

**Rancang Bangun Sistem *Internet of Things* Pada Timbangan Digital  
Dengan Kapasitas 100 Kg Berbasis ESP32 & Thingsboard IoT  
Platform Untuk Mengukur Massa**

**SKRIPSI**

**Disusun Oleh :  
Gregorius Ivan Satrio Erwandha  
NIM. 175090807111010**

**PROGRAM STUDI: S1 INSTRUMENTASI**



**JURUSAN FISIKA**

**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM**

**UNIVERSITAS BRAWIJAYA**

**MALANG**

**2021**



**LEMBAR PENGESAHAN  
SKRIPSI**

**Rancang Bangun Sistem *Internet of Things* Pada Timbangan  
Digital Dengan Kapasitas 100 Kg Berbasis ESP32 dan  
*Thingsboard IoT Platform* Untuk Mengukur Massa**

**Oleh:**

Gregorius Ivan Satrio Erwandha  
175090807111010

Setelah dipertahankan di depan majelis penguji  
Pada tanggal 12 Juli 2021 dan  
dinyatakan memenuhi syarat untuk memperoleh gelar Sarjana Sains  
dalam bidang Instrumentasi

**Pembimbing I**



Prof. Dr.-Ing. Setyawan P. Sakti, M.Eng.  
NIP. 196508251990021001

**Pembimbing II**



Dr. Eng. Agus Naba, S.Si., MT  
NIP. 197208061995121001

**Mengetahui,**

Ketua Jurusan Fisika  
Fakultas MIPA Universitas Brawijaya



Prof. Dr. rer. nat. Muhammad Nurhuda  
NIP. 196409101990021001

## LEMBAR PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Gregorius Ivan Satrio Erwandha  
NIM : 1750908071111010  
Jurusan : Instrumentasi  
Penulis Skripsi berjudul :

**“Rancang Bangun Sistem *Internet of Things* Pada Timbangan Digital Dengan Kapasitas 150 Kg Berbasis ESP32 dan *Thingsboard IoT Platform* Untuk Mengukur Massa”**

**Dengan ini menyatakan bahwa:**

1. Isi dari Skripsi yang saya buat adalah benar-benar karya sendiri dan tidak menjiplak karya orang lain. Nama-nama yang termaktub di isi dan tertulis di daftar pustaka digunakan sebagai referensi pendukung dalam skripsi ini.
2. Apabila di kemudian hari ternyata Skripsi yang saya tulis terbukti hasil jiplakan, maka saya akan bersedia menanggung segala resiko yang akan saya terima.

**Demikian pernyataan ini dibuat dengan segala kesadaran.**

Malang, 16 Juli 2021

Yang menyatakan,



**GREGORIUS IVAN SATRIO ERWANDHA**

**1750908071111010**



# Rancang Bangun Sistem *Internet of Things* Pada Timbangan Digital Dengan Kapasitas 100 Kg Berbasis ESP32 dan *Thingsboard IoT Platform* Mengukur Massa

## ABSTRAK

Timbangan digital dirancang dengan mengaplikasikan *Internet of Things* (IoT) pada sistem dengan *Thingsboard IoT Platform* yang memiliki sistem pengukuran untuk mendeteksi dan menghasilkan data massa dengan nilai akurasi yang baik juga minim *error*. Studi literatur dilakukan untuk mencapai konsep dasar dan juga mempelajari teori penunjang sistem yang dibutuhkan. Perancangan dan pembuatan alat meliputi perancangan *hardware* dan *software*. Pengujian alat ini untuk memastikan bahwa sistem timbangan digital berjalan dengan baik. Pengambilan data dilakukan dengan mencatat dan menganalisa seberapa akurat nilai data yang didapatkan untuk mengukur massa dan juga minim kesalahan. Berdasarkan pengambilan data pada sistem timbangan berbasis IoT dengan sensor *load cell* didapatkan *error* sebesar 0,08%. Karakteristik dan spesifikasi *load cell* yakni, dihasilkan kenaikan tegangan dengan rata-rata 4,93 mV/ kg. Pada *Thingsboard IoT Platform*, data hasil pengukuran massa akan dimonitor, direkam dan disimpan pada *server* SenTLab. Data tersebut bisa dilihat dengan cara menentukan waktu spesifik pada *Thingsboard*. Sensor dikalibrasi dengan digunakannya pembeban yang memiliki massa yang tetap dan sudah memiliki standar kalibrasi yang sesuai. Kemudian, sistem diintegrasikan dengan *fuse* untuk meningkatkan keamanan dari sistem yang dibuat agar terhindar dari arus pendek. Lalu, dalam penelitian diusahakan pada saat alat atau sistem dirancang, diusahakan juga menggunakan *datasheet* agar dapat memahami spesifikasi komponen-komponen yang akan digunakan untuk diintegrasikan.

**Kata Kunci:** *Load Cell*, *Internet of Things*, Timbangan, *Thingsboard IoT Platform*





# **Internet of Things System Design on Digital Scales with a Capacity of 100 Kg Based on ESP32 and Thingsboard IoT Platform for Measuring Mass**

## **ABSTRACT**

Designing a digital weighing system by applying the Internet of Things (IoT) to a system with the Thingsboard IoT Platform which has a measurement system to detect and produce mass data with good accuracy and minimal errors. Literature studies are carried out to achieve basic concepts and also study the required system support theory. The design and manufacture of tools includes designing hardware and software. Testing this tool to ensure that the digital weighing system is running well. Data collection is done by recording and analyzing how accurate the value of the data is to measure the mass and also the minimum error. Based on data collection on the IoT-based weighing system with a load cell sensor, an error of 0.08% was obtained. The characteristics and specifications of the load cell, that is, the resulting voltage increases with an average of 4.93 mV / kg. On the Thingsboard IoT Platform, mass measurement data will be monitored, recorded and stored on the SenTLab server. This data can be seen by determining a specific time on Thingsboard. The sensor is calibrated by the use of a loader which has a fixed mass and has the appropriate calibration standard. Then, the system is integrated with the fuse to increase the security of the system to avoid short circuits. Then, in research it is endeavored when the tool or system is designed, it is also endeavored to use the datasheet in order to understand the specifications of the components that will be used for integration.

**Keywords: Load Cell, Internet of Things, Weighing Scale, Thingsboard IoT Platform**



## KATA PENGANTAR

Puji syukur kepada Tuhan Yang Maha Esa atas berkat Rahmat dan Karunia-Nya kepada kita semua sehingga kami dapat menyelesaikan laporan skripsi dengan judul “**Rancang Bangun Sistem *Internet of Things* Pada Timbangan Digital Dengan Kapasitas 100 Kg Berbasis ESP32 dan *Thingsboard IoT Platform* Untuk Mengukur Massa**”. Pembuatan laporan skripsi ini disusun sebagai salah satu syarat untuk mengerjakan skripsi pada program Strata-1 di Jurusan Fisika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Brawijaya, Malang. Dalam penyusunan laporan skripsi ini banyak pihak yang turut serta membantu, membimbing dan memberikan dukungan moral maupun materil yang sangat besar kepada penulis sehingga laporan akhir kegiatan praktek kerja lapangan ini dapat terselesaikan. Oleh karena itu, penulis mengucapkan banyak terima kasih kepada :

1. Ayah, Ibu, serta kedua adikku, yang selalu memberikan semangat dan support secara moriil maupun materil.
2. Prof. Dr. rer. nat. Muhammad Nurhuda selaku Ketua Jurusan Fisika Universitas Brawijaya
3. Prof. Dr. Ing. Setyawan P. Sakti, M. Eng. selaku dosen pembimbing di Laboratorium Teknologi Sensor yang selalu membimbing, memberikan nasehat, dan juga mendukung untuk penelitian ini.
4. Dr. Eng. Agus Naba, S.Si., MT selaku dosen pembimbing kedua yang selalu memberikan nasehat dan bimbingan selama penelitian ini dilakukan.
5. Prof. Drs. Arinto Yudi Ponco Wardoyo, M.Sc., Ph.D. selaku dosen pembimbing akademik yang selalu memberikan nasehat dan wejanganannya.
6. Mas Adin Okta Triqadafi sebagai pendamping di Laboratorium Teknologi Sensor yang selalu membantu,



mengajarkan, dan memberikan saran nasehat pada saat penelitian dilaksanakan.

7. Adam, Akhyar, Aldi, Farkhan, Revano, Essam, Fais dan Riyana sebagai teman-teman seperjuangan di Laboratorium Teknologi Sensor yang sangat membantu dalam penelitian ini.
8. Andro, Daffa, Putra, Rio, Yufa, Nivan, Joseph, Herlambang, Intan, dan Firza selaku teman seperjuangan di jurusan yang selalu mendukung dan mendoakan saya selama masa kuliah.
9. Ariel, Salomo, Gerald, Terry, Denny, Ritter, Aubrey, Yonathan, Aurel, Dira, Tasya, Putri, dan Theofila yang turut membantu untuk menghibur dan sebagai penyemangat dalam melaksanakan penelitian tugas akhir ini.
10. Keluarga besar serta teman-teman penulis yang tidak dapat disebutkan satu persatu atas dukungan moril maupun materil yang telah diberikan.

Penulis memahami masih terdapat kekurangan dalam penyusunan laporan skripsi ini, oleh karena itu penulis memohon maaf yang sebesar-besarnya dan membuka saran yang membangun atas laporan ini. Akhir kata semoga laporan ini dapat bermanfaat bagi kita semua.

Malang, 26 Mei 2021

Penulis,



Gregorius Ivan Satrio Erwandha

NIM. 175090807111010



## DAFTAR ISI

<b>LEMBAR PENGESAHAN</b> .....	<b>i</b>
<b>LEMBAR PERNYATAAN</b> .....	<b>ii</b>
<b>ABSTRAK</b> .....	<b>iii</b>
<b>ABSTRACT</b> .....	<b>iv</b>
<b>KATA PENGANTAR</b> .....	<b>v</b>
<b>DAFTAR ISI</b> .....	<b>vii</b>
<b>DAFTAR GAMBAR</b> .....	<b>x</b>
<b>DAFTAR TABEL</b> .....	<b>xii</b>
<b>BAB I PENDAHULUAN</b> .....	<b>1</b>
<b>1.1 Latar Belakang</b> .....	<b>1</b>
<b>1.2 Rumusan Masalah</b> .....	<b>4</b>
<b>1.3 Batasan Masalah</b> .....	<b>4</b>
<b>1.4 Tujuan Penelitian</b> .....	<b>4</b>
<b>1.5 Manfaat Penelitian</b> .....	<b>5</b>
<b>BAB II TINJAUAN PUSTAKA</b> .....	<b>6</b>
<b>2.1 Sensor <i>Load Cell</i></b> .....	<b>6</b>
<b>2.2 <i>Strain Gauge</i> Pada <i>Beam Load Cell</i></b> .....	<b>9</b>
<b>2.11 Pengujian Akurasi Alat</b> .....	<b>12</b>
<b>2.3 Modul Timbangan HX711</b> .....	<b>13</b>
<b>2.4 Mikrokontroler ESP32</b> .....	<b>14</b>
<b>2.5 7-Segment <i>Display</i></b> .....	<b>16</b>
<b>2.6 <i>Buzzer</i></b> .....	<b>18</b>
<b>2.7 <i>Push Button</i></b> .....	<b>20</b>
	<b>vii</b>





<b>2.8 Internet of Things (IoT)</b> .....	<b>21</b>
<b>2.9 Thingsboard IoT Platform</b> .....	<b>25</b>
<b>2.10 Data Sharing</b> .....	<b>26</b>
<b>2.12 Timbangan Digital</b> .....	<b>28</b>
<b>BAB III METODOLOGI PENELITIAN</b> .....	<b>30</b>
<b>3.1 Waktu dan Tempat Penelitian</b> .....	<b>30</b>
<b>3.2 Alat dan Bahan</b> .....	<b>30</b>
<b>3.3 Prosedur Penelitian</b> .....	<b>31</b>
3.3.1 Studi Literatur .....	31
3.3.2 Perancangan Alat .....	32
3.3.2.1 Perancangan Perangkat Keras .....	32
3.3.2.2 Perancangan Perangkat Lunak .....	34
3.3.2.3 Perancangan Desain Tempat Sistem .....	35
3.3.2.4 Perancangan <i>Dashboard IoT Platform</i> .....	36
3.3.3 Pembuatan Alat .....	37
3.3.4 Pengujian Alat .....	37
3.3.5 Pengambilan Data .....	37
3.3.6 Analisa Hasil .....	38
<b>BAB IV HASIL DAN PEMBAHASAN</b> .....	<b>39</b>
<b>4.1 Desain Sistem</b> .....	<b>39</b>
4.1.1 Hasil Perancangan Perangkat Keras .....	39
4.1.2 Hasil Perancangan Perangkat Lunak .....	41
<b>4.2 Pengujian Perangkat Keras dan Perangkat Lunak</b> .....	<b>54</b>
4.2.1 Pengambilan Data Untuk Uji Kalibrasi Sistem .....	54



4.2.2 Pengambilan Data Untuk Uji Kesalahan ( <i>Error</i> ) Pada Sistem .....	59
<b>4.3 Pembahasan dan Analisa Pengujian Sistem .....</b>	<b>62</b>
4.3.1 Pengujian Kalibrasi Pada Sistem .....	62
4.3.2 Pengujian Kesalahan ( <i>Error</i> ) Pada Sistem .....	64
<b>BAB V PENUTUP.....</b>	<b>66</b>
5.1 Kesimpulan .....	66
5.2 Saran.....	66
<b>DAFTAR PUSTAKA.....</b>	<b>67</b>



## DAFTAR GAMBAR

Gambar 2.1 Sensor strain gauge beam load cell.....	6
Gambar 2.2 Mengukur kapasitas truk.....	7
Gambar 2.3 Strain gauge pada beam load cell .....	9
Gambar 2.4 Skema rangkaian jembatan wheatstone .....	10
Gambar 2.5 Load Cell .....	11
Gambar 2.6 Modul timbangan HX711 .....	13
Gambar 2.7 Mikrokontroler Pada ESP32.....	14
Gambar 2.8 Struktur Sistem Pada ESP32.....	15
Gambar 2.9 Skema pin LED 7 segment .....	16
Gambar 2.10 Skema pin karakter 7 segment.....	17
Gambar 2.11 Common katoda (kiri) common anoda (tengah) dan 4 pin (kanan).....	18
Gambar 2.12 Komponen buzzer.....	18
Gambar 2.13 Piezo pada komponen buzzer .....	19
Gambar 2.14 Elektroda kutub negatif dan positif pada buzzer.....	19
Gambar 2.15 Rangkaian skematik pada push button.....	20
Gambar 2.16 Chart kegunaan dari Internet of Things .....	21
Gambar 2.17 Sistem arsitektur dari Internet of Things .....	24
Gambar 2.18 Tampilan Situs Thingsboard Pada Server SenTLab ...	25
Gambar 2.19 Sistem Timbangan Digital Berskala Advanced .....	28
Gambar 3.1 Diagram blok prosedur penelitian.....	31
Gambar 3.2 Rangkaian skematik EAGLE pada timbangan digital ..	32
Gambar 3.3 Rangkaian Papan PCB Pada Aplikasi EAGLE.....	33
Gambar 3.4 Diagram flowchart pada Arduino IDE.....	34
Gambar 3.5 Hasil Desain Tempat Sistem Pada Autodesk Inventor ..	35
Gambar 3.6 Tahap Perancangan Dasbor Pada Situs Thingsboard IoT Platform .....	36
Gambar 4.1 Hasil Rancangan Sistem Pada Rangkaian Papan PCB ..	39
Gambar 4.2 Hasil Akhir Rancangan Pada Perangkat Keras.....	40
Gambar 4.3 Hasil Akhir Rancangan Tempat Sistem Perangkat Keras .....	41
Gambar 4.4 Hasil Akhir Perancangan Dashbor Thingsboard IoT Platform Pada Perangkat Lunak .....	53





Gambar 4.5 Sensor Load Cell dikalibrasi menggunakan massa air sebagai kalibrator.....	54
Gambar 4.6 Hasil Pengukuran Pada Massa Air Maksimal 38 kg Pada Situs Thingsboard.....	57
Gambar 4.7 Grafik Perbandingan Keluaran Massa Dengan Tegangan Keluaran .....	63
Gambar 4.8 Grafik Perbandingan Massa Aktual dengan Massa Keluaran .....	64



## DAFTAR TABEL

Tabel 1. Data Hasil Pengujian Pada Kalibrasi Sistem.....	55
Tabel 2. Spesifikasi Sensor <i>Load cell</i> .....	58
Tabel 3. Data Hasil Pengujian Kesalahan ( <i>Error</i> ) Pada Sistem.....	60



# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Seiring perkembangan zaman saat ini, IPTEK (Ilmu Pengetahuan dan Teknologi) berkembang sangat pesat, kemajuan IPTEK sangatlah membantu berbagai aktivitas manusia dan memberi manfaat maupun mempermudah yang sangat tak terbatas dari yang dihasilkan oleh pengembangan teknologi tersebut. Selain itu, perkembangan teknologi juga telah menjangkau hampir seluruh aspek kehidupan manusia. Digitalisasi alat ukur analog merupakan salah satu contoh dan hampir diterapkan pada semua jenis alat ukur modern. Dari sejumlah piranti alat ukur yang beralih dari sistem analog ke digital tersebut, timbangan juga merupakan salah satunya. Timbangan merupakan alat bantu yang berfungsi untuk mengetahui berat suatu benda. (Afrianto, 2020).

Susu yang dijadikan sebagai objek untuk penelitian pengukuran massa pada timbangan digital ini merupakan salah satu bahan komoditas yang sangat penting dalam kehidupan masyarakat. Untuk memenuhi kebutuhan komoditas, timbangan digital dengan fitur IoT dirancang untuk membantu dan mempermudah proses pencatatan data hasil pengukuran massa susu di pedesaan khususnya pada unit koperasi susu. Susu memiliki prospek pasar yang baik sehingga termasuk dalam komoditas unggulan nasional. Selain itu, susu juga merupakan salah satu peran penting untuk manusia dalam menyempurnakan istilah empat sehat lima sempurna yang dikenal sebagai jargon untuk memenuhi kebutuhan gizi yang seimbang pada manusia, supaya tubuh tetap sehat secara jasmani. Istilah tersebut yang dimaksud adalah nasi, lauk-pauk, sayur, buah, dan susu.

Warna pada air susu murni dapat berubah dari satu warna ke warna yang lain. Faktor yang mempengaruhinya adalah sumber





ternaknya, jenis pakan, jumlah lemak, bahan padat, dan juga bahan pembentuk warna. Air susu murni yang baik dan sehat memiliki warna yang berkisar dari putih kebiruan hingga kuning keemasan. Air susu memiliki massa jenis yang sedikit lebih besar daripada air putih atau bening, yaitu 1,027-1,035 kg/liter dengan rata-rata yang dimiliki 1,031 kg/liter yang dimana air putih atau bening memiliki massa jenis sebesar 1 kg/liter. Berdasarkan codex susu, berat jenis pada air susu lebih tepatnya pada angka 1,028 kg/liter. Codex susu merupakan daftar satuan yang harus dipenuhi pada air susu sebagai bahan makanan yang dimana sudah disepakati oleh para ahli gizi dan kesehatan dunia, walaupun di samping itu setiap negara atau daerah mempunyai regulasi atau ketentuan tersendiri. Kemudian, massa jenis susu juga dipengaruhi oleh zat penyusunnya, penambahan bahan kering tanpa lemak, atau pengurangan lemak juga akan mempengaruhi massa jenis air susu. Penetapan massa jenis berfungsi untuk diketahuinya bahan kering, bahan kering tanpa lemak yang terdapat dalam kandungan susu. Selain itu, juga untuk menduga banyaknya air dalam kandungan susu. Selain itu, terdapat viskositas atau kekentalan susu yang sama seperti massa jenis. Kekentalan air susu berkurang pada suhu tinggi pada 100,16 derajat celcius, dan sebaliknya akan bertambah pada suhu rendah di suhu titik beku air normal ada -0,53 hingga 0,56 derajat celcius, begitu pula jika susu mengalami goncangan, kekentalan susu akan berkurang. Kekentalan susu secara umum, banyak dipengaruhi oleh kasein, butiran lemak dan derajat asam pada kandungan susu. (Sukmawati, 2014)

Untuk mengukur berat objek, dari penelitian sebelumnya dilakukan perancangan dan pembuatan sistem timbangan digital yang memiliki fitur IoT (*Internet of Things*) yang dapat membantu dalam mencatat data berat yang akan dikirimkan ke dalam internet yaitu *Web Server* menggunakan situs *thingspeak*. Timbangan digital dapat dirancang dengan menggunakan *load cell* sebagai sensor gaya yang banyak digunakan dalam industri yang memerlukan peralatan untuk



mengukur massa. Secara umum, *load cell* dan sensor gaya berisi pegas (*spring*) logam mekanik dengan mengaplikasikan beberapa *foil* metal *strain gauge*. *Strain* dari pegas mekanik muncul sebagai pengaruh dari pembebanan yang kemudian ditransmisikan pada *strain gauges*. Pengukuran sinyal yang dihasilkan dari *load cell* adalah dari perubahan resistansi *strain gauge* yang linear dengan gaya yang diaplikasikan. *Load cell* sebagai sensor yang dimana massa, gaya berat, dan massa dapat dideteksi dapat memiliki kemampuan mengukur berat badan dengan maksimal timbangan 200 kg. Data dari sensor *load cell* dapat dikirim ke *database* pada situs *thingspeak* menggunakan perangkat keras yaitu, modul *Wifi* nodemcu esp8266-12E dan arduino. Dengan modul tersebut pengguna dapat memonitor dan mengontrol data yang diinginkan dan informasi data ditampilkan pada komputer atau *smartphone* masing-masing pengguna dalam bentuk *website*.(Asnawi, 2019).

Berdasarkan uraian permasalahan diatas maka pada penelitian kali ini, dibutuhkan suatu alat bantu yang bisa menyelesaikan masalah yaitu, timbangan digital dengan menggunakan mikrokontroler ESP32 sebagai pusat pengendalian sistem dan pengelolaan data sensor, dan juga menggunakan jaringan internet/*Wi-Fi* dan ditampilkan ke dalam *server* dengan menggunakan *NAS server* dari Synology tipe DS220+ yang sudah terdapat *platform IoT* yaitu *Thingsboard*, sensor beban *load cell* sebagai pendeteksi beban, HX711 sebagai penguat sinyal masukan yang dimana data analog yang diperoleh akan diterima oleh ADC akan diubah menjadi data digital. Data digital tersebut akan diproses kembali oleh bahasa C++ pada arduino IDE untuk menampilkan hasil pengukuran massa pada 7-segment dengan satuan kilogram (kg). Kemudian, kelebihan muatan beban pada sensor *load cell* maka akan memicu *buzzer*, dan LED. Oleh karena itu, akan dilakukannya penelitian dengan topik tugas akhir yang berjudul “Rancang Bangun Sistem *Internet of Things* Pada Timbangan Digital

Dengan Kapasitas 100 Kg Berbasis ESP32 dan *Thingsboard IoT Platform* Untuk Mengukur Massa”.

## 1.2 Rumusan Masalah

Berdasarkan latar belakang diatas, maka didapatkan rumusan masalah untuk penelitian ini adalah sebagai berikut:

1. Bagaimana merancang sistem timbangan digital dengan mengaplikasikan *Internet of Things* (IoT) untuk mengukur massa, dan menampilkan juga menyimpan data pengukuran pada *Thingsboard IoT Platform*?
2. Bagaimana cara menghasilkan sistem timbangan digital dengan mengaplikasikan *Internet of Things* (IoT) pada sistem dengan *Thingsboard IoT Platform* yang memiliki sistem pengukuran yang baik pada *load cell* untuk mendeteksi dan menghasilkan data massa dengan nilai akurasi yang baik juga minim *error*?

## 1.3 Batasan Masalah

Pada penelitian ini ditetapkan beberapa batasan masalah, yaitu:

1. Penelitian secara fokus dan ditekankan pada akurasi nilai yang dihasilkan dan kesalahan (*error*) dari model rangkaian dan sistem timbangan digital dengan fitur IoT.
2. Sensor *load cell* digunakan untuk mengukur massa.
3. Massa maksimal yang dapat diukur oleh sensor *load cell* sebesar 100 kg.
4. Menggunakan Synology DS220+ sebagai penyimpanan data pada *server* SenTLab dan *Thingsboard IoT Platform*.

## 1.4 Tujuan Penelitian

Tujuan dari Penelitian ini adalah menghasilkan sistem timbangan digital dengan mengaplikasikan *Internet of Things* (IoT)





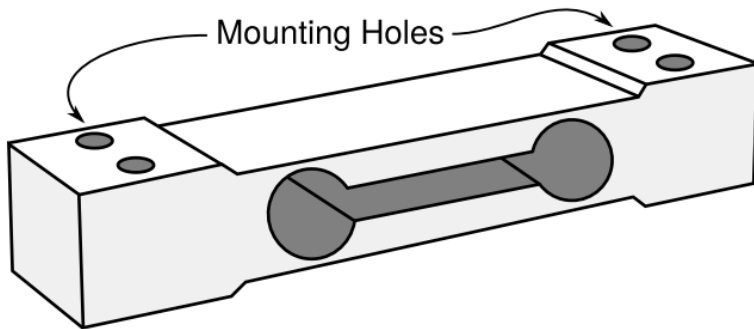
yang memiliki sistem pengukuran untuk mendeteksi dan menghasilkan data massa dengan nilai akurasi yang baik, dan juga minim *error*.

### 1.5 Manfaat Penelitian

Diharapkan dari hasil penelitian ini dapat menghasilkan timbangan digital dengan mengaplikasikan fitur teknologi *Internet of Things* yang dapat membantu proses penimbangan massa yang dapat dimonitor pada *Thingsboard IoT Platform* pada komputer/PC maupun *smartphone* masing-masing pengguna.

## BAB II TINJAUAN PUSTAKA

### 2.1 Sensor *Load Cell*



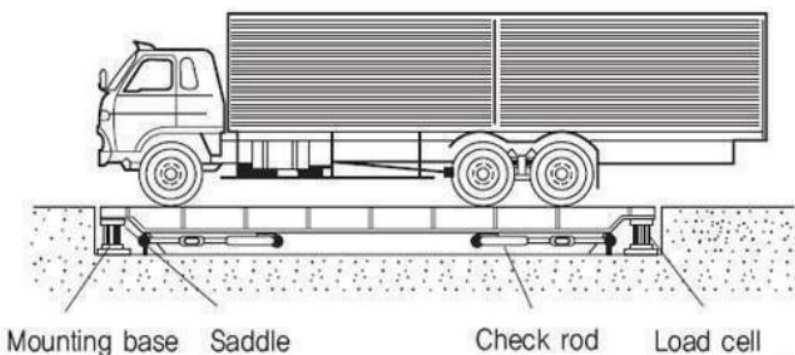
Gambar 2.1 Sensor *strain gauge beam load cell*

(Dunn, 2016)

*Load Cell* merupakan transduser gaya atau tekanan memiliki fungsi yang dapat diaplikasikan di industri penelitian dan pengembangan dalam sains dan teknologi. Sebagian besar, banyak digunakan pada industri kendaraan, kedirgantaraan, mesin industri, robotika dan otomasi pada industri, konstruksi, pertambangan, tekstil, militer, dan juga pemantauan lalu lintas. Selain itu, *load cell* banyak digunakan untuk pengukuran beban atau berat pada industri makanan dan produk hewan. Banyak mesin industri modern, turbin, jembatan, bendungan, yang banyak diinstrumentasikan oleh *load cell* baik untuk pemantauan kesehatan struktural dan juga memeriksa gaya yang diberikan. Dalam aplikasi tersebut, terdapat sifat gaya dalam beberapa aplikasi ini sedikit tidak pasti sehingga *load cell* dapat menghadapi gaya besar yang tidak dapat diduga sehingga menyebabkan kerusakan sensor. Misalnya, dilaporkan bahwa *load cell* untuk objek pembobotan menunjukkan kegagalan keluaran setelah dipakai dalam jangka waktu

yang lama, bisa dikatakan sekitar kurang lebih satu tahun penggunaan.(Aghili, 2010)

*Load cell* atau sensor berat terutama digunakan di perusahaan manufaktur kendaraan untuk menghitung kapasitas kendaraan setelah produksi setiap kendaraan. Pada umumnya, *load cell* di jembatan timbang digunakan yang berbasis *piezo* atau *strain gauge*. ini ditanamkan ke permukaan batang dan beban yang diciptakan oleh setiap sumbu diukur. Inovasi terkini adalah yang disebut dengan teknologi timbangan (WIM) dimana truk dapat ditimbang secara akurat tanpa perlu berhenti. sistem ini menggunakan kombinasi sel beban dan loop induktif, yang mendeteksi keberadaan kendaraan. mereka cepat dan akurat dan yang terpenting menghilangkan kebutuhan setiap truk untuk berhenti ditimbang. hal ini mengatasi masalah cadangan lalu lintas yang dialami pada waktu sibuk yang seringkali memaksa penutupan sementara jembatan timbang. Seperti halnya sistem truk, tersedia untuk pengukuran statis dan WIM. ini dapat menentukan beban gandar individu, beban *bogey*, dan bahkan berat seluruh gerobak atau lokomotif. sel beban digunakan dalam sistem ini dan memiliki akurasi +/- 1% atau lebih baik.(Vishnupriya, 2020)



Gambar 2.2 Mengukur kapasitas truk

(Vishnupriya, 2020)



Pada Gambar 2.2, dapat dilihat bagaimana *load cell* atau sensor berat dipasang untuk menghitung kapasitas kendaraan. Dalam hal ini, sel beban dipasang di bawah permukaan bidang kayu atau bidang logam bersama dengan beberapa peralatan seperti alas pemasangan, pelana, dan batang pemeriksa. Sekarang kendaraan akan ditempatkan di atas bidang kayu atau bidang logam pada saat itu kompresi yang diberikan oleh kendaraan akan dideteksi oleh *load cell* dan terakhir menghitung kapasitas kendaraan. (Vishnupriya, 2020)

*Load cell* digunakan pada beberapa jenis alat ukur seperti timbangan laboratorium, timbangan industri dan komersial, timbangan *platform*, dan jembatan timbang. Ada banyak konfigurasi, tetapi yang paling populer dan fokus penelitian kami adalah *Strain Gauge Load Cell* (SGLC) yang menunjukkan contoh khas SGLC, elemen sel beban lentur ganda. SGLC mengukur regangan dan kemudian mentransfer gaya itu menjadi energi listrik yang dimanifestasikan sebagai pengukuran untuk interpretasi. Mengukur efek regangan membantu menjaga integritas unit di bawah tekanan serta melindungi peralatan dan orang di sekitarnya. (Bwalya, 2019)

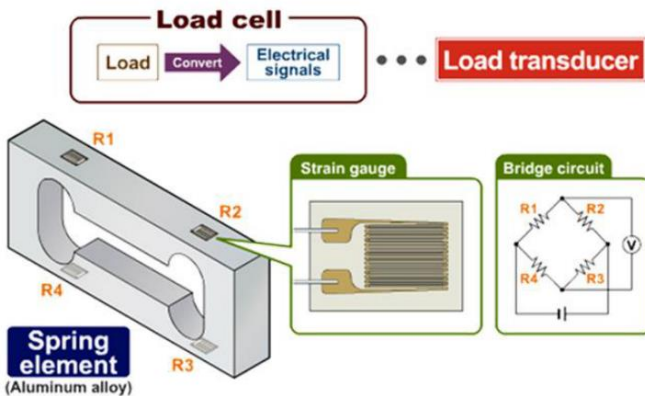
*Load cell* digunakan sebagai input dan hasil pembacaan digunakan diproses oleh Arduino untuk ditampilkan dalam nilai densitas. Dalam prosesnya dibutuhkan HX711 agar keluaran *load cell* dapat terbaca dan diproses oleh Arduino. Modul tersebut juga berfungsi sebagai penguat keluaran *load cell* dengan range mV. HX711 memiliki 6 pin input yang digunakan oleh *load cell* yaitu E +, E-, A-, A +, B-, dan B +. Namun, hanya E +, E-, A-, dan A + yang digunakan dalam penelitian ini. Setiap pin secara berturut-turut dihubungkan dengan kabel merah, hitam, hijau, dan putih dari sel pemuatan. Selain itu, terdapat empat buah pin keluaran yang akan dihubungkan ke mikrokontroler, yaitu GND, DT, SCK, dan VCC, yang berturut-turut dihubungkan ke pin Arduino. GND dihubungkan



ke pin GND, pin DT 3, pin SCK 2, dan pin VCC 5V ke pin Arduino. (Megantoro, 2020)

## 2.2 Strain Gauge Pada Beam Load Cell

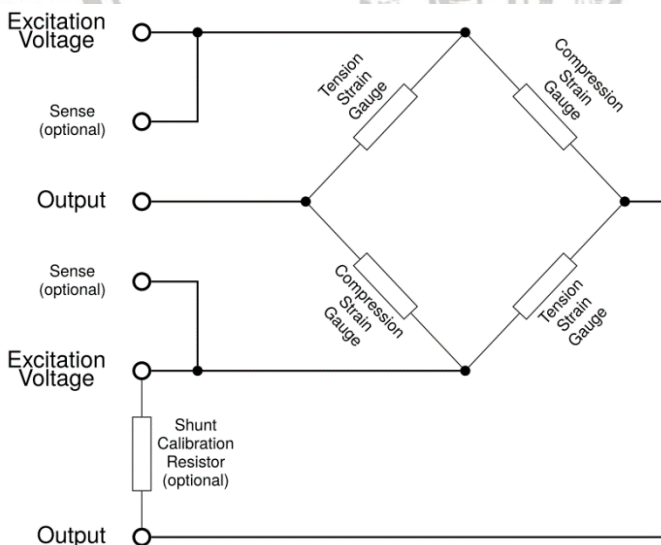
*Load cell* merupakan sepotong logam yang dapat dibengkokkan dengan memanfaatkan gaya mekanis beban dan mengubahnya menjadi sinyal listrik. Lekukan pada *load cell* tidak melewati batas elastisitas logam dan diukur dengan pengukur regangan yang diikat pada titik-titik tertentu pada *load cell*. Selama beban diterapkan di tempat yang tepat pada *load cell*, sinyal listrik akan diberikan oleh pengukur regangan agar sinyal listrik yang didapatkan dapat proporsional. Diketahui bahwa keakuratan *load cell* merupakan salah satu faktor kunci keakuratan timbangan. Teknologi desain dan manufaktur tidak mudah digabungkan secara organik dan *load cell* sangat sensitif terhadap material, mesin, dan proses produksi. Sehingga, dalam membuat *load cell* yang cukup baik adalah yang persyaratannya dapat dipenuhi dengan memiliki keseimbangan akurasi yang tinggi, tetapi *load cell* dapat digunakan secara akurat jika digunakan sesuai dengan kriteria yang dibuat dan dirancang pada sistem. (Thakkar & Prajapati, 2013)



Gambar 2.3 Strain gauge pada beam load cell

(Mcainsh, 2014)

Balok dengan elemen pegas dirancang untuk dapat bekerja hanya pada arah regangan yang diinginkan dapat dikendalikan. Pengukur dapat ditempatkan secara tepat hanya untuk mengukur beban efektif yang diinginkan. Pada gambar diatas, perangkat dirancang untuk menjadi jauh lebih kaku pada bidang halaman sehingga setiap regangan yang disebabkan oleh beban samping yang diabaikan dan tidak digunakan. Lubang menghilangkan tegangan dan regangan geser pada badan, atau permukaan samping dari balok. Pada efek memilin atau puntir, beam dapat dirancang sedemikian rupa agar lebih kaku dan dapat diarahkan. Beban aksial dalam keadaan ini adalah tidak dihiraukan karena semua pengukur akan mencatat tegangan dengan nilai yang sama dan dihilangkan begitu saja dari perhitungan selanjutnya.(Mcainsh, 2014)



Gambar 2.4 Skema rangkaian jembatan *wheatstone*

(Dunn, 2016)

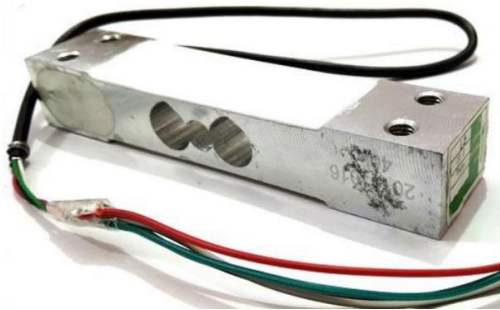
Pada umumnya, teknologi pengukuran yang paling umum yang diterapkan pada *load cell* adalah pengukur regangan *foil*. Sensor yang memiliki keuntungan dalam efisiensi biaya ini dapat mengemas





elemen resistansi atau hambatan yang panjang ke dalam ruang fisik kecil dengan cara yang memaksimalkan kepekaannya terhadap perubahan panjang karena regangan material yang mendasarinya. Penggunaan keempat sensor ini yang dihubungkan bersama dalam jembatan listrik (*electrical bridge*) seperti yang ditunjukkan pada gambar diatas memungkinkan untuk pengukuran tegangan dan kompresi dan mengkompensasi perubahan dimensi yang disebabkan oleh variasi inputan atau masukkan besaran fisis.(Dunn, 2016)

Dalam sel beban pengukur regangan batang, sel diatur dalam formasi "Z" sehingga torsi diterapkan ke batang dan empat pengukur regangan pada sel akan mengukur distorsi tekukan, dua kompresi pengukuran, dan dua tegangan. Ketika keempat pengukur regangan ini dipasang dalam formasi jembatan *Wheatstone*, mudah untuk secara akurat mengukur perubahan kecil dalam hambatan dari pengukur regangan. Sejumlah komponen, kelelahan luar biasa karakteristik. Pengukur regangan adalah perangkat yang mengukur perubahan hambatan listrik sebagai respons, dan gaya proporsional yang diterapkan ke perangkat. Pengukur regangan yang paling umum terdiri dari kawat yang sangat halus, atau *foil*, diatur dalam pola kisi sedemikian rupa sehingga ada perubahan linier dalam hambatan listrik saat regangan diterapkan dalam satu arah tertentu, paling sering ditemukan dengan alas resistansi 120 $\Omega$ , 350 $\Omega$ , dan 1.000 $\Omega$ .



Gambar 2.5 Load Cell  
(Latha and Murthy, 2017)

Setiap *strain gauge* memiliki kepekaan yang berbeda terhadap regangan, yang dinyatakan secara kuantitatif sebagai faktor pengukur (GF). Faktor pengukur didefinisikan sebagai rasio perubahan fraksional dalam hambatan listrik dengan perubahan fraksional dalam panjang (regangan). Faktor pengukur untuk pengukur regangan logam biasanya sekitar 2. (Latha and Murthy, 2017)

## 2.11 Pengujian Akurasi Alat

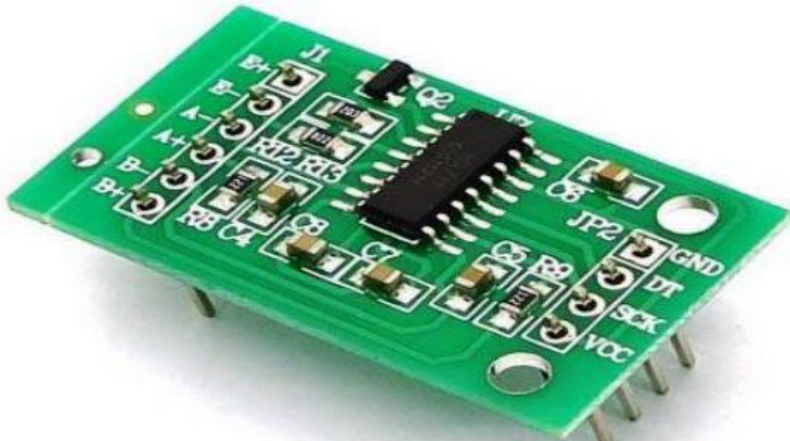
Dalam pengujian suatu alat diperlukan sebuah metode perhitungan khusus untuk dapat mengetahuinya. Setelah dapat diperoleh dari hasil pengujian dan pengukuran berat buah oleh sensor *load cell* dan timbangan konvensional, maka langkah yang harus dilakukan selanjutnya adalah menganalisa data tersebut dengan melakukan perhitungan dengan dianalisisnya nilai presentase (%) keberhasilan dan nilai presentase (%) kesalahan (*error*). Rumus-rumus yang digunakan sehingga presentase kesalahan dan keberhasilan pengukuran sensor *load cell* maupun timbangan konvensional didapatkan berdasarkan persamaan berikut ini:

1. Persentase rata-rata hasil pengukuran *Load Cell* =  $\frac{S1+S2+S3}{3}$
2. Persentase kesalahan pengukuran pada *Load Cell* =  $\frac{S-X}{S} \times 100\%$
3. Persentase keberhasilan pengukuran *Load Cell* =  $\frac{X}{S} \times 100\%$
4. Persentase kesalahan pengukuran pada timbangan manual =  $\frac{X-S}{S} \times 100\%$
5. Persentase keberhasilan pengukuran timbangan manual =  $\frac{S}{X} \times 100\%$



Dimana S1, S2, dan S3 merupakan total penjumlahan dari pengukuran dan pengujian, S adalah nilai hasil ukur dari *load cell* dan timbangan manual dan X adalah nilai rentang atau batasan kapasitas pengukuran. (Wahyudi, Rahman, & Nawawi, 2018)

### 2.3 Modul Timbangan HX711



Gambar 2.6 Modul timbangan HX711

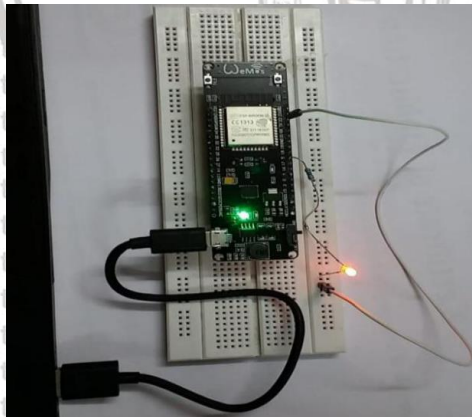
(Akindele, Matthews, and Idowu, 2018)

Modul penguat *load cell* HX711 merupakan konverter analog ke digital presisi tinggi yang memperkuat sinyal keluaran listrik rendah dari *load cell*, dan mengubah sinyal keluaran listrik rendah dari *load cell* yang didapat dari energi mekanik (berat) dan mengubahnya menjadi bentuk digital. Bentuk digital ditransmisikan ke arduino uno untuk menghasilkan bobot. Pada saat penguat *load cell* dihubungkan ke mikrokontroler, perubahan tahanan dari *load cell* akan dibaca oleh mikrokontroler dengan beberapa kalibrasi. Ini menyebabkan pengukuran berat badan sangat akurat. Gambar untuk modul penguat *load cell* yang digunakan pada gambar diatas. (Akindele, Matthews, and Idowu, 2018)



## 2.4 Mikrokontroler ESP32

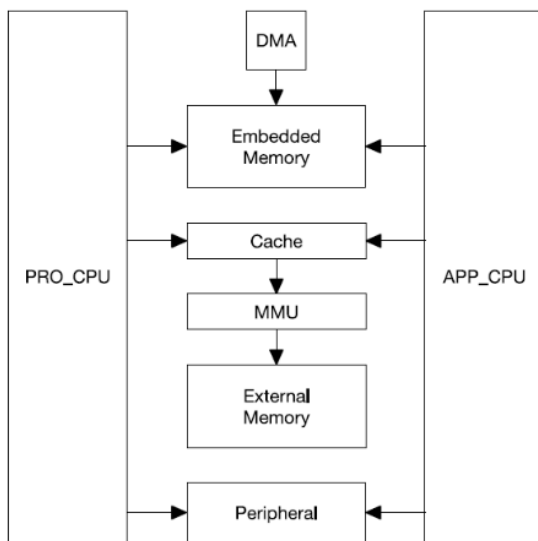
ESP32 merupakan peranti yang memiliki fitur yang dapat mendukung teknologi aplikasi IoT dalam berbagai hal. ESP32 tersedia dalam berbagai versi yang di mana ramah pengguna atau mudah digunakan. Bahkan, cocok digunakan untuk para pemula untuk kalangan siswa maupun mahasiswa. Dengan menghubungkannya ke breadboard dengan begitu ESP32 dapat digunakan untuk implementasi dan aplikasi pendidikan maupun laboratorium. ESP32-Dev Kit dalam sistemnya terdapat modul Wi-Fi built in untuk menyediakan jaringan Wi-Fi sehingga sensor yang terhubung dapat dipantau secara efisien dari rumah atau daerah terpencil menggunakan smartphone maupun laptop/PC. (Swati and Raghavendra Ra, 2018)



Gambar 2.7 Mikrokontroler Pada ESP32

(Bhadane and Lal, 2018)

Pada gambar merupakan gambar yang menunjukkan papan mikrokontroler ESP32 yang dimana ESP32 memiliki sistem dual-core dengan dua harvard arsitektur Xtensa LX6 CPU. Terdapat memori tertanam, memori eksternal, dan periferal semua itu terletak di bus data dan atau bus instruksi dari CPU ini. Mikrokontroler memiliki dua inti PRO\_CPU untuk protokol dan APP\_CPU untuk aplikasi, namun tujuannya tidak tetap. Ruang alamat untuk bus data dan instruksi adalah 4GB dan ruang alamat periferal 512KB. Memori yang ditanam adalah ROM 448KB, SRAM 520KB, dan dua memori RTC \*KB. Memori eksternal dapat mendukung hingga empat kali 16MB flash. (Maier, Sharp, and Vagapov, 2017)



Gambar 2.8 Struktur Sistem Pada ESP32

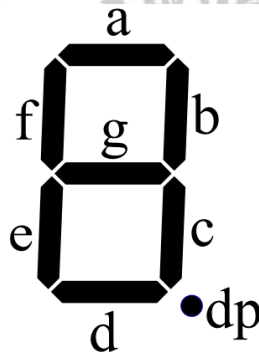
(Maier, Sharp, and Vagapov, 2017)

Struktur sistem pada ESP32 bersamaan dengan dikeluarkannya ESP32, sistem Espressif juga menawarkan modul yang sesuai ESP-WROOM-32. Meskipun ukurannya kecil (25.5 x 18



x 2.8mm), namun modul ini sangat mudah digunakan karena komponen terintegrasi seperti antena, osilator, dan flash. Modul yang sama untuk mikrokontroler lain sering digunakan untuk tes dan prototipe atau bisa juga untuk hobi. (Maier, Sharp, and Vagapov, 2017)

### 2.5 7-Segment Display



Gambar 2.9 Skema pin LED 7 segmen

(Akanle and Oguntosin, 2019)

*Light Emitting Diodes (LEDs)* dan *Liquid Crystal Displays (LCDs)* adalah dua jenis tampilan visual digital yang umum. Untuk tampilan 7-segmen, tujuh dioda pemancar cahaya diatur sedemikian rupa untuk memberikan cahaya setiap kali arus mengalir melalui rangkaian bertenaga. Sebaliknya, LCD bekerja berdasarkan prinsip cahaya terpolarisasi yang diaktifkan ketika cahaya tidak dipantulkan, sehingga huruf, huruf, angka, dan karakter tampak hitam. Namun, inaktivasi menyebabkan insiden cahaya yang dipantulkan menjadi tidak terlihat. Segmen LED dalam tampilan 7 segmen diatur seperti yang ditunjukkan pada Gambar di atas dan dikarakterisasi sebagai a, b, c, d, e, f, g, dan dp seperti yang ditunjukkan. Saat segment LED

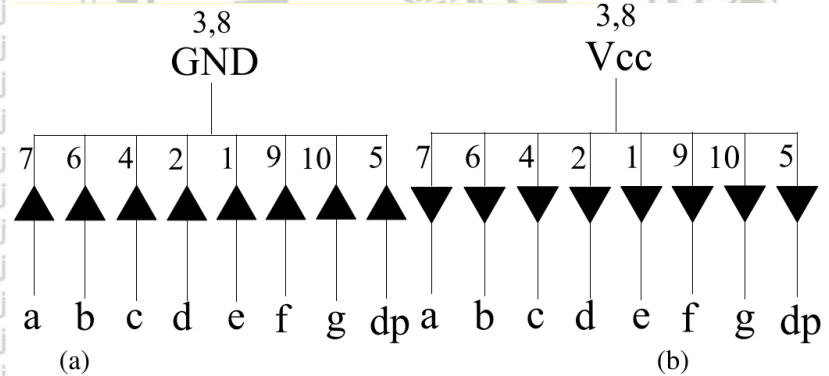




yang dipilih ON atau OFF, huruf atau angka alfabet tertentu dibuat. (Akanle and Oguntosin, 2019)

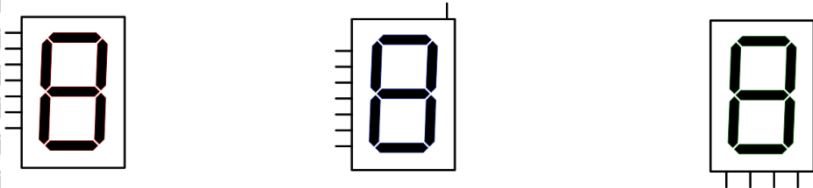
(Akanle and Oguntosin, 2019)

Tampilan *Common Anode* (CA) memiliki anoda dari semua pin LED yang dihubungkan bersama ke 5V, oleh karena itu, tegangan



Gambar 2.10 Skema pin karakter 7 segment

RENDAH yang diterapkan secara individual pada setiap pin akan menyalakan LED dari pin tersebut. Sebaliknya, tampilan *Common Cathode* (CC) memiliki katoda dari semua pin LED yang terhubung bersama ke ground, oleh karena itu, tegangan TINGGI yang diterapkan secara individual pada pin tertentu akan MENYALAKAN LED dari pin tertentu. Gambar 2 menunjukkan struktur internal dari satu pin. digit layar 7-segmen. (Akanle and Oguntosin, 2019)



Gambar 2.11 *Common* katoda (kiri) *common* anoda (tengah) dan 4 pin (kanan)

Tampilan 7 segmen biasanya memiliki 8 pin yang mewakili setiap LED pada layar (7 segmen ditambah titik membuat 8 LED) (Gambar 3a dan b). Namun, terdapat juga tampilan 7 segmen yang disebut *Binary Coded Display* (BCD) yang hanya memiliki 4 pin (Gambar 3c. Dalam hal ini kombinasi biner seperti yang ditunjukkan akan menghasilkan angka 0 - 9 dan huruf A - F sebagai ditampilkan pada 7 segme tersebut.(Akanle and Oguntosin, 2019)

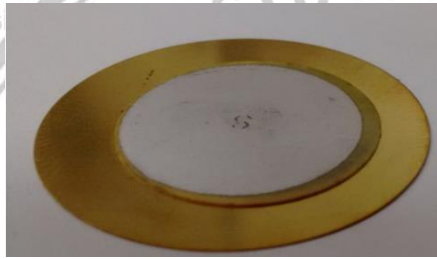
## 2.6 Buzzer



Gambar 2.12 Komponen *buzzer* (Scalet, 2018)

Saat ini, ada beberapa yang tersedia secara komersial bahan berupa disk atau papan yang dapat digunakan untuk membuat transduser. Di antara perangkat yang dibuat dengan PZT, yang paling

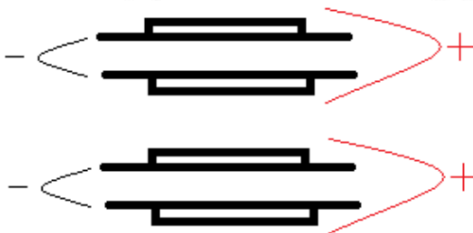
umum adalah bel yang digunakan sebagai sounder peralatan. Gambar 1 menunjukkan foto salah satu *buzzer* ini. (Scalet, 2018)



Gambar 2.13 *Piezo* pada komponen *buzzer*

(Scalet, 2018)

Elektroda tempat energi listrik dikumpulkan diatur pada permukaan datar *buzzer*. *Buzzer* ketika ditekan, mengubah sebagian energi mekanik dari tumbukan menjadi energi listrik, yang beroperasi serupa dengan sumber arus listrik berdenyut. Untuk meningkatkan kapasitas pembangkit arus, *buzzer* ditumpuk membentuk kelompok 4, 6, 8, 10, 12, dan 14 *buzzer* yang dihubungkan secara paralel. Bel yang digunakan dalam pengujian sebelumnya dipilih berdasarkan kesamaan spektrum frekuensi yang mengidentifikasi resonansi fundamental. Spektrum frekuensi diperoleh dengan alat penganalisis impedansi HP4294A, yang dengannya resonansi dan anti-resonansi dari setiap bel diverifikasi. Gambar 2 menunjukkan skema koneksi semacam itu. (Scalet, 2018)

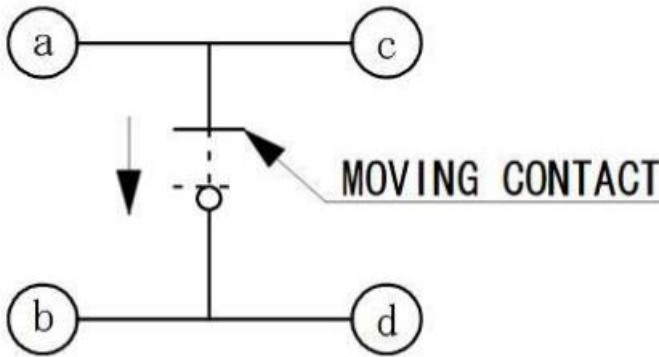


Gambar 2.14 Elektroda kutub negatif dan positif pada *buzzer*

(Scalet, 2018)



## 2.7 Push Button

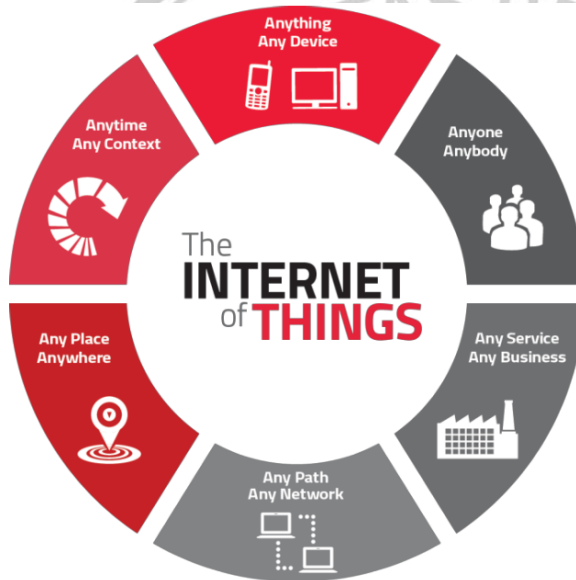


Gambar 2.15 Rangkaian skematik pada *push button*

(Mahbub, 2019)

*Tactile switch Push-Button* adalah sakelar taktil yang biasanya terbuka. Ini memungkinkan kita untuk menyalakan sirkuit atau membuatnya koneksi tertentu hanya ketika tombol ditekan. Sederhananya, ini mengarahkan sirkuit yang terhubung saat ditekan dan memutus sirkuit saat dilepaskan. Ini dapat digunakan untuk memberi makan sistem input analog atau digital MCU untuk mengarahkan MCU untuk pengambil keputusan yang diinginkan. Ini adalah salah satu tombol paling umum yang digunakan dalam komponen elektronik kehidupan sehari-hari kita. Penerapannya mencakup penggunaan dalam kalkulator, telepon tombol tekan, peralatan dapur, kunci magnet, dan berbagai peralatan mekanis dan elektronik baik di rumah maupun iklan. (Mahbub, 2019)

## 2.8 Internet of Things (IoT)



Gambar 2.16 Chart kegunaan dari *Internet of Things*

(Patel & Scholar, 2016)

Saat ini, kita hidup di era teknologi pintar yang mewakili "ubiquitous computing" or "web 0.3". *Internet of Things* (IoT) telah muncul dengan kuat sebagai area yang lebih makmur untuk mengekspresikan teknologi baru semacam ini. Ini bukan teknologi pertama di bidang ini, tetapi juga teknologi komputasi awan telah digunakan untuk mewakili dunia komputasi di mana-mana. Seri ketujuh dari Laporan Internet ITU awalnya diluncurkan pada tahun 1997 dengan judul "*Challenges to the Network*", dan pertama kali diciptakan oleh Kevin Ashton dalam jurnal RFID 1999, pada tahun 2005 nama ini diubah menjadi "*Internet of things*". Visi IoT menurut visi Kevin adalah memungkinkan perangkat berjejaring untuk menyebarkan informasi mereka tentang objek dunia fisik melalui situs. Dalam beberapa tahun terakhir, sebagian besar arsitektur yang

diusulkan IoT digunakan, semantik situs untuk mempublikasikan informasi melalui jejaring sosial. Sebagai contoh, iPhone memiliki inovasi layanan Nike+ iPod untuk mencatat informasi dan mempublikasikannya di jejaring sosial dan jejaring sosial teman.(Zozo & Ali, 2015)

Pada dasarnya, definisi IoT bervariasi berdasarkan dengan siapa Anda berbicara, tetapi secara formal, IoT dapat didefinisikan sebagai infrastruktur jaringan global yang dinamis dengan konfigurasi diri dan komunikasi yang dapat dioperasikan. Sederhananya, IoT berarti kemampuan untuk membuat segala sesuatu di sekitar kita mulai dari (yaitu mesin, perangkat, ponsel, dan mobil) bahkan (kota dan jalan) diharapkan dapat terkoneksi ke internet dengan perilaku yang cerdas dan dengan memperhatikan keberadaan jenis otonomi dan privasi. Sementara itu, lingkungan IoT mengandung sejumlah besar objek / hal yang berbeda yang dapat diklasifikasikan menjadi dua jenis yaitu, hal-hal yang dapat diisi ulang baterai: sebagian besar adalah ponsel misalnya laptop, tablet, dan ponsel, dan hal-hal yang tidak dapat diisi ulang: benda-benda ini statis dari sudut pandang mobilitas. Secara umum, IoT mencakup tiga tuntutan utama yaitu: pertama, pemahaman bersama tentang situasi penggunaannya dan aplikasinya. Kedua, arsitektur perangkat lunak dan jaringan komunikasi yang menyebar mencakup dan memproses informasi kontekstual, dan terakhir, alat analitik di IoT yang bertujuan untuk perilaku otonom dan cerdas.(Zozo & Ali, 2015)

Karakteristik dasar IoT adalah sebagai berikut:

- Inter-Konektivitas: Berkenaan dengan IoT, apa pun dapat dihubungkan dengan infrastruktur informasi dan komunikasi global.
- Layanan terkait hal: IoT mampu menyediakan layanan terkait hal dalam batasan hal-hal, seperti perlindungan privasi dan



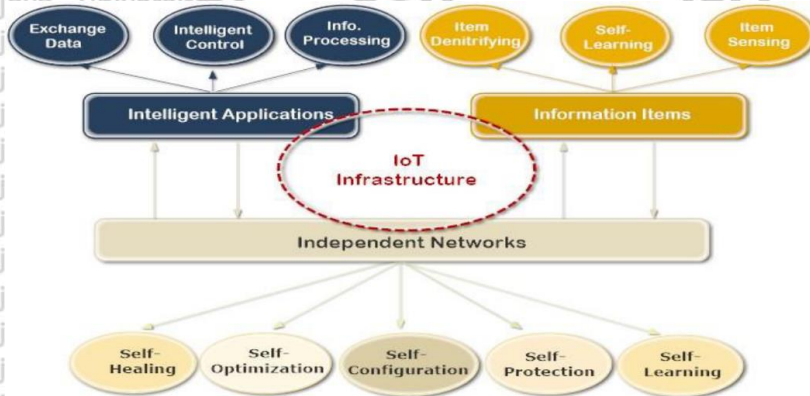


konsistensi semantik antara hal-hal fisik dan hal-hal virtual terkait. Untuk menyediakan layanan yang berhubungan dengan hal dalam batasan hal, baik teknologi di dunia fisik dan dunia informasi akan berubah.

- Heterogenitas: Perangkat di IoT bersifat heterogen berdasarkan *platform* dan jaringan perangkat keras yang berbeda. Mereka dapat berinteraksi dengan perangkat lain atau *platform* layanan melalui jaringan yang berbeda.
- Perubahan dinamis: Status perangkat berubah secara dinamis, mis., Tidur dan bangun, terhubung dan / atau terputus serta konteks perangkat termasuk lokasi dan kecepatan. Apalagi jumlah perangkat bisa berubah secara dinamis.
- Skala yang sangat besar: Jumlah perangkat yang perlu dikelola dan yang berkomunikasi satu sama lain setidaknya akan memiliki urutan yang lebih besar daripada perangkat yang tersambung ke Internet saat ini. Yang lebih penting adalah pengelolaan data yang dihasilkan dan interpretasinya untuk tujuan aplikasi. Ini berkaitan dengan semantik data, serta penanganan data yang efisien.
- Keamanan: Saat kita mendapatkan manfaat dari IoT, kita tidak boleh melupakan keamanan. Sebagai pencipta dan penerima IoT, kita harus merancang demi keamanan. Ini termasuk keamanan data pribadi kami dan keamanan kesejahteraan fisik kami. Mengamankan titik akhir, jaringan, dan data yang bergerak melintasi semuanya berarti menciptakan paradigma keamanan yang akan diskalakan.
- Konektivitas: Konektivitas memungkinkan aksesibilitas dan kompatibilitas jaringan. Aksesibilitas berada di jaringan sementara kompatibilitas menyediakan kemampuan umum



untuk mengonsumsi dan menghasilkan data. (Patel & Scholar, 2016)

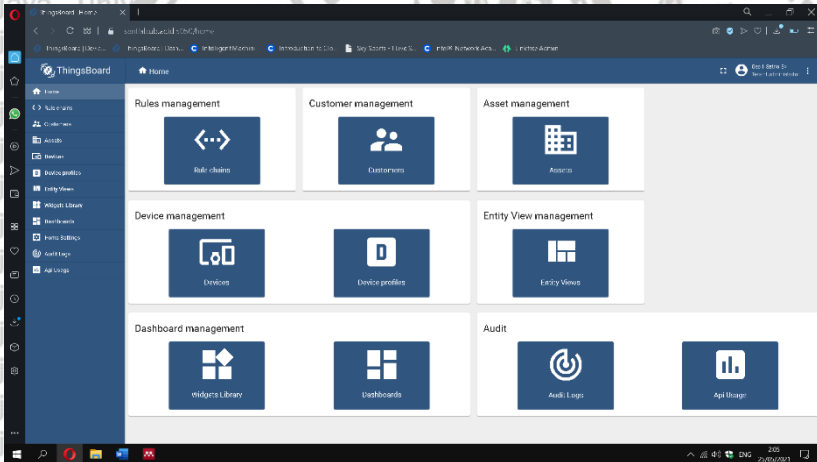


Gambar 2.17 Sistem arsitektur dari *Internet of Things* (Zozo and Ali, 2015)

Desain arsitektur terbaik adalah batu pondasi untuk membangun sistem IoT yang istimewa, arsitektur ini membantu mengatasi banyak masalah di lingkungan IoT seperti skalabilitas, perutean, jaringan, dll. Biasanya, pendekatan arsitektur IoT berdasarkan tiga dimensi utama adalah item informasi, mencakup semua item yang terhubung ke lingkungan IoT mungkin merasakan item, mengidentifikasi item dan item kontrol. Jaringan independen, yang mencakup beberapa fitur seperti konfigurasi diri, perlindungan diri, adaptasi diri, dan pengoptimalan diri dan aplikasi cerdas, yang memiliki perilaku cerdas melalui internet secara umum. Perilaku cerdas dapat berupa kontrol cerdas, metode pertukaran data melalui item jaringan, pemrosesan data, semua aplikasi yang terkait dengan IoT dapat diklasifikasikan menurut dimensi ini. Perpotongan antar dimensi tersebut menciptakan ruang baru bernama “infrastruktur IoT”, yang menyediakan sistem pendukung untuk melayani hal-hal khusus, yang dapat menyediakan berbagai layanan seperti identifikasi barang,

identifikasi lokasi, dan perlindungan data. Gambar 2.17 menggambarkan tiga dimensi IoT dan hubungan di antara mereka. (Zozo and Ali, 2015)

## 2.9 Thingsboard IoT Platform



Gambar 2.18 Tampilan Situs Thingsboard Pada Server SenTLab

ThingsBoard merupakan situs *web* resmi yang berfungsi sebagai *platform* IoT yang terbuka (*open source*) untuk pengumpulan data, pemrosesan, visualisasi, dan manajemen perangkat. Ini memungkinkan konektivitas perangkat melalui protokol IoT standar industri - MQTT, CoAP dan HTTP dan mendukung penyebaran *cloud* dan lokal. ThingsBoard menggabungkan skalabilitas, toleransi kesalahan dan kinerja. (Kukkamäki, 2018)

Thingsboard memberi administrator atau pengguna penyewa dengan antarmuka *web* yang banyak fitur untuk menentukan dan mengelola perangkat. Setiap perangkat diwakili oleh token akses unik sehingga dapat diatur berdasarkan profil dan kepemilikannya. ThingsBoard memungkinkan penetapan aturan, yang dapat diterapkan ke pesan masuk dan *plugin*. Operasi yang dapat diimplementasikan





melalui aturan adalah perbandingan antara nilai yang diterima dan beberapa ambang statis (untuk memeriksa, misalnya, jika suhu telah melebihi ambang batas aman). ThingsBoard menawarkan kemampuan untuk melakukan interoperabilitas di antara kontainer heterogen, aset bernama, di sekitar kita. Thingsboard juga dapat menentukan alarm pengingat untuk aset dan perangkat yang dimana Thingsboard dapat digunakan untuk mengirim pemberitahuan informasi selama fase pemrosesan.(Traboulsi and Knauth, 2020)

Berdasarkan fungsinya, platform IoT ThingsBoard pada saat melakukan proses pengumpulan, pemrosesan, visualisasi, dan manajemen perangkat data. Konektivitas, ThingsBoard mendukung protokol IoT standar seperti MQTT, CoAP, dan HTTP. Manajemen Perangkat, berfungsi dimana ThingsBoard memberi pengguna kemampuan untuk mendaftar, mengelola, dan memantau perangkat yang berbeda. Selain itu, ia menyediakan API untuk aplikasi sisi *server* untuk mengirim perintah ke perangkat dan sebaliknya. Penyimpanan dan visualisasi data, berfungsi dimana ThingsBoard mendukung database seperti HSQLDB, PostgreSQL, dan Cassandra. Dalam kasus kami, Thingsboard dapat menerapkan Cassandra. Ini menawarkan dasbor yang disesuaikan kepada pengguna untuk memantau data secara *real-time* dengan banyak *widget* yang dapat dikonfigurasi. Analisis Data, berfungsi dimana ThingsBoard memiliki mesin perannya untuk yang utama analisis pesan masuk dan dapat diintegrasikan dengan Kafka dan Apache Spark untuk pemrosesan yang lebih kompleks.(Ismail, Hamza, and Kotb, 2019)

## 2.10 Data Sharing

Terdapat beberapa hal yang dapat dimanfaatkan dalam adanya jaringan yaitu;



- *Sharing Perangkat Keras*

- a. *Sharing Printer*

Pemakaian ini secara bersama jelas menghemat biaya dalam pembelian printer karena cukup satu printer saja dapat digunakan oleh beberapa komputer, selain itu juga dapat menghemat waktu dalam pekerjaan untuk mencetak dokumen.

- b. *Sharing Media Penyimpanan*

Dengan menggunakan jaringan LAN, pemakaian secara bersama pada media penyimpanan seperti HDD, disket, ataupun *flashdisk* akan mudah dilakukan. Fasilitas ini membuat pekerjaan akan lebih cepat dan efisien.

- c. *Sharing Data*

Fasilitas ini terdapat pada setiap komputer yang di mana dapat menggunakan data pada komputer lain sehingga akan menghemat waktu dan memudahkan suatu pekerjaan, dan sangat mudah lagi jika jarak antara komputer satu dengan lainnya itu berjauhan.

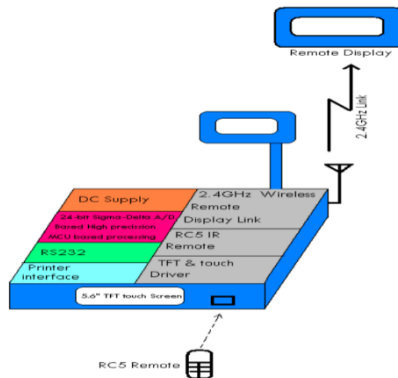
- d. *Sharing Internet*

Pemakaian internet secara bersama-sama sudah sangat efisien pada biaya karena seperti yang dijelaskan di atas bahwa dengan adanya jaringan komputer seperti LAN, pengguna dapat menggunakan dengan cukup satu buah modem atau router dan juga line telepon. (Putra and Sidik, 2018)



## 2.12 Timbangan Digital

Dalam perkembangan tren, dengan digunakannya mesin penimbangan digital daripada sistem penimbangan klasik yang sebagian besar digunakan di dunia, sehingga permintaan meningkat untuk merancang berbagai mesin penimbang digital yang canggih pada tingkat kompetitif dan juga sesuai dengan perkembangan teknologi. Skala timbangan digital banyak digunakannya sensor *load cell* untuk mengukur berat. Sensor tersebut mengubah tekanan menjadi level tegangan yang sesuai. Level tegangan difilter dan diubah menjadi data digital menggunakan 24-bit sigma-delta ADC dan diproses oleh perangkat keras yang ditanam didalam dengan mikroprosesor atau mikrokontroler tertentu bersama dengan berbagai aplikasi yang dikendalikan dengna perangkat keras tertanam. (Kataria and Vaghela, 2012)



Gambar 2.19 Sistem Timbangan Digital Berskala *Advanced*

(Kataria and Vaghela, 2012)

Sistem yang ditunjukkan ada gambar diatas sistem dapat ditunjukkan yaitu timbangan digital yang miliki fitur canggih dengan fitur karies. Sistem ini mencakup panel LCD dengan kontrol sentuh 5,6 inch TFT untuk memberikan antarmuka GUI yang lengkap kepada



pengguna yang lebih mudah dan lebih dapat diandalkan untuk setiap jenis pengguna. Sistem ini memiliki tautan display jarak jauh nirkabel GHz membantu pengguna untuk memantau data penimbangan waktu nyata dari jarak jauh dalam setiap operasi pabrik otomatis, operasi jarak jauh IR yang dipasang dengan sistem untuk dilakukannya operasi yang dimana dikontrol oleh timbangan baik menggunakan layar sentuh atau dengan remote control, dan juga aplikasi yang berguna dan diperlukan konektivitas RS232 dan pilihan pencetakan termal juga dirancang. Tren para perancang timbangan meningkat ke arah akurasi yang lebih tinggi dan biaya produksi yang lebih rendah. Itu akan meningkatkan ketertaikan pada pemrosesan sinyal analog yang bekerja dengan tinggi dan juga dengan biaya yang rendah atau efisien. Cakupan persyaratan ini tidak jelas, sebagian besar timbangan mengeluarkan nilai bobot akhir pada resolusi 1:3000 atau 1:10000 yang mudah dipenuhi oleh ADC 12-bit hingga 14-bit. Namun, pemeriksaan timbangan yang lebih teliti dan cermat menunjukkan bahwa memenuhi resolusi 1:200000 diperlukan untuk pengukuran yang sangat tepat. Persyaratan ini tidak mudah dicapai secara langsung melainkan akurasi ADC harus mendekati 24-bit. Dalam makalah ini, area utama yang dipertimbangkan adalah resolusi noise puncak-ke-puncak, noise RMS, rentang dinamis ADC, penyimpangan penguatan, pemfilteran, dan teknik rata-rata data. (Kataria and Vaghela, 2012)



## BAB III METODOLOGI PENELITIAN

### 3.1 Waktu dan Tempat Penelitian

Penelitian ini dilakukan pada bulan Februari 2021 sampai dengan bulan Mei 2021 di Laboratorium Teknologi Sensor Gedung Fisika Lantai 2 Jurusan Fisika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Brawijaya Malang.

### 3.2 Alat dan Bahan

Alat-alat yang digunakan dalam penelitian ini adalah sebagai berikut:

Perangkat keras:

1. Modul Mikrokontroler ESP32
2. *Personal Computer (PC) / Laptop*
3. *Synology ds220+ NAS server*

Perangkat lunak:

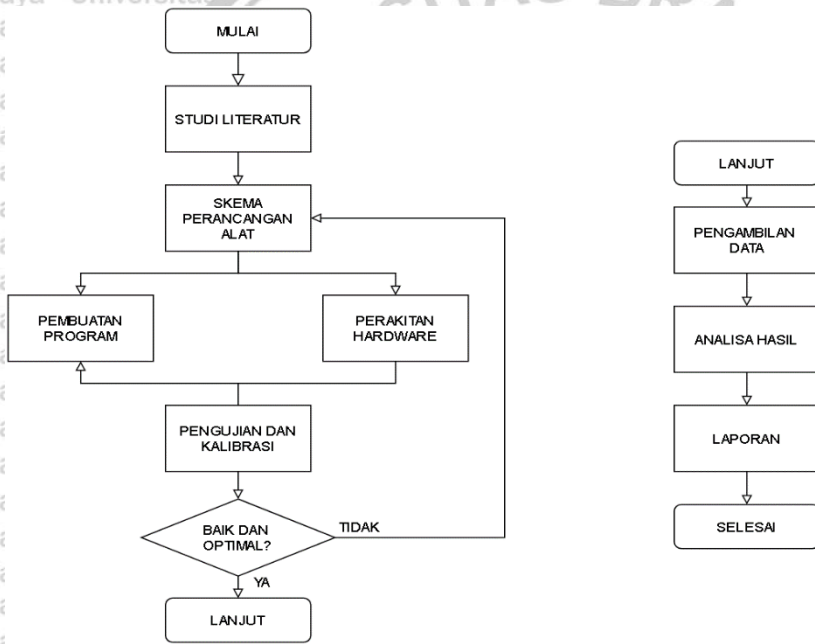
1. *Autodesk EAGLE & Inventor*
2. *Arduino IDE*
3. *Windows Operating System*
4. *ThingsBoard IoT Platform*

Bahan-bahan yang diperlukan dalam penelitian ini adalah sebagai berikut:

1. *Load Cell*
2. *Modul HX711*
3. *TM1637 modul 4-digits 7 Segment*
4. *Active Buzzer*
5. *LED*
6. *Push Button*

### 3.3 Prosedur Penelitian

#### 3.3.1 Studi Literatur



Gambar 3.1 Diagram blok prosedur penelitian

Studi literatur merupakan tahap awal dalam proses penelitian.

Studi literatur dilakukan untuk mencapai konsep dasar dan juga mempelajari teori penunjang sistem yang dibutuhkan dalam perencanaan dan pembuatan alat. Studi literatur dilakukan dengan mencari referensi-referensi yang berhubungan antara lain teori mengenai timbangan digital, sensor *load cell*, Modul HX711, ESP32, IoT dengan prinsip kerjanya, dan lain-lain yang bisa didapatkan melalui artikel, buku, jurnal, maupun tugas akhir dengan narasumber yang jelas dan terpercaya.

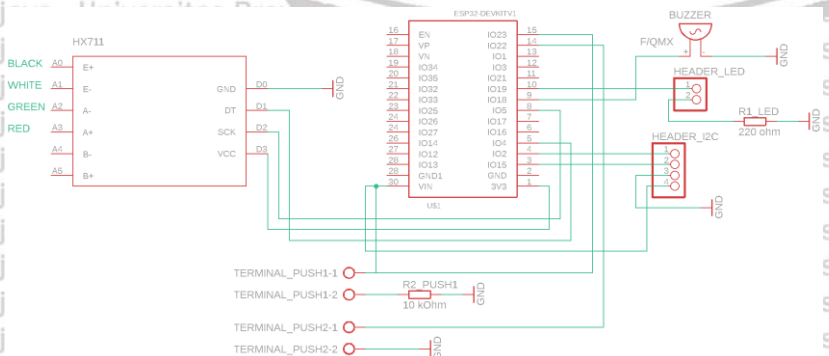


### 3.3.2 Perancangan Alat

Perancangan alat dalam penelitian ini dibagi menjadi dua bagian, yaitu hardware untuk sistem timbangan digital dan *software* untuk membangun sistem *Server* pada IoT. Sebelum melaksanakan pembuatan terhadap alat, dilakukan perancangan terhadap alat yang meliputi merancang rangkaian keseluruhan alat, serta perancangan terhadap *Server*-nya.

#### 3.3.2.1 Perancangan Perangkat Keras

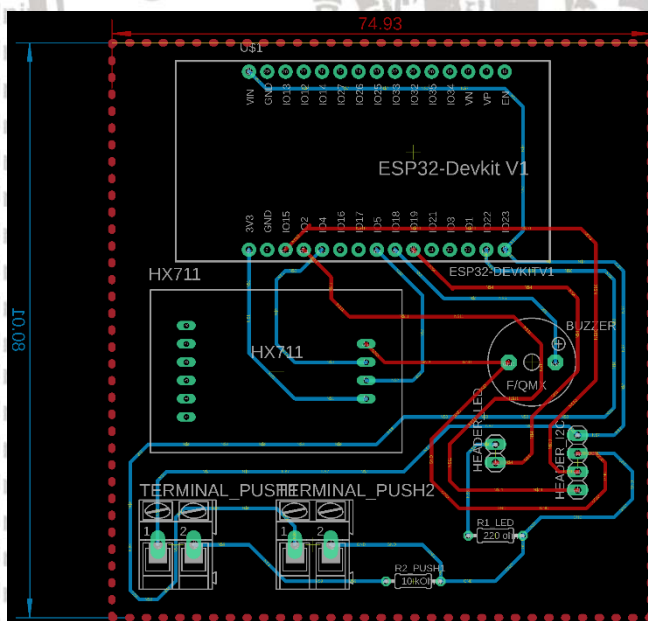
Pada tahap ini, dilakukan perancangan *hardware* yaitu perancangan dalam pembuatan sistem timbangan digital-nya dimana di dalamnya terdapat macam-macam komponen dengan fungsinya masing-masing. Rangkaian sistem timbangan digital terdiri dari komponen *load cell*, modul timbangan HX711 sebagai ADC, ESP32, TM1637 modul 4-digits 7 Segment,, *buzzer*, *push button*, *switch*, dan juga sebuah LED. Lalu, pada *Server* akan dibuat untuk memonitor massa, menginput, dan menyimpannya data input secara *cloud* di *Server*. Berikut adalah skematik untuk perancangan *hardware*:



Gambar 3.2 Rangkaian skematik EAGLE pada timbangan digital

Setelah melakukan desain sistem melalui rangkaian skematik, gambar diatas telah dilakukannya mendesain skematik rangkaian

timbangan digital yang akan dibuat pada penelitian kali ini. Desain gambar diatas menggunakan *software* Autodesk Eagle yang dimana *software* ini digunakan nantinya untuk membuat PCB sebagai tempat agar komponen-komponen dapat disolder dengan rapi dan baik. Gambar skematik diatas menggambarkan adanya komponen-komponen yang akan masuk ke dalam satu PCB yaitu, esp8266-12E, TM1637 modul 4-digits 7 Segment, *push button*, *buzzer*, LED beserta resitornya.



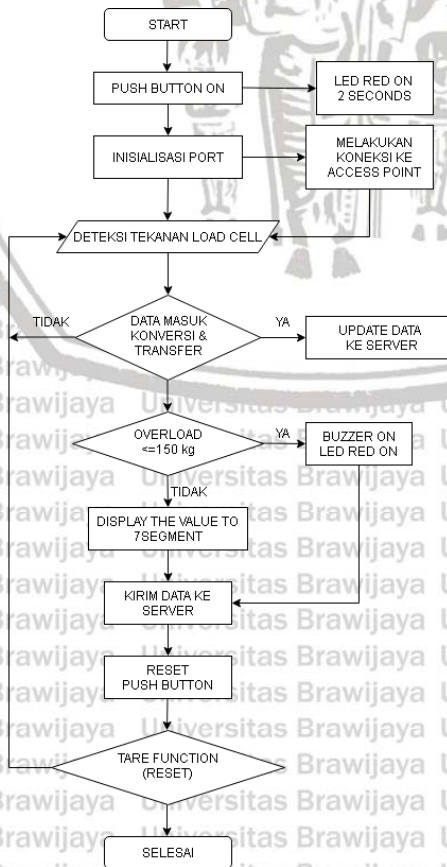
Gambar 3.3 Rangkaian Papan PCB Pada Aplikasi EAGLE

Berdasarkan desain sistem pengukuran pada timbangan digital berbasis IoT ini telah dibuat dengan menggunakan mikrokontroler ESP32 yang terhubung dengan modul timbangan yaitu HX711 dan sensor *load cell* beserta komponen lainnya, yaitu LED, *buzzer*, dua *push-button* kuning dan hijau, *display* 4-digits seven

segment. Kemudian, komponen disolder dan dipadukan dengan papan PCB yang sudah dicetak seperti gambar diatas dari rangkaian skematik yang telah dibuat

### 3.3.2.2 Perancangan Perangkat Lunak

Pada tahap ini, dilakukan pembuatan program untuk menjalankan sistem timbangan digital dalam bentuk *flowchart*. Berikut ini merupakan gambar diagram *flowchart* untuk program di Arduino IDE:

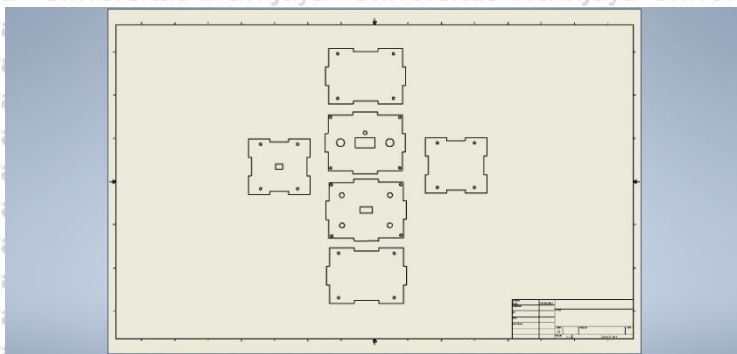


Gambar 3.4 Diagram *flowchart* pada Arduino IDE



Pada Gambar 3.3, flowchart menjelaskan sistem akan mulai berjalan pada saat *push-button* sebagai *switch* ON/OFF untuk mendeklarasikan port atau pin pada ESP-nya. Pada saat keadaan ON, ESP akan menjalankan perintah untuk menginisialisasi port dan menyalakan LED selama 2 detik sebagai indikator bahwa sistem siap digunakan. Kemudian, sensor akan mendeteksi input berupa tekanan atau massa dan jika diberikan tekanan atau massa akan menghasilkan yang di mana akan dikonversi dan ditransfer ke ESP tersebut. Jika tidak, akan menunggu hingga diberikan masukan. Lalu, data sudah dikonversi dan ditransfer maka, ESP akan memberikan perintah siap untuk memperbaharui data pada *server*. Kemudian, jika terjadi kelebihan beban makan ESP akan memicu *buzzer* dan LED sebagai indikator peringatan dan juga mengirim data ke *server*. Lalu, jika tidak terjadinya kelebihan maka, nilai pengukuran akan ditampilkan dalam satuan kilogram pada tampilan 7-segment dan akan mengirim data ke *server*. Setelah terkirim secara otomatis, diperlukan untuk menekan tombol RESET yang berguna sebagai fungsi tare yang di mana fungsi tersebut berfungsi untuk mendapatkan massa bersih pada pengukuran selanjutnya.

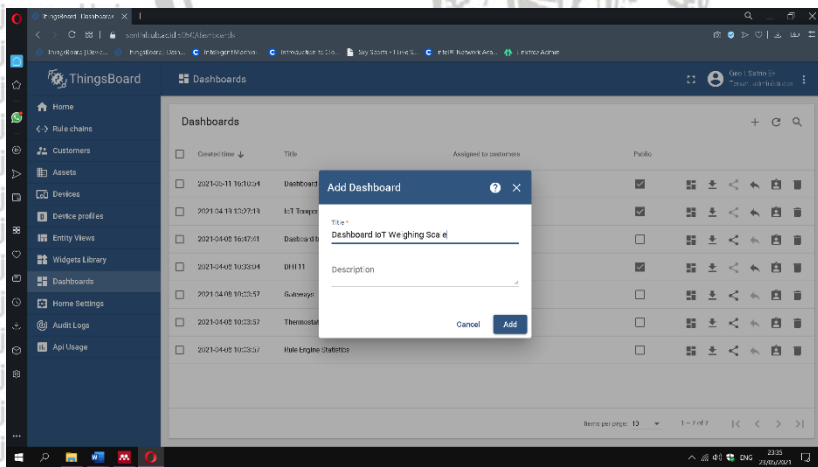
### 3.3.2.3 Perancangan Desain Tempat Sistem



Gambar 3.5 Hasil Desain Tempat Sistem Pada Autodesk Inventor

Pada metode ini yaitu perancangan desain mekanik itu melakukan konsep rancangan untuk tempat/box (casing) yang akan digunakan sebagai wadah sistem berupa PCB dan komponen-komponen yang sudah diintegrasikan atau disolder dengan baik sesuai dengan desain skematik dan desain PCB yang sudah dibuat dan direncanakan. Tujuan dibuatnya tempat agar sistem terlihat baik dan rapih saat digunakan dalam pengoperasiannya. Pada desain dibuat tempat untuk menaruh komponen yang terlihat oleh pengguna, tepatnya dibagian tutup kotak sistem itu terdapat tempat untuk meletakkan display, dua *push-button*, dan LED.

### 3.3.2.4 Perancangan *Dashboard IoT Platform*



Gambar 3.6 Tahap Perancangan Dasbor Pada Situs *Thingsboard IoT Platform*

Pada tahap ini, dilakukan perancangan dashboard IoT pada Thingsboard untuk menampilkan elemen telemetri sebagai penunjuk nilai hasil pengukuran pada saat pengukuran dilakukan atau saat sistem terhubung dengan *server* melalui link address yang sudah

ditentukan melalui program arduino yang diunggah ke dalam ESP32 beserta access point yaitu terhubung ke Wi-Fi.

### 3.3.3 Pembuatan Alat

Setelah melakukan perancangan alat, maka desain dari skematik tersebut akan diterapkan ke pembuatan alat yaitu dibuatnya PCB. Hal yang selanjutnya dilakukan adalah memasang/menyolder komponen yang dibutuhkan dalam rancangan PCB tersebut. Kemudian, merancang *dashboard* Thingsboard IoT Platform untuk menampilkan data hasil pengukuran ke elemen telemetri.

### 3.3.4 Pengujian Alat

Tujuan dari pengujian alat ini untuk memastikan bahwa alat ini telah terkoneksi dengan baik antar komponen dan *server*-nya sehingga sistem timbangan digital dapat digunakan. Selain itu, dilakukan kalibrasi sensor agar sensor dapat membaca massa dengan memiliki akurasi yang baik dan minim kesalahan. Kemudian, massa air digunakan sebagai kalibrator untuk dapat dinyatakan bahwa, timbangan digital yang dirancang ini digunakan untuk mengukur massa.

### 3.3.5 Pengambilan Data

Pengambilan data dilakukan setelah pengujian alat selesai dilakukan. Langkah ini dilakukan untuk memastikan bahwa komponen pada sistem timbangan digital dan *server* telah bekerja sesuai dengan yang diharapkan. Langkah-langkah yang dilakukan untuk pengambilan data adalah meletakkan beban di atas sensor *load cell* yang terdapat di sistem. Selanjutnya memberikan daya agar sistem timbangan digital dapat bekerja dengan cara memasang kabel USB ke *port* USB MINI yang telah disediakan. Untuk pengambilan data massa dapat dilakukan dengan mencatat dan menganalisa seberapa akurat





nilai data yang didapatkan untuk mengukur massa dan juga minim kesalahan (*error*).

### 3.3.6 Analisa Hasil

Setelah selesai melakukan pengambilan data, maka dilakukan analisa hasil. Sehingga dapat disimpulkan bahwa kinerja dari timbangan digital ini bekerja dengan baik. Setelah analisa hasil didapatkan maka selanjutnya adalah penyusunan laporan. Penyusunan ini penting dilakukan supaya hasil dari penelitian ini dapat dipelajari dan digunakan calon pengguna maupun peneliti selanjutnya



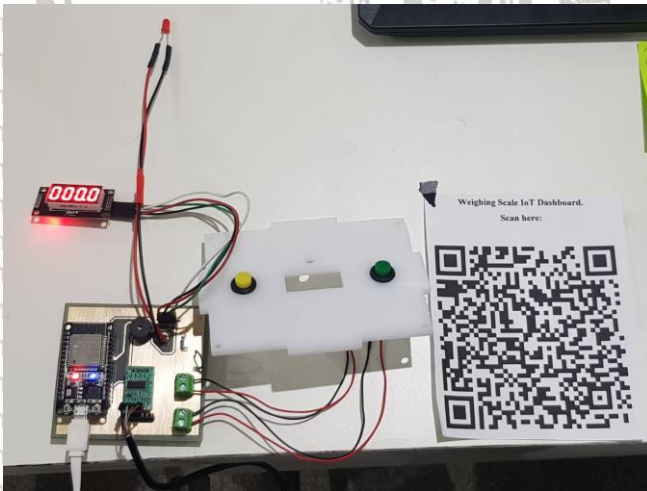
## BAB IV

### HASIL DAN PEMBAHASAN

#### 4.1 Desain Sistem

Desain sistem pengukuran untuk merancang timbangan digital dengan kemampuan IoT, perancangan dibagi menjadi dua bagian diantaranya terdapat perancangan perangkat keras dan perancangan perangkat lunak. Perancangan *software* terdapat juga perancangan dalam pembuatan *dashboard* IoT pada *Thingsboard IoT Platform*.

##### 4.1.1 Hasil Perancangan Perangkat Keras



Gambar 4.1 Hasil Rancangan Sistem Pada Rangkaian Papan PCB

Pada gambar diatas, merupakan gambar wujud sistem yang sudah disolder dan disusun secara benar sesuai yang direncanakan dengan papan PCB yang dimana setelah komponen-komponen telah disolder, disusun, dan dihubungkan pada sistem dan dicoba untuk mencegah terjadinya *error* atau kesalahan pengkabelan pada saat

dimasukkan ke dalam kotak untuk sistem tersebut. Lalu, terdapat ESP32 sebagai mikrokontroler, HX711 sebagai modul ADC, dua buah terminal untuk menjepit kabel dari dua *push-button*, *header* dipasangkan untuk menyambungkan kabel I2C dari display juga kabel pin dan *ground* dari LED, dan juga *buzzer* yang semua komponen sudah disolder ke dalam rangkaian PCB.



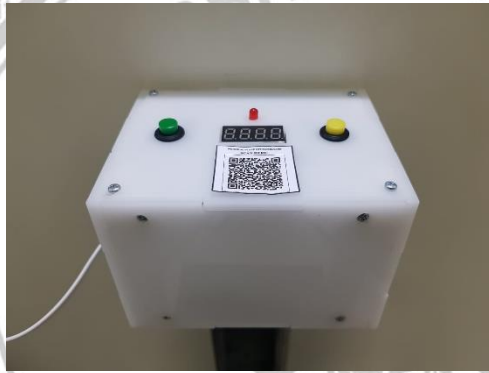
Gambar 4.2 Hasil Akhir Rancangan Pada Perangkat Keras

Setelah semua komponen disolder ke rangkaian PCB, PCB dimasukkan ke dalam *box* atau kotak yang sudah didesain dan sudah dicetak menggunakan laser cutting dengan bahan dasar dinding kotaknya itu terbuat dari akrilik yang memiliki ketebalan 5mm. Kemudian, kotak dibaut sehingga dapat tertutup rapat agar kabel rangkaian PCB tidak berantakan atau tercecer dan juga kotak sistem





dipasangkan di leher badan timbangan dengan dibaut bagian bawah kotak alias alas kotak.



Gambar 4.3 Hasil Akhir Rancangan Tempat Sistem Perangkat Keras

Pada gambar diatas merupakan gambar hasil perancangan tempat sebagai wadah sistem yang dimana untuk meletakkan rangkaian PCB yang sudah disolder dengan komponen-komponen. Kemudian, pada desain perancangan tempat sistem ini dibuat beberapa lubang untuk meletakkan dua *push-button*, LED, dan tampilan 7-Segment. Lalu, dia tiap bagian sisi dinding dan alas kotak wadah sistem didesain dan dibuat beberapa lubang untuk menyatukan antar dinding, alas, tutup, lubang untuk kabel sensor dan juga menaruh kotak di atas leher timbangan.

#### 4.1.2 Hasil Perancangan Perangkat Lunak

```
//TM1637 7segment
#include <TM1637Display.h>
#define CLK_PIN 15
#define DIO_PIN 2
TM1637Display display(CLK_PIN, DIO_PIN);
uint8_t blank[] = { 0x00, 0x00, 0x00, 0x00 };
```

```

const uint8_t done[] = {0xff, 0xff, 0xff, 0xff};
uint8_t turno[] = {
  SEG_A | SEG_B | SEG_C | SEG_D | SEG_E | SEG_F,
  SEG_A | SEG_E | SEG_F | SEG_G,
  SEG_A | SEG_E | SEG_F | SEG_G,
  SEG_A | SEG_E | SEG_F | SEG_G
};
//HX711&LoadCell
#include "HX711.h"
#define LOADCELL_DOUT_PIN 4
#define LOADCELL_SCK_PIN 5
HX711 scale(LOADCELL_DOUT_PIN, LOADCELL_SCK_PIN);
float calibration_factor = 25055;
float weight;
float w;
float maxweight=100;
float almostmax=90;
float minweight=0.1;
//others
int State = 0;
byte mulai;
boolean buttonUp = false;
int count = 0;
//components
int LED = 19;
int buzzer = 18;
//buttons
int buttonPin = 23;
int buttonPin1 = 22;
int buttonNew;
int buttonNew1;
int buttonOld=1;
int buttonOld1=1;
#include "WiFi.h"
#include <ThingsBoard.h>
#define WIFI_AP_NAME "//yourwifi"
#define WIFI_PASSWORD "//yourwifipass"
#define TOKEN "//yourtoken"
#define THINGSBOARD_SERVER "sentlab.ub.ac.id" //account di sentlab

```



```
#define SERIAL_DEBUG_BAUD 115200
```

```
WiFiClient espClient;  
ThingsBoard tb(espClient);  
int status = WL_IDLE_STATUS;  
int quant = 20;  
int send_passed = 0;  
int send_delay = 2000;
```

Pada program diatas, merupakan hasil rancangan software yaitu pemrograman C++ yang menampilkan pendeklarisan variabel, pin digital yang digunakan pada komponen, juga deklarasi variabel agar jaringan Wi-Fi dan *server* Thingsboard agar dapat dihubungkan dengan ESP32 dan juga library yang digunakan pada komponen-komponen. Contohnya, display 7-segment, LED, dua *push-button*, modul ADC HX711, buzzer, dll.

```
void setup() {  
  awal();  
} //end setup  
void awal(){  
  //components  
  pinMode(buttonPin, INPUT_PULLUP);  
  pinMode(buttonPin1, INPUT_PULLUP);  
  pinMode(LED, OUTPUT);  
  pinMode(buzzer, OUTPUT);  
  //display  
  display.setBrightness(0x0f);  
  display.setSegments(done);  
  //loadcell  
  scale.set_scale();  
  scale.tare();  
  long zero_factor = scale.read_average(); //Get a baseline reading  
  Serial.print("Zero factor: ");  
  Serial.println(zero_factor);  
  // Initialize serial for debugging  
  Serial.begin(SERIAL_DEBUG_BAUD);  
  //inisiate WiFi
```





```

WiFi.begin(WIFI_AP_NAME, WIFI_PASSWORD);
InitWiFi();
while (WiFi.status() != WL_CONNECTED) {
Serial.print("Connecting to WiFi..");
Serial.print(".");
delay(500);
}
Serial.println("Connected to the WiFi network");
Serial.print("IP address: ");
Serial.println(WiFi.localIP());
}

```

Kemudian, pada program diatas akan digunakan untuk sebagai awal pada saat program dijalankan. Program pada void setup dibuat fungsi “awal”. Tujuan dibuatkan fungsi, agar program rapih. Jadi, jika terjadi kesalahan atau *error*, proses pencariin syntax yang salah akan mudah ditemukan. Lalu, pada saat program awal dijalankan, program secara otomatis akan mendeklarasikan baudrate untuk menjalan serial monitor, menginisialisaikan port input dan output komponen-komponen dan sensor, dan juga menginisialisasikan Wi-Fi agar langsung terhubung pada saat program awal dijalankan.

```

void loop() {
thingsboard();
pejetan1();
ukur();
sikonukur();
calibration();
tare();
} //end loop

```

Pada fungsi loop, di bagian fungsi program ini fungsi void loop akan mengulang program secara terus menerus. Dalam fungsi



loop terdapat beberapa fungsi yaitu, fungsi thingsboard yang berguna untuk menghubungkan *server* dan juga mengirim data hasil pengukuran ke *server*. Lalu, fungsi pejetan1 berguna untuk mengatur push-button yang dimana push button ini akan mengatur jalannya sistem dengan menggunakan kondisi yang bernama “*state-machine*”. *State-machine* yang digunakan 1 dan 0 yang berfungsi untuk memulai dan menghentikan pengukuran. Kemudian, fungsi ukur berfungsi untuk menjalankan sistem pengukuran dari sensor *loadcell* dan diterima oleh ESP32. Lalu, fungsi sikonukur berfungsi sebagai pengkodisi pengukuran yang akan terjadi pada saat proses pengukuran berjalan. Kemudian, terdapat fungsi *calibration* dan tare yang berfungsi sebagai kalibrasi sensor *loadcell* dan juga fungsi tare yang berguna untuk mendapatkan massa bersih dalam melakukan proses pengukuran berlangsung.

```
void thingsboard(){
if(State==1){
delay(quant);
send_passed += quant;
// Reconnect to ThingsBoard, if needed
while (!tb.connected()) {
// Connect to the ThingsBoard
Serial.print("Connecting ");
Serial.print(THINGSBOARD_SERVER);
Serial.print(" with token ");
Serial.println(TOKEN);
digitalWrite(buzzer, LOW);
delay(500);
digitalWrite(buzzer, HIGH);
delay(100);
digitalWrite(buzzer, LOW);
delay(100);
digitalWrite(buzzer, HIGH);
```



```

delay(100);
digitalWrite(buzzer, LOW);
delay(500);
if (!tb.connect(THINGSBOARD_SERVER, TOKEN)) {
  Serial.println("Failed to connect");
  digitalWrite(buzzer, HIGH);
  delay(1500);
  return;
}
Serial.println("[SENDING DATA TO THINGSBOARD]");
scale.set_scale(calibration_factor); //Adjust to this calibration factor
weight = scale.get_units(),10;
voltage = scale.read();
v = (voltage/16777215.0)*3300;
if (weight<0){
  weight=0;
}
if(weight==0){
  // Send data to thingsboard
  tb.sendTelemetryFloat("Weight(kg)", weight);
  tb.sendTelemetryFloat("Voltage(mV)", v);
}
if (weight>0){
  // Send data to thingsboard
  tb.sendTelemetryFloat("Weight(kg)", weight);
  tb.sendTelemetryFloat("Voltage(mV)", v);
}
send_passed = 0;
tb.loop();
}
//end loop*/

```





Pada program diatas, merupakan program fungsi yang menjalankan perintah untuk terhubung ke Thingsboard IoT Platform. Di dalam fungsi program diatas, diberikan kondisi pada saat ESP32 melakukan perintah untuk terhubung ke Thingsboard dan sudah terhubung ESP32 akan memicu buzzer untuk berbunyi 2 kali sebagai tanda atau indikator jika sudah terhubung ke Thingsboard dan bisa memulai pengukuran. Selanjutnya, setelah terhubung program yang selanjutnya dilakukan adalah melakukan mengunggah data dari pembacaan sensor yang sudah dikonversi dari library yang digunakaa, yaitu HX711. Selain itu, disamping mengunggah data pengukuran dari sensor, terdapat data pengukuran tegangan keluaran dari sensor yang akan ditampilkan dalam Thingsboard IoT Platform.

```
void pejetan1(){
  buttonNew=digitalRead(buttonPin);
  delay(1);
  if(buttonOld==0 && buttonNew==1){
    if (State==0){
      digitalWrite(LED, HIGH);
      display.setSegments(done);
      delay(2000);
      digitalWrite(LED, LOW);
      display.setSegments(blank);
      State=1;
    }
    else {
      digitalWrite(LED, LOW);
      digitalWrite(buzzer, LOW);
      display.setSegments(turnoff);
      delay(5000);//displaying "OFF" for 5s, then
      display.setSegments(blank);
      State=0;
    }
  }
}
```

```
} buttonOld=buttonNew;
```

Pada program diatas, merupakan program fungsi yang berguna untuk menjalankan proses *state-machine* yang dimana *state-machine* tersebut akan dikerjakan jika *push-button* hijau ditekan. Pada saat *push-button* ditekan dan menggunakan *input-pullup* internal, maka akan bernilai kondisi 1, yang dimana akan memicu LED untuk menyala 2 detik dan bisa memulai pengukuran, terhubung ke Thingsboard dan mengirim data pengukurannya. Kemudian, jika *push-button* hijau ditekan sekitar 5 detik pada saat proses pengukuran berlangsung, proses pengukuran akan berhenti dan juga tidak akan terhubung ke dalam Thingsboard. Maka, tidak ada proses pengukuran yang terjadi dan data tidak diunggah ke dalam Thingsboard.

```
void ukur(){  
if(State==1){  
scale.set_scale(calibration_factor);  
voltage = scale.read();  
v = (voltage/16777215.0)*3300;  
w = scale.get_units(5);  
float iw= w/0.1;  
if (weight<0){  
w=0;  
}  
if(weight==0){  
Serial.print("WEIGHT (max.100kg): ");  
Serial.print(w,1);  
Serial.print(" kg");  
Serial.print(" calibration_factor:");  
Serial.print(calibration_factor);  
Serial.print(" Voltage:");
```

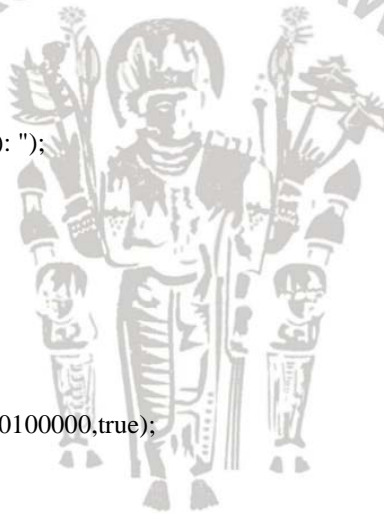


```

Serial.print(v);
display.setBrightness(0x0f);
display.showNumberDecEx(iw,false);
Serial.println();
delay(500);
}
if (weight<0.89){
Serial.print("WEIGHT (max.100kg: ");
Serial.print(w,1);
Serial.print(" kg");
Serial.print(" calibration_factor: ");
Serial.print(calibration_factor);
Serial.print(" Voltage: ");
Serial.print(v);
display.setBrightness(0x0f);
display.showNumberDecEx(iw,0b00100000,true);
Serial.println();
delay(500);
}
if (weight>0.9){
Serial.print("WEIGHT (max.100kg: ");
Serial.print(w,1);
Serial.print(" kg");
Serial.print(" calibration_factor:");
Serial.print(calibration_factor);
Serial.print(" Voltage:");
Serial.print(v);
display.setBrightness(0x0f);
display.showNumberDecEx(iw,0b00100000,false);
Serial.println();
delay(500);
}
}
}
}

```

UNIVERSITAS BRAWIJAYA





Pada program diatas merupakan fungsi ukur yang berguna untuk menjalankan proses pengukuran yang dimana membaca data hasil pengukuran, jika sensor diberikan input. Dalam fungsi ini, terdapat deklarasi dari library HX711 yang dimana syntax scale.read() dan scale.get\_units() membaca data hasil pengukuran yang kemudian dikirimkan ke ESP32 untuk ditampilkan dalam serial monitor dan tampilan 7-Segment. Selain data hasil pengukuran, terdapat juga membaca, mengkonversi dan juga menampilkan data pengukuran tegangan keluaran dari sensor.

```
void sikonukur(){  
  if(State==1){  
    if(weight>=maxweight){  
      digitalWrite(LED, HIGH);  
      digitalWrite(buzzer, HIGH);  
    }  
    else if (weight>almostmax && weight<=maxweight){  
      digitalWrite(LED, HIGH);  
      digitalWrite(buzzer, HIGH);  
    }  
    else if(weight>minweight){  
      digitalWrite(LED, LOW);  
      delay(500);  
      digitalWrite(LED, HIGH);  
      delay(100);  
      digitalWrite(LED, LOW);  
      delay(100);  
      digitalWrite(LED, HIGH);  
      delay(100);  
      digitalWrite(LED, LOW);  
      delay(500);  
      digitalWrite(buzzer,LOW);  
    }  
  }  
}
```



```

else{
display.setBrightness(0x0f);
digitalWrite(LED, HIGH);
delay(250);
digitalWrite(LED, LOW);
delay(2000);
digitalWrite(buzzer, LOW);
}
}
}

```

Kemudian, pada program diatas merupakan program fungsi yang dimana berguna untuk memberikan kondisi pada saat melakukan proses pengukuran berlangsung. Pada program, terdapat kondisi if dan else yang semuanya berguna sebagai indikator jika, pengukuran mendekati batas maksimal yaitu 100kg, maka ESP32 akan memicu LED untuk menyala dan juga *buzzer*. Selain itu, terdapat kondisi yang lain seperti jika terdapat objek diatas penampang timbangan maka ESP32 akan memicu LED menyala berkedip cepat dengan jeda 1 detik tiap kedipan LED. Kemudian, jika tidak ada barang atau objek diatas penampang timbangan maka ESP32 akan memicu LED dengan menyalakan LED untuk berkedip tidak cepat tetapi mati hidup secara konstan sebagai indikator bahwa, alat bersedia dan siap melakukan proses pengukuran.

```

void tare(){
if(digitalRead(buttonPin1)==LOW){
scale.set_scale();
scale.tare(); //reset the scale into 0
}
}
void calibration(){
if(Serial.available())

```



```

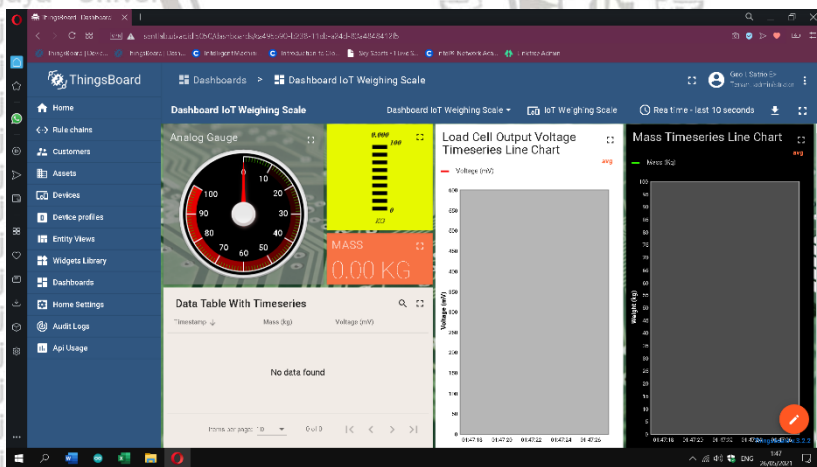
    {
    char temp = Serial.read();
    if(temp == '+' || temp == 'a')
    calibration_factor += 1;
    else if(temp == '-' || temp == 'z')
    calibration_factor -= 1;
    else if(temp == 's')
    calibration_factor += 5;
    else if(temp == 'x')
    calibration_factor -= 5;
    else if(temp == 'd')
    calibration_factor += 10;
    else if(temp == 'c')
    calibration_factor -= 10;
    else if(temp == 'f')
    calibration_factor += 100;
    else if(temp == 'v')
    calibration_factor -= 100;
    else if(temp == 'g')
    calibration_factor += 1000;
    else if(temp == 'b')
    calibration_factor -= 1000;
    else if(temp == 'h')
    calibration_factor += 10000;
    else if(temp == 'n')
    calibration_factor -= 10000;
    else if(temp == 'j')
    calibration_factor += 10000;
    else if(temp == 'm')
    calibration_factor -= 10000;
    else if(temp == 't')
    scale.tare(); //Reset the scale to zero
    }
}

```





Pada program diatas, terdapat program yang menyatakan dua fungsi yaitu fungsi tare dan *calibration*. Pada fungsi tare, program ini akan dijalankan jika *push-button* berwarna kuning dinyalakan atau ditekan dengan memanfaatkan *input-pullup* internal yang dimana jika ditekan akan memberikan nilai LOW dan bisa mendapatkan massa bersih dalam suatu proses pengukuran. Kemudian, fungsi *calibration* berfungsi untuk mengkalibrasi sensor melalui serial monitor dan memasukan karakter yang sudah diprogram secara berbeda, sehingga memiliki nilai faktor kalibrasi yang berbeda tiap karakter yang dimasukkan melalui keyboard komputer/laptop pada perangkat lunak yang digunakan, yaitu Arduino IDE.



Gambar 4.4 Hasil Akhir Perancangan Dashbor *Thingsboard IoT Platform* Pada Perangkat Lunak

Kemudian setelah program telah selesai dirancang dan sudah diunggah ke dalam ESP32, pada gambar diatas merupakan hasil



rancangan pada *dashboard Thingsboard IoT Platform*. Pada rancangan ini, dibuat beberapa elemen yaitu, telemetri, tabel, dan grafik untuk menunjukkan dan menyimpan data hasil pengukuran yang sudah dikirimkan melalui ESP32 yang sudah terhubung dengan jaringan *Wi-Fi* dan *server Thingsboard*. Pada elemen telemetri dibuat untuk mengukur massa analog (*analog gauge*), *digital gauge*, dan elemen tampilan untuk menampilkan data nilai massa yang memiliki dua angka dibelakang koma. Kemudian, dibuat elemen tabel dan grafik yang memiliki interval waktu untuk menampilkan data hasil pengukuran, yaitu berupa data nilai massa dan nilai tegangan keluaran dari sensor. Selain itu, dilakukan dalam penampilan hasil nilai pengukuran pada situs thingsboard dengan tiga angka dibelakang koma dan pada tampilan 7-segment ditampilkan nilai hasil pengukuran satu angka dibelakang koma.

## 4.2 Pengujian Perangkat Keras dan Perangkat Lunak

### 4.2.1 Pengambilan Data Untuk Uji Kalibrasi Sistem



Gambar 4.5 Sensor *Load Cell* dikalibrasi menggunakan massa air sebagai kalibrator

Pada pengujian perangkat keras dan perangkat lunak, dilakukan untuk menyatakan bahwa alat yang dibuat yaitu, timbangan bisa dikatakan untuk mengukur massa. Untuk itu, pada uji pengambilan data untuk kalibrasi dilakukan dengan massa air sebagai kalibrator pada pengkalibrasian sistem. Dengan menggunakan massa air sebagai kalibrator, maka sistem timbangan ini dapat dinyatakan sebagai timbangan digital yang memiliki fitur IoT untuk mengukur massa.

Hasil pengujian yang dilakukan ditunjukkan oleh tabel perbandingan dengan membandingkan nilai keluaran tegangan dengan hasil keluaran pengukuran massa menggunakan ESP32 dan juga digunakan perangkat lunak *Thingsboard IoT Platform*. Pengujian ini menggunakan perangkat lunak *Thingsboard IoT Platform* untuk mendapat nilai akurasi yang baik, dikarenakan pada situs Thingsboard memiliki nilai hasil pengukuran massa tiga angka di belakang koma. Lalu, pengujian sistem ini bertujuan untuk mengetahui rangkaian berfungsi dengan baik ketika dimasukkan massa dan menghasilkan tegangan keluaran yang di mana hasil pengukuran massa dan tegangan ditampilkan di dalam situs Thingsboard. Data hasil pengukuran dipresentasikan dalam bentuk tabel berikut ini.

Tabel 1. Data Hasil Pengujian Pada Kalibrasi Sistem

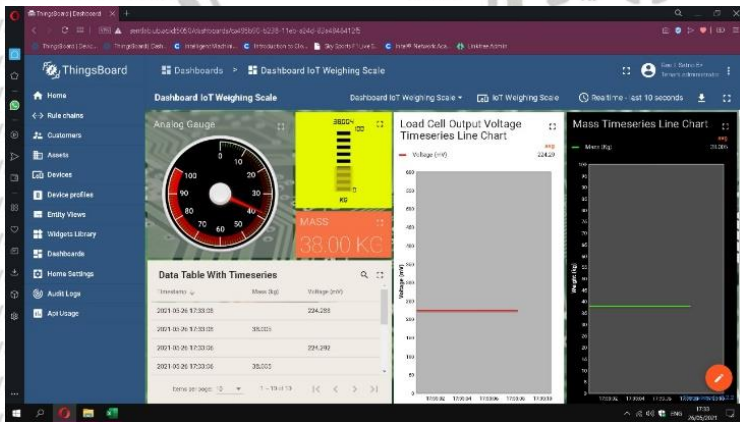
MASSA AIR (kg)	VOLUME AIR (ml)	MASSA OUTPUT (kg)	Vout (mV)
0	0	0	36,99
1	1000	1,013	41,98
2	2000	2,006	46,88
3	3000	3,006	51,81
4	4000	4,009	56,74
5	5000	5,007	61,67
6	6000	6,01	66,61





7	7000	7,007	71,52
8	8000	8,002	76,43
9	9000	9,004	81,37
10	10000	10,002	86,28
11	11000	11,002	91,21
12	12000	12,001	96,12
13	13000	13	101,05
14	14000	14,002	106
15	15000	15,002	110,92
16	16000	16,003	115,86
17	17000	17,002	120,78
18	18000	18,001	125,7
19	19000	19,002	130,63
20	20000	20,005	135,58
21	21000	21,009	140,53
22	22000	22,006	145,44
23	23000	23,007	150,37
24	24000	24,009	155,32
25	25000	25,006	160,23
26	26000	26,004	165,14
27	27000	27,004	170,07
28	28000	28,002	174,99
29	29000	29,003	179,92
30	30000	30,004	184,86
31	31000	31,005	189,79
32	32000	32,004	194,71
33	33000	33,004	199,64
34	34000	34,002	204,56
35	35000	35	209,48
36	36000	36	214,41
37	37000	37	219,33
38	38000	38,005	224,29
AVERAGE =			130,65

Pengukuran dilakukan dengan mencatat tegangan dan massa keluaran pada saat dimasukkan air ke dalam wadah galon. Wadah sudah diukur terlebih dahulu sebelum mengukur massa air dengan menggunakan fungsi tare yang sudah ada di dalam sistem hanya dengan menekan tombol warna kuning pada sistem. Setelah wadah galon diukur, proses pengukuran dilakukan dengan cara menuangkan air ke dalam galon yang di mana air sudah diukur menggunakan gelas ukur. Kapasitas maksimal volume air sebesar 1000ml yang di mana secara teori massa jenis air sebesar 1 kg/L dan 1000ml itu sama dengan 1 liter. Sehingga, didapatkan hasil keluaran massa air dalam kilogram, yaitu 1 liter sama dengan 1 kilogram. Proses pengukuran keluaran tegangan dan massa dicatat hingga berat maksimal sebesar 38 kg yang di mana sesuai dengan kapasitas dua wadah galon 38 liter dan diketahui satu wadah galon memiliki kapasitas volume air sebesar 19 liter.



Gambar 4.6 Hasil Pengukuran Pada Massa Air Maksimal 38 kg Pada Situs Thingsboard

Setelah diukur hingga 38 kg, semua data pengukuran yang sudah diunggah ke situs Thingsboard dicatat dan dimasukkan di dalam excel



untuk diolah dan analisa dari grafik yang diplot dari data tersebut. Dari tabel 4.1, didapatkan data pada massa air 0 kg didapatkan 36,99 mV untuk yang terendah dan tertinggi pada massa 38 kg dihasilkan tegangan keluaran sebesar 224,29 mV, dan juga memiliki rata-rata tegangan keluaran sebesar 130,65 mV.

Berdasarkan rumus

$$V_{selisih} = V_{out_n} - V_{out_{n-1}}$$

dan tabel 1, didapatkan rata-rata kenaikan tegangan per 1 kg dengan ditunjukkan pada tabel berikut ini:

Tabel 2. Spesifikasi Sensor *Load cell*

MASSA AIR (kg)	MASSA OUTPUT (kg)	Vout (mV)	Selisih Tegangan Setiap Naik 1 kg (mV)
0	0	36,99	0
1	1,013	41,98	4,99
2	2,006	46,88	4,9
3	3,006	51,81	4,93
4	4,009	56,74	4,93
5	5,007	61,67	4,93
6	6,01	66,61	4,94
7	7,007	71,52	4,91
8	8,002	76,43	4,91
9	9,004	81,37	4,94
10	10,002	86,28	4,91
11	11,002	91,21	4,93
12	12,001	96,12	4,91
13	13	101,05	4,93
14	14,002	106	4,95





15	15,002	110,92	4,92
16	16,003	115,86	4,94
17	17,002	120,78	4,92
18	18,001	125,7	4,92
19	19,002	130,63	4,93
20	20,005	135,58	4,95
21	21,009	140,53	4,95
22	22,006	145,44	4,91
23	23,007	150,37	4,93
24	24,009	155,32	4,95
25	25,006	160,23	4,91
26	26,004	165,14	4,91
27	27,004	170,07	4,93
28	28,002	174,99	4,92
29	29,003	179,92	4,93
30	30,004	184,86	4,94
31	31,005	189,79	4,93
32	32,004	194,71	4,92
33	33,004	199,64	4,93
34	34,002	204,56	4,92
35	35	209,48	4,92
36	36	214,41	4,93
37	37	219,33	4,92
38	38,005	224,29	4,96
AVERAGE=			4,93

Berdasarkan tabel 2, dapat dikatakan bahwa sensor *load cell* menghasilkan tegangan keluaran yang konstan tiap 1 kg, yaitu di angka 4,91-4,99 mV dengan rata-rata 4,93 mV per kg.

#### 4.2.2 Pengambilan Data Untuk Uji Kesalahan (*Error*) Pada Sistem

Pengujian error pada sistem bertujuan untuk mengetahui berapa presentase kesalahan sistem atau alat dalam mendeteksi massa masukan dan menampilkan hasil pengukuran massa yang dihasilkan



oleh sistem. Dengan mengurangi massa keluaran dengan massa aktual kemudian dibagi dengan massa aktual dan dikali 100, sehingga didapatkan hasil presentase kesalahan sistem dalam mendeteksi massa masukan pada saat proses pengujian ini. Pengujian ini sebagian besar prosesnya sama dengan pengujian kalibrasi. Dengan dilakukannya dengan pertama-pertama mengisi air di gelas ukur dan mengukur volume air dengan gelas ukur, hingga gelas ukur berisi air sebanyak 1000 ml untuk mendapatkan massa air aktual sebesar 1 kg. Pengukuran dilakukan dengan cara dituangnya air dari gelas ukur tersebut ke dalam wadah, yaitu galon hingga mencapai massa 38 kg, dikarenakan masih menggunakan wadah yang sama yaitu dua galon yang tiap galon memiliki kapasitas 19 liter. Data hasil pengukuran dipresentasikan dalam bentuk tabel berikut ini.

Tabel 3. Data Hasil Pengujian Kesalahan (*Error*) Pada Sistem

MASSA AIR (kg)	VOLUME AIR (ml)	MASSA OUTPUT (kg)	ERROR (%)
0	0	0	0
1	1000	1,013	1,30
2	2000	2,006	0,30
3	3000	3,006	0,20
4	4000	4,009	0,23
5	5000	5,007	0,14
6	6000	6,01	0,17
7	7000	7,007	0,10
8	8000	8,002	0,03
9	9000	9,004	0,04
10	10000	10,002	0,02
11	11000	11,002	0,02
12	12000	12,001	0,01
13	13000	13	0,00
14	14000	14,002	0,01



15	15000	15,002	0,01
16	16000	16,003	0,02
17	17000	17,002	0,01
18	18000	18,001	0,01
19	19000	19,002	0,01
20	20000	20,005	0,02
21	21000	21,009	0,04
22	22000	22,006	0,03
23	23000	23,007	0,03
24	24000	24,009	0,04
25	25000	25,006	0,02
26	26000	26,004	0,02
27	27000	27,004	0,01
28	28000	28,002	0,01
29	29000	29,003	0,01
30	30000	30,004	0,01
31	31000	31,005	0,02
32	32000	32,004	0,01
33	33000	33,004	0,01
34	34000	34,002	0,01
35	35000	35	0,00
36	36000	36	0,00
37	37000	37	0,00
38	38000	38,005	0,01
AVERAGE =			0,08

Untuk mendapatkan hasil data perhitungan kesalahan (*error*) dihitung dengan menggunakan rumus sebagai berikut,

$$\%error = \frac{(m_{out} - m_{real})}{m_{real}} \times 100$$

Kemudian dari tabel pengujian diatas, ditampilkan data hasil pengujian kemudian, ditunjukan bahwa hasil pengukuran massa aktual dengan hasil massa yang dideteksi atau keluaran oleh sistem





dan dikirimkan ke situs Thingsboard memiliki rata-rata kesalahan sebesar 0,08%. Dapat ditunjukkan pada massa terendah 0 kg alat mendeteksi sebesar 0 kg, sehingga alat tidak mendeteksi adanya massa diatas penampang timbangan. Selain itu, pada massa tertinggi yaitu sebesar 38 kg, massa yang dideteksi oleh alat dan data dikirim ke Thingsboard kemudian dibaca sebesar 38,005 kg, sehingga hanya memiliki kelebihan massa 5 gr dari massa aktualnya. Dari hasil kesalahan ini, dapat disimpulkan bahwa faktor kalibrasi pada program di Arduino IDE sudah cukup baik dan akurat nilainya, karena hasil pengukuran pada nilai massa aktual hampir sama dengan massa yang dideteksi atau keluarannya hanya kelebihan sekitar 0,001-0,013 kg (1 gr – 13gr).

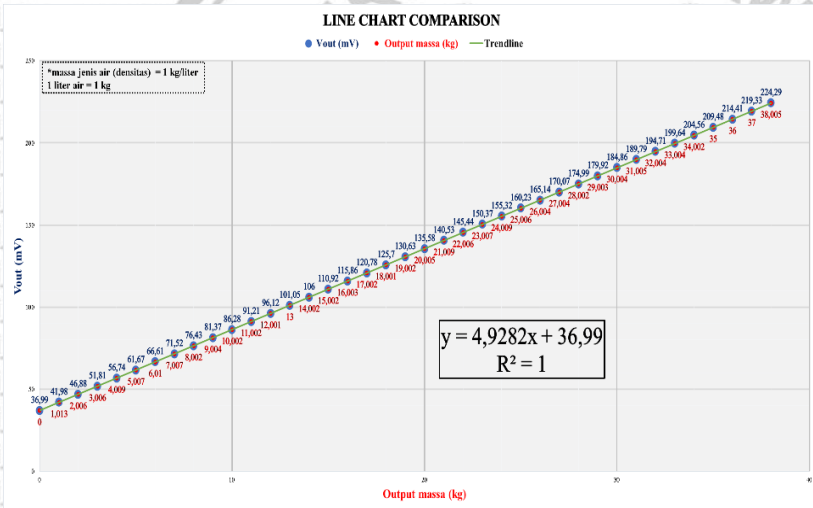
### **4.3 Pembahasan dan Analisa Pengujian Sistem**

#### **4.3.1 Pengujian Kalibrasi Pada Sistem**

Pembahasan dalam pengujian kalibrasi sistem pada sensor *Load Cell* dilakukan dengan *non-zero calibration* pada tegangan, di mana tegangan keluaran yang diukur tidak ditunjukkan nol pada saat tidak diberikan massa sebagai pembeban. Air PDAM sebagai kalibrator digunakan sebanyak 1000ml setara 1 liter volumenya dan juga memiliki massa pada air PDAM sebesar 1 kg, karena pada umumnya massa jenis atau densitas air sebesar 1kg/L. Hasil kalibrasi terhadap 38 L air yang telah diukur menggunakan gelas ukur berkapasitas 1000ml ditunjukkan pada tabel 1. Air yang sudah diketahui massanya digunakan sebagai data kalibrasi *load cell* untuk mendapatkan hubungan karakteristik antara massa, dengan tegangan. Variasi massa diperoleh dengan menambahkan air ke dalam galon yang menyebabkan perubahan tegangan yang diukur dengan ESP32 dan ditampilkan di situs Thingsboard. Hasil kalibrasi *load cell* dengan air ditunjukkan pada tabel 1, yang sama dengan grafik karakteristik yang ditunjukkan pada gambar 4.7. Pada kalibrasi *load cell* ini



dilakukan dengan dituangnya air liter per liter sampai dengan 38 kg dan selanjutnya respon tegangan dicatat. Grafik karakteristik pada pengukuran ditunjukkan pada gambar 4.7 di bawah ini:



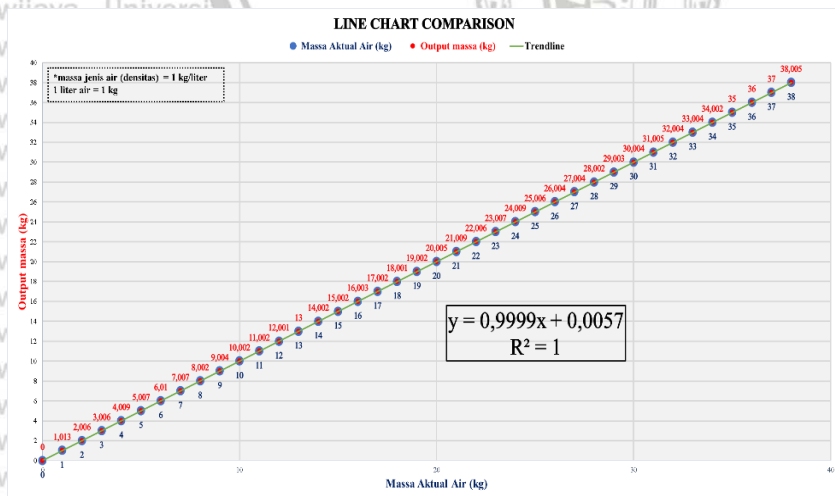
Gambar 4.7 Grafik Perbandingan Keluaran Massa Dengan Tegangan Keluaran

Penempatan air dalam galon diatur sedemikian rupa sehingga *load cell* tersebar secara setimbang. Dalam karakterisasi *load cell*, telah dilakukan penempatan air di atas penampung timbangan dengan digunakannya galon sebagai wadah. Hasil pengukuran dengan dituangkannya air ke dalam galon setiap 1 liter hingga 38 liter memberikan respon tegangan yang setimbang setiap penambahan massa pada galon. Hal ini terjadi karena pembebanan pada salah satu ujung *load cell* pada satu titik ditekan. Dari hasil karakteristik pada *load cell*, pada gambar 4.7 diperoleh persamaan  $y = 4,9282x + 36,99$ . Uji regresi menunjukkan koefisien regresi ( $R^2$  rata-rata) adalah  $R^2 =$



1. Nilai koefisien regresi R sama dengan 1 atau mendekati 1 dan regresi linier  $y = ax + b$ . Nilai  $R^2$  menunjukkan benar bahwa kedua variabel yaitu tegangan dan massa saling mempengaruhi. Persamaan tersebut juga dapat dinyatakan sebagai hubungan antara tegangan dan massa yang selanjutnya akan digunakan untuk antarmuka pada tampilan 7-segment dan situs Thingsboard. Selain itu, dapat diketahui juga bahwa *load cell* telah berunjuk kerja dengan linieritas tinggi dan error histeresis yang sangat rendah. Artinya, alat dan sistem penimbang sesuai dengan yang direncanakan hubungan linier kurva massa dan tegangan menunjukkan kinerja *load cell* baik. Karakteristik *load cell* ini dapat digunakan untuk mengukur massa sesuai dengan judul penelitian ini.

#### 4.3.2 Pengujian Kesalahan (*Error*) Pada Sistem



Gambar 4.8 Grafik Perbandingan Massa Aktual dengan Massa Keluaran





Pembahasan dalam pengujian kesalahan (*error*) sistem pada sensor *Load Cell* dilakukan masih sama digunakan massa air sebagai pembeban. Pengujian ini dilakukan dengan membandingkan massa masukan dengan massa keluaran (terukur) pada situs Thingsboard. Pembacaan hasil pengukuran (massa keluaran) dilakukan pada situs Thingsboard dikarenakan pada situs Thingsboard sudah desain sedemikian rupa agar hasil pengukuran dapat memunculkan angka desimal dan memiliki ketelitian tiga angka di belakang koma. Hasil pengukuran dan pengujian dicatat dan diplot pada grafik seperti gambar 4.8. Hasil pengukuran dengan dituangkannya air ke dalam galon setiap 1 liter hingga 38 liter memberikan massa yang berbeda setelah ditimbang. Berdasarkan tabel 2, ditunjukkan bahwa rata-rata kesalahan sistem dalam membaca massa yang diukur hanya sebesar 0,08%. Kemudian, dari hasil plot grafik yang ditunjukkan pada gambar 4.8 diperoleh persamaan  $y = 0,9999x + 0,0057$ . Uji regresi menunjukkan koefisien regresi ( $R^2$  rata-rata) adalah  $R^2 = 1$ . Nilai koefisien regresi R sama dengan 1 atau mendekati 1 dan regresi linier  $y = ax + b$ . Nilai  $R^2$  menunjukkan benar bahwa kedua variabel yaitu massa aktual air dan massa keluaran saling berdekatan atau juga bisa dikatakan hampir sama. Selain itu, dapat diketahui juga bahwa sistem atau timbangan telah dirancang berunjuk kerja dengan minim error, dan akurat dalam pembacaan beban atau massa. Artinya, alat dan sistem penimbang sesuai dengan yang direncanakan antara hubungan linier kurva massa aktual air dan massa keluaran menunjukkan kinerja timbangan itu dalam kondisi baik dan terkalibrasi. Karakteristik ini dapat digunakan untuk mengukur massa yang di mana sesuai dengan judul penelitian ini.



## BAB V PENUTUP

### 5.1 Kesimpulan

Berdasarkan penelitian yang telah dilakukan dapat disimpulkan:

1. Pada penelitian ini telah dirancang timbangan digital berbasis ESP32 sebagai mikrokontroler dan sensor *load cell* yang memiliki fitur Internet of Things (IoT) dengan menggunakan Thingsboard IoT Platform sebagai dasbor IoT-nya. Pada Thingsboard, data hasil pengukuran massa akan dimonitor, direkam dan disimpan pada server SenTLab, yang di mana data tersebut bisa dilihat lagi dengan cara menentukan hari, tanggal, dan waktu yang spesifik pada Thingsboard.
2. Sistem timbangan digital yang telah dirancang berbasis IoT dengan menggunakan sensor *load cell* didapatkan bahwa error yang dihasilkan sebesar 0,08% dan sensor *load cell* memiliki karakteristik dan spesifikasi yakni, setiap kenaikan massa maka tegangan yang dihasilkan (keluaran) akan mengalami kenaikan tegangan dengan rata-rata 4,93 mV/kg.

### 5.2 Saran

Pada perancangan sistem IoT pada timbangan digital berkapasitas 100 kg berbasis ESP32 dan *Thingsboard IoT Platform* untuk penelitian selanjutnya, disarankan untuk mengkalibrasi sensor digunakan pembeban yang memiliki massa yang tetap (tidak berubah) dan sudah memiliki standar kalibrasi yang sesuai. Kemudian, pada sistem diintegrasikan dengan *fuse* (sekring) untuk mencegah hal-hal yang tidak diinginkan, lebih tepatnya untuk meningkatkan keamanan dari sistem yang dibuat agar terhindar dari arus pendek.

## DAFTAR PUSTAKA

- Afrianto, Rahmad Bahrudin. 2020. *Timbangan Digital Otomatis*. Jurusan Teknik Elektro, Fakultas Teknik, Universitas Muhammadiyah Surakarta: 19. <http://eprints.ums.ac.id/81965/>.
- Aghili, Farhad. 2010. *Design of a Load Cell with Large Overload Capacity*. Transactions of the Canadian Society for Mechanical Engineering 34(3–4): 449–61. [https://www.researchgate.net/publication/229018703\\_Design\\_of\\_a\\_load\\_cell\\_with\\_large\\_overload\\_capacity](https://www.researchgate.net/publication/229018703_Design_of_a_load_cell_with_large_overload_capacity).
- Akanle, Matthew B., and Victoria Oguntosin. 2019. *A Digital Indicator System With 7-Segment Display*. Journal of Physics: Conference Series 1299(1). [https://www.researchgate.net/publication/336310027\\_A\\_digital\\_indicator\\_system\\_with\\_7-segment\\_display](https://www.researchgate.net/publication/336310027_A_digital_indicator_system_with_7-segment_display).
- Akindele, E. Ayoola, O. Victor Matthews, and K. Oriyomi Idowu. 2018. *Development of an Electronic Weighing Indicator for Digital Measurement*. International Research Journal of Engineering and Technology (IRJET) 5(9): 19–25. [https://www.researchgate.net/publication/327471393\\_Development\\_of\\_an\\_Electronic\\_Weighing\\_Indicator\\_for\\_Digital\\_Measurement/link/5b911b4a45851540d1d40ff9/download](https://www.researchgate.net/publication/327471393_Development_of_an_Electronic_Weighing_Indicator_for_Digital_Measurement/link/5b911b4a45851540d1d40ff9/download).
- Asnawi, R, C Nugraha, B Hertanto, and F Surwi. 2019. Development and Testing of Microcontroller-Based Learning Media for the Internet of Things Lab Work. *Journal of Physics: Conference Series* 1413(1). <https://iopscience.iop.org/article/10.1088/1742-6596/1413/1/012007/pdf>.





Bhadane, Pravin, and Aparna Lal. 2018. *Beginners Approach to the Open-Source Programming Case Study Arduino with ESP32*. International Journal of Computer Sciences and Engineering 6(10): 445–48.

[https://www.google.com/url?sa=t&ret=j&q=&esc=s&source=web&cd=&cad=rja&uact=8&ved=2ahUKEwihgKG3zN\\_wAhUMqksFHRgDDooQFnoECBIOAA&url=http%3A%2F%2Fwww.ijcseonline.org%2Fpub\\_paper%2F74-IJCSE-05015.pdf&usg=AOvVaw0joHtPRrfjAtyCdrPG25i9](https://www.google.com/url?sa=t&ret=j&q=&esc=s&source=web&cd=&cad=rja&uact=8&ved=2ahUKEwihgKG3zN_wAhUMqksFHRgDDooQFnoECBIOAA&url=http%3A%2F%2Fwww.ijcseonline.org%2Fpub_paper%2F74-IJCSE-05015.pdf&usg=AOvVaw0joHtPRrfjAtyCdrPG25i9).

Bwalya, Apolinalious, Jackson Phiri, Monica M. Kalumbilo, and David Zulu. 2019. *Passenger and Luggage Weight Monitoring System for Public Transport Based on Sensing Technology: A Case of Zambia*. International Journal of Advanced Computer Science and Applications 10(6): 480–89.

[https://www.researchgate.net/publication/334185439\\_Passenger\\_and\\_Luggage\\_Weight\\_Monitoring\\_System\\_for\\_Public\\_Transport\\_based\\_on\\_Sensing\\_Technology\\_A\\_Case\\_of\\_Zambia/link/5d2856a0458515c11e27691e/download](https://www.researchgate.net/publication/334185439_Passenger_and_Luggage_Weight_Monitoring_System_for_Public_Transport_based_on_Sensing_Technology_A_Case_of_Zambia/link/5d2856a0458515c11e27691e/download).

Dunn, Brian. 2016. *Load Cells: Introduction and Applications*. BD Tech Concepts LLC: 1–11.

[http://bdtechconcepts.com/portfolio/load\\_cells.pdf](http://bdtechconcepts.com/portfolio/load_cells.pdf).

Ismail, Ahmed A., Haitham S. Hamza, and Amira M. Kotb. 2019. *Performance Evaluation of Open Source IoT Platforms*. 2018 IEEE Global Conference on Internet of Things, GCIoT 2018 (October 2019).

[https://www.researchgate.net/publication/330582133\\_Performance\\_Evaluation\\_of\\_Open\\_Source\\_IoT\\_Platforms](https://www.researchgate.net/publication/330582133_Performance_Evaluation_of_Open_Source_IoT_Platforms).



Kataria, Satish M, and Vanaraj B Vaghela. 2012. *Design and Implementation of High Precision Advanced Weighing Machine with TFT Panel*. 1(8): 1–7.

[https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&cad=rja&uact=8&ved=2ahUKEwjAsa6M2d\\_wAhXbfn0KHc8iDmoQFnoECA4QAA&url=https%3A%2F%2Fwww.ijert.org%2Fresearch%2Fdesign-and-implementation-of-high-precision-advanced-weighing-machine-with-tft-panel-IJERTV1IS8314.pdf&usg=AOvVaw10XG7xhHRn1g8nqoNs73yS](https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&cad=rja&uact=8&ved=2ahUKEwjAsa6M2d_wAhXbfn0KHc8iDmoQFnoECA4QAA&url=https%3A%2F%2Fwww.ijert.org%2Fresearch%2Fdesign-and-implementation-of-high-precision-advanced-weighing-machine-with-tft-panel-IJERTV1IS8314.pdf&usg=AOvVaw10XG7xhHRn1g8nqoNs73yS).

Kukkamäki, Joni, Robson Costa, Vilson Heck Júnior, and Eduardo Natan Bitencourt. 2018. *IoT Centralization and Management Applying ThingsBoard Platform*. Häme University of Applied Sciences - HAMK University Centre-Hämeenlinna Smart Services Research Unit (August): 18–19.

[https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&cad=rja&uact=8&ved=2ahUKEwjA59no\\_uLwAhUjqksFHA6G0CbAQFnoECBsQAA&url=https%3A%2F%2Fwww.hamk.fi%2Fwp-content%2Fuploads%2F2019%2F03%2FProject-Report\\_bitencourt\\_anjos.pdf&usg=AOvVaw2c0YfOo5xDWhx8h9YZDqDI](https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&cad=rja&uact=8&ved=2ahUKEwjA59no_uLwAhUjqksFHA6G0CbAQFnoECBsQAA&url=https%3A%2F%2Fwww.hamk.fi%2Fwp-content%2Fuploads%2F2019%2F03%2FProject-Report_bitencourt_anjos.pdf&usg=AOvVaw2c0YfOo5xDWhx8h9YZDqDI).

Latha, N Anju, and B Rama Murthy. 2017. *Arduino Based Weighting Scale Using Load Cell*. International Journal of Science Technology & Engineering Technology & Engineering 3(6): 704–7. <http://ijsrst.com/paper/2443.pdf>.

Mahbub, Mobasshir. 2019. *Automated Control Signal Reception Acknowledgement System Using NRF24L01P Wireless*



*Transceiver Module and Arduino*. ICST Transactions on Mobile Communications and Applications 5(16): 159337. [https://www.researchgate.net/publication/332277135\\_Automated\\_Control\\_Signal\\_Reception\\_Acknowledgement\\_System\\_Using\\_Nrf24l01p\\_Wireless\\_Transceiver\\_Module\\_and\\_Arduino](https://www.researchgate.net/publication/332277135_Automated_Control_Signal_Reception_Acknowledgement_System_Using_Nrf24l01p_Wireless_Transceiver_Module_and_Arduino).

Maier, Alexander, Andrew Sharp, and Yuriy Vagapov. 2017. *Comparative Analysis and Practical Implementation of the ESP32 Microcontroller Module for the Internet of Things*. 2017 Internet Technologies and Applications, ITA 2017 - Proceedings of the 7th International Conference (September): 143–48. [https://www.researchgate.net/publication/320273388\\_Comparative\\_Analysis\\_and\\_Practical\\_Implementation\\_of\\_the\\_ESP32\\_Microcontroller\\_Module\\_for\\_the\\_Internet\\_of\\_Things](https://www.researchgate.net/publication/320273388_Comparative_Analysis_and_Practical_Implementation_of_the_ESP32_Microcontroller_Module_for_the_Internet_of_Things).

Mcainsh, R K. 2014. *Design and Engineering of an Accurate Bicycle Power Meter*. Verve Design srl Via Supercera (May): 4–11. [https://www.researchgate.net/publication/303517617\\_Design\\_and\\_Engineering\\_of\\_an\\_Accurate\\_Bicycle\\_Power\\_Meter/download](https://www.researchgate.net/publication/303517617_Design_and_Engineering_of_an_Accurate_Bicycle_Power_Meter/download).

Megantoro, Prisma et al. 2020. *The Design of Digital Liquid Density Meter Based on Arduino*. Journal of Robotics and Control (JRC) 1(1): 1–6. <https://www.semanticscholar.org/paper/The-Design-of-Digital-Liquid-Density-Meter-Based-on-Megantoro-Widjanarko/402aac19eb6f4993aa8d74cf7bd4ba34bdb9bc6a>.

Patel, Keyur K, Sunil M Patel, and P G Scholar. 2016. *Internet of Things-IOT: Definition, Characteristics, Architecture, Enabling Technologies, Application & Future Challenges*.





International Journal of Engineering Science and Computing  
6(5): 1–10. <http://ijesc.org/>.

Putra, Muhammad, and Sidik. 2018. *Implementasi Network Attached Storage (NAS) Menggunakan Synology Disk Station Manager (DSM 5.2) Untuk Optimalisasi Data Sharing Center*. Jurnal Teknik Komputer IV(2): 39–47.

<http://ejournal.bsi.ac.id/ejournal/index.php/jtk>.

Scalet, Thais, João Francisco Viana, Rangel Arthur, and Francisco José Arnold. 2018. *Experimental Evaluation of Piezoelectric Buzzers as Devices for Energy Harvesting*. University of Campinas (August): 7–10.

[https://www.researchgate.net/publication/326905148\\_Experimental\\_Evaluation\\_of\\_Piezoelectric\\_Buzzers\\_as\\_Devices\\_for\\_Energy\\_Harvesting](https://www.researchgate.net/publication/326905148_Experimental_Evaluation_of_Piezoelectric_Buzzers_as_Devices_for_Energy_Harvesting).

Sukmawati, Ni Made Suci. 2014. “Faktor-Faktor Yang Mempengaruhi Susunan Dan Keadaan Air Susu.” *Bahan Ajar Ilmu Ternak Perah, LABORATORIUM ILMU TERNAK PERAH FAKULTAS PETERNAKAN UNIVERSITAS UDAYANA DENPASAR*: 1–26.

[https://simdos.unud.ac.id/uploads/file\\_pondidikan\\_dir/8589c2c12e6a92d04c3f81c847ecff1b.pdf](https://simdos.unud.ac.id/uploads/file_pondidikan_dir/8589c2c12e6a92d04c3f81c847ecff1b.pdf).

Swati, Thimmapuram, and K. Raghavendra Ra. 2018. *Industrial Process Monitoring System Using Esp32*.

International Journal of Engineering and Advanced Technology 8(2C2): 123–27.

<https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&cad=rja&uact=8&ved=2ahUKEwihgKG3>



[zN\\_wAhUMqksFHRgDDooQFnoECBMQAA&url=https%3A%2F%2Fwww.ijrte.org%2Fwp-content%2Fuploads%2Fpapers%2Fv7i6s4%2FF10550476S419.pdf&usg=AOvVaw2Cgu3HuaUiuDw4VR9qaO7g.](https://www.ijrte.org/wp-content/uploads/2013/01/17.pdf)

Thakkar, Kamlesh H., and Vipul M. Prajapati. 2013. *Performance Evaluation of Strain Gauge Based Load Cell to Improve Weighing Accuracy*. International Journal of Latest Trends in Engineering and Technology (IJLTET) 2(1): 103–7. <https://www.ijltet.org/wp-content/uploads/2013/01/17.pdf>.

Traboulsi, Salam, and Stefan Knauth. 2020. *Towards Implementation of an IoT Analysis System for Buildings Environmental Data and Workplace Well-Being with an IoT Open Software*. Procedia Computer Science 170(2019): 341–46. <https://doi.org/10.1016/j.procs.2020.03.048>.

Vishnupriya, B. et al. 2020. *Load Detection and Monitoring System*. International Journal of Scientific and Technology Research 9(2): 5756–58. <http://www.ijstr.org/final-print/feb2020/Load-Detection-And-Monitoring-System.pdf>.

Wahyudi, Abdur Rahman, and Muhammad Nawawi. 2018. *Perbandingan Nilai Ukur Sensor Load Cell Pada Alat Penyortir Buah Otomatis Terhadap Timbangan Manual*. ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika 5(2): 207. [https://www.researchgate.net/publication/319477344\\_Perbandingan\\_Nilai\\_Ukur\\_Sensor\\_Load\\_Cell\\_pada\\_Alat\\_Penyortir\\_Buah\\_Otomatis\\_terhadap\\_Timbangan\\_Manual](https://www.researchgate.net/publication/319477344_Perbandingan_Nilai_Ukur_Sensor_Load_Cell_pada_Alat_Penyortir_Buah_Otomatis_terhadap_Timbangan_Manual).



Zozo, Hassan, and Hesham Arafat Ali. 2015. *Internet of Things (IoT): Definitions, Challenges and Recent Research Directions*. International Journal of Computer Applications 128(1): 37–47. [https://www.researchgate.net/publication/320532203\\_Internet\\_of\\_Things\\_IoT\\_Definitions\\_Challenges\\_and\\_Recent\\_Research\\_Directions](https://www.researchgate.net/publication/320532203_Internet_of_Things_IoT_Definitions_Challenges_and_Recent_Research_Directions).

