

# SENSOR-SENSOR PEMANTAU MENGGUNAKAN ARDUINO UNO DAN ESP8266 DIUNGGAH KE THINGSPEAK SERVER UNTUK PENGUKURAN GAS DI TEMPAT PEMBUANGAN AKHIR MANGGAR BALIKPAPAN

AGUS BENY IMRON<sup>1</sup>, Dr. Ir. Charles Pangaribuan, DEA, CES<sup>2</sup>

**Abstract**— Oil and gas industri is one of very development sector in technology. The data acquisition system or telemetry with remote monitoring and storage are one of them. Today, this technology can be applied into society to be usefull and give more advantages. By using an open data platform for Internet of Things (IoT), it is very possible. - Final landfil has produced many gasses such as methane and karbondioksida, especially methane gas which is produced by deterioration of organic waste. These emit gasses are one global warming root caused, we call it green house effect gasses. The gas concentrations is hard to be measured without a device help, it is safe and prefered. - On this final project the we build build a prototype that contain Arduino Uno, ESP826 Wi-Fi module and Sensors. It detects reading for Methane (CH<sub>4</sub>), Carbon Dioxide (CO<sub>2</sub>), Temperature and Humidity. All reading data for gasess, temperature and humidity then send through Wi-Fi internet be uploaded into [www.thingspeak.com](http://www.thingspeak.com), using internet browser we can view the reading graph from ThingSpeak, anyone who has internet acces can be seen all the data remotely

**Keywords** : Arduino Uno, Gas sensor, Temperature sensor, Humidity sensor ESP8266 Wi-Fi Module, Wi-Fi modem internet connection, ThingSpeak open data platform and internet of things (IoT).

**Intisari**— Industri minyak dan gas adalah salah satu sektor yang sangat berkembang dengan teknologinya. Sistem data akuisisijarak jauh atau telemetri adalah satu diantaranya. Hari ini teknologi tersebut dapat diaplikasikan di masyarakat dan memberi banyak manfaat. Dengan memakai platfond data terbuka *Internet of Things* (IoT) hal tersebut dapat terwujud. - Tempat Pembuangan Akhir (TPA) di Manggar, Balikpapan bisa kemungkinan dapat menghasilkan gas-gas berbahaya seperti, karbondioksida dan metana khususnya yang dihasilkan dari proses pembusukan sampah organik. Emisi gas-gas tersebut faktor penyebab pemanasan global, kita menyebutnya efek gas rumah kaca. Konsentrasi gas-gas ini sulit diukur tanpa bantuan alat, dan alat tersebut dibutuhkan untuk memperoleh data. -Penulis membuat sebuah prototipe alat yang terdiri dari Arduino Uno, modul Wi-Fi ESP8266 dan sensor-sensor. Alat ini mampu mendeteksi gas metana (CH<sub>4</sub>), karbondioksida (CO<sub>2</sub>), suhu dan kelembapan. Data-data tersebut dikirim dan disimpan di [www.thingspeak.com](http://www.thingspeak.com), kemudian dengan menggunakan internet browser dapat ditampilkan hasilnya dalam bentuk grafik dan orang yang memiliki akses internet dapat melihat data-data tersebut dari kejauhan

**Kata Kunci**— Arduino Uno, Sensor gas, Sensor suhu, Sensor kelembapan, Modul Wi-Fi ESP8266, koneksi internet modem Wi-Fi, ThingSpeak platform data terbuka *Internet of Things* (IoT).

## I. PENDAHULUAN

Pemanasan global (*global warming*) adalah proses meningkatnya suhu rata-rata atmosfer, laut, dan daratan Bumi disebabkan oleh meningkatnya konsentrasi gas-gas rumah kaca. Ga-gas tersebut terdiri dari Uap air, karbondioksida,

atmosfer akibat penguapan air dari laut, danau dan sungai.

Gas karbondioksida dan metana adalah penyumbang terbanyak kedua ketiga penyebab pemanasan global. Metana mampu menyumbang panas 20-25 kali lebih banyak bila dibandingkan karbondioksida. Metana banyak dihasilkan dari pembusukan limbah organik di tempat pembuangan akhir (TPA), contohnya di TPA Manggar, Balikpapan. Gas-gas yang dihasilkan TPA antara lain adalah amonia, sulfida, metana, dan karbondioksida. Amonia dan sulfida adalah penyumbang bau busuk menyegat di TPA. Metana dan karbondioksida selain penyumbang pemanasan global, mereka juga sebagai polusi, mampu menyingkirkan oksigen disekitar, mampu berkumpul di area rendah dan bangunan sekitar, dikarenakan berat jenisnya yang lebih berat.

Sifat gas metana mudah terbakar bahkan dapat meledak diruang tertutup, sehingga dimanfaatkan sebagai bahan bakar kompor untuk warga 2 RT disekitar TPAManggar, Balikpapan dan pembangkit listrik terbatas digunakan untuk kebutuhan internal TPA. Dengan inovasi yang terus berkembang menjadikan TPA Manggar, Balikpapan banyak menoreh prestasi, ditahun ini TPA Manggar, Balikpapan memperoleh 2 penghargaan.

Untuk menambahkan inovasi di TPA Manggar, Balikpapan, pemantauan konsentrasi gas metana (CH<sub>4</sub>), gas karbondioksida (CO<sub>2</sub>), suhu (celsius) dan kelembapan (%) secara periodik di TPA perlu, untuk data mengetahui tingkat polusi dan kandungan gas yang dihasilkan. Untuk itu perlu dibuat suatu alat yang dapat digunakan untuk mengumpulkan data kandungan berupa tingkat konsentrasi gas yang lepas ke atmosfer atau dapat mengukur langsung di sistem tertutup pengolahan gas metana. Data-data hasil pengukuranyang diperoleh alat ini dapat ditransmisikan, diunggah atau diupload, tersimpan dan dipantau jarak jauh, menggunakan ThingSpeak. Ditambah *display* LCD OLED dialat yang dapat membantu menampilkan hasil pengukuran di alat.

## II. TINJAUAN PUSTAKA

### Arduino Uno

Adalah sebuah papan mikrokontroler yang didasarkan pada ATmega328. Arduino Uno mempunyai 14 pin digital *input* dan *output* (6 di antaranya dapat digunakan sebagai *output* PWM), 6 *input* analog, sebuah osilator Kristal 16 MHz, sebuah koneksi USB, sebuah *powerjack*, sebuah ICSP *header*, dan sebuah tombol *reset*. Arduino Uno memuat semua yang dibutuhkan untuk menunjang mikrokontroler, mudah menghubungkannya ke sebuah computer dengan sebuah kabel USB atau mensuplainya dengan sebuah adaptor AC ke DC atau menggunakan baterai untuk memulainya. Arduino Uno menggunakan ATmega16U2 yang diprogram sebagai USB-*to*-*serial converter* untuk komunikasi serial ke computer melalui

<sup>1,2</sup>Program Studi Teknik Elektro Fakultas Teknik Universitas Balikpapan, Jln. Pupuk Raya Gn. Bahagia Balikpapan 76114 INDONESIA (e-mail: [agusbenyimron@gmail.com](mailto:agusbenyimron@gmail.com), [pangaribuan\\_charles@yahoo.com](mailto:pangaribuan_charles@yahoo.com))

port USB. Arduino board diatas memiliki fisik ukurannya sebesar kartu kredit.



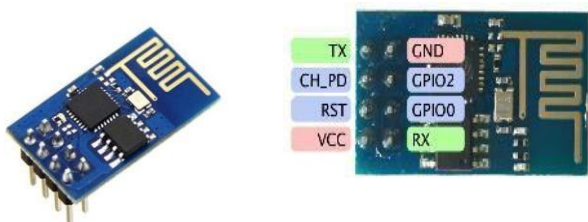
Gambar 2.1 Blok Arduino Uno

**Modul Wi-Fi ESP8266**

Adalah modul komunikasi Wi-Fi dengan harga ekonomisi untuk dapat menyambungkan rangkaian elektronika dengan internet secara nirkabel karena modul elektronika ini menyediakan akses ke jaringan Wi-Fi secara transparan dengan mudah melalui interkoneksi serial (UART RX/TX).

Keunggulan utama modul ini adalah tersedianya mikrokontroler RISC (*Tensilica 106µ Diamond Standard Core LX3*) dan *flash memory* SPI 4 Mbit Winbond W2540BVNIG terpadu, dengan demikian Anda dapat langsung menginjeksi kode program aplikasi langsung ke modul ini.

Modul ini juga sangat mudah untuk dihubungkan dengan perangkat Arduino atau dengan kata lain menjadi Arduino Wi-Fi *shield*. Karena memiliki *processor* sendiri, maka modul ini dapat berdiri sendiri tanpa Arduino atau mikrokontroler, serta mendukung APSD (*Automatic Power Save Delivery*) untuk aplikasi VoIP (*Voice over IP*).

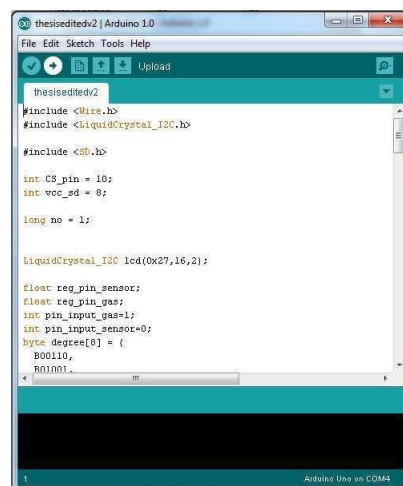


Gambar 2.2 Modul ESP8266-01

**Arduino Integrated Development Environment**

Arduino *Integrated Development Environment* (IDE) merupakan perangkat lunak yang terdiri dari editor teks untuk menulis kode, sebuah area pesan, sebuah konsol, sebuah *toolbar* dengan tombol-tombol fungsi yang umum dan beberapa menu. Arduino IDE terhubung ke Arduino board untuk meng-*upload* program dan juga untuk berkomunikasi dengan Arduino board. Perangkat lunak yang ditulis menggunakan Arduino IDE disebut *sketch*. *Sketch* ditulis pada

editor teks. *Sketch* disimpan dengan file berekstensi ino. Area pesan memberikan informasi dan pesan *error* ketika kita menyimpan atau membuka *sketch*. Konsol menampilkan teks dari Arduino IDE dan juga menampilkan pesan *error* ketika kita meng-*compile sketch*. Pada sudut kanan bawah dari jendela Arduino IDE menunjukkan jenis *board* dan serial *port* yang sedang digunakan. Tombol *toolbar* digunakan untuk mengecek dan meng-*upload sketch*, membuat, membuka atau menyimpan *sketch*, dan menampilkannya di monitor serial.



