



Jurnal Politeknik Caltex Riau

Terbit Online pada laman <https://jurnal.pcr.ac.id/index.php/jkt/>
 | e- ISSN : 2460-5255 (Online) | p- ISSN : 2443-4159 (Print) |

Sistem Monitoring Tangki dan Penghitung RunHour Genset Otomatis Berbasis Internet of Things (IoT)

Cyntia Widiyari, S. ST., M. T.¹, Putri Insani, A. Md², dan Muhammad Diono, S. ST., M. T.³

¹Program Studi Teknik Elektronika Telekomunikasi, Politeknik Caltex Riau email: insani6tt@mahasiswa.pcr.ac.id

²Program Studi Teknik Telekomunikasi, Politeknik Caltex Riau, email: cyntia@pcr.ac.id

³Program Studi Teknik Elektronika Telekomunikasi, Politeknik Caltex Riau, email: diono@pcr.ac.id

Abstrak

Semakin berkembangnya mesin-mesin dalam kehidupan manusia guna mempermudah pekerjaan manusia selama 24 jam, tidak dapat dipungkiri bahwa Generator Set atau yang lebih dikenal dengan Genset sangat berperan penting dalam menjaga mesin mesin tetap hidup selama dalam kondisi listrik padam. Untuk itu perlunya sebuah monitoring tangki bahan bakar genset untuk mempermudah mengetahui volume bahan bakar genset yang berubah ubah, dan sistem penghitungan otomatis runhours agar pemeliharaan genset dapat dengan mudah ditentukan berdasarkan penghitungan lamanya genset hidup. Sistem monitoring tangki dan perhitungan otomatis runhours genset yang dirancang berbasis IoT. Sistem ini menggunakan sensor ultrasonik untuk mengetahui ketinggian bahan bakar ditangki, sensor arus sebagai pendeteksi genset hidup, NodeMCU Esp8266 berfungsi sebagai penerima data dari sensor dan mengirimkannya ke aplikasi web yang telah dibuat. Dari hasil pengujian dapat diketahui bahwa persentase error pada sensor ultrasonik sebesar 5,2%, dan pengujian pada sensor arus yang mendeteksi arus pada beban dengan error waktu penghitungan manual dengan waktu pada tampilan website hanya berkisar 1-2 detik.

Kata kunci: Tangki Genset, NodeMCU Esp8266, Sensor Ultrasonik, Sensor Arus, Servo, Server, Web

Abstract

The increasing development of engines in human life to facilitate human work for 24 hours, cannot be denied that the Generator Set or better known as Genset is very important role in keeping the engine alive during the power outages. For this reason, it is necessary to monitor a Genset fuel tank to make it easier to find out the changing conditions of the Genset fuel, and the automatic runhours calculation system. So that the maintenance of the genset can be easily determined based on the calculation of the life span of the genset. Tank monitoring system and automatic calculation of runhours genset that is designed based on IoT, this system uses an ultrasonic sensor to find out the height of the fuel tank in captivity, the current sensor as a live genset detector, Nodemcu Esp8266 functions as data receiver from the sensor and sends it to the website application that has been created. From the test results it can be seen that the percentage of errors in the ultrasonic sensor is 5.2%, and testing on the current sensor that detects the current in the load with an error in the manual calculation time with the time on the web display only ranges from 1-2 seconds.

Keywords: Tank Generator sets, NodeMCU Esp8266, Ultrasonic Sensors, Current Sensors, Servo, Server, Web.

1. Pendahuluan

Generator set atau yang lebih dikenal dengan sebutan genset sangat dikenal oleh masyarakat luas karena fungsinya sebagai penghasil daya listrik saat listrik padam. Tidak hanya digunakan oleh perorangan dengan ukuran yang kecil, genset juga digunakan pada perusahaan baik perusahaan besar maupun kecil, milik swasta ataupun pemerintah dengan ukuran genset sampai dengan kapasitas 250 KVA (Kilo Volt Ampere [3].

Seiring dengan kemajuan teknologi dalam berbagai bidang termasuk pada bidang penerbangan, genset merupakan suatu alat yang harus ada dalam lingkup Bandar udara karena, alat navigasi maupun telekomunikasi udara yang berguna sebagai panduan arah jalan pesawat tidak boleh mati dalam artian harus hidup selama 24 jam sesuai dengan peraturan penerbangan. Untuk itu genset harus memiliki tangki sebagai penampung bahan bakar yang digunakan untuk menghidupkan genset. Kondisi bahan bakar di dalam tangki yang berubah ubah karena digunakan harus di *monitoring* secara berkala sehingga para pekerja (teknisi) bisa mengetahui kapan bahan bakar akan diisi kembali, dan sebagai laporan bertanggung jawaban kepada pihak perusahaan tentang banyaknya bahan bakar yang telah digunakan. Tidak hanya *memonitoring* tangki genset, *runhours* (lamanya genset hidup) juga harus di *monitoring* sehingga kita dapat menentukan waktu pemeliharaan atau perawatan pada genset, sehingga genset terawat dengan baik dan dapat beroperasi dalam jangka waktu yang lama sesuai dengan fungsinya.

