TUGAS AKHIR – RM184831

RANCANG BANGUN INSTRUMEN *LOW-COST ULTRASONIC TIDE GAUGE* DENGAN SISTEM *INTERNET OF THINGS* (*IoT*) BERBASIS ARDUINO

NABIL AMIRUL HAQ
NRP 03311640000087

Dosen Pembimbing
Khomsin, S.T., M.T.
Danar Guruh Pratomo, S.T., M.T., Ph.D.

DEPARTEMEN TEKNIK GEOMATIKA
Fakultas Teknik Sipil, Perencanaan, dan Kebumian
Institut Teknologi Sepuluh Nopember
Surabaya 2020

TUGAS AKHIR – RM184831

RANCANG BANGUN INSTRUMEN *LOW-COST ULTRASONIC TIDE GAUGE* DENGAN SISTEM *INTERNET OF THINGS* (*IoT*) BERBASIS ARDUINO

NABIL AMIRUL HAQ
NRP 03311640000087

Dosen Pembimbing
Khomsin, S.T., M.T.
Danar Guruh Pratomo, S.T., M.T., Ph.D.

DEPARTEMEN TEKNIK GEOMATIKA
Fakultas Teknik Sipil, Perencanaan, dan Kebumian
Institut Teknologi Sepuluh Nopember
Surabaya 2020

"Halaman ini sengaja dikosongkan"

TUGAS AKHIR – RM184831

THE DESIGN OF AN ARDUINO BASED LOW-COST ULTRASONIC TIDE GAUGE WITH INTERNET OF THINGS (IoT) SYSTEM

NABIL AMIRUL HAQ
NRP 03311640000087

Supervisor
Khomsin, S.T., M.T.
Danar Guruh Pratomo, S.T., M.T., Ph.D.

GEOMATICS ENGINEERING DEPARTEMENT
Faculty of Civil, Planning, and Geo Engineering
Institut Teknologi Sepuluh Nopember
Surabaya 2020

"Halaman ini sengaja dikosongkan"

RANCANG BANGUN INSTRUMEN *LOW-COST ULTRASONIC TIDE GAUGE* DENGAN SISTEM *INTERNET OF THINGS* (*IoT*) BERBASIS ARDUINO

Mahasiswa : Nabil Amirul Haq
NRP : 03311640000087
Departemen : Teknik Geomatika
Pembimbing : 1. Khomsin, S.T., M.T.
2. Danar Guruh Pratomo, S.T., M.T., Ph.D.

ABSTRAK

Pasang surut air laut atau pasut merupakan salah satu komponen penting dalam beberapa pekerjaan seperti hidrografi dan navigasi. Secara umum, pasang surut bisa diukur dengan mengamati elevasi muka air laut menggunakan palem pasut, akan tetapi cara tersebut memiliki beberapa kelemahan, yaitu menghabiskan banyak tenaga dan waktu, serta data pasut yang dihasilkan cenderung bersifat subjektif tergantung pengamatnya. Salah satu cara untuk mengatasi kelemahan tersebut, dapat digunakan teknologi *automatic tide gauge* yang dapat memberikan data secara *near continue* dan *real time*. Akan tetapi, harga *automatic tide gauge* yang beredar di pasar saat ini masih cukup tinggi.

Pada penelitian ini, akan dirancang sebuah *automatic tide gauge* yang memiliki biaya pembuatan minimum menggunakan mikrokontroler Arduino, sensor ultrasonik, modul *GPS low-cost*, dan dilengkapi sistem *Internet of Things* (*IoT*) yang dapat mengupload dan monitoring data ke sebuah website secara online dan near real-time, serta. Pengujian instrumen dilakukan bersamaan dengan pengamatan pasut manual menggunakan palem pasut. Data pasut instrumen akan dilakukan kontrol kualitas data dengan aturan 3σ untuk mendeteksi *noise* pada data. Untuk menghilangkan noise, penelitian ini menggunakan metode *moving*

average dan *moving median* untuk *data filtering*. Uji ketelitian instrumen akan dilakukan dengan membandingkan data pasut bacaan instrumen dengan data pasut manual menggunakan palem pasut, perbandingan nilai pasang surut yang dilakukan akan menggunakan perhitungan kesalahan relatif dan *Root Mean Square Error* atau RMSE. Berdasarkan hasil uji ketelitian instrumen, didapatkan nilai kesalahan relatif dan RMSE sebesar 0,226% dan 6,629mm untuk data instrumen, 0,636% dan 18,542mm untuk data *moving average*, serta 0,437% dan 13,242mm untuk data *moving median*, dengan metode *filtering* terbaik adalah *moving median*. Hasil uji ketelitian instrumen, menunjukkan bahwa instrumen memiliki tingkat akurasi yang bagus dengan biaya produksi instrumen Rp. 2.240.000, atau 7,5% dari harga *tide gauge* termurah di pasaran.

Kata Kunci: *Tide Gauge, Low-Cost,* Pasang Surut Air Laut, Sensor Ultrasonik, Arduino

THE DESIGN OF AN ARDUINO BASED LOW-COST ULTRASONIC TIDE GAUGE WITH INTERNET OF THINGS (IoT) SYSTEM

Name	: Nabil Amirul Haq
NRP	: 03311640000087
Departement	: Geomatics Engineering
Supervisor	: 1. Khomsin, S.T., M.T.
	2. Danar Guruh Pratomo, S.T., M.T., Ph.D.

ABSTRACT

Ocean tides play essential roles in the field of hydrography and navigation. The conventional method to measure tides is by observing the vertical variation of sea surface using a tide staff. However, this method has several disadvantages. It consumes more manpower and time to observe and record the variation. Moreover, the recorded data tends to be subjective, depending on the observer. In order to overcome these disadvantages is by using an automatic tide gauge. This instrument can also provide almost continuous and near real-time tide data. The price of automatic tide gauge in the Indonesia market is very expensive.

This research attemped to build a low-cost ultrasonic tide gauge by utilizing an Arduino microcontroller, ultrasonic sensor, GPS module. The system is embed with the Internet of Things (IoT), so that the real-time data can be uploaded and monitored on a webserver. The assestment of the instrument testing was accomplished with the conventional method simultaneously. The quality of the data collected from the instrument is controlled using 3σ rule to detect and remove outliers. In order to eliminate the noise, a Moving Average and a Moving Median filtering methods are applied in the system. The data accuracy-test was performed using combined a relative error and Root Mean Square Error

(RMSE) methods. The test compares the data collected with the ultrasonic tide gauge and the direct observation method. The result shows the relative error and RMSE values 0.226% and 6.629mm for raw data, 0.636% and 18.542mm for moving average data, 0.437% and 13.242mm for the moving median data, with the best filtering method is moving median. The instrument accuracy test results show that this instrument has excellent accuracy, with a 2,240,000 IDR instrument production budget or 7.5% of the lowest tide gauge prices on the market.

Keywords: Tide Gauge, Low Cost, Ocean Tides, Ultrasonic Sensors, Arduino

LEMBAR PENGESAHAN

RANCANG BANGUN INSTRUMEN *LOW-COST ULTRASONIC TIDE GAUGE* DENGAN SISTEM *INTERNET OF THINGS* (*IoT*) BERBASIS ARDUINO

TUGAS AKHIR

Diajukan Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Teknik
Pada
Program Studi S-1 Teknik Geomatika
Fakultas Teknik Sipil, Perencanaan, dan Kebumian

Oleh :
NABIL AMIRUL HAQ
NRP.03311640000087

Disetujui oleh Pembimbing Tugas Akhir :

Khomsin, S.T., M.T. (.....)
19750705 200012 1 001
Danar Guruh Pratomo, S.T., M.T., Ph.D. (.....)
19800507 200312 1 001

SURABAYA, JULI 2020

KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
FAKULTAS TEKNIK SIPIL, PERENCANAAN DAN KEBUMIAN
DEPARTEMEN
TEKNIK GEOMATIKA

"Halaman ini sengaja dikosongkan"

KATA PENGANTAR

Puji syukur kehadirat Tuhan Yang Maha Esa atas segala limpahan rahmat, taufiq, dan hidayah-Nya, sehingga penulis dapat menyelesaikan tugas akhir dengan judul **"Rancang Bangun Instrumen *Low-Cost Ultrasonic Tide Gauge* Dengan Sistem *Internet Of Things* (*IoT*) Berbasis Arduino"** ini dengan baik dan lancar.

Laporan Tugas Akhir ini disusun untuk memenuhi salah satu syarat untuk menyelesaikan studi dan memperoleh gelar sarjana di Departemen Teknik Geomatika, Fakultas Teknik Sipil, Lingkungan dan Kebumian, Institut Teknologi Sepuluh Nopember Surabaya. Penulis menyampaikan terimakasih dan penghargaan sebesar-besarnya kepada semua pihak yang telah membantu, mendukung, dan memberikan kontribusi hingga Tugas Akhir ini dapat diselesaikan kepada:

1. Kedua orang tua dan segenap keluarga Penulis yang senantiasa memberikan dukungan baik spiritual maupun material.
2. Bapak Danar Guruh Pratomo, S.T., M.T., Ph.D. selaku Kepala Departemen Teknik Geomatika FTSPK ITS.
3. Bapak Khomsin, S.T., M.T. dan Bapak Danar Guruh Pratomo, S.T., M.T., Ph.D. yang membimbing penulis dalam menyelesaikan tugas akhir.
4. Bapak Wawan Sulistiawan selaku Direktur Utama dan seluruh pegawai PT. Alur Pelayaran Barat Surabaya yang telah memberikan ijin untuk melakukan penelitian tugas akhir di lingkungan perusahaan.
5. Teman-Teman seperjuangan di Kementerian Riset dan Teknologi "GLORY" BEM ITS yang telah memberikan dukungan dan semangat selama Penulis mengenyam Pendidikan di ITS.
6. Teman-Teman seperjuangan di Kementerian Riset dan Teknologi "ARGON" BEM ITS yang telah memberikan

dukungan dan semangat selama Penulis mengenyam Pendidikan di ITS.

7. Seluruh rekan seperjuangan Teknik Geomatika ITS angkatan 2016 (G18) yang telah banyak membantu dan berjuang bersama selama 4 tahun ini.

Penulis menyadari masih terdapat kekurangan dalam pembuatan tugas akhir ini. Oleh karena itu, kritik dan saran dari pembaca sangat kami harapkan untuk perbaikan laporan ini. Penulis berharap semoga laporan ini bisa bermanfaat, khususnya bagi penulis dan bagi pembaca umumnya.

Surabaya, Juli 2020

Penulis

DAFTAR ISI

HALAMAN JUDUL i
ABSTRAK v
ABSTRACT vii
LEMBAR PENGESAHAN ix
KATA PENGANTAR xi
DAFTAR ISI xiii
DAFTAR GAMBAR xvii
DAFTAR TABEL xix
DAFTAR LAMPIRAN xxi
BAB I PENDAHULUAN 1
1.1 Latar Belakang 1
1.2 Rumusan Masalah 3
1.3 Batasan Masalah 3
1.4 Tujuan Penelitian Tugas Akhir 4
1.5 Manfaat Tugas Akhir 4
BAB II TINJAUAN PUSTAKA 5
2.1 Pasang Surut Air Laut 5
2.2 Tipe Pasang Surut 6
2.3 Instrumen Pengukur Pasang Surut 9
2.3.1 *Tide Staff* 9
2.3.2 *Floating Gauge* 10
2.3.3 *Pressure Gauge* 11
2.3.4 *Radar Gauge* 12
2.3.5 *Ultrasonic Gauge* 13
2.4 Instrumentasi Berbasis Mikrokontroler Arduino 14

2.5 Arduino Mega 2560 Pro 15
2.5.1 Spesifikasi Arduino Mega 2560 Pro.....16
2.5.2 Konfigurasi Pin Arduino Mega 2560 Pro 17
2.6 Arduino IDE 20
2.7 Sensor Ultrasonik.....22
2.7.1 Prinsip Kerja Sensor Ultrasonik 22
2.7.2 Sensor JSN-SR04T.....24
2.7.3 Koreksi Kecepatan Suara.....25
2.8 Modul *SD Card*.....26
2.9 Modul RTC DS3231 27
2.10 Modul WiFi ESP8266 27
2.10.1 Modul WiFi NodeMCU ESP866.....28
2.11 Sensor Suhu DHT22.....30
2.12 Modul GPS NEO-6MV2 31
2.13 *Internet of Things* (*IoT*) 32
2.14 Perhitungan Kesalahan Relatif.....33
2.15 *Root Mean Square Error* 34
2.16 *Data Filtering* 34
2.16.1 *Moving Average* 35
2.16.2 *Moving Median* 36
2.17 Kontrol Kualitas Data.....36
2.18 Penelitian Terdahulu 37
2.18.1 Penelitian Oleh Mufidul Khoir.....37
2.18.2 Penelitian Oleh Abdul Chobir Dkk 38
2.18.3 Penelitian Oleh Naufal Ananda Dkk 38
BAB III METODOLOGI PENELITIAN.....41
3.1 Waktu dan Lokasi Penelitian 41

3.2 Alat dan Bahan.....42
3.2.1 Alat.....42
3.2.2 Bahan.....43
3.3 Metode Penelitian.....43
3.3.1 Tahap Persiapan.....45
3.3.2 Tahap Perancangan.....46
3.3.3 Tahap Pengujian.....46
3.3.4 Tahap Akhir.....49
BAB IV HASIL DAN PEMBAHASAN.....51
4.1 Pembuatan Instrumen.....51
4.1.1 Desain *Electrical Circuit*.....52
4.1.2 Perangkaian Komponen Elektronik.....61
4.1.3 Pembuatan Koding Instrumen.....62
4.1.4 Perancangan Bodi Instrumen.....67
4.1.5 Total Biaya Pembuatan Instrumen.....70
4.2 Hasil dan Analisis Hasil Pengujian Instrumen.....72
4.2.1 Data Pasang Surut dari Server Thingspeak.....72
4.2.2 Data Pasang Surut dari SD-*Card*.....76
4.2.3 Data Koordinat dari Thingspeak.....76
4.2.4 Kontrol Kualitas Data Pasang Surut.....77
4.2.5 Penyaringan Data Pasang Surut.....78
4.2.6 Perangkat Lunak Antar-Muka untuk *Data Filtering*..80
4.2.7 Perbandingan Bacaan Instrumen dengan Bacaan Manual.....81
4.2.8 Analisa Hasil Penyaringan Data.....85
BAB V KESIMPULAN DAN SARAN.....87
5.1 Kesimpulan.....87

5.2 Saran.....88
DAFTAR PUSTAKA.....89
LAMPIRAN.....93
BIODATA PENULIS.....107

DAFTAR GAMBAR

Gambar 2.1 Grafik *Diurnal Tide* 6
Gambar 2.2 Grafik *Semi Diurnal Tide* 7
Gambar 2.3 Grafik *Mixed Tide Prevailing Diurnal* 7
Gambar 2.4 Grafik *Mixed Tide Prevailing Semi Diurnal* 8
Gambar 2.5 *Tide Staff* 10
Gambar 2.6 *Floating Gauge* 11
Gambar 2.7 *Pressure Gauge* 12
Gambar 2.8 *Radar Gauge* 13
Gambar 2.9 *Ultrasonic Gauge* MOTIWALI 13
Gambar 2.10 Arduino Mega 2560 Pro 16
Gambar 2.11 Konfigurasi *Board* Arduino Mega 2560 Pro 17
Gambar 2.12 Konfigurasi Pin Arduino Mega 2560 Pro 18
Gambar 2.13 Arduino IDE 21
Gambar 2.14 Prinsip Kerja Sensor Ultrasonik 23
Gambar 2.15 Prinsip Pemantulan Sensor Ultrasonik 24
Gambar 2.16 Sensor JSN-SR04 24
Gambar 2.17 Modul SD-*Card* 26
Gambar 2.18 Modul RTC DS3231 27
Gambar 2.19 Modul WiFi ESP8266 28
Gambar 2.20 NodeMCU ESP8266 29
Gambar 2.21 Sensor Suhu DHT22 31
Gambar 2.22 Modul GPS NEO-6MV2 32
Gambar 2.23 Konsep *Internet of Things* 33
Gambar 2.24 Metode *Data Filtering* 35
Gambar 3.1 Lokasi Pengujian Alat 41
Gambar 3.2 Lokasi Pengujian Modul GPS 42
Gambar 3.3 Diagram Alir Penelitian 44
Gambar 3.4 Konfigurasi *Ultrasonic Tide Gauge* 47
Gambar 4.1 Diagram Kerja Alat 51
Gambar 4.2 *Electrical Circuit* Instrumen *Ultrasonic Tide Gauge* 52

Gambar 4.3 *Circuit* Sensor Ultrasonik.....53
Gambar 4.4 *Circuit* Sensor Suhu.....54
Gambar 4.5 *Circuit* Modul WiFi 55
Gambar 4.6 *Circuit* Modul SD-Card 56
Gambar 4.7 *Circuit* Modul RTC DS3231.....57
Gambar 4.8 *Circuit* LCD 16×2.....58
Gambar 4.9 *Circuit* Sumber Daya 59
Gambar 4.10 *Circuit* Sumber Daya NodeMCU 60
Gambar 4.11 *Circuit* ModulGPS NEO-6MV2 61
Gambar 4.12 Perangkaian Komponen Elektronik 61
Gambar 4.13 Desain Kotak Komponen Elektronik 68
Gambar 4.14 Hasil Pembuatan Kotak Komponen Elektronik.....68
Gambar 4.15 Desain Lengan Sensor.....69
Gambar 4.16 Hasil Pembuatan Lengan Sensor 69
Gambar 4.17 Data Pasang Surut di Website Thingspeak.....72
Gambar 4.18 Data Pasang Surut di Thingview 73
Gambar 4.19 Delay Perekaman Data.....73
Gambar 4.20 Grafik Pasut Data Thingspeak.....75
Gambar 4.21 Grafik Pasut Data SD-*Card*.....76
Gambar 4.22 Data Koordinat pada Thingspeak 77
Gambar 4.23 Data Koordinat pada ThingView.....77
Gambar 4.24 Grafik Pasut *Moving Average* 79
Gambar 4.25 Grafik Pasut *Moving Median*.....79
Gambar 4.26 *Overlay* Grafik Pasut Semua Metode Penyaringan80
Gambar 4.27 Tampilan Perangkat Lunak FilTide 80
Gambar 4.28 Grafik Data Pasut Manual.....82
Gambar 4.29 Grafik Kesalahan Relatif.....83
Gambar 4.30 Grafik Akurasi.....83
Gambar 4.31 Grafik RMSE.....84

DAFTAR TABEL

Tabel 2.1 Spesifikasi Arduino Mega 2560 Pro.....16
Tabel 4.1 Biaya Peralatan dan Jasa…….……..70
Tabel 4.2 Biaya Bahan Habis Pakai.....70
Tabel 4.3 Total Biaya.....71
Tabel 4.4 Data Pasut Thingspeak 74
Tabel 4.5 Hasil Kontrol Kualitas Data.....78
Tabel 4.6 Data Pengamatan Manual 81
Tabel 4.7 Hasil Perhitungan Kesalahan Relatif dan Akurasi 82
Tabel 4.8 Hasil Perhitungan RMSE.....84
Tabel 4.9 Perbandingan Hasil Penyaringan Data 85

"Halaman ini sengaja dikosongkan"

DAFTAR LAMPIRAN

Lampiran 1. Data Pasang Surut.....93
Lampiran 1.1 Data Pasang Surut dari Server Thingspeak.....93
Lampiran 1.2 Data Pasang Surut dari SD-*Card* 93
Lampiran 1.3 Data Pasang Surut Penyaringan *Moving Average* .93
Lampiran 1.4 Data Pasang Surut Penyaringan *Moving Median* ..93
Lampiran 1.5 Data Pengamaran Pasang Surut Manual.....93
Lampiran 2. Koding_n2.Pembuatan *Channel* Thingspeak….94
Lampiran 2.1 Koding Instrumen.....94
Lampiran 2.2 Koding NodeMCU dan Modul GPS 99
Lampiran 2.3 Koding Penyaringan Data.....102
Lampiran 2.4 Pembuatan *Channel* Thingspeak.....102
Lampiran 3. Dokumentasi.....104
Lampiran 3.1 Pembuatan Instrumen 104
Lampiran 3.2 Pengujian Instrumen.....105

"Halaman ini sengaja dikosongkan"

BAB I

PENDAHULUAN

1.1 Latar Belakang

Pasang surut air laut (pasut) merupakan fenomena naik turunnya muka air laut yang bersifat periodik, karena disebabkan oleh pengaruh siklus rotasi bumi. Pasut disebabkan oleh gaya tarik benda-benda di angkasa, terutama bulan dan matahari (Peterson, dkk 2015).

Beberapa pantai mengalami dua pasang tinggi hampir sama dan dua pasang surut setiap hari, yang disebut gelombang *semi-diurnal*. Beberapa lokasi mengalami hanya satu tinggi dan satu surut setiap hari, disebut pasang *diurnal*. Beberapa lokasi mengalami dua pasang yang merata dalam sehari, atau kadang-kadang satu tinggi dan satu rendah setiap hari, ini disebut pasang campuran. Waktu dan amplitudo pasang surut dipengaruhi oleh keselarasan matahari dan bulan bergantung pada pola pasang surut di laut dalam, sistem *amphidromi* lautan, dan dengan bentuk garis pantai dan juga pada profil batimetri (Haristyana 2012).

Informasi pasang surut air laut sangat penting dalam dunia hidrografi, navigasi, serta rekayasa pantai dan laut. Karena pasang surut bervariasi pada rentang waktu, maka dari itu, diperlukan catatan ketinggian pasang surut dengan informasi waktu. Untuk membuat catatan yang akurat, dibutuhkan sebuah stasiun pengamat pasang surut yang terdapat instrumen *tide gauge* yang harus tetap mengukur tinggi air laut dari waktu ke waktu. Pengukuran boleh mengabaikan variasi yang disebabkan oleh gelombang dengan periode yang lebih pendek dalam hitungan menit. Kemudian, dari data pasang surut, dapat digunakan untuk melakukan perhitungan referensi elevasi (atau datum) permukaan laut (Hasibuan 2009).

Sebelum berkembangnya teknologi, pengukuran pasang surut air laut dilakukan menggunakan sebuah palem pasang surut yang memiliki skala ukuran dalam satuan tertentu, dan dipasang di

tepi laut dan dipantau serta dicatat secara manual bacaan naik turunnya air laut. Seiring berkembangnya teknologi, instrumen pengukur pasang surut mulai diciptakan dengan istilah *Tide Gauge* (Kurniawan 2016).

Tide gauge memiliki berbagai macam jenis, antara lain *pressure tide gauge*, *radar tide gauge*, dan *floating tide gauge* (Triadmodjo, 2008). Akan tetapi, *automatic tide gauge* memiliki satu kelemahan, yaitu harganya yang cukup mahal di pasaran. Contoh *automatic tide gauge* yang beredar di Indonesia adalah *pressure gauge* yang diciptakan salah satu perusahaan peralatan survei yang memiliki harga sekitar Rp. 30.000.000 dan *pressure gauge* dari Valeport yang memiliki harga sekitar Rp. 95.000.000. Karena tingginya harga *automatic tide gauge* di pasaran, orang-orang yang membutuhkan data pasang surut untuk suatu pekerjaan melakukan pengamatan pasang surut menggunakan palem pasut, atau menggunakan data dari stasiun pasut terdekat. Tetapi, terdapat kelemahan pada kedua cara tersebut, yaitu pada hasil pengamatan pasut menggunakan palem pasut hasilnya subjektif tergantung pengamatnya, dan tidak semua lokasi memiliki stasiun pasut.

