

IMPLEMENTASI WIRELESS SENSOR NETWORK UNTUK MONITORING DAN CONTROLLING GAS KARBON MONOKSIDA PADA RUANGAN TERTUTUP

Implementation Of Wireless Sensor Network For Monitoring And Controlling Carbon Monoxide in Isolated Room

Piove wiraguna¹, Dr. Ir Rendy Munadi., MT.², Unang Sunarya, ST., MT.³

¹Electrical Engineering Faculty of Telkom University, ²School of Applied Science Telkom University Bandung, Jawa Barat 40257

¹piovewiraguna@gmail.com, ²rendymunadi@telkomuniversity.co.id, ³unangsunarya@telkomuniversity.co.id

ABSTRAK

Gas karbon monoksida (CO) merupakan salah satu polutan udara yang memiliki karakteristik tidak berbau, tidak berwarna, dan tidak berasa sehingga sulit dideteksi oleh indra manusia. CO memiliki sifat racun sehingga sangat berbahaya. Manusia sulit menyadari tingkat kadar gas CO yang berada pada level berbahaya. Bila kadar gas CO berlebih pada suatu lingkungan akan mengancam kesehatan manusia hingga berujung pada kematian.

Meninjau permasalahan tersebut, perlu adanya sistem yang dapat melakukan monitoring gas CO agar dapat mengetahui kadar gas pada ruangan dan sistem *controlling* untuk mengurangi kadar gas yang berlebih. Sistem yang dibuat terdiri dari sensor MQ-7, relay, *exhaust fan*, NodeMCU dan teknologi *Wireless Sensor Network*. Sistem dimulai dengan mendeteksi gas kemudian data dikirim dan ditampilkan. Bila gas CO melebihi batas aman yaitu 500 ppm, maka *exhaust fan* akan aktif.

Untuk mengetahui kualitas jaringan yang dibuat maka perlu dilakukan pengujian dalam beberapa skenario. Berdasarkan hasil pengujian jarak jangkauan maksimum koordinator sebesar 12 meter. Ketika jarak *node* yang terhubung semakin menjauhi koordinator, *delay* cenderung akan semakin membesar dan *throughput* akan semakin mengecil. Ketika sistem dijalankan secara keseluruhan, *delay* pada aktuator lebih besar dari pada *delay* sensor *node*. Pada saat menggunakan MQTT, Sesuai standar ITU-T G114 kualitas sistem berada pada kualitas yang baik untuk MQTT dan pada HTTP sensor *node* berada pada kualitas yang baik sedangkan aktuator *node* berada pada kualitas yang buruk. Pada sistem ini penggunaan protokol MQTT lebih baik dibanding HTTP karena pada protokol MQTT hanya melakukan satu kali proses koneksi diawal sehingga membuat *delay* lebih kecil dibanding HTTP. sedangkan pada HTTP akan selalu melakukan proses koneksi setiap data ingin dikirim lalu menutup koneksi ketika data telah selesai dikirim sehingga membuat *delay* HTTP lebih besar. *Throughput* pada MQTT lebih kecil dibanding *throughput* pada HTTP

Kata Kunci: Wireless Sensor Network, karbon monoksida, monitoring dan controlling

ABSTRACT

Carbon monoxide (CO) is one of the air pollutants that has characteristic odorless, colorless, and tasteless CO difficult to be detected by the human senses. Co is toxic air so it is very dangerous. When CO levels in a closed room is over, will threatening health and lead to death. The victim of poisoning does not feel poisoned but only feeling tired, weak and want to rest until death.

Reviewing the problem, human need a system that can monitor the CO levels in the room and controlling system to reduce CO. The system consists of MQ-7 sensor, relay, exhaust fan, NodeMCU and Wireless sensor network technology. The system starts with CO detection and then sent data and display data. When the CO exceeds the safe limit of 500 ppm, the exhaust fan will be active.

To determine the quality of the network made it is necessary to test in several scenarios. Based on the result of testing the range of the coordinator's maximum range of 12 meters. As the distance of the connected node farther away from the coordinator, the delay tends to increase and the throughput diminishes.. On overall system, the delay in the actuator is greater than the sensor node delay. When using MQTT, according to the ITU-T G114 standard the quality of the system in good quality for the MQTT and on the HTTP, node sensor in acceptance quality and actuator node in bad quality. In this system, using the MQTT protocol is better during HTTP because in the MQTT protocol only one time the connection process. That is stored makes the delay smaller than HTTP. While on HTTP it will always make the connection process every packet will to send and closing connection if sending packet finished. That is make large delay in HTTP. Throughput in the MQTT is less likely throughput in HTTP

Keywords: Wireless Sensor Network, Carbon Monoxide, monitoring and controlling

1. Pendahuluan

Perkembangan teknologi *Wireless Sensor Network* (WSN) begitu pesat dan dapat di implementasikan dalam berbagai bidang. Contohnya pertanian, *smarthome*, industri, lingkungan dan masih banyak lagi. Implementasi WSN dilakukan untuk kebutuhan *monitoring*, *controlling*, dan *tracking*. Kadar gas dalam suatu lingkungan menjadi salah satu konsentrasi bidang yang diimplementasi oleh WSN. Gas yang berada di lingkungan mampu mempengaruhi keadaan manusia hingga dapat menyebabkan keracunan dan kematian bila terdapat gas berbahaya.