Penelitian mengenai model sistem *monitoring* tangki bahan bakar minyak SPBU dengan menggunakan *website* aplikasi dan *sms gateway* (Riki Ruli A Siregar, 2015), memanfaatkan mikrokontroler Arduino, sensor ultrasonic, sensor suhu LM35, bahasa pemrograman PHP, Gammu SMS Gateway dan Visual C# serta memanfaatkan media komunikasi wireless untuk membangun sebuah sistem Telemetry. Melalui penelitian ini diperoleh sebuah sistem yang dapat digunakan untuk memonitor volume bahan bakar minyak pada SPBU berbasis *web* aplikasi, dan terintegrasi dengan *sms gateway* untuk memberikan informasi kondisi bahan bakar minyak pada tangki BBM apabila BBM hampir mencapai level terendah. Dalam penelitian ini perbandingan penggunaan sensor ultrasonik HC-SR04 dan sensor suhu LM35 dan pengukuran manual dengan interval jarak 5 cm, tingkat error rata-rata 3,02%, dengan menggunakan model tangki bahan bakar.

Penelitian mengenai implementasi sistem pengukuran otomatis bahan bakar di tangki genset dengan sistem *monitoring* berbasis jaringan yang dilakukan oleh Mangaraja (2005) [5]. Menjelaskan tentang pengukuran tangki bahan bakar menggunakan sensor ultrasonik, setelah sensor mendeteksi jarak selanjutnya arduino akan pengolah data pengukuran isi dari tangki saat itu, setelah data diperoleh, data tersebut dikirim ke *web* monitoring melalui arduino *ethernet shield*. Dimana *ethernet shield* akan diperintahkan untuk mengirim data ke *website* secara berkala. Namun dalam penelitian ini fitur-fitur pada *website* serta pengaksesan untuk melakukan monitoring belum lengkap dan mudah.

Dari kekurangan-kekurangan penelitian diatas, peneliti merancang suatu Sistem Monitoring dan Penghitung *Runhours* pada Genset Otomatis Berbasis *Internet of Things*. Sistem ini dapat memonitor bahan bakar minyak genset secara riil time pada saat genset digunakan, menggunakan sensor ultrasonik yang diletakkan pada tangki bahan bakar, lalu data sensor akan ditampilkan pada aplikasi *personal computer* (PC). Sensor akan mengirimkan data ke NodeMCU Esp8266, lalu data akan dikirim ke *website* server, dimana *website* server berfungsi sebagai penyimpanan data, data akan ditampilkan pada aplikasi *personal computer* berupa kolom kolom data yang diperlukan yaitu hari dan tanggal, ketinggian bahan bakar dalam tangki, volume bahan bakar, fitur pengisian bahan bakar otomatis untuk tangki 1 serta *runhour*. User dapat melakukan monitoring tangki genset serta *runhour* genset dari fitur yang telah dirancang pada aplikasi *personal computer*.

2. Dasar Teori

2.1 Generator Set (Genset) [11]

Generator set atau yang sering disebut sebagai genset merupakan sebuah perangkat yang bisa menghasilkan listrik yang terdiri dari gabungan *altenator* dan *engine* yang berfungsi sebagai alat pembangkit listrik. Prinsip kerja dari generator set sendiri yaitu terdiri dari *engine* atau motor penggerak yang menggunakan solar atau mesin untuk menghidupnya, dan terdiri dari generator yang merupakan gulungan kawat yang di buat dari tembaga yang terdiri atas kumparan statis atau stator dan dilengkapi pula dengan kumparan berputar atau rotor.

2.2 Sensor Ultrasonik [10]

Sensor ultrasonik adalah sebuah sensor yang memanfaatkan pancaran gelombang ultrasonik. Sensor ultrasonik ini terdiri dari rangkaian pemancar ultrasonik yang disebut *transmitter* dan rangkaian penerima ultrasonik disebut *receiver*.