Gambar 2.3 Arduino IDE

**Mengenal Pustaka Arduino**

Linkungan kerja kerja Arduino dapat di perluas lagi melalui pustaka-pustakanya (*libraries*), seperti kebanyakan *platform* program lainnya. *Library* menyediakan subrutin yang memiliki fungsi-fungsi ekstra untuk pemakain *sketch*, baik itu untuk perangkat kerasnya maupun dalam hal memanipulasi data. Untuk memakai *library* dalam sebuah *sketch*, pilih dari *Sketch>ImportLibrary*.

**Mengenal Wi-Fi**

Wi-Fi (*Wireless Fidelity*) adalah satu standar Wireless Networking tanpa kabel, hanya dengan komponen yang sesuai dapat terkoneksi ke jaringan. Awalnya Wi-Fi ditujukan untuk penggunaan perangkat nirkabel dan LAN (*Local Area Network*), namun saat ini lebih banyak digunakan untuk mengakses Internet. Hal ini memungkinkan seseorang dapat menggunakan komputer, laptop dan perangkat lainnya untuk terhubung dengan Internet menggunakan titik akses (*accespoint*) atau dikenal dengan hotspot.

Acces point memungkinkan perangkat nirkabel untuk terhubung ke jaringan menggunakan Wi-Fi. Jangkauan pada Access Point sekitar 100 meter. Jika melebihi jarak yang dapat di jangkau oleh *AccessPoint* maka sinyal akan hilang. Untuk memperkuat frekuensi sinyal bias memakai penguat atau memasang *Access Point* yang lebih dari 1 di beri jarak sekitar 30 meter lebih untuk memperkuat frekuensi sinyal

sekitar.

*Access Point* berfungsi sebagai *Hub Switch* yang bertindak untuk menghubungkan jaringan lokal dengan jaringan *wireless*, di *access point* inilah koneksi data internet dipancarkan atau dikirim melalui gelombang radio, ukuran kekuatan sinyal juga mempengaruhi radius yang dapat dijangkau, semakin besar kekuatan sinyal (ukurannya dalam satuan *dBm* atau *mW*) semakin luas jangkauannya.

**Sensor Gas MQ4 dan MQ135**

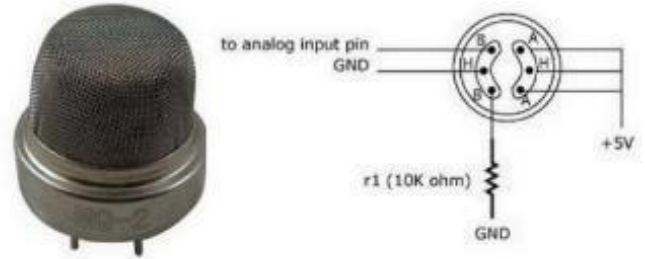
Sensor gas dengan kode MQ terdiri dari 2 bagian, yaitu sensor elektrokimia dan sebuah pemanas (*internalheater*) didalamnya. Sensor ini dapat mendeteksi berbagai tipe gas, dan akan lebih sensitif untuk jenis gas tertentu, tergantung jenis sensor yang terpasang. Semua sensor gas tipe ini dapat dikalibrasi dengan mengukurnya pada udara atau gas yang telah diketahui konsentrasinya.

Keluaran sensor ini berupa data analog yang dapat dibaca oleh pin-pin Analog Arduino. Untuk menggunakan jenis sensor MQ ini, diperlukan datasheet guna mengetahui karakteristik dari sensor tersebut. Hal ini berhubungan dengan bagaimana sensor ini diaktifkan, dan bagaimana korelasi data keluaran dengan besaran gas yang akan Anda ukur.

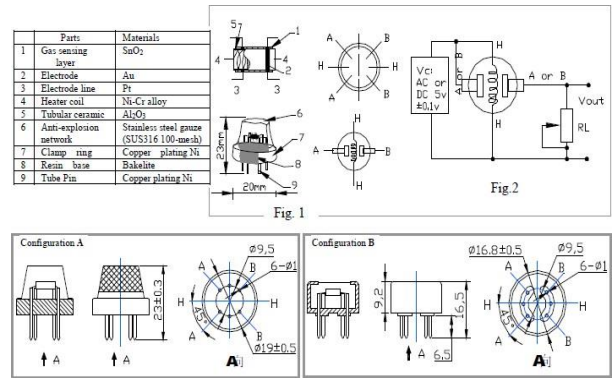
Berhati-hatilah saat menghubungkannya, karena jika salah, maka sensor akan mempengaruhi keluaran sensor, atau bahkan bisa rusak seketika. Perhatikan kembali datasheet dan cara menghubungkannya dengan mikrokontroler Arduino.

Yang terpenting kedua adalah mengenai tegangan untuk internal heater. Sebagian sensor memerlukan tegangan 5V terus menerus untuk memanaskan heaternya, sebagian lagi memerlukan 2V, dan yang sebagian lagi memerlukan tegangan 5V dan 1.4V secara bergantian. Tegangan 2V dan 1.4V bisa didapat dengan menggunakan analogWrite pada pin PWM Arduino. Namun perlu diperhatikan pula (di datasheet) jumlah arus yang diperlukan untuk pemanasan tersebut. Jika lebih besar dari 40mA, maka sebaiknya menggunakan mosfet atau transistor switching, karena jika memaksakan menggunakan Arduino maka akan kelebihan beban.

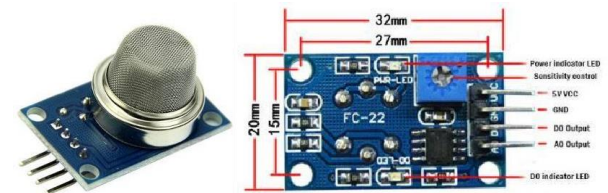
Sensor yang menggunakan tegangan 5V untuk internal heater-nya, biasanya sangat cepat untuk mencapai 50-60 derajat celsius. Setelah *burn-in-time*, internal heater umumnya diperlukan untuk mencapai pembacaan yang stabil (contohnya pada MQ4 dan MQ135). Beberapa datasheet menggunakan istilah *preheat* atau sebuah prasyarat untuk membuat pembacaan sensor lebih stabil. *Burn-in-time* atau *preheat* biasanya dalam 12 hingga 48 jam hanya sekali di awal sebelum di gunakan pada rangkaian, setelah itu normal digunakan setelah pemanasan 2-3 menit. Untuk tugas akhir ini penulis memilih sensor MQ4 untuk melakukan pengukuran konsentrasi gas metana (CH4) dengan jangkauan 200-10,000 PPM dan sensor MQ135 untuk pengukuran konsentrasi karbondioksida (CO2) dengan jangkauan sensor bisa di atur sendiri limit, karena *datasheet* MQ135 tidak menerangkan jangkauannya.



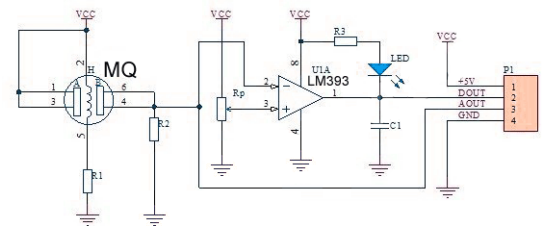
Gambar 2.4 Sensor Gas MQ



Gambar 2.5 Komponen Dasar Gas MQ



Gambar 2.6 Modul Sensor Gas MQ

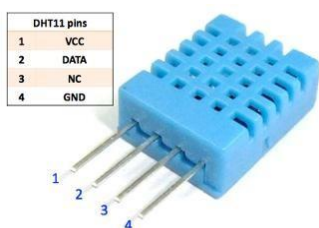


Gambar 2.7 Rangkaian Modul Sensor MQ

**Sensor Suhu dan Kelembapan (DHT11)**

Alat yang biasanya digunakan untuk mengukur kelembapan udara adalah higrometer. DHT11 adalah sensor keluaran digital yang dapat mengukur suhu dan kelembapan udara di sekitarnya. Sensor ini sangat mudah digunakan

bersamadengan Arduino.Memiliki tingkat stabilitas yang sangat baik serta fitur kalibrasiyang sangat akurat.Koefisien kalibrasi disimpan dalam program memory,sehingga ketika internal sensor mendeteksi sesuatu, maka module inimenyertakan koefisien tersebut dalam kalkulasinya,DHT11 ini termasuk sensoryang memiliki kualitas terbaik, dinilai dari respon, pembacaan data yang cepat,dan kemampuan anti-*interference*. Ukurannya yang kecil, dan dengan transmisisinyal hingga 20 meter,dengan sepsifikasi: Catu daya: +5 V, jangkauan suhu 0-50 °C *error of* ± 2 °C, kelembapan : 20-90% RH ± 5% RH *error*,denganspesifikasi *digital interfacing system*. membuat produk ini cocok digunakan untukbanyak aplikasi-aplikasi pengukuran suhu dan kelembapan.



Gambar 2.8 Sensor DHT11

Tabel 2.1 Tabel karakteristik sensor DHT11

Model	DHT11
Catu Daya	3-5.5V DC
Keluaran	Sinyal digital
Jangkauan dan akurasi	Kelembapan 20-90% RH ; ± 5% RH Suhu 0-50C ; ±2Celsius
Resolusi atau sensitifitas	Kelembapan 1%RH; Suhu 0.1Celsius
Repeatability	Kelembapan ± 1%RH; Suhu ±1Celsius
Kelembapan <i>hysteresis</i>	± 1%RH
Kestabilan jangka panjang	± 0.5%RH per tahun
Periode pembacaan	Rata-rata 2detik
<i>Interchangeability</i>	<i>fully interchangeable</i>
Dimensi	12x15.5x5.5mm

Dari penjelasan Tabel 2.4 diatas bahwa struktur yang merupakan carakerja dari sensor suhu dan kelembapan udaraDHT11 memiliki empat buah kakiyaitu: pada bagian kaki(VCC), dihubungkan ke bagian Vss yg bernilai sebesar5V,pada ArduinoUno dan untuk bagian kaki GND dihubungkan ke ground(GND, sedangkan pada bagian kaki data yangmerupakan keluaran dari hasil pengolahan data analog dari sensorDHT11 yang dihubungkan ke bagian analog input pin Arduino Uno dan terdapat satu kaki tambahan yaitu kaki NC (*Not Connected*), yangtidak terhubung.

**128X64 OLED LCD DISPLAY**

*Display* ini kecil berukuran 0.96 inch diameter, namun

sangat mudah terbaca dikarenakan tingkat cahaya kontras yang tinggi sebagai penampil OLED (*Organic Light Emitting Diode*). *Display* ini terbuat dari 128x64 individu OLED *pixels*, masing-masing dihidupkan dan dimatikan oleh *chip* pengontrol SSD1306. Dikarenakan *display* tersebut memakai cahayanya sendiri maka tidak diperlukan cahaya latar belakang (*background light*). Hal ini mengurangi konsumsi daya yang dibutuhkan untuk menjalankan OLED dan itulah sebabnya *display* ini memiliki cahaya kontras yang tinggi dan mampu menampilkan tulisan dan gambar.