Salah satu gas yang berbahaya adalah karbon monoksida (CO). Gas ini memiliki karakteristik tidak berwarna, tidak berbau dan tidak berasa sehingga sulit dideteksi indra manusia. Gas CO yang masuk kedalam tubuh mampu mengikat darah 200 kali lebih kuat dibanding oksigen sehingga peran oksigen tergantikan. Efek dari hal tersebut adalah korban akan merasa pusing, mual, konsentrasi menurun hingga berujung pada kematian. Pada kasus keracunan, banyak korban tidak menyadari kadar gas CO telah pada level berbahaya dan merasakan gejala keracunan sebagai kelelahan dan sakit biasa. Kasus keracunan akibat paparan gas CO tercatat sebanyak 40.000 kasus gawat darurat dan 5.000 sampai dengan 6.000 kematian per tahun di Amerika. [1] Di Indonesia terdapat 81 kasus keracunan gas CO dan 16 orang diantaranya meninggal dunia pada bulan Januari – September tahun 2014. [2]. Penelitian terkait dalam penelitian ini antara lain:

1. “*Indoor Air Quality Monitoring Using Wireless Sensor Networks*” yang dilakukan oleh Sayantiani Bhattacharya, Sridevi S, dan Pitchiah R pada tahun 2012 [3] telah dibuat rancangan jaringan WSN untuk memonitoring kadar gas CO pada ruangan dan mampu mengontrol ventilasi udara. Pada penelitian terdapat 5 ruangan yang diamati. Masing masing ruangan dipasang sensor gas CO dan setiap sensor terhubung dengan menggunakan 1 koordinator Zigbee.
2. “*Industrial Air Pollution Monitoring System Based on Wireless sensor network*” oleh Daudi S Simbeye tahun 2017 [4]. Dalam penelitian tersebut dibuat jaringan WSN menggunakan LoRa network untuk memonitoring gas - gas berbahaya pada industri. Terdapat 2 sensor *node* pada zona area yang berbeda. Hasil yang didapat kadar emisi pada zona 1 lebih tinggi dibanding zona 2.
3. “*Comparison with HTTP and MQTT on required Network Resources for IOT*” oleh Tetsuya Yokotani, dan Yuya Sasaki tahun 2016 [5] membahas mengenai perbandingan protokol MQTT dan HTTP pada *Wireless sensor network*. Parameter perbandingan yang digunakan adalah *overhead*, *payload size*, dan kapasitas *resource server* yang dibutuhkan. Kesimpulan yang didapat adalah MQTT lebih baik dibandingkan HTTP.
4. “Rancang Bangun Aplikasi Monitoring Sebagai informasi Gas Karbon Monoksida pada Jaringan Sensor Nirkabel” oleh Ahmad Alfian Ilmi tahun 2014 [6] membuat sistem monitoring gas CO yang nirkabel dalam ruangan menggunakan X-Bee sebagai media transmisi nirkabel. Alat monitoring ini mempunyai empat titik sensor *node*. Jarak maksimal yang dapat dijangkau untuk pengiriman data pada masing-masing *node* adalah 50m. Sedangkan jarak maksimal yang dapat dijangkau oleh keseluruhan sistem adalah 100m.

2. Dasar Teori

2.2 Karbon Monoksida

Karbon monoksida (CO) adalah salah satu jenis polutan gas yang ada di lingkungan. Gas CO berasal dari hasil pembakaran yang tidak sempurna seperti pada proses pembakaran pada kendaraan bermotor, penggunaan kompor dalam proses memasak, pembakaran briket untuk penghangat ruangan, hasil pembakaran mesin industri, konsumsi rokok dan masih banyak lagi.

Gas CO memiliki karakteristik yang tidak berbau, tidak berwarna, dan tidak berasa sehingga sangat sulit dikenali oleh indra manusia [7]. Manusia tidak akan sadar bahwa dirinya telah keracunan gas CO dan lambat laun akan mengalami kematian tanpa rasa sakit. Saat mengalami keracunan, gas CO dalam tubuh akan menggantikan posisi oksigen untuk mengikat hemoglobin dalam darah sehingga membentuk karboksihemoglobin kemudian mengalir keseluruh organ termasuk jantung, otak, dan organ vital lainnya.

Batas paparan karbon monoksida yang diperbolehkan oleh OSHA (*Occupational Safety and Health Administration*) adalah 35 ppm untuk waktu 8 jam/hari kerja, sedangkan yang diperbolehkan oleh ACGIH TLV-TWV adalah 25 ppm untuk waktu 8 jam. Kadar yang dianggap langsung berbahaya terhadap kehidupan atau kesehatan adalah 1.500 ppm (0,15%). Paparan dari 1.000 ppm (0,1%) selama beberapa menit dapat menyebabkan 50% kejenuhan dari karboksi hemoglobin dan dapat berakibat fatal [8]

2.3 Wireless Sensor Network

Wireless Sensor Network (WSN) adalah suatu sistem jaringan nirkabel yang terdiri dari beberapa sensor *node*. WSN digunakan untuk proses pemantauan dan pengendalian terhadap objek tertentu.

WSN dapat diaplikasikan dalam kegiatan manusia sehari - hari, baik kegiatan produksi maupun kegiatan konsumsi seperti *smarthome*, pertanian, lalu lintas jalan, kebutuhan industri, monitoring lingkungan, dll. Dalam pengaplikasiannya, WSN harus mampu beroperasi dalam jumlah *node* yang banyak mampu aktif dalam jangka waktu yang lama dan mampu mengirim data secara *realtime* sesuai kebutuhan. Sehingga terjadi permasalahan pada kecepatan proses pengiriman, kapasitas penyimpanan, konsumsi daya, dan *bandwith*. Beberapa hal untuk membangun WSN terdiri dari mikroprosesor, penyimpanan data, sensor, *Analog Digital Converter* (ADC), *transceiver* data, sistem kontrol dan catu daya. Ketersediaan catu daya dapat berasal dari berbagai sumber.

Mikroprosesor memiliki kapasitas penyimpanan terbatas. Biasanya kurang dari 10 Kbytes RAM dan 100 Kbytes ROM. Kapasitas tersebut 10.000 kali lebih kecil dibanding penyimpanan PC [9].