Sensor tersebut bekerja pada frekuensi diatas 20kHz, biasanya yang digunakan untuk mengukur jarak benda adalah 40kHz. Sinyal tersebut di bangkitkan oleh rangkaian pemancar ultrasonik. Sinyal yang dipancarkan tersebut kemudian akan merambat sebagai sinyal atau gelombang bunyi dengan kecepatan bunyi yang berkisar 340 m/s. Sinyal tersebut kemudian akan dipantulkan dan akan diterima kembali oleh bagian penerima ultrasonik. Setelah sinyal tersebut sampai di penerima ultrasonik, kemudian sinyal tersebut akan diproses untuk menghitung jaraknya. Jarak dihitung berdasarkan rumus:

$$S = 340.t/2.....[1]$$

Dimana S merupakan jarak antara sensor ultrasonik dengan benda (bidang pantul), dan t adalah selisih antara waktu pemancaran gelombang oleh *transmitter* dan waktu ketika gelombang pantul diterima *receiver*.

2.3 Sensor Arus [10]

ACS712 merupakan suatu IC terpaket yang mana berguna sebagai sensor arus menggantikan transformator arus, modul sensor arus DC maupun AC ini relatif besar dalam hal ukuran. Keluaran (output) dari sensor adalah sinyal analog yang proporsional terhadap arus listrik yang mengalir di antara pin pendeteksinya. Pada prinsipnya ACS712 sama dengan sensor efek hall lainnya yaitu dengan memanfaatkan medan magnetik disekitar arus kemudian dikonversi menjadi tegangan yang linier dengan perubahan arus. Nilai variabel dari sensor ini merupakan input untuk mikrokontroler yang kemudian diolah.

Sensor arus ACS712 mempunyai karakteristik tegangan *suplay* dengan *symbol* Vcc dan dengan nilai maksimal adalah 8 V, output tegangan (Vout) dengan nilai 8V dan toleransi arus lebih (Ip) adalah sebesar 100A.

Tabel 1. Spesifikasi Sensor Arus

No.	Karakteristik	Simbol	Rating Maksimal
1	Tegangan Supply	Vcc	8 V
2	Tegangan Output	Vout	8 V
3	Toleransi Arus Lebih	Ip	100 A

2.4 NodeMCU Esp8266[12]

NodeMCU merupakan sebuah *open source platform* IoT dan pengembangan kit yang menggunakan bahasa pemrograman Lua untuk membantu dalam membuat prototype produk IoT atau bisa dengan memakai sketch dengan arduino IDE. Pengembangan kit ini didasarkan pada

modul ESP8266, yang mengintegrasikan GPIO, PWM (*Pulse Width Modulation*), IIC, 1-Wire dan ADC (*Analog to Digital Converter*) semua dalam satu *board*.

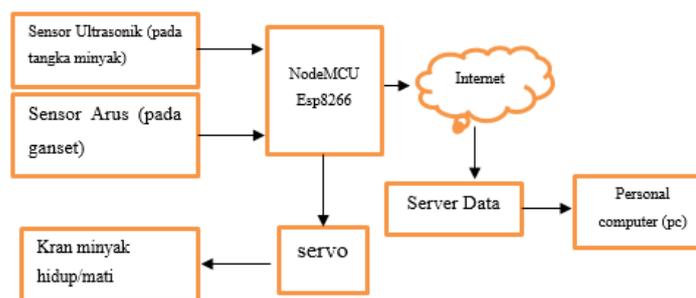
2.5 Motor Servo[10]

Servomotor adalah sebuah motor dengan sistem *closed feedback* di mana posisi dari motor akan diinformasikan kembali ke rangkaian kontrol yang ada di dalam *servomotor*. Motor ini terdiri dari sebuah motor, serangkaian *gear*, potensiometer dan rangkaian kontrol. Potensiometer berfungsi untuk menentukan batas sudut dari putaran servo. Sedangkan sudut dari sumbu motor servo diatur berdasarkan lebar pulsa yang dikirim melalui kaki sinyal dari kabel motor. Pulsa 1,5 ms pada periode selebar 2 ms maka sudut dari sumbu motor akan berada pada posisi tengah. Semakin lebar pulsa OFF maka akan semakin besar gerakan sumbu ke arah jarum jam dan semakin kecil pulsa OFF maka akan semakin besar gerakan sumbu ke arah yang berlawanan dengan jarum jam.