Gambar 2.9 OLED LCD 128x64

SSD1306 dilengkapi dengan kontrol kontras, RAM *display* dan osilator, yang dapat mengurangi komponen luar dan konsumsi daya. Memiliki 256 level kontrol cahaya. Data atau perintah yang dikirim dari MCU memiliki *Interfaces* (antarmuka) pilihan perangkat seri 6800/8000 *parallel Interfaces*, I2C atau SPI. *Display* ini cocok untuk banyak alat yang portabel seperti *handphone*, *display* MP3 *player*, kalkulator dan lain-lain. Berikut adalah daftar fitur yang dimiliki:

- a. Resolusi 128 x 64 dot panel matrik.
- b. Tegangan operasi VDD = 1.65V sampai 3.3V untuk IC logika dan VCC = 7V sampai 15V untuk panel matrik.
- c. Untuk panel matrik maksimum tegangan 15V DC dengan arus 100uA sampai 15mA, dan 256 level kecerahan.
- d. Dilengkapi 128 x 64 bit SRAM *display buffer*.
- e. Pin pilihan antarmuka MCU : 8-bit 6800/8080-SPI , 3/4 kabel SPI atau I2C *Interface*.
- f. Fungsi penyimpanan layar kontinu untuk arah horisontal dan vertikal.
- g. RAM tulis untuk sinyal sinkronisasi.
- h. *Programmable Frame Rate and Multiplexing Ratio*.
- i. Pemetaan baris dan kolom .
- j. Osilator.
- k. Jangkauan suhu: -40°C sampai dengan 85°C.

**ThingSpeak**

*ThingSpeak* merupakan *open source* untuk aplikasi *InternetofThings* (IoT) dan *Aplication Programing Interface* (API), untuk mengunggah dan mengunduh data memakai protokol internet HTTP. Dengan *ThingSpeak* dapat dilakukan

pengumpulan data-data sensor, aplikasi pelacak koordinat lokasi, dan melakukan sosial media dengan status yang *ter-update*. Dapat menjadi server data base dan juga terintegrasi dengan layanan *Twitter* sosial media. Terdapat aplikasi *ThingTweet* dan *React* yang dikombinasikan, sehingga bisa dipakai sebagai alat peringatan dini jika level konsentrasi gas mencapai angka tertentu, pengaturannya cukup mudah di *ThingSpeak*, bisa dilihat *tutorial*-nya di lampiran.



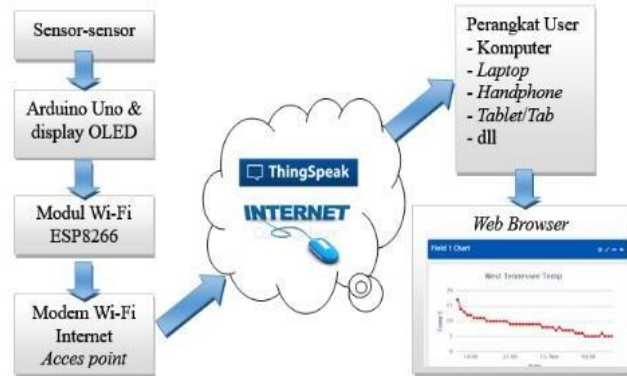
Gambar 2.10 Ilustrasi ThingSpeak

*Www.thingspeak.com* pertama kali diluncurkan oleh *ioBrige* di tahun 2010 sebagai pelayan dalam mendukung aplikasi *Internet of Thing (IoT)*. ThingSpeak memiliki dukungan integrasi dari *MATLAB*, membolehkan para pemakai untuk melakukan analisa dan visualisasi data menggunakan *Matlab* tanpa harus membeli lisensi dari *Matworks*.

### III. METODE PENELITIAN

#### 3.1 Perancangan Diagram Blok

Sebagai gambaran umum alat yang dibuat berupa pembacaan sensor-senor gas, suhu dan kelembapan. Alat ini mengolah pembacaan sensor oleh *Arduino Uno* ditampilkan di *LCD OLED*, melakukan koneksi menggunakan modul *Wi-Fi ESP8266* terhubung internet *Wi-Fi, hotspot* atau *accespoint*. Setelah terkoneksi dengan internet semua pembacaan parameter akan dikirim ke *ThingSpeak*, hasil ditampilkan secara grafis. Ilustrasi blok-blok sistem dari rangkaian terlihat pada gambar 3.1 dibawah.



Gambar 3.1 Ilustrasi Diagram Perancangan

Keluaran dari sensor-sensor diteruskan menuju *Arduino Uno*, diproses untuk mendapatkan parameter pembacaan yang diinginkan. Keluaran dari *MQ4* dan *MQ135* berupa analog, dan hasil yang kita inginkan adalah pembacaan tingkat konsentrasi *CH4* dan *CO2* dalam satuan *PPM (Parts per Million)*. Sedangkan untuk sensor *DHT11* memiliki keluaran digital dengan hasil pembacaan suhu (celcius) dan kelembapan (%). Dengan dibantu fungsi-fungsi library, maka penulis cukup memberikan perintah sederhana guna mendapatkan hasil yang diinginkan. Tanpa diperlukan penulisan panjang program *ADC (Analog to Digital Converter)*. Berikut dibawah ini tabel perintah utama yang digunakan untuk mendapatkan parameter yang kita inginkan.

Tabel 3.1 Perintah Utama Pembacaan Sen

Perintah di Arduino IDE	keterangan
<code>float t = dht.readTemperature()</code>	Membaca suhu celsius
<code>float h = dht.readHumidity()</code>	Membaca kelembapan %
<code>int k = mq135.getPPM()</code>	Membaca CO2 dalam PPM
<code>int m = mq4.readCH4()</code>	Membaca CH4 dalam PPM

*Display LCD OLED* digunakan untuk mempermudah pembacaan proses koneksi *Wi-Fi* dan hasil pembacaan sensor. Untuk proses koneksi dengan *Wi-Fi* fungsi *OLED* dimasukan kedalam `void setup()` sedangkan untuk pembacaan sensor dimasukan kedalam `void loop()`. Agar lebih mudah dan pendek penulisan program, penulis membuat fungsi terpisah khusus untuk tampilan *OLED*. Terdapat 2 perintah sederhana untuk fungsi tampilan *OLED* yaitu `u8g.drawStr()` dan `u8g.println()`

.ESP8266 diperintahkan untuk melakukan koneksi dengan modem Wi-Fi, semua perintah bersumber dari Arduino Uno. Setelah terbangun koneksi Wi-Fi dan mendapatkan akses internet,

kemudian Arduino memulai koneksi TCP/IP dengan [www.thingspeak.com](http://www.thingspeak.com). Untuk berhubungan dengan ThingSpeak diperlukan API key yang sesuai.

Selanjutnya terjadi proses pengunggahan (upload ) data parameter sensor-sensor menuju ThingSpeak. Semua data yang diterima akan di publikasikan, sehingga perangkat lain di tempat yang jauh dapat melihat hasil pantauan data parameter. Untuk melihat hasilnya cukup melakukan *browsing* membuka [www.thingspeak.com](http://www.thingspeak.com) dengan *channel ID* yang sesuai. Berikut adalah tabel daftar perintah yang di pakai menggunakan AT *command*:

**Tabel 3.2 AT Command Untuk ESP8266**

Perintah AT Command	keterangan
AT+RST\r\n	Reset modul W
AT+CWMODE=1\r\n	Konfigurasi sebagai <i>access point and client</i>
AT+CWQAP\r\n	Keluar dari AP Wi-Fi
AT+CWJAP="SSID","PASSWORD"	Koneksi ke Wi-Fi <i>access point</i> SSID = Speedy PASSWORD = 1234567890
AT+CIFSR\r\n	Cek koneksi IP <i>address</i>
AT+CIPSTART=TCP, IP,80	Membangun koneksi TCP/ <i>port</i> 80 dengan <a href="http://www.thingspeak.com">www.thingspeak.com</a> IP= 184.106.153.149
AT+CIPSEND=GET /update?api_key=QWZJCF7WF2PHMXXSY cmd += "&field1="; cmd +=String(temp); cmd += "&field2="; cmd +=String(humid); cmd += "&field3="; cmd +=String(co2); cmd += "&field4="; cmd +=String(ch4); cmd += "\r\n\r\n";	Unggah (upload) data ke <a href="http://www.thingspeak.com">www.thingspeak.com</a>
AT+CIPCLOSE	Menghentikan pengiriman data

**1) 3.2 Teknik Pengumpulan Data**

Beberapa metode dilakukan penulis dalam pengumpulan data, metode tersebut antara lain adalah :

- a. Pembelajaran menggunakan literatur  
Mengumpulkan dan membaca teori yang berkaitan dengan tugas akhir ini, berupa buku-buku referensi, artikel, jurnal, internet, tutorial dan lain-lain. Mempelajari datasheet MQ4, MW135 dan DHT11, bagaimana pembacaan di Arduino, menampilkan di OLED dan mencari *library* yang tepat.
- b. Latihan Uji coba  
Melakukan uji coba Arduino dengan demo-demo latihan, bertujuan untuk memahami dan mempelajari Arduino secara praktek. Seiring waktu belajar teori dibuku mengenai pendalaman materi perangkat lunak Arduino IDE.

- c. Diskusi Alat dan Materi  
Berupa diskusi dan konsultasi mengenai topik tugas akhir dengan dosen pembimbing yang telah di tentukan oleh pihak departemen Teknik Elektro Universitas Balikpapan.

**Alat dan Bahan**

Dalam perancangan dan pembuatan alat akan di butuhkan persiapan dua perangkat utama yaitu perangkat keras dan perangkat lunak.