Seiring berjalannya waktu, perkembangan teknologi instrumen mulai memanfaatkan konektivitas internet untuk melakukan transfer data, melakukan kontrol instrumen, dan sebagainya. Sistem tersebut biasa dikenal dengan istilah *Internet of Thing* atau disingkatan *IoT*. Dengan menggunakan sistem *IoT* ini, seluruh pekerjaan yang dilakukan secara manual dan harus ke lokasi, dapat dilakukan di lokasi yang jauh, dimonitor secara *real-time*, dan terotomasi secara online menggunakan koneksi internet

Berdasarkan pada kondisi tersebut, Penulis melakukan penelitian berupa perancangan instrumen *low-cost automatic tide gauge* dengan menggunakan sensor ultrasonik dan dilengkapi juga dengan modul GPS *low-cost*. Pada densor ultrasonik diberikan koreksi kecepatan suara menggunakan sensor suhu agar hasil jarak yang dihasilkan dapat lebih teliti. Instrumen ini dilengkapi dengan sistem *IoT* untuk menyimpan dan mengamati data secara daring dan

near real-time. Instrumen tersebut diharapkan dapat menyelesaikan permasalahan mahalnya harga instrumen *automatic tide gauge* yang beredar di pasar dengan menggunakan komponen hemat biaya, dan memiliki keunggulan pada sistem *IoT* yang akan diterapkan di instrumen pada penelitian ini. Untuk mengetahui ketelitian instrumen, akan dilakukan uji ketelitian bacaan *ultrasonic tide gauge* dengan bacaan pada palem pasang surut, sehingga produk dari penelitian ini tidak hanya memiliki kelebihan pada harganya yang murah jika disbandingkan dengan harga *tide gauge* di pasaran. *Ultrasonic tide gauge* yang dihasilkan pada penelitian ini, selain memiliki biaya pembuatan yang terjangkau, diharapkan juga memiliki kualitas ketelitian yang sesuai standar *automatic tide gauge* yang ada.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang diuraikan, maka dapat dirumuskan permasalahan dalam penelitian sebagai berikut:

1. Bagaimana perancangan instrumen *low-cost ultrasonic tide gauge* dengan sistem *IoT* berbasis Arduino?
2. Bagaimana komparasi data instrumen *low-cost ultrasonic tide gauge* dengan sistem *IoT* berbasis Arduino dengan data bacaan pasang surut manual menggunakan palem pasang surut?

1.3 Batasan Masalah

Berdasarkan rumusan masalah yang telah dipaparkan di atas, penelitian Tugas Akhir ini dibatasi pada:

1. Uji ketelitian alat akan dilakukan dengan komparasi hasil instrumen *low-cost ultrasonic tide gauge* dengan sistem *IoT* berbasis Arduino dan bacaan manual pada palem pasang surut.
2. Lokasi pengujian alat akan dilakukan pada dermaga PT. Pelindo Marine Service yang memiliki kondisi perairan yang tenang.
3. Pengujian alat akan dilakukan terbatas hanya 3 hari 2 malam

1.4 Tujuan Penelitian Tugas Akhir

Tujuan dari penelitian Tugas Akhir ini adalah sebagai berikut:

1. Membuat sebuah instrumen *low-cost ultrasonic tide gauge* dengan sistem *IoT* berbasis Arduino.
2. Mengetahui komparasi data instrumen *low-cost ultrasonic tide gauge* dengan sistem *IoT* berbasis Arduino dengan data pengamatan pasang surut menggunakan palem pasang surut.

1.5 Manfaat Tugas Akhir

Manfaat dari penelitian Tugas Akhir ini adalah sebagai berikut:

1. Bagi seorang surveyor, instrumen ini diharapkan dapat mempermudah pekerjaan pengamatan pasang surut.
2. Bagi mahasiswa dan peneliti, penelitian ini diharapkan dapat membantu untuk melakukan penilitian baru yang dapat mengembankan instrumen serupa dengan kualitas yang lebih baik.

BAB II

TINJAUAN PUSTAKA

2.1 Pasang Surut Air Laut

Pasang surut merupakan salah satu gejala alam yang nyata di permukaan laut, yaitu berupa gerakan vertikal (naik dan turun) secara teratur dan berulang-ulang dari seluruh partikel massa air laut sampai bagian terdalam dari dasar laut. Pasang surut air laut adalah fluktasi muka air laut yang disebabkan oleh gaya tarik bendaa-benda langit terutama matahari dan bulan terhadap masa air laut bumi. Dalam hal ini bulan lebih berpengaruh terhadap pasang surut air laut, dibandingkan dengan matahari. Hal ini dikarenakan jarak antara bulan dengan Bumi relatif lebih dekat jika dibandingkan dengan jarak antara Bumi dan Matahari (Fadhilah 2014).

Menurut Ongkosongo (1989), pembentukan pasang surut air laut dipengaruhi oleh beberapa gerakan, diantaranya ialah :

- Revolusi bulan terhadap bumi, dimana orbitnya berbentuk elips dan memerlukan periode untuk menyelesaikan revolusi itu selama 29,5 hari.
- Revolusi bumi terhadap matahari dengan orbitnya berbentuk elips, periode yang diperlukan adalah 365,25 hari.
- Perputaran bumi terhadap sumbunya sendiri, periode yang diperlukan untuk gerakan ini adalah 24 jam.

Data pasang surut air laut memiliki peran penting dalam mengimplementasikan Undang-Undang Republik Indonesia Nomor 27 Tahun 2007 tentang pengelolaan Wilayah Pesisir dan Pulau-pulau Kecil, yaitu untuk menentukan referensi tinggi yang digunakan pada pengelolaan wilayah pesisir dan laut. Pengetahuan tentang pasang surut sangat penting dalam rekayasa laut, hidrografi, bangunan pantai, dan pelabuhan. Elevasi muka air tertinggi (pasang) dan terendah (surut) sangat penting untuk merencanakan bangunan bangunan tersebut. Sebagai contoh, elevasi puncak

bangunan pemecah gelombang dan dermaga ditentukan oleh elevasi muka air pasang, sementara kedalaman alur pelayaran/ pelabuhan ditentukan oleh muka air surut (Triatmodjo 2008).

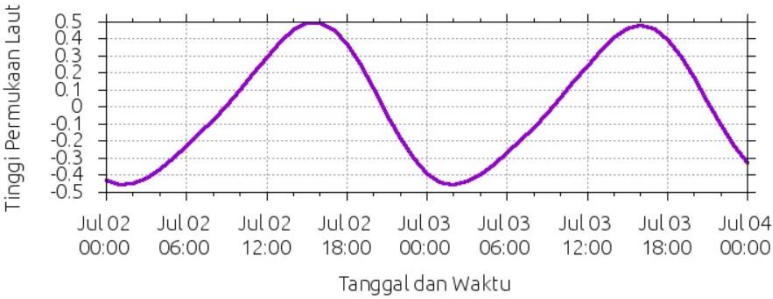
2.2 Tipe Pasang Surut

Pasang surut air laut memiliki tipe yang bervariasi, hal tersebut dapat disebabkan oleh beberapa faktor seperti sistem *amphidromi* lautan, bentuk garis pantai, dan juga pada profil batimetri, dan masih banyak lagi (Haristyana 2012).

Secara umum pasang surut air laut dapat dibedakan menjadi empat tipe, yaitu :

- Pasang surut harian tunggal (*diurnal tide*)

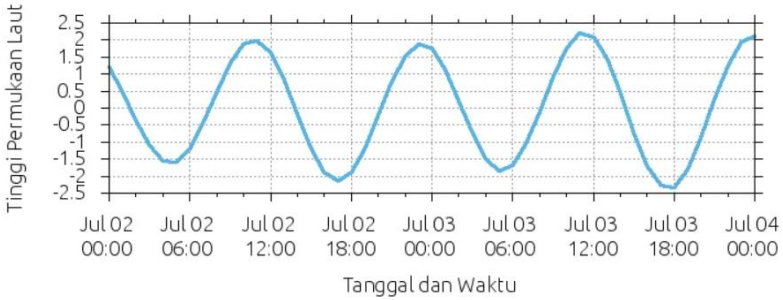
  Merupakan pasang surut dimana dalam satu hari terjadi satu kali air pasang dan satu kali air surut. Periode pasang surut tipe ini adalah 24 jam 50 menit. Pasang surut ini biasanya terdapat di Selat Karimata.



Gambar 2.1 Grafik *Diurnal Tide*
(Taufiqurrahman 2017)

- Pasang surut harian ganda (*Semi Diurnal Tide*)

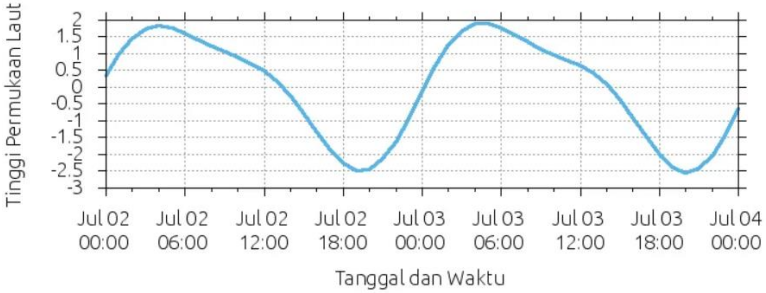
  Merupakan pasang surut yang terjadi dua kali pasang dan dua kali surut yang tingginya hampir sama dalam satu hari. Periode pasang surut rata-rata adalah 12 jam 24 menit. Pasang surut ini biasanya terjadi di Selat Malaka hingga Laut Andaman.



Gambar 2.2 Grafik *Semi Diurnal Tide*
(Taufiqurrahman 2017)

- Pasang surut campuran condong harian tunggal (*Mixed Tide, Prevailing Diurnal*)

  Merupakan pasang surut yang tiap harinya terjadi satu kali pasang dan satu kali surut tetapi terkadang dengan dua kali pasang dan dua kali surut yang sangat berbeda dalam tinggi dan waktu. Pasang surut ini biasanya terdapat di Pantai Selatan Kalimantan dan Pantai Utara Jawa Barat.

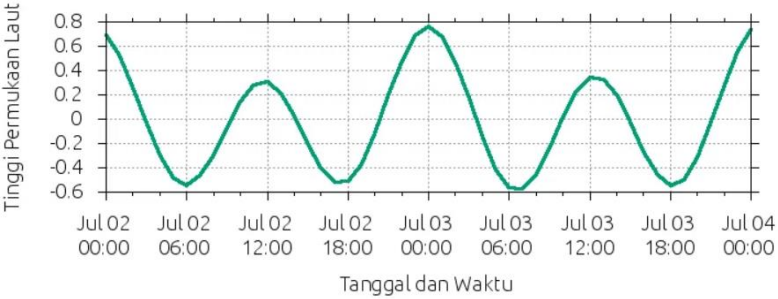


Gambar 2.3 Grafik *Mixed Tide Prevailing Diurnal*
(Taufiqurrahman 2017)

- Pasang surut campuran condong harian ganda (*Mixed Tide, Prevailing Semi Diurnal*)

  Merupakan pasut yang terjadi dua kali pasang dan dua kali surut dalam sehari tetapi terkadang terjadi satu kali pasang dan satu kali. Surut dengan memiliki tinggi dan

waktu yang berbeda. Pasang surut ini terdapat di Pantai Selatan Jawa dan Indonesia Bagian Timur.



Gambar 2.4 Grafik *Mixed Tide Prevailing Semi Diurnal* (Taufiqurrahman 2017)

Tipe pasang surut di suatu perairan dapat diketahui dengan menggunakan persamaan yang biasa disebut dengan bilangan Formzhal (Pugh 1987) seperti persamaan berikut ini:

$$F = \frac{(AO1+AK1)}{(AM2+AS2)} \quad (2.1)$$

Keterangan :

F : Bilangan Formzhal

AO1 : Amplitudo unsur pasut tunggal utama yang disebabkan oleh gaya tarik bulan (meter)

AK1 : Amplitudo unsur pasut tunggal utama yang disebabkan oleh gaya tarik matahari (meter)

AM2 : Amplitudo unsur pasut ganda utama yang disebabkan oleh gaya tarik bulan (meter)

AS2 : Amplitudo unsur pasut ganda utama yang disebabkan oleh gaya tarik matahari (meter)

Dari perhitungan bilangan Formzhal, dapat ditentukan jenis pasutnya dengan syarat sebagai berikut:

$F \leq 0.25$: Pasut ganda

$0.25 < F \leq 1.5$: Pasut campuran dominan ganda

$1.5 < F \leq 3.0$: Pasut campuran dominan tunggal

$F > 3.0$: Pasut tunggal

2.3 Instrumen Pengukur Pasang Surut

Informasi pasang surut air laut sangat penting dalam dunia hidrografi, navigasi, serta rekayasa pantai dan laut. Karena tinggi muka air laut bervariasi terhadap waktu, untuk mengetahui variasi tersebut, diperlukan pengamatan ketinggian muka air laut sebagai fungsi dari waktu. Untuk melakukan pengamatan pasang surut, dibutuhkan stasiun pengamat pasang surut yang dilengkapi dengan *tide gauge. Tide Gauge* berfungsi untuk mengukur tinggi muka air laut. Terdapat dua tipe instrumen pengukur pasang surut, yaitu tipe langsung yang merupakan instrumen *tide staff* atau palem (palem) pasut, dan tipe tidak langsung yang terdiri dari *floating gauge*, *pressure gauge*, *radar gauge*, dan *ultrasonic gauge* (Djaja 1987).

Berikut ini beberapa contoh dari alat-alat pengukur pasang surut air laut secara langsung dan tidak langsung.

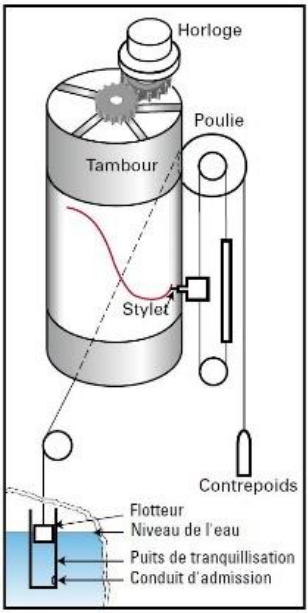
2.3.1 *Tide Staff*

Tide staff atau biasa disebut palem pasang surut merupakan alat pengukur pasang surut yang paling sederhana berupa papan mistar memiliki ketebalan antara 1 sampai 2 inchi dengan lebar 4 sampai 6 inchi, dan dengan pembagian skala yang umumnya dalam sistem meter, sedangkan panjangnya harus lebih besar dari tunggang pasut (*tidal range*). Misalnya, pada perairan dengan tunggang pasut sebesar 2 m, maka ukuran papan skala ini harus lebih dari 2 m (Djaja 1987). Ilustrasi dari *tide staff* dapat dilihat pada Gambar 2.5.

Gambar 2.5 *Tide Staff*

2.3.2 *Floating Gauge*

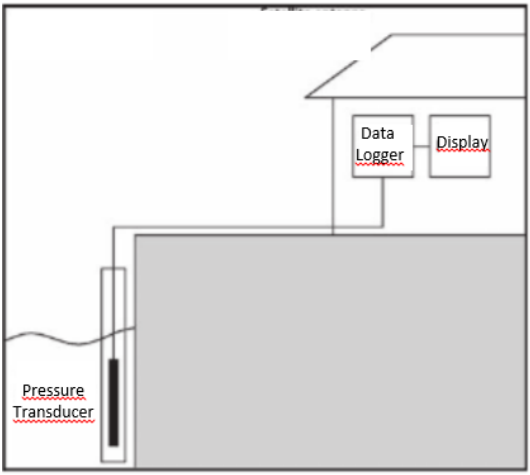
Prinsip kerja alat ini berdasarkan gerakan naik turunnya permukaan laut yang dapat diket ahui melalui pelampung yang dihubungkan dengan alat pencatat. Pengukuran tinggi muka air oleh alat ini dilakukan dengan mendeteksi pergerakan naik turun dari air. Perubahan tinggi pada permukaan sair akan menyebabkan pelampung begerak vertikal (naik turun), pelampung dan penahan beban diikat dengan kabel dan dihubungkan dengan sebuah katrol yang terdapat pada enkoder, sehingga gerakan pelampung dapat memutar katrol. Perputaran yang terjadi pada katrol akan dikonversikan menjadi suatu sinyal digital dan ditransfer ke unit data *logger* melalui kabel transduser. Di dalam data *logger* unit sinyal listrik tersebut diproses sehingga menjadi nilai yang terukur pada instrumen (Djaja 1987). Berikut ini ilustrasi komponen dari *floating gauge* dapat dilihat pada Gambar 2.6.



Gambar 2.6 *Floating Gauge*
(Simon 2007)

2.3.3 *Pressure Gauge*

Prinsip kerja *pressure gauge* hampir sama dengan *floating tide gauge*, namun perubahan naik-turunnya air laut direkam melalui perubahan tekanan pada dasar laut dan tekanan atmosfer di permukaan air laut yang dihubungkan dengan alat pencatat (Djaja 1987). Berikut ini ilustrasi komponen dari *pressure tide gauge* dapat dilihat pada Gambar 2.7.



Gambar 2.7 *Pressure Gauge*
(IOC 2006)

2.3.4 *Radar Gauge*

Alat ini dilengkapi dengan pemancar pulsa radar (*transmitter*), penerima pulsa radar (*receiver*), serta jam berakurasi tinggi. Pada sistem ini, radar memancarkan pulsa-pulsa gelombang radio ke permukaan laut. Pulsa-pulsa tersebut dipantulkan oleh permukaan laut dan diterima kembali oleh radar. Sistem radar ini dapat mengukur ketinggian radar di atas permukaan laut dengan menggunakan waktu tempuh dari pulsa radar yang dikirimkan ke permukaan laut, dan dipantulkan kembali ke radar (Adityayuda 2012). Ilustrasi dari *radar tide gauge* dapat dilihat pada Gambar 2.8.

Gambar 2.8 *Radar Gauge*
(IOC 2006)

2.3.5 *Ultrasonic Gauge*

Ultrasonic gauge merupakan instrumen pengukur pasang surut air laut yang memanfaatkan gelombang ultrasonik. Prinsip kerjanya adalah sensor ultrasonik pada alat memancarkan gelombang ultrasonik, kemudian gelombang tersebut diterima oleh receiver yang ada di alat, kemudian waktu tempuh gelombang akan dicatat dan akan dihitung jarak antara sensor dan muka air laut (Adityayuda 2012).

Gambar 2.9 *Ultrasonic Gauge* MOTIWALI
(Adityayuda 2012)

2.4 Instrumentasi Berbasis Mikrokontroler Arduino

Pada saat ini, *automatic tide gauge* yang beredar di pasaran belum dilengkapi dengan sistem pengiriman data secara telemetri ke sebuah server komputer ataupun website. Kalaupun sistem telemetri tersebut ada, biasanya sistem tersebut dijual secara terpisah. Pada penelitian Tugas Akhir ini, akan dirancang sebuah *automatic tide gauge* yang dilengkapi dengan sistem telemetri ke sebuah server website, atau yang sekarang dikenal dengan istilah *internet of things* (*IoT*). Dengan menggunakan sebuah mikrokontroler, *automatic tide gauge* ini dapat dirancang dengan menggunakan berbagai sensor dan modul, contohnya sensor ultrasonik yang menjadi sensor utama pengukur muka air laut dan modul WiFi yang berfungsi untuk mengirim data ke sebuah server website.

Peranan mikrokontroler untuk instrumentasi menjadi semakin besar diabad ke-21 ini. Berbagai macam produk purwarupa instrumen telah diciptakan menggunakan mikrokontroler. Mikrokontroler merupakan suatu sistem komputer bersifat fungsional yang terdapat didalam suatu *chip*, yang memiliki fungsi sebagai sistem kendali utama pada instrumen yang kita rancang. Salah satu jenis mikrokontroler yang berfungsi sebagai sistem kendali suatu instrumen adalah Arduino (Mahasari 2017).

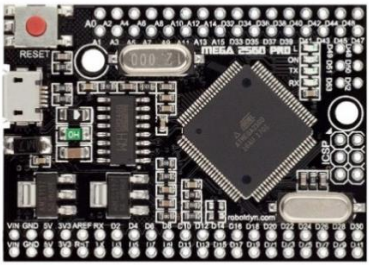
Arduino merupakan sebuah platform pembuatan purwarupa elektronik yang bersifat *open source*, berdasarkan pada perangkat keras dan lunak yang bersifat fleksibel serta mudah digunakan. Arduino bukan hanya sekedar alat pengembangan lebih kepada sebuah alat kombinasi dari *hardware*, bahasa pemrograman dan *Integrated Development Envirorment* (IDE) yang canggih. *Integrated circuit* Arduino dapat berfungsi sampai 10 tahun, asalkan suhu pengoperasiannya tidak lebih dari 85°C (Arduino 2009).

Terdapat banyak proyek atau pengembangan yang dilakukan oleh akademisi menggunakan arduino ini, juga terdapat banyak modul-modul pendukung lainnya agar dapat dihubungkan atau dikombinasikan dengan arduino. Yang membuat arduino digemari banyak orang karena sifat *open source*, baik untuk hardware dan software (Mahasari 2017).

Pada penelitian Tugas Akhir ini, penulis menggunakan papan Arduino Mega 2560 Pro, yang merupakan versi baru dari Arduino Mega 2560. Karena pada penelitian ini penulis akan menggunakan 4 modul dan 2 sensor yang terhubung ke Arduino serta merancang instrumen dengan desain yang *portable*, penulis memilih Arduino Mega 2560 pro dengan alasan memiliki keunggulan berupa ukurannya yang kecil dan dilengkapi 86 pin fungsional untuk koneksi pada sensor maupun modul. Arduino sebenarnya memiliki ketahanan sampai 10 tahun jika suhu pengoperasiannya kurang dari 85°C, karena pada penelitian ini instrumen dibuat dengan desain *portable* dan tidak memiliki sistem pendingin, maka instrumen ini hanya dapat beroperasi sampai beberapa hari saja.

2.5 Arduino Mega 2560 Pro

Arduino Mega 2560 Pro atau Mega Pro merupakan mikrokontroler pengembangan dari *board* Arduino Mega 2560 revisi ke-3 (R3). Mega 2560 Pro ini memiliki ukuran yang lebih kecil disbanding dengan Mega 2560, yaitu 38×55mm. Pada Mega Pro terpasang mikrokontroler ATmega2560 sama seperti Mega 2560 dan menggunakan USB-UART CH340 untuk menghubungkan mikrokontroler ini ke komputer melalui kabel USB. Perbedaan yang juga mencolok pada Mega Pro jika dibandingkan dengan Mega 2560 adalah tidak adanya *power jack*, dan digantikan dengan sebuah pin VIN yang berfungsi untuk menerima tegangan DC 9-12V. Di pasaran, Arduino Mega Pro dijual seharga Rp. 300.000. Arduino Mega Pro dapat ditunjukkan pada Gambar 2.10.



Gambar 2.10 Arduino Mega 2560 Pro
(RobotDyn 2017)

2.5.1 Spesifikasi Arduino Mega 2560 Pro

Spesifikasi dari Arduino Mega 2560 Pro dapat dilihat pada Tabel 2.1 di bawah ini.