3. Perancangan

3.1 Blok Diagram Sistem

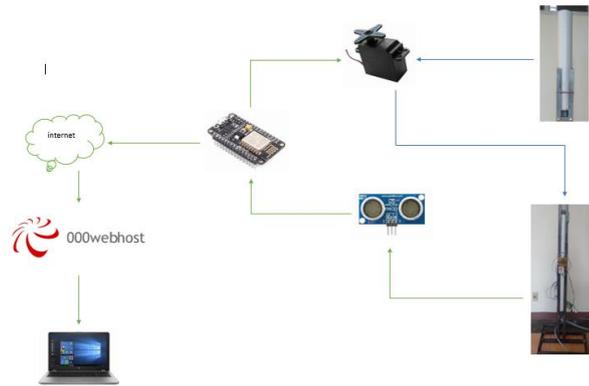
Blok diagram sistem dan proses kerja sistem ini menggunakan Sensor Ultrasonik untuk *me-monitoring* Tangki Genset dan Sensor Arus untuk penghitungan otomatis *runhour* genset. Blok perancangan sistem dapat dilihat pada Gambar 1.



Gambar 1. Blok Diagram Sistem

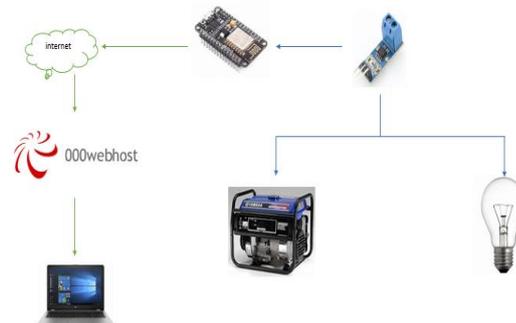
Masing masing sensor akan memperoleh data yang dibutuhkan, data tersebut menjadi *input* bagi NodeMCU untuk selanjutnya akan dikirim ke *server* menggunakan modul WiFi yang telah ada pada NodeMCU Esp8266 sehingga data yang akan di *monitoring* bisa dilihat pada PC (*user*). Pada *monitoring* tangki, pengisian otomatis memerlukan motor servo sebagai penggerak kran (pembuka dan penutup kran). Berikut fungsi dari alat alat yang berada pada blok diagram diatas:

1. Sensor Ultrasonik: Berperan sebagai mendeteksi jumlah minyak dalam tangki (pipa).
2. Sensor Arus: Berguna sebagai pendeteksi arus pada genset, sehingga dapat diketahui genset sedang dalam kondisi hidup.
3. NodeMCU Esp8266: Berfungsi sebagai Arduino yaitu sebagai *processor* sekaligus modul wifi untuk mengirimkan data, karena NodeMCU ini dilengkapi modul wifi esp8266.
4. Server: Berperan sebagai *server* untuk menyimpan data yang akan dikirimkan ke *Personal Computer* (PC).
5. Servo: berfungsi sebagai pembuka dan penutup kran untuk pengisian otomatis minyak solar ke tangki (pipa).



Gambar 2. Arsitektur Perancangan Sistem Monitoring Tangki

Pada Gambar 2 ilustrasi arsitektur perancangan sistem *monitoring* tangki dapat dilihat adanya pipa 1 dan 2, pipa 1 merupakan tangki cadangan apabila pipa 2 mempunyai minyak solar dalam tinggi *minimum* yang sudah ditentukan yaitu ≤ 50 cm, maka kran akan membuka otomatis, menggunakan motor servo. Sensor ultrasonik berfungsi untuk mendeteksi ketinggian minyak dalam pipa 2, letak dari sensor ultrasonik nantinya adalah pada bagian atas pipa yang akan dibuatkan penyangganya. Selanjutnya NodeMCU Esp8266 mendapatkan data tersebut (mengolah), dan mengirimkannya ke *server* data dengan bantuan WiFi sehingga bisa diakses oleh *client* melalui aplikasi *website* yang telah dibuat.

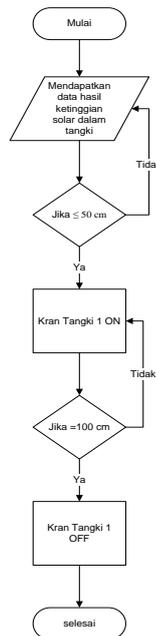


Gambar 3. Gambar Arsitektur Perancangan Perhitungan *runhour* Genset

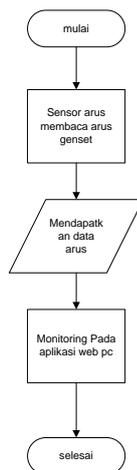
Pada Gambar 3 ilustrasi perancangan perhitungan *runhour* genset, sensor arus ACS712 akan dirangkaikan pada genset sehingga dapat mendeteksi sensor arus yang mempunyai beban untuk kemudian bisa di-*monitoring* lama genset tersebut hidup pada aplikasi *website* yang telah dirancang sebelumnya dengan prinsip kerja seperti *stopwatch*. Dengan NodeMCU Esp8266 sebagai penerima data sensor Arus ACS712.