**Perangkat Keras**

Komponen utama tersusun menjadi beberapa bagian anantara lain yaitu sensor-sensor, mikrokontroler Arduino, Wi-Fi modul dan modem Wi-Fi untuk internet. Berikut adalah perangkat keras yang digunakan penulis :

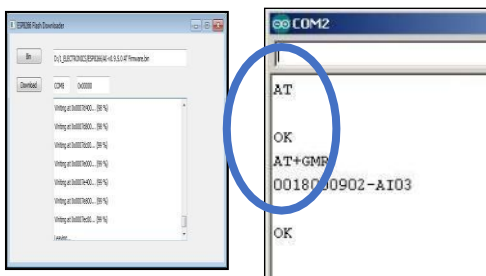
- a. Sensor  
Terdiri sensor gas MQ4, sensor gas MQ135 dan sensor DHT11. Untuk sensor gas penulis memilih MQ4 untuk metana dan karbondioksida dari MQ135, keluarannya analog. DHT11 untuk pengukuran suhu dan kelembapan udara, dengan keluarannya digital.
- b. ESP8266  
Adalah modul kecil Wi-Fi bertipe SP-01, dengan pin TX/RX dengan mikrokontroler unit Atmeg328 di Arduino *board* . Modul ini memiliki catu daya 3.3 volt dan GPIO secara mandiri, sehingga dapat beroperasi sendiri tanpa mikrokontroler. Namun untuk tugas akhir ini module tersebut hanya berfungsi sebagai modul Wi-Fi. Bahasa AT *command* digunakan untuk mengeksekusi perintah yang dikirim dari Arduino Uno.
- c. Arduino Uno  
Dengan komponen utama mikrokontroler berbasis Atmega328, merupakan otak kerja semua sistem. Dari sini sensor-sensor dibaca dan diolah menjadi satuan konsentrasi gas CO2/CH4 dalam PPM, suhu dalam celsius dan kelembapan dalam persen. Arduino memberikaan perintah-perintah khusus kepada modul Wi-Fi untuk dapat terkoneksi dengan internet dan *ThingSpeak* dapat dilihat di Tabel 3.2 diatas.
- d. Display LCD 0.96 in OLED 128x64  
Display LCD OLED hanya berfungsi sebagai penampil *monochrome* saja untuk melihat pembacaan sensor tanpa laptop monitor serial. Hasil pembaccan dapat dilihat seperti pada Gambar 2.15 diatas.
- e. Modem Wi-Fi  
Untuk mendukung koneksi internet via Wi-Fi penulis pergunakan modem Wi-Fi *router* Speedy dari Telkom (ADSL TP-link TD-W8951ND). Dapat juga menggunakan *handphoneandroid* sebagai *accesspoint* dengan fasilitas *tethering* dan *portablehotspot*.

- f. Kabel dan Papan Terminal  
Untuk menghubungkan Arduino Uno dengan laptop kita gunakan kabel data USB. Fungsinya sebagai catu daya 5 volt dan sebagai penghubung serial data untuk proses upload bahasa pemrograman kedalam Arduino board. Terdapat juga kabel jumper untuk koneksi antar pin di papan terminal dan Arduino Uno.
- g. Netbook Lonovo S100C  
Digunakan untuk mengakomodasi semua perangkat lunak yang diperlukan untuk Arduino, meng-upload program *sketch*, *flashing* modul Wi-Fi dan melihat hasil di website.
- h. Alat-Alat Pendukung lainnya  
Untuk mempermudah perancangan alat dibutuhkan alat-alat pendukung lainnya seperti :Solder listrik, timah solder, obeng, tang potong, tang jepit, gergaji, PCB dua lapis, pisau, baterai *lithium* 3.7 VDC (2 buah), komponen *regulator* 5V IC LM7805, adaptor DC, *regulated DC Power Supply* dan multimeter osiloskop.

**Perangkat Lunak**

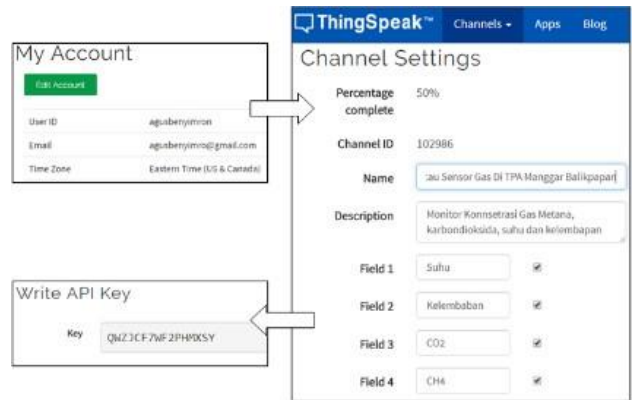
Aplikasi perangkat lunak adalah komponen penting, karena disinilah kita memprogram Arduino sesuai tujuan penulis. Proses yang terjadi berupa *downloading*, *writing*, *compile*, *serial monitoring*, *uploading* dan *browsing*. Ada tiga perangkat lunak yang harus di miliki, antara lain adalah :

- a. ESP8266 Flash Downloader  
Sebelum melakukan perancangan dengan Arduino Uno, perlu kita tetapkan bahasa program apa yang akan digunakan ESP8266. Bahasa pemrograman AT *command* lebih tepat digunakan untuk berkomunikasi dengan Arduino Uno. Terkadang *firmware* asli bawaan dari pabrikan menggunakan bahasa LUA, jika didapati *firmware* ESP8266 tidak support AT *command* maka diperlukan perangkat lunak *flasher*, perangkat lunak *flasherfirmware* ini banyak pilihannya bisa di-download tanpa bayar. Gunakan *firmware* terbaru untuk ESP8266 (Versi 0.9.5.0) dan gunakan ESP8266 *Flash Downloader* lihat Gambar 3.2 proses format *firmware*. Untuk memastikan apakah ESP8266 sudah menggunakan AT *command* dapat dilihat pada serial monitor dengan mengetik AT, maka akan muncul pesan OK.



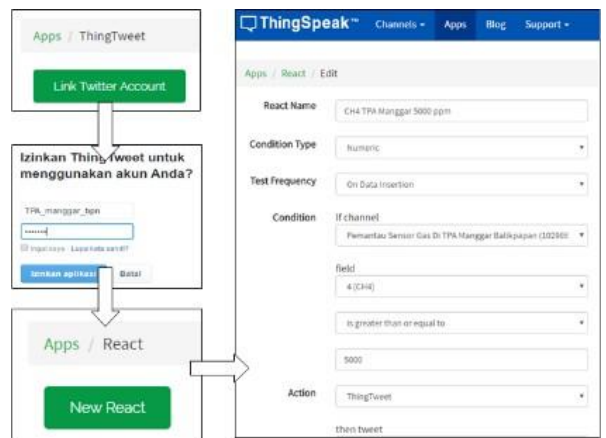
Gambar 3.2 Flash Downloader dan AT command

- b. ThingSpeak  
Masuk [www.thingspeak.com](http://www.thingspeak.com), buat *account* baru, kemudian masuk *My Channels* dan buat *channel* baru isikan empat *field* pembacaan untuk suhu, kelembapan, CO2 dan CH4, klik kotak *make public*. Jika berhasil akan diperoleh hasil seperti pada Gambar 3.3 dibawah. Sehingga diperoleh channel ID 102986 dan API key (QWZJCF7WF2PHMXXSY).



Gambar 3.3 Account dan Channel Baru

Untuk fasilitas *Twitter* itu sendiri, bisa kita gunakan *ThingTweet* dan *React*, dimana pengaturannya cukup mudah hanya di lakukan didalam *ThingSpeak*, tanpa menuliskan bahasa pemrograman Arduino. Contoh aplikasinya, yaitu mengirimkan pesan bahaya jika tingkat konsentrasi gas mencapai level tertentu. Atau memberikan informasi secara kontinu mengenai kualitas udara dengan periode waktu yang diatur. Pastikan sudah memiliki akun *User ID Twitter* yang bisa digunakan. Pertama masuk ke *ThingTweet* lakukan registrasi (*Link Twitter Account*), dari sini kita akan diminta izin agar *ThingTweet* menggunakan akun *Twitter* anda, selanjutnya masuk ke *React* dan atur kondisi serta pesan apa yang kita inginkan, lihat Gambar 3.4 dibawah.



Gambar 3.4 ThingTweet dan React

- c. Arduino IDE
 

Perancangan perangkat lunak utamanya menggunakan Arduino IDE dengan bahasa C. Pilihlah *library* yang tepat untuk sensor yang akan digunakan, hal ini untuk memudahkan penulisan bahasa pemrograman Arduino. Setelah selesai tersusun bahasa pemrograman Arduino Uno dan pastikan masukan API key QWZJCF7WF2PHMXXSY, langkah berikutnya adalah proses *compile* dan *uploading* menuju Arduino board. Setelah muncul komentar *doneuploading* tanpa error, maka Arduino Uno akan menjalankan program sesuai tujuannya. Untuk melihat proses jalanya program dapat dilihat di jendela serial monitor, disini dapat terlihat apa yang terjadi di Arduino Uno.

```

DHT11_MQ2_PIR_Rain_Thingspeak
}

void setup(){
  delay(1000);
  pinMode(PIR_Pin, INPUT);
  pinMode(RAIN_Pin, INPUT);
  pinMode(led1, OUTPUT);
  pinMode(led2, OUTPUT);
  esp8266.begin(9600);
  Serial.begin(9600);
  sendDebug();
  delay(400);
  if(esp8266)
}

DHT11_MQ2_PIR_Rain_Thingspeak7
void loop(){
  char buffer[10];
  Serial.println();
  Serial.println("<---- Didalam Loop Utama ---->");
  //-----Sensor Hujan-----\\:
  int inputRAIN = digitalRead(RAIN_Pin);
  if (inputRAIN == HIGH){
    Serial.println("Tidak Hujan");
    rain = 0;
  }else{
    Serial.println("Hujan");
  }
}
    
```

Gambar 3.5 Pemrograman Arduino IDE

Pada dasarnya bahasa pemrograman Arduino terbagi menjadi dua bagian fungsi yaitu void setup() dan void loop(). Untuk melihat penulisan lengkap pemrograman Arduino Uno dapat dilihat di lampiran.

- d. Web Browser
 

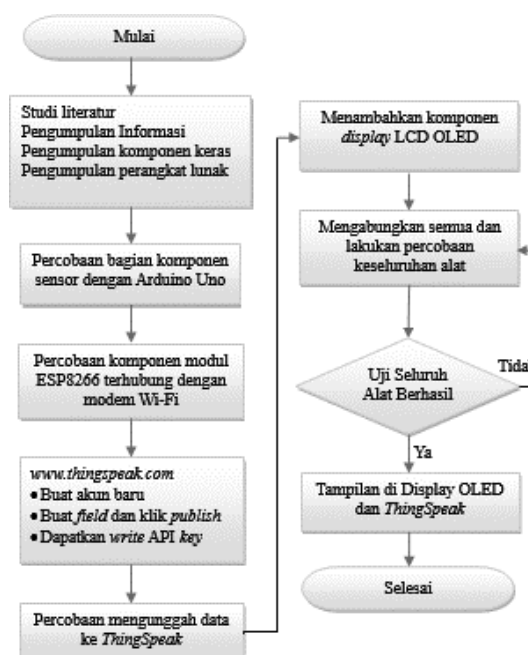
Untuk melihat hasilnya di [www.thingspeak.com](http://www.thingspeak.com), bisa dilakukan dengan *Internetexplorer* atau *googlechrome*. Banyak alternatif lain jika ingin menggunakan perangkat seperti *smarthphone* atau *tabletandroid*.

