Pins	14 (6 PWM Outputs)
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3 V pin	50 mA
Flash Memory	32 KB (Atmega328)
SRAM	2 KB (Atmega328)
EEPROM	1 KB (Atmega328)
Clock Speed	16 MHz
Length, Width	68.6 mm, 53.4 mm
Weigth	25 g

Tabel 2. 6 Keuntungan dan Kelemahan *USB Interfacing* [10]

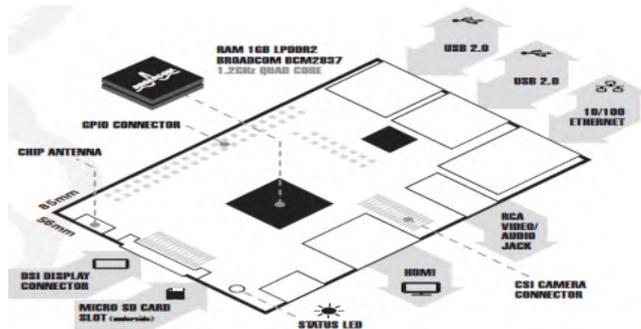
Keuntungan	Kelemahan
<ul style="list-style-type: none">- Mudah digunakan karena dapat terkonfigurasi secara otomatis oleh OS.- Dapat digunakan untuk banyak jenis <i>peripheral</i>.- Dapat dihubungkan dengan komputer dalam kondisi hidup. Sehingga tidak perlu mematikan komputer terlebih dahulu (<i>hot pluggable</i>)- Kecepatan yang berlipat. Dapat mencapai 5 Gbps untuk USB dengan empat bus.- Murah, dan mengurangi pemakaian energi.	<ul style="list-style-type: none">- Tidak dapat digunakan dalam komunikasi <i>peer to peer</i>.- Hanya efektif untuk jarak yang pendek. Dibutuhkan hub untuk jarak yang lebih jauh.- Tidak dapat digunakan untuk komputer-komputer tua karena ketidaktersediaan <i>port</i>.

2.7 Raspberry Pi

Raspberry Pi, sering juga disingkat dengan nama *Raspi*, adalah komputer papan tunggal (*Single Board Circuit /SBC*) yang memiliki ukuran sebesar kartu kredit. *Raspi* bisa digunakan untuk berbagai keperluan, seperti *spreadsheet*, *game*, dan lain-lain.



Gambar 2. 9 *Raspberry Pi 3 Model B* [11]



Gambar 2. 10 Komponen pada Raspberry Pi 3 [11]

Raspberry Pi yang digunakan pada tugas akhir ini adalah *Raspberry Pi 3* model B. *Raspberry Pi 3* memiliki prosesor yang lebih kuat dari generasi sebelumnya yaitu 10 kali lebih cepat dari generasi pertama. *Raspberry Pi* ini dilengkapi dengan fitur *Wireless LAN* dan koneksi *Bluetooth* yang tidak dimiliki dua generasi sebelumnya [11].

Tabel 2. 7 Spesifikasi *Raspberry Pi 3*[11]

Prosesor	<i>Broadcom BCM2387 chipset. 1.2GHz Quad-Core ARM Cortex-A53 802.11 b/g/n Wireless LAN and Bluetooth 4.1 (Bluetooth Classic and LE)</i>
GPU	<i>Dual Core VideoCore IV® Multimedia Co-Processor. Provides Open GL ES 2.0, hardware-accelerated OpenVG, and 1080p30 H.264 high-profile decode. Capable of 1Gpixel/s, 1.5Gtexel/s or 24GFLOPs with texture filtering and DMA infrastructure</i>
Memori	1GB LPDDR2
Operating System (OS)	<i>Linux atau Windows 10 IoT</i>
Dimensi	85 x 56 x 17mm
Power	Micro USB socket 5V1, 2.5A

Tabel 2. 8 Konektor pada *Raspberry Pi 3* [11]

<i>Ethernet</i>	<i>10/100 BaseT Ethernet socket</i>
<i>Video Output</i>	<i>HDMI (rev 1.3 & 1.4), Composite RCA</i>
<i>Audio Output</i>	<i>Audio Output 3.5mm jack, HDMI USB 4 x USB 2.0 Connector</i>
<i>GPIO Connector</i>	<i>40-pin 2.54 mm (100 mil) expansion header: 2x20 strip Providing 27 GPIO pins as well as +3.3 V, +5 V and GND supply lines</i>
<i>Camera Connector</i>	<i>15-pin MIPI Camera Serial Interface (CSI-2)</i>
<i>Display Connector</i>	<i>Display Serial Interface (DSI) 15 way flat flex cable connector with two data lanes and a clock lane</i>
<i>Memory Card Slot</i>	<i>Push/pull Micro SDIO</i>

Hardware Raspi tidak memiliki *real-time clock* (RTC), sehingga OS harus memanfaatkan timer jaringan server sebagai pengganti. Namun komputer yang mudah dikembangkan ini dapat ditambahkan dengan fungsi *real-time* (seperti DS1307) dan banyak lainnya, melalui saluran GPIO (*General-purpose input/output*) via antarmuka I²C (*Inter-Integrated Circuit*). *Raspi* bersifat *open source* (berbasis *Linux*), *Raspi* bisa dimodifikasi sesuai kebutuhan penggunaannya. Sistem operasi utama *Raspi* menggunakan *Debian GNU/Linux* dan bahasa pemrograman *Python*. Salah satu pengembang OS untuk *Raspi* telah meluncurkan sistem operasi yang dinamai *Raspbian*, *Raspbian* diklaim mampu memaksimalkan perangkat *Raspi*. Sistem operasi tersebut dibuat berbasis *Debian* yang merupakan salah satu distribusi *Linux* OS [11].

2.8 Lazarus

Lazarus adalah sebuah IDE (*Integrated Development Environment*), lingkungan pengembangan perangkat lunak yang terintegrasi sehingga pembuatan *software* menjadi RAPID, dapat diselesaikan dalam waktu singkat. *Lazarus* bersifat *open source*,

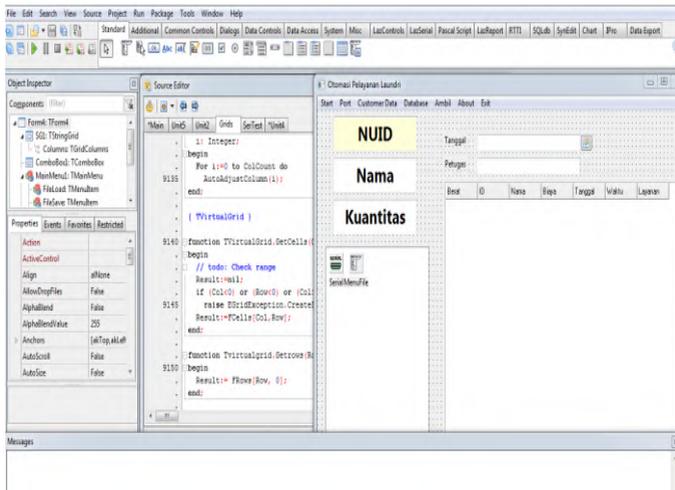
tersedia untuk banyak *platform* terutama *Linux*, *Windows* dan *Macintosh*. Bahasa pemrograman yang dijadikan landasan dalam *Lazarus* adalah *Pascal*. Karena itu, saat pengembangan aplikasi, apa yang disediakan oleh *Lazarus* terasa sebagaimana yang terdapat di *Delphi*, IDE *Pascal visual* yang terkenal di lingkungan *Windows* [12].

2.8.1 *Interface Lazarus*

Tampilan *Lazarus* yang mirip dengan *Delphi* membuat pengguna baru mudah menyesuaikan diri. Lingkungan kerja *Lazarus* terdiri dari jendela-jendela yang terpisah. Masing-masing jendela memiliki fungsi yang berbeda. Menu-menu penting dalam pemanfaatan *Lazarus* ditampilkan pada bagian paling atas dari jendela *Lazarus*. Seperti pada *software* lain, menu *File*, *Edit*, *Search*, *View*, *Window* dan *Help* terdapat pada *Lazarus* dan dengan mudah dipahami kegunaan dan cara menggunakannya. Menu lain yang akan sering digunakan adalah *Project* dan *Run*. Menu *Project* digunakan untuk membuka jendela tertentu untuk mengatur proyek pengembangan *software* yang sedang dilakukan. Dalam *Lazarus*, setiap program komputer yang dikembangkan dinamakan *project*. Menu *Run* digunakan untuk menjalankan, *mentrace* atau menghentikan eksekusi program atau proyek.

Bagian di sebelah kanan *toolbar* tadi, di bawah menu utama adalah daftar komponen yang dapat digunakan untuk membangun aplikasi visual. Komponen-komponen ini dikelompokkan berdasarkan fungsinya. Pada tab *Standard* terdapat komponen-komponen untuk membuat tampilan *standard* dari jendela aplikasi seperti tombol dan teks. Tab *Data Controls* berisi komponen-komponen untuk mengendalikan akses ke suatu *database*. Sedangkan akses ke *database* tersebut dilakukan melalui komponen yang terdapat pada tab *Data Access*.

Di bawah daftar menu, di sebelah kiri, terdapat beberapa *toolbar*. Letakkan kursor *mouse* di atasnya, dan anda akan langsung mengetahui kegunaannya. *Toolbar* ini digunakan untuk membuka proyek, menyimpan proyek, mengeksekusi proyek dan beberapa hal lain terkait pengerjaan proyek [12].