3.2 Flowchart Sistem

Pada flowchart sistem monitoring tangki genset sesuai Gambar 4, system ini bekerja setelah NodeMCU Esp8266 mendapatkan data hasil ketinggian minyak di dalam tangki, jika ketinggian minyak ≤ 50 cm maka kran dari tangki cadangan atau tangki 1 akan on, apabila ketinggian minyak dalam tangki sudah mencapai ketinggian 100 cm kembali, Maka kran akan menutup yang menandakan tangki 2 sudah penuh. Data ketinggian didapat dari sensor ultrasonic, dengan mikrokontroller menggunakan NodeMcu Esp8266. Untuk website menggunakan 000webhost dan personal computer sebagai tempat user untuk mengakses data yang dikirimkan lewat internet. Adapun Flowchart untuk Sistem Monitoring runhour Genset dapat dilihat pada Gambar 5.



Gambar 4. Flowchart system



Gambar 5. Sistem RounHour Genset

Pada flowchart monitoring *runhour* genset sesuai gambar 5, diawali dengan pembacaan arus genset pada sensor arus, kemudian setelah data arus didapatkan, data tersebut bisa diakses pada PC melalui aplikasi *website* yang telah dirancang sebelumnya.

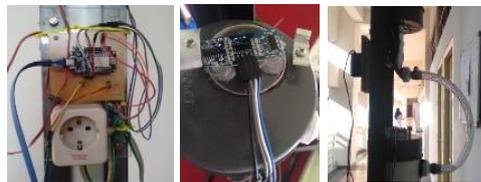
3.3 Rancangan Alat (Hardware)

Pada hardware ini terdiri dari komponen sensor ultrasonik, sensor arus, NodeMCU esp8266, dan motor servo. NodeMCU Esp 8266 berfungsi sebagai pusat kontrol dari sistem yang mengontrol perangkat-perangkat sensor seperti sensor ultrasonik, sensor arus untuk mendapatkan data pengukuran terkini. Sensor ultrasonik berfungsi sebagai pengukur ketinggian air pada tangki, sensor tersebut menghasilkan data berupa sinyal-sinyal yang kemudian, diolah oleh NodeMCU Esp 8266 menjadi data hasil pengukuran. NodeMCU juga berfungsi sebagai pengatur Motor Servo serta data hasil pendeteksian oleh sensor arus juga akan diolah pada NodeMCU untuk kemudian dapat dimonitoring. Motor Servo berfungsi untuk membuka atau menutup kran pada pengisian otomatis. Selain pengolahan data NodeMCU Esp8266 juga berfungsi sebagai modul

WiFi yang akan mengirimkan data sehingga dapat dimonitoring pada *website* yang telah dibuat sebelumnya.

Tangki pada Monitoring ini disimulasikan pada pipa berdiamter ± 8.7 cm, pipa terdiri dari pipa 1 (tangki pengisi) dengan panjang 60 cm dan pipa 2 (tangki genset) dengan panjang 110 cm, sensor ultrasonic diletakan pada pipa 2 untuk memonitoring ketinggian minyak pada pipa, apabila isi minyak pada pipa 2 ≤ 50 cm maka kondisi servo membuka, apabila =100 kondisi servo akan menutup pada pipa 2, volume maksimum hanya sampai ketinggian=100.

Setelah tinggi minyak didapat maka data tersebut masuk ke NodeMcu Esp8266. Pada NodeMCU Esp8266 dihitung tinggi terkini dari tinggi minyak yang didapat dari sensor ultrasonik, kemudian hasil perhitungan tinggi akan dikirimkan ke *database*, lalu disimpan diserver. Dan akan ditampilkan pada *website*. Tampilan pada *website* berupa kolom-kolom informasi yang terdiri dari hari, tanggal, waktu dan ketinggian minyak serta *runhour* genset dan fitur untuk mereset *runhour* genset.

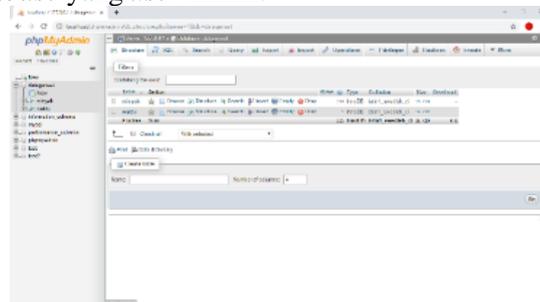


Gambar 6. Rangkaian dan sensor

3.4 Database

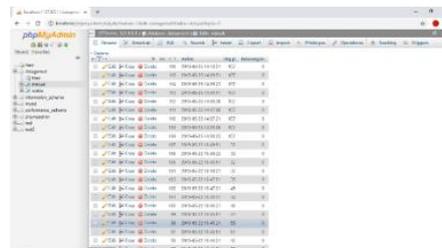
Database memiliki fungsi untuk menyimpan data ketinggian minyak pada pipa. Ketinggian minyak tersebut akan disimpan dan ditampilkan pada *website* yang telah dibuat. Pada Gambar 7 dapat dilihat tampilan pada *database*.