**Teknik Perancangan**

Teknik perancangan alat dilakukan dengan melakukan beberapa tahapan untuk menyelesaikan alat pemantau sensor-sensor menggunakan Arduino Uno dan ESP8266 yang diunggah ke-*serverThingSpeak*. Tahapan-tahapan tersebut antara lain adalah :

1. Studi literatur dan mengumpulkan informasi, dari mempelajari buku, teori, *datasheet*, contoh-contoh *sketch*, *libraries* dan jurnal-jurnal.
2. Mengumpulkan komponen perangkat keras dan perangkat lunak.
3. Percobaan komponen bagian sensor-Sensor dengan Arduino Uno, untuk pembacaan MQ135, MQ4, dan DHT11. Sampai diperoleh *library* dan bahasa pemrograman yang tepat.
4. Percobaan komponen bagian modul Wi-Fi ESP8266, dengan membuat bahasa pemrograman protokol AT-*Command* untuk terhubung WI-FI *aces point*, sampai diperoleh bahasa pemrograman yang efektif dan efisien.
5. Membuat akun baru di [www.thingspeak.com](http://www.thingspeak.com), tentukan *channel* baru dan atur *field chart* yang akan digunakan. Jangan lupa klik *publish*, ambil *write API key* yang nantinya dipakai dalam pemrograman Arduino Uno.
6. Buat pemrograman mengunggah data ke [www.thingspeak.com](http://www.thingspeak.com) berdasar API key yang diperoleh.
7. Menambahkan komponen OLED 128x64, tentukan *library* dan buat program penampil di LCD tersebut.
8. Percobaan seluruh rangkaian dengan menggabungkan semua komponen keras dan lunak. Dilakukan berulang kali sampai diperoleh bahasa pemrograman dengan hasil yang diinginkan.
9. Diperoleh hasil pembacaan di monitor serial, tampilan LCD OLED dan [www.thingspeak.com](http://www.thingspeak.com).

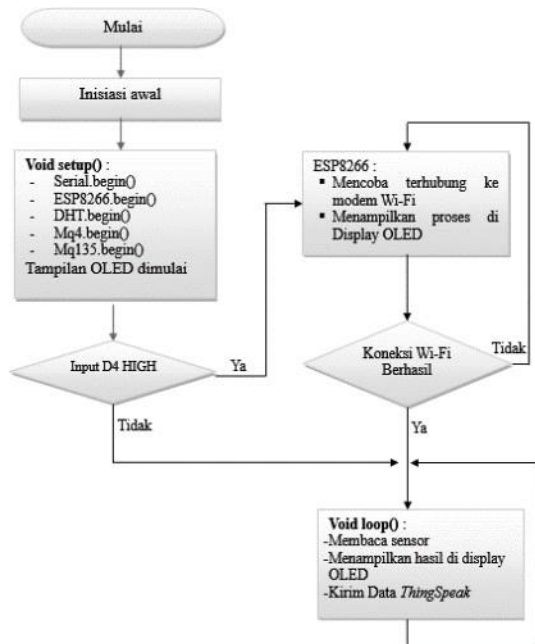
**Diagram Alir Teknik Perancangan**



Gambar 3.6 Diagram Alir Teknik Perancangan



Diagram Alir Program Arduino Uno



Gambar 3.7 Diagram Alir Program Arduino Uno

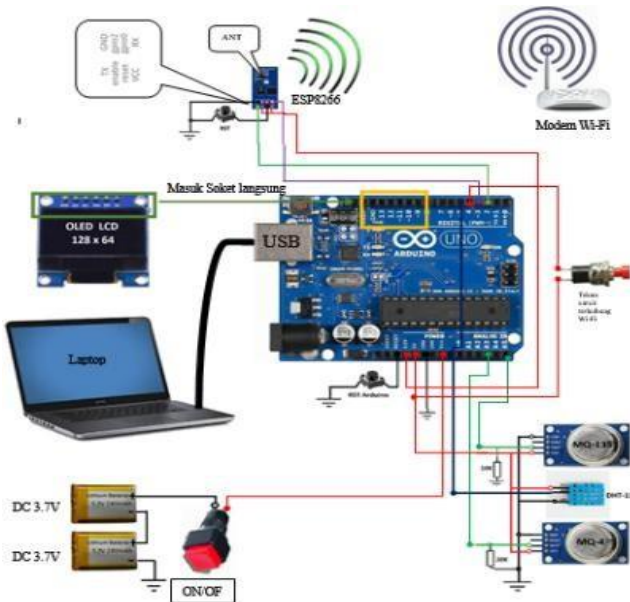


Gambar 4.2 Tampilan Keseluruhan

IV. HASIL DAN PEMBAHASAN

Rangkaian Modul Dan Pengabelan

Agar rangkaian dapat bekerja dengan sempurna maka semua modul harus digabungkan, lihat Gambar 4.1 dibawah. Dari sini dapat diperoleh gambaran utuh dan jelas tentang susunan modul, yang kemudian dapat dilakukan pengujian per blok sesuai pembahasan di bab-bab sebelumnya.



Gambar 4.1 Rangkaian Pengkabelan

Prinsip Kerja

Selama uji coba, alat ini selalu terhubung dengan laptop melalui USB. USB selain sebagai catu daya 5V, berfungsi utama sebagai kabel serial untuk komunikasi antara laptop dengan Arduino Uno. Dari sini fokus pengerjaan dan uji coba lebih kepada penulisan program di Arduino IDE. Kemudian hasil dari eksekusi dapat di lihat langsung di jendela monitor serial.

Terjadi proses inisiasi di awal program dengan memanggil library, penentuan kaki pin yang digunakan, konstanta dan isi variabel. Kemudian masuk ke dalam dua struktur penting yaitu, fungsi pengaturan awal atau disebut void setup() dan fungsi berulang atau disebut void loop(). Didalam voidsetup() terdapat beberapa proses antara lain :

1. Menampilkan kata di OLED, menggunakan *oledPrint()*.
2. Memulai komunikasi serial dengan ESP8266, menggunakan *Serial.begin(9600)*.
3. Memulai sensor DHT11, MQ4 dan MQ135, menggunakan *dht.begin()*, *mq4.begin()* dan *mq135.begin()*.
4. Fungsi tombol logika untuk pin D4, jika ditekan (*HIGH*) maka akan proses koneksi ke Wi-Fi melalui ESP8266. Jika tidak ditekan (*LOW*) maka proses untuk ESP8266 dilewati.

Selanjutnya masuk fungsi pengulangan *voidloop()* yang merupakan proses utama alat bekerja. Urutan proses kerja adalah sebagai berikut :

1. Pembacaan data-data sensor.
2. Konversi data pembacaan sensor menjadi variabel *string*.
3. Data-data *string* hasil pembacaan sensor tersebut kemudian di tampilkan di *display* LCD OLED.
4. Proses konektivitas dengan *www.thingspeak.com*, dengan pengiriman data-data ke *ThingSpeak*.

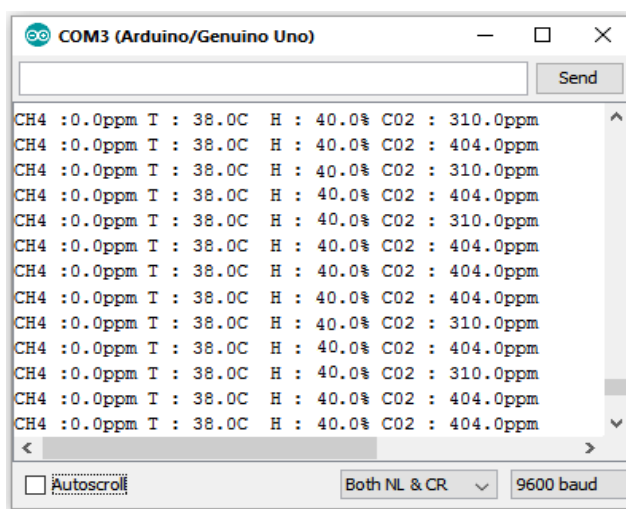
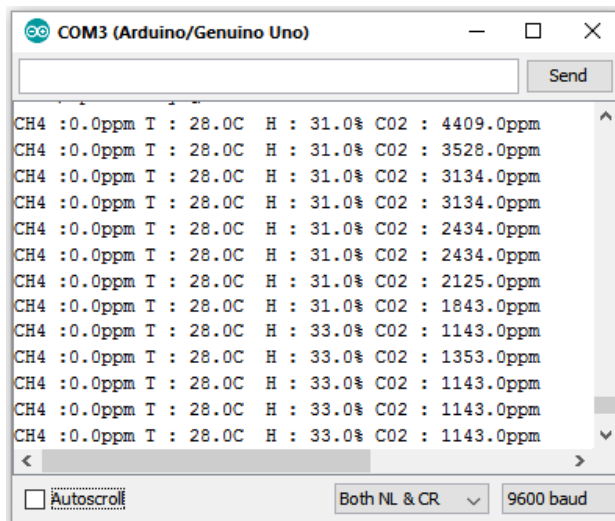
5. Dan seterusnya proses akan berulang.

Terdapat fungsi-fungsi tambahan yang berada diluar dua struktur tersebut diatas. Fungsi-fungsi tambahan itu untuk mempermudah penulisan program, diantaranya adalah fungsi penulisan kata di OLED dan fungsi-fungsi untuk terhubung Wi-Fi.

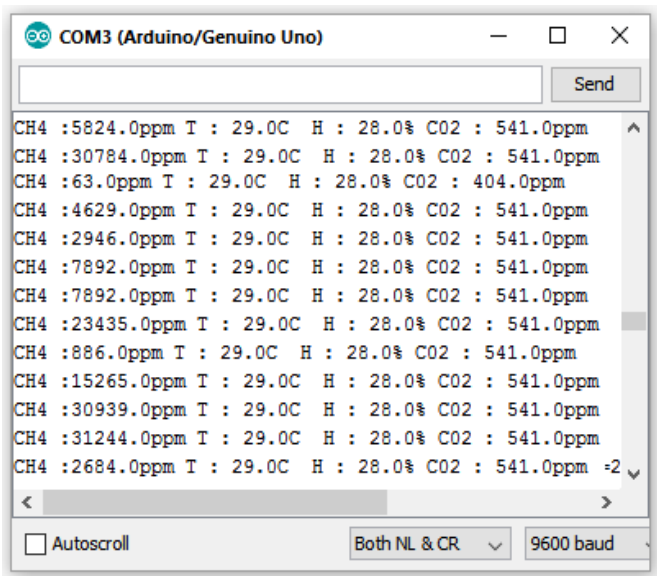
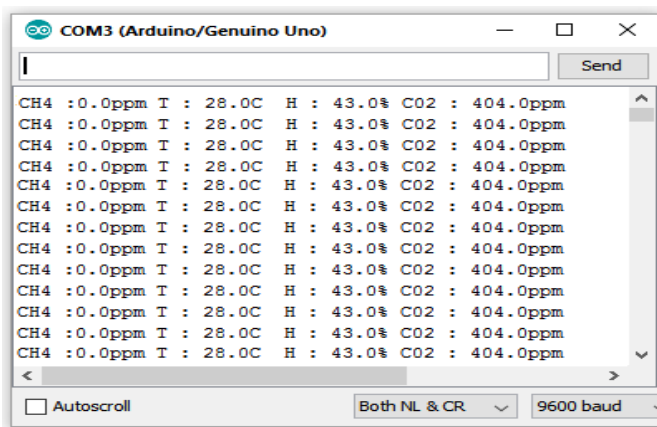
**Pengujian Sensor**

Untuk pengujian sensor, komponen utama yang digunakan berupa sensor-sensor dan Arduino Uno. Catu daya sensor diambil dari Arduino Uno pin 5V dan *output* sensor masuk ke pin analog Arduino Uno, *output* MQ4 masuk ke pin A3, *output* DHT11 masuk ke pin D5 dan *output* MQ135 masuk ke pin A5. Untuk lebih jelasnya dapat dilihat di Gamber 4.1 diatas. Diperlukan waktu sekitar 3 menit minimum untuk memanaskan filamen di sensor MQ.