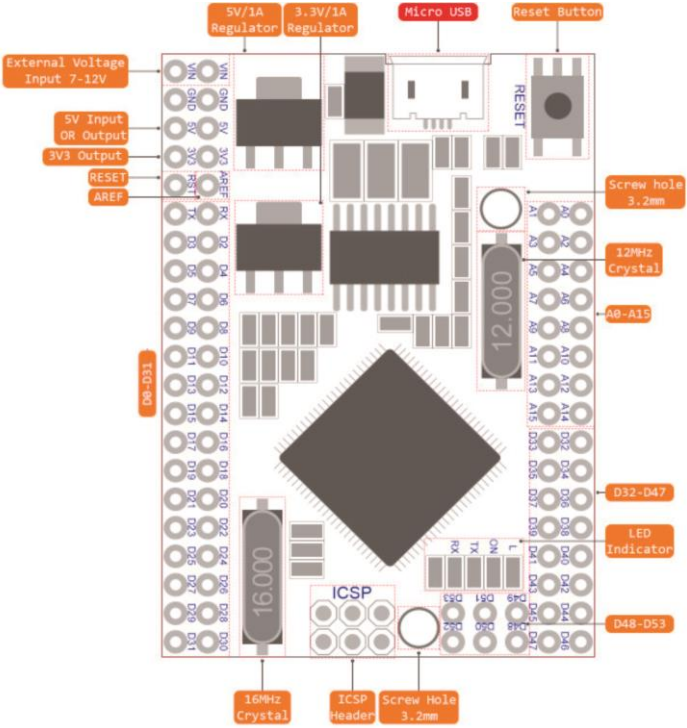
Tabel 2.1 Spesifikasi Arduino Mega 2560 Pro
(RoboDyn 2017)

System	***Specification***
Microcontroller	ATmega2560
USB-TTL converter	CH340
Power Out	5V-800mA
Power IN. System	5V
Power IN. VIN	9-12V
Power Consumption	5V 220mA
Logic Level	5V
USB	Micro USB
Clock Frequency	16MHz
Operating Supply Voltage	5V
Digital I/O	54
Analog I/O	16
Memory Size	256kb

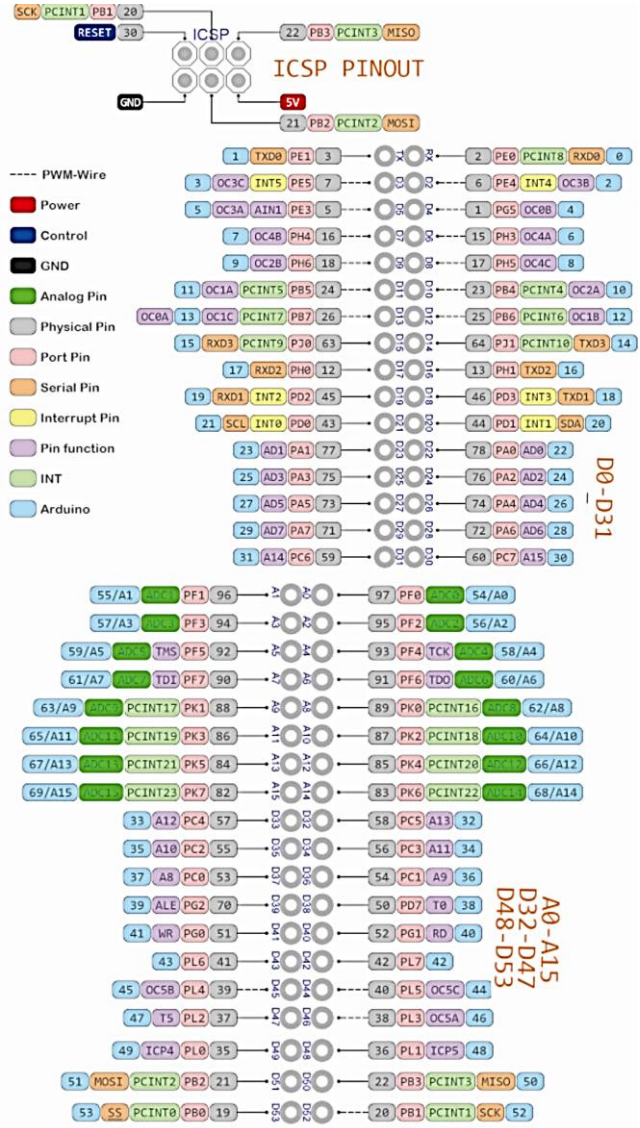
System	***Specification***
Data RAM Type/Size	8Kb
Data ROM Type/Size	4Kb
Interface Type	ISP
Operating temperature	−40C°/+85C°
Length×Width	38×54mm

2.5.2 Konfigurasi Pin Arduino Mega 2560 Pro

Konfigurasi dari Pin Arduino Mega 2560 Pro dapat dilihat pada Gambar 2.11 dan Gambar 2.12.



Gambar 2.11 Konfigurasi *Board* Arduino Mega 2560 Pro (RoboDyn 2017)



Gambar 2.12 Konfigurasi Pin Arduino Mega 2560 Pro (RobotDyn 2017)

Berikut ini penjelasan beberapa pin yang terdapat pada Arduino Mega 2560 Pro:

a. 5V : Pin output yang mengalirkan tegangan 5Volt yang telah terregulator, pada pin ini tegangan telah di atur dari regulator yang terdapat pada papan. Papan Arduino dapat rusak jika memberikan tegangan langsung tanpa melewati regulator melalui pin 5V dan 3,3V.
b. GND : Pin *Ground* atau tegangan negatif.
c. VIN : Pin yang digunakan untuk memberikan tegangan ke papan Arduino dengan tegangan yang disarankan.
d. 3V3 : Sebuah pin output yang menghasilkan tegangan 3,3 V yang dihasilkan oleh regulator pada papan. Arus maksimum yang dihasilkan adalah 50 mA.
e. IOREF : Pin yang memberikan referensi tegangan mikrikontroler. Sebuah shield dikonfigurasi dengan benar agar dapat membaca pin tegangan IOREF dan memilih sumber daya yang tepat atau mengaktifkan penerjemah tegangan (voltage translator) pada output untuk bekerja pada tegangan 5 Volt atau 3,3 Volt.
f. Serial, terdapat 4 serial yang terdiri dari 2 pin.
    - Serial 0 : Pin 0 (RX) dan pin 1 (TX).
    - Serial 1 : Pin 19 (RX) dan pin 18 (TX).
    - Serial 2 : Pin 17 (RX) dan pin 16 (TX).
    - Serial 3 : Pin 15 (RX) dan pin 14 (TX). RX pakai sebagai penerima dan TX untuk transmit data serial TTL. Pin 0 dan pin 1 merupakan pin yang terhubung oleh chip *USB-to-Serial* TTL ATmega16U2.
g. External Interrups: Berfungsi sebagai pemicu perubahan logika pada suatu program di papan Arduino. Pada papan Arduino Mega Pro terdiri dari 6 buah pin Interupt.
    - In 2 (interrupt 0)
    - Pin 3 (interrupt 1)
    - Pin 18 (interrupt 5)
    - Pin 19 (interrupt 4)

- Pin 20 (interrupt 3)
- Pin 21 (interrupt 2).

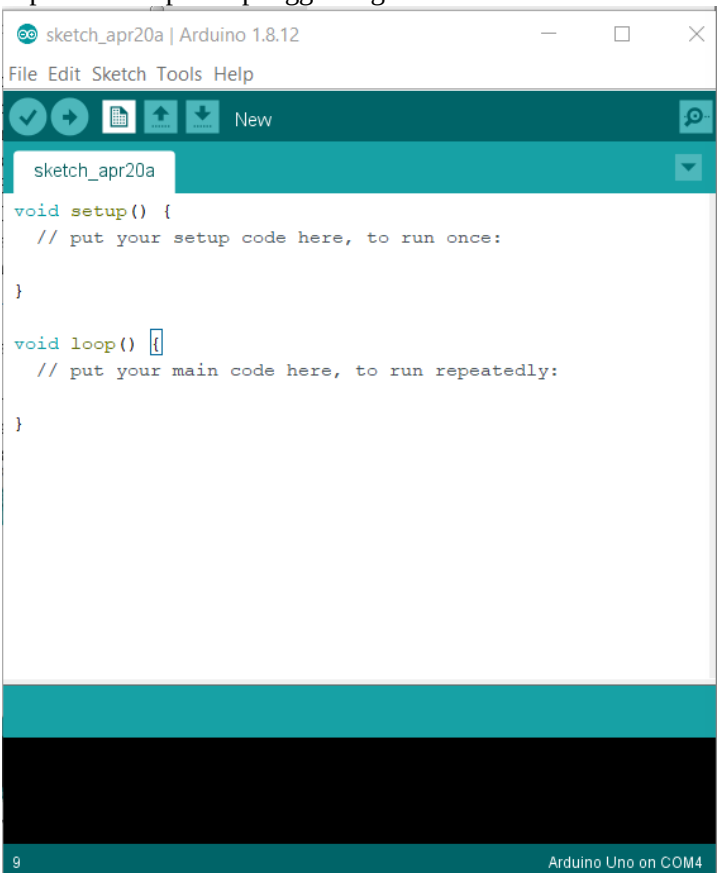
h. PWM: Memiliki fungsi untuk memodulasi lebar pulsa. Pin 2 sampai 13 dan pin 44 sampai 46, yang menyediakan output PWM sebesar 8-bit dengan menggunakan analogWrite.
i. SPI : Pin 50 (MISO), pin 51 (MOSI), pin 52 (SCK), dan pin 53 (SS) untuk mendukung komunikasi SPI dengan menggunakan SPI Library.
j. LED : Pin 13. Pada pin 13 terhubung dengan built-in LED yang terkoneksi pada digital pin 13. Ketika LED menyala (ON) berarti pin ter-set HIGH sedangkan ketika LED mati (OFF) maka pin ter-set LOW.
k. TWI : Pin 20 (SDA) dan pin 21 (SCL) mendukung komunikasi TWI dengan memanfaatkan Wire Library.
l. AREF: Pin referensi tegangan input analog A?D Converter.
m. RESET: Jalur LOW yang digunakan untuk menghidupkan ulang mikrokontroler.

2.6 Arduino IDE

Arduino *Development Environtment* atau disingkat Arduino IDE ini terdiri dari *editor* teks untuk menulis kode, sebuah area pesan, sebuah konsol, sebuah toolbar dengan tools unuk fungsi yang umum dan beberapa menu. Arduino IDE terhubung ke Arduino *Board* untuk mengunggah program dan juga untuk berkomunikasi dengan Arduino (Simanjuntaks 2012).

Software yang ditulis menggunakan Arduino IDE disebut *sketch*. *Sketch* ditulis pada *editor* teks, disimpan dengan berekstensi .ino. Area pesan memberikan informasi dan pesan *error* ketika penyimapanan atau pada saat membuka *sketch*. Pada fungsi konsol menampilkan *output* teks dari Arduino IDE dan juga menampilkan pesan *error* ketika akan mengkompile *sketch*. Pada sudut kanan bawah dari jendela Arduino IDE menunjukan jenis *board* dan *port* serial yang sedang digunakan. Tombol *toolbar* pada Arduino IDE untuk mengecek dan mengunggah *sketch*, membuat,

membuka dan menyimpan *sketch*, dan mengunggah *sketch*, membuat,membuka dan menyimpan *sketch*,dan menampilkan serial monitor (Simanjuntak 2012). Arduino IDE menggunakan bahasa pemprograman C++ dengan tingkat dan versi yang disederhanakan, sehingga lebih mudah dalam pengerjaan programnya. Arduino menggunakan *software processing* yang digunakan untuk menulis program kedalam Arduino yang merupakan merupakan penggabungan antara bahasa C++ dan Java.



Gambar 2.13 Arduino IDE
(Arduino 2009)

2.7 Sensor Ultrasonik

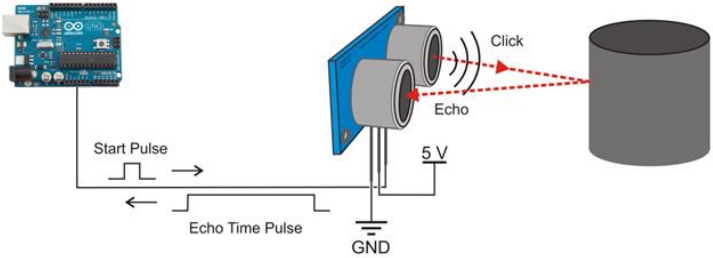
Sensor ultrasonik merupakan sensor yang bekerja berdasarkan prinsip kerja pantulan gelombang suara, dimana sensor menghasilkan gelombang suara yang kemudian menangkap kembali dengan perbedaan waktu sebagai dasar pengindra. Perbedaan waktu antara gelombang suara yang dipancarkan dan diterima kembali adalah berbanding lurus dengan jarak atau tinggi objek yang memantulkannya. Jenis objek yang dapat diindranya adalah zat padat, zat cair dan butiran. Sensor ultrasonik dapat dengan mudah dihubungkan dengan mikrokontroler melalui satu pin I/O.

Sensor ultrasonik pada umumnya digunakan untuk menentukan jarak sebuah objek. Sensor ultrasonik mempunyai kemampuan mendeteksi objek lebih jauh terutama untuk benda-benda yang keras. Pada benda-benda yang keras yang mempunyai permukaan yang kasar gelombang ini akan dipantulkan lebih kuat dari pada benda yang permukaannya lunak. Tidak seperti pada sensor-sensor lain seperti inframerah atau sensor leser. Sensor ulrasonik ini memiliki jangkauan deteksi yang relative luas. Sehingga dengan demikian untuk jarak deteksi yang didapat tanpa menggunakan pengolahan lanjutan (Suspimiany 2015).

2.7.1 Prinsip Kerja Sensor Ultrasonik

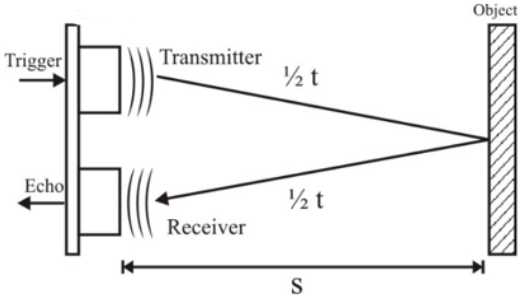
Sensor ultrasonik terdiri dari dua unit, yaitu unit pemancar dan unit penerima. Struktur unit pemancar dan penerima sangatlah sederhana, sebuah kristal piezoelectric dihubungkan dengan jangkar mekanik yang terhubung dengan diafragma penggetar. Struktur atom dari kristal piezoelectric akan berkontraksi (mengikat), mengembang atau menyusut terhadap polaritas tegangan yang diberikan dan ini disebut efek piezoelectric. Kontraksi yang terjadi kemudian diteruskan ke diafragma penggetar, sehingga terjadi gelombang ultrasonik dengan frekuensi 40-400 KHz yang dipancarkan ke udara.

Pantulan gelombang ultrasonik akan terjadi bila ada objek tertentu dan pantulan diterima kembali oleh unit sensor penerima. Selanutnya unit sensor penerima akan menyebabkan diafragma penggetar akan bergetar dan efek piezoelectric menghasilkan sebuah tegangan bolak-balik dengan freukuensi yang sama (Rochim 2016). Berikut ini visualisasi prinsip kerja dari sensor ultrasonik dapat dilihat pada Gambar 2.14.



Gambar 2.14 Prinsip Kerja Sensor Ultrasonik
(Rochim 2016)

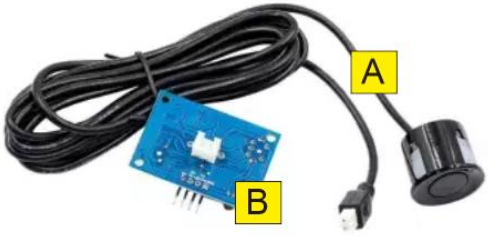
Proses sensoring yang dilakukan pada sensor ini menggunakan metode pantulan untuk menghitung jarak antara sensor dengan objek sasaran. Jarak antara sensor tersebut dihitung dengan cara mengalikan setengah waktu yang digunakan oleh sinyal ultrasonik dalam perjalanannya dari rangkaian pengirim (Tx) sampai diterima oleh rangkaian penerima (Rx) dengan kecepatan rambat dari sinyal ultrasonik tersebut pada media rambat yang digunakannya, yaitu udara. Waktu dihitung ketika pemancar aktif dan sampai ada input dari rangkaianpenerima dan bila pada melebihi batas waktu tertentu rangkaian penerima tidak ada sinyal input maka dianggap tidak ada halang didepannya (Rochim 2016). Prinsip pantulan dari sensor ulrasonik ini dapat dilihat pada Gambar 2.15.



Gambar 2.15 Prinsip Pemantulan Sensor Ultrasonik
(Rochim 2016)

2.7.2 Sensor JSN-SR04T

Sensor JSN-SR04T adalah sensor pengukur jarak berbasis gelombang ultrasonik yang memiliki kelebihan berupa tranduser yang *waterproof* atau kedap air. Prinsip kerja sesnsor ini mirip dengan radar ultrasonik. Gelombang ultrasonik di pancarkan kemdian di terima balik oleh receiver ultrasonik. Jarak antara waktu pancar dan waktu terima adalah representasi dari jarak objek. Sensor ini memiliki ketelitian 0,2cm dan ketelitiannya akan memburuk seiring bertambahnya jarak sensor dengan objek, yaitu pada jarak 3 meter, kesalahan bacaan sensor dapat mencapai 5cm (Chobir 2017). Dipasaran, sensor ini dijual dengan harga Rp. 150.000. Bentuk sensor ultrasonik JSN-SR04T dapat dilihat pada Gambar 2.16.



Gambar 2.16 Sensor JSN-SR04
(Chobir 2017)

Seperti yang ditunjukkan pada Gambar 2.15, sensor ultrasonik JSN-SR04T terdiri dua bagian yaitu:

A. Transduser, berfungsi untuk memancarkan gelombang ultrasonik dan menangkap gelombang ultrasonik yang dipantulkan.
B. *Sensor board*, berfungsi untuk menghubungkan transduser dengan mikrokontroler.

2.7.3 Koreksi Kecepatan Suara

Sensor ultrasonik bekerja dengan menggunakan gelombang suara yang merambat di udara dengan kecepatan suara, kemudian menghitung jarak dengan mengalikan waktu tempuh gelombang dengan kecepatannya. Akan tetapi, kecepatan gelombang suara tergantung pada suhu, dan kecepatan suara meningkat dengan sekitar 0,6 m/s per 1°C (Bies 2009). Untuk mengkoreksi jarak yang dihitung oleh sensor ultrasonik, maka diperlukan sebuah sensor suhu untuk melakukan koreksi kecepatan suara sesuai dengan suhu yang ada dilingkungan alat.

Berdasarkan Bies (2009), persamaan pengaruh suhu terhadap kecepatan suara di udara adalah berikut:

$$V = 331{,}3 + (0{,}606 \times T) \quad \text{.....(2.2)}$$

Keterangan : V = Kecepatan Suara (m/s)
T = Suhu dalam (℃)
$0{,}606$ = Kenaikan kecepatan suara per derajat celcius ($\frac{(m/s)}{1℃}$)
$0{,}331{,}1$ = Kecepatan suara pada suhu 0℃ (m/s)

Proses koreksi kecepatan suara dilakukan pada mikrokontroler, dengan mendapat data suhu dari sensor suhu untuk menghitung kecepatan suara yang ada di area sekitar sensor sesuai dengan Persamaan 2.2. Sensor ultrasonik berguna untuk mencatat waktu tempuh gelombang ultrasonik. Dari data kecepatan suara dengan koreksi suhu dan data waktu tempuh gelombang ultrasonik

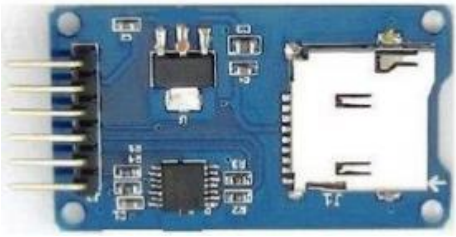
dari sensor ultrasonik, kemudian dapat dihitung sebuah jarak dari sensor ultrasonik ke permukaan yang memantulkan gelombang ultrasonik. Persamaan perhitungan jarak dari data kecepatan suara dan waktu tempuh gelombang ultrasonik adalah berikut:

$$Jarak = \frac{(V \times t)}{2} \quad \text{.....(2.3)}$$

Keterangan : V = Kecepatan Suara (m/s)
t = waktu tempuh gelombang ultrasonik (s)

2.8 Modul *SD Card*

Modul SD *Card* ini menggunakan mikro SD *Card* sebagai sarana untuk penyimpanan dan pengamatan data. Hal ini memungkinkan sistem untuk menambahkan penyimpanan dan data *logging* untuk penyimpanan data sistem, sehingga data-data yang dihasilkan dari sistem yang kita buat dapat secara otomatis tersimpan di *memory card* (Hartono 2013). Di pasaran, modul *SD Card* dijual dengan harga Rp. 50.000. Gambar Modul SD *Card* dapat dilihat pada Gambar 2.17.



Gambar 2.17 Modul SD-*Card*
(Hartono 2013)

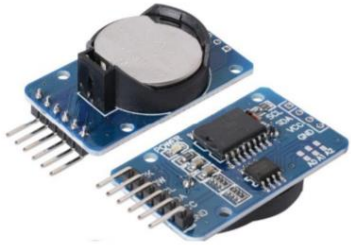
Spesifikasi Modul SD *Card* :

1. Board untuk standar Micro SD (TF) kartu.
2. Pin out untuk penyambungan pada Arduino Uno.
3. Berisi tombol memlilih slot kartu flash.

2.9 Modul RTC DS3231

Modul RTC (*Real-Time* Clock) adalah sebuah chip elektronik yang berupa jam yang dapat menghitung waktu mulai dari detik hingga tahun dengan akurat dan menyimpannya datanya secara real time. Modul RTC biasa digunakan untuk menunjukkan pewaktuan digital yang akan berintegrasi dengan sensor melalui mikrokontroler. Modul ini paling jauh hanya bergeser kurang dari 1 menit per tahunnya, dengan demikian modul ini cocok untuk aplikasi kritis yang sensitif terhadap akurasi waktu yang tidak perlu disinkronisasikan secara teratur terhadap jam eksternal.

Modul ini juga sudah dilengkapi dengan IC AT24C32 yang memberikan EEPROM tambahan sebesar 4 KB (32.768 bit) yang dapat digunakan untuk berbagai keperluan, misalnya untuk menyimpan jadwal (time schedule), menyimpan setelan waktu alarm, menyimpan data hari libur pada kalender, merekam absensi dan lain-lain (Hartono 2013). Di pasaran, modul RTC DS3231 dijual dengan harga Rp.50.000.



Gambar 2.18 Modul RTC DS3231
(Hartono 2013)

2.10 Modul WiFi ESP8266

Modul ESP8266 adalah sebuah komponen chip terintegrasi yang didesain untuk keperluan dunia masa kini yang serba tersambung. Chip ini menawarkan solusi *networking* Wi-Fi yang lengkap dan menyatu, yang dapat digunakan sebagai penyedia aplikasi atau untuk memisahkan semua fungsi *networking* Wi-Fi ke pemproses aplikasi lainnya. ESP8266 memiliki kemampuan *on-*

board prosesing dan *storage* yang memungkinkan chip tersebut untuk diintegrasikan dengan sensor-sensor atau dengan aplikasi alat tertentu melalui pin *input output* yang diintegrasikan dengan Arduino hanya dengan pemrograman singkat.

Modul komunikasi WiFi dengan IC SoC ESP8266EX *Serial-to*-WiFi *Communication Module* ini merupakan modul WiFi dengan harga ekonomis. Kini Anda dapat menyambungkan rangkaian elektronika Anda ke internet secara nirkabel karena modul elektronika ini menyediakan akses ke jaringan WiFi secara transparan dengan mudah melalui interkoneksi serial (UART RX/TX) (Arafat 2019). Dipasaran, modul WiFi ESP8266 dijual dengan harga Rp. 60.000. Ilustrasi Modul ESP8266 dapat dilihat pada Gambar 2.19.

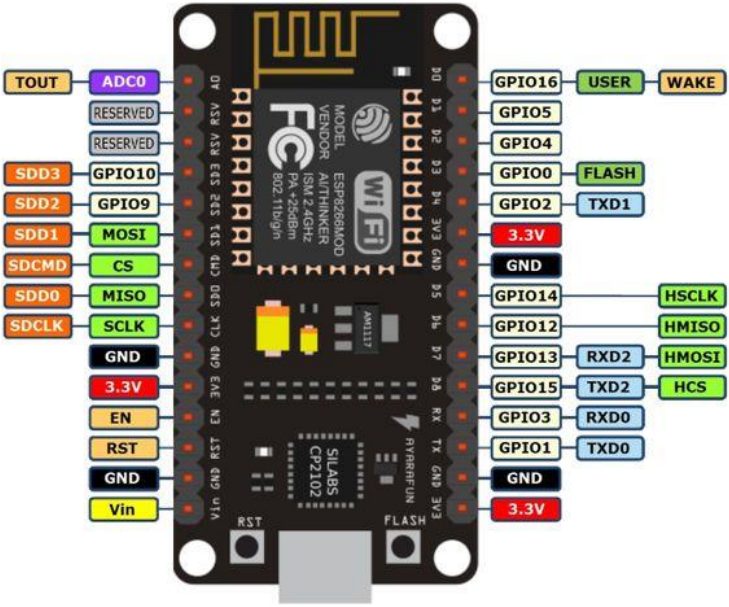
Gambar 2.19 Modul WiFi ESP8266
(Arafat 2019)

2.10.1 Modul WiFi NodeMCU ESP866

NodeMCU merupakan papan pengembangan produk *Internet of Things* (*IoT*) yang berbasiskan *firmware* eLua dan *System on a Chip* (SoC) ESP8266-12E. ESP8266 sendiri merupakan *chip* WiFi dengan *protocol stack* TCP/IP yang lengkap (NodeMCU 2015).

NodeMCU dapat dianalogikan sebagai board arduino-nya ESP8266, sehingga NodeMCU ESP866 ini dapat deprogram menggunakan Arduino IDE. Untuk memprogramnya hanya

diperlukan ekstensi kabel data USB mikro yang menghubungkan NodeMCU dengan PC. NodeMCU di pasaran memiliki harga Rp. 90.000. NodeMCU ESP8266 dapat dilihat pada Gambar 2.20.



Gambar 2.20 NodeMCU ESP8266
(NodeMCU 2015)

NodeMCU memiliki beberapa pin yang memiliki fungsi beragam, berikut ini pin yang terdapat pada NodeMCU ESP8266:

a. ADC : *Analog Digital Converter*, berfungsi untuk mengkonversi data sensor analog menjadi dijital.
b. RST : Berfungsi untuk mereset NodeMCU.
c. EN : *Chip Enable*.
d. IO16 : GPIO16, digunakan untuk membangunkan *chipset* dari mode *deepsleep*.
e. IO14 : GPIO14, HSPCLK.
f. IO12 : GPIO12, HSPI_MISO.

g. IO13 : GPIO13, HSPI_MOSI, UART0_CTS
h. VCC : Catu daya 3.3V
i. CS0 : *Chip selection*
j. MISO : *Slave output, Main input*.
k. IO9 : GPIO9.
l. IO10 : GBIO10.
m. MOSI : *Main output slave input.*
n. SCLK : Jam serial.
o. GND : Pin negatif, *ground.*
p. IO15 : GPIO15, MTDO, HSPICS, UART0_RTS.
q. IO2 : GPIO2, UART1_TXD
r. IO0 : GPIO0
s. IO4 : GPIO4
t. IO5 : GPIO5
u. RXD : UART0_RXD, GPIO3.
v. TXD : UART0_TXD, GPIO1.

2.11 Sensor Suhu DHT22

DHT - 22 (juga disebut sebagai AM2302) adalah sensor kelembaban dan suhu yang memiliki *output* dijital. Sensor ini dilengkapi dengan kapasitor dan thermistor untuk mengukur udara di sekitarnya, dan data kelembapan serta suhu akan memiliki *output* sinyal digital pada pin data. Dalam projek ini menggunakan sensor ini dengan Arduino, suhu kamar & kelembaban akan dicetak ke monitor serial. DHT22 adalah sensor digital yang dapat mengukur suhu dan kelembaban udara di sekitarnya. Sensor ini sangat mudah digunakan bersama dengan Arduino. DHT22 memiliki tingkat stabilitas yang sangat baik serta fitur kalibrasi yang sangat akurat. Koefisien kalibrasi disimpan dalam *One-Time Program memory* (*OTP*), sehingga ketika internal sensor mendeteksi sesuatu, maka modul ini menyertakan koefisien tersebut dalam kalkulasinya. DHT22 termasuk sensor yang memiliki kualitas terbaik, dinilai dari respon, pembacaan data yang cepat, dan kemampuan anti-*interference*. Ukurannya yang kecil, dan dengan transmisi sinyal hingga 20 meter, membuat produk ini cocok digunakan untuk banyak aplikasi-aplikasi pengukuran suhu

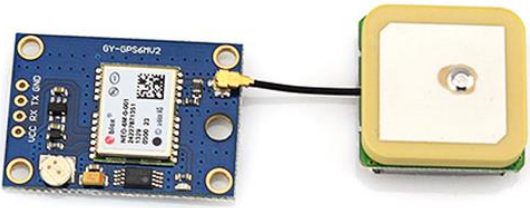
dan kelembaban (Adityayuda 2012). Di pasaran, sensor suhu DHT22 dijual dengan harga Rp. 50.000. Bentuk dari sensor suhu DHT22 dapat dilihat pada Gambar 2.21.

Gambar 2.21 Sensor Suhu DHT22
(Adityayuda 2012)

2.12 Modul GPS NEO-6MV2

Modul GPS (Global Positioning System) APM2.5 NEO-6M berukuran 25x35mm untuk *board* modul, 25x25mm untuk antena. Modul GPS APM2.5 NEO-6MV2 berfungsi sebagai penerima GPS *(Global Positioning System Receiver)* yang dapat mendeteksi lokasi dengan menangkap dan memproses sinyal dari satelit navigasi. Aplikasi dari modul ini melingkupi sistem navigasi, sistem keamanan terhadap kemalingan pada kendaraan/perangkat bergerak, akuisisi data pada sistem pemetaan medan, penjejak lokasi, dan lainnya. Akurasi horisontal dari modul GPS ini mencapai 2,5 meter, walaupun hanya memiliki harga Rp. 68.000. Kelemahan modul ini adalah tidak adanya tidak adanya LNA (*Low Noise Amplifier*) yang berfungsi untuk menangkap sinyal GPS dengan daya rendah, sehingga modul ini harus digunakan pada tempat yang benar-benar terbuka dan lapang, dan harus menunggu beberapa menit agar modul ini dapat menangkap sinyal dari satelit (Laszlo 2016)

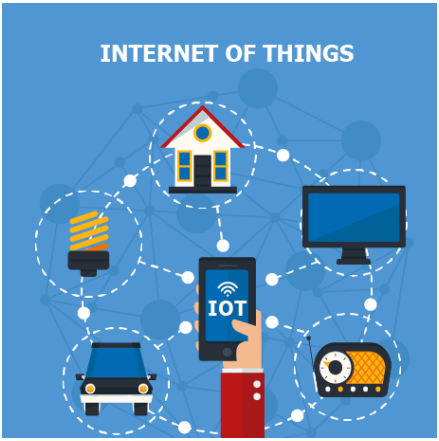
Modul ini kompatibel dengan APM2 dan APM2.5 dengan EEPROM (Electrically Erasable Programmable Read Only Memory) terpadu yang dapat digunakan untuk menyimpan data konfigurasi. Komunikasi antarmuka menggunakan serial TTL (Transistor Transistor Logic) (RX/TX) yang dapat diakses dari 10 mikrokontroler yang memiliki fungsi UART (Universal Asynchronous Receiver Transmitter) atau emulasi serial TTL (pada Arduino dapat menggunakan pustaka komunikasi serial / serial communication library yang sudah tersedia dalam paket Arduino IDE). Baud rate diset secara default pada 9600 bps (Laszlo 2016).



Gambar 2.22 Modul GPS NEO-6MV2
(Laszlo 2016)

2.13 *Internet of Things* (*IoT*)

Internet of Things atau dikenal juga dengan singkatan *IoT*, merupakan sebuah konsep yang bertujuan untuk memperluas manfaat dari konektivitas internet yang tersambung secara terus-menerus. Adapun kemampuan seperti berbagi data, *remote control*, dan sebagainya, termasuk juga pada benda di dunia nyata. Contohnya bahan pangan, elektronik, koleksi, peralatan apa saja, termasuk benda hidup yang semuanya tersambung ke jaringan lokal dan global melalui sensor yang tertanam dan selalu aktif. *Internet of Things* (*IoT*) adalah sebuah konsep/skenario dimana suatu objek yang memiliki kemampuan untuk mentransfer data melalui jaringan tanpa memerlukan interaksi manusia ke manusia (Yudhanto 2007). Ilustrasi konsep *IoT* dapat dilihat pada Gambar 2.23.



Gambar 2.23 Konsep *Internet of Things*

2.14 Perhitungan Kesalahan Relatif

Dalam sebuah pengujian instrumen, terdapat dua jenis perhitungan kesalahan, yaitu hitung kesalahan absolut dan kesalahan relatif. Kesalahan absolut adalah nilai penyimpangan atau kesalahan yang sebenarnya dalam pengukuran sesuatu dengan nilai yang sebenarnya atau dianggap benar. Sedangkan kesalahan relatif adalah kesalahan absolut pengukuran dibandingkan dengan ukuran sebenarnya (Helfrick 1990).

Berikut ini persamaan untuk menghitung kesalahan absolut dan relative dapat dilihat di persamaan nomor (2.4) dan (2.5).

$$EA = |X - X_0| \quad \text{.....(2.4)}$$

$$ER = \frac{EA}{X} \times 100\% \quad \text{.....(2.5)}$$

Dari perhitungan kesalahan relatif, kita dapat menghitung nilai akurasi dari hasil pengukuran dengan persamaan:

$$Akurasi = 100\% - EA \quad \text{.....(2.6)}$$

Keterangan: EA = Kesalahan absolut (milimeter)

X = Nilai sebenarnya atau yang dianggap benar (milimeter)

X_0 = Nilai hasil pengukuran (milimeter)

ER = Kesalahan relatif (persen)

Semakin kecil persentase kesalahan relatif yang didapatkan dari suatu pengukuran, semakin baik dan akurat data hasil pengukuran yang dilakukan.

2.15 *Root Mean Square Error*

Root Mean Square Error atau RMSE adalah metode alternatif untuk mengevaluasi data hasil peramalan atau hasil observasi yang digunakan untuk mengukur tingkat akurasi hasil prakiraan suatu model atau hasil observasi. RMSE merupakan nilai rata-rata dari jumlah kuadrat kesalahan, juga dapat menyatakan ukuran besarnya kesalahan yang dihasilkan oleh suatu model prakiraan. Nilai RMSE rendah menunjukkan bahwa variasi nilai yang dihasilkan oleh suatu model prakiraan mendekati variasi nilai obeservasinya (Wintolo 2019). Berikut ini adalah persamaan untuk menghitung nilai RMSE dapat dilihat pada persamaan (2.7):

$$RMSE = \frac{\sqrt{\sum_{i=1}^{n}(Oi-Ai)^2}}{n} \quad \text{.....(2.7)}$$

Keterangan: Oi = Nilai data observasi atau prediksi pada periode i (milimeter)

Ai = Nilai data aktual pada periode I (milimeter)

n = Jumlah data

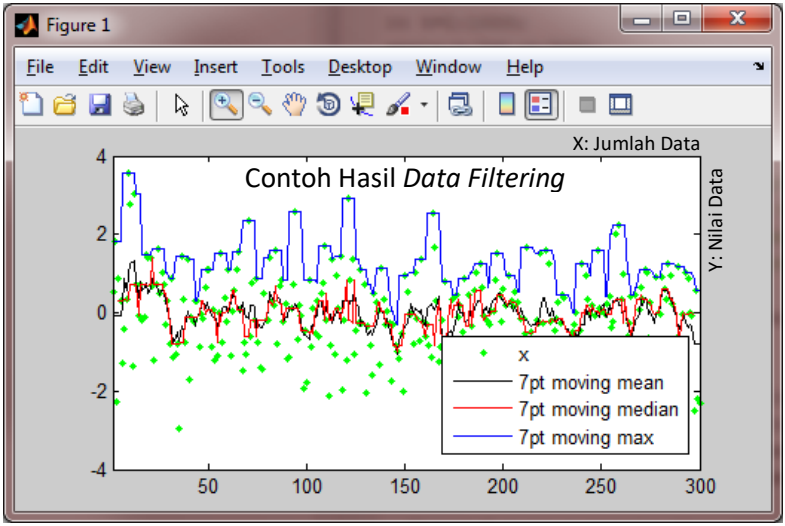
2.16 *Data Filtering*

Data Filtering atau penyaringan data adalah teknik statistik yang berfungsi untuk menghilangkan *outlier* atau *noise* dari kumpulan data untuk membuat pola-pola tertentu dari data lebih terlihat. Gagasan di balik *filtering* data adalah untuk

mengidentifikasi perubahan yang disederhanakan untuk membantu memprediksi tren dan pola yang berbeda. *filtering* data berperan penting bagi para ahli statistik, elektronika, intrumentasi, dan masih banyak lagi (Simonoff 1998).

Dalam penelitian Tugas Akhir ini, akan digunakan teknik *filtering* data untuk menghilangkan *outlier* dan memperhalus kurva pasang surut hasil pengukuran instrumen *ultrasonic tide gauge.*

Metode *filtering* ada beragam tergantung perangkat lunak pemrograman atau perangkat lunak statistik yang digunakan. Pada penelitian ini, penulis menggunakan perangkat lunak MATLAB yang terdapat beberapa metode untuk *filtering data* yang akan dijelaskan di bawah ini.



Gambar 2.24 Metode *Data Filtering*

2.16.1 *Moving Average*

Metode ini merupakan penyaringan data menggunakan rata-rata data sebelum dan data sesudah yang nantinya akan diganti. Dimana, data sebelum dan sesudahnya ditentukan oleh panjang

rentang data atau *span* yang diinginkan (Smith 1999). Secara umum persamaan metode *moving average* dapat dilihat pada persamaan (2.8):

$$y_s(i) = \frac{1}{2N+1}(y(i+N) + y(i+N-1) + \cdots + y(i-N) \ldots (2.8)$$

Keterangan $y_s(i)$ = Data hasil moving average filtering ke i (milimeter)

$2N + 1$ = Rentang data sebagai pembagi

N = Jumlah data tetangga di sisi sebelah

2.16.2 *Moving Median*

Metode ini merupakan penyaringan data yang tepat di tengah dari data sesudah dan sebelum yang masuk ke dalam rentang yang ditentukan (Gonzalo 2005). Secara umum persamaan metode *moving median* adalah sebagai berikut:

$$Moving\ Median = Median\ (N1, N2, N3, \ldots, N_n) \ldots\ldots\ldots\ldots\ldots (2.9)$$

Keterangan $Median$ = Fungsi untuk mencari nilai tengah di dalam rentang data (milimeter)

$N1 \ldots Nn$ = Urutan data di dalam rentang

2.17 Kontrol Kualitas Data

Verifikasi data pasut dapat dilakukan dengan menggunakan kontrol kualitas data pasut sehingga dapat menemukan keanehan atau anomali terhadap data pasut. Kontrol kualitas data pasut dapat dilakukan dengan menggunakan uji global dengan rentang kepercayaan 3 kali standar deviasi (3 σ) atau 99,7%. Rentang ini dipilih berdasarkan pada rentang kepercayaan yang dipakai oleh Badan Informasi Geospasial (BIG) (Khasanah 2014). Persamaan dapat dilihat pada Persamaan (2.10)

$$\sigma = \sqrt{\frac{\sum_{i=1}^{n}(Xn-\overline{X})^2}{n-1}}) \quad (2.10)$$

Keterangan $\sigma =$ Standar deviasi (milimeter)

$Xn =$ Data observasi ke-n (milimeter)

$\overline{X} =$ Nilai rata-rata data pada sampel data (milimeter)

$n =$ Jumlah data pada sampel data

Batas kiri dan batas kanan akan menentukan data pengamatan pasut. Data yang diterima adalah data yang terletak antara batas kiri dan batas kanan. Apabila data kurang dari batas kiri dan lebih dari batas kanan, data tersebut akan tertolak. Batas kiri dan batas kanan dapat dihitung dengan persamaan sebagai berikut:

Batas kiri = $\overline{X}$-3σ, dalam satuan meter.....(2.11)

Batas Kanan = $\overline{X}$+3σ, dalam satuan meter.....(2.12)

2.18 Penelitian Terdahulu

2.18.1 Penelitian Oleh Mufidul Khoir

Pada penelitian yang berjudul "Rancang Bangun Alat Monitoring Pasang Surut Air Laut Berbasis Internet of Thing (*IoT*)", dihasilkan sebuah instrumen pengukur pasang surut air laut dengan sistem *IoT* (Khoir 2018). Instrumen pengukur pasang surut dilengkapi sensor HC-SR04, yang merupakan sensor ultrasonic dengan jangkauan sensor 2-400cm, sudut pengukuran sebesar 15^0, frekuensi 40kHz, dan resolusi 0,3cm. Akan tetapi, sensor HC-SR04 ini tidak memiliki kemampuan tahan air, sehingga kurang cocok untuk digunakan untuk pengukuran di daerah air. Pada penelitian ini juga menggunakan *webserver* Thingspeak.com untuk menyimpan dan menampilkan data pasang surut secara *real-time*. Pengukuran dilakukan selama 3 hari dengan durasi per-hari 24 jam, Perhitungan tingkat ketelitian alat dengan membandingkan data ukuran sensor dengan data pasang surut milik BIG. Dari hasil

penelitian ini, didapatkan hasil perhitungan ketelitian alat sebesar 99,8398%, atau dengan tingkat kesalahan alat sebesar 0,1602%.

2.18.2 Penelitian Oleh Abdul Chobir Dkk

Penelitian dengan judul "Sistem Deteksi Elevasi Permukaan Air Sungan Dengan Sensor Ultrasonik Berbasis Arduino" menghasilkan instrumen pengukur elevasi permukaan air sungai dengan menggunakan sensor ultrasonik JSN-SR04T (Chobir, dkk 2017). Sensor JSN-SR04T dipilih karena memiliki kualitas yang lebih bagus disbanding dengan HC-SR04, karena harganya yang juga lebih mahal empat kali lipat. Pada penelitian ini, system monitoring hanya menggunakan monitor serta modul SD-*card* untuk menyimpan data. Pada penelitian ini, dilakukan uji akurasi untuk mengukur tingkat kesalahan alat, dengan membandingkan data ukuran alat dengan ukuran menggunakan meteran. Didapatkan hasil perhitungan tingkat kesalahan alat pada penelitian ini sebesar 0,75%.

2.18.3 Penelitian Oleh Naufal Ananda Dkk

Penelitian dengan judul "Rancang Bangun Instrumen *Tide Gauge Multi-Parameter*" menghasilkan instrumen yang memiliki berbagai macam sensor untuk mengukur pasang surut air laut, pengukur curah hujan, dan untuk mengukur suhu laut (Ananda, dkk 2019). Sensor ultrasonik yang digunakan adalah JSN-SR04T yang memiliki kelebihan tahan air, memiliki jangkauan jarak 25-400cm, sudut pengukuran 45^0, frekuensi 40kHz, dan resolusi 0,2cm. Sensor DS18B20 mampu mengukur suhu di air mulai dari -55^0C sampai dengan 125^0C dengan akurasi kurang lebih $0,5^0$C. Alat ukur curah hujan bernama *Pyrex Rain Gauge*. Pada penelitian ini, system monitoring menggunakan system *internet of thing*, dan *platform* yang digunakan untuk menampilkan data adalah sebuah perangkat lunak android. Pada penelitian ini, dilakukan kalibrasi untuk mengukur tingkat kesalahan alat, dengan membandingkan data ukuran alat dengan ukuran menggunakan meteran. Dilakukan

juga perhitungan kesalahan pengukuran suhu, dan kesalahan pengukuran curah hujan. Kemudian perhitungan Didapatkan hasil perhitungan tingkat kesalahan sensor JSN-SR04T pada penelitian ini sebesar 0,55%. Kesalahan dari pengukuran suhu laut didapatkan sebesar $0{,}85^0$C. Kesalahan pengukuran curah hujan sebesar 0,1%

"Halaman ini sengaja dikosongkan"

BAB III

METODOLOGI PENELITIAN

3.1 Waktu dan Lokasi Penelitian

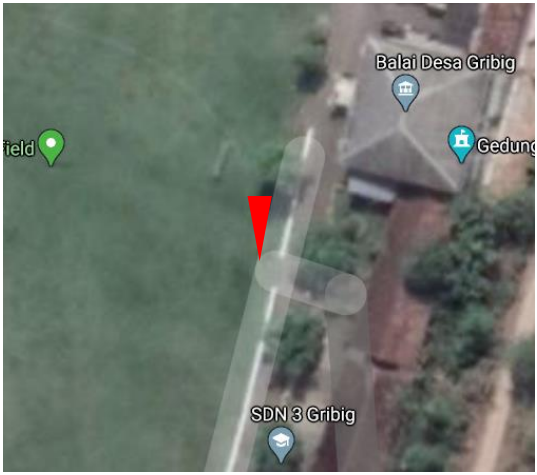
Penelitian Tugas Akhir Rancang Bangun dan Uji Ketelitian Instrumen *Low-Cost Ultrasonic Tide Gauge* Dengan Sistem *Internet of Things* (*IoT*) Berbasis Arduino ini terdapat dua tahap. Tahap pertama adalah pembuatan instrumen *low-cost ultrasonic tide gauge* yang akan dilakukan di Laboratorium Geomarine, Departemen Teknik Geomatika ITS.

Tahap kedua merupakan pengujian instrument yang akan dilakukan di Dermaga Parkiran PT. Pelindo Marine Service yang berlokasi di Jl. Prapat Kurung Utara No.58 Surabaya dengan koordinat 7° 12' 34"S dan 112° 43' 31"E. Lokasi pengujian alat ditunjukkan oleh panah merah pada Gambar 3.1. Pengujian dilakukan di lokasi tersebut karena dermaga memiliki perairan yang lebih tenang.

Pengujian instrument dilakukan selama 3 hari 2 malam, tepatnya tanggal 14 Februari 2020 pukul 19.30 WIB sampai 16 Februari 2020 pukul 20.00 WIB.

Gambar 3.1 Lokasi Pengujian Alat

Pada tanggal 15 April 2020, dilakukan penambahan modul GPS Neo6mV2, dan pengujian modul GPS dilakukan pada tanggal 18 April 2020 dan dilakukan di Lapangan Desa Gribig, Kecamatan Gebog, Kabupaten Kudus dengan koordinat 6° 47' 10"S dan 110° 49' 39"E yang ditunjukkan pada panah merah pada Gambar 3.2.



Gambar 3.2 Lokasi Pengujian Modul GPS

3.2 Alat dan Bahan

Alat dan bahan yang digunakan dalam penelitian Tugas Akhir ini meliputi alat dan bahan yang digunakan pada kegiatan pembuatan instrument, pengujian instrument, dan pengolahan data. Berikut ini daftar alat dan bahan yang digunakan:

3.2.1 Alat

Berikut adalah peralatan yang digunakan dalam penelitian Tugas Akhir:

1. Arduino IDE
2. Solder
3. Microsoft Office 2013
4. MATLAB R2016a
5. Gergraji Paralon

6. Palem Pasang Surut
7. Adaptor AC ke DC 9V

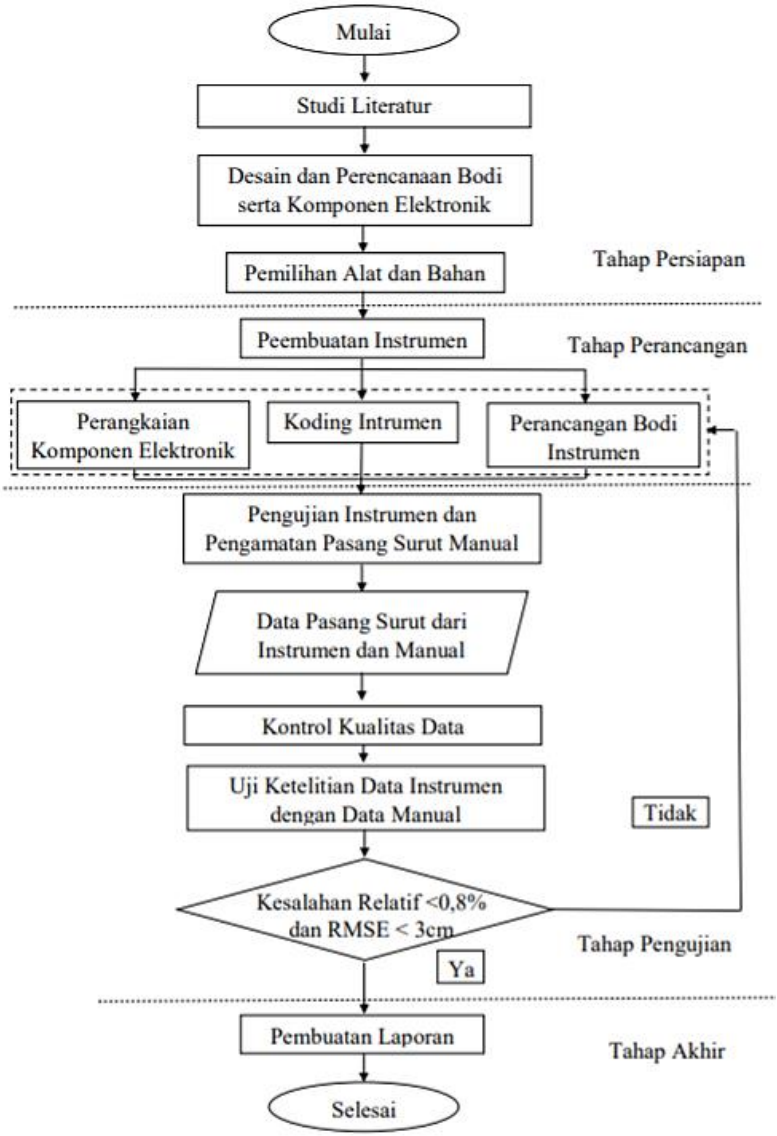
3.2.2 Bahan

Bahan yang digunakan dalam pelaksanaan penelitian Tugas Akhir ini antara lain:

1. Arduino Mega 2560 Pro
2. Sensor Ultrasonik JSN-SR04T
3. Modul SD-*Card*
4. Modul RTC DS3231
5. Modul WiFi ESP8266
6. Modul LCD 1602
7. Modul GPS Neo6mV2
8. Timah
9. Paralon
10. Klem besi
11. Kabel Konektor
12. Kotak *lock and lock*

3.3 Metode Penelitian

Metodologi yang akan digunakan pada penelitian Tugas Akhir ini ditunjukkan pada diagram alir Gambar 3.3.



Gambar 3.3 Diagram Alir Penelitian

Berikut ini adalah penjelasan dari tahapan diagram alir yang terdapat pada Gambar 3.3:

3.3.1 Tahap Persiapan

1. Studi Literatur

   Studi literatur dilakukan untuk mengumpulkan referensi yang menunjang pengerjaan penelitian Tugas Akhir mulai dari tahap perancangan instrumen sampai uji ketelitian dengan hasil bacaan manual. Adapun studi literatur yang menjadi pokok bahasan utama yaitu:

   - *Automatic tide gauge*
   - Arduino
   - Sensor ultrasonik JSN-SR04T
   - Sensor suhu DHT22
   - Modul RTC DS3231
   - Modul SD-*Card*
   - Modul WiFi ESP8266
   - Modul GPS Neo6mV2
   - Serta *ultrasonic tide gauge*

2. Desain dan perencanaan bodi serta komponen elektronik

   Pembuatan desain bodi instrumen *ultrasonic tide gauge* yang ergonomis dan dapat digunakan secara efektif serta efisien. Kemudian perencanaan dan desain dari komponen elektronik yang akan digunakan, sehingga kita dapat mengetahui desain jalur komponen elektronik yang sesuai dengan bodi instrumen.

3. Pemilihan alat dan bahan

   Pemilihan alat dan bahan dilakukan sesuai dengan hasil desain dan perencanaan dari bodi dan komponen elektronik instrumen *ultrasonic tide gauge*.

3.3.2 Tahap Perancangan

1. Perangkaian komponen elektronik

   Dari hasil pemilihan alat dan bahan untuk komponen elektronik, kemudian dilakukan perangkaian sesuai desain jalur komponen elektronik yang sudah direncanakan.

2. Melakukan koding untuk pemrograman instrumen

   Setelah komponen elektronik selesai dirangkai, kemudian dilakukan pembuatan kode untuk pemrograman instrumen yang selanjutnya akan di masukkan kedalam mikrokontroler Arduino Mega 2560 Pro. Serta melakukan pemrograman untuk sistem mengunggah data secara *real-time* saat alat mulai beroperasi pada website.

3. Melakukan perancangan bodi

   Setelah komponen elektronik dan programnya selesai, kemudian dilakukan perancangan bodi instrumen *ultrasonic tide gauge* dengan bahan yang sesuai dari tahap desain dan perencanaan.

3.3.3 Tahap Pengujian

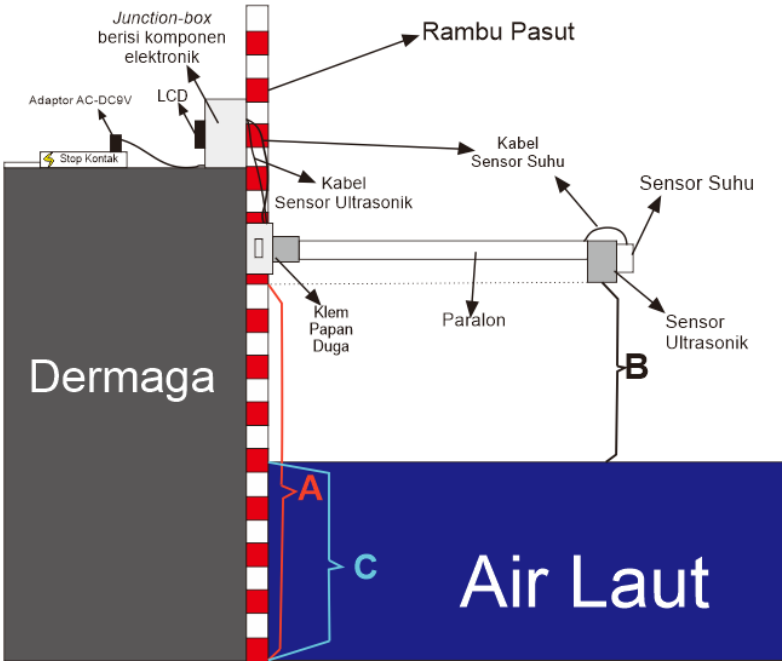
1. Pengujian instrumen *ultrasonic tide gauge* dan pengamatan pasang surut manual

   Setelah alat selesai dirancang, kemudian dilakukan pengujian alat tersebut di lokasi yang telah ditetapkan. Alat dipasang dan dibiarkan untuk melakukan pengamatan pasang surut secara otomatis selama 3 hari. Sedangkan kita juga perlu melakukan pengamatan pasang surut secara manual selama 3 hari juga, sambil mengawasi kinerja alat.

2. Konversi data instrument dengan data palem pasut

   Sebelum dilakukan perhitungan ketelitian instrument, dilakukan konversi bacaan instrument untuk menyamakan referensi pembacaan data dengan bacaan palem pasut. Hal tersebut dilakukan karena instrument mengukur jarak dari sensor ke permukaan laut, sedangkan palem pasut mengukur dari nol palem ke permukaan laut. Penjelasan konversi data

intrumen dengan palem pasut disesuaikan dengan konfigurasi pada saat pemasangan alat yang dijelaskan pada Gambar 3.4.



Gambar 3.4 Konfigurasi *Ultrasonic Tide Gauge*

Berikut ini persamaan konversi bacaan instrument:

$$C = A - B \quad (3.1)$$

Keterangan:

A = Tinggi sensor terhadap nol palem pasut (milimeter)

B = Bacaan sensor (milimeter)

C = Elevasi pasut (milimeter)

Setelah konfigurasi ditentukan, di mana faktor yang terpenting adalah tinggi sensor terhadap nol palem, kemudian

persamaan (9) akan dimasukkan ke dalam program untuk Arduino agar hasil bacaan alat langsung memiliki referensi yang sama dengan nol palem pasut.

3. Kontrol kualitas data

Setelah dilakukan pengamatan pasut, data dari instrumen akan dilakukan kontrol kualitas data untuk menemukan beberapa anomali seperti *noise*, *spike*, atau *outliers* pada data. Kontrol kualitas data ini akan mengacu pada uji global dengan rentang kepercayaan yang dipakai adalah tiga standar deviasi 3σ atau 99,7%. Apabila terdapat anomali.

4. Uji ketelitian dan kesalahan instrumen

Uji ketelitian dan kesalahan instrumen akan dilakukan dengan perhitungan kesalahan relatif dan ketelitiannya, serta perhitungan RMSE untuk mengetahui tingkat ketelitian alat dalam persen dan tingkat kesalahan alat dalam satuan jarak jika dibandingkan dengan bacaan manual dari palem pasut.

Berdasarkan penelitian yang telah dilakukan oleh Abdul Chobir, didapatkan kesalahan relatif alat sebesar 0,75%, pada alat Abdul Chobir menggunakan sensor ultrasonik JSN-SR04T,sama seperti instrumen *low-cost ultrasonic tide gauge* dengan sistem *IoT* yang dirancang penulis, tetapi tidak menggunakan koreksi kecepatan suara dengan sensor suhu.

Dari penelitian Abdul Chobir tersebut, penulis menetapkan batas kesalahan relatif alat adalah 0,8% atau ketelitiannya lebih dari 99,2% dan nilai RMSE kurang dari 3cm, maka pengujian alat akan penulis anggap berhasil . Jika target tingkat kesalahan alat belum dicapai, maka akan dilakukan proses *coding* ulang serta penggantian dan kalibrasi sensor atau dengan mengubah bodi dari intsrumen agar mendapatkan hasil pengukuran yang diharapkan. Selain itu, pemilihan batas kesalahan alat didasarkan pada tingkat

ketelitian sensor yang hanya 0,2cm dan akan semakin buruk jika jarak antara sensor dan objek semakin jauh.

3.3.4 Tahap Akhir

Pada tahap akhir, akan dilakukan pembuatan laporan hasil penelitian dari rancang bangun instrumen *ultrasonic tide gauge*, yang berisi mengenai hasil dari tahap perencanaan alat, perancangan alat, dan pengujian alat, pengujian data.

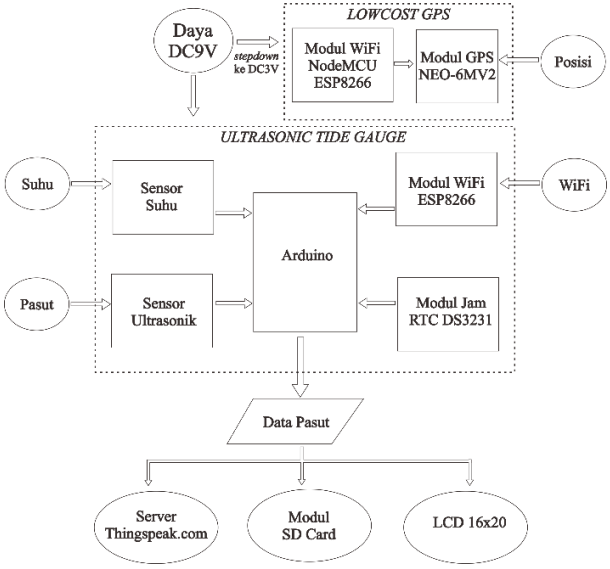
"Halaman ini sengaja dikosongkan"

BAB IV

HASIL DAN PEMBAHASAN

4.1 Pembuatan Instrumen

Pada penelitian Tugas Akhir ini, hasil utamanya adalah instrumen *low-cost ultrasonic tide gauge* dengan sistem *IoT* berbasis Arduino. Prinsip kerja instrumen *low-cost ultrasonic tide gauge* dengan sistem *IoT* ini pada dasarnya adalah mengukur jarak antara sensor ultrasonik ke permukaan air laut dengan dilakukan koreksi kecapatan suara dengan sensor suhu, kemudian data jarak tersebut dikonversi untuk mendapatkan elevasi air laut yang datanya akan disimpan ke dalam sebuah *SD-Card* , dengan modul WiFi, data akan dikirim ke server website *Thingspeak.com* agar bisa dipantau secara *realtime* dari tempat lain, dan data akan ditampilkan pada LCD yang terdapat pada instrumen. Modul GPS akan bekerja secara terpisah menggunakan NodeMCU. Berikut ini diagram kerja instrument dapat dilihat pada Gambar 4.1.

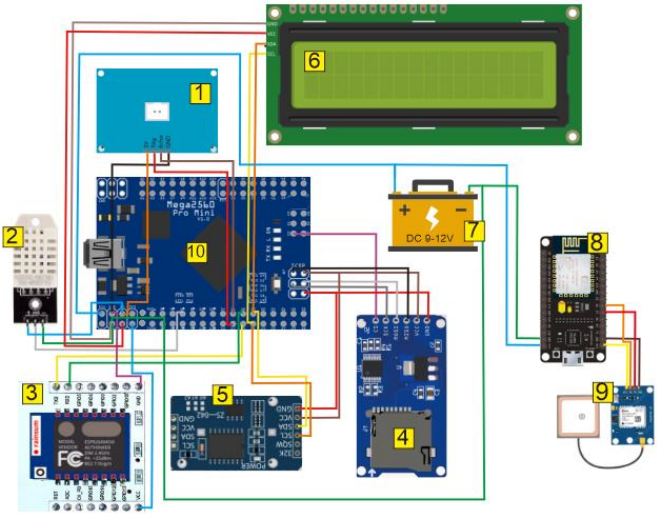


Gambar 4.1 Diagram Kerja Alat

Dalam pembuatan instrument *low-cost ultrasonic tide gauge* dengan sistem *IoT* ini, terdapat beberapa tahap yang harus dilakukan, berikut ini merupakan hasil pengerjaain setiap tahapan dalam pembuatan instrument *ultrasonic tide gauge.*

4.1.1 Desain *Electrical Circuit*

Tahapan paling awal dalam pembuatan instrument adalah merencanakan *electrical circuit* atau jalur koneksi antar komponen elektronik, hal ini bertujuan agar tiap komponen dapat dikoneksikan dengan benar dan instrument dapat bekerja. Adapun *electrical circuit* pada instrument *low-cost ultrasonic tide gauge* dengan sistem *IoT* berbasis Arduino adalah sebagai berikut.



Gambar 4.2 *Electrical Circuit* Instrumen *Ultrasonic Tide Gauge*

Berikut ini penjelasan dari Gambar 4.2 mengenai koneksi *electrical circuit* di tiap komponen:

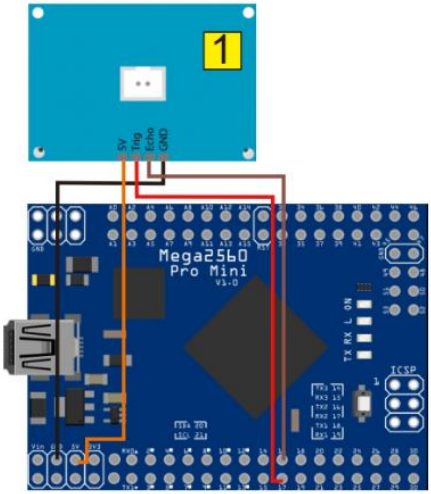
1. Koneksi pada sensor suhu DHT

   Pada sensor ultrasonik JSN-SR04 memiliki 4 pin, tiap pin tersebut memiliki fungsi yang berbeda. Berikut ini nama dan

fungsi 4 pin yang terdapat pada sensor ultrasonik JSN-SR04T dan koneksinya dengan pin di Arduino Mega 2560 Pro:

- Pin VCC/5v: untuk sumber tegangan positif 5V sensor disambung ke pin 5V Arduino Mega 2560 Pro.
- Pin Trig: pin *trigger* digunakan untuk pembangkit gelombang ultrasonik disambung ke pin 17 Arduino Mega 2560 Pro.
- Pin Echo: pin *echo* digunakan untuk menangkap gelombang pantulan sensor ultrasonik disambung ke pin 16 Arduino Mega 2560 Pro.
- Pin GND: untuk sumber tegangan negatif sensor disambung ke pin GND Arduino Mega 2560 Pro.

Berikut ini gambaran koneksi keempat pin sensor ultrasonik JSN-SR04T ke Arduino Mega 2560 Pro.



Gambar 4.3 *Circuit* Sensor Ultrasonik

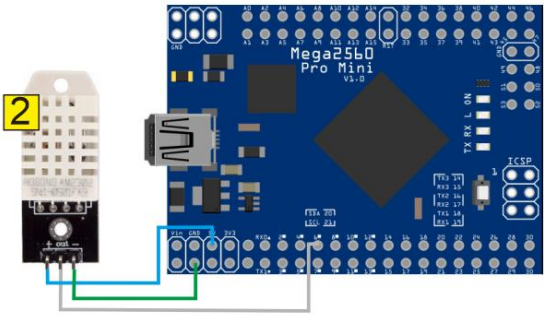
2. Koneksi pada sensor suhu DHT22

Pada sensor suhu DHT22 memiliki 3 pin, tiap pin tersebut memiliki fungsi yang berbeda. Berikut ini nama dan fungsi 3

pin yang terdapat pada sensor suhu DHT22 dan koneksinya dengan pin di Arduino Mega 2560 Pro:

- Pin + : untuk sumber tegangan positif 5V sensor disambung ke pin 5V Arduino Mega 2560 Pro.
- Pin Out: pin *out* berfungsi untuk menyalurkan data bacaan suhu ke mikrokontroler, disambung ke pin 6 Arduino Mega 2560 Pro.
- Pin - : untuk sumber tegangan negativf sensor disambung ke pin GND Arduino Mega 2560 Pro.

Berikut ini gambaran koneksi keempat pin sensor suhu DHT22 ke Arduino Mega 2560 Pro.



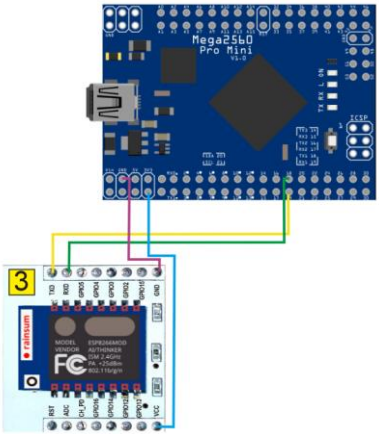
Gambar 4.4 *Circuit* Sensor Suhu

3. Koneksi pada Modul WiFi ESP8266

Pada modul WiFi ESP8266 memiliki 16 pin, tiap pin tersebut memiliki fungsi yang berbeda. Pada instrumen *ultrasonic tide gauge* dengan sistem *IOT* ini, hanya 4 pin yang akan digunakan karena penggunaan modul ini di instrumen ini hanya untuk menerima data dari Arduino dan mentransfer data ke server Thingspeak.com. Berikut ini nama dan fungsi ke 4 pin yang terdapat pada ESP8266 dan koneksinya dengan pin di Arduino Mega 2560 Pro:

- Pin 3V3 : untuk sumber tegangan positif 3V ESP8266 disambung ke pin 3V Arduino Mega 2560 Pro.
- Pin TXD: untuk mentransfer data ke server Thingspeak.com, disambung ke pin 19 Arduino Mega 2560 Pro.
- Pin RXD: untuk menerima data dari Arduino, disambung ke pin 18 Arduino Mega 2560 Pro.
- Pin GND : untuk sumber tegangan negatif modul disambung ke pin GND Arduino Mega 2560 Pro.

Berikut ini gambaran koneksi ke 4 pin ESP8266 ke Arduino Mega 2560 Pro.



Gambar 4.5 *Circuit* Modul WiFi

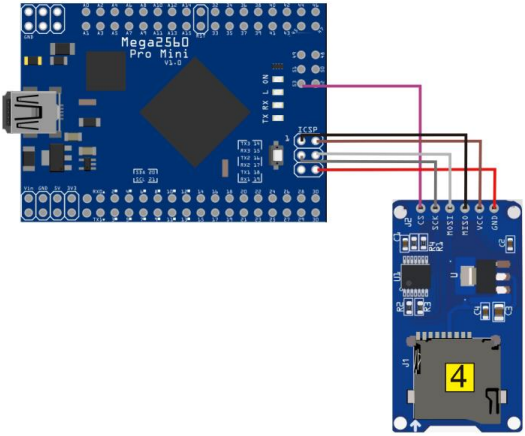
4. Koneksi pada modul SD-*Card*

Pada modul SD-*Card* memiliki 6 pin, tiap pin tersebut memiliki fungsi yang berbeda. Berikut ini nama dan fungsi ke 6 pin yang terdapat pada modul SD-*Card* dan koneksinya dengan pin di Arduino Mega 2560 Pro:

- Pin VCC : untuk sumber tegangan positif 5V pada modul SD-*Card* disambung ke pin 5V ISCP Arduino Mega 2560 Pro.

- Pin MISO: untuk komunikasi input *Serial Pheriperal Interface*, disambung ke pin MISO ISCP Arduino Mega 2560 Pro.
- Pin MOSI: untuk komunikasi output *Serial Pheriperal Interface*, disambung ke pin MOSI ISCP Arduino Mega 2560 Pro.
- Pin SCK: untuk komunikasi jam *Serial Pheriperal Interface*, disambung ke pin MOSI ISCP Arduino Mega 2560 Pro.
- Pin CS: untuk transfer data dari Arduino ke SD-*Card*, disambung ke pin 53 Arduino Mega 2560 Pro.
- Pin GND : untuk sumber tegangan negatif modul disambung ke pin GND ISCP Arduino Mega 2560 Pro.

Berikut ini gambaran koneksi ke 6 pin modul SD-*Card* ke Arduino Mega 2560 Pro.



Gambar 4.6 *Circuit* Modul SD-*Card*

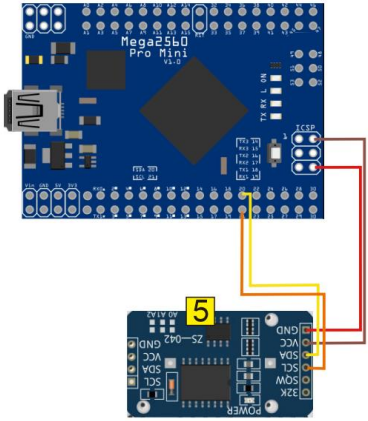
5. Koneksi pada modul jam RTC DS3231

Pada modul RTC DS3231 memiliki 6 pin dan hanya digunakan 4 pin karena 2 pin yang tidak digunakan hanya berfungsi untuk mengatur kompensasi temperatur dan alaram,

tiap pin tersebut memiliki fungsi yang berbeda. Berikut ini nama dan fungsi ke 4 pin yang terdapat pada modul SD-*Card* dan koneksinya dengan pin di Arduino Mega 2560 Pro:

- Pin VCC : untuk sumber tegangan positif 5V pada modul SD-*Card* disambung ke pin 5V ISCP Arduino Mega 2560 Pro.
- Pin SCL: untuk komunikasi serial jam, disambung ke pin 21 Arduino Mega 2560 Pro.
- Pin SDA: untuk komunikasi serial data, disambung ke pin 20 Arduino Mega 2560 Pro.
- Pin GND : untuk sumber tegangan negatif modul disambung ke pin GND ISCP Arduino Mega 2560 Pro.

Berikut ini gambaran koneksi ke 4 pin modul RTC DS3231 ke Arduino Mega 2560 Pro.



Gambar 4.7 *Circuit* Modul RTC DS3231

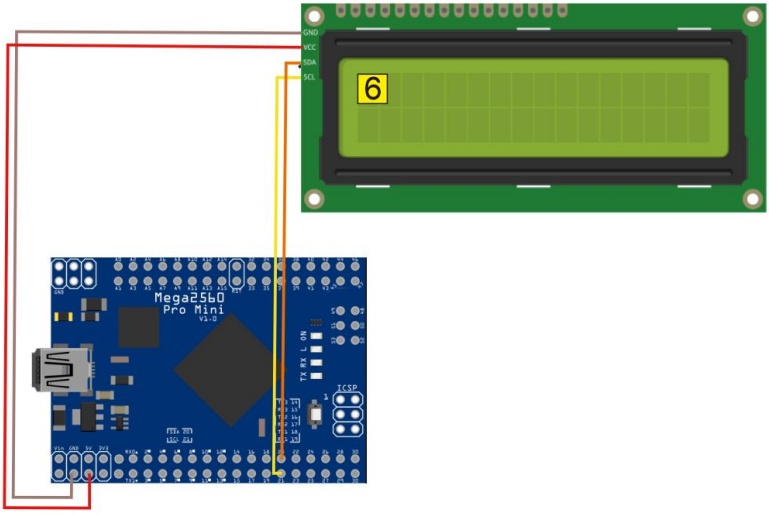
6. Koneksi pada LCD 16×2

Pada LCD 16×2 memiliki 4 pin, tiap pin tersebut memiliki fungsi yang berbeda. Berikut ini nama dan fungsi ke 4 pin yang

terdapat pada LCD 16×2 dan koneksinya dengan pin di Arduino Mega 2560 Pro:

- Pin VCC : untuk sumber tegangan positif 5V pada modul SD-*Card* disambung ke pin 5V Arduino Mega 2560 Pro.
- Pin SCL: untuk komunikasi serial jam, disambung ke pin 21 Arduino Mega 2560 Pro.
- Pin SDA: untuk komunikasi serial data, disambung ke pin 20 Arduino Mega 2560 Pro.
- Pin GND : untuk sumber tegangan negatif modul disambung ke pin GND Arduino Mega 2560 Pro.