2.4 Protokol MQTT

MQTT (*Message Queuing Telemetry Transport*) adalah sebuah protokol yang cocok digunakan untuk komunikasi *Internet of Thing* (IoT) dan komunikasi *Machine to Machine* (M2M) yang bekerja diatas TCP/IP, dan mempunyai ukuran paket data yang kecil sehingga mampu mengurangi penggunaan daya pada perangkat. MQTT bersifat open source dirancang untuk perangkat yang terbatas dan bandwidth rendah. MQTT bekerja dengan model *publish/subscribe* [10]. model *publish/subscribe* membutuhkan semacam agen distribusi yang disebut Broker. Publish digunakan apabila *client* ingin mengirim data ke Broker. *Subscribe* digunakan apabila *client* ingin mengambil data dari Broker.

MQTT memiliki 3 tingkatan layanan QoS yaitu level 0,1,dan 2. Pada level 0 pesan dikirim maksimal satu kali dan memiliki kemungkinan pesan tidak dikirim dan hilang. Level 0 bekerja tanpa adanya *acknowledge message* (ACK) dan broker tidak menyimpan pesan bila tidak terkirim atau saat koneksi terputus. Level 1 akan mengirim pesan minimal 1 kali dan pengirim akan menerima ACK. Pesan akan selalu dikirim bila pengirim tidak menerima atau tidak mengakui ACK. Pada level satu akan terjadi kemungkinan *receiver* akan menerima pesan yang sama dan memprosesnya beberapa kali. Pada level 2 pesan akan dijamin dapat diterima satu kali dan tidak terjadi duplikasi seperti yang terjadi pada level 1. Level 2 memiliki metode pengiriman yang paling lambat dibanding level - level yang lain. [11]

2.5 Protokol HTTP

Hypertext Transfer Protocol (HTTP) merupakan sebuah protokol yang menyediakan perintah dalam komunikasi jaringan *client* dengan web *server*. *Client* akan melakukan permintaan melalui alamat dan website yang akan di akses. *Server* menerima permintaan berdasarkan kode protokol yang diinputkan. Setiap perintah di jalankan secara independen tidak dapat terpengaruh dengan input sebelumnya.

Urutan transaksi permintaan dan respon *server* yang diberikan pada protokol HTTP disebut sesi HTTP. *Client* akan membuka koneksi *Transmission Control Protocol* (TCP) ke sebuah *port*. *Server* akan menunggu permintaan *client* melalui *port* tersebut. Ketika permintaan datang, *server* akan memberikan respon yang sudah ditetapkan.

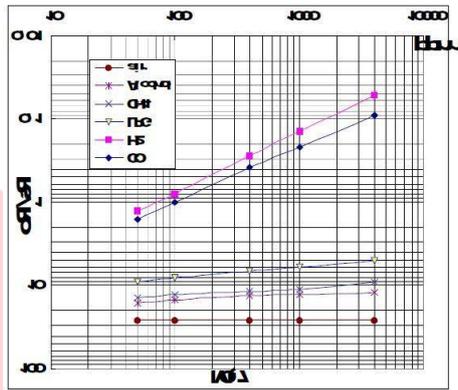
Bentuk permintaan *client* dibagi menjadi beberapa metode permintaan. Metode yang sering digunakan adalah GET, POST, dan DELETE. GET adalah metode dengan mengambil data dari web *server* dengan menentukan parameter dari url. Penggunaan GET dapat dilakukan ketika mengirim tipe data yang sensitive seperti username, password, nomor id dan digunakan untuk memperbarui nilai pada *server*. POST adalah metode pengiriman data ke *server* dengan memanfaatkan *body* pesan. DELETE digunakan untuk menghapus data dari *server*.

2.6 Sensor MQ-7

MQ-7 adalah salah satu jenis sensor pendeteksi gas. Sensor MQ-7 mampu mendeteksi gas karbon monoksida (CO) yang berada di lingkungan. MQ-7 memiliki sensitivitas yang tinggi, stabil,dan memiliki umur yang panjang. Agar sensor berjalan baik sensor harus dipanaskan dalam waktu tertentu dengan menggunakan catu daya *heater* 3V AC/DC dan selain itu menggunakan catudaya rangkaian sebesar 5V DC. Sensor MQ-7 mampu melakukan pengukuran gas CO diantara 20-2000 ppm. [12]



Gambar 2. 1Modul Sensor MQ-7



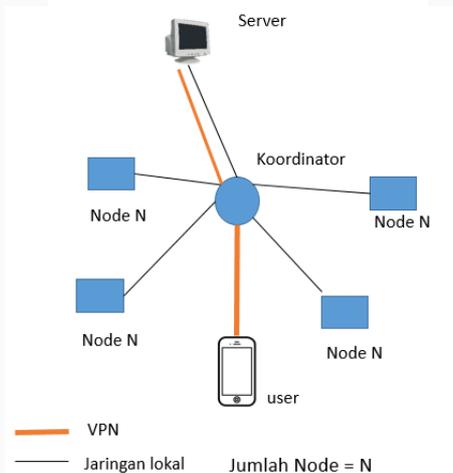
Gambar 2. 2 Grafik Karakteristik Sensitivitas MQ-7 [12]

Hubungan antara R_s/R_o terhadap nilai ppm digambarkan pada gambar 2.2. R_o bernilai tetap dan R_s bernilai berubah ubah sesuai gas CO di lingkungan. Bila kadar gas CO tinggi maka nilai hambatan R_s semakin kecil. Pada gambar 2.2 dapat terlihat bila kadar CO 100 ppm maka nilai R_s/R_o adalah 1. Artinya nilai R_s dan R_o sama saat kadar gas CO sebesar 100 ppm.

3. Perancangan Sistem

3.1 Gambaran umum

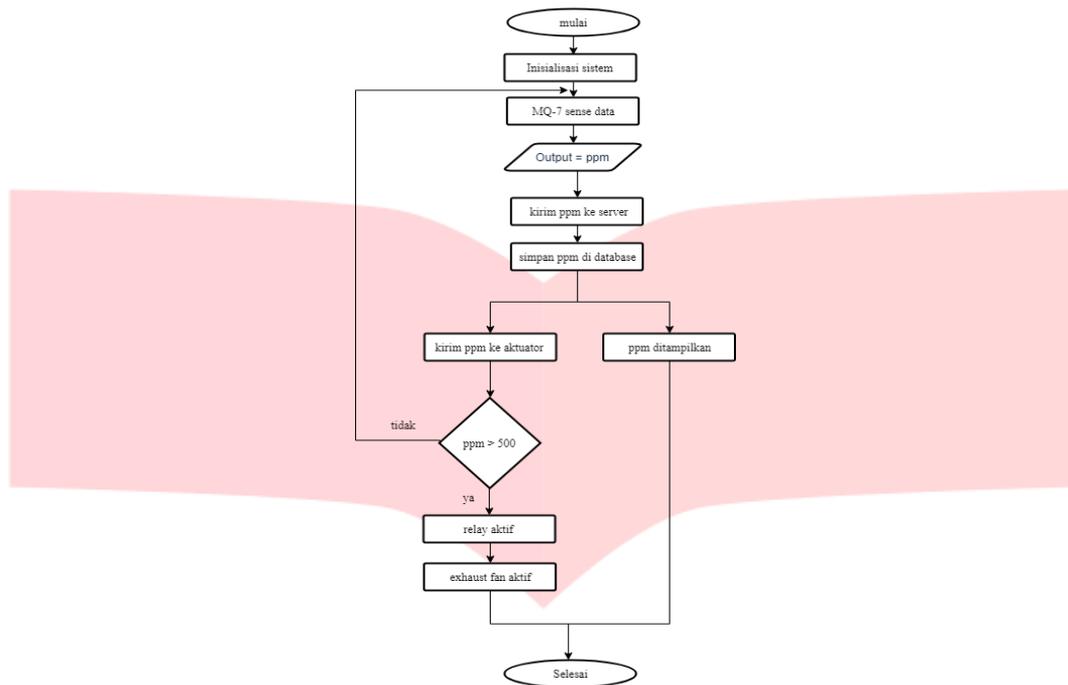
Sistem yang dibuat adalah sebuah jaringan sensor gas CO yang dapat ditampilkan secara *realtime* melalui halaman *web*. Bila gas CO terdeteksi lebih dari 500 ppm maka akan menghidupkan *exhaust fan* secara otomatis. Sistem terdiri dari beberapa node, koordinator, dan dan sebuah server lokal. Terdapat dua jenis node yaitu sensor *node* dan aktuator *node*. Jumlah node yang terpasang Sejumlah N disesuaikan dengan rancangan pengujian. Topologi star digunakan untuk sistem ini. Pada gambar 3.1 semua pengiriman data pada setiap *node* akan dikirim menuju *server*. Vpn digunakan untuk dapat mengakses *server* melalui jaringan internet.



Gambar 3. 1 Gambaran Umum

3.2 Diagram alir

MQ-7 digunakan untuk mengambil nilai ppm pada lingkungan dengan melakukan sense data. Data yang dihasilkan berupa data analog dan perlu diproses lebih lanjut. NodeMCU akan memproses data analog menjadi nilai ppm. Ppm akan dikirim ke *server* oleh NodeMCU. Nilai ppm yang diterima *server* akan disimpan ke dalam *database*. Setelah data disimpan, *user* memiliki 2 pilihan untuk melihat data. *User* dapat mengakses data melalui web lokal atau melalui aplikasi *smartphone* dengan jalur internet. *Server* mengirim data ke aktuator. Apabila nilai ppm lebih dari 500 maka akan mengaktifkan relay. Bila relay aktif, *exhaust fan* juga akan ikut aktif. Pada tahap ini sistem sudah mampu mengirim data sampai pada *user* dan mengaktifkan *exhaust fan* secara otomatis.

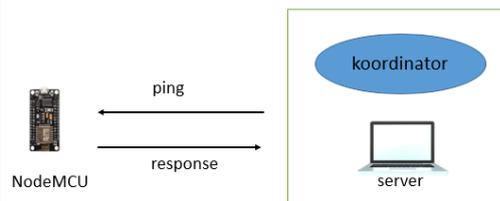


Gambar 3. 2 Diagram Alir Pengiriman Data Sensor hingga ke User

3.3 Skenario Pengujian dan Rancangan Sistem

Sistem akan diuji dengan beberapa skenario. Hal ini dilakukan untuk mengetahui kinerja dari sistem. Kesiapan sistem seperti *hardware* dan *software* yang digunakan mesti diperiksa. Terdapat 4 macam skenario pengujian yang dilakukan yaitu. Skenario untuk mencari jarak jangkauan maksimal sistem, kinerja sistem terhadap jarak, dan kinerja sistem secara keseluruhan.

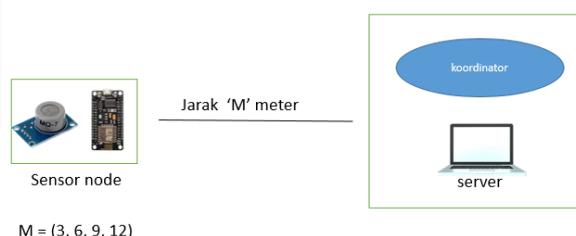
Pengujian jarak jangkauan sistem dilakukan untuk mengetahui jarak maksimal *node* dapat terhubung dan mengirim data. Pengujian dilakukan dengan cara menghubungkan *node* ke koordinator. Bila sudah terhubung, *server* akan melakukan ping ke alamat ip *node* secara terus menerus. *Server* akan menerima respon apakah *node* terhubung atau tidak. *Node* bergerak menjauhi koordinator hingga *node* tidak dapat terhubung ke koordinator. *Node* yang tidak terhubung ditandai dengan response berupa “request timed out”.



Gambar 3.3 Skenario Pengujian Jarak Jangkauan Sistem

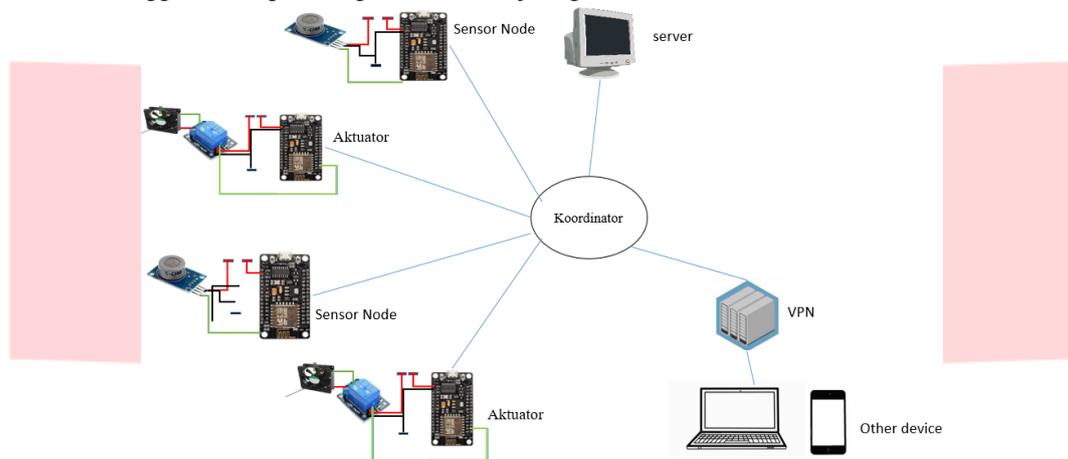
Pengujian jaringan sensor terhadap jarak dilakukan untuk mengetahui nilai kualitas jaringan di setiap jarak. Parameter yang diukur adalah *delay* dan *throughput*. Sensor *node* yang terhubung akan ditempatkan sejauh M meter diukur dari letak koordinator. Sensor *node* mengirimkan paket ke *server*. Aplikasi wireshark pada *server* akan melakukan *capture* paket yang ada selama proses pengiriman berlangsung

Pengujian dilakukan menggunakan protokol MQTT kemudian protokol HTTP. Durasi pengamatan selama 1 menit dengan pengulangan sebanyak 30 kali.



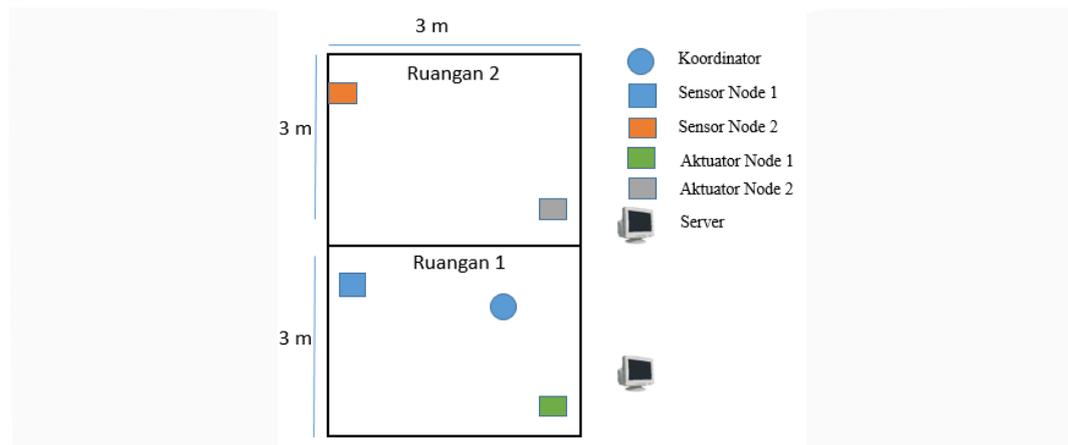
Gambar 3.4 Rancangan Pengujian Terhadap Jarak

Sistem keseluruhan yang dibangun terdiri dari 2 sensor *node*, 2 aktuator *node*, koordinator dan sebuah *server*. Gambar 3.12 menjelaskan bentuk jaringan sistem. Topologi yang digunakan adalah topologi star. Semua sensor *node*, aktuator dan *server* terhubung oleh koordinator. VPN digunakan untuk mengakses *server* diluar jaringan lokal sehingga *user* dapat mengakses melalui jaringan internet.



Gambar 3.5 Rancangan Keseluruhan Sistem

Pada pengujian digunakan dua ruangan tertutup. Luas ruangan yang digunakan sebesar 3 x 3 meter persegi. Pada setiap ruangan terdapat dua ventilasi udara berukuran 20 x 20 sentimeter. Ruangan satu ditempatkan sensor *node*, aktuator dan koordinator. Ruangan dua ditempatkan sensor *node* dan aktuator *node*. *Server* berada di luar ruangan.



Gambar 3.6 Denah Pemasangan Sistem

4. Analisa Pengujian

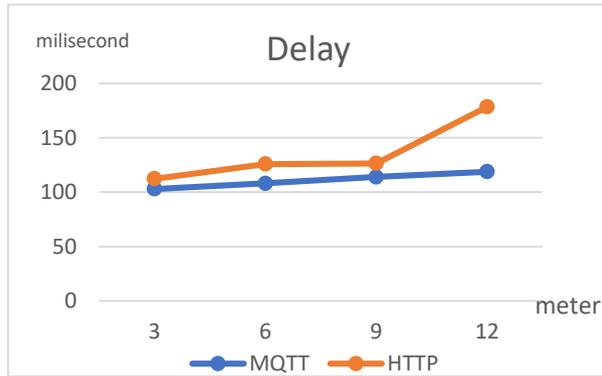
4.1 Pengujian Jarak Jangkauan Koordinator

Pengujian ini dilakukan untuk mengetahui jarak jangkauan maksimal *node* dan koordinator dapat terhubung. Pengujian dilakukan dengan cara menghubungkan satu *node*, *server*, dengan koordinator. Kemudian *node* akan bergerak dari jarak terdekat hingga *node* tidak dapat terhubung ke koordinator. *server* akan melakukan ping *request* ke *node* untuk mengetahui *node* terhubung atau tidak. Tempat pengujian dilakukan disebuah ruangan yang memanjang. Setelah dilakukan pengujian, jarak jangkauan maksimal NodeMCU adalah kurang lebih 12 meter. Pada jarak 13 dan 14 meter *node* sudah tidak dapat terhubung ke koordinator.

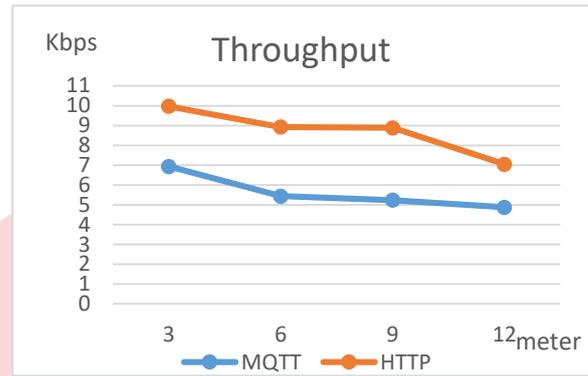
4.2 Pengujian Jaringan Sensor terhadap Jarak dan Jumlah Node

Jarak jangkauan maksimal koordinator adalah 12 meter. Selain jarak jangkauan, perlu diketahui kinerja jaringan sensor terhadap jarak. Untuk itu perlu dilakukan pengujian jaringan sensor terhadap jarak.

Pengujian jaringan sensor terhadap jarak dilakukan dengan cara menghitung *delay* dan *throughput* di setiap jarak yang berbeda. Jarak dihitung dari letak koordinator ke *node*. Jarak pengujian adalah 3 m, 6 m, 9 m, dan 12 meter. Aplikasi Wireshark akan melakukan *capture* paket.



Gambar 4.1 Delay Terhadap jarak



Gambar 4.2 Throughput Terhadap jarak

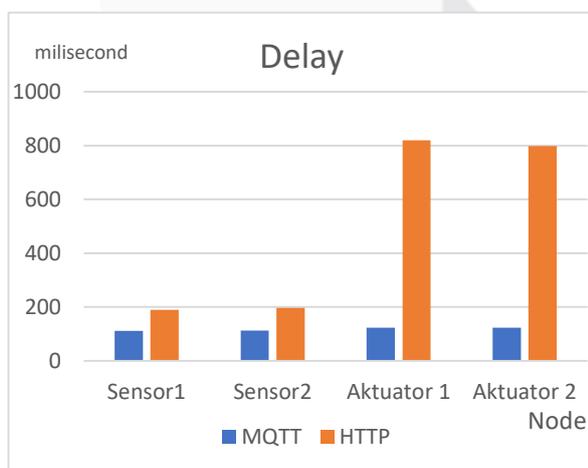
Pada Gambar 4.1 terlihat pengaruh jarak *node* terhadap nilai *delay* yang didapat. Sumbu horizontal adalah jarak dalam satuan meter dan sumbu vertikal adalah *delay* dalam satuan *milisecond*. *Delay* terkecil berada pada jarak 3 meter yang menggunakan protokol MQTT yaitu sebesar 105,065 *milisecond*. *Delay* terbesar berada pada jarak 12 meter menggunakan protokol HTTP yaitu sebesar 178,427 *milisecond*. Dari keseluruhan grafik, *delay* pada protokol MQTT lebih kecil dibanding nilai *delay* pada protokol HTTP, selain itu *delay* cenderung lebih besar disaat jarak antara koordinator dan *node* semakin jauh. Hal ini disebabkan semakin jauh jaraknya, semakin lama waktu yang dibutuhkan data sampai pada *destination* nya

Throughput terkecil berada pada jarak 12 meter yang menggunakan protokol MQTT. Besarnya *throughput* sebesar 4,875 Kbps. *Throughput* terbesar berada pada jarak 3 meter yang menggunakan protokol HTTP sebesar 9,969 Kbps. Pada Gambar 4.2, Sumbu horizontal adalah jarak dalam satuan meter dan sumbu vertikal adalah nilai *throughput* dalam satuan Kbps. Terlihat bahwa nilai *throughput* akan cenderung menurun apabila jarak semakin besar. Hal tersebut berbanding terbalik dengan nilai *delay* yang semakin besar apabila jarak semakin besar.

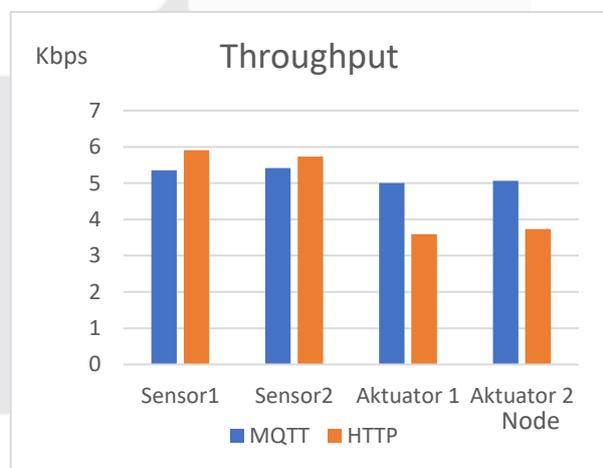
Pada pengujian terlihat bahwa *delay* dan *throughput* pada MQTT selalu lebih kecil dibanding dengan HTTP. Hasil *delay* yang kecil menandakan protokol MQTT mampu mengirim paket lebih cepat dibanding HTTP. Perbedaan *delay* diantara kedua protokol dapat terjadi karena adanya perbedaan proses pengiriman paket pada protokol HTTP dan MQTT. Pada protokol MQTT hanya melakukan satu kali proses koneksi yaitu saat *node* terhubung kemudian setelah itu *node* akan selalu dapat mengirim paket selama koneksi tidak terputus. Sedangkan pada protokol HTTP akan selalu melakukan koneksi ulang setiap paket akan dikirim karena setiap paket selesai dikirim, akan dilakukan *close connection* sehingga perlu membuat koneksi ulang setiap pengiriman paket.

4.3 Pengujian Sistem secara Keseluruhan

Pengujian dilakukan dengan cara menghubungkan semua *node* dan *server* ke koordinator. Sensor MQ-7 dan *exhaust fan* juga sudah terpasang. Wireshark dijalankan untuk *capture* paket kemudian *delay* dan *throughput* dapat dihitung. Terdapat 3 ruangan tertutup yang digunakan untuk pengujian. Tiap ruangan akan ditempatkan 1 *node* sensor. *Server* berada di luar ruangan. 1 *aktuator* berada di *node* yang terpisah dari sensor hal ini dilakukan sebagai *prototype* bila sistem dilakukan pada ruangan yang luas.



Gambar 4.3 Grafik Delay pada Sistem



Gambar 4.4 Grafik Throughput pada Sistem

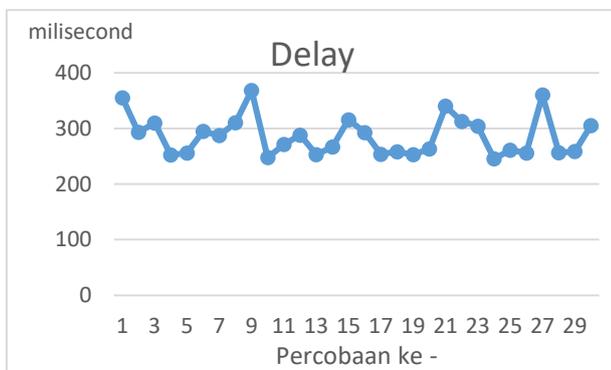
Pada gambar 4.3 menunjukkan nilai *delay* di setiap *node*. Pada protokol MQTT *delay* terkecil terdapat pada sensor *node* 1 yaitu 111,529 *milisecond*. Setiap *node* memiliki nilai *delay* yang berbeda beda karena jarak dan kondisi lingkungan berbeda sehingga dapat mempengaruhi nilai *delay* yang dihasilkan. *delay* pada protokol MQTT lebih kecil dibandingkan *delay* pada protokol HTTP. Hal ini menunjukkan pengiriman data pada MQTT lebih cepat dan aktuator akan lebih cepat merespon dibanding menggunakan HTTP.

Standar ITU – T G114 menjelaskan kategori kualitas jaringan berdasarkan nilai *delay*. Bila mengacu pada standar tersebut, sistem yang menggunakan protokol MQTT berada pada kategori baik dengan range *delay* berada dibawah 150 *milisecond* dan berkategori cukup pada protokol HTTP. Namun pada protokol HTTP, aktuator *node* memiliki kualitas yang buruk karena nilai *delay* yang lebih besar dari 400 *milisecond*.

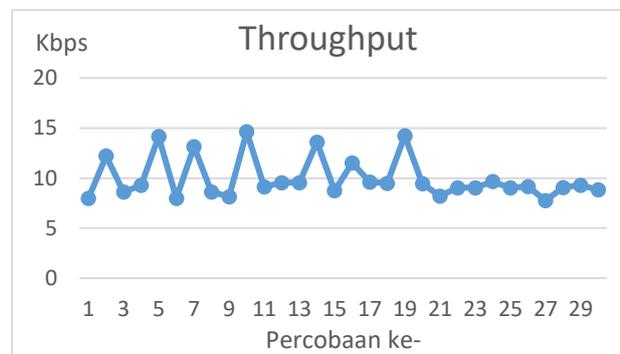
Pada hasil perhitungan *throughput* (Gambar 4.4), sensor satu memiliki *throughput* yang paling besar pada protokol MQTT yaitu sebesar 5,3584 Kbps. Sedangkan pada protokol HTTP sensor 1 memiliki *throughput* yang paling besar yaitu 5,913 Kbps. *Throughput* Pada sensor *node* protokol HTTP lebih besar dibanding MQTT namun pada aktuator *node*, *throughput* MQTT lebih besar dengan protokol HTTP karena *delay* yang jauh lebih besar dibanding MQTT.

4.4 Pengujian Jaringan Internet

User dapat mengakses data melalui jaringan internet. Agar sistem dapat diakses dari luar jaringan, *server* dan user dihubungkan oleh *Virtual Private Network* (VPN) menggunakan aplikasi OpenVPN yang terpasang pada *server*. Pengujian dilakukan dengan cara menjalankan sistem selama 1 menit sebanyak 30 kali perulangan.



Gambar 4.5 Delay Pada Jaringan Internet



Gambar 4.6 Throughput Pada Jaringan Internet

Pada Gambar 4.5 dan Gambar 4.6 terlihat bahwa *delay* dan *throughput* fluktuatif disetiap percobaan. Hal ini disebabkan karena pada setiap pengiriman, paket dapat melewati jalur yang berbeda atau sama. *Delay* terkecil terdapat pada percobaan ke 24 yaitu sebesar 245,35 *milisecond* dan *delay* terbesar terdapat pada percobaan ke 9 yaitu sebesar 368,52 *milisecond*. *Throughput* terkecil terdapat pada percobaan ke 27 yaitu 7.752 Kbps dan *throughput* terbesar terdapat pada percobaan ke 10 yaitu sebesar 14,632 Kbps.

Pada kedua gambar dapat terlihat hubungan antara *delay* dan *throughput*. Bila *delay* kecil maka *throughput* akan besar. Hal ini bisa terjadi karena bila paket lewat dengan kecepatan data yang lebih tinggi maka *delay* akan menjadi lebih kecil.

Tabel 3. 1 Packetloss, Availability, dan Realibility

Send	Failed	Packetloss (%)	Availability (%)	Realibility (%)
6319	1	0,0158	99,984	99,984

Selama dilakukan pengujian, Total keseluruhan paket yang di terima adalah 6319 paket. Paket yang gagal diterima adalah 1 paket. Dari data tersebut dapat dihitung nilai packetloss, Availability dan Realibility. Packetloss sistem adalah 0,0158 %, Availability dan realibility sebesar 99,984 %.

5. Kesimpulan

Berdasarkan dari hasil perancangan, pengujian, dan analisa data maka dapat disimpulkan beberapa hal yaitu:

1. Sistem monitoring dan *controlling* gas CO dapat bekerja dengan baik, dapat memonitoring kadar gas disekitar lingkungan, dapat mengirim data dan diterima *user*, dan melakukan *controlling* apabila kadar gas CO berada pada level berbahaya.
2. Jarak jangkauan maksimum koordinator adalah sebesar 12 meter. Pada pengujian jaringan terhadap jarak, *delay* akan cenderung semakin membesar apabila jarak antara koordinator dan *node* semakin jauh. Sedangkan *throughput* akan cenderung mengecil apabila jarak antara koordinator dan *node* semakin jauh. Pada pengujian jaringan sensor terhadap jumlah *node*, *delay* pada HTTP cenderung meningkat apabila jumlah *node* yang terhubung bertambah namun pada protokol MQTT tidak memberikan efek yang signifikan. *Throughput* akan semakin bertambah besar apabila jumlah *node* yang terpasang semakin banyak.
3. Ketika sistem dijalankan secara keseluruhan, *delay* pada aktuator lebih besar dari pada *delay-delay* sensor *node*. Pada saat menggunakan MQTT, *delay* sensor *node* terbesar adalah 111,877 *milisecond* dan *delay* aktuator *node* terbesar adalah 123,253 *milisecond*. Pada saat menggunakan HTTP, *delay* sensor *node* terbesar adalah 196,853 *milisecond* dan *delay* aktuator *node* terbesar adalah 798,226 *milisecond*. Sesuai standar ITU-T G114 kualitas sistem berada pada kualitas yang baik untuk MQTT dan pada HTTP sensor *node* berada pada kualitas yang baik sedangkan aktuator *node* berada pada kualitas yang buruk..
4. Pada sistem ini penggunaan protokol MQTT lebih baik dibanding HTTP karena pada protokol MQTT hanya melakukan satu kali proses koneksi diawal sehingga membuat *delay* lebih kecil dibanding HTTP. sedangkan pada HTTP akan selalu melakukan proses koneksi setiap data ingin dikirim lalu menutup koneksi ketika data telah selesai dikirim sehingga membuat *delay* HTTP lebih besar. Ukuran panjang paket yang dikirim menggunakan protokol MQTT sebesar 73 Bytes dan pada HTTP sebesar 140 Bytes. *Throughput* pada MQTT lebih kecil dibanding *throughput* pada HTTP.
5. Saat mengakses data melalui internet, sistem memiliki *delay* rata rata sebesar 286,3213 *milisecond* dan *throughput* 9,989 Kbps. *Packetloss* 0,0158 %, *availability* sebesar 99,984 % dan *reability* 99,984%.

DAFTAR PUSTAKA

- [1] Clardy, Peter F; Manaker, scott; Perry, Holly, "Carbon monoxide poisoning," UpToDate, Inc, 2014. [Online]. Available: <http://www.uptodate.com/contents/car>. [Accessed 15 April 2018].
- [2] BPOM, "Sentra informasi keracunan nasional," BPOM, 2014. [Online]. Available: <http://ik.pom.go.id/v2014/beritakeracunan/>. [Accessed 28 April 2018].
- [3] Bhattacharya, Sayantiani; S, Sridevi; R, Pitchiah, Indoor Air quality Monitoring Using Wireless Sensor Network, Chennai: International Conference on Sensing Technology, 2012.
- [4] Simbeye, Daudi S, "Industrial Air Pollution Monitoring System Based on Wireless Sensor Network," *Kournal Of Information Science and Computing Technologies*, vol. 6, no. 2, pp. 612-624, 2017.
- [5] Yokotani, Tetsuya; Sasaki, Yuya, "Comparison with HTTP and MQTT on Required Network Resources for IoT," in *International Conference on Control, Electronics, Renewable Energy and Communications*, Japan, 2016.
- [6] Ilmi, Ahmad Alfian, Rancang Bangun Aplikasi Monitoring Sebagai Informasi Gas Karbon Monoksida Pada Jaringan Sensor Nirkabel, Surabaya: Open Jurnal Institut Bisnis dan Informatika Stikom Surabaya, 2014.
- [7] Iqbal, S; Clower, JH; Hernandez, SA, "A Review Of Disaster Related Carbon Monoxide Poisoning," *AmJ Public Health*, 2012.
- [8] Hadiyani, Murti, Keracunan Karbon Monoksida, Badan Pengawas Obat dan Makanan Republik Indonesia.
- [9] Culler, David; Estrin, Deborah; Srivastava, Mani, Overview of Sensor Network, IEEE, 2004.
- [10] Cohn, Raphael J, "MQTT-V 3.1.1 OS Standards Track Work Product," 2014.
- [11] IBM, "Qualities of Services provided by an MQTT Client," IBM Knowledge Center, [Online]. Available: https://www.ibm.com/support/knowledgecenter/en/SSFKSJ_9.0.0/com.ibm.mq.dev.doc/q029090_.htm. [Accessed 12 February 2018].
- [12] Hanwei Electronics, "Technical data MQ-7 Gas Sensor," Hanwei Electronics, [Online]. Available: <https://www.sparkfun.com/datasheets/Sensors/Biometric/MQ-7.pdf>. [Accessed 10 september 2017].