Sebelum *user* membuka *database*, akan nada tampilan login yang harus memasukkan *username* dan *password* dari *database* yang *user* miliki.



Gambar 7. Tampilan awal database

Jika *user* ingin melihat data yang tersimpan pada *database* *user* dapat memilih menu “Data masuk” sehingga akan muncul tampilan seperti Gambar 7.



Gambar 8. Tampilan data pada database

Berdasarkan Gambar 8, dapat terlihat data dari awal pengukuran, karena *database* menyimpan secara *continue*. Pada menu *database*, *user* dapat menghapus data jika diinginkan. Gambar 8 terlihat tampilan *database* berupa nomor pengukuran, tanggal pengukuran (tanggal data

masuk), waktu, dan ketinggian air pada pipa. Dari menu “Structure” yang ada pada *database*, dapat diketahui apa saja tampilan yang akan tersimpan didatabase (dapat dilihat pada gambar 9).



Gambar 9. tampilan pada menu “structure”

4. Pengujian dan Analisa

4.1 Pengujian Sensor ultrasonik

Pengujian ini dilakukan untuk mengukur seberapa akurat sensor ultrasonik mengukur ketinggian minyak dibandingkan pengukuran dengan cara manual. Pengujian ini dilakukan dengan cara sebagai berikut:

1. Pipa berdiameter 8.7 cm dan tinggi 100 cm diisi dengan volume 1-6 liter.
2. Kemudian tinggi minyak diambil berdasarkan penggaris (meteran) yang sudah ditempatkan didalam pipa. Dapat dilihat seperti Gambar 10.

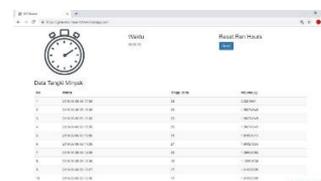


Gambar 10. Pengujian ketinggian minyak

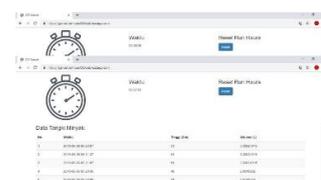
3. Selanjutnya setelah nilai ketinggian pada pipa didapatkan, maka dibandingkan dengan nilai yang ditampilkan pada *website*.
Nilai yang ditampilkan pada *website* dapat dilihat pada Gambar 11.



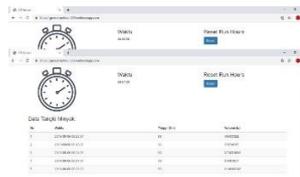
(a) Gambar data 1 liter



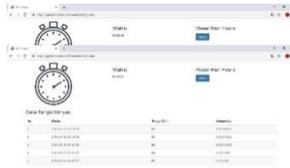
(b) Gambar data 2 liter



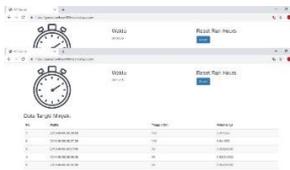
(c) Gambar data 3 liter



(d) Gambar data 4 liter



(e) Gambar data 5 liter



(f) Gambar data 6 liter

Gambar 11. Tampilan data pada web

Selisih dari kedua nilai pengukuran tersebut adalah nilai *error* dari pengukuran sensor ultrasonik. Pengujian dilakukan sebanyak 6 kali dimana nilai pengukuran volume 1 sampai 6 liter dalam keadaan pipa kosong lalu diisi dan sebanyak 6 kali dengan volume 1 sampai 6 liter dalam keadaan pipa terisi, lalu dikosongkan.

Nilai pengukuran secara manual diperoleh dari ketinggian minyak menggunakan meteran, yang ditempatkan diatas pipa dan disejajarkan dengan pipa yang ada, sedangkan nilai pengukuran sensor ultrasonik diperoleh dari data pada *website*. Pengujian ini hanya berfokus pada akurasi dari pengukuran sensor ultrasonik yang ditampilkan pada *website*. Pada tabel 2 dapat dilihat data hasil pengujian dari pengukuran dengan cara manual (menggunakan penggaris) dan sensor ultrasonik.

