

BAB IV

HASIL PENGUJIAN DAN PENGAMATAN

Dalam bab ini penulis akan menguraikan dan menjelaskan beberapa hasil pengujian dari hasil penelitian tugas akhir ini. Pengujian yang dilakukan meliputi pengujian perangkat lunak (*software*), perangkat keras (*hardware*), dan kinerja keseluruhan sistem serta analisis hasil transmisi data dari *Child Node* ke *server* melalui *Parent Node* serta *Cluster Head* masing-masing *Child Node*.

4.1 Pengujian Sensor DT-SENSE CARBON MONOXIDE

Pengujian CO dengan menggunakan sensor DT-SENSE CARBON MONOXIDE dan aplikasi arduino IDE dengan program mengirimkan data nilai tertinggi.

4.1.1 Tujuan

Pengujian ini dilakukan untuk mengetahui apakah sensor DT-SENSE CARBON MONOXIDE yang digunakan dapat berfungsi dengan baik atau tidak

4.1.2 Alat yang digunakan

- a. Sensor DT-SENSE CARBON MONOXIDE
- b. Usb adapter
- c. Arduino
- d. Komputer/ laptop
- e. *Software* Arduino IDE

4.1.3 Prosedur pengujian

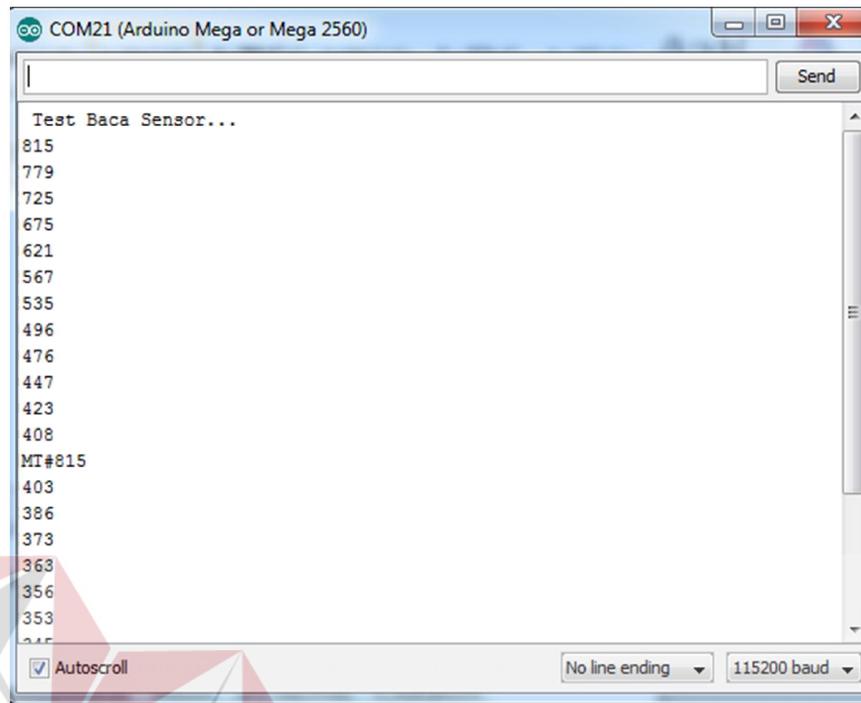
- a. Hubungkan Arduino mega dengan kabel usb

- b. Hubungkan sensor DT-SENSE CARBON MONOXIDE pada arduino mega sesuai dengan yang sudah digambarkan pada gambar 3.6
- c. Nyalakan komputer kemudian hubungkan kabel usb tadi dengan komputer.
- d. Buka *software* Arduino IDE dan isi perintah dalam bahasa C. Sebagai contoh penulis memasukkan perintah sebagai berikut :

```
while ( hitung < 12 )
{
  i2c_start(0xE0); // Alamat I2C modul gas Sensor A
  i2c_write(0x41);
  i2c_stop();
  delay(10);
  i2c_start(0xE1);
  data = (int) i2c_readAck();
  data = data*255 + i2c_readNak();
  i2c_stop();
  data1= data1+data;
  Serial.println(data);
  delay(5000);
  hitung = hitung + 1;
}
```

4.1.4 Hasil Pengujian

Gambar 4.1 menunjukkan bahwa data dikirimkan sesuai dengan perintah program yang telah diisi pada arduino. Dengan begitu sensor DT-SENSE CARBON MONOXIDE ini dapat bekerja dengan baik, dan dapat digunakan untuk sistem.



Gambar 4.1 Hasil Pengujian sensor DT-SENSE CARBON MONOXIDE

4.2 Pengujian Sensor DT-SENSE CARBON DIOXIDE

Pengujian CO₂ dengan menggunakan sensor DT-SENSE CARBON DIOXIDE dan aplikasi arduino IDE dengan program mengirimkan data nilai tertinggi.

4.2.1 Tujuan

Pengujian ini dilakukan untuk mengetahui apakah sensor DT-SENSE CARBON DIOXIDE yang digunakan dapat berfungsi dengan baik atau tidak

4.2.2 Alat yang digunakan

- a. Sensor DT-SENSE CARBON DIOXIDE

- b. Usb adapter
- c. Arduino
- d. Komputer/ laptop
- e. *Software* Arduino IDE

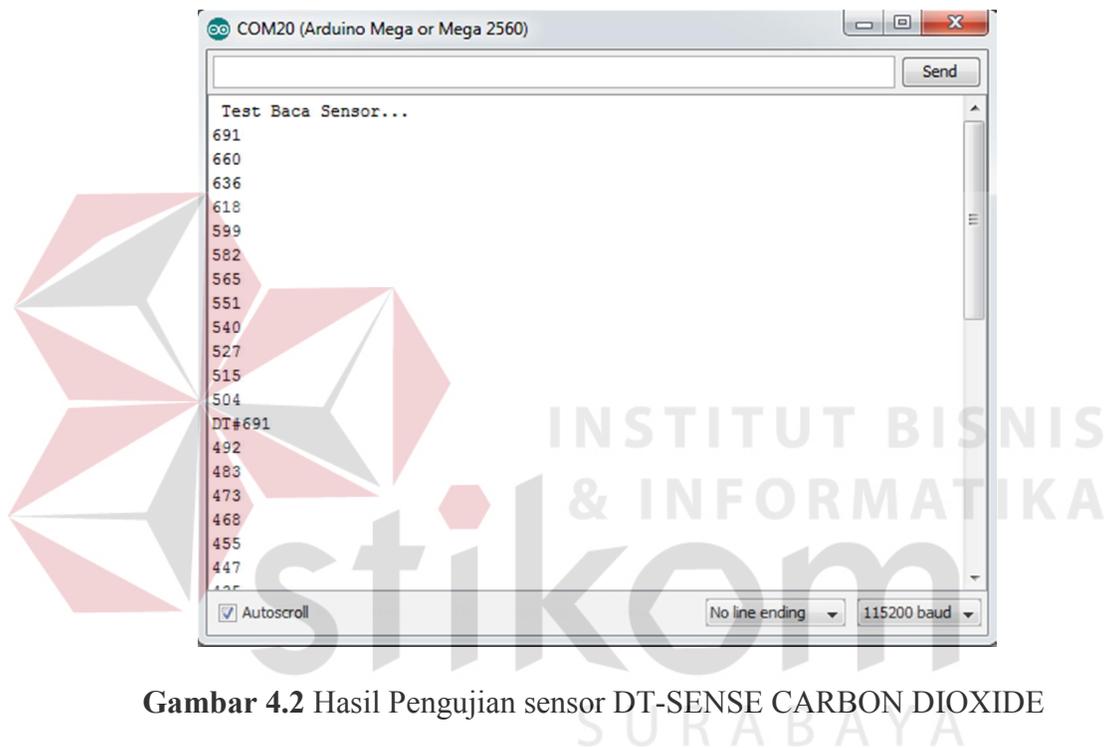
4.2.3 Prosedur pengujian

- a. Hubungkan Arduino mega dengan kabel usb
- b. Hubungkan sensor DT-SENSE CARBON DIOXIDE pada arduino mega sesuai dengan yang sudah digambarkan pada gambar 3.6
- c. Nyalakan komputer kemudian hubungkan kabel usb tadi dengan komputer.
- d. Buka *software* Arduino IDE dan isi perintah dalam bahasa C. Sebagai contoh penulis memasukkan perintah sebagai berikut :

```
while ( hitung < 12 )
{
  i2c_start(0xE0); // Alamat I2C modul gas Sensor A
  i2c_write(0x41);
  i2c_stop();
  delay(10);
  i2c_start(0xE1);
  data = (int) i2c_readAck();
  data = data*255 + i2c_readNak();
  i2c_stop();
  data1= data1+data;
  Serial.println(data);
  delay(5000);
  hitung = hitung + 1;
}
```

4.2.4 Hasil Pengujian

Gambar 4.2 menunjukkan bahwa data dikirimkan sesuai dengan perintah program yang telah diisi pada arduino. Dengan begitu sensor DT-SENSE CARBON MONOXIDE ini dapat bekerja dengan baik, dan dapat digunakan untuk sistem.



Gambar 4.2 Hasil Pengujian sensor DT-SENSE CARBON DIOXIDE

4.3 Pengujian Xbee

Pengujian Xbee dilakukan dengan menggunakan program X-CTU. Program X-CTU merupakan *open source* yang digunakan untuk menkonfigurasi awal Xbee.

4.3.1 Tujuan

Pengujian ini dilakukan untuk mengetahui apakah Xbee yang digunakan dapat berfungsi dengan baik atau tidak.