Hasil pembacaan diperoleh saat masuk ke fungsi *voidloop()*, pembacaan dapat dilihat dari monitor serial Arduino IDE. Dari hasilnya bisa diteruskan tampil di LCD OLED dan dikirim ke *ThingSpeak*. Namun disini kita hanya akan melihat khusus di area monitor serial Arduino IDE. Lihat Gamber 4.3 dibawah, hasil pembacaan berulang semua sensor yang ada di dalam *voidloop()*.



Gambar 4.3 Hasil Uji Sensor



Dari Gamber 4.3 diatas T mewakili suhu, H mewakili kelembapan, CO2 dan CH4 mewakili konsentrasi gas. Perubahan dapat dilihat dengan coba memanipulasi sensor dengan panas solder untuk melihat perubahan suhu, melihat perubahan kelembapan diwaktu pagi dan siang. Sedangkan untuk manipulasi sensor gas CH4 kita gunakan tabung gas butana dibantu kipas untuk manipulasi kondisi kemudian untuk CO2 menggunakan asap hasil pembakaran. Konsentrasi gas fisik yang sebenarnya tidak bisa didapatkan karena keterbatasan alat kalibrasi sebagai referensi, jadi penulis hanya menguji hasil pembacaan dan perubahan data.

Sensor DHT11 dengan keluaran digital terhubung ke pin D5, sensor ini memiliki data terbaca untuk suhu dan kelembapan yang baik. Dibandingkan dengan termometer infrared maka diperoleh selisih suhu sekitar ± 2-3 °C. Reaksi sensor cenderung lambat seperti respon termometer manual air raksa.

Sensor MQ135 yang terhubung dengan pin A5,

merespon cukup peka bukan hanya gas karbondioksida, sensor ini dapat juga membaca gas metana, amonia, nitrogen oksida, bensin, alkohol dan butana. Dengan karakteristiknya yang umum tersebut maka sering dipakai untuk sensor kualitas udara. Sensor ini memang terlalu umum, sehingga diperlukan sensor lain yang lebih sempit spektrumnya untuk karbondioksida, meskipun ada sensor seri MQ7 yang bisa membaca karbon monoksida (CO). Sementara untuk seri sensor MQ hanya MQ135 yang menyebutkan pembacaan karbondioksida (CO<sub>2</sub>).

Sensor MQ4 yang terhubung dengan pin A3, memiliki respon yang kurang peka dibandingkan dengan MQ135. Hasil pembacaan cenderung naik turun cepat, pemberian gas uji yang kurang stabil juga salah satu faktor, sensor ini lebih tepat ditempatkan di sistem tertutup dengan konsentrasi gas metana yang stabil, kurang tepat dipakai di ruang terbuka. Pembacaan data terkadang diperoleh 0 PPM, meski sensor MQ4 mendeteksi gas, terlihat jelas dari indikator lampu LED menyala di modul sensor atau dari *analogRead*(A3). Hal ini dikarenakan kelemahan *library* atau pembacaan melebihi jangkauan sensor, untuk mensiasatinya cukup dilakukan pembacaan ulang saja.

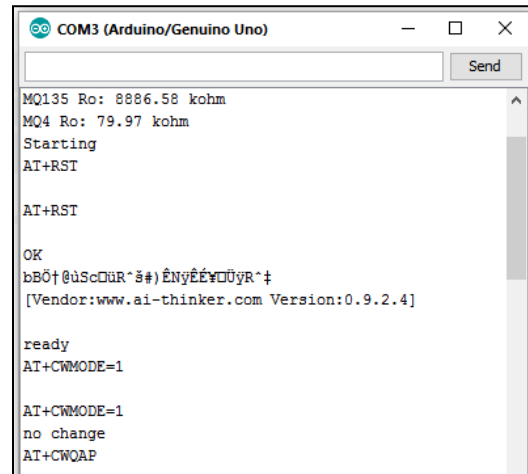
Interferensi atau gangguan dapat terjadi antara dua sensor analog, ini adalah salah satu kelemahan Arduino Uno karena memiliki satu ADC sehingga harus bergantian (*multiplexed*). Ada yang berpendapat karena impedansi yang tinggi dari keluaran sensor ikut mempengaruhi kestabilan tegangan keluaran. Saat MQ4 diuji mencapai pembacaan tinggi, MQ135 juga mengalami kenaikan. Untuk mensiasatinya maka dibuat pengaturan waktu tunda yang tepat saat membaca sensor-sensor. Kemudian antisipasi juga ditambahkan satu pin analog input yang terbaca kosong sebagai penyegaran atau *refresh* di pin A1 yang terhubung GND.

Penting untuk mengurutkan program yang tepat agar tidak tumpang tindih, sehingga ADC Arduino bekerja tepat dan efektif, pengulangan pembacaan sensor bisa dilakukan agar diperoleh pembacaan yang stabil. Pengaturan Ro dan RL di dalam *library* dilakukan diawal, setelah kita peroleh pembacaan Ro saat sensor begini didalam *voidsetup()*. Secara teori penentuan Ro dan RL merupakan proses kalibrasi yang bersifat teori berdasar formula didalam *library* yang dipakai, dan bukan kalibrasi menggunakan referensi alat kalibrasi pengukuran fisik tingkat konsentrasi. Nilai RL ditentukan sesuai dengan *datasheet* untuk MQ4 minimal 20Kohm dan MQ135 dipakai 10Kohm.

### Pengujian Modul ESP8266 Dan Thingspeak

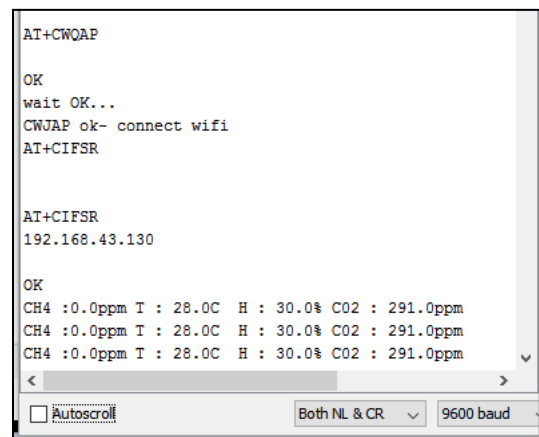
Untuk pengujian komponen ESP8266 digabungkan dengan pengujian sebelumnya dengan sensor-sensor dan Arduino Uno. Proses ada di *void setup()* untuk menghubungkan Arduino Uno dengan modem Wi-Fi. Kemudian didalam *voidloop()* modul melakukan komunikasi dengan *ThingSpeak* untuk pengiriman data. Modul Wi-Fi ESP8266 mendapat catu daya 3.3V dari Arduino Uno. Ada fungsi logika tombol pilihan di pin D4, jika ditekan (*HIGH*)

maka Arduino memerintahkan modul ESP8266 agar terhubung dengan *access point*, jika dibiarkan tanpa ditekan (*LOW*) maka proses perintah Arduino ESP8266 akan dilewati dan langsung masuk ke fungsi *void loop()*, namun demikian modul ESP8266 masih mencoba terhubung dengan pengaturan Wi-Fi sebelumnya.



```

COM3 (Arduino/Genuino Uno)
MQ135 Ro: 8886.58 kohm
MQ4 Ro: 79.97 kohm
Starting
AT+RST
AT+RST
OK
bBÖ†@úScÖUR~š#) ÈNyÉÉ#DÜyR†
[Vendor:www.ai-thinker.com Version:0.9.2.4]
ready
AT+CWMODE=1
AT+CWMODE=1
no change
AT+CWQAP
  
```



```

AT+CWQAP
OK
wait OK...
CWJAP ok- connect wifi
AT+CIFSR
AT+CIFSR
192.168.43.130
OK
CH4 :0.0ppm T : 28.0C H : 30.0% CO2 : 291.0ppm
CH4 :0.0ppm T : 28.0C H : 30.0% CO2 : 291.0ppm
CH4 :0.0ppm T : 28.0C H : 30.0% CO2 : 291.0ppm
  
```

Gambar 4.4 Koneksi Wi-Fi

Didalam *void setup()*, jika D4 kondisi *HIGH* maka ESP8266 diperintahkan untuk terhubung dengan *accesspoint* (*Speedy*) dengan *password* (1234567890). Dapat dilihat hasil pengujian pada Gambar 4.4 diatas berupa urutan yang terjadi diambil dari monitor serial.

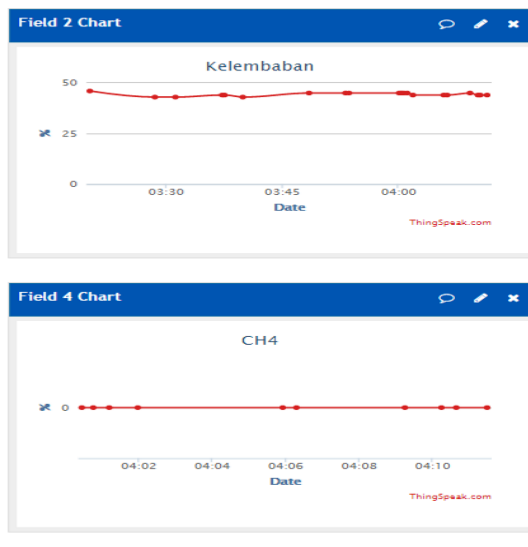
Didalam fungsi *void loop()*, terjadi proses koneksi TCP untuk pengiriman data menuju *www.thingspeak.com*.