Tabel 2. Pengujian Sensor Ultrasonik

No.	Jarak Manual (Liter)	Liter Manual	Pengujian Sensor Ultrasonik (liter)	Error Liter Web	%
1	17,2	1	17	1,01	1,1
2	34,5	2	34	2,02	1,4
3	51,2	3	51	3,03	0,39
4	68,3	4	68	4,04	0,4
5	88,5	5	85	5,05	4,1
6	104	6	100	5,941	4
7	104	6	100	5,941	4
8	88,1	8	85	5,05	3,6
9	68,2	9	68	4,04	0,2
10	50,9	10	51	3,03	1,1
11	35	11	34	2,02	2,9
12	17	12	17	1,01	0
Rata-rata				0,034	1,9

Pada tabel 2 diatas dapat terlihat jika rata-rata *error* 1,9 % dalam satuan centimeter (cm) dan 0,034 dalam satuan liter. Hasil *error* yang didapatkan pada pengujian dapat dipengaruhi oleh sensitifitas pada sensor *ultrasonic*, tambahan pipa dalam pipa utama, dan posisi atau massa dari meteran itu sendiri pada saat dimasukkan ke dalam pipa untuk pengambilan data, serta tutup bawah dari pipa yang berbeda ukuran beberapa millimeter dengan pipa. Pada pengujian pertama, terlihat persentase *error* sebesar 1,1%. Untuk perhitungan *error* diambil contoh pada pengujian no.1:

$$\begin{aligned} \text{Error} &= \left| \frac{17-17,2}{17} \right| \times 100\% \\ \text{Error} &= \left| \frac{0,2}{17} \right| \times 100\% \\ \text{Error} &= 0,011 \times 100\% \\ \text{Error} &= 1,1\% \end{aligned}$$

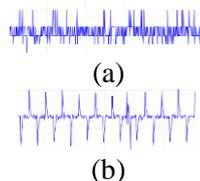
Untuk perhitungan *volume* adalah sebagai berikut:

$$\begin{aligned} D &= 83 \text{ mm} \\ \text{Luas Lingkaran} &= \pi * D * 0.25 \\ &= 3.14 * 83 * 0.25 \\ &= 65,115 \text{ mm}^2 \\ &= 0.65115 \text{ cm}^2 \end{aligned}$$

$$\begin{aligned} \text{Volume} &= T * \text{Luas lingkaran}, \\ \text{Volume} &= T(\text{dm}) * 0.0065115(\text{dm}^2) \end{aligned}$$

4.2 Pengujian Sensor Arus

Pada pengujian ini hanya berfokus pada ada atau tidaknya arus yang dideteksi berapapun besarnya karena hanya untuk mengetahui apakah genset yang digunakan hidup atau tidak, untuk mengetahuinya ditambahkan beban, dari penelitian yang dilakukan, semakin besar daya dari beban yang dipakai semakin mudah dan terlihat sinyal arus pada serial plotter yang ada pada IDE Arduino. Gambar 12 merupakan hasil dari pengujian sensor arus dengan tanpa beban dan dengan beban berupa lampu 16 watt.



Gambar 12 Hasil pendeteksian sensor arus (a) tanpa beban (b) lampu 16watt.

Apabila sensor arus mendeteksi adanya arus, maka secara otomatis NodeMCU mengolah data yang terimanya untuk kemudian dikirimkan ke *server* sebagai waktu dari lamanya genset tersebut hidup. Berikut tampilan penghitungan data *runhour* genset *website*, dengan perhitungan manual dengan menggunakan *Stopwatch*.

Untuk perhitungan tegangan yang dihasilkan oleh sensor arus dapat dilihat dengan cara membuka serial plotter yang ada pada IDE ARDUINO. Board Arduino Uno memiliki resolusi 10 bit, dengan nilai terbesar 1023, yang berarti untuk *converter analog to digital* mempunyai nilai $0 - 1023 = 0 - 5$ volt

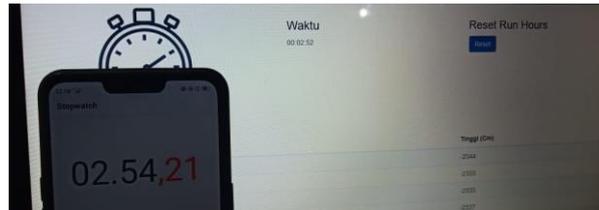
Contoh:

Sensor arus mendeteksi adanya 30 nilai besaran dari data terbesar dikurangi data terkecil, untuk perhitungannya sebagai berikut:

$$\frac{5}{1023} = 0,0049$$

$$0,00499 \times 30 = 0,147 \text{ volt.}$$

Untuk data selisih yang tersensor oleh sensor arus dengan nilai 30. Waktu yang dikirimkan pada web berasal dari data selisih yang selalu dihitung kontiniu pada sensor arus.