Berikut ini gambaran koneksi ke 4 pin LCD 16×2 ke Arduino Mega 2560 Pro.

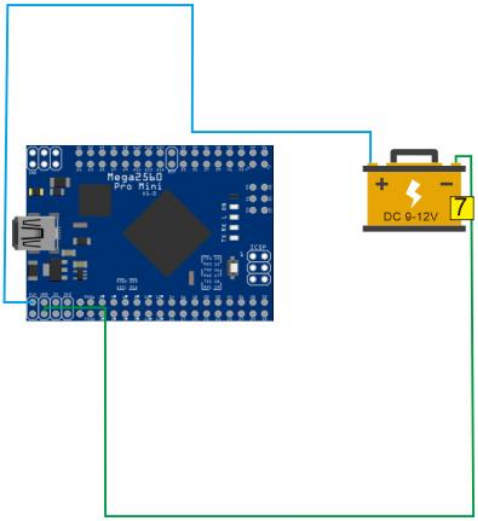


Gambar 4.8 *Circuit* LCD 16×2

7. Koneksi sumber daya

Arduino Mega 2560 Pro tidak menyediakan soket adaptor AC-DC untuk input daya 9-12V DC, tapi disediakan pin Vin untuk menerima daya positif dan GND untuk daya negatif, dari kedua pin tersebut, kemudian dapat dipasang soket adaptor AC-

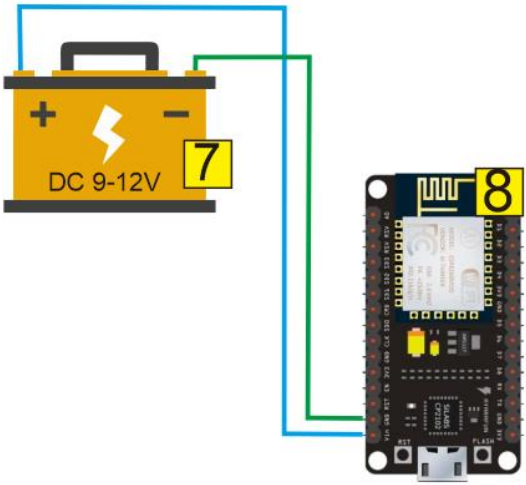
DC menggunakan kabel *jumper*. Berikut ini koneksi sumber daya pada Arduino Mega 2560 Pro:



Gambar 4.9 *Circuit* Sumber Daya

8. Koneksi sumber daya ke NodeMCU

   NodeMCU tidak menyediakan soket adaptor AC-DC untuk input daya 9-12V DC, tapi disediakan pin Vin untuk menerima daya positif dan GND untuk daya negatif, dari kedua pin tersebut, kemudian dapat dipasang soket adaptor AC-DC menggunakan kabel *jumper*. Berikut ini koneksi sumber daya pada NodeMCU ESP8266 dapat dilihat pada Gambar 4.10.



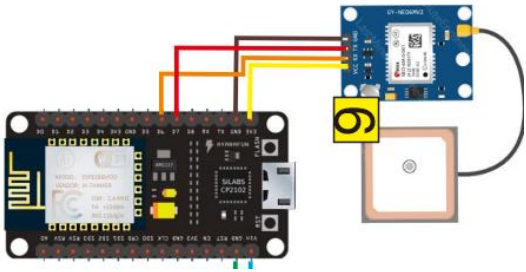
Gambar 4.10 *Circuit* Sumber Daya NodeMCU

9. Koneksi pada Modul GPS NEO-6MV2

Pada modul GPS NEO-6MV2 memiliki 4 pin, tiap pin tersebut memiliki fungsi yang berbeda. Berikut ini nama dan fungsi ke 4 pin yang terdapat pada Modul GPS NEO-6MV2 dan koneksinya dengan pin di NodeMCU:

- Pin VCC : untuk sumber tegangan positif 3V pada modul GPS disambung ke pin 3V NodeMCU.
- Pin RX: untuk menerima data, disambung ke pin D6 NodeMCU.
- Pin TX: untuk transfer data, disambung ke pin D7 NodeMCU.
- Pin GND : untuk sumber tegangan negatif disambung ke pin GND NodeMCU.

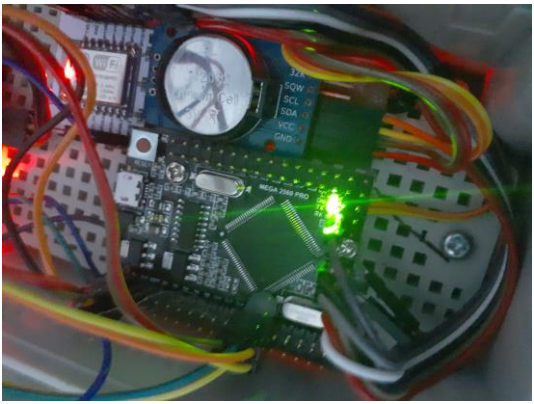
Berikut ini gambaran koneksi ke 4 pin modul GPS NEO-6MV2 ke NodeMCU dapat dilihat pada Gambar 4.11.



Gambar 4.11 *Circuit* ModulGPS NEO-6MV2

4.1.2 Perangkaian Komponen Elektronik

Setelah desain *circuit* dari instrumen sudah ditetapkan, kemudian akan dilakukan pemasangan tiap komponen elektronik sesuai dengan desain *circuit*. Pemasangan tiap komponen ke Arduino tidak perlu memerlukan proses penyolderan, hanya cukup menggunakan kabel *jumper*, baik itu *male-male, female-female, male-female.* Berikut ini hasil dari perangkaian komponen elektronik dapat dilihat pada Gambar 4.12.



Gambar 4.12 Perangkaian Komponen Elektronik

4.1.3 Pembuatan Koding Instrumen

Dalam pembuatan koding untuk instrumen *low-cost ultrasonic tide gauge*, digunakan perangkat lunak Arduino IDE yang merupakan perangkat lunak bawaan dari mikrokontroler Arduino untuk pembuatan koding. Berikut ini hasil dari pembuatan koding untuk instrumen *low-cost ultrasonic tide gauge*. Untuk koding lengkapnya bisa dilihat di Lampiran 2.1.

1. Membuat *header file*

   Header File berfungsi untuk memanggil fungsi dari sebuah *library* tiap modul dan sensor yang terkoneksi dengan Arduino.

```
#include <Wire.h>
#include "RTClib.h"
#include <SPI.h>
#include <SD.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include<Timer.h>
#include <DHT.h>
#include <DHT_U.h>
```

2. Koding modul WiFi ESP8266

   Koding untuk ESP8266 berisi perintah untuk melakukan koneksi dengan WiFi yang diketahui nama dan kata sandinya. Untuk melakukan transfer data website Thingspeak diperlukan sebuah server *channel* dan kode API, langkah pembuaan *channel* Thingspeak dapat dilihat pada Lampiran 2.4, waktu minmum untuk *delay* pengiriminan data menggunakan modul ini adalah 15 detik atau 1500 mikro detik.

```
char *api_key="T10BIAETKHK8J3OP";    // kode API dari channel Thingspeakmu
connect_wifi("AT",1000);
connect_wifi("AT+CWMODE=1",1000);
connect_wifi("AT+CWQAP",1000);
connect_wifi("AT+RST",5000);
connect_wifi("AT+CWJAP=\"Nama WiFi\",\"Pasword\"",10000);
Serial.println("Wifi Connected");
t.every(1500, send2server); // delay pengiriman data ke Thingspeask dalam
microsecond

sprintf(postUrl, "update?api_key=%s&field1=%d",api_key,jarak);
  httpGet("api.thingspeak.com", postUrl, 80);
connect_wifi(atHttpGetCmd,1000);
```

3. Koding sensor suhu DHT22

   Berisi perintah untuk mendefinisikan Pin *Out* dan penyimpanan data suhu.

```
#define DHTPIN 6
#define DHTTYPE DHT22
DHT dht(DHTPIN, DHTTYPE);

float temperature = dht.readTemperature();
  Serial.print("Temperature: ");
  Serial.print(temperature, 1);
  Serial.println("*C");
```

4. Koding sensor ultrasonik JSN-SR04T

   Berisi perintah untuk mendefinisikan Pin *Out*, pengukuran durasi pancaran dan pantulan gelombang suara, perhitungan koreksi kecepatan suara dengan data ukuran dan data sensor suhu, serta menghitung jarak terukur.

```
#define trigPin 17
#define echoPin 16
pinMode(trigPin, OUTPUT);
pinMode(echoPin, INPUT);

digitalWrite(trigPin, LOW);
  delayMicroseconds(5);
digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
digitalWrite(trigPin, LOW);
duration = pulseIn(echoPin, HIGH); //microsecond

float speedOfSound = 331.3 + 0.606 * temperature;   //satuan m/s
  Serial.print("speed of sound: ");
  Serial.print(speedOfSound);
  Serial.println(" m/s");
int jarak = offset_alat - (((duration * speedOfSound)/20000)*10); //
satuan mm
Serial.print(jarak);
```

5. Koding LCD

   Berisi perintah untuk mendefinisikan Pin LCD, menampilkan informasi waktu, dan jarak terukut.

```
LiquidCrystal_I2C lcd(0x27, 2, 1, 0, 4, 5, 6, 7, 3, POSITIVE);
lcd.begin(16, 2)
  lcd.setCursor(0,0);
  printAngka(now.day());
  lcd.print("-");
  printAngka(now.month());
  lcd.print("-");
  printAngka(now.year());

  lcd.print(" ");
  printAngka(now.hour());
  lcd.print(":");
  printAngka(now.minute());

  lcd.setCursor(0,1);
  lcd.print("TIDES = ");
  lcd.print(jarak);
  lcd.print(" mm");
```

6. Koding modul jam RTC DS3231

   Berisi perintah untuk inisiasi modul RTC, inisiasi penggunaan baterai pada modul RTC, pengambilan informasi waktu.

```
RTC_DS3231 rtc;
  if (! rtc.begin()) {
    Serial.println("Couldn't find RTC");
  }
  if (rtc.lostPower()) {
    Serial.println("RTC lost power, lets set the time!");
    rtc.adjust(DateTime(F(__DATE__), F(__TIME__)));

DateTime now = rtc.now();
```

7. Koding modul SD-*Card*

   Berisi perintah untuk pendifinisian Pin, inisiasi modul SD-*Card*, penyimpanan data waktu dan jarak.

```
const int chipSelect = 53;
File myFile;
if (!SD.begin(chipSelect)) {
   Serial.println("kartu rusak, atau tidak ada");

  myFile = SD.open("log.csv", FILE_WRITE);
  if (myFile) {
    myFile.print(now.day());
    myFile.print("/");
    myFile.print(now.month());
    myFile.print("/");
    myFile.print(now.year());
    myFile.print(",");
    myFile.print(now.hour());
    myFile.print(":");
    myFile.print(now.minute());
    myFile.print(":");
    myFile.print(now.second());
    myFile.print(",");
    myFile.println(jarak);
    myFile.close(); // close the file
  }
```

8. Koding NodeMCU dan modul GPS

   NodeMCU diprogram terpisah dari Arduino, koding lengkapnya dapat dilihat pada Lampiran 2.2. koding ini berisi perintah untuk pendefinisian *header file*, pendifinisian Pin, koneksi ke WiFI, inisiasi modul GPS, pengiriman data koordinat ke server thingspeak.

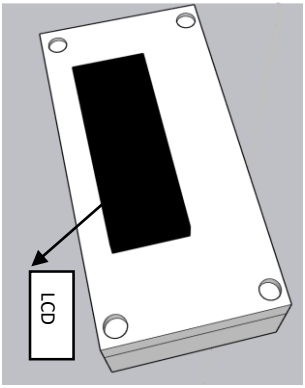
```
#include <SoftwareSerial.h>
#include <TinyGPS++.h>
#include "ThingSpeak.h"
#include <ESP8266WiFi.h>
static const int RXPin = 12, TXPin = 13;
static const uint32_t GPSBaud = 9600;

// nama wifi dan password
  const char* ssid     = "Nama Wifi";
  const char* password = "Password";
//input nomor channel dank ode API
   unsigned long myChannelNumber = Nomer Channel Thingspeak;
   const char * myWriteAPIKey = "kode API";
// TinyGPS++ object
   TinyGPSPlus gps;
    WiFiClient  client;
// mengambil data koordinat(F("Location: "));
    double latitude = (gps.location.lat());
    double longitude = (gps.location.lng());
    String latbuf;
    latbuf += (String(latitude, 6));
    Serial.println(latbuf);
    String lonbuf;
    lonbuf += (String(longitude, 6));
    Serial.println(lonbuf);
//mengirim data ke thingspeak
   ThingSpeak.setField(2, latbuf);
    ThingSpeak.setField(3, lonbuf);
    ThingSpeak.writeFields(myChannelNumber, myWriteAPIKey);
    delay(300000); //delay milisecond
```

4.1.4 Perancangan Bodi Instrumen

Bodi pada *low-cost ultrasonic tide gauge* ini akan didesain *portable* dan *compact*. Terdapat dua komponen utama dalam bodi *ultrasonic tide gauge* ini, yaitu.

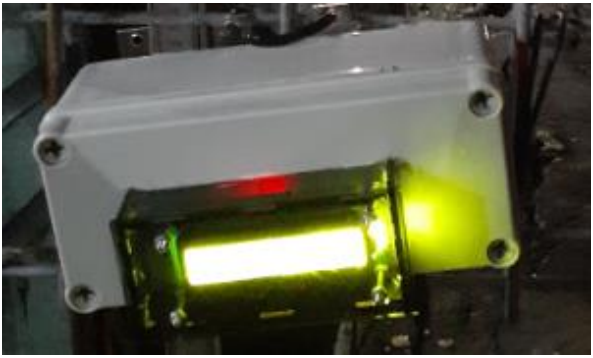
1. Kotak Komponen Elektronik
   Berfungsi untuk tempat komponen elektronik berada, kotak ini menggunakan kotak *junction*, yang merupakan kotak tahan air

atau *waterproof*. Dimensi panjang, lebar, tinggi kotak ini secara berturut-turut adalah 18cm, 9cm, 9cm. Kotak ini berbahan PVC. Pada kotak ini, penulis memasang LCD dibagian tutup kotak, dan diberi pelindung LCD yang terbuat dari akrilik warna hitam tembus pandang tahan air yang memiliki dimensi panjang 10cm, lebar 5cm, dan tinggi 2cm. Berikut ini merupakan desain bodi kotak *junction* tempat komponen elektronik.



Gambar 4.13 Desain Kotak Komponen Elektronik

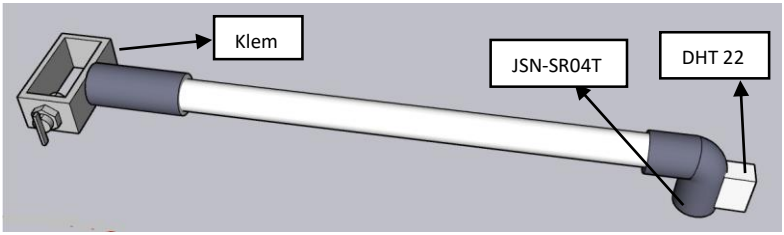
Dan berikut ini hasil dari pembuatan kotak *junction* .



Gambar 4.14 Hasil Pembuatan Kotak Komponen Elektronik

2. Lengan Sensor

Berfungsi untuk menghubungkan kabel yang terhubung ke Arduino dari kotak komponen elektronik ke Pin yang ada di sensor ultrasonik JSN-SR04T dan sensor suhu DHT22. Komponen lengan sensor ini memiliki 2 bagian, yang pertama adalah klem yang terbuat dari aluminium untuk memasang lengan sensor pada palem pasut atau kayu dengan ukuran penampang 8cm×3cm, yang kedua batang sensor yang terbuat dari bahan pipa PVC sepanjang 60cm dan sambungan pipa. Berikut ini adalah desain dari lengan sensor.



Gambar 4.15 Desain Lengan Sensor

Dan berikut ini hasil dari pembuatan lengan sensor.

Gambar 4.16 Hasil Pembuatan Lengan Sensor

4.1.5 Total Biaya Pembuatan Instrumen

Pada penelitian Tugas Akhir ini, komponen yang digunakan dalam perancangan instrumen *low-cost ultrasonic tide gauge* dengan sistem *IoT* benar-benar dipilih yang memiliki harga minimum, tetapi dengan kualitas yang juga bagus. Berikut ini rincian biaya yang dikeluarkan dalam pembuatan *low-cost ultrasonic tide gauge* ini:

1. Peralatan dan Jasa

Tabel 4.1 Biaya Peralatan dan Jasa

Jenis Peralatan	**Volume**	**Harga Satuan (Rp)**	**Nilai (Rp)**
Gergaji Pipa	2 buah	Rp. 30.000	Rp. 60.000
Solder	1 buah	Rp. 50.000	Rp. 50.000
Palem Pasut	1 buah	Rp. 500.000	Rp. 500.000
Klem Besi	2 buah	Rp. 20.000	Rp. 40.000
Akrilik	10 cm^2	Rp. 2.200	Rp. 22.000
Jasa	2 hari	Rp. 100.000	Rp. 200.000
SUB TOTAL (Rp)			Rp. 872.000

2. Bahan Habis Pakai

Tabel 4.2 Biaya Bahan Habis Pakai

Bahan Habis Pakai	**Volume**	**Harga Satuan (Rp)**	**Nilai (Rp)**
Pipa PVC ¾	2 meter	Rp. 10.000	Rp. 20.000
Isi lem tembak	5 buah	Rp. 2.000	Rp. 10.000
Lem G	5 buah	Rp. 6.000	Rp. 30.000
Sensor Ultrasonik JSN SR-04T	1 buah	Rp. 150.000	Rp. 150.000
Arduino Mega 2560 Pro	2 buah	Rp. 300.000	Rp. 300.000

Bahan Habis Pakai	**Volume**	**Harga Satuan (Rp)**	**Nilai (Rp)**
Modul LCD 1602	1 buah	Rp. 50.000	Rp. 50.000
Modul RTC DS3231	1 buah	Rp. 50.000	Rp. 50.000
Modul SD-Card	1 buah	Rp. 50.000	Rp. 50.000
Modul Wifi ESP-8266	1 buah	Rp.60.000	Rp. 60.000
Sensor suhu DHT22	1 buah	Rp. 50.000	Rp. 50.000
Adaptor DC 9V	1 buah	Rp. 100.000	Rp. 100.000
NodeMCU ESP8266	1 buah	Rp. 90.000	Rp. 90.000
Modul GPS NEO-6MV2	1 buah	Rp. 68.000	Rp. 68.000
Kabel konektor	30 buah	Rp. 3.000	Rp. 90.000
Kotak Junction	1 buah	Rp. 200.000	Rp. 200.000
Timah	2 roll	Rp. 25.000	Rp. 50.000
SUB TOTAL (Rp)			Rp. 1.368.000

3. Total Biaya

Tabel 4.3 Total Biaya

Peralatan	Rp. 872.000
Bahan Habis	Rp. 1.368.000
TOTAL BIAYA	Rp. 2.240.000

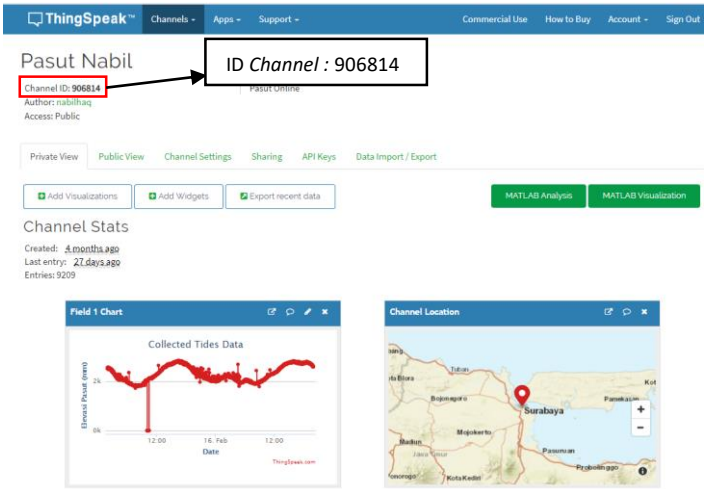
Biaya pembuatan instrumen hanya 7,5% dari harga *automatic tide gauge* termurah yang beredar di Indonesia (Rp. 30.000.000).

4.2 Hasil dan Analisis Hasil Pengujian Instrumen

Pengujian instrument *low-cost ultrasonic tide gauge* dengan sistem *IoT* berbasis Arduino dilakukan selama 3 hari 2 malam atau lebih tepatnya selama 48,5 jam *non-stop*. Hasil pengujian instrument merupakan data pasang surut air laut yang disimpan di dalam server Thingspeak dan SD-*Card*.

4.2.1 Data Pasang Surut dari Server Thingspeak

Pada saat pengambilan data, kondisi pasang surut air laut dapat dipantau secara daring menggunakan website Thingspeak atau perangkat lunak android Thingview. Data yang sudah terekam oleh server Thingspeak dapat diunduh dan dilihat melalui website Thingspeak, atau hanya dapat dilihat melalui perangkat lunak android Thingview. Akan tetapi, jumlah data yang dapat ditampilkan dalam grafik pada website maupun android terbatas pada 8000 data saja. Untuk melihat data dan grafik pasang surut di server Thingspeak, kita cukup memasukkan ID *channel* yang telah dibuat, yaitu **906814** pada kolom pencarian *channel*. Berikut ini tampilan data pasang surut di website Thingspeak dapat dilihat pada Gambar 4.17.



Gambar 4.17 Data Pasang Surut di Website Thingspeak

Tampilan data pasang surut di perangkat lunak Thingview dapat dilihat pada Gambar 4.18.

Gambar 4.18 Data Pasang Surut di Thingview

Data yang didapatkan dari server thingspeak berjumlah 9209 data. Dengan interval 15-19 detik, akan tetapi, instrument *low-cost ultrasonic tide gauge* ini mengandalkan WiFi yang bersumber dari *hotspot* seluler untuk mengirim data ke server Thingspeak, data banyak yang mengalami *delay* penerekaman sampai beberapa menit. Contoh *delay* perekaman data tersebut dapat dilihat pada Gambar 4.19.

Gambar 4.19 *Delay* Perekaman Data

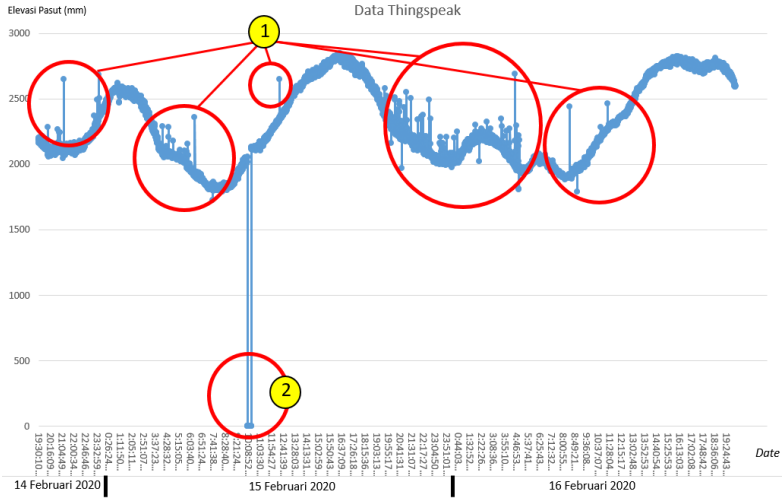
Karena sistem penyimpanan data pada instrument ini tidak hanya melalui server Thingspeak, melainkan juga menggunakan SD-*Card*, maka semua data yang tidak terekam karena *delay* sudah terekam ke dalam SD-*Card*.

Berikut ini sampel data pasang surut keseluruhan dari website Thingspeak ditunjukkan pada Tabel 4.4. Untuk data lengkapnya tercantum pada bagian Lampiran 1.1.

Tabel 4.4 Data Pasut Thingspeak

Tanggal	**Jam**	**Elevasi (mm)**
2/14/2020	19:30:10	2160
2/14/2020	19:30:28	2194
2/14/2020	19:30:45	2201
2/14/2020	19:31:03	2180
2/14/2020	19:31:20	2163
2/14/2020	19:31:38	2166
……	……	……
2/16/2020	20:01:07	2610
2/16/2020	20:01:24	2601
2/16/2020	20:01:42	2591
2/16/2020	20:02:00	2595
2/16/2020	20:02:18	2602
2/16/2020	20:02:35	2593
2/16/2020	20:02:53	2599

Berikut ini visualisasi grafik pasang surut data Thingspeak dapat dilihat pada Gambar 4.20.



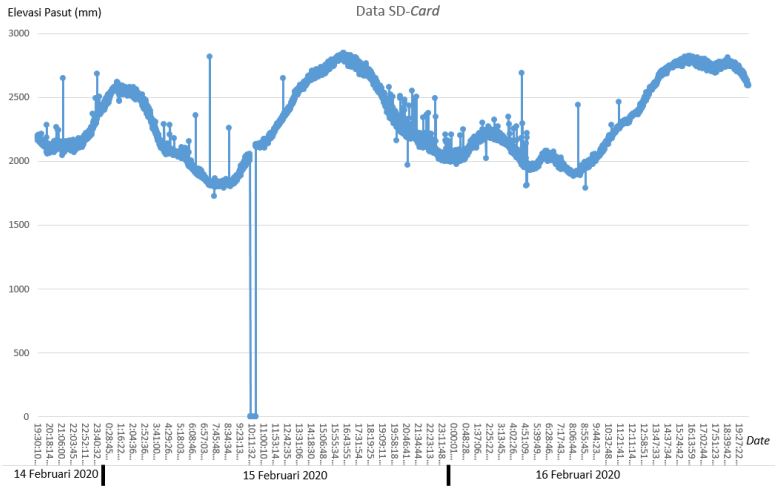
Gambar 4.20 Grafik Pasut Data Thingspeak

Dari grafik pada Gambar 4.20 di atas, terdapat beberapa *noise* yang terjadi saat pengukuran. Noise tersebut terjadi karena beberapa hal pada saat pengujian alat, yaitu:

1. Sampah yang menghalangi gelombang ultrasonik ke permukaan air, sehingga mengakibatkan *noise* yang ditunjukkan pada nomor 1 di Gambar 4.20.
2. Karena sensor ultrasonik memiliki sistem PING yang bekerja berdasarkan prinsip pemantulan gelombang suara, terkadang pantulan gelombangnya menjadi tidak periodik.
3. Karena penempatan sensor suhu hanya pada tranduser sensor ultrasonik berada, data suhu pada area yang dilalui gelombang suara tidak terekam, jadi jika terjadi perbedaan suhu antara area yang dilalui gelombang dengan area tranduser sensor, maka nilai elevasi pasut yang dihasilkan akan berbeda.
4. *Noise* yang mencapai nilai elevasi 0 (nol) yang ditunjukkan pada nomor 2 di Gambar 4.20 disebabkan karena konektor kabel untuk sensor suhu terputus karena suhu di dermaga yang panas, sehingga bacaan elevasinya menjadi nol.

4.2.2 Data Pasang Surut dari SD-*Card*

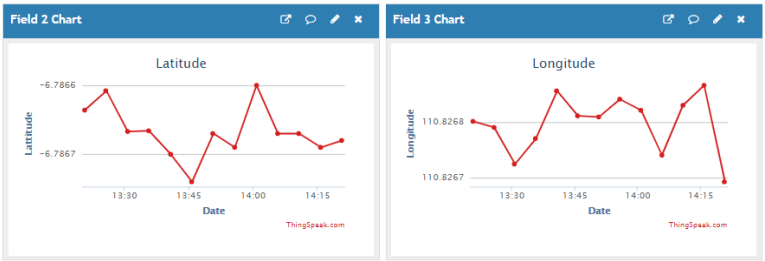
Sistem penyimpanan data pasang surut pada instrument *low-cost ultrasonic tide gauge* selain melalui server Thingspeak adalah dengan SD-*Card*. Dengan menggunakan SD-*Card*, data yang tidak terkirim ke server Thingspeak tetap akan terekam di dalam SD-*Card*. Data yang disimpan di dalam SD-*Card* memiliki format .CSV yang dapat dibuka melalui Microsoft Excel. Jumlah data yang terekam pada SD-*Card* adalah 9621, selisih 412 data dari server Thingspeak. Data pasang surut SD-*Card* dapat dilihat pada Lampiran 1.2. Grafik pasang surut data SD-*Card* dapat dilihat pada Gambar 4.21.



Gambar 4.21 Grafik Pasut Data SD-*Card*

4.2.3 Data Koordinat dari Thingspeak

Hasil pengujian modul GPS dapat didapatkan 13 koordinat lintang dan bujur yang dapat dipantau secara daring melalui website Thingspeak, server penyimpanan data koordinat dan data pasut sama, yaitu pada *channel* 906814. Berikut ini data koordinat dari modul GPS yang ditampilkan di Thingspeak dapat dilihat pada Gambar 4.22.



Gambar 4.22 Data Koordinat pada Thingspeak

Data koordinat yang direkam pada server Thingspeak juga dapat dilihat melalui perangkat lunak android ThingView. Tampilan data koordinat pada ThingView dapat dilihat pada Gambar 4.23.

Gambar 4.23 Data Koordinat pada ThingView

4.2.4 Kontrol Kualitas Data Pasang Surut

Kontrol kualitas data bertujuan untuk mendeteksi data *outliers* dari data pasut yang telah lengkap. Kontrol kualitas data dilakukan

dengan menggunakan kepercayaan 99,7% atau tiga sigma (3σ). Hasil dari perhitungan kontrol kualitas data pasut ditunjukkan pada Tabel 4.5.

Tabel 4.5 Hasil Kontrol Kualitas Data

Data	**Data Diterima**	**Data Ditolak**	**Diterima (%)**	**Ditolak (%)**
24 jam pertama	4714	75	98,434	1,566
24 jam kedua	4832	0	100	0

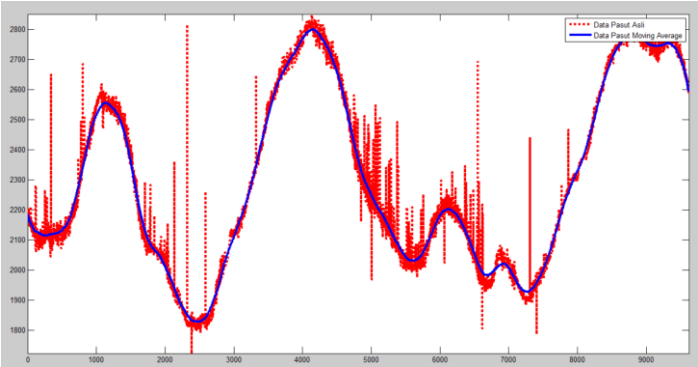
Pada Tabel 4.6, ditunjukkan bahwa mayoritas data dari instrumen *ultrasonic tide gauge* diterima dari perhitungan kontrol kualitas data. Terdapat 75 data yang ditolak pada sampel data 24 jam pertama merupakan *outliers* yang memiliki nilai nol (0) yang disebabkan karena terputusnya kabel konektor pada sensor saat pengujian alat.

4.2.5 Penyaringan Data Pasang Surut

Penyaringan atau *filtering* bertujuan untuk menghilangkan *noise* atau derau yang terdapat pada data pasang surut hasil pengujian instrument *low-cost ultrasonic tide gauge*. Penyaringan data dilakukan menggunakan dua metode, yaitu metode *moving average* dan *moving median*. Proses penyaringan data pasut ini dilakukan dengan menggunakan MATLAB, untuk kodingnya dapat dilihat pada Lampiran 2.2.

1. *Moving Average*

   Berikut ini grafik hasil dari penyaringan data metode *moving average* dengan rentang data 300 dapat dilihat pada Gambar 4.24.

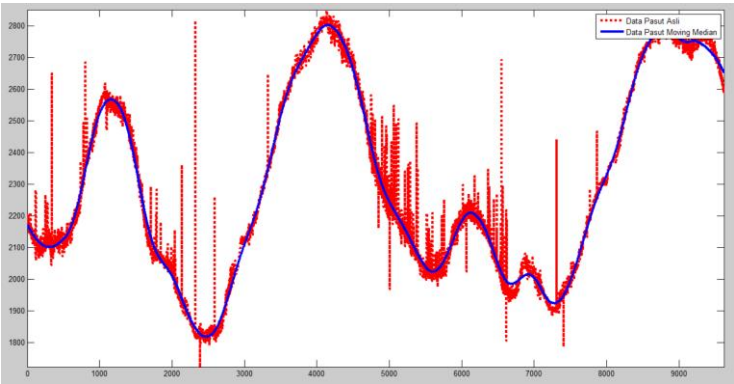


Gambar 4.24 Grafik Pasut *Moving Average*

Data pasang surut yang disaring menggunakan metode *moving average* dapat dilihat pada Lampiran 1.3.

2. *Moving Median*

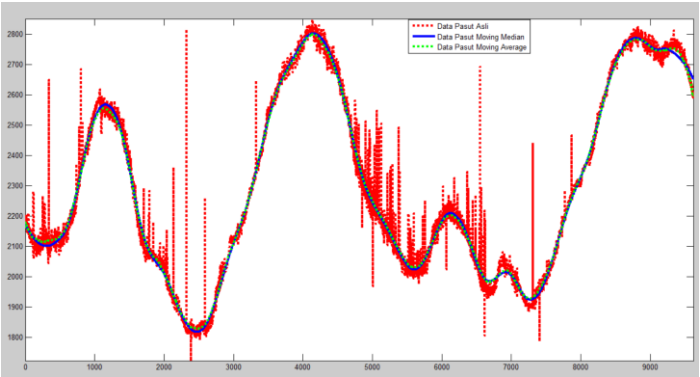
Berikut ini grafik hasil dari penyaringan data metode *moving median* dengan rentang data 100, dapat dilihat pada Gambar 4.25.



Gambar 4.25 Grafik Pasut *Moving Median*

Data pasang surut yang disaring menggunakan metode *moving median* dapat dilihat pada Lampiran 1.4.

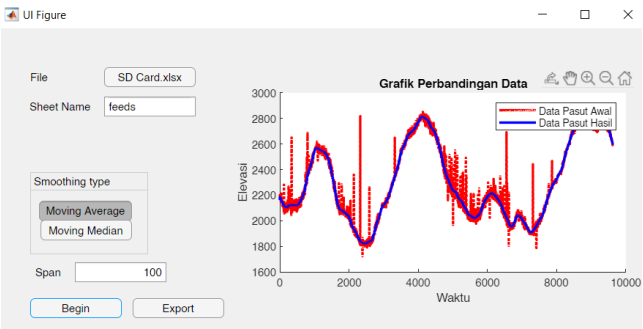
Berikut ini *overlay* data asli, dan data disaring metode *moving average* dan *moving median* dapat dilihat pada Gambar 4.26.



Gambar 4.26 *Overlay* Grafik Pasut Semua Metode Penyaringan

4.2.6 Perangkat Lunak Antar-Muka untuk *Data Filtering*

Untuk mempermudah melakukan *filtering* pada data pasut dari instrumen *low-cost ultrasonic tide gauge* ini, dibuatlah sebuah perangkat lunak antar muka khusus *filtering* yang bernama FilTide. Metode yang ditawarkan pada perangkat lunak ini adalah metode *moving median* dan *moving average*. Berikut ini tampilan antar-muka dari FilTide dapat dilihat pada Gambar 4.27.



Gambar 4.27 Tampilan Perangkat Lunak FilTide

4.2.7 Perbandingan Bacaan Instrumen dengan Bacaan Manual

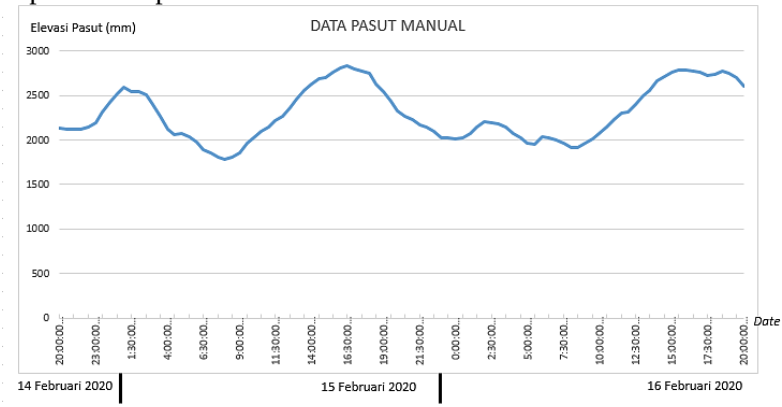
Untuk mengetahui tingkat ketelitian dan kesalahan dari instrument *low-cost ultrasonic tide gauge*. Data dari instrument akan dilakukan perhitungan kesalahan relatif untuk mengetahui akurai alat dan perhitungan RMSE untuk mengetahui besar penyimpangan bacaan alat. Kedua perhitungan tersebut akan dilakukan menggunakan data alat, data *moving average*, dan data *moving median* sebagai data observasi, dan data pengamatan pasang surut manual dengan palem pasut sebagai data yang dianggap benar dan aktual.

Data pengamatan manual menggunakan palem pasut memiliki interval 30 menit dan pasang surut dibaca sebanyak 3 kali pada saat kondisi gelombang tertinggi, tengah dan terendah. Akan tetapi pada tanggal 14 Februari 2020, pukul 20.00-21.00 pengukuran hanya dilakukan dengan interval 1 jam, hal tersebut disebabkan karena kesalahan pengamat. Berikut ini data pasang surut manual dapat dilihat pada Tabel 4.6. Data lengkapnya dapat dilihat pada Lampiran 1.5.

Tabel 4.6 Data Pengamatan Manual

Hari	**Jam**	**Menit**	**Elevasi (mm)**	**rata-rata(mm)**
Jumat, 14-2-2020	20	0	2150	2133,333
Jumat, 14-2-2020			2130	
Jumat, 14-2-2020			2120	
……	……	……	……	……
Minggu, 16-2-2020	19	30	2710	2700
Minggu, 16-2-2020			2700	
Minggu, 16-2-2020			2690	
Minggu, 16-2-2020	20	0	2610	2605
Minggu, 16-2-2020			2605	
Minggu, 16-2-2020			2600	

Berikut ini grafik data pasang surut air laut bacaan manual dapat dilihat pada Gambar 4.28.



Gambar 4.28 Grafik Data Pasut Manual

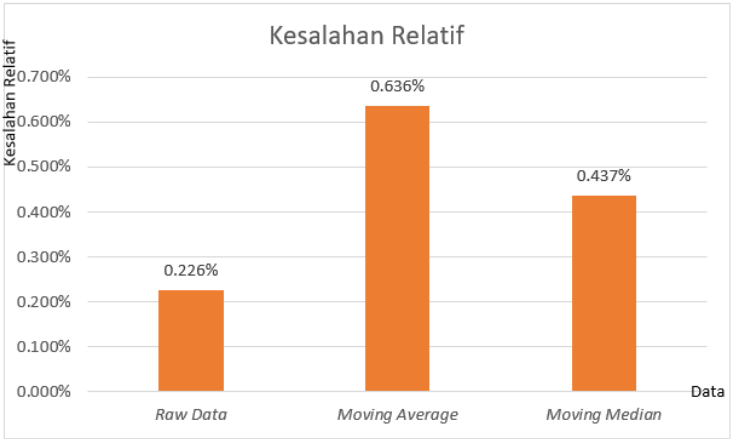
1. Perhitungan Kesalahan Relatif dan Akurasi

Berikut ini hasil dari perhitungan kesalahan relatif dan akurasi dengan data pasut instrumen, data *moving average*, dan data *moving median* dapat dilihat pada Tabel 4.7.

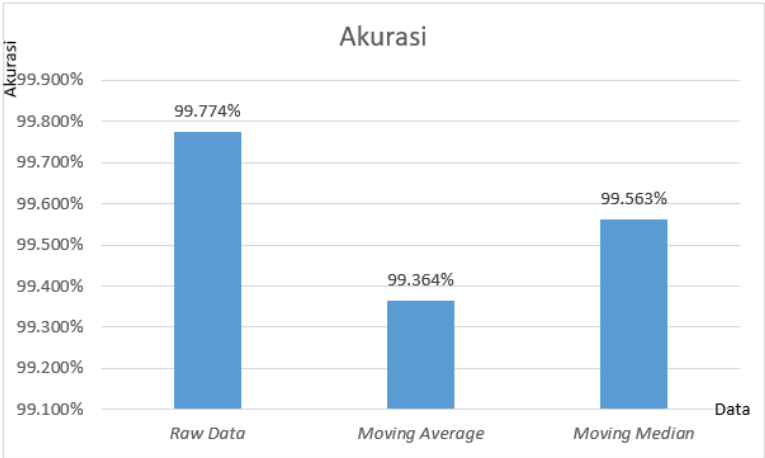
Tabel 4.7 Hasil Perhitungan Kesalahan Relatif dan Akurasi

Data	**Kesalahan Relatif (%)**	**Akurasi (%)**
Instrumen	0,226	99,774
Moving Average	0,636	99,364
Moving Median	0,437	99,563

Nilai kesalahan relatif dan akurasi pada Tabel 4.7, secara grafik ditunjukkan pada Gambar 4.29 dan Gambar 4.30.



Gambar 4.29 Grafik Kesalahan Relatif



Gambar 4.30 Grafik Akurasi

Data dengan akurasi terbaik adalah data mentah instrumen, akan tetapi data tersebut masih memiliki banyak *outliers* atau *noise*, dan data hasil *filtering* paling bagus adalah *moving median.*

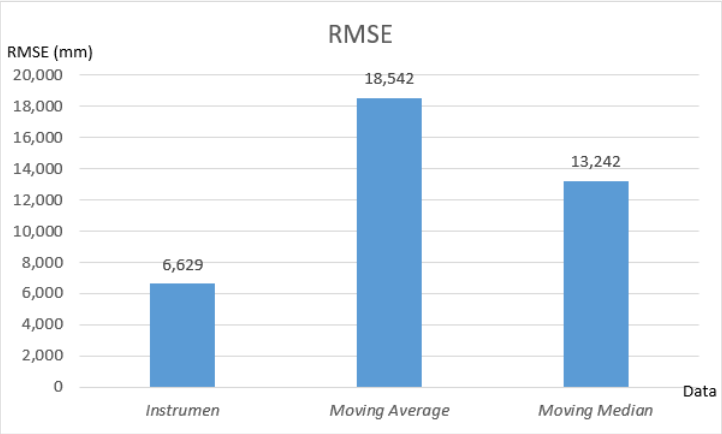
2. Perhitungan RMSE

Berikut ini hasil dari perhitungan RMSE dengan data pasut instrumen, data *moving average*, dan data *moving median* dapat dilihat pada Tabel 4.8.

Tabel 4.8 Hasil Perhitungan RMSE

Data	**RMSE (mm)**
Instrumen	6,629
Moving Average	18,542
Moving Median	13,242

Nilai RMSE pada Tabel 4.8, secara grafik ditunjukkan pada Gambar 4.31.



Gambar 4.31 Grafik RMSE

Data dengan RMSE terendah adalah data mentah instrumen, akan tetapi data tersebut masih memiliki banyak *outliers* atau *noise*, dan data hasil *filtering* dengan RMSE terendah adalah *moving median.*

4.2.8 Analisa Hasil Penyaringan Data

Berdasarkan hasil penyaringan atau *filtering* data yang telah dilakukan, didapatkan bahwa untuk penyaringan metode *moving average* rentang yang diperlukan adalah 300 data, sedangkan pada *moving median* rentangnya hanya 100 data, dan dari hasil perhitungan kesalahan relatif dan RMSE pada kedua metode *filtering* yang dapat dilihat pada Tabel 4.9, didapatkan bahwa metode *filtering moving median* lebih bagus dari pada *moving average*.

Tabel 4.9 Perbandingan Hasil Penyaringan Data

Data	**Kesalahan Relatif (%)**	**Akurasi (%)**	**RMSE (mm)**
Moving Average	0,636	99,364	18,542
Moving Median	0,437	99,563	13,242

Pada metode *moving median* juga hanya membutuhkan rentang 100 data yang diambil nilai tengahnya untuk menyaring *noise* atau derau sehingga grafik pasang surut bisa mendekati hasil bacaan manual. Sedangkan pada metode *moving average* diperlukan rentang data 300 yang dirata-rata untuk menyaring derau sehingga grafik pasang surut bisa mendekati hasil bacaan manual. Bisa disimpulkan hasil *filtering* data dengan metode *moving median* lebih baik dari metode *moving average.*

"Halaman ini sengaja dikosongkan"

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil penelitian Tugas Akhir dengan judul Rancang Bangun Instrumen *Low-Cost Ultrasonic Tide Gauge* dengan Sistem *Internet of Things* (*IoT*) Berbasis Arduino, dapat diambil kesimpulan sebagai berikut:

1. Instrumen *low-cost ultrasonic tide gauge* yang dihasilkan memiliki biaya perancangan sebesar **Rp.2.240.000**. Biaya tersebut jauh lebih murah, yaitu hanya 7,5% dari harga *tide gauge* termurah yang ada di pasaran, yaitu Rp. 30.000.000.
2. Dari hasil Dari hasil uji kualitas instrumen low-cost ultrasonic tide gauge dengan menggunakan perhitungan nilai kesalahan relatif dan RMSE dengan data pengamatan pasut manual sebagai acuan data yang benar, didapatkan hasil **0,226%** dan **6,629**mm untuk data mentah, **0,636**% dan **18,542**mm untuk data moving average, serta **0,437**% dan **13,242**mm untuk data moving median. Dari hasil tersebut, instrumen low-cost ultrasonic tide gauge memiliki tingkat akurasi yang bagus. Berdasarkan dari hasil perhitungan kesalahan relatif dan RMSE tersebut, didapatkan bahwa data baik data asli dari instrumen *low-cost ultrasonic tide gauge* maupun data dari hasil *filtering* metode *moving average* dan *moving median* memiliki kesalahan relatif dan RMSE yang lebih kecil dari nilai yang ditetapkan pada penelitian Tugas Akhir ini, yaitu 0,8%.
3. Instrumen *low-cost ultrasonic tide gauge* yang dihasilkan masih memiliki masalah pada *noise* atau derau yang disebabkan oleh sampah, kualitas sensor, dan kesalahan teknis kabel konektor yang putus, akan tetapi hal tersebut dapat diatasi dengan penyaringan data. Selain derau, terdapat

masalah *delay* pengiriman data ke server Thingspeak karena koneksi internet tidak stabil, hal tersebut sudah teratasi dengan data yang disimpan pada SD-*card*.

5.2 Saran

Dari hasil dan kesimpulan yang telah diberikan, saran untuk penelitian Tugas Akhir ini dan penelitian selanjutnya adalah:

1. Agar tidak adal kesalahan teknis seperti kabel konektor terputus karena suhu yang panas di lapangan, instrumen dapat diletakkan pada sebuah *panel box* atau hanya sekedar diletakkan di tempat yang teduh setiap waktu.
2. Untuk menghilangkan *delay* perekaman data pada server Thingspeak, dapat dilakukan dengan pemasangan *router* WiFi yang memiliki kecepatan tinggi.
3. Jika pada area survey tidak terdapat wifi, modul WiFi yang ada pada instrumen dapat diganti dengan modul GPRS 4G, sehingga kita tinggal menambahkan kartu perdana yang memiliki sinyal kuat di area survey tersebut dan pastinya memiliki kuota internet.