Proses tersebut terjadi setelah semua data variabel *float* di konversi menjadi *string* kemudian, karena protokol yang diminta adalah data *string*. Arduino mengirimkan data berulang dengan waktu tunda yang kurang dari 3 detik, lihat Gambar 4.5 dibawah. Namun data yang ter-update disistem *ThingSpeak* tercepat sekitar 18 detik per data, hal ini terjadi karena ada proses panjang melalui jaringan internet. Sehingga terdapat pembacaan *UPDATE TO THINGSPEAK: Error*, hal ini tidak menjadi masalah karena disaat tertentu data terkirim juga setiap 18 detik dengan komentar *UPDATE TO THINGSPEAK OK*. Untuk menghemat jumlah data terkirim ke *ThingSpeak* waktu tunda bisa diatur sesuai kebutuhan. Hasil uji di [www.thingspeak.com/channels/102986](http://www.thingspeak.com/channels/102986), diperlihatkan pada Gambar 4.6 dibawah.

```

> UPDATE TO THINGSPEAK: OK
CH4 :0.0ppm T : 30.0C H : 33.0% CO2 : 404.0ppm
CH4 :0.0ppm T : 30.0C H : 33.0% CO2 : 404.0ppm
CH4 :0.0ppm T : 30.0C H : 33.0% CO2 : 404.0ppm
CH4 :0.0ppm T : 30.0C H : 33.0% CO2 : 404.0ppm
CH4 :0.0ppm T : 30.0C H : 33.0% CO2 : 404.0ppm
CH4 :0.0ppm T : 30.0C H : 33.0% CO2 : 404.0ppm
CH4 :0.0ppm T : 30.0C H : 33.0% CO2 : 404.0ppm
CH4 :0.0ppm T : 30.0C H : 33.0% CO2 : 404.0ppm
CH4 :0.0ppm T : 30.0C H : 33.0% CO2 : 404.0ppm
CH4 :0.0ppm T : 30.0C H : 33.0% CO2 : 404.0ppm
TCP link : OK
GET /update?key=QWZJCF7WF2PHMXY&field1=30.0&field2=3
> UPDATE TO THINGSPEAK: OK
CH4 :0.0ppm T : 30.0C H : 33.0% CO2 : 301.0ppm
CH4 :0.0ppm T : 30.0C H : 33.0% CO2 : 404.0ppm
CH4 :0.0ppm T : 30.0C H : 33.0% CO2 : 301.0ppm
CH4 :0.0ppm T : 30.0C H : 33.0% CO2 : 404.0ppm
CH4 :0.0ppm T : 30.0C H : 33.0% CO2 : 301.0ppm
CH4 :0.0ppm T : 30.0C H : 33.0% CO2 : 404.0ppm
CH4 :0.0ppm T : 30.0C H : 33.0% CO2 : 301.0ppm
CH4 :0.0ppm T : 30.0C H : 33.0% CO2 : 404.0ppm
CH4 :0.0ppm T : 30.0C H : 33.0% CO2 : 301.0ppm
CH4 :0.0ppm T : 30.0C H : 33.0% CO2 : 404.0ppm
TCP link : OK
GET /update?key=QWZJCF7WF2PHMXY&field1=30.0&field2=3
> UPDATE TO THINGSPEAK: Error
    
```

Gambar 4.5 Proses Pengiriman Data

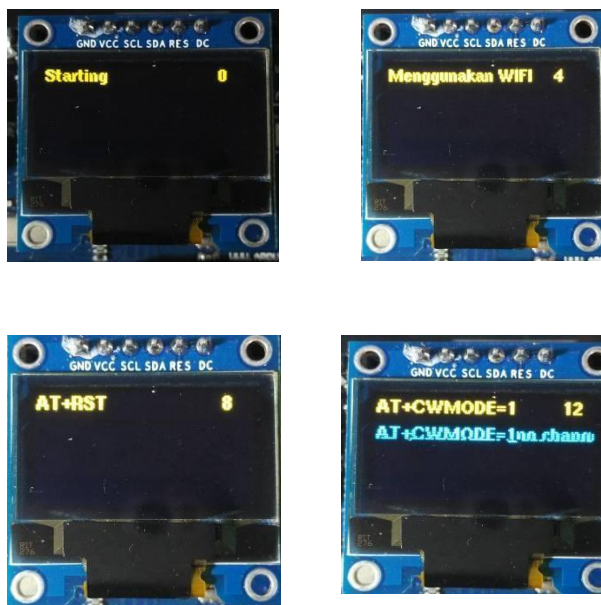


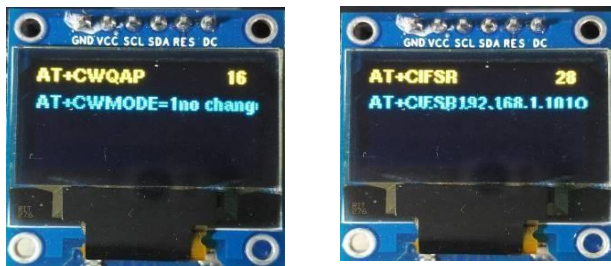
Gambar 4.6 Tampilan di *ThingSpeak*

### Pengujian Display LCD OLED

LCD OLED sebenarnya merupakan tambahan di akhir proyek, untuk melengkapi alat agar lebih sempurna. Tujuannya untuk memudahkan melihat proses dan hasil pembacaan secara lokal, jika tidak terdapat internet (*offline*), tidak ada modem Wi-Fidan tidak ada laptop untuk melihat hasil data di monitor serial. Terdapat dua proses tampilan, yaitu tampilan di dalam *voidsetup()* dan tampilan di dalam *void loop()*.

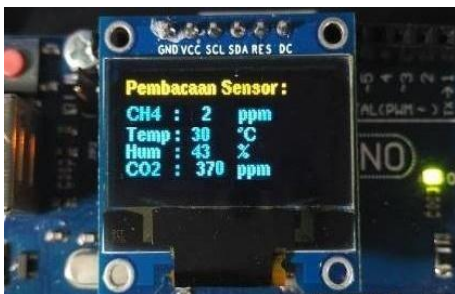
Untuk *voidsetup()*, OLED akan menunjukkan bagaimana proses yang terjadi saat ESP266 mencoba terhubung dengan modem wifi, diperlukan waktu tercepat sekitar 40 detik untuk terkoneksi dengan modem Wi-Fi. Dapat dilihat pada Gambar 4.7 dibawah hasil uji bagaimana proses Wi-Fi terbangun.





Gambar 4.7 Proses Koneksi Wi-Fi

Selanjutnya didalam `voidsetup()`, terjadi proses menampilkan data-data terbaca kedalam LCD OLED, ada empat parameter yang ditampilkan yaitu suhu kelembapan, konsentrasi gas karbondioksida dan konsentrasi gas metana. Dapat dilihat Gambar 4.8 hasil uji tampilan OLED untuk fungsi `voidloop()`.



Gambar 4.8 Tampilan Data Pembacaan

### Pengujian Keseluruhan Perangkat

Pengujian dilakukan dengan rangkaian keseluruhan lengkap, baik itu dengan catu daya DC eksternal, catu daya USB maupun menggunakan baterai lithium. Pengujian dengan kondisi tidak terhubung Wi-Fi dan terhubung Wi-Fi, dengan melihat hasil tampilan akhir di [www.thingspeak.com](http://www.thingspeak.com).

Seluruh bagian digabungkan menjadi satu kesatuan, kemudian semua bahasa program digabung sehingga sesuai dengan urutan proses diatas. Konsumsi daya rangkaian ini terbaca antara 300 sampai 390 mA.

Pada *booting* awal tombol set untuk pin D4 tidak ditekan (*LOW*) maka fungsi penyambungan Wi-Fi akan dilewati, dan langsung menuju pembacaan sensor di `voidloop()`. Jika tombol set ditekan maka kondisi pin D4 akan *HIGH*, akan terhubung modem Wi-Fi. Jika berhasil dengan Wi-Fi maka proses selanjutnya adalah `voidloop()` pembacaan sensor dan pengiriman data ke *ThingSpeak*. Melihat hasil uji di Gambar 4.5 diatas, monitor serial memperlihatkan

pembacaan sensor 8 kali pengulangan, dilanjutkan dengan pengunggahan data ke *ThingSpeak*, proses tersebut berulang di dalam `voidloop()`.

Perlu dicatat meski proses koneksi Wi-Fi terlewat, namun secara otomatis dan mandiri ESP8266 masih bisa terhubung dengan Wi-Fi dari *setting-an* awal sebelumnya. Jadi meski tombol untuk D4 tidak ditekan, pada akhirnya akan terhubung juga dengan internet, jika masih tersedia fasilitas modem Wi-Fi yang sesuai dengan pengaturan di program Arduino. Dengan bantuan LCD OLED pembacaan proses koneksi Wi-Fi dan data pengukuran menjadi lebih mudah tampil, terutama jika tidak ada laptop pendukung untuk melihat di monitor serial.

### Analisa Dan Hasil Pengamatan

Konsumsi memori untuk *sketch* yang dipakai oleh arduino bisa dilihat setelah selesai proses *upload* program ke board besarnya sekitar 22,566 bytes yaitu 70% data total memori 32,256 bytes. Fungsi tombol untuk D4 hanya untuk memberi pilihan langsung ke pembacaan `voidloop()`, melewati proses Wi-Fi dari Arduino namun ESP8266 masih bisa mandiri terhubung dengan Wi-Fi.

Catu daya untuk ESP8266 menggunakan pin 3.3 V dan untuk sensor menggunakan catu daya 5 V. Rangkaian dapat membaca data dan menampilkan di display LCD OLED selain dapat terhubung juga dengan Wi-Fi internet. Jika sudah terhubung internet maka data-data pembacaan dapat diunggah di [www.thingspeak.com](http://www.thingspeak.com), dapat diakses dari mana saja kapan saja, dan siapa saja. Di dalam *ThingSpeak* data tersebut tersimpan sebagai database, menjadi sumber data yang dapat di download, untuk kepentingan riset atau perencanaan lainnya.

Jumlah data yang dapat di terima *ThingSpeak* 240 pengukuran dalam satu jam, jadi berkisar 15 detik jarak antara titik pengukuran. Yang penulis peroleh adalah 18 detik jarak antar data yang terkirim. Sementara data yang dapat diunduh 8000 titik dalam sekali *download*.

Pembacaan keseluruhan sensor tertampil di LCD OLED dan monitor serial adalah 8 kali, kemudian diteruskan dengan *upload* ke *ThingSpeak*. Pembacaan 2 sensor analog dapat mengalami interferensi khususnya disaat membaca CH4 yang menyebabkan kenaikan di CO2, hal ini diketahui diawal pengujian alat. Masalah ini umum terjadi disaat satu ADC bargantian bekerja untuk multi sensor analog. Melihat penjabaran solusi di sub bab pengujian sensor diatas, kita melihat terdapat ruang untuk melakukan pengaturan. Dengan mencoba pengaturan sinkronisasi waktu tunda disetiap pembacaan sensor analog, megurutkan proses pembacaan dengan benar, maka dapat diperoleh pembacaan ideal yang kita inginkan. Untuk mengatasi pembacaan 0 PPM di sensor CH4 cukup dilakukan pengulangan pembacaan sampai terbaca konsentrasi yang benar, hal ini akan mejadikan pembacaan CH4 lebih lambat. Sensor MQ135 memiliki respon lambat saat membaca naik dan turun, sedangkan sensor MQ4 cenderung cepat membaca naik dan cepat turunnya, hal ini dikarenakan gas tersebut terbakar oleh filament yang panas,

tidak konsistensinya aliran gas dengan konsentrasi yang stabil dan minimnya peralatan lab gas. Pembacaan konsentrasi gas tidak dapat dibandingkan dengan nilai sebenarnya karena tidak ada alat pendukung untuk referensi. Untuk sensor DHT11 dengan keluaran digital memiliki respon lambat seperti termometer air raksa yang lambat begitu juga dengan kelembapannya.

#### Uji Coba Alat di TPA Manggar, Balikpapan

Tujuan penulis melakukan uji alat di lokasi TPA dengan harapan dapat memperoleh hasil pengukuran, kemudian membandingkan hasilnya dengan alat ukur gas metana yang dimiliki TPA Manggar. Namun dari kunjungan awal tanggal 18 Juli 2016, penulis mendapati bahwa TPA Manggar ternyata belum memiliki alat ukur gas metana, sehingga yang penulis dapat lakukan adalah menguji coba alat itu sendiri dan mengambil data pengukuran. Berikut adalah gambar lokasi pengujian alat di TPA Manggar, teras ruang generator.



Gambar 4.9 Pengujian alat mengukur kandungan gas dilokasi TPA Manggar, Balikpapan

Pada kunjungan ke-3 dan ke-4 di bulan Agustus 2016, penulis sedikit merubah bahasa program dibagian cara pembacaan sensor, penambahan pembacaan *analogRead* dan komentar koneksi *ThingSpeak* yang lebih sederhana agar lebih mudah dipahami pembacaannya di monitor.

Terbaca fluktuatif untuk CH<sub>4</sub>, dan sempat menyentuh angka sekitar 30,000 ppm. Tidak terjadi interferensi pembacaan di CO<sub>2</sub>, arah angin dari samping sensor sehingga gas metana tidak mengenai sensor CO<sub>2</sub>. Jika kedua sensor terpapar gas metana maka keduanya menampilkan hasil pengukuran mendekati sama, meski hasil pengukuran belum bisa dikonfirmasi karena tidak adanya alat ukur sebagai referensi di TPA Manggar.

Untuk koneksi internet di area TPA sangat lemah mengandalkan koneksi Wi-Fi *Theatering* dari sinyal GPRS, sehingga data terkirim ke ThingSpeak kurang optimal. Maka untuk mendapatkan data yang lengkap bisa diambil dari pembacaan di monitor serial dan di simpan dalam *notepad* seperti terlihat pada gambar dibawah, untuk informasi tambahan dapat dilihat di lampiran.

```

data3 - Notepad
File Edit Format View Help
CH4:15424ppm T:31°C H:24% 28C02:363ppm TCP-ERROR,
23,CH4:4271ppm T:31°C H:24% 28C02:363ppm TCP-ERROR,
24,CH4:32078ppm T:31°C H:24% 28C02:363ppm TCP-ERROR,
25, 25, 8197
CH4:8197ppm T:31°C H:23% 28C02:363ppm TCP-ERROR,
25,CH4:14547ppm T:31°C H:23% 29C02:363ppm TCP-ERROR,
28, 28, 9866
CH4:9866ppm T:31°C H:23% 29C02:363ppm TCP-ERROR,
30, 30, 1783
CH4:1783ppm T:31°C H:36% 29C02:363ppm TCP-ERROR,
33,CH4:21275ppm T:31°C H:36% 28C02:363ppm TCP-ERROR,
32,CH4:11269ppm T:31°C H:36% 28C02:363ppm TCP-ERROR,
31,CH4:19591ppm T:31°C H:23% 28C02:363ppm TCP-ERROR,
33, 35, 13408
CH4:13408ppm T:31°C H:23% 28C02:363ppm TCP-OK, -> UPLOAD-Error
41, 41, 4921
CH4:4921ppm T:31°C H:23% 28C02:363ppm TCP-OK, -> UPLOAD-Error
45,CH4:30815ppm T:31°C H:23% 27C02:328ppm TCP-OK, -> UPLOAD-Error
50, 51, 3034
CH4:3034ppm T:31°C H:23% 27C02:363ppm TCP-OK, -> UPLOAD-OK
56,CH4:13988ppm T:31°C H:29% 28C02:363ppm TCP-OK, AT+CIPSEND-Error, UPLOAD-Error
64,CH4:8732ppm T:30°C H:24% 28C02:363ppm TCP-OK, AT+CIPSEND-Error, UPLOAD-Error
74, 74, 28848
CH4:28848ppm T:30°C H:24% 30C02:402ppm TCP-OK, -> UPLOAD-Error
79,CH4:4896ppm T:30°C H:25% 30C02:442ppm TCP-ERROR,
80, 81, 2960
CH4:2960ppm T:30°C H:25% 31C02:486ppm TCP-OK, AT+CIPSEND-Error, UPLOAD-Error
87,CH4:19072ppm T:30°C H:25% 32C02:532ppm TCP-ERROR,
89, 89, 7808
CH4:7808ppm T:30°C H:25% 32C02:581ppm TCP-OK, -> UPLOAD-Error
94, 94, 26368
CH4:26368ppm T:30°C H:25% 31C02:486ppm TCP-ERROR,
95,CH4:16896ppm T:30°C H:25% 31C02:486ppm TCP-ERROR,
96,CH4:512ppm T:30°C H:25% 31C02:486ppm TCP-ERROR,
97, 97, 31616
CH4:31616ppm T:30°C H:25% 30C02:486ppm TCP-ERROR,
98, 98, 21248
CH4:21248ppm T:30°C H:25% 30C02:442ppm TCP-ERROR,

```

Gambar 4.10 Contoh Pengambilan Data di TPA

## V. PENUTUP

### Kesimpulan

Berdasarkan pembahasan pada bab-bab sebelumnya maka dapat diambil beberapa kesimpulan sebagai berikut :

1. Rangkaian listrik yang dibuat dapat mengukur suhu, kelembapan, konsentrasi gas metana konsentrasi gas karbondioksida dan ditampilkan dilayar LCD OLED.
2. Alat ini dapat mengunggah data pembacaan ke [www.thingspeak.com](http://www.thingspeak.com) Dan menjadi sumber data yang dapat diunduh.
3. Interferensi di pembacaan 2 sensor analog bisa diatasi dengan mengurutkan program dengan benar dan waktu tunda yang tepat.
4. Parameter yang ditampilkan belum dikalibrasi secara fisik, masih sifatnya kalibrasi secara teori didalam *library*.
5. Pengiriman pesan ke *Twitter* bisa dilakukan dengan 2 cara. Cara yang termudah adalah dengan pengaturan di fasilitas *ThingTweet* dan *React* di dalam *website* ThingSpeak itu sendiri, tanpa menuliskan program di Arduino, namun hasilnya masih dibatasi dan jauh dari harapan.

### Saran

Untuk meningkatkan kinerja alat, diperlukan adanya saran agar menjadi alat yang lebih baik, berikut adalah :

1. Disarankan mencoba alternatif seri sensor lain diluar seri MQ yang memiliki konsumsi arus lebih rendah dan jangkauan konsentrasi gas yang lebih besar misalkan 0-100,000 PPM. Karena rata-rata konsentrasi gas metana sekitar 40,000 – 60,000 PPM. Gunakanlah sensor DHT22 seri diatasnya yang lebih baik dari DHT11.
2. Maksimalkan modul ESP266 sebagai lokal *werserver* misal sebagai pengatur koneksi Wi-Fi acces point dan password agar bisa diganti tanpa harus menggunakan Aduino IDE.
3. Jika masih diperoleh interferensi antar input analog, maka untuk menghindari pengaturan yang rumit dan bisa mengurangi beban Arduino Uno dengan ADC tunggal bekerja bergantian *multiplexed*, disarankan untuk menambah ADC eksternal diluar ArduinoUno.
4. Penambahan kontrol relai, lampu atau bunyi tanda konsentrasi gasyang tinggi, bisa dilakukan untuk peringatan dini.
5. Untuk pengiriman pesan *Twitter* yang lebih baik, cara kedua yaitu menuliskan programnya didalam Arduino IDE secara langsung, sehingga ThingSpeak tidak terbebani dan pesan yang terkirim lebih maksimal tanpa pembatasan.
6. Masukan dari pihak TPA Manggar, Balikpapan yaitu data yang diperoleh dari hasil pengukuran akan lebih baik jika dapat direkam secara offline menggunakan laptop atau kartu memori, sebagai *backup* data dan analisa dikemudian hari.
7. Dan juga tampilan di LCD bisa diperbesar, seperti menggunakan *dot matrix* atau *seven segment display*.

### REFERENSI

- [1] Andrianto, Heri. 2015. Pemrograman Mikrokontroler AVR Atmega16 Menggunakan Bahasa C. Bandung : Informatika Bandung.
- [2] Kadir, Abdul. 2015. Buku Pintar Pemrograman Arduino. Yogyakarta :Mediakom
- [3] Syahwil, Muhammad. Panduan Mudah Simulasi & Praktek Mikrokontroler Arduino. Yogyakarta : CV. Andi Offset.
- [4] Wikipedia (2016). Pengertian Pemanasan Global. [artikel\_ online].[https://id.wikipedia.org/wiki/Pemanasan\\_global](https://id.wikipedia.org/wiki/Pemanasan_global). Diakses tanggal 1 Maret 2016.
- [5] Wikipedia (2016). Definisi Gas Rumah Kaca. [artikel\_ online].[https://id.wikipedia.org/wiki/Gas\\_rumah\\_kaca](https://id.wikipedia.org/wiki/Gas_rumah_kaca). Diakses tanggal 1 Maret 2016.
- [6] Arduino. (2016). Deskripsi Arduino dan jenis-jenisnya. [artikel\_ online].<https://www.arduino.cc/en/Main/ArduinoBoardUno>. Diakses tanggal 6 Maret 2016.
- [7] Fatmawati, Esa. (2015). Sampah TPA Manggar Akan Dikembangkan Jadi Pembangkit Listrik [artikel\_ online]. <http://www.klikbalikpapan.co/berita-1774-sampah-tpa-manggar-akan-dikembangkan-jadi-pembangkit-listrik.html>. Diakses tanggal 1 Maret 2016.
- [8] Sudarman. (2010). Meminimalkan Daya Dukung Sampah Terhadap Pemanasan Global [artikel online]. <http://id.portalgaruda.org/>. Diakses tanggal 3 Maret 2016.
- [9] Gironi, Davide (2014), Cheap CO2 meter using the MQ135 sensor with AVR Atmega [artikel\_ online]. <http://davigeroni.blogspot.co.id/2014/01/cheap-co2-meter-using-mq135-sensor-with.html#.V1nLIvmLTIU>. Diakses Tanggal 1 April 2016