Pengembangan Sistem Monitoring Kualitas Udara Terintegrasi Cloud IOT Untuk Diterapkan Pada Smart City

Hermawan Heri Wijaya¹, Sabriansyah Rizqika Akbar, S.T, M.Eng.², Mochammad Hannats Hanafi 1., S.ST.³

¹Mahasiswa, Universitas Brawijaya, Jalan Veteran Kota Malang, Jawa Timur, e-mail: jyuubi.revenged@gmail.com

² Dosen Pembimbing, Jurusan Teknik Komputer Fakultas Ilmu Komputer Universitas Brawijaya, Jalan. Veteran No. 8 Kota Malang, Jawa Timur email:sabriansyah@ub.ac.id

3 Dosen Pembimbing, Jurusan Teknik Komputer Fakultas Ilmu Komputer Universitas Brawijaya, Jalan. Veteran No.8 Kota Malang, Jawa Timur email: hanas.hanafi@ub.ac.id

Abstrak— Pada penelitian ini, penulis mengembangkan sistem monitoring kualitas udara yang dimana sistem menginformasikan kualitas udara disekitarnya memberitahukan hasil pembacaan ke masyarakat melalui sosial media. Sistem secara umum terdiri atas 2 buah sensor udara, yaitu Air Quality dan Carbon Dioxide, yang terhubung ke jaringan internet melalui transmitter ESP8266 dengan mikrokontroler Arduino Mega 2560 yang bertujuan untuk mengupload data ke Cloud IOT dan update status berupa twitter feed sebagai sarana informasi. Pengujian dilakukan dengan 3 cara yaitu konektivitas, fungsionalitas dan delay. Uji konektivitas membuktikan bahwa transmitter dapat terhubung ke access point dan jaringan internet. Uji fungsionalitas membuktikan bahwa sensor dapat membaca udara dan mengambil data. Dan yang terakhir adalah pengujian delay, dimana data mulai dari booting alat hingga data sampai di cloud IOT dan sosial media akan diperhitungkan lama proses dan pengiriman. Dari hasil perhitungan dalam script didapatkan bahwa data akan terkirim sebanyak 24 data dalam 1 jam, namun dari <mark>5</mark> percobaan 4 diantaranya hanya mengirim rata-rata 22 data dalam 1 jam, walaupun selisih jumlah data lumayan besar, namun tujuan penelitian sudah sesuai dengan apa yang diharapkan.

I. PENDAHULUAN

Smart CIty adalah sebutan untuk daerah perkotaan yang hampir seluruh benda di kota tersebut terhubung dengan Mikrokomputer dengan konsep Internet of Things (IoT), seperti Smart Home dimana lampu akan menyala secara otomatis apabila matahari sudah tenggelam^[1]

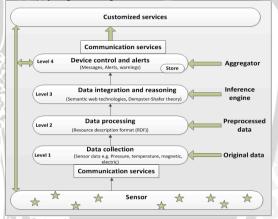
Istilah "Internet of Things" atau dikenal dengan singkatan IOT adalah suatu perangkat keras yang biasanya tertanam dalam berbagai macam benda nyata sehingga benda tersebut dapat tersambung dengan internet [2].

Data WHO memasukkan kota-kota besar di Indonesia dalam pemantauan tingkat polusi udara yang melebihi ambang batas maksimal kadar partikel dalam udara yang di rekomendasikan WHO [3]. Tetapi hal ini tidak didukung dengan adanya sistem monitoring kondisi polusi sehingga kualitas hidup penduduk perkotaan maupun kondisi kota itu sendiri semakin buruk dari waktu ke waktu.

Berdasarkan uraian diatas penulis melakukan pengenbangan dari alat yang sudah ada agar dapat diterapkan pada *Smart City* yang dipadu dengan konsep *Internet of Things* berbentuk Cloud.

II. DASAR TEORI

Dari Penelitian yang dilakukan Aditya Gaur adalah bagaimana IoT diterapkan dalam *Smart City* serta arsitektur dari *Smart City* yang dikategorikan dalam Multi-level ^[1]



Gambar 2.1 Multi-Leve Smart City Architecture

1. Arduino Mega

Arduino Mega 2560 adalah board Arduino yang merupakan perbaikan dari board Arduino Mega sebelumnya. Arduino Mega awalnya memakai chip ATmega1280 dan kemudian diganti dengan chip ATmega2560, oleh karena itu namanya diganti menjadi Arduino Mega 2560. Pada saat tulisan ini dibuat, Arduino Mega 2560 sudah sampai pada revisinya yang ke 3 (R3). Berikut spesifikasi Arduino Mega 2560 R3



Gambar 2.2 Arduino Mega

2. ESP8266

Modul WiFi ini dapat menyambungkan rangkaian elektronika nirkabel karena modul elektronika ini menyediakan akses ke

ja in da

jaringan WiFi secara transparan dengan mudah melalui interkoneksi serial (UART RX/TX) modul WiFi ini bekerja dalam jaringan 802.11b/g/n.



Gambar 2.3 ESP8266

3. MQ-135

MQ-135 Air Quality sensor adalah modul sensor gas yang dapat digunakan untuk menentukan kadar konsentrasi gas-gas berbahaya dalam udara. Modul ini berbasiskan sensor MQ-135, yaitu sensor yang dapat mendeteksi gas amonia, bensol, alkohol, serta gas berbahaya lainnya. Modul ini cocok digunakan pada proses penentuan kualitas udara (air quality control).



Gambar 2.4 MQ-135

4. MG-811

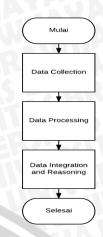
CO2 Gas Sensor Modul ini dirancang untuk memungkinkan mikrokontroler untuk menentukan kapan tingkat gas Karbon Dioksida ditetapkan telah tercapai atau terlampaui. Modul ini dapat digunakan untuk menentukan kadar karbon dioksida yang terdapat pada udara. Modul ini berbasiskan sensor MG-811 yang mampu melakukan pendeteksian gas karbon dioksida dengan range 1 - 10000 ppm.



Gambar 2.5 MG-811

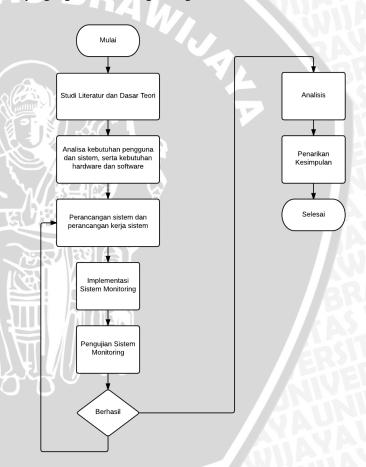
III. METODOLOGI

Adapun dari gambar 2.1 yang diambil untuk penelitian ini hanya sebagian, seperti pada gambar 3.1



Gambar 3.1 Multi-leveling yang diterapkan

Berikut merupakan tahapan-tahapan metodologi penelitian yang digambarkan dengan diagram alir



Gambar 3.2 Diagram alir

IV. PERANCANGAN

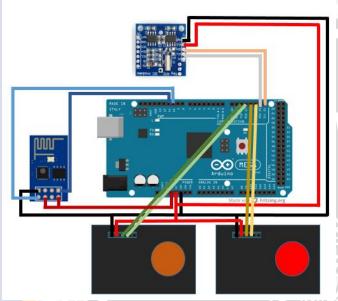
A. Perancangan perangkat keras dan lunak
 Berikut merupakan gambaran umum sistem



Gambar 4.1 Gambaran Kerja Sistem

Coding dari Arduino decompile dan di run oleh perangkat keras dan menghubungkan dirinya ke access point yang ditentukan, kemudian mengirim data ke thingspeak setelah melakukan pembacaan data, setelah data masuk thingspeak, data juga dipost ke twitter.

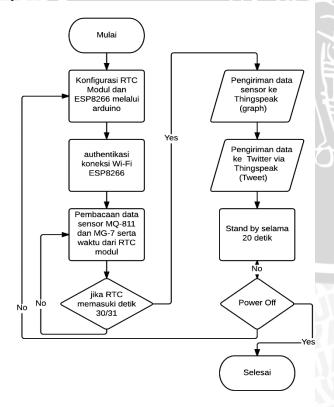
Perancangan perangakt keras
 Perancangan alat ini terdiri dari Arduino Mega 2560,
 ESP8266, RTCModule1307, MG-811 dan MQ-135 yang dihubungkan dengan pin sesuai dengan gambar 4.2



Gambar 4.2 Perancangan Alat

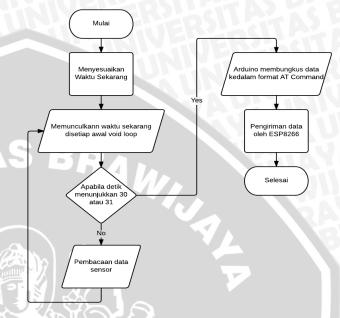
2. Perancangan perangkat lunak

Berikut gambar keseluruhan perangkat lunak dapat dilihat pada Gambar 4.3



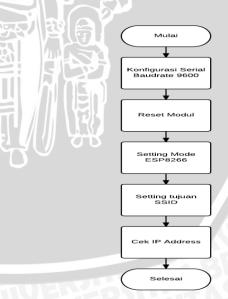
Gambar 4.3 Diagram Alir Algoritma Monitoring Udara

Diawali dengan pembacaan RTC module, ESP8266 dan kemudian sensor, apabila RTC menunjukkan waktu yang sudah direncanakan, data akan dikirim. Adapun diagram alir RTC Module, ESP8266 dan sensor dapat dilihat pada Gambar 4.4, Gambar 4.5 dan Gambar 4.6



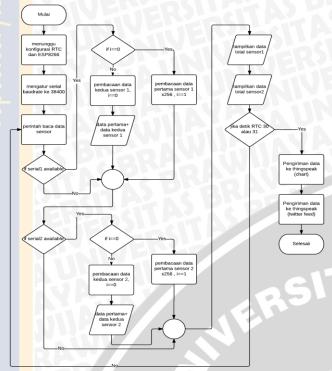
Gambar 4.4 Diagram Alir RTC Module

Setelah menyesuaikan waktu, RTC akan dipanggil terus menerus setiap detik untuk menunjukkan waktu ke Arduino, apabila detik dari waktu yang ditunjukkan adalah 30 atau 31 maka data akan dikirim.



Gambar 4.5 Diagram alir ESP8266

Setelah menyamakan baudrate, ESP disetting dengan Bahasa AT Command, dan diperintah dengan Bahasa AT Command juga.



Gambar 4.7 Diagram alir pembacaan dan pengiriman data sensor

Setelah konfigurasi ESP dan RTC berjalan, pembacaan sensor pun berjalan, apabila salah satu dari kedua konfigurasi tersebut terjadi kesalahan maka sensor tidak akan melakukan pembacaaan apapun.

V. PENGUJIAN DAN ANALISIS

Adapun hasil jadi alat dapat dilihat pad gambar 5.1



Gambar 5.1 hasil jadi alat

Pada Gambar 5.1 terdapat angka dan lingkaran, berikut penjelasan mengenai obyek yang ditandai

- Sensor MQ-135 yang bekerja dengan ditandai indikator hijau.
- 2. Sensor MG-811 yang bekerja dengan ditandai indikator hijau.
- 3. RTC Modul tidak memiliki indikator, sehingga ditambahkan LED berwarna merah.
- 4. Arduino Mega 2560 sebaga mikrokontroler.
- 5. ESP8266 yang dalam keadaan standby, ditandai dengan led merah, apabila dalam keadaan bekerja, led biru akan nyala dan mati secara bergantian

Setelah dibangun sesuai dengan rancangan yang sudah direncanakan, alat akan diuji dalam 3 tahapan, yaitu

- 1. Konektivitas dari ESP8266, baik dari ESP8266 ke access point, ataupun dari ESP8266 ke thingspeak dan thingtweet.
- Fungsionalitas dari sensor dan pengolahan data yang dilakukan oleh sensor dan Arduino.
- 3. Analisa delay yang dihasilkan atau waktu yang dibutuhkan oleh sensor untuk mengakuisisi data hingga terkirim ke thingspeak dalam waktu 1 jam.

Adapun hasil uji yang dihasilkan oleh pengujian adalah sebagai berikut

A. Konektivitas dari ESP8266

1. ESP8266 Menuju access point

Koneksi ini dibentuk dengan perintah AT Command yang dieksekusi didalam Arduino Mega. Adapun hasilnya dapat dilihat pada gambar 5.2

🚥 COM3 (Arduino/Genuino Mega or Mega 2560)

```
DS1307 configured Time=
05:34:30, Date=May 11 2016
AT+RST

OK
bBÖ†@uRcâuR‰,#`BÎÿÊÉ¥DÎÿÂID
[System Ready, Vendor:www.ai-thinker.com]
AT+CWMODE=1
no change
AT+CWJAP="cocacola","SadangWoy12312"

OK
AT+CIFSR
192.168.1.58

OK
ESP8266 Ready!
```

Gambar 5.2 proses koneksi ESP8266 dengan Access Point

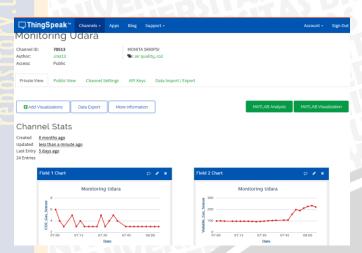
2. ESP8266 Menuju Thingspeak

Pengiriman juga dilakukan dengan perintah AT Command, adapun hasil Serial Monitor dapat dilihat pada Gambar 5.3 dan hasil posting di Thingspeak dapat dilihat pada Gambar 5.4

```
Ok, Time = 5:35:29, Date (D/M/Y) = 11/5/2016
CO2 39
N 210
Ok, Time = 5:35:31, Date (D/M/Y) = 11/5/2016
CO2 38
N 210
AT+CIPSTART="TCP", "184.106.153.149", 80

OK
Linked
GET /update?api_key=06II03N65FVLT39T&field1=38&field2=210
S
OK
Linked
Twitter Status Updated!Autorestart Program!!
```

Gambar 5.3 Pengiriman data ke Thingspeak



Gambar 5.4 Hasil data pada Thingspeak

3. Menuju Twitter

Pengiriman data ke twitter sama seperti pada Gambar 5.3, hanya saja perintah yang di eksekusi agak sedikir berbeda dengan yang dieksekusi untuk menuju thingspeak. Adapun hasil pada Twitter seperti pada Gambar 5.5



Gambar 5.5 hasil upload data ke Twitter

B. Fungsionalitas Sensor

Pada alat yang dibuat terdapat 2 sensor yang bekerja secara bersamaan, namun data yang dibaca dilakukan satu persatu, adapun hasil pembacaan data dapat dilihat pada gambar 5.6

Ok,	Time	_	5:34:57,	Date	(D/M/Y)	= 11/5/2016
Ok,	Time		5:34:58,	Date	(D/M/Y)	= 11/5/2016
Ok,	Time	_	5:34:59,	Date	(D/M/Y)	= 11/5/2016
CO2	44					
1AL O						
Ok,	Time	_	5:35:1,	Date	(D/M/Y) :	= 11/5/2016
C02	44					
N 22	25					
Ok,	Time	=	5:35:3,	Date	(D/M/Y) =	= 11/5/2016
C02	43					
N 225						
Ok,	Time	=	5:35:5,	Date	(D/M/Y) =	= 11/5/2016
CO2	43					
N 22						
Ok,	Time	-	5:35:7,	Date	(D/M/Y) :	= 11/5/2016
CO2						
N 22						
			5:35:9,	Date	(D/M/Y) ·	= 11/5/2016
CO2						
N 22						
		-	5:35:11,	Date	(D/M/Y)	= 11/5/2016
CO2						
N 22						
		-	5:35:13,	Date	(D/M/Y)	= 11/5/2016
C02						
N 21						
			5:35:15,	Date	(D/M/Y)	= 11/5/2016
CO2						
N 21				_		
			5:35:17,	Date	(D/M/Y)	= 11/5/2016
CO2						
CO2		_	5:35:19,	Date	(D/M/Y)	= 11/5/2016
N 21						
		_	E. 2E. 21	Doto	(D (M (V)	= 11/5/2016
CO2			5:35:21,	Date	(D/M/Y)	- 11/5/2016
1002	-± U					

Gambar 5.6 Pengujian pembacaan Sensor

C. Analisa Delay

Dari analisa perhitungan yang dilakukan berdasarkan program yang dibuat didapatkan hasil seperti pada Gambar 5.7

Event Trigger	Delay (ms)
AT+RST	2000
AT+CWMODE	1000
AT+CWJAP	20000
AT+CISFR	1000
RTC	1000
RTC (Loop)	1000
DELAY DISPLAY (Loop)	1000
DELAY SENDING THINGSPEAK	16000
DELAY SENDING TWEETER	16000
DELAY STAND BY	21000
DELAY RESTART	4000
TOTAL DELAY	84.000

Gambar 5.7 hasil perhitungan total delay berdasarkan coding program

Gambar 5.7 didapatkan total delay 84.000 atau 84 detik. Estimasikan looping yang terjadi paling lama 1 menit atau 60 detik, maka pengiriman data dan akuisisi data pada thingspeak membuat totalnya menjadi 144 detik atau kurang lebih dalam 2,5 menit 1 data didapatkan, maka akan ada 24 data yang didapatkan dalam 1 jam pengujian

Pengujian dilakukan selama 1 jam, dari 5 jam pengujian didapatkan hasil berbentuk grafik dalam Gambar 5.8



Gambar 5.8 Grafik hasil pengujian alat

Dari Gambar 5.8 dapat dilihat bahwa rata-rata pengiriman adalah 2.73 dan terjadi peningkatan pada pengujian ke 4 dengan rata-rata pengiriman 2.5, estimasi pengiriman yang berupa 2,5 menit per data terbukti benar namun dikarenakan adanya kemungkinan data tidak sampai atau tidak terkirim karena tidak adanya IP address saat terjadi request DHCP atau IP telah habis, atau adanya timelapse yang terjadi pada Arduino dikarenakan sebuah script yang menunggu sebuah event.

VI. PENUTUP

A. Kesimpulan

Berdasarkan hasil pengujian konektivitas ESP8266, pengujian Fungsionlitas Sensor dan Pengujian Total Delay didapatkan beberapa kesimpulan sebagai berikut 1. ESP8266 dapat digunakan dengan AT Command pada mikrokontroler Arduino dan dapat digunakan sebagai station, access point ataupun keduanya, tergantung dari penggunaan ESP8266 yang diinginkan.

 Sensor MG-811 dan MQ-135 memerlukan daya yang lebih untuk melakuan pembacaan sensor, sehingga diperlukan voltase yang lebih besar dari 6 volt namun tidak membuat mikrokontroler overheat karena tegangan yang terlalu besar.

- 3. Estimasi perhitungan data yang dikirimkan berdasarkan pada Gambar 6.7 yaitu sebanyak 30 data yang masuk dan data yang sampai tidak sepenuhnya sesuai dengan estimasi yang diperhitungkan, berdasarkan hasil yang ditunjukkan oleh Gambar 6.8 dengan rata-rata data yang masuk sebanyak 2,73. Karena banyaknya kemungkinan kesalahan yang dapat terjadi pada saat data mulai dikirim, baik dari IP address, besarnya bandwidth, banyaknya request yang ditangani oleh access point atau server dan masih banyak lagi.
- 4. Sistem yang dirancang sudah dapat dikatakan bisa dipergunakan dengan baik karena data sudah dapat sampai ke thingspeak dan terposting ke twitter, namun hanya estimasi banyak data yang harusnya terkirim ternyata tidak sesuai. Namun secara keseluruhan sistem yang sudah dirancang sesuai dengan apa yang diharapkan oleh penulis.

B. Saran

Ada beberapa saran untuk para peneliti yang ingin melakukan pengembangan pada penelitian ini, sarannya sebagai berikut

- Berfokus untuk mengakurasikan data yang didapatkan oleh sensor, mempelajari sensor lebih jauh dan mencari data tentang karbonmonoksida dan zat berbahaya yang terdapat di udara sebesar berapa PPM dan mengaplikasikannya dalam area outdoor.
- Meneliti lebih dalam lagi agar estimasi data terkirim dan data yang diterima agar tidak terpaut terlalu jauh dan lebih baik.
- 3. ESP8266 tidak dapat bekerja dengan baik apabila traffic upsteam dan downstream pada internet yang digunakan terlalu banyak, selain itu sensor MG-811 dan MQ-135 mudah overheat apabila digunakan di area outdoor, akan lebih baik apabila mencari referensi sensor lainnya atau melakukan rancang bangun sensor sendiri.
- Menambahkan fitur untuk konfigurasi melalui panel LCD dan dengan input berupa keypad, sehingga tidak perlu melakukan *coding* ulang apabila SSID diganti atau berpindah SSID, hanya cukup menginput SSID dan *Password*.

REFERENSI

Aditya, B. G. a. S., 2015. Smart City Architecture and its Applications based on IoT, Coleraine, Co., Londonderry BT52 1SA. UK: A School of Computing and Information Engineering. University of Ulster.

ThingSpeak, 2016. ThingSpeak. [Online]
Available at: http://www.thingspeak.com
[Accessed 6 Juni 2016]

Kompas, 2013. Dampak Polusi Udara Bagi Penduduk Jakarta. [Online]

Available a http://green.kompasiana.com/polusi/2013/12/04/dampak-polusi-udara-bagi-penduduk-jakarta-616543.html

[Accessed 18 September 2014]