Gambar 13. Tampilan web waktu *Runhour*

Hasil yang didapat dari pengujian sesuai Gambar 13, *delay* waktu pengiriman di *website* dengan waktu yang dihitung secara manual menggunakan *stopwatch* adalah ± 2 detik. Perhitungan ini dimulai secara Bersama-sama dengan membandingkan waktu hidup *Runhour* di *website* dengan *stopwatch*. Dengan data waktu 02.54 detik pada *stopwatch* dan 02.52 pada *website*.

4.3 Pengujian Motor Servo

Pengujian pada motor servo ini hanya pada 2 kondisi yaitu, pipa dalam kondisi minyak didalamnya ≤ 50 cm dan =100. Pada saat ketinggian minyak dalam pipa kecil sama dengan 50 cm maka kondisi servo membuka, apabila =100 maka kondisi servo otomatis menutup sesuai tabel 3.

Tabel 3. Hasil Pengujian Motor Servo

No.	Status Ketinggian Minyak	Status Servo	Hasil
1	50	BUKA	BERHASIL
2	100	TUTUP	BERHASIL

5. Kesimpulan dan Saran

Dari pengujian yang sudah dilakukan, dapat disimpulkan bahwa Sistem Monitoring Tangki dan Penghitung RunHour Genset Otomatis Berbasis Internet of Things (IoT) membutuhkan waktu 30 detik agar data yang terbaca oleh sensor bisa ditampilkan pada *website* sehingga bisa dilakukan monitoring. Persentase *error* kinerja sensor ultrasonik dimana diambil berdasarkan perbandingan secara perhitungan yaitu sebesar 1,9 % dalam satuan centimeter (jarak) dan 0.034 dalam satuan liter (Volume). Sedangkan Persentase *error* kinerja *Runhour* pada *website* dibandingkan dengan pengukuran menggunakan stopwatch adalah sebesar ± 2 detik.

Daftar Pustaka

[1] Azhari, D. W., Lestari, I., Aryani, R. D., & BETA, S. (n.d.). KONTROL LEVEL AIR BERBASIS ARDUINO.

[2] Daniel Penta Mangaraja, T. A. (2015). Implementasi Sistem Pengukuran Otomatis Bahan Bakar di Tangki Ganset dengan Monitoring Berbasis Jaringan. *e-Proceeding of Applied Science*, 2587.

faudin, A. (2017, July 26). *Nyebartilmu.com*. Retrieved from Apa itu NodeMCU: <https://www.nyebartilmu.com/apa-itu-module-nodemcu-esp8266/>

- [3] Hargen. (2014, Mei 21). *Hargen for Electric*. Retrieved Mei 21, 2014, from apa itu kva ? : <https://www.hargen.co.id/news/2014/05/apa-itu-kva>
- [4] Junaidi, A. (2015). INTERNET OF THINGS, SEJARAH, TEKNOLOGI DAN. *Jurnal Ilmiah Teknologi Informasi terapan*, 2407-3911.
- [5] M. Saputra Tambun, N. S. (2015). Rancang bangun model monitoring underground tank spbu menggunakan gelombang ultrasonik berbasis
- [6] Riki Ruli A Siregar, R. R. (2015). Model Sistem Monitoring Tangki Bahan Bakar Minyak SPBU dengan Menggunakan WEB Aplikasi dan SMS gateway. *Jetri*, 59-72.
- [7] Risna, & Pradana, H. A. (2014). Rancang Bangun Aplikasi Monitoring Penggunaan Air PDAM Berbasis Mikrokontroler Arduino Uno.
- [8] Sandi, H. S. (2018). Rancang Bangun Sistem Monitoring Jumlah Sisa Volume Minyak Underground Tank Berbasis Mikrokontroler. *Universitas Negeri Medan*.
- [9] Suharjono, A., Rahayu, L. N., & Afwah, R. (2015). Aplikasi Sensor Flow Water Untuk Mengukur Penggunaan Air Pelanggan Secara Digital Serta Pengiriman Data Secara Otomatis Pada PDAM Kota Semarang.
- [10] Ulumuddin, M. S. (2017). Prototipe sistem monitoring air pada tangki berbasis internet of things menggunakan nodemcy esp8266 dan sensor ultrasonik. *Senter 2017*, 100-105.
- [11] Wholesale, a. &. (2005). *Alternator basic*. Retrieved from wholesale: <https://www.aspwholesale.com/alternator-basics-d12.html>
- [12] faudin, A. (2017, July 26). *Nyebarilmu.com*. Retrieved from Apa itu NodeMCU: <https://www.nyebarilmu.com/apa-itu-module-nodemcu-esp8266/>