DAFTAR PUSTAKA

Adityayuda, Anugrah. 2012. Pengukuran Faktor Koreksi Jarak pada Instrumen MOTIWALI. Bogor: Institut Pertanian Bogor.

Ananda, Naufal, dkk. 2019. Rancang Bangun Instrumen *Tide Gauge Multi-Parameter*. Tangerang Selatan: Prosiding Seminar Nasional Bumi dan Atmosfer.

Arafat. 2019. Sistem Pengendalian Suhu dan Kelembaban Kumbung Jamur Tiram secara Realtime Menggunakan ESP8266. Banjarmasin: Jurnal Fisika FLUX.

Ardiansyah. 2016. Sistem Monitoring Air Layak Konsumsi Berbasis Arduino (Studi Kasus Pdam Patalassang). Makassar: Universitas Islam Negeri Alauddin.

Arduino. 2009. Introsuction Arduino. http://www.arduino.cc. Diakses pada tanggal 22 Oktober 2019 pukul 21.58.

Azhari, dkk. 2014. Pembuatan Prototipe Alat Ukur Ketinggian Air Laut Menggunakan Sensor Inframerah Berbasis Mikrokontroller Atmega328. Pontianak: Universitas Tanjungpura.

Bies, D.A., dkk. 2009. *Engineering Noise Control – Theory and Practice* 4th *Edition*. New York: CRC Press.

Boer, Attila Laszlo, dkk. 2016. *Comparative Study Between Global Positioning System on Remotely Piloted Aircraft System*. Brasov: AFASES.

Chamim, A. N. 2010. Penggunaan Microcontroller Sebagai Pendeteksi Posisi. Yogyakarta: Jurnal Informatika Universitas Ahmad Dahlan.

Chobir, Abdul, dkk. 2017. Sistem Deteksi Elevasi Permukaan Air Sungai dengan Sensor Ultrasonik Berbasis Arduino. Tasikmalaya: Universitas Siliwangi.

Djaja, R. 1987. Pengamatan pasang-surut laut untuk penentuan datum ketinggian. In O. S. R. Ongkosongo dan Suyarso (Ed.), Pasang-surut. Jakarta: Lembaga Ilmu Pengetahuan Indonesia. Pusat Penelitian dan Pengembangan Oseanografi.

Gonzalo, R. Arce. 2005. *Nonlinear Signal Processing*: *A Statistical Approach*. New York: Wiley Interscience.

Haristyana, Ayu. 2012. Prediksi Kenaikan Muka Air Laut di Pesisir Kabupaten Tuban Akibat Perubahan Iklim. Surabaya: Institut Teknologi Sepuluh Nopember.

Hartono, Rudi. 2013. Perancangan Sistem Data Logger Temperatur Baterai Berbasis Arduino Duemilanove. Jember: Universitas Jember.

Hasibuan, G.H. 2009. Analisis Surut Astronomis Terendah di Perairan Sabang, Sibolga, Padang, Cilacap, dan Benoa Mengunakan Superposisi Komponen Harmonik Pasang Surut. Sripsi .Bogor: Institut Pertanian Bogor.

Helfrick, Albert D. 1990. Modern Electronic Instrumentation and Measurement Techniques. Englewood Cliffs: Prentice Hall International.

Hutabarat, Sahala dan M.Evans, Stewart. 2008. Pengantar Oseanografi. Depok: Universitas Indonesia.

IOC. 1994. *Manual on sea level measurement an interpretation vol II - emerging technologies. IOC Manual and Guides*. Paris: UNESCO.

IOC. 2002. *Manual on sea level measurement an interpretation vol III - reappraisals and recommendation as of year 2000*. IOC Manual and Guides. Paris: UNESCO.

IOC. 2006. Manual on sea level measurement an interpretation vol IV - an update to 2006. IOC Manual and Guides. Paris: UNESCO.

Khasanah, Isna Uswatun. 2014. Perhitungan Nilai *Chart Datum* Stasiun Pasang Surut Jepara berdasarkan Periode Pergerakan Bulan. Yogyakarta: Universitas Gadjah Mada.

Khoir, M. Mufidul. 2018. Rancang Bangun Alat Monitoring Pasang Surut Air Laut Berbasis *Internet Of Thing* (*IoT*). Surabaya: Universitas Islam Negeri Sunan Ampel.

Kurniawan, Dedy. 2016 Pengujian Ketelitian Hasil Pengamatan Pasang Surut dengan Sensor Ultrasonik (Studi Kasus: Desa Ujung Alang, Kampung Laut, Cilacap). Surabaya: Institut Teknologi Sepuluh Nopember.

Mahasari, Susanti. 2017. Rancang Bangun *Internet of Things* (*IoT*) *Node* Sensor Cuaca Dengan Menggunakan Website. Palembang: Politeknik Negeri Sriwijaya.

NodeMCU. 2015. NodeMCU LUA *Firmware*. https://www.nodemcu.com. Diakses pada tanggal 14 April 2020 pukul 18.00.

Peterson, James, dkk. 2015. *Fundamental of Physical Geography*. Boston: Cengage Learning.

Pratama, Hadijaya. 2012. Akuisisi Data Kinerja Sensor Ultrasonik Berbasis Sistem Komunikasi Serial Menggunakan Mikrokontroler ATMEGA 32. Bandung: Jurnal UPI.

Pugh, D. T. 1987. *Tides, Surges and Mean Sea-Level*. Cirencester: John Wiley & Sons, Ltd.

Ramdhan, Muhammad. 2011. Komparasi Hasil Pengamatan Pasang Surut di Perairan Pulau Pramuka dan Kabupaten Pati Dengan Prediksi Pasang Surut *Tide Model Driver*. Jakarta Utara: Kementerian Kelautan dan Perikanan.

RoboDyn. 2017. Mega 2560 Pro *User Manual*. http://www.robodyn.com. Diakses pada tanggal 2 Januari 2020 pukul 19.35.

Rochim, Adian Fatchur. 2016. *Handsight : Hand-Mounted Device* untuk Membantu Tunanetra Berbasis Ultrasonik dan Arduino. Semarang: Universitas Diponegoro.

Simanjuntak, Matur. 2012. Rancang bangun teknologi Pemurni air menggunakan Arduino. Medan: Uniersitas Sumatera Utara.

Simon, Bernard. 2007. La Marée Océanique et Côtière. Paris: *Edition Institut* Océanographique.

Smith, S. W. 1999. *Digital Signal Processing*. San Diego: California Technical Publishing.

Suspimiany, Mayang Sari. 2015. Aplikasi Sensor Ultrasonik SRF04 dan Sensor Proximity Pada Level Pengisian Tangki Air Berbasis ATMEGA8535. Palembang: Politeknik Negeri Sriwijaya.

Taufiqurrahman. 2017. Depok. Seluk Beluk Pasang Surut Air Laut. https://langitselatan.com/2017/06/13/seluk-beluk-pasang-surut-air-laut/. Diakses pada tanggal 29 Desember 2019 pukul 19.00.

Triadmodjo, Bambang. 2008. Teknik Pantai. Yogyakarta: Beta Offset Yogyakarta.

Wintolo, Djoko. 2019. Pengantar Statistik dan Geostatistik. Yogyakarta: UGM Press.

LAMPIRAN

Lampiran 1. Data Pasang Surut

Lampiran 1.1 Data Pasang Surut dari Server Thingspeak

Data pasang surut dari server Thingspeak terlampir pada *Compact Disk* (CD).

Lampiran 1.2 Data Pasang Surut dari SD-Card

Data pasang surut dari SD-*Card* terlampir pada *Compact Disk* (CD).

Lampiran 1.3 Data Pasang Surut Penyaringan Moving Average

Data pasang surut dari hasil penyaringan metode *moving average* terlampir pada *Compact Disk* (CD).

Lampiran 1.4 Data Pasang Surut Penyaringan Moving Median

Data pasang surut dari hasil penyaringan metode *moving median* terlampir pada *Compact Disk* (CD).

Lampiran 1.5 Data Pengamaran Pasang Surut Manual

Data pasang surut dari hasil penyaringan metode *moving median* terlampir pada *Compact Disk* (CD).

Lampiran 2. Koding dan Pembuatan *Channel* Thingspeak

Lampiran 2.1 Koding Instrumen

```
#include <Wire.h>
#include "RTClib.h"
#include <SPI.h>
#include <SD.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include<Timer.h>
#include <DHT.h>
#include <DHT_U.h>
Timer t;

LiquidCrystal_I2C lcd(0x27, 2, 1, 0, 4, 5, 6, 7, 3,
POSITIVE);  // Set the LCD I2C address
RTC_DS3231 rtc;



const int chipSelect = 53;
File myFile;

#define DHTPIN 6
#define DHTTYPE DHT22
DHT dht(DHTPIN, DHTTYPE);

#define trigPin 17
#define echoPin 16
long duration, inches;


char *api_key="T10BIAETKHK8J3OP"; // Enter your Write API
key from ThingSpeak
static char postUrl[150];
void httpGet(String ip, String path, int port=80);

void setup()
{
 lcd.begin(16, 2);
  lcd.backlight();
  delay(250);
  lcd.noBacklight();
  delay(250);
  lcd.backlight();

  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
```

```
 lcd.clear();
 lcd.print("   ULTRASONIC   ");
 lcd.setCursor(0,1);
 lcd.print("   TIDE GAUGE   ");
 delay(2000);
 lcd.clear();
 lcd.print("     OLEH :     ");
 lcd.setCursor(0,1);
 lcd.print("NABIL AMIRUL HAQ");
 delay(2000);

 Serial1.begin(9600);
 Serial.begin(9600);

 pinMode(trigPin, OUTPUT); //set the pinmodes
 pinMode(echoPin, INPUT);
 dht.begin();

  if (! rtc.begin()) {
    Serial.println("Couldn't find RTC");
  }

  if (rtc.lostPower()) {
    Serial.println("RTC lost power, lets set the time!");
    rtc.adjust(DateTime(F(__DATE__), F(__TIME__)));     //
following line sets the RTC to the date & time this sketch
was compiled

    // Set the current date, and time in the following
format:
    // rtc.adjust(DateTime(2014, 1, 21, 3, 0, 0));     //
This line sets the RTC with an explicit date & time, for
example to set
  }

  if (!SD.begin(chipSelect)) {
    Serial.println("Card failed, or not present");
    // don't do anything more:
  }
 lcd.clear();
 lcd.print("WIFI Connecting");
 lcd.setCursor(0,1);
 lcd.print("Please wait....");
 Serial.println("Connecting Wifi....");
 connect_wifi("AT",1000);
 connect_wifi("AT+CWMODE=1",1000);
 connect_wifi("AT+CWQAP",1000);
 connect_wifi("AT+RST",5000);
 connect_wifi("AT+CWJAP=\"Nabil\",\"00480048\"",10000);
```

```
 Serial.println("Wifi Connected");
 lcd.clear();
 lcd.print("WIFI Connected.");
 delay(2000);
 t.every(2000, send2server);
}

void loop()
{



  delay(1000);
  t.update();
}
boolean data=false;
void send2server()
{
  float temperature = dht.readTemperature();
  Serial.print("Temperature: ");
  Serial.print(temperature, 1);
  Serial.println("*C");

  digitalWrite(trigPin, LOW);
  delayMicroseconds(5);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  duration = pulseIn(echoPin, HIGH);

  float speedOfSound = 331.3 + 0.606 *
temperature;      //formula for speed of sound in m/s
  Serial.print("speed of sound: ");
  Serial.print(speedOfSound);
  Serial.println(" m/s");

  int jarak = offset_sensor-((((duration *
speedOfSound)/20000)+1.8)*10);     //

  Serial.print("Jarak=");
  Serial.print(jarak);
  Serial.print("mm");
  Serial.println();

  lcd.setCursor(0,0);
  lcd.print("                 ");

  lcd.setCursor(0,1);
  lcd.print("                 ");
```

```
   DateTime now = rtc.now();

  lcd.setCursor(0,0);
  printAngka(now.day());
  lcd.print("-");
  printAngka(now.month());
  lcd.print("-");
  printAngka(now.year());

  lcd.print(" ");
  printAngka(now.hour());
  lcd.print(":");
  printAngka(now.minute());

  lcd.setCursor(0,1);
  lcd.print("TIDES = ");
  lcd.print(jarak);
  lcd.print(" mm");

  sprintf(postUrl,
"update?api_key=%s&field1=%d",api_key,jarak);
  httpGet("api.thingspeak.com", postUrl, 80);


  myFile = SD.open("log.csv", FILE_WRITE);
  if (myFile) {
    myFile.print(now.day());
    myFile.print("/");
    myFile.print(now.month());
    myFile.print("/");
    myFile.print(now.year());
    myFile.print(",");
    myFile.print(now.hour());
    myFile.print(":");
    myFile.print(now.minute());
    myFile.print(":");
    myFile.print(now.second());
    myFile.print(",");
    myFile.println(jarak);
    myFile.close(); // close the file
  }
  // if the file didn't open, print an error:
  else {
    Serial.println("error opening File");
  }

}

void httpGet(String ip, String path, int port)
{
```

```
  int resp;
  String atHttpGetCmd = "GET /"+path+" HTTP/1.0\r\n\r\n";
  //AT+CIPSTART="TCP","192.168.20.200",80
  String atTcpPortConnectCmd =
"AT+CIPSTART=\"TCP\",\""+ip+"\","+port+"";
  connect_wifi(atTcpPortConnectCmd,1000);
  int len = atHttpGetCmd.length();
  String atSendCmd = "AT+CIPSEND=";
  atSendCmd+=len;
  connect_wifi(atSendCmd,1000);
  connect_wifi(atHttpGetCmd,1000);
}

void connect_wifi(String cmd, int t)
{
  int temp=0,i=0;
  while(1)
  {
    Serial.println(cmd);
    Serial1.println(cmd);
    while(Serial1.available())
    {
      if(Serial1.find("OK"))

      i=8;
    }
    delay(t);
    if(i>5)
    break;
    i++;
  }
  if(i==8)
  {
   Serial.println("OK");
  }
  else
  {
   Serial.println("Error");
  }
}
void printAngka(int digits){
   if(digits < 10){
   lcd.print('0');
   lcd.print(digits);

   }
   else lcd.print(digits);

 }
```

Lampiran 2.2 Koding NodeMCU dan Modul GPS

```
#include <SoftwareSerial.h>
#include <TinyGPS++.h>
#include "ThingSpeak.h"
#include <ESP8266WiFi.h>

static const int RXPin = 12, TXPin = 13;
static const uint32_t GPSBaud = 9600;

// repace your wifi username and password
const char* ssid     = "NABILLA";
const char* password = "21139802";

unsigned long myChannelNumber = 906814;
const char * myWriteAPIKey = "T10BIAETKHK8J3OP";

// The TinyGPS++ object
TinyGPSPlus gps;
WiFiClient  client;

// The serial connection to the GPS device
SoftwareSerial ss(RXPin, TXPin);

void setup()
{
  Serial.begin(115200);
  ss.begin(GPSBaud);
  Serial.println(F("DeviceExample.ino"));
  Serial.println(F("A simple demonstration of TinyGPS++ with
an attached GPS module"));
  Serial.print(F("Testing TinyGPS++ library v. "));
Serial.println(TinyGPSPlus::libraryVersion());
  Serial.println();

  Serial.print("Connecting to ");
  Serial.println(ssid);
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
   Serial.println("");
  Serial.println("WiFi connected");
  Serial.println("IP address: ");
  Serial.println(WiFi.localIP());
  Serial.print("Netmask: ");
  Serial.println(WiFi.subnetMask());
  Serial.print("Gateway: ");
  Serial.println(WiFi.gatewayIP());
```

```
  ThingSpeak.begin(client);

}

void loop()
{
  // This sketch displays information every time a new
sentence is correctly encoded.
  while (ss.available() > 0)
    if (gps.encode(ss.read()))
      displayInfo();

  if (millis() > 5000 && gps.charsProcessed() < 10)
  {
    Serial.println(F("No GPS detected: check wiring."));
    while(true);
  }
}


void displayInfo()
{
 // Serial.print(F("Location: "));
  if (gps.location.isValid())
  {

    double latitude = (gps.location.lat());
    double longitude = (gps.location.lng());

    String latbuf;
    latbuf += (String(latitude, 6));
    Serial.println(latbuf);

    String lonbuf;
    lonbuf += (String(longitude, 6));
    Serial.println(lonbuf);

    ThingSpeak.setField(2, latbuf);
    ThingSpeak.setField(3, lonbuf);
    ThingSpeak.writeFields(myChannelNumber,
myWriteAPIKey);
    delay(20000);

  }
  else
  {
    Serial.print(F("INVALID"));
  }

  Serial.print(F("  Date/Time: "));
```

```
  if (gps.date.isValid())
  {
    Serial.print(gps.date.month());
    Serial.print(F("/"));
    Serial.print(gps.date.day());
    Serial.print(F("/"));
    Serial.print(gps.date.year());
  }
  else
  {
    Serial.print(F("INVALID"));
  }

  Serial.print(F(" "));
  if (gps.time.isValid())
  {
    if (gps.time.hour() < 10) Serial.print(F("0"));
    Serial.print(gps.time.hour());
    Serial.print(F(":"));
    if (gps.time.minute() < 10) Serial.print(F("0"));
    Serial.print(gps.time.minute());
    Serial.print(F(":"));
    if (gps.time.second() < 10) Serial.print(F("0"));
    Serial.print(gps.time.second());
    Serial.print(F("."));
    if (gps.time.centisecond() < 10) Serial.print(F("0"));
    Serial.print(gps.time.centisecond());
  }
  else
  {
    Serial.print(F("INVALID"));
  }

  Serial.println();
}
```

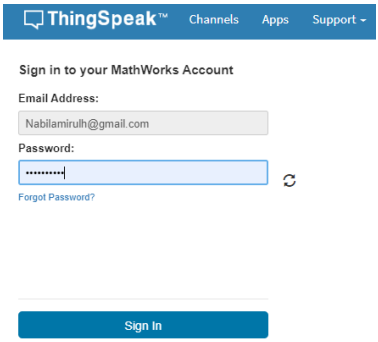
Lampiran 2.3 Koding Penyaringan Data

```
clc, clear
% Properti excel yang akan dimuat
filename = 'Data SD Card.xlsx';
sheet = 'feeds';
xlRange = 'C2:C9622';

% Memuat data elevasi dari excel
data = xlsread(filename, sheet, xlRange);

% Membuat array index urutan elevasi (0 - Jumlah data)
indexes =  0 : length(data) - 1;

% Mengganti elevasi bernilai 0 dengan NaN (missing)
data(data==0) = NaN;

% Memakai fungsi 'fillmissing' untuk melakukan
interpolasi linear ke
% data yang kosong
filled_data = fillmissing(data , 'linear',
'SamplePoints', indexes);

% Filtering data dengan 'moving average'
filter_data = smooth(indexes, filled_data, 300,
'moving');

% Filtering data dengan 'moving median'
filter_data2 = movmedian(filled_data, 100);

% Visualisasi(optional)
plot(indexes,filled_data,':r', indexes,
filter_data2,'-b',indexes,
filter_data,':g','linewidth',3)
legend('Data Pasut Asli', 'Data Pasut Moving Median',
'Data Pasut Moving Average')
axis tight
```

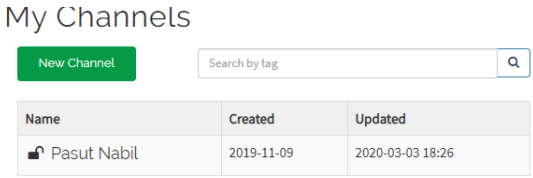
Lampiran 2.4 Pembuatan *Channel* Thingspeak

1. Buka website thingspeak.com

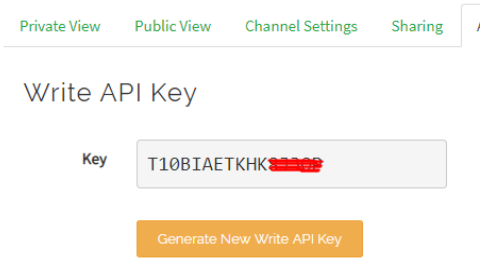
2. Kemudian *log in* dengan email dan kata sandi yang sudah terdaftar di akun MathWorks.



3. Selanjutnya, buat sebuah *Channel* untuk menyimpan data pasut dan menampilkan grafiknya.



4. Kemudian salin kode API pada *channel* untuk disisipkan dalam kodingan instrumen, karena berfungsi untuk menerima data yang dikirim dari instrumen.

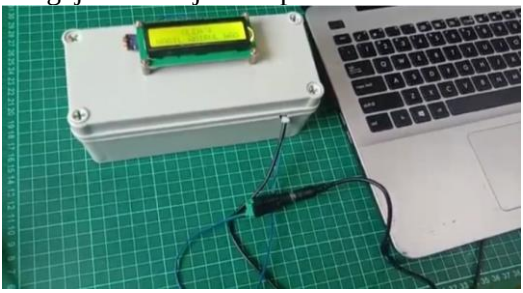


Lampiran 3. Dokumentasi

Lampiran 3.1 Pembuatan Instrumen

1. Perangkaian Komponen Elektronik

2. Pengujian Kinerja Komponen Elektronik

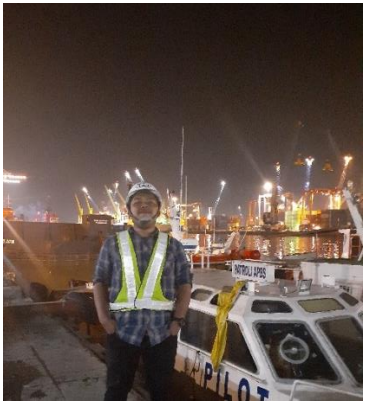


3. Pembuatan Klem

4. Perakitan Bodi Instrumen

Lampiran 3.2 Pengujian Instrumen

1. Penulis



2. Pemasangan Kotak Komponen Elektronik dan Palem Pasut

3. Pemasangan Payung untuk Melindungi Instrumen dari Panas

4. Pengujian Modul GPS

BIODATA PENULIS

Penulis bernama Nabil Amirul Haq, dilahirkan di Kudus pada tanggal 31 Oktober 1998. Penulis merupakan anak pertama dari dua bersaudara. Penulis telah menempuh pendidikan formal di TK Aisyiyah Terpadu Birrul Walidain Kudus, SD Muhammadiyah 1 Kudus, SMPN 1 Kudus, dan SMAN 1 Kudus. Setelah lulus SMA, penulis melanjutkan pendidikan ke jenjang sarjana di Departemen Teknik Geomatika – FTSPK ITS. Selama menjadi mahasiswa, penulis aktif dalam organisasi kemahasiswaan di lingkup ITS sebagai pengurus di HIMAGE-ITS dan BEM ITS, yaitu sebagai staff Departemen Keprofesian dan Keilmiahan HIMAGE-ITS 2017/2018, staff magang Kementerian Riset dan Teknologi BEM ITS 2017/2018, staff Kedirjenan Riset dan Keilmiahan – Kementerian Riset dan Teknologi BEM ITS 2018/2019, direktur jenderal Riset dan Keilmiahan – Kementerian Riset dan Teknologi BEM ITS 2019/2020. Selain melakukan aktivitas di organisasi kemahasiswaan, penulis juga aktif berpartisipasi dalam Program Kreativitas Mahasiswa, dan berhasil mendapatkan pendanaan proposal sebanyak dua kali. Selain itu, penulis juga pernah melaksanakan pekerjaan dan pelatihan yang berkaitan dengan keprofesian teknik geomatika di bidang survey terestris, fotogrametri, survey GNSS, dan survey hidrografi. Dalam penelitian tugas akhir, penulis memilih bidang keahlian hidrografi yang berfokus pada pembuatan instrumen dengan judul tugas akhir "Rancang Bangun Instrumen *Low-Cost Ultrasonic Tide Gauge* dengan Sistem *Internet of Things* (*IoT*) Berbasis Arduino".