4.3.2 Alat yang digunakan

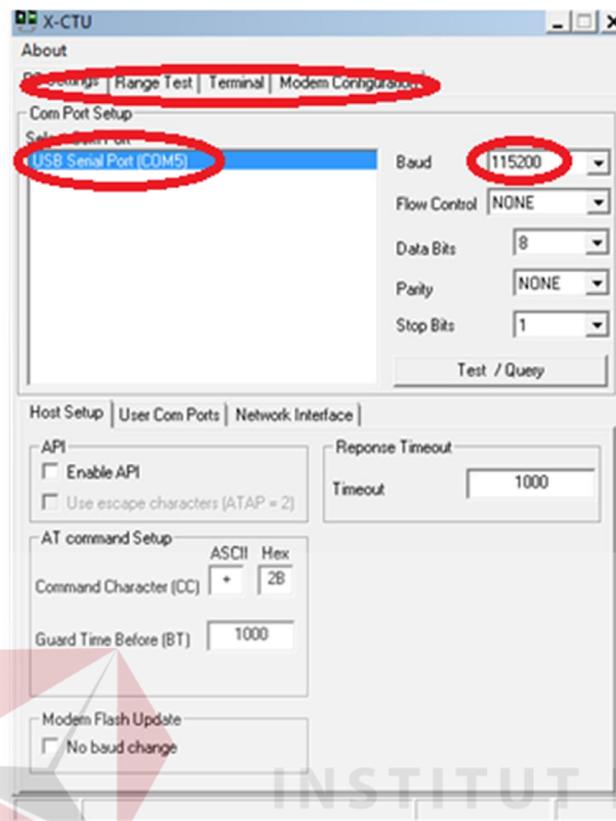
Untuk melakukan percobaan ini maka diperlukan beberapa alat sebagai berikut.

- f. Usb adapter
- g. Xbee adapter
- h. Xbee
- i. Komputer/ laptop
- j. *Software X-CTU*

4.3.3 Prosedur Pengujian

Prosedur pengujian alat :

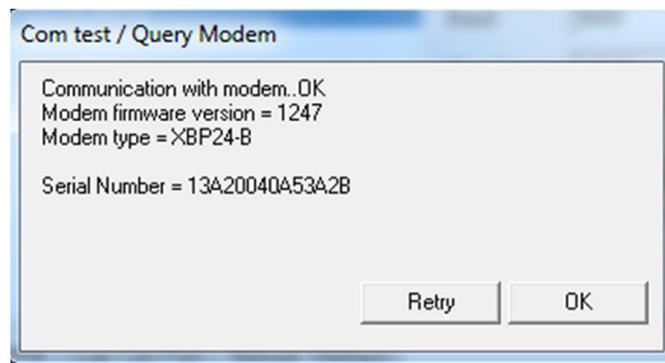
- a. Hubungkan xbee adapter dengan kabel usb adapter.
- b. Nyalakan komputer kemudian hubungkan kabel usb adapternya ke komputer/laptop.
- c. Buka *software X-CTU* dan tekan tombol “ Test / Query” pada tab “PC Setting”.
- d. Maka akan muncul dialog yang dapat mengetahui apakah Xbee yang digunakan dapat terbaca oleh X-CTU atau tidak.



Gambar 4.3 Tampilan Software X-CTU

4.3.4 Hasil Pengujian

Pada Gambar 4.4 tertulis “*Communication with Modem OK*” hal ini menandakan bahwa Xbee yang digunakan dapat berkomunikasi dengan X-CTU. Dengan demikian maka Xbee dapat digunakan pada pengerjaan tugas akhir ini



Gambar 4.4 Xbee dalam keadaan normal

4.4 Pengujian Komunikasi Xbee

Pengujian komunikasi Xbee dilakukan dengan mengatur PAN ID, DL, DH sesuai dengan yang telah dijelaskan pada BAB III. Komunikasi yang baik ketika Xbee yang digunakan sebagai *Cluster Head* dapat menerima pesan dari *Child Node* anggotanya yaitu *Cluster Head 1* dapat menerima pesan dari *Child Node 1* dan *Child Node 2* sedangkan *Cluster Head 2* dapat menerima pesan dari *Child Node 3* dan *Child Node 4* dan Xbee yang digunakan sebagai *Parent Node* atau *coordinator* dapat menerima pesan dari *Cluster Head 1* dan *Cluster Head 2*.

4.4.1 Tujuan

Pengujian ini dilakukan untuk mengetahui apakah Xbee yang digunakan dapat berkomunikasi dengan baik.

4.4.2 Alat yang digunakan

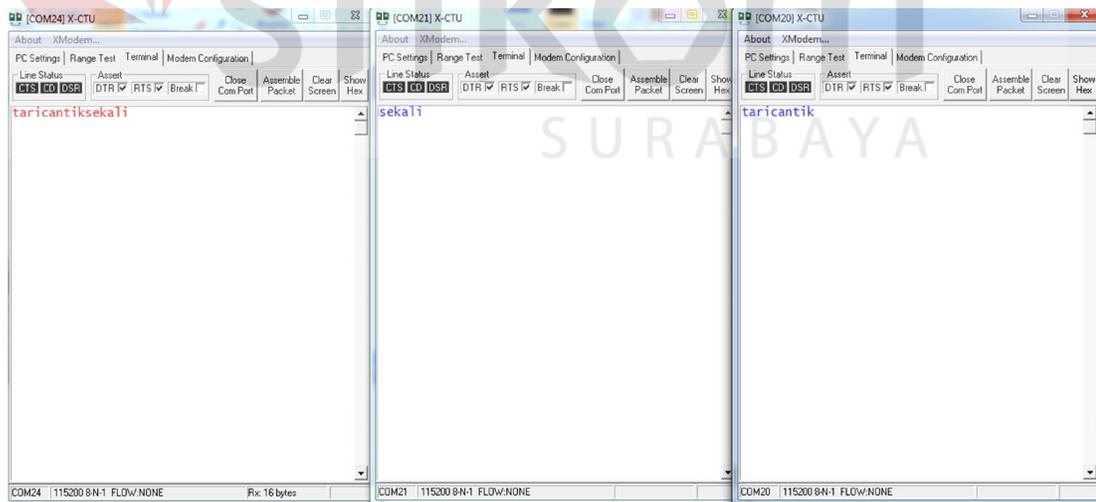
Alat yang digunakan untuk melakukan pengujian antara lain :

- a. Usb adapter
- b. Xbee adapter
- c. Xbee
- d. Komputer/ laptop
- e. *Software X-CTU*

4.4.3 Prosedur Pengujian

PAN ID ketiga Xbee di samakan nilainya yaitu dengan nilai 1313, dan DH pada Xbee disamakan nilainya sesuai nilai DL yang berada pada belakang Xbee 13A200, DH di *Child Node 1* dan *Child Node 2* diberi nilai sesuai dengan SL pada Xbee yang digunakan untuk *Cluster Head 1*.

4.4.4 Hasil Pengujian



Gambar 4.5 Komunikasi *multipoint* Xbee

Pada gambar 4.5 diatas, kalimat dengan warna biru menunjukkan bahwa xbee *Child Node 1* dan xbee *Child Node 2* sedang mengirim data kepada xbee yang menjadi *Cluster Head*, sedangkan kalimat dengan warna text merah menandakan bahwa xbee *Cluster Head* sedang menerima kiriman data dari xbee *Child Node 1* dan xbee *Child Node 2*. Pada gambar 4.5 tersebut terlihat xbee *Cluster Head* dapat menerima dengan baik data dari masing – masing xbee yang digunakan sebagai *Child Node*. Hal ini ditandai dengan data yang diterima oleh xbee *Cluster Head* sama dengan data yang dikirim oleh xbee *Child Node 1* dan xbee *Child Node 2*

4.5 Pengujian Komunikasi Serial pada Arduino

Pengujian arduino dilakukan dengan memasukan skrip program sederhana pada arduino menggunakan aplikasi arduino IDE. Arduino yang baik dapat mengeksekusi program dengan baik.

4.5.1 Tujuan

Pengujian ini dilakukan untuk mengetahui apakah arduino yang digunakan tidak mengalami kerusakan. Sehingga saat arduino digunakan pada sistem dapat membantu sistem berjalan dengan baik.

4.5.2 Alat yang digunakan

Alat yang digunakan untuk melakukan pengujian antara lain :

- a. Kabel usb
- b. Arduino Mega 2560

- c. Komputer/laptop
- d. *Software* Arduino IDE

4.5.3 Prosedur pengujian

- a. Hubungkan Arduino dengan kabel usb
- b. Nyalakan komputer kemudian hubungkan kabel usb tadi dengan komputer.
- c. Buka *software* Arduino IDE dan isi perintah dalam bahasa C. Sebagai contoh

penulis memasukkan perintah sebagai berikut :

```
void setup()
{
    Serial.begin(9600);
    Serial.println("Cek Mulai :");
}
int i=0;
void loop()
{
    Serial.print("Data ke");
    Serial.println(i);
    delay(1000);
    i++;
}
```

- d. Apabila telah selesai untuk mengisi perintah, maka tekan “*Verify*” untuk mengecek apabila terdapat perintah yang salah dalam bahasa C. Dan tekan “*Upload*” untuk memasukkan perintah tersebut ke dalam Arduino Mega 2560.
- e. Setelah *program* telah berhasil dimasukkan, maka tekan *icon Serial monitor* pada kanan atas. Maka akan muncul tampilan serial monitor.

- f. Setelah window serial monitor muncul, amati kiriman data serial oleh arduino.

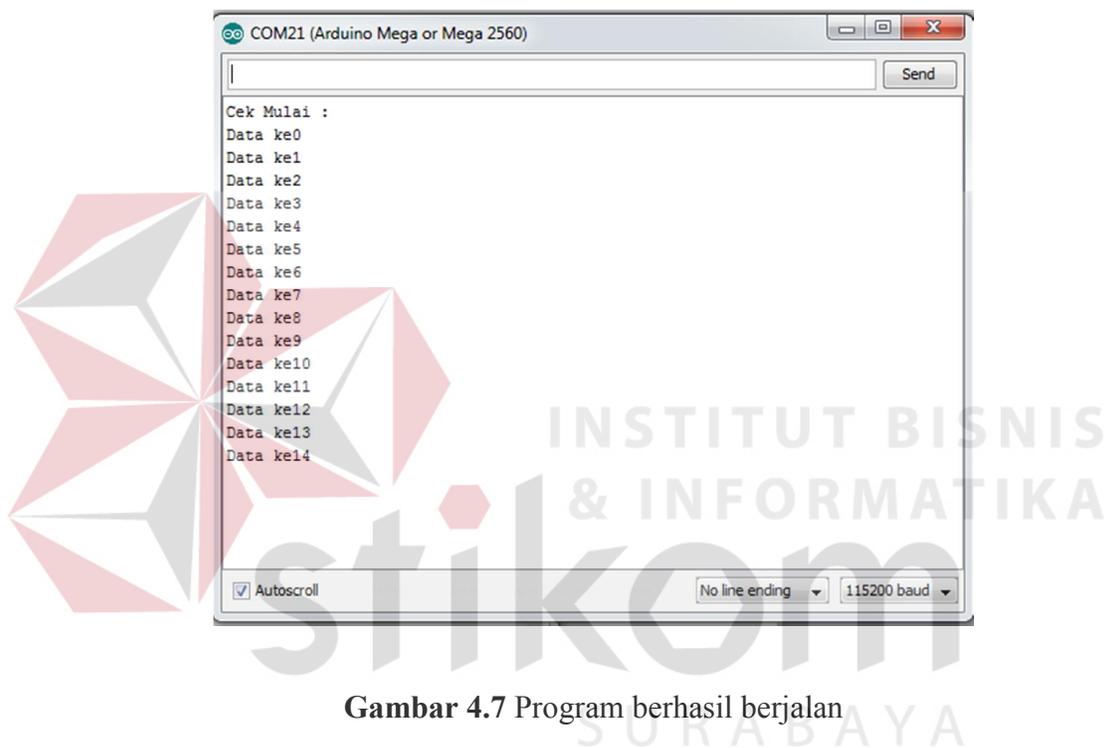
4.5.4 Hasil pengujian

Hasil dari pengujian pengisian program ke arduino dapat dilihat pada gambar 4.6. Pada kiri bawah terdapat tulisan *done uploading* menunjukkan bahwa arduino yang digunakan berhasil diisi dengan program yang telah ditulis dalam *software* arduino IDE.



Gambar 4.6 Upload program berhasil

Program yang dimasukkan kedalam arduino merupakan program untuk mengirimkan data menggunakan serial. Proses pengiriman ini apabila arduino masih dihubungkan dengan USB PC maka kita dapat menerima data yang dikirim menggunakan menu serial monitor pada *software* arduino IDE. Hasil dari serial monitor dapat dilihat pada gambar 4.7.



Gambar 4.7 Program berhasil berjalan

Gambar 4.7 menunjukkan bahwa data dikirimkan sesuai dengan perintah program yang telah diisi pada arduino. Dengan begitu arduino ini dapat bekerja dengan baik, dan dapat digunakan untuk sistem.

4.6 Pengujian Sistem

Pengujian ini dilakukan untuk pengambilan data agar data yang telah di transmisikan dapat di hitung paket *loss* dan *delay*nya yang terjadi ketika proses

mentransmisikan data dari *Child Node* pada masing – masing *Cluster Head*, *Cluster Head* pada *Parent Node*, *Child Node* pada *Parent Node*, *Parent Node* pada *server* dan *Child Node* pada *server*. Analisa dilakukan dengan melakukan beberapa pengujian.

4.6.1 Tujuan

Pengujian ini bertujuan untuk mendapatkan data kualitas udara yaitu CO dan CO₂, dan dari data tersebut penulis dapat menganalisa berapa persen data yang hilang serta berapa *delay* yang dibutuhkan untuk mengirimkan hasil *monitoring* ke tujuan masing – masing. Sehingga dapat disimpulkan apakah *memonitoring* udara dengan menggunakan metode *cluster* yang dibuat ini berjalan dengan baik.

4.6.2 Alat yang digunakan

Alat yang digunakan untuk pengujian sistem ini antara lain:

- a. Sensor CO DT-SENSE CARBON DIOXIDE
- b. Sensor CO₂ DT-SENSE CARBON MONOXIDE
- c. Arduino Mega 2560
- d. Arduino IDE
- e. Xbee
- f. Xbee Shield
- g. Kabel USB
- h. Komputer/laptop
- i. *Software* Visual Basic
- j. *Software* Snagit

4.6.3 Prosedur Pengujian

- a. Penentuan lokasi untuk pengambilan data bisa dilihat pada gambar 3.1
- b. Hubungkan Arduino dan komputer dengan menggunakan kabel USB.
- c. Letakkan Xbee dan Xbee Shield pada atas Arduino
- d. Letakkan Sensor CO pada CN 1 dan CN 3 dan sensor CO₂ pada CN 2 dan CN 4
- e. Lakukan pengambilan data selama 30 menit
- f. Amati data, apakah data masing – masing *Child Node* dapat di terima sampai dengan *Cluster Head* dengan aplikasi Arduino IDE
- g. Amati data, apakah data masing – masing *Cluster Head* dapat di terima sampai dengan *Parent Node* dengan aplikasi Arduino IDE
- h. Amati data, apakah data masing – masing *Child Node* dapat di terima sampai dengan *Parent Node* dengan aplikasi Arduino IDE
- i. Amati data, apakah data masing – masing *Parent Node* dapat di terima sampai dengan *server* dengan aplikasi Arduino IDE
- j. Amati data, apakah data masing – masing *Child Node* dapat di terima sampai dengan *server* dengan aplikasi Arduino IDE
- k. Pada masing – masing komputer dilakukan perekaman video untuk menghitung *delay*

1. Kumpulkan data *monitoring* yang telah didapat kedalam sebuah komputer agar dapat dianalisa.

4.6.4 Hasil Pengujian

Dari hasil pengujian ini bahwa topologi *cluster* dapat mengirimkan data karena data dari CN 1, CN 2 dapat diterima oleh CH 1, CN 3 dan CN 4 dapat diterima oleh CH 2 sesuai dengan data yang ditransmisikan. Kemudian data diterima oleh CH 1 dan CH 2 tersebut dapat ditransmisikan ke PN dan dapat diterima oleh PN sesuai dengan data yang dikirim dari CH maupun data yang ada pada CN. Dan yang terakhir data dari PN dapat ditransmisikan ke *server* dan dapat diterima oleh *server* sesuai dengan data yang dikirim dari PN maupun data yang ada pada CH dan CN. Pada pengujian ini nilai keluaran adalah nilai keluaran ADC. Berikut ini adalah contoh tampilan pada saat transmisi data yang menunjukkan bahwa topologi *cluster* ini bekerja dengan baik dan dapat mentransmisikan data sesuai dengan format pengiriman data yang telah di jelaskan pada BAB III.



```

17
17
17
MR#16#
16
15
15
17
15
16
15
16
17
16
15
15
MR#15#
15
16
16
15
16
15

```

Gambar 4.8 Tampilan transmisi data pada CN 1

```
328
328
327
DR#327#
328
328
328
329
328
329
329
329
328
328
DR#328#
328
327
326
326
325
325
█
```

Gambar 4.9 Tampilan transmisi data pada CN 2



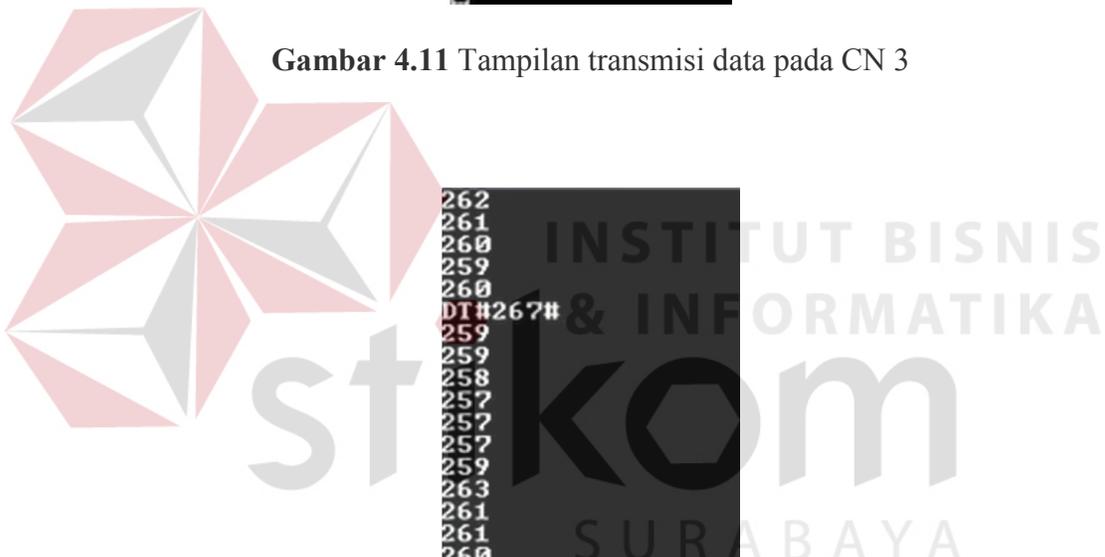
```
PDR#343#
PMR#16#
PDR#347#
PMR#16#
PDR#344#
PMR#16#
PDR#341#
PMR#16#
PDR#337#
PMR#28#
PDR#332#
PMR#16#
PDR#328#
PMR#16#
PDR#328#
```

INSTITUT BISNIS
& INFORMATIKA
stikom
SURABAYA

Gambar 4.10 Tampilan transmisi data pada CH 1

```
44
44
44
44
MI #45#
44
44
44
44
44
44
44
44
44
44
44
44
44
44
MI #44#
44
44
44
44
44
□
```

Gambar 4.11 Tampilan transmisi data pada CN 3

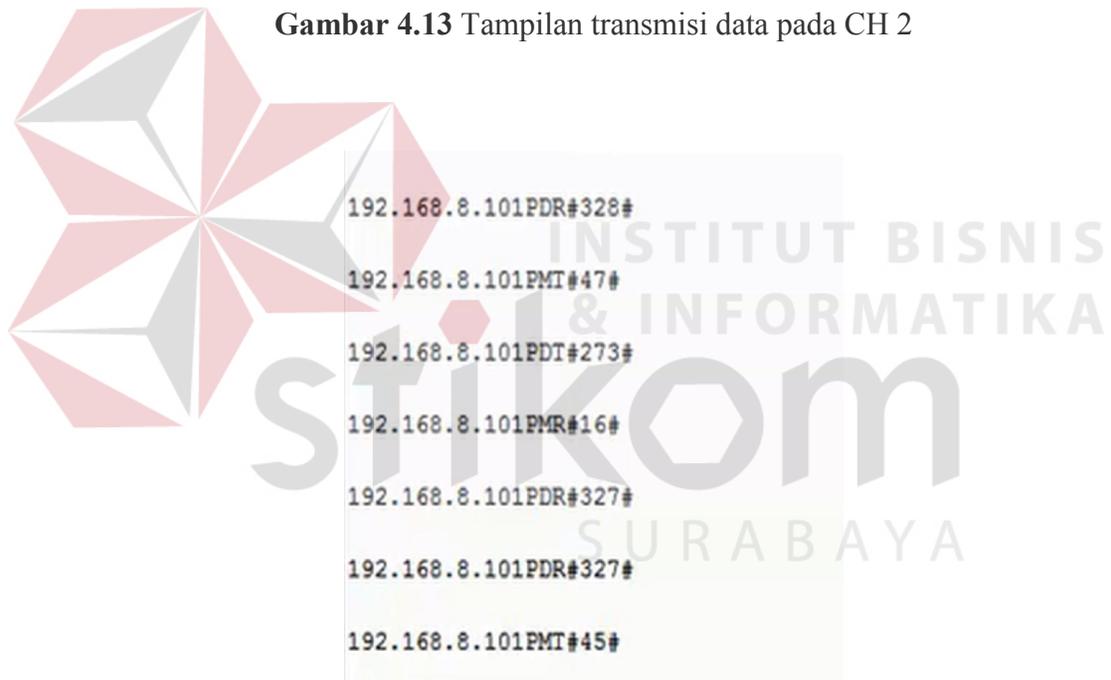


```
262
261
260
259
260
DT #267#
259
259
258
257
257
257
259
263
261
260
259
DT #263#
259
258
258
258
□
```

Gambar 4.12 Tampilan transmisi data pada CN 4

```
PDI#258#  
PMT#48#  
PDI#258#  
PMT#48#  
PDI#263#  
PMT#48#  
PDI#288#  
PMT#48#  
PDI#275#  
PMT#48#  
PDI#266#  
PMT#48#  
PDI#284#  
PMT#48#  
PDI#265#  
PMT#47#  
PDI#257#  
PMT#47#  
PDI#273#  
PMT#45#  
PDI#267#  
PMT#44#  
PDI#263#  
□
```

Gambar 4.13 Tampilan transmisi data pada CH 2



Gambar 4.14 Tampilan transmisi data pada PN



Gambar 4.15 Tampilan transmisi data pada server

1. Berikut adalah hasil data pengujian nilai rata - rata karbon monoksida

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
19		16				16				16						
20		28				28				28						
21		16				16				16						
22		16				16				16						
23		16				16				16						
24		15				15				15						
25		15				15				15						
26		15				15				15						
27		15				15				15						
28		15				15				15						
29		15				15				15						
30		14				14				14						
31		14				14				14						
32		14				14				14						
33		14				14				14						
34		14				14				14						
35		14				14				14						
36		13				14	data double			14						
37		13				13				13						
38		13				13				13						
39		13				13				13						
40		13				13				13						
41		13				13				13						
42		13				13				13						
43		14				13				13						
44		14				14				14						
45		13				14				14						
46		14				13				13						
47		14				14				14						
48		13				14				14						
49		13				13				13						
50						13				13						

Gambar 4.16 Data nilai rata - rata karbon monoksida

Dari gambar di atas dapat dilihat bahwa kolom A adalah data dari CN, kolom F adalah data dari CH, kolom J adalah data dari PN dan kolom N data dari *server*. Data dari CN terdapat 30 data yang di ambil setiap menit selama 30 menit dimulai dari nomor tabel 20, ketika data itu dikirim ke CH terdapat paket *loss* berupa data *double* yang dapat dilihat pada *text highlight* berwarna kuning . Kemudian data dari CH dikirimkan ke PN tidak ada paket *loss*. Dan yang terakhir data dari PN akan di kirim ke *server* empat data *loss* yang dapat dilihat pada *text highlight* berwarna merah. Untuk *delay* dihitung dengan melihat dari selisih waktu antara data pertama dikirim dibandingkan dengan waktu saat data pertama diterima yang ditunjukkan melalui aplikasi snagit.

2. Berikut adalah hasil data pengujian nilai rata - rata karbon dioksida

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
23	332				332				332				328		
24	328				328				328				data loss		
25	328				328				328				data loss		
26	327				327				327				data loss		
27	328				328				327				327		
28	325				325				328				328		
29	324				324				325				325		
30	324				324				324				data loss		
31	324				324				324				324		
32	325				325				324				324		
33	332				332				325				325		
34	330				330				332				332		
35	330				330				330				330		
36	331				330	data double			330	data double			331		
37	331				331				330	data double			331		
38	333				331				331				333		
39	340				333				331				340		
40	340				340				333				340		
41	337				340				340				337		
42	334				337				340				334		
43	332				334				337				332		
44	329				332				334				329		
45	327				329				332				327		
46	332				327				329				332		
47	340				332				327				340		
48	339				340				332				339		
49	345				339				340				345		
50	334				345				339				345		
51	324				334				345	data double			345		
52	320				324				345	data double			334		
53					320				345				324		
54									334				320		
55									324						
56									320						

Gambar 4.17 Data nilai rata - rata karbon dioksida

Dari gambar di atas dapat dilihat bahwa kolom A adalah data dari CN, kolom E adalah data dari CH, kolom I adalah data dari PN dan kolom M data dari server. Data dari CN terdapat 30 data yang di ambil setiap menit selama 30 menit dimulai dari nomor tabel 23, ketika data itu dikirim ke CH terdapat paket *loss* berupa data *double* yang dapat dilihat pada *text highlight* berwarna kuning . Kemudian data dari CH dikirimkan ke PN terdapat empat paket *loss* berupada data *double* yang dapat dilihat pada *text highlight* berwarna kuning Dan yang terakhir data dari PN akan di kirim ke server empat data *loss* yang dapat dilihat pada *text highlight* berwarna merah. Untuk *delay* dihitung dengan melihat dari selisih waktu antara data pertama dikirim dibandingkan dengan waktu saat data pertama diterima yang ditunjukkan melalui aplikasi snagit.

3. Berikut adalah hasil data pengujian nilai tertinggi karbon monoksida

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
10	48				48				48				48		
11	48				48				48				48		
12	48				48				48				48		
13	47				47				47				47		
14	47				47				47				47		
15	45				45				45				45		
16	44				44				44				44		
17	44				44				44				44		
18	43				43				43				43		
19	43				43				43				43		
20	45				45				45				45		
21	45				45				45				45		
22	45				45				45				45		
23	45				45				45				45		
24	45				45				45				45		
25	48				48				48				48		
26	49				49	data double			49	data loss			49		
27	48				48				48				48		
28	46				46				46				46		
29	46				46				46				46		
30	46				46				46				46		
31	45				45				45				45		
32	43				43				43				43		
33	44				44				44				44		
34	45				45				45				45		
35	45				45				45				45		
36	45				45				45				45		
37	45				45				45				45		
38	44				44				44				44		
39	43				43				43				43		
40					43				43				43		
41					43				43				43		

Gambar 4.18 Data nilai tertinggi karbon monoksida

Dari gambar di atas dapat dilihat bahwa kolom A adalah data dari CN, kolom E adalah data dari CH, kolom I adalah data dari PN dan kolom M data dari *server*. Data dari CN terdapat 30 data yang di ambil setiap menit selama 30 menit dimulai dari nomor tabel 10, ketika data itu dikirim ke CH terdapat paket *loss* berupa data *double* yang dapat dilihat pada *text highlight* berwarna kuning . Kemudian data dari CH dikirimkan ke PN terdapat dua paket *loss* yang dapat dilihat pada *text highlight* berwarna merah ada data bernilai “48” yang hilang dan pada *text highlight* berwarna kuning kedua ada paket *loss* berupa data *double*. Dan yang terakhir data dari PN akan di kirim ke *server* empat data *loss* yang dapat dilihat pada *text highlight* berwarna merah. Untuk *delay* dihitung dengan melihat dari selisih waktu antara data pertama dikirim dibandingkan dengan waktu saat data pertama diterima yang ditunjukkan melalui aplikasi snagit.

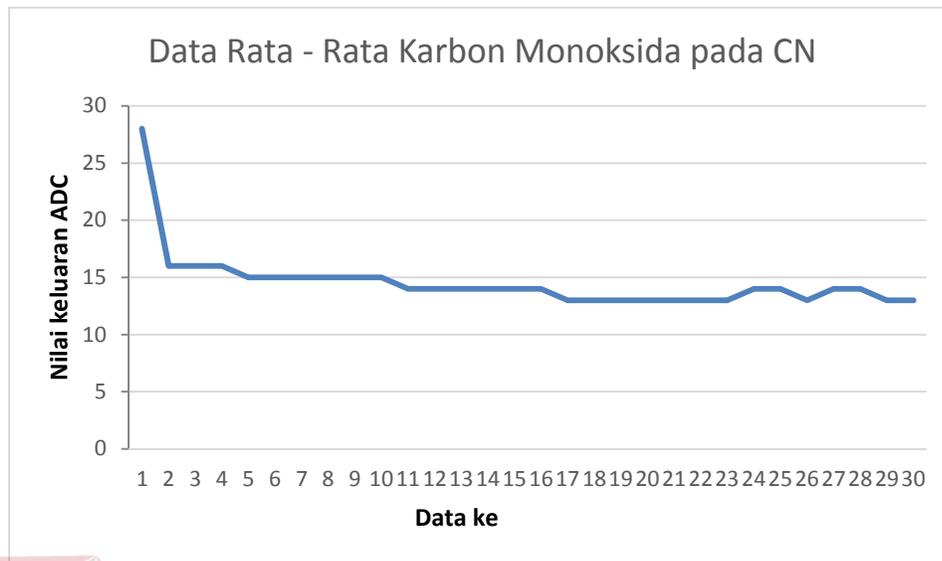
4. Berikut adalah hasil data pengujian nilai tertinggi karbon dioksida

X	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
10	284					284				284						
11	265					265				265					data loss	
12	257					257				257						
13	273					273				273						
14	267					267				267						
15	263					263				263						
16	266					266				data loss						
17	263					263				263						
18	262					262				262						
19	278					278				278						
20	310					310				310					data loss	
21	299					299				299						
22	271					271				271						
23	262					262				262						
24	255					255				255						
25	262					262				262						
26	261					261				261						
27	255					255				255						
28	250					250				250						
29	302					302				302						
30	359					359				359						
31	277					data loss				274					data loss	
32	274					274				275						
33	275					275				305						
34	305					305				292						
35	292					292				270						
36	270					270				data double						
37	267					267				267						
38	253					253				253						
39	270					270				270						
40																

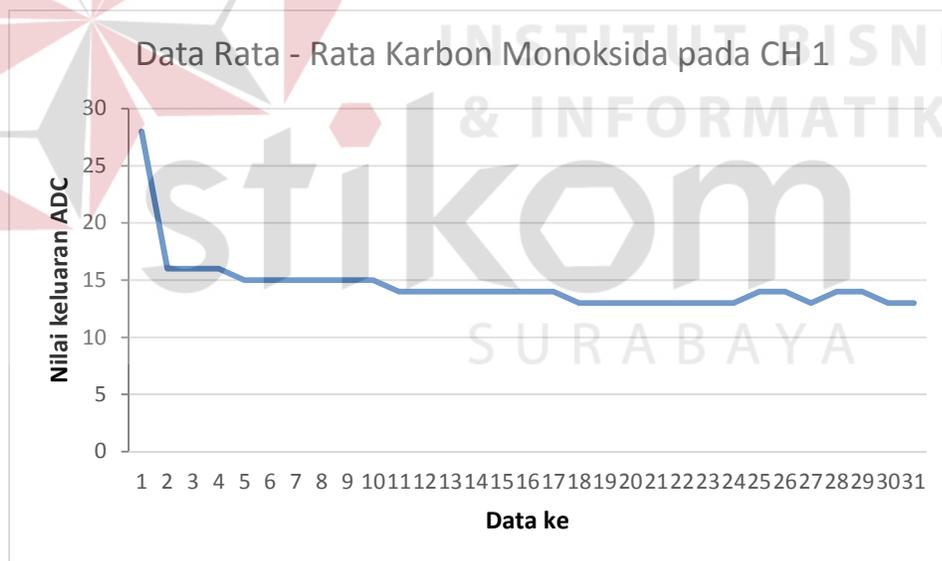
Gambar 4.19 Data nilai tertinggi karbon dioksida

Dari gambar di atas dapat dilihat bahwa kolom A adalah data dari CN, kolom F adalah data dari CH, kolom J adalah data dari PN dan kolom N data dari *server*. Data dari CN terdapat 30 data yang di ambil setiap menit selama 30 menit dimulai dari nomor tabel 10, ketika data itu dikirim ke CH terdapat paket *loss* yang dapat dilihat pada *text highlight* berwarna merah . Kemudian data dari CH dikirimkan ke PN terdapat dua paket *loss* yang dapat dilihat pada *text highlight* berwarna merah ada data bernilai “266” yang hilang dan pada *text highlight* berwarna kuning kedua ada paket *loss* berupa data *double*. Dan yang terakhir data dari PN akan di kirim ke *server* delapan data *loss* yang dapat dilihat pada *text highlight* berwarna merah.dan *text highlight* berwarna kuning berupa data *loss* yang masuk ke PN tanpa peneliti mengetahui data tersebut darimana. Untuk *delay* dihitung dengan melihat dari selisih waktu antara data pertama dikirim dibandingkan dengan waktu saat data pertama diterima yang ditunjukkan melalui aplikasi snagit.

5. Data dari masing – masing *Child Node* ke *Cluster Head*
 - a. CN 1 ke CH 1

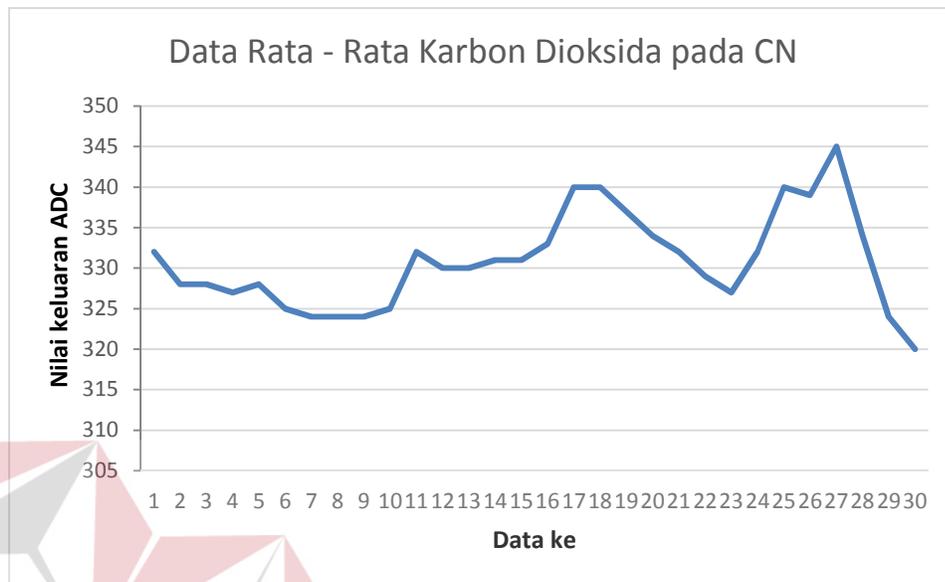


Gambar 4.20 Data rata-rata CO pada CN 1 untuk CH 1

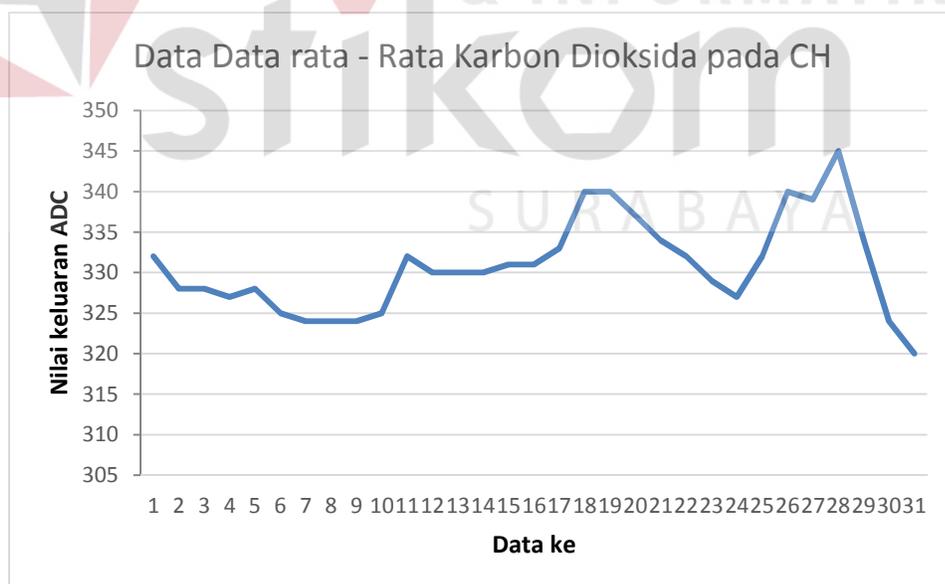


Gambar 4.21 Data rata-rata CO pada CH 1 untuk CN 1

b. CN 2 ke CH 1

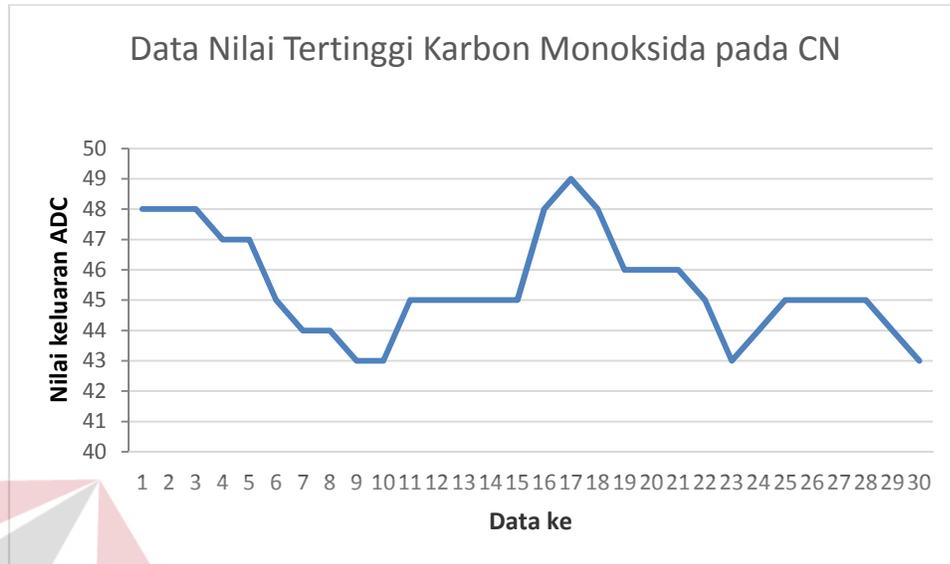


Gambar 4.22 Data rata-rata CO₂ pada CN 2 untuk CH 1

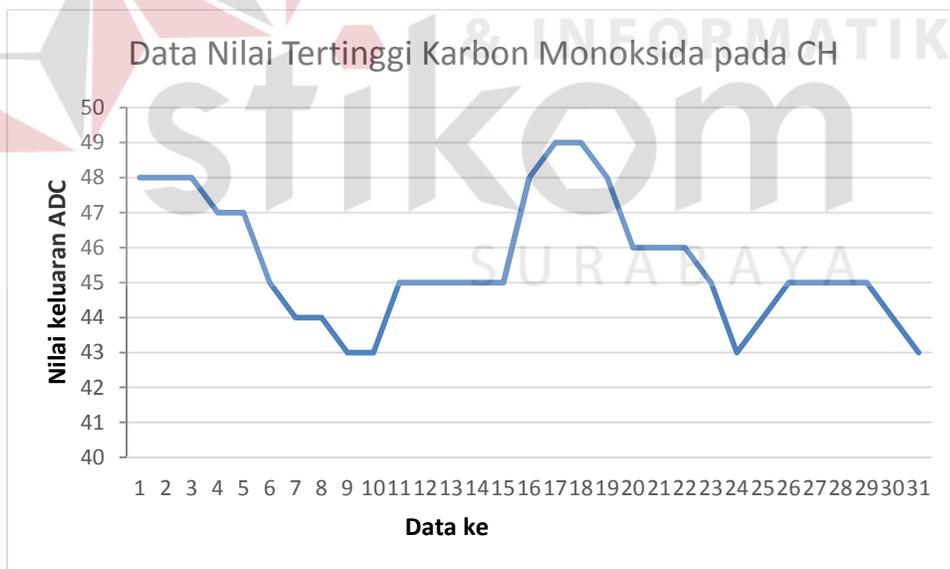


Gambar 4.23 Data rata-rata CO₂ pada CH 1 untuk CN 2

c. CN 3 ke CH 2

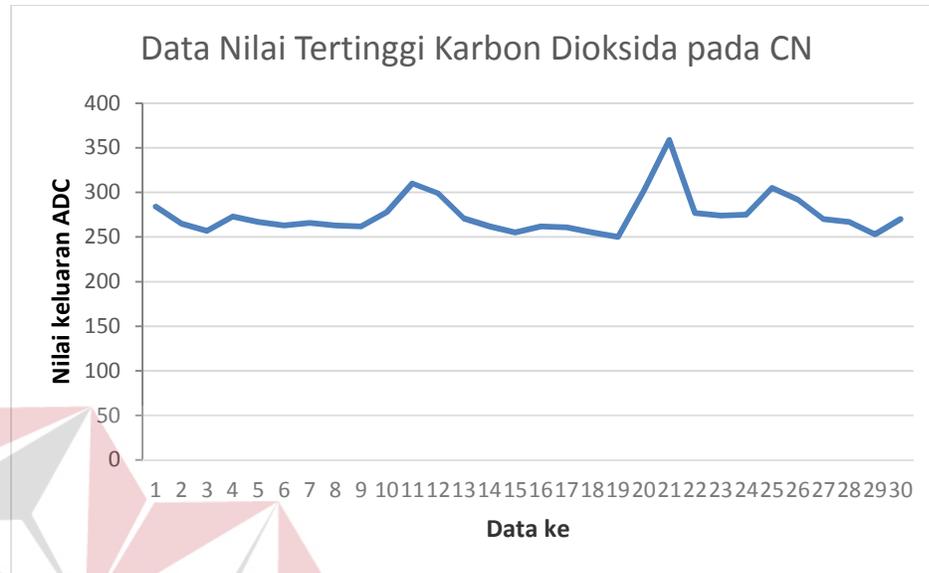


Gambar 4.24 Data tertinggi CO pada CN 3 untuk CH 2

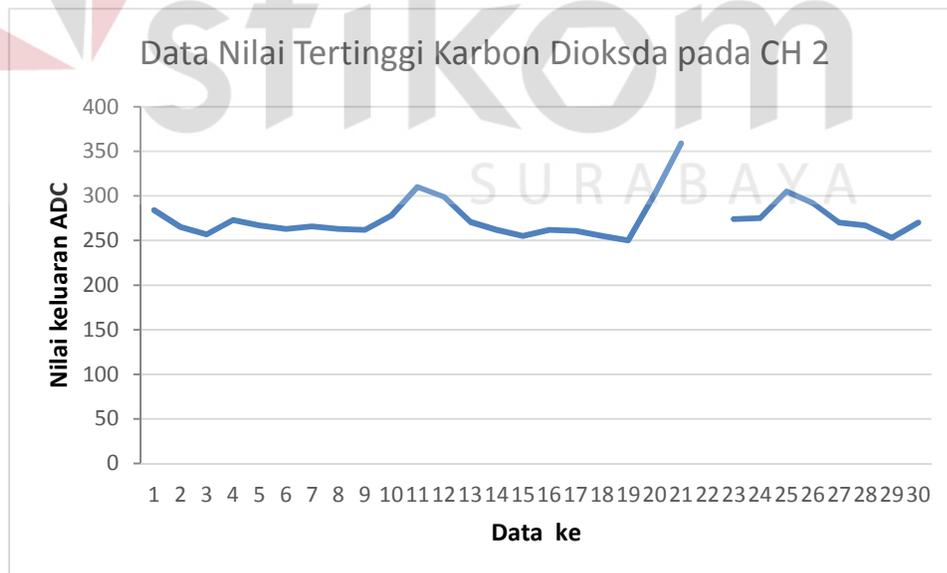


Gambar 4.25 Data tertinggi CO pada CH 2 untuk CN 3

d. CN 4 ke CH 2



Gambar 4.26 Data tertinggi CO₂ pada CN 4 untuk CH 2



Gambar 4.27 Data tertinggi CO₂ pada CN 4 untuk CH 2

Tabel 4.1 Hasil analisa data masing-masing CN ke CH

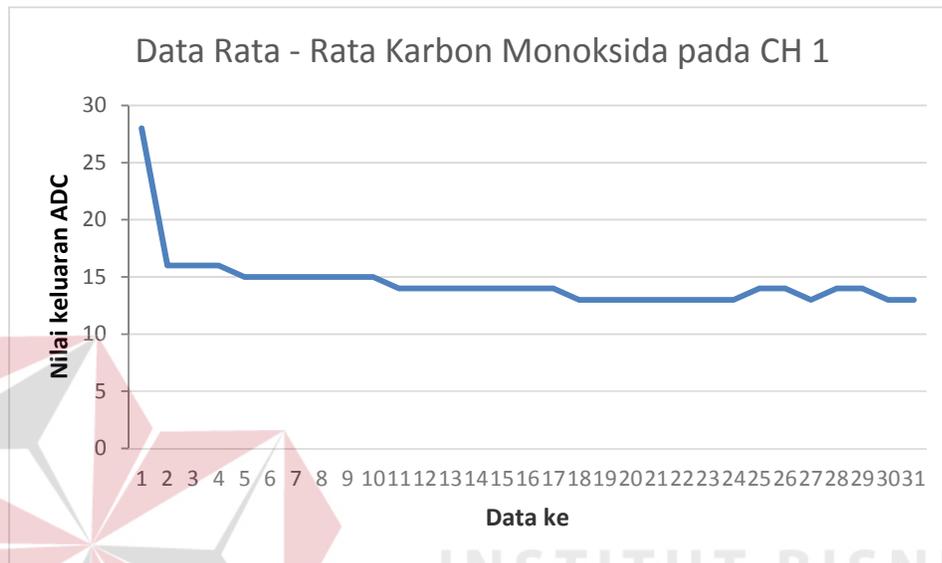
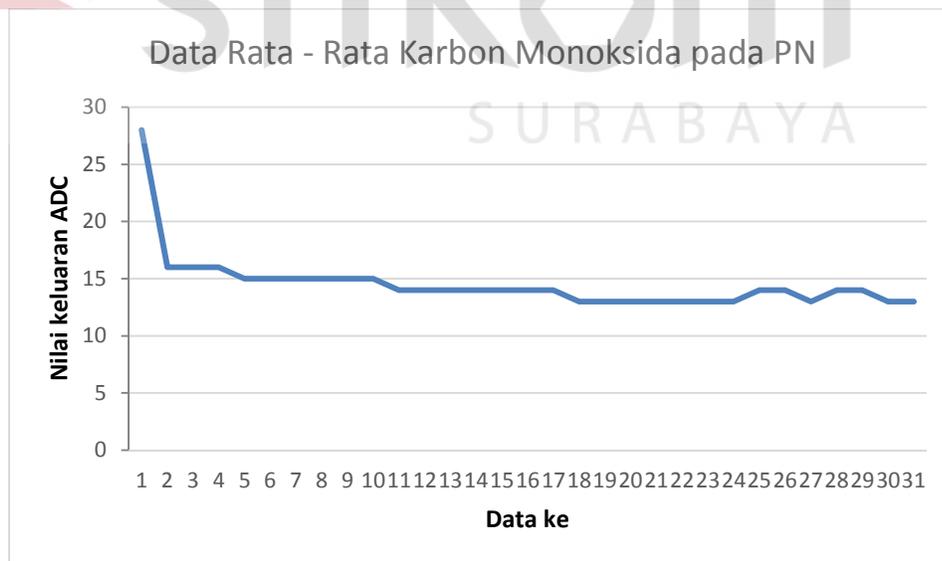
Node Asal	Node Tujuan	Delay	Paket Loss
CN 1	CH 1	2 detik	3.33%
CN 2	CH 1	2 detik	3.33%
CN 3	CH 2	3 detik	3.33%
CN 4	CH 2	3 detik	3.33%
Rata - rata		2.5 detik	3.33%

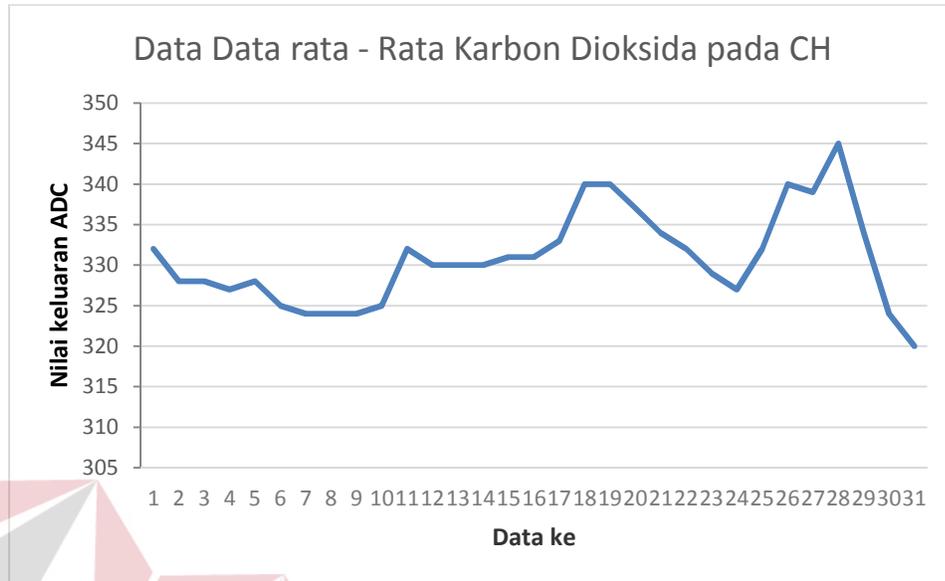
Dari analisa data dari masing – masing CN ke CH dapat dilihat pada table di atas bahwa rata-rata paket *loss* sangat kecil dan dalam katagori bagus. Hal tersebut di dapat dari 1 data yang hilang dari 30 data sehingga dapat dihitung sebagai berikut $1 / 30 * 100\% = 3.33\%$. Delay transmisi dari masing – masing CN sehingga menuju ke CH secara rata – rata 2.5 detik. Hal ini menunjukkan bahwa data yang di terima masih bisa di toleransi, mengingat lama waktu pengiriman adalah 1 menit, sehingga delay 2.5 detik tidak terlihat oleh user.

Pada grafik gambar 4.20 sampai gambar 4.27 terdapat perbedaan grafik antara grafik pengiriman data dan grafik penerimaan data, hal ini dikarenakan terdapat *noise* pada saat data dikirim. Dan apabila dilihat dari grafik yang tampil, noise tersebut mengakibatkan pengulangan penerimaan data yang sama. Hal tersebut dapat dilihat pada gambar 4.16 sampai dengan 4.19 terdapat pada *text highlight* berwarna kuning berupa data *double* , sehingga mengakibatkan grafik penerimaan dan pengiriman berbeda.

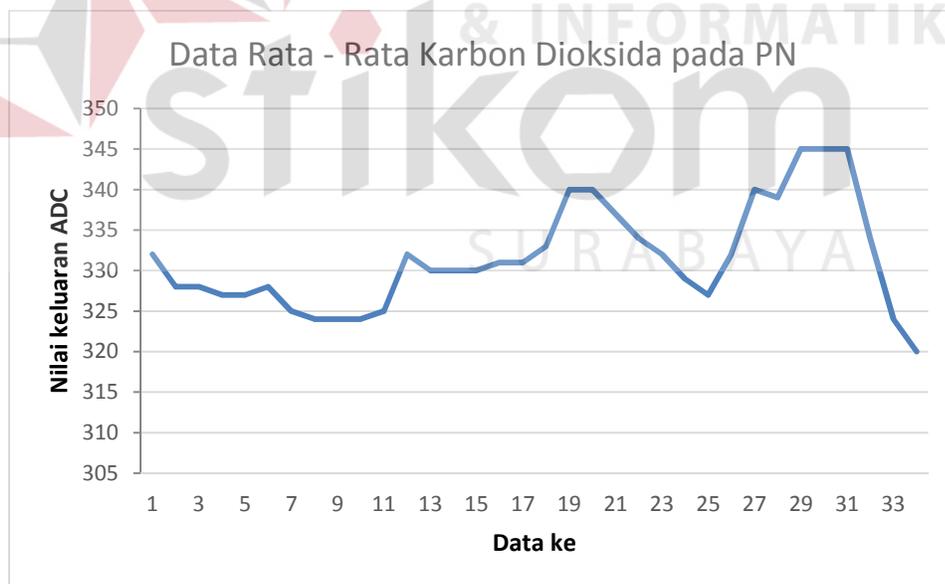
6. Data dari *Cluster Head* ke *Parent Node*

a. CH 1 ke PN

**Gambar 4.28** Data rata – rata CO pada CH 1**Gambar 4.29** Data rata – rata CO pada PN

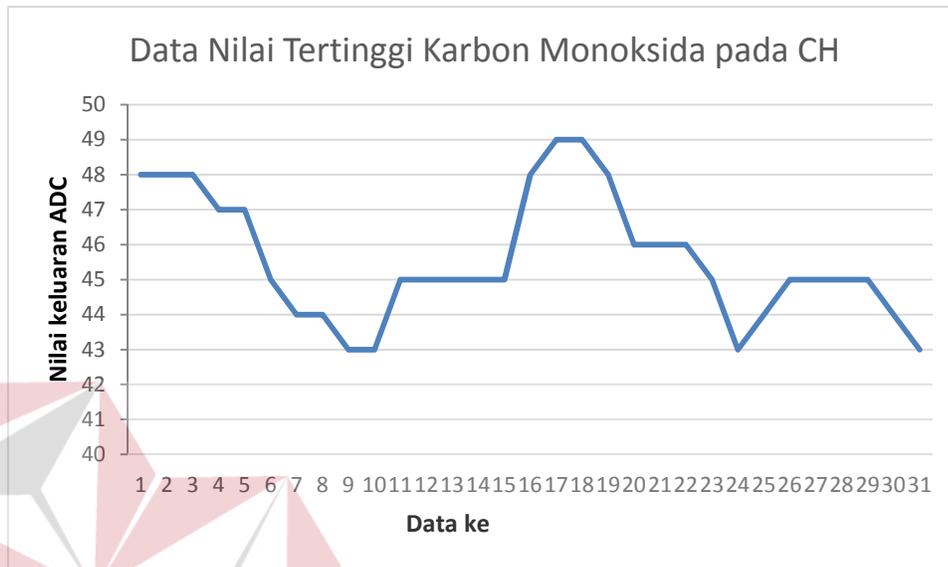


Gambar 4.30 Data rata – rata CO₂ pada CH 1

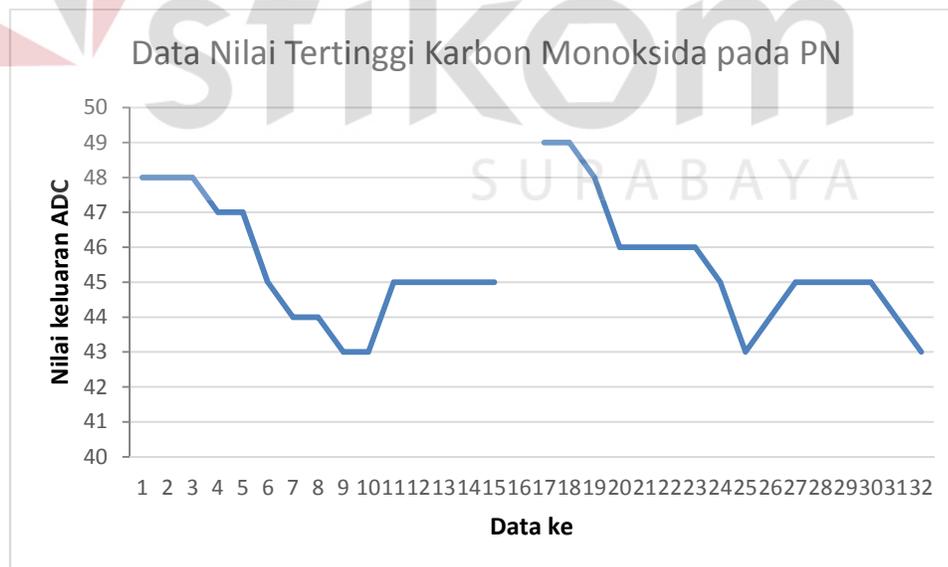


Gambar 4.31 Data rata – rata CO₂ pada PN

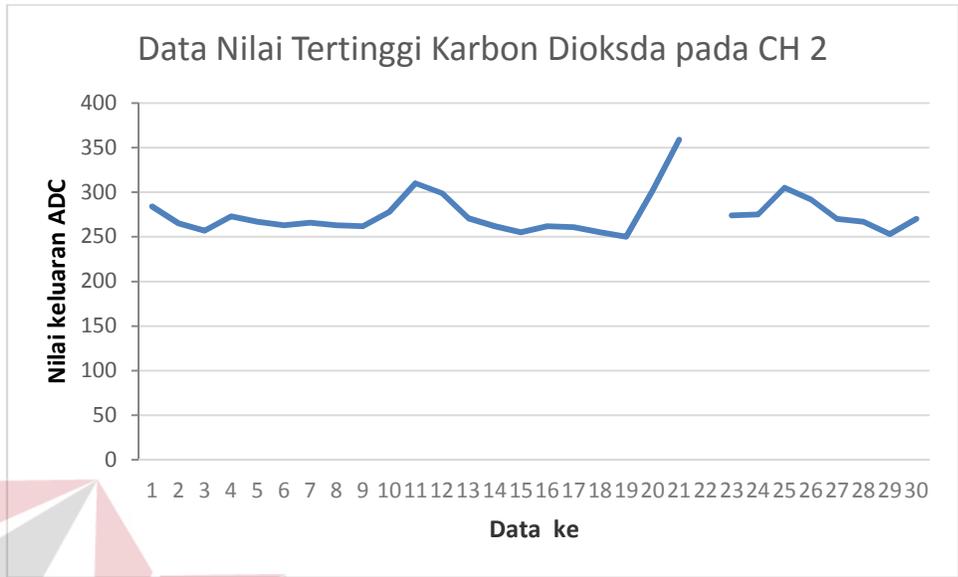
b. CH 2 ke PN



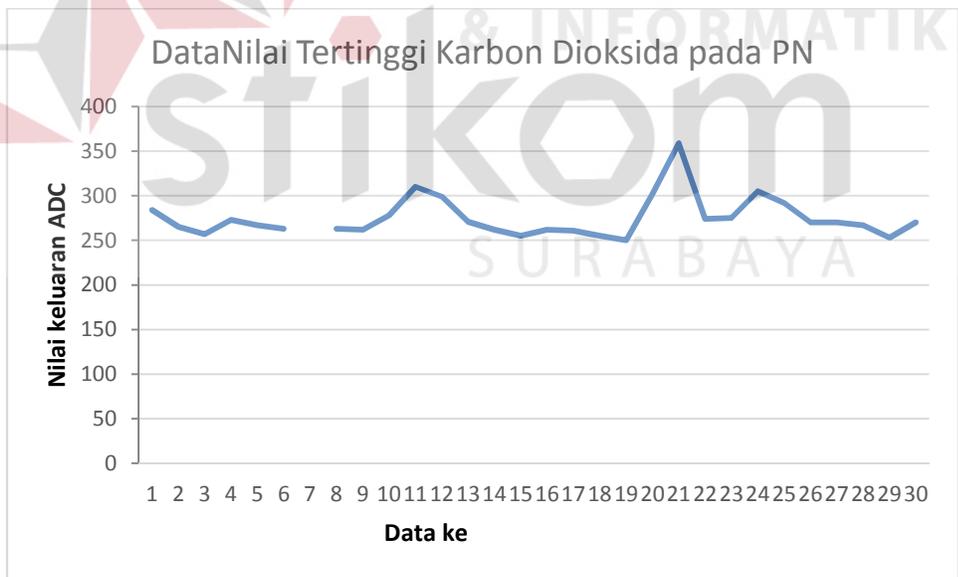
Gambar 4.32 Data tertinggi CO pada CH 2



Gambar 4.33 Data tertinggi CO pada PN



Gambar 4.34 Data tertinggi CO₂ pada CH 2



Gambar 4.35 Data tertinggi CO₂ pada PN

Tabel 4.2 Hasil analisa data dari CH ke PN

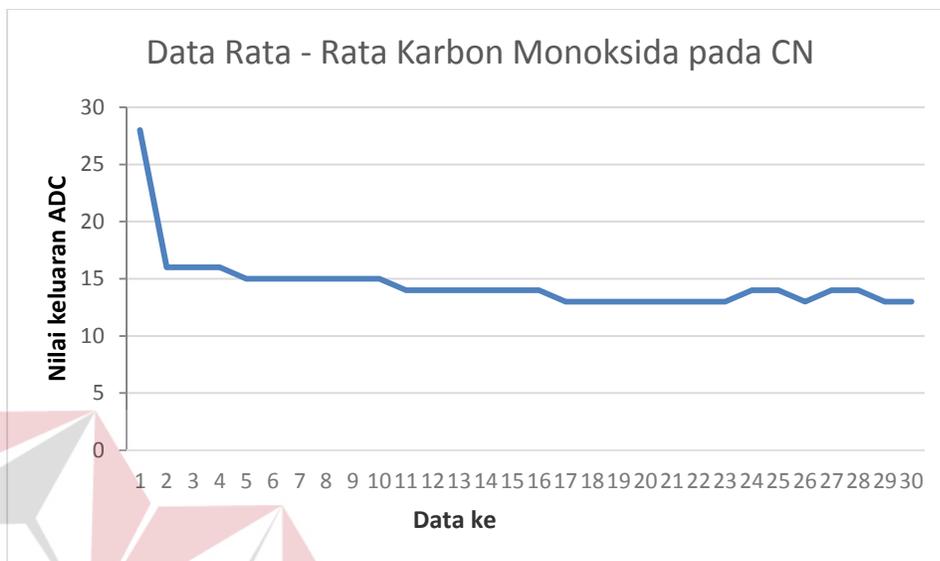
Node Asal	Node Tujuan	Delay	Paket Loss
CN 1 pada CH 1	PN	2 detik	0%
CN 2 pada CH 1	PN	2 detik	10%
CN 3 pada CH 2	PN	2 detik	6.67%
CN 4 pada CH 2	PN	3 detik	6.67%
Rata – rata		2.25 detik	5.83%

Dari analisa data dari CH ke PN dapat dilihat pada tabel di atas bahwa paket *loss* kecil yaitu dengan rata – rata sebesar 5.83%. Hal ini menunjukkan bahwa paket *loss* yang didapat termasuk dalam kategori bagus, sesuai dengan teori pada tabel 2.3. Komunikasi antara CH sampai dengan PN ini menggunakan xbee. Delay pengirimanpun juga memiliki waktu yang relative sama, yaitu antara 2-3 detik, hal ini menunjukkan bahwa waktu penerimaan data masih bisa ditoleransi, mengingat lama waktu pengiriman setiap data adalah 1 menit, sehingga *delay* tidak terlihat signifikan oleh *user*.

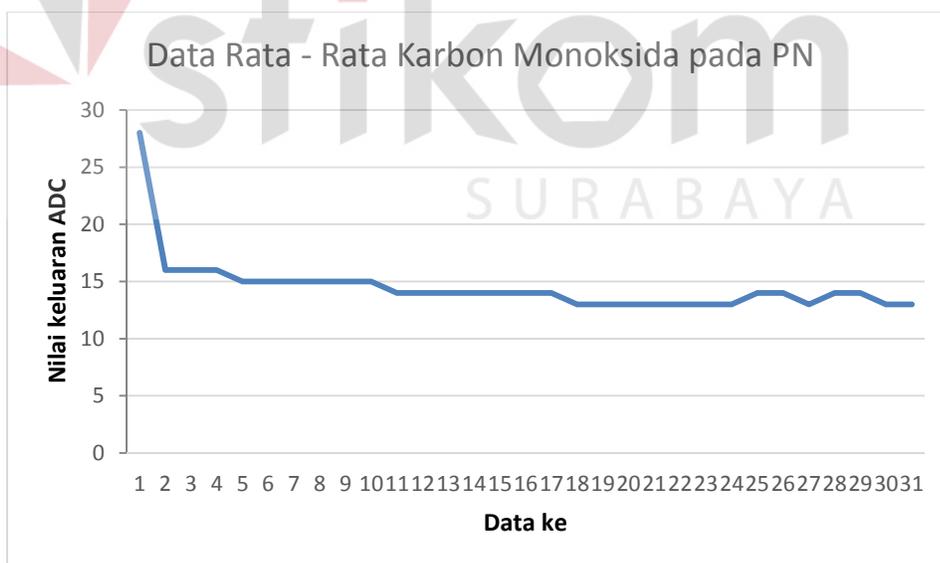
Pada grafik gambar 4.28 sampai gambar 4.35 terdapat perbedaan grafik antara grafik pengiriman data dan grafik penerimaan data, hal ini dikarenakan terdapat *noise* pada saat data dikirim. Dan apabila dilihat dari grafik yang tampil, *noise* tersebut mengakibatkan pengulangan penerimaan data yang sama. Hal tersebut dapat dilihat pada gambar 4.16 sampai dengan 4.19 terdapat pada *text highlight* berwarna kuning berupa data *double* , sehingga mengakibatkan grafik penerimaan dan pengiriman berbeda.

7. Data dari masing – masing *Child Node* ke *Parent Node*

a. CN 1 ke PN