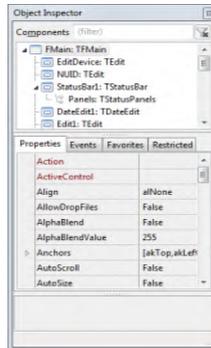


Gambar 2. 11 Tampilan *Lazarus*

Pada sisi kiri terdapat sebuah kotak dengan judul *Object Inspector*. Jendela ini digunakan untuk mengatur perilaku dari suatu kontrol atau komponen yang telah diletakkan di atas suatu *Form* atau jendela aplikasi yang sedang kita kembangkan. Setiap jenis komponen akan mempunyai properti yang berbeda. Properti-properti ini dapat diubah pada saat pengembangan memanfaatkan *object Inspector* atau saat eksekusi program dengan menuliskan perintah-perintah dalam kode program [12].

2.8.2 Jendela *Object Inspector*

Pada sisi kiri terdapat sebuah kotak dengan judul *Object Inspector*. Jendela ini digunakan untuk mengatur perilaku dari suatu kontrol atau komponen yang telah diletakkan di atas suatu *Form* atau jendela aplikasi yang sedang kita kembangkan. Setiap jenis komponen akan mempunyai properti yang berbeda. Properti-properti ini dapat diubah pada saat pengembangan memanfaatkan *object Inspector* atau saat eksekusi program dengan menuliskan perintah-perintah dalam kode program [12].



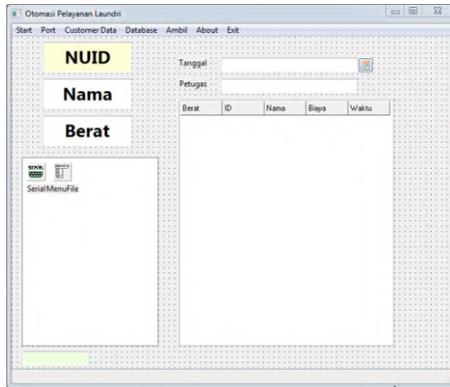
Gambar 2. 12 *Jendela Object Inspector*

Jendela Object Inspector terdiri dari *Object Tree View* dan tab (*Object Properties, Evens, Favorites* dan *Restricted*) .

- *Object Tree View*: Merupakan sebuah diagram pohon yang menggambarkan hubungan logis semua komponen yang terdapat dalam suatu *form*.
- *Object Properties*: Berisi informasi nilai atribut tentang objek yang sedang aktif.
- *Evens*: Berisi informasi kode yang akan dijalankan pada kejadian tertentu misalkan pada saat tombol diklik.
- *Favorites*: Berisi properti objek dan *event* yang sering digunakan.
- *Restricted*: Berisi informasi aturan-aturan properti suatu objek.

2.8.3 *Jendela Form Designer*

Jendela *form designer* merupakan jendela untuk mendesain tampilan visual aplikasi yang biasanya berjudul *Form1*. Jendela ini adalah bagian utama dari sebuah aplikasi visual. *Form* merupakan komponen paling penting dalam pemrograman visual. Di atas *form* inilah diletakkan komponen-komponen lain [12].



Gambar 2. 13 Contoh Jendela *Form Designer*

2.8.4 Jendela *Komponen Palette*

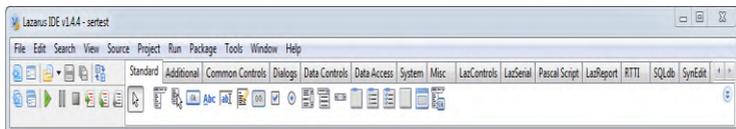
Komponen *palette* berisi ikon-ikon (*tools*) yang digunakan untuk merancang tampilan (*form*) aplikasi. Dalam komponen *palette* semua ikon dikelompokkan dalam berbagai komponen sesuai dengan fungsinya masing-masing [12]. Jendela komponen *palette* dijelaskan pada gambar 2. 14.

2.8.5 Jendela *Source Code Editor*

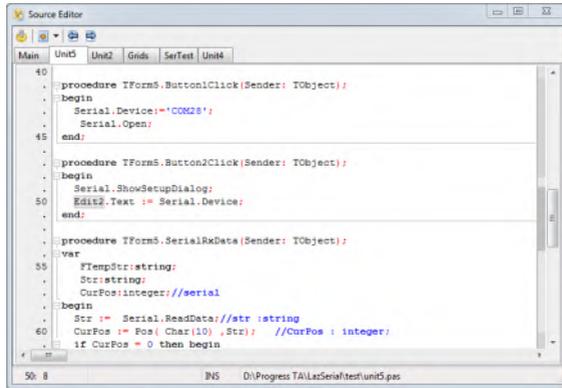
Jendela ini tempat menulis kode. Di sinilah programmer menuliskan baris-baris kode program untuk mengatur apa yang harus dilakukan oleh Aplikasi. Kode program ditulis mengikuti aturan dalam bahasa *Pascal* [12]. Jendela *Source Code Editor* dijelaskan pada gambar 2.15.

2.8.6 Jendela *Messages*

Jendela tempat melihat pesan kesalahan (*error*) saat melakukan kompilasi atau *debug* (mencari error) [12]. Jendela *Messages* dijelaskan pada gambar 2.16.



Gambar 2. 14 Jendela *Komponen Palette*



Gambar 2. 15 *Jendela Source Code Editor*



Gambar 2. 16 *Jendela Messages*

2.9 File CSV (*Comma Separated Value*)

CSV sering digunakan untuk pertukaran data diantara aplikasi yang berbeda. CSV telah menjadi standar *pseudo* di seluruh industri teknologi informasi (IT) [13].

File CSV memiliki aturan seperti berikut,

- Setiap *record* tidak dari satu baris
- Kolom dipisahkan koma (.). Contoh: *Doe, Mary, "99 Longstreet", Anytown, 08123*
- Data yang terdiri dari dua kata atau lebih harus dibatasi tanda petik (" "). Contoh: *"99 Longstreet"*
- Baris pertama dari data akan menjadi judul (*header*) data.
Contoh: *Lastname, FirstName, Address, City, Zipcode*
Doe, John, Cabin, "Desert Kota", 12345
Doe, Mary, "99 Longstreet", Anytown, 08123

File CSV dapat dibuka dengan lembaran kerja *excell* seperti pada gambar 2. 17.,

Lastname	FirstName	Address	City	Zipcode
Doe	John	Cabin	"Desert To	12345
Doe	Mary	"99 Longst	Anytown	8123

Gambar 2. 17 File CSV pada *Excell*

2.10 XAMPP (Apache, MySQL, PHP)

XAMPP merupakan perangkat lunak yang mendukung banyak sistem operasi dan merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai *server* yang berdiri sendiri (*localhost*). Nama XAMPP sendiri merupakan singkatan dari X (empat sistem operasi apapun), Apache, MySQL, PHP, dan Perl. Dengan menginstall XAMPP maka tidak perlu lagi melakukan instalasi dan konfigurasi *web server* Apache, PHP, dan MySQL secara manual. XAMPP akan menginstallasi dan mengonfigurasi secara otomatis [14].

Dalam satu paket XAMPP tersedia :

1. Apache Cgi-Bin
2. FTP
3. Mercury Mail (SMTP)
4. PHP
5. MySQL
6. Perl
7. PHPMyAdmin
8. Webalizer

2.10.1 Apache

Apache berfungsi untuk menampilkan halaman web yang benar, sesuai dengan PHP yang telah dibuat. Apache bersifat *open source*, artinya setiap orang boleh menggunakan, mengambil, dan mengubah kode programnya [14].

2.10.2 Personal Home Page (PHP)

PHP merupakan bahasa pemrograman yang digunakan untuk membuat web yang bersifat *server-side scripting*. PHP memungkinkan untuk membuat halaman web yang bersifat

dinamis, yakni isi informasi *website* berubah-ubah dan interaktif dua arah baik dari pemilik maupun pengguna *website*. PHP dapat dijalankan pada berbagai macam OS, seperti Windows, Linux, dan Mac OS.

Sistem manajemen *database* yang sering digunakan bersama adalah MySQL. Namun PHP juga mendukung sistem manajemen *database* Oracle, Microsoft Access, Interbase, d-Base, PostgreSQL, dan lain-lain. PHP juga bersifat *open source* [14].

2.10.3 MySQL

Database yang digunakan untuk membangun sebuah aplikasi berbasis *client-server* ini yaitu MySQL. SQL sendiri merupakan kepanjangan dari *Structured Query Language*. SQL adalah sebuah bahasa pemrograman yang digunakan untuk berkomunikasi dengan *server*, dimana SQL sendiri terbagi menjadi beberapa bagian, yaitu :

- ✓ DML (*Data Manipulation Language*) yaitu bahasa untuk memanipulasi data
- ✓ DDL (*Data Definition Language*) yaitu bahasa untuk mendefinisikan struktur *database*, seperti CREATE, ALTER, DAN DROP
- ✓ *Transact SQL*, bagian ini adalah bahasa pemrograman yang diletakkan pada *database server*. Konten dari bahasa ini adalah untuk menjaga *automacy* sebuah transaksi pertukaran data.

MySQL merupakan sistem manajemen *database* yang bersifat *open source*. MySQL digunakan untuk membuat dan mengelola *database* beserta isinya, seperti menambahkan, mengubah, dan menghapus data. MySQL juga bersifat *relational*, artinya data-datayang dikelola akan diletakkan pada beberapa tabel terpisah, sehingga proses manipulasi data akan menjadi lebih cepat [14].

2.10.4 PHPMyAdmin

Salah satu perangkat lunak yang digunakan untuk mengelola *database* dalam MySQL adalah PHPMyAdmin. Dengan PHPMyAdmin kita dapat dengan mudah membuat

tabel, mengisi data, dan banyak lagi hal lainnya yang tanpa harus hafal perintahnya, namun cukup dengan mengisi tabel-tabel yang telah tersedia [14].

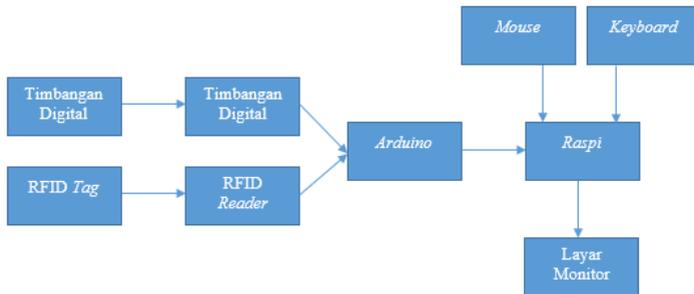
BAB III PERANCANGAN SISTEM

Perancangan otomasi sistem layanan binatu terdiri dari beberapa bagian, yakni perancangan mekanik, perancangan perangkat lunak, perancangan *user interface*, dan perancangan *webbrowser*. Dalam perancangan semua sistem saling terkait antara satu dengan yang lainnya.

3.1 Diagram Blok Sistem

Sistem ini membutuhkan sensor *load cell*, modul RFID *reader*, kartu RFID, LCD 16×2, *arduino uno*, *raspberry pi 3*, layar LCD, *keyboard*, dan *mouse*. Diagram blok sistem dijelaskan pada gambar 3.1.

Sistem dimulai dengan peletakan beban pakaian pada timbangan digital. Data pembacaan nilai analog *load cell* akan dikonversi oleh modul HX711 menjadi nilai digital. Nilai digital akan diproses oleh *arduino* menjadi nilai kuantitas yang sesungguhnya dalam satuan Kg. Setelah itu nomor identitas kartu RFID pelanggan akan dibaca melalui RFID reader. RFID reader akan mengirim data digital pembacaan ke *Arduino* untuk diproses pembacaan nomornya. Nomor identitas memiliki format *hex* 8 karakter. Ketika nomor identitas telah didapatkan, *Arduino* akan mengirimkan data kuantitas bebas pakaian serta identitas pelanggan ke *raspberry pi*. Pada *raspberry pi* terdapat suatu aplikasi dengan menggunakan *software Lazarus*. Aplikasi ini akan mencatat data masukan dari *Arduino*. Data dari *arduino* akan diproses menjadi beberapa informasi yaitu kuantitas pakaian, nomor identitas pelanggan, nama pelanggan, biaya, waktu, dan jenis pelayanan binatu yang diinginkan. Data ini akan terus diperbarui seiring bertambahnya pelanggan pada hari tersebut. Setelah pelayanan hari tersebut selesai. Kumpulan data pelanggan yang didapatkan hari itu akan otomatis disimpan ke *database* berformat CSV pada memori *raspberry pi*. Data harian yang telah didapatkan akan otomatis diperbarui ke dalam bentuk *webserver* yang bisa diakses *online*. Data pada *web* akan diperbarui untuk memperlihatkan proses layanan binatu pakaian pelanggan.



Gambar 3. 1 Diagram Blok Perangkat Keras Sistem

Raspberry pi dilengkapi dengan layar LCD, *keyboard*, dan *mouse* untuk memudahkan kegiatan operasional. Layar LCD digunakan untuk menampilkan aplikasi, mengolah database, dan mengatur isi pada *webserver*. LCD dan keyboard digunakan untuk memasukan *input* atau masukan yang diperlukan.

3.2 Perancangan Perangkat Keras

Perangkat keras pada tugas akhir ini adalah yaitu timbangan digital, Komponen yang dibutuhkan timbangan digital adalah sensor *load cell*, besi, baut penghubung, modul HX711, LCD 2×16, kabel dan tempat beban.

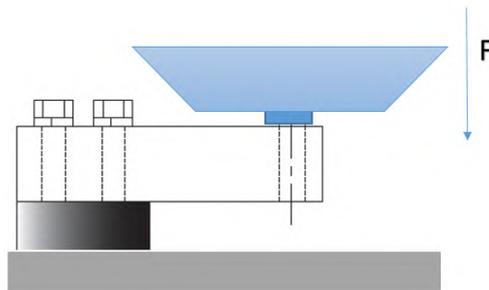
Komponen utama timbangan terdiri dari 4, yaitu *load cell*, penyangga *load cell*, alas timbangan, dan kota beban timbangan diletakkan. Terdapat beberapa hal yang harus diperhatikan dalam pembuatan mekanik timbangan digital. Pertama, saat pengukuran, komponen yang boleh berubah hanyalah *load cell*. Perubahan ini berbentuk defleksi kemiringan *load cell* ketika beban dimasukkan. Oleh karena itu, alas timbangan harus dalam posisi dan bentuk statis ketika pengukuran dilakukan. Untuk mendapatkan hal tersebut dibutuhkan bahan alas yang kuat sebagai dasar dari timbangan tersebut. Bahan yang dipakai adalah besi dengan ketebalan 1 cm. Dimensi alas yang dipakai berukuran 20×15 cm. Penyangga juga dibutuhkan untuk mempertahankan kondisi statis timbangan ketika dilakukan pengukuran. Penyangga menggunakan bahan besi agar sistem

statis saat pengukuran. Dimensi dari besi penyangga adalah $8 \times 3 \times 2$ cm. Penyangga harus berada dalam posisi tegak lurus dengan alas timbangan ketika terjadi pengukuran. Sebagai tambahan, penggunaan besi pada alas dan penyangga dikarenakan timbangan ini mampu mengukur beban sampai 10 Kg. Oleh karena itu dibutuhkan suatu bahan yang kuat untuk menjaga kondisi statis timbangan saat pengukuran. Terakhir, tempat beban timbangan menggunakan media berbentuk mangkuk yang terbuat dari bahan aluminium. Bahan ini dipilih karena ringan sehingga hanya memberi tekanan yang sedikit saat beban timbangan bernilai nol.

Setelah semua komponen didapatkan, hal selanjutnya yang dirancang adalah perhitungan titik beban efektif agar sistem setimbang saat pengukuran dilakukan. Pengaturan titik beban *load cell* dapat dilihat pada gambar 3.2,

Semua komponen harus tersambung dengan kuat agar tidak terjadi defleksi luar yang berasal bukan dari *load cell*. Defleksi dari luar akan menyebabkan *noise* (gangguan) terhadap pembacaan nilai sensor. Hal tersebut akan menyebabkan keakuratan nilai pengukuran timbangan akan berkurang.

Prinsip kerja, beban akan memberi tekanan pada ujung *load cell*. Setiap perubahan lengkungan *load cell* akan menimbulkan defleksi pada *load cell*. Defleksi tersebut akan mempengaruhi nilai resistansi *load cell*. Selanjutnya, nilai keluaran analog *load cell* akan mengalami perubahan. Nilai analog *load cell* akan dikonversi menjadi nilai digital menggunakan modul HX711 untuk seterusnya diproses melalui mikrokontroler.



Gambar 3. 2 Pengaturan Titik Beban *Load Cell*

3.3 Perancangan Elektrik

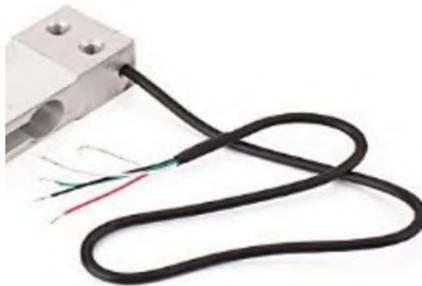
Perangkat elektrik dalam tugas akhir ini meliputi rangkaian catu daya, rangkaian sensor timbangan, rangkaian RFID.

3.3.1 Rangkaian Sensor Timbangan

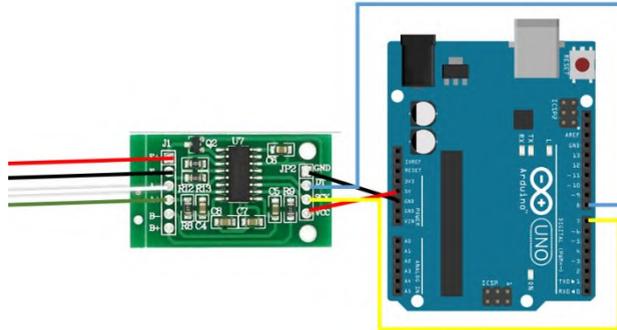
Sensor *load cell* dihubungkan dengan modul HX711. *Load cell* mempunyai empat kabel penghubung yaitu VCC (kabel merah), *Ground* (kabel hitam), Data + (kabel putih), dan Data – (kabel hijau). Kabel penghubung tersebut dijelaskan pada gambar 3.3.

Setiap kabel penghubung dihubungkan dengan pin *input* sensor pada modul HX711. Pin *input* sensor pada HX711 terdiri dari pin E+, E-, A+ (Data1+), A- (Data1-), B+(Data2+), dan B- (Data2-). Karena jumlah *load cell* hanya satu, maka pin Data yang dipakai hanya sepasang yaitu A+ dan A-.

Modul HX711 mempunyai empat pin sebagai masukan untuk mikrokontroler (*arduino*). Pin-pin tersebut adalah Vcc, SCK, DT, dan *Ground*. Dalam hal ini pin diatur sesuai dengan kebutuhan. Pin Vcc dihubungkan dengan supply 5V *arduino*, pin *ground* dihubungkan dengan *ground arduino*, pin DT dihubungkan dengan pin digital 8 *arduino* dan pin SCK dihubungkan dengan pin digital 7 *arduino*.



Gambar 3.3 Kabel *Load Cell*

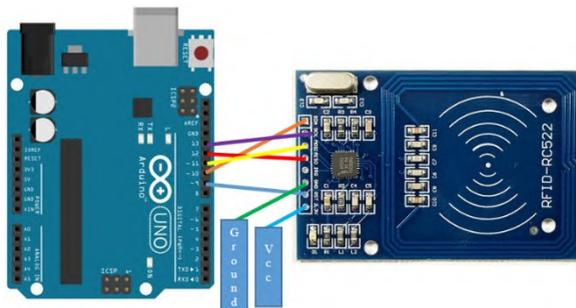


Gambar 3. 4 Konfigurasi modul HX711 dengan *Arduino*

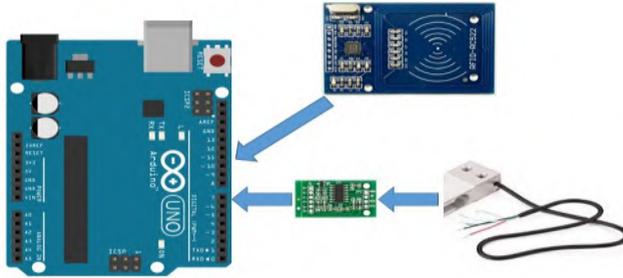
3.3.2 Rangkaian RFID

Untuk bekerja, modul RFID membutuhkan mikrokontroler sebagai pengolah data dan *power supply* untuk bekerja. Pada tugas akhir ini mikrokontroler yang digunakan adalah *arduino* sedangkan *power supply* menggunakan *supply* internal dari *arduino* sebesar 3.3V dan *ground* internal. Pin-pin yang harus dihubungkan dengan *arduino* adalah pin *reset*, SCK, Dout, MISO, MOSI, SDA(SS), dan CLK.

Pengaturan pin modul RFID dengan *arduino* yang saya rancang adalah pin *Reset* → pin 9, pin SDA(SS) → pin 10, pin MOSI → pin 11, pin MISO → pin 12, dan pin SCK → 13. Rancangan pengaturan pin MFRC522 dengan *arduino* dijelaskan pada gambar 3.5,



Gambar 3. 5 Konfigurasi MFR522 dengan *Arduino*



Gambar 3. 6 Rancangan Konfigurasi *Arduino*

3.3.3 Mikrokontroler *Arduino*

Dalam tugas akhir ini, mikrokontroler yang digunakan adalah *Arduino Uno R3*. *Arduino* digunakan sebagai pengakuisisi data sensor timbangan dan modul RFID MFRC522. *Arduino* juga digunakan sebagai pengatur tampilan LCD.

Arduino membutuhkan *power supply* sebesar 5 Volt. Untuk memenuhi kebutuhan tersebut, arduino memakai sumber tegangan dari *raspberry pi* melalui USB. Pemrograman yang digunakan menggunakan bahasa C. Konfigurasi *arduino* yang digunakan secara keseluruhan dijelaskan pada gambar 3.6.

3.3.4 *Raspberry Pi* dan Kelengkapannya

Raspberry pi memiliki fungsi komputer pada tugas ini. *Raspberry pi* berfungsi sebagai *database*, pusat *human machine interface* dan pusat pengendalian *webserver*.

Raspberry pi tidak mempunyai *clock (real time clock)* internal, sehingga ditambahkan *clock* eksternal. Konfigurasi RTC eksternal dengan *raspberry pi* adalah dengan menghubungkan pin-pin RTC dengan pin GPIO *raspberry pi*. *Raspberry pi* membutuhkan *power supply* dengan tegangan 5 Volt dan arus ± 2.0 A. Arus besar digunakan untuk menopang penggunaan *mouse*, *keyboard*, dan *arduino uno*. *Supply* tersebut didapatkan dari adaptor yang telah tersedia di pasaran. Untuk *input* dan *output* data, *raspberry pi* membutuhkan monitor LCD, *keyboard* dan *mouse*. Monitor digunakan untuk menampilkan data dalam bentuk visual. *Keyboard* dibutuhkan untuk memasukkan *input* data berupa karakter. *Mouse* dibutuhkan untuk memudahkan *interface* pengguna dengan sistem.



Gambar 3. 7 *Raspberry Pi* dan Kelengkapannya

3.4 Perancangan Perangkat Lunak (*Software*)

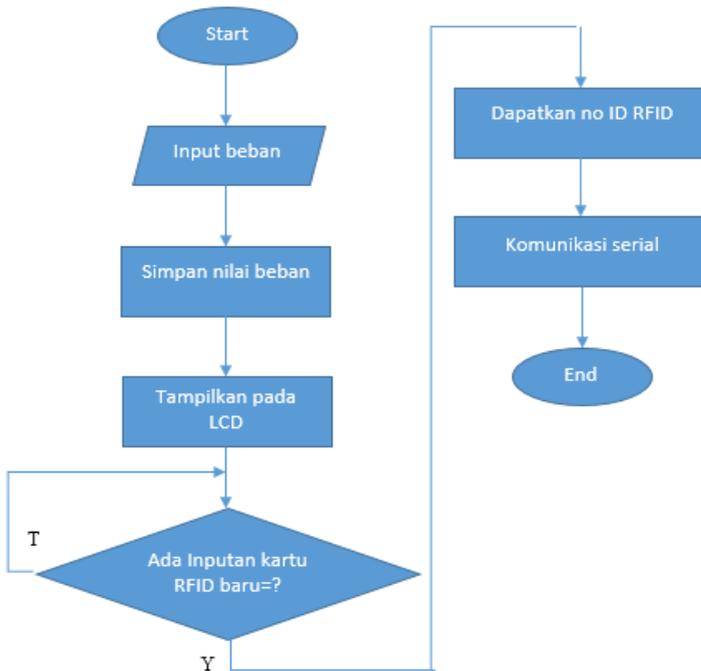
Perangkat lunak pada tugas akhir ini terdiri dari program timbangan digital, RFID, komunikasi serial *arduino* dengan *raspberry pi*, *human machine interface* otomasi layanan binatu, dan program *webserver*.

3.4.1 Program *Arduino*

Program *arduino* terdiri dari program timbangan digital, dan RFID. Konfigurasi pin *arduino* yang digunakan dijelaskan pada tabel 3.1.

Tabel 3. 1 Konfigurasi Pin *Arduino*

	Pin	Arduino
Program Timbangan	Vcc (5 V)	5,5 V
	<i>Ground</i>	<i>Ground</i>
	DT	7
	SCK	8
Program RFID	Vcc (3,3 V)	3,3 V
	<i>Ground</i>	<i>Ground</i>
	RST	9
	SDA (SS)	10
	MOSI	11
	SCK	13



Gambar 3. 8 Diagram Alir Program *Arduino*

Diagram alir dari program pada *arduino* dijelaskan pada gambar 3.8. Prinsip kerjanya, pertama beban diukur dengan menggunakan timbangan digital. Nilai timbangan akan disimpan sementara dan ditampilkan pada LCD. Saat kartu RFID dimasukkan, data RFID dan data kuantitas beban akan dikirim melalui komunikasi serial ke *Raspberry Pi*.

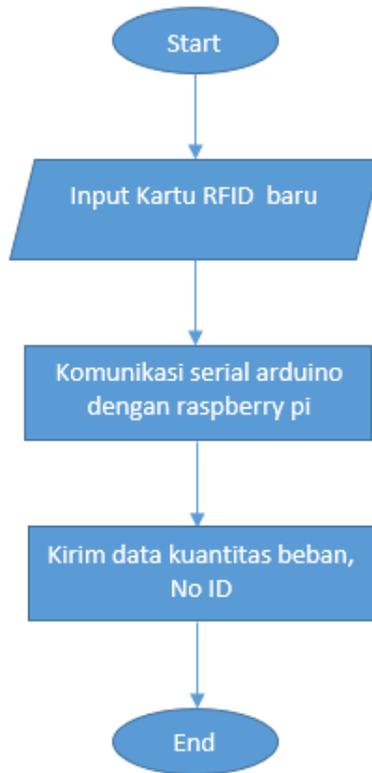
3.4.2 Komunikasi Serial *Arduino* dengan *Raspberry Pi*

Komunikasi serial *arduino* dengan *Raspberry Pi* menggunakan media usb. Konfigurasi komunikasi serial yang digunakan dijelaskan pada tabel 3.2.

Tabel 3. 2 Konfigurasi Komunikasi Serial

<i>Port</i>	/dev/ttyACM0
<i>Baud Rate</i>	9600
<i>Data bits</i>	8
<i>Stop bits</i>	1

Diagram alir dari komunikasi serial yang dirancang dan digunakan dijelaskan pada gambar 3.9.



Gambar 3. 9 Diagram Alir Komunikasi Serial

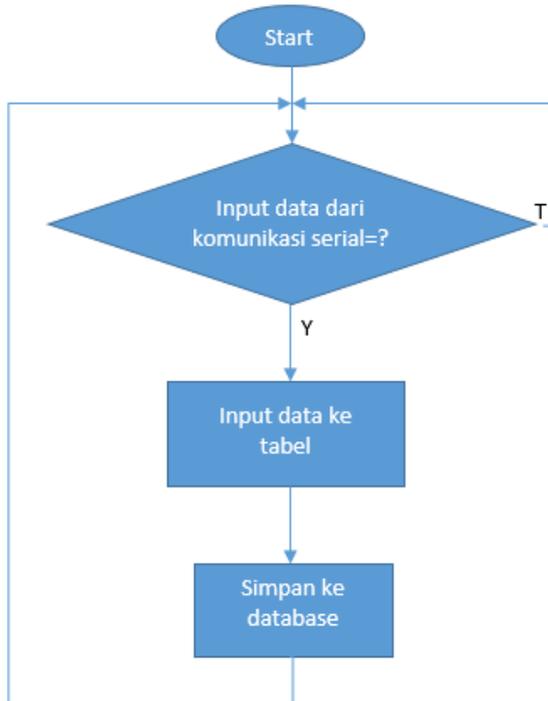
Prinsip kerjanya, ketika RFID kartu di deteksi, *arduino* akan mengirim data kuantitas pakaian dan nomor ID ke *Raspberry Pi*.

3.4.3 Program *Raspberry Pi*

Program pada *Raspberry Pi* terdiri dari program aplikasi *human interface* otomasi layanan binatu. Program terdiri dari memasukkan data, menyimpan dan memanggil *database*.

Diagram alir dari program aplikasi yang dirancang dijelaskan pada gambar 3.10.

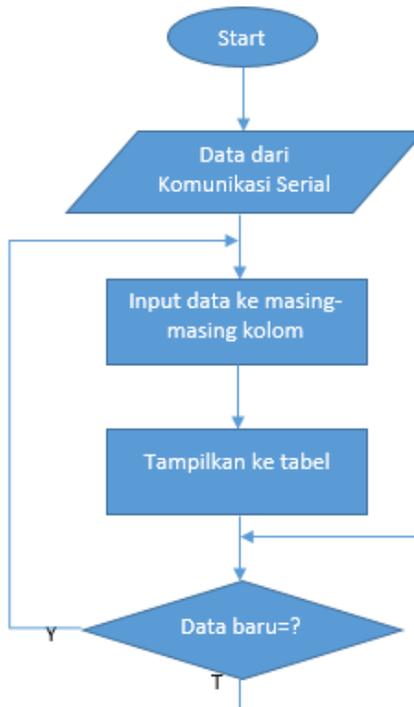
Prinsip kerja program yaitu ketika data dikirim dari *arduino* ke *Raspberry Pi*, data dengan otomatis dimasukkan ke dalam tabel. Kemudian data akan otomatis disimpan pada *database*.



Gambar 3. 10 Diagram Alir Aplikasi

3.4.3.1 Program *Input Data ke Tabel*

Diagram alir dari program ini dijelaskan pada gambar 3.11. Prinsip kerja dari program ini terdiri dari beberapa proses. Proses ini terjadi ketika tombol “*Start*” ditekan. Masukan data didapatkan dari komunikasi serial. Setiap data akan dimasukkan dengan metode CSV. Kolom berat dan ID akan diisi sesuai dengan data yang didapat dari *arduino*. Kolom nama diisi sesuai dengan ID yang dimasukkan. Prosesnya akan dijelaskan pada program memanggil *database*. Biaya akan diisi sesuai dengan layanan yang dipilih. Kolom tanggal dan waktu diisi sesuai dengan kondisi waktu proses. Terakhir, kolom layanan akan diisi manual oleh pengguna.



Gambar 3. 11 Diagram Alir *Input Data ke Tabel*

3.4.3.2 Program Memanggil Database

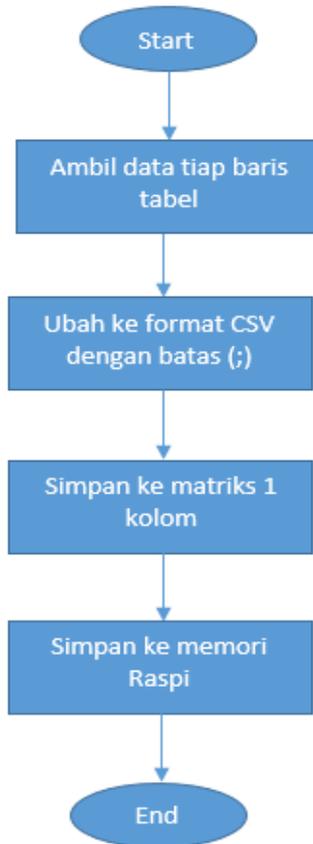
Diagram alir dari program ini dijelaskan pada gambar 3.12. Prinsip kerja dari program ini terdiri dari beberapa proses. Saat *database* dipanggil, program akan otomatis menuju alamat *file* CSV yang disimpan pada memori *Raspberry Pi*. Ketika data telah didapatkan data tersebut akan diubah menjadi matriks kolom imajiner. Proses memanggil ini digunakan saat mengisi kolom nama pada tabel, melihat *database* nama pelanggan dan menampilkan data harian pelanggan.



Gambar 3. 12 Diagram Alir Pemanggilan Database

3.4.3.3 Program Menyimpan Database

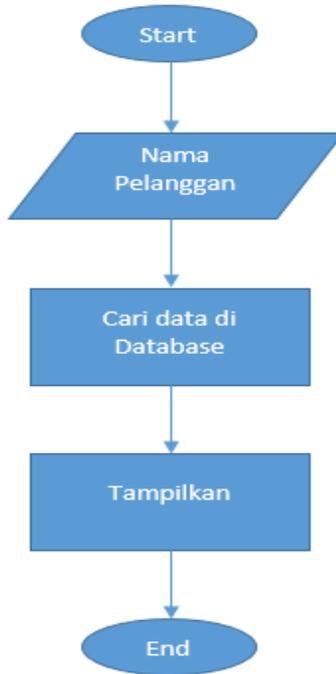
Diagram alir dari program ini dijelaskan pada gambar 3.14. Prinsip kerja dari program ini terdiri dari beberapa proses. Data setiap baris akan dikonversi menjadi *file* CSV. Pembatas antar kolom akan memakai tanda baca titik koma(;). *File* akan disimpan dalam bentuk matriks kolom imajiner. Matriks tersebut akan dikonversi menjadi *file* CSV.



Gambar 3. 13 Diagram Alir Menyimpan ke Database

3.4.3.4 Program *Webserver*

Diagram alir dari program ini dijelaskan pada gambar 3.15. Prinsip kerja dari program ini terdiri dari beberapa proses. Pelanggan memasukan nama ke dalam kotak yang disediakan pada *web*. Selanjutnya data pelanggan akan ditampilkan sesuai nama yang dimasukan.



Gambar 3. 14 *Flowchart Webserver*

BAB IV PENGUJIAN DAN ANALISIS

Pengujian pada sistem terdiri dari beberapa bagian yaitu pengujian perangkat keras, dan perangkat lunak. Tujuan dari perancangan ini untuk mendapatkan parameter dari perancangan sistem tersebut.

4.1 Pengujian Perangkat Keras

Pengujian perangkat keras dilakukan untuk mendapatkan parameter dalam perancangan sistem pada sistem elektronis. Pada sistem elektronis yang akan diukur dalam pengukuran yaitu timbangan digital, RFID MFRC522.

4.1.1 Pengujian Timbangan Digital

Timbangan digital merupakan komponen utama untuk mendapatkan kuantitas beban pakaian. Terdapat beberapa pengujian untuk timbangan digital.

Bentuk fisik timbangan digital yang direalisasikan dijelaskan pada gambar 4.1. Pengujian pertama adalah pengujian untuk melihat apakah *load cell* dapat bekerja dengan baik atau tidak. Pengujian dilakukan dengan melihat sinyal keluaran analog *load cell* ketika adanya perubahan beban. Karena sinyal keluaran *load cell* sangat kecil, diperlukan penguatan sinyal. Penguatan yang digunakan sebesar 1000 kali. Pengujian dengan menggunakan beban 0 Kg, 0.5 Kg, 1.0 Kg, dan 1.5 Kg. Hasil pengujian dapat dilihat pada tabel 4.1.



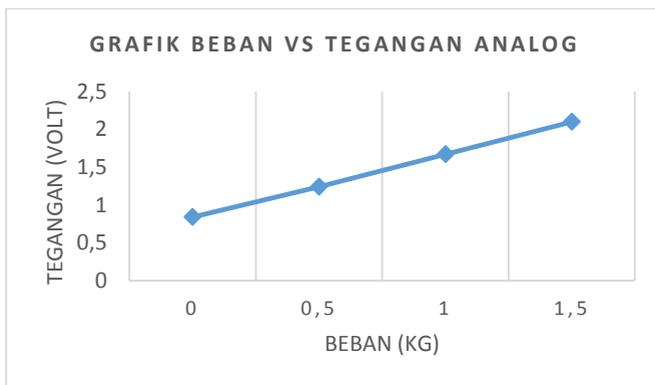
Gambar 4. 1 Realisasi Timbangan Digital

Tabel 4. 1 Tabel Pengujian Beban

Beban	Nilai Tegangan Analog dengan Gain 1000×
0	0.84 V
0.5 Kg	1.24 V
1.0 Kg	1.67 V
1.5 Kg	2.10 V

Dalam pengujian terlihat setiap kenaikan beban 0.5 Kg akan meningkatkan nilai keluaran analog ± 0.4 Volt. Hal tersebut sesuai dengan karakteristik *load cell* yang memiliki perubahan resistansi yang linier seiring perubahan beban. Grafik hasil pengujian dijelaskan pada gambar 4.2.

Selanjutnya kalibrasi timbangan digital untuk pengukuran beban. Kalibrasi menggunakan modul HX711. Diperlukan dua nilai untuk kalibrasi. Pertama saat tidak ada beban (*zero calibration factor*). Nilai *zero factor* yang diuji dari 5 pengukuran dapat dilihat pada tabel 4.2. Nilai selanjutnya adalah nilai *calibration factor*, nilai ini didapatkan dengan cara menyesuaikan nilai beban sesungguhnya dengan nilai yang didapatkan dari pengukuran.



Gambar 4. 2 Grafik Beban vs Tegangan Analog

```
COM28 (Arduino/Genuino Uno)

HX711 calibration sketch
Remove all weight from scale
After readings begin, place known weight on scale
Press + or a to increase calibration factor
Press - or z to decrease calibration factor
Zero factor: 8602770
```

Gambar 4.3 Kalibrasi Nilai *Zero Factor*

```
COM28 (Arduino/Genuino Uno)

Reading: 0.002 kg calibration_factor: 410000.00

COM28 (Arduino/Genuino Uno)

Reading: 1.500 kg calibration_factor: 409120.00
```

Gambar 4.4 Kalibrasi Nilai *Calibration Factor*

Tabel 4. 2 Tabel *Zero Factor* Dan *Calibration Factor*

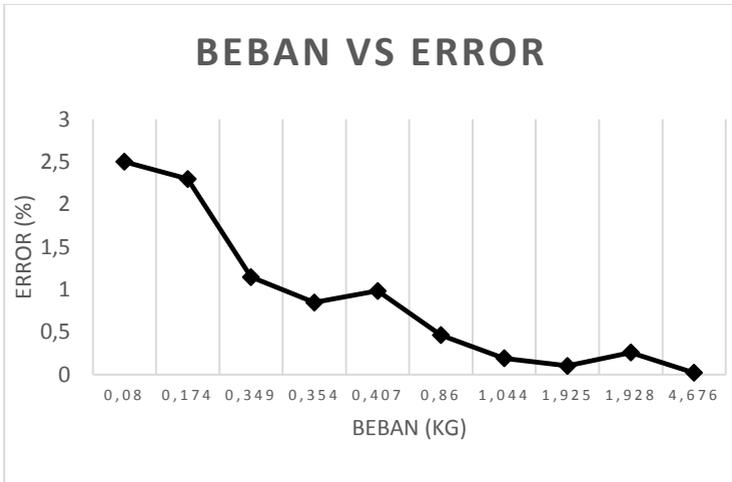
Pengujian	Nilai Zero Factor
1	8601707
2	8602770
3	8602388
4	8601974
5	8601269
Pengujian	Calibration Factor
Beban 1.5 Kg	409120

Nilai tersebut digunakan untuk program pengukuran kuantitas beban. Nilai *zero factor* yang dipakai merupakan nilai rata-rata dari lima pengujian. Nilai *zero factor* yang digunakan adalah 8602022. Sedangkan nilai *calibration factor* yang digunakan 409120.

Selanjutnya nilai *zero factor* dan *calibration factor* digunakan untuk program pengukuran. Pengujian dilakukan dengan pengukuran 10 beban yang diambil secara acak (*random*). Hasil pengukuran dijelaskan pada tabel 4.3.

Tabel 4. 3 Pengujian Timbangan Digital

Pengukuran	Nilai Sebenarnya	Nilai Pengukuran	Error
1	80 gram	78 gram	2,5 %
2	174 gram	170 gram	2,29 %
3	349 gram	345 gram	1,1 %
4	354 gram	351 gram	0,9 %
5	407 gram	403 gram	0,9 %
6	860 gram	856 gram	0,5 %
7	1044 gram	1042 gram	0,2 %
8	1925 gram	1923 gram	0,1 %
9	1928 gram	1923 gram	0,2 %
10	4676 gram	4675 gram	0,02%

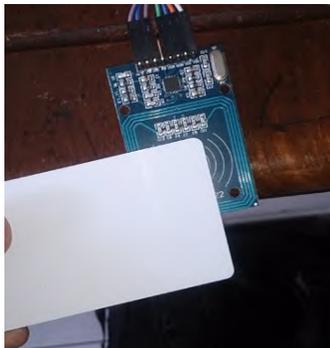


Gambar 4. 5 Beban vs eror pengukuran

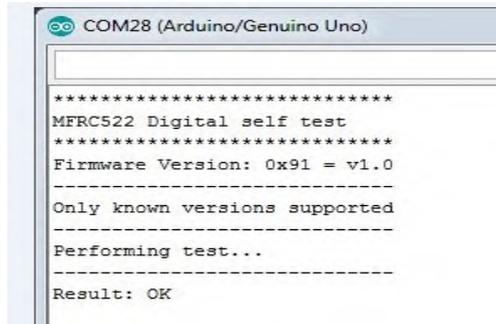
Dari pengujian disimpulkan bahwa eror rata-rata dari timbangan digital sebesar 0.88%.

4.1.2 Pengujian Modul RFID MFRC522

Modul RFID berfungsi untuk membaca data NUID pada kartu RFID (*tag*). NUID berfungsi sebagai acuan untuk membaca nama pelanggan pada *database*.



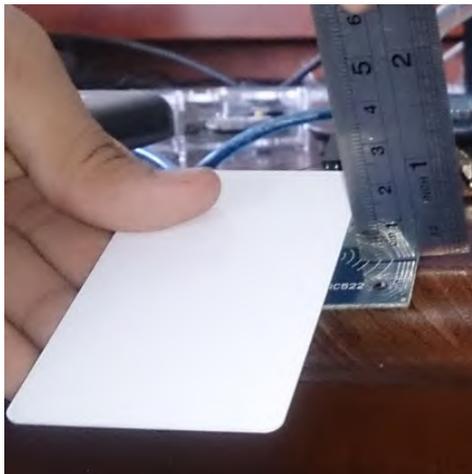
Gambar 4. 6 Pengujian RFID



Gambar 4. 7 Hasil Pengujian RFID

Pengujian pertama yang dilakukan adalah menguji apakah modul RFID dapat berfungsi dengan baik. Hasil pengujian dapat dilihat pada gambar 4.7. Dari hasil pengujian menunjukkan bahwa modul RFID berfungsi dengan baik. Hal tersebut diketahui dengan penjelasan hasil *Performing Test* yang menghasilkan keterangan OK.

Pengujian kedua, merupakan pengujian jarak optimal pembacaan kartu RFID. Hasil pengujian dijelaskan pada tabel.

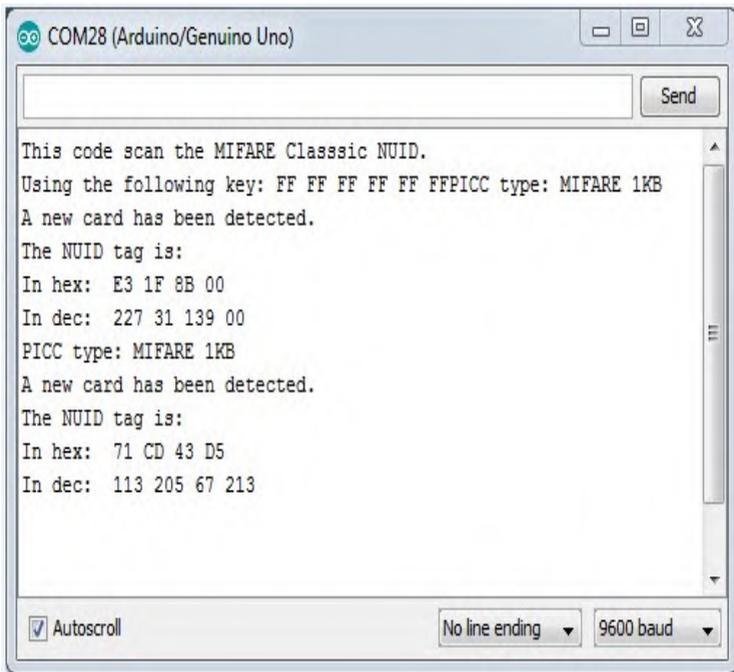


Gambar 4. 8 Pengujian Jarak Pembacaan

Tabel 4. 4 Hasil Pengujian Jarak Optimum Pembacaan RFID

Jarak (cm)	Pembacaan
2	Tidak bisa dibaca
1	Tidak bisa dibaca
0,5	Bisa dibaca
0,3	Bisa dibaca
0,1	Bisa dibaca
0	Bisa dibaca

Selanjutnya adalah pengambilan data NUID pada kartu-kartu RFID yang digunakan oleh pelanggan. Pengambilan data menggunakan 2 sampel kartu. NUID memiliki 2 format, yaitu format bilangan *hex* (8 digit) dan desimal (11 digit). Hasil yang didapatkan dapat dilihat pada gambar 4.9 dan tabel 4.5.



Gambar 4. 9 Pengambilan Data NUID

Tabel 4. 5 Data NUID

Kartu	NUID dalam decimal	NUID dalam hex
1	227 31 139 00	E3 1F 8B 00
2	113 205 67 213	71 CD 43 D5
3	89 79 61 213	59 4F 3D D5
4	107 182 61 213	6B B6 3D D5
5	244 45 59 213	F4 2D 3B D5
6	246 175 61 213	F6 AF 3D D5
7	64 70 59 213	40 46 3B d5
8	109 78 61 213	6D 4E 3D D5
9	195 204 61 213	C3 CC 3D D5
10	25 126 61 213	19 7E 3D D5

Data NUID yang disimpan hanya dalam *hex*. Pertimbangannya data dalam *hex* menggunakan jumlah digit yang lebih sedikit dan konstan (delapan digit).

4.2 Pengujian Perangkat Lunak

Pengujian perangkat lunak terdiri dari pengujian komunikasi serial, pengujian *input* data, pengujian simpan data ke *database*, pengujian pemanggilan *database* dan pengujian *webservice*.

4.2.1 Pengujian Komunikasi Serial

Komunikasi serial digunakan untuk mengirim data dari *arduino* ke *Raspberry Pi*. Data yang dikirim berupa kuantitas beban dan NUID dari kartu RFID.



Gambar 4. 10 Pengujian Komunikasi Serial

Pengujian pertama adalah pemeriksaan *port arduino* pada *Raspberry Pi*. Pengujian terbacanya *usb arduino* dilakukan dengan mengetikkan *lsusb* pada command prompt. Pengujian letak *port usb arduino* dilakukan dengan mengetikkan *ls /dev/tty**. Gambar 4.11 Adalah hasil pengujian pemeriksaan *port arduino* pada *raspberry pi*. *Port arduino* yang terdeteksi oleh *Raspberry Pi* adalah */dev/ttyACM0*. *Port* ini yang digunakan untuk program komunikasi serial antara *arduino* dengan *Raspberrry pi*.

Pengujian komunikasi serial dilakukan dengan mengirim data kuantitas dengan data NUID kartu. Konfigurasi komunikasi serial yang digunakan dijelaskan pada tabel 4.6.

```

danh@rpi ~/Arduino2RPI $ lsusb
Bus 001 Device 002: ID 0424:9512 Standard Microsystems Corp.
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 001 Device 003: ID 0424:ec00 Standard Microsystems Corp.
Bus 001 Device 006: ID 2341:0001 Arduino SA Uno (CDC ACM)

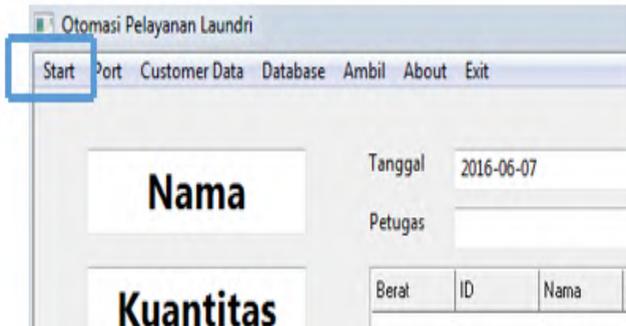
/dev/tty0 /dev/tty2 /dev/tty30 /dev/tty41 /dev/tty52 /dev/tty63
/dev/tty1 /dev/tty20 /dev/tty31 /dev/tty42 /dev/tty53 /dev/tty7
/dev/tty10 /dev/tty21 /dev/tty32 /dev/tty43 /dev/tty54 /dev/tty8
/dev/tty11 /dev/tty22 /dev/tty33 /dev/tty44 /dev/tty55 /dev/tty9
/dev/tty12 /dev/tty23 /dev/tty34 /dev/tty45 /dev/tty56 /dev/ttyACM0
/dev/tty13 /dev/tty24 /dev/tty35 /dev/tty46 /dev/tty57 /dev/ttyHMM0
/dev/tty14 /dev/tty25 /dev/tty36 /dev/tty47 /dev/tty58 /dev/ttyprintk
/dev/tty15 /dev/tty26 /dev/tty37 /dev/tty48 /dev/tty59
/dev/tty16 /dev/tty27 /dev/tty38 /dev/tty49 /dev/tty6
/dev/tty17 /dev/tty28 /dev/tty39 /dev/tty5 /dev/tty60
/dev/tty18 /dev/tty29 /dev/tty4 /dev/tty50 /dev/tty61

```

Gambar 4. 11 Pengujian USB Arduino pada Raspberry Pi

Tabel 4. 6 Konfigurasi Komunikasi Serial

Port	/dev/ttyACM0
Baud Rate	9600
Data bits	8
Stop bits	1



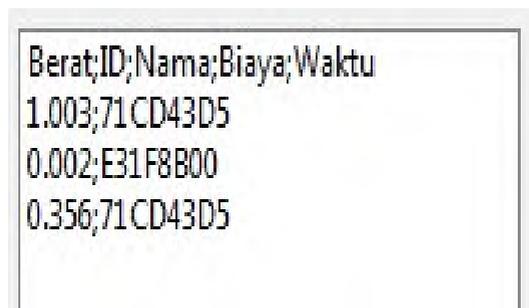
Gambar 4. 12 Tombol *Start*

Pengujian dilakukan dengan menekan tombol *start* untuk menjalankan program. Tombol *Start* dijelaskan pada gambar 4. 12.

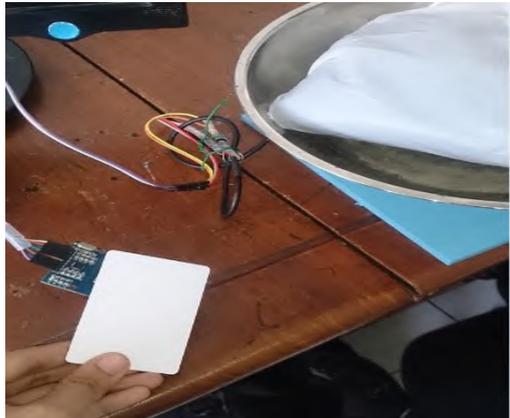
Hasil pengujian dijelaskan pada gambar 4.13. Pada gambar dapat dilihat data yang dikirim *arduino* telah diterima oleh *Raspberry Pi*. Dari hasil pengujian dapat disimpulkan bahwa komunikasi serial *arduino* dengan *raspberry pi* berfungsi dengan baik.

4.2.2 Pengujian *Input Data*

Pengujian *input data* dilakukan dengan menkoversi data yang didapat dari komunikasi serial. Pengujian dilakukan dengan menjalankan aplikasi sesuai prosedur. Pertama, data diterima melalui komunikasi serial. Data dari komunikasi serial dimasukkan sesuai kolom. Proses ini memasukkan ini (*input*) berjalan secara otomatis. Hasil pengujian dijelaskan pada gambar 4.15.



Gambar 4. 13 Hasil Pengujian Komunikasi Serial pada *Raspberry Pi*



Gambar 4. 14 Pengujian *Input Data*

Otomasi Pelayanan Laundry

Start Port Customer Data Database Ambil About Exit

cikampek

Tanggal: 2016-06-07

Petugas:

1.565 Kg

Berat;ID;Nama;Biaya;Waktu
 1.565;71CD43D5
 1.565;594F3DD5
 1.566;197E3DD5
 1.565;F6AF3DD5
 1.565;F42D38D5
 1.564;E31F8800
 1.564;404638D5
 1.565;6D4E3DD5

Berat	ID	Nama	Biaya	Tanggal	Waktu	Layanan
1.565	71CD43D5	habsyi	Rp20000	11:15:33	2016-06-07	
1.565	594F3DD5	fajar	Rp20000	11:15:38	2016-06-07	
1.566	197E3DD5	tango	Rp20000	11:15:49	2016-06-07	
1.565	F6AF3DD5	chopper	Rp20000	11:15:53	2016-06-07	
1.565	F42D38D5	patrick	Rp20000	11:15:55	2016-06-07	
1.564	E31F8800	al	Rp20000	11:15:57	2016-06-07	
1.564	404638D5	ussop	Rp20000	11:15:59	2016-06-07	
1.565	6D4E3DD5	cikampek	Rp20000	11:16:02	2016-06-07	

COM28

ReadCount : 3

Gambar 4. 15 Hasil Pengujian *Input Data*

Dari hasil pengujian dapat disimpulkan bahwa proses *input* data telah berfungsi dengan baik. Pengujian selanjutnya adalah melihat bagaimana pemakaian *processor* (CPU) saat aplikasi dijalankan. Hasil pengujian dijelaskan pada tabel 4.7. Dari pengujian dapat disimpulkan pemakaian CPU *Raspberry Pi* saat aplikasi dijalankan sebesar 5%.

4.2.3 Pengujian Simpan Data Ke *Database* (*Export*)

Pengujian simpan data ke *database* dilakukan dengan memeriksa apakah *database* telah tersimpan di memori dan dapat diakses dengan lembaran kerja *Excell*. Hasil pengujian dapat dilihat pada gambar 4. 16 .

Tabel 4. 7 Pengujian pemakain CPU *Raspberry Pi*

Pengujian	Beban CPU
1	5%
2	4%
3	5%
4	5%
5	5%

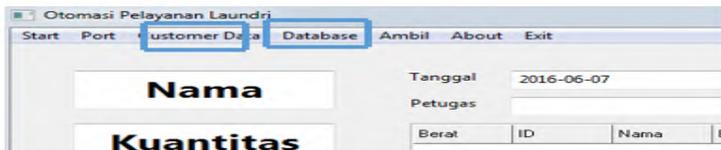
	1	2	3	4	5	6
1	Berat	ID	Nama	Biaya	Waktu	Layanan
2	1.0	71CD43D5	habsyi	Rp20.000	11:33:33	
3	1.5	E31F8B00	al	Rp20.000	11:33:53	
4	0.5	71CD43D5	habsyi	Rp20.000	11:34:21	
5						

Gambar 4. 16 Hasil Pengujian Simpan Data ke *Database*

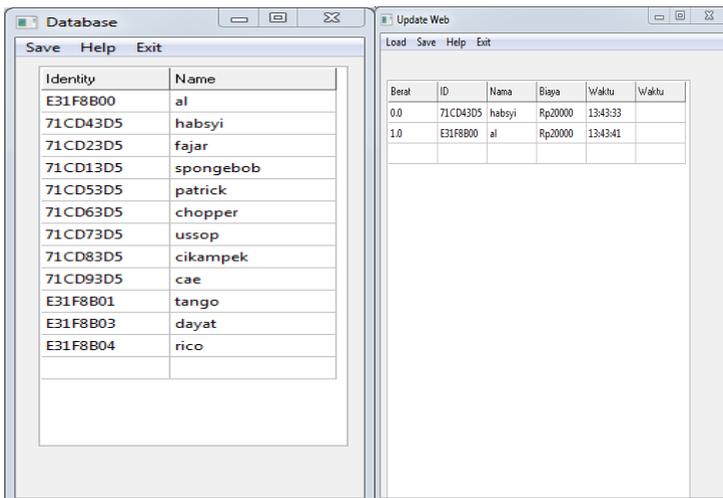
Dari hasil pengujian dapat disimpulkan bahwa penyimpanan *database* ke memori berfungsi dengan baik. Untuk tampilan *database* pada lembaran kerja *Excell* masih bersifat sederhana. Belum adanya batas (*border*) yang jelas diantara data. Oleh karena itu perlu adanya pemberian batas manual untuk kerapian penyimpanan data.

4.2.4 Pengujian Pemanggilan *Database (Import)*

Pengujian *database* digunakan untuk menampilkan *database* yang berada di memori (diluar aplikasi) ke tabel yang berada di aplikasi. Pengujian dilakukan dengan memanggil *database* data pelanggan dan *database* data harian operasional layanan binatu. Caranya adalah dengan menekan tombol *database* dan *customer data* seperti pada gambar 4.17. Hasil pengujian dijelaskan pada gambar 4.18.



Gambar 4. 17 *Customer Data* dan *Database*



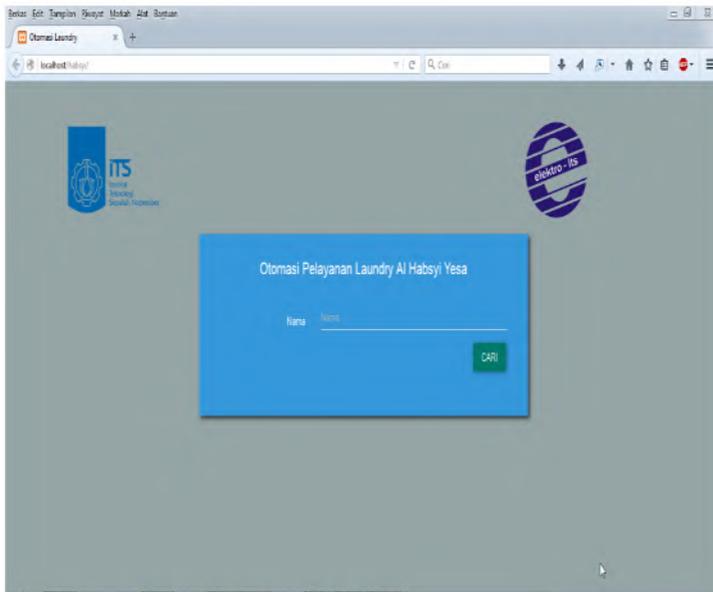
Gambar 4. 18 Pengujian Pemanggilan *Database (Import)*

Dari hasil pengujian dapat disimpulkan bahwa proses pemanggilan data dapat dilakukan dengan baik.

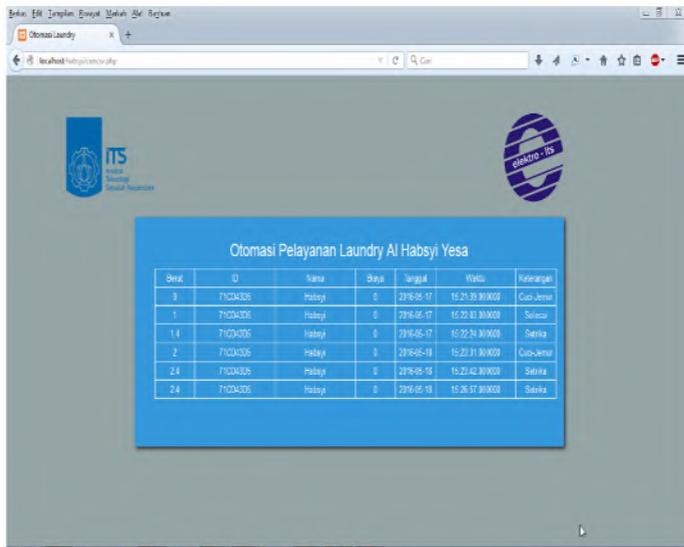
4.2.5 Pengujian *Websver*

Websver berfungsi untuk menampilkan data pelanggan melalui *web*. Pengujian dilakukan dengan menguji coba *web* dengan cara menginputkan nama pelanggan yang diinginkan. Setelah itu data pelanggan akan ditampilkan sesuai *database* yang berada pada memori.

Proses pengujian dilakukan dengan menjalankan *web*. Halaman utama *web* dijelaskan pada gambar 4. 19. Setelah halaman utama *web* terbuka. Nama pelanggan akan dimasukkan. Contohnya : pelanggan dengan nama habsyi. Hasil pengujian *web* dijelaskan pada gambar 4. 20,



Gambar 4. 19 Halaman Utama *Websver*



Gambar 4. 20 Halaman Ketika Nama Pelanggan Setelah Dimasukkan

Hasil pengujian *webserver* menunjukkan *webserver* dapat digunakan sebagai penyedia layanan informasi bagi pelanggan.

BAB V PENUTUP

5.1 Kesimpulan

Pada tugas akhir ini telah dibuat dan dirancang sistem otomatisasi layanan binatu berbasis *Raspberry Pi*, timbangan digital menggunakan *load cell*, RFID sebagai kartu anggota dan webserver sebagai layanan informasi pelanggan. Berdasarkan pengujian yang telah dilakukan, timbangan digital memiliki eror rata-rata 0,88 %. Metode RFID dapat memasukan data pelanggan secara otomatis dengan metode pengambilan nomor identitas jenis *hex* 8 digit. *Raspberry* dapat mengakomodasi aplikasi yang efektif dan efisien untuk menyimpan data pelanggan secara otomatis dengan pemakaian CPU rata-rata 5%. *Webserver* yang digunakan dapat menyediakan informasi layanan binatu bagi pelanggan.

5.2 Saran

Adapun untuk perbaikan dan pengembangan alat kedepannya diantaranya adalah perlu diadakan penelitian lebih lanjut untuk menghilangkan fungsi *Arduino* sehingga mikrokontroller yang dipakai hanya *Raspberry Pi*.

DAFTAR PUSTAKA

- [1] Kurniawati, Y., “Memulai Bisnis Laundry”, STMIK Amikom, Yogyakarta, 2011.
- [2] *Load Cell Application and Test Guideline*, Scale Manufacturers Association, Approved, April, 2010.
- [3] *Load Cell and Weigh Module Handbook*, Rice Lake Weighing Systems, 2010.
- [4] *Datasheets YZC1-B*, Single Point Load Cell.
- [5] *Datasheets 24-Bit Analog-to-Digital Converter (ADC) for Weigh Scales*, AVIA Semiconductor.
- [6] Rachmat, H., H., dan Hutabarat, G., A., “Pemanfaatan Sistem RFID sebagai Pembatas Akses Ruangan”, Jurnal ELKOMIKA, Institut Teknologi Nasional, Bandung, 2014.
- [7] Huang, X., Yeung, Y.,Y., dan Gao, W., “Application of Radio Frequency Identification (RFID) in Science Education”, International Journal of Information and Education Technology, Vol.1, No.3, August, 2011.
- [8] *Datasheets MFRC522*, Company Public.
- [9] Durfee, W., “Arduino Microcontroller Guide”, University of Minnesota, 2011.
- [10] Tambupolon, M., “Interfacing Serial, Paralel, USB Port”, 2011.
- [11] *Datasheets Raspberry Pi 3 Model B*, Raspberry Pi.
- [12] Husni, “Tutorial Lazarus Pemograman Pascal Console, Visual, dan Database”, Universitas Trunojoyo.
- [13] *Build Your Own Database Engine CSV Files*, Available: <http://www.festra.com/eng/les14.htm>
- [14] *XAMPP Tutorial: How to Use XAMM to Run Your Own Web Server*, September, 2013. Available : <https://blog.udemy.com/xampp-tutorial/>

BIODATA PENULIS



Al Habsyi Yesa, lahir di Piladang, 21 November 1995. Penulis memulai jenjang pendidikan di sekolah dasar di SDN 4 Salasa Tengah, setelah lulus SD tahun 2007 penulis melanjutkan ke SMPN 6 Bukittinggi, lulus SMP pada tahun 2010, penulis kemudian melanjutkan ke SMAN 1 Bukittinggi, setelah lulus SMA pada tahun 2012 penulis melanjutkan studi di Institut Teknologi Sepuluh Nopember jurusan Teknik Elektro dengan konsentrasi di bidang studi Teknik Elektronika. Penulis bisadihubungi melalui alamat email: alhabsyi_yesa@yahoo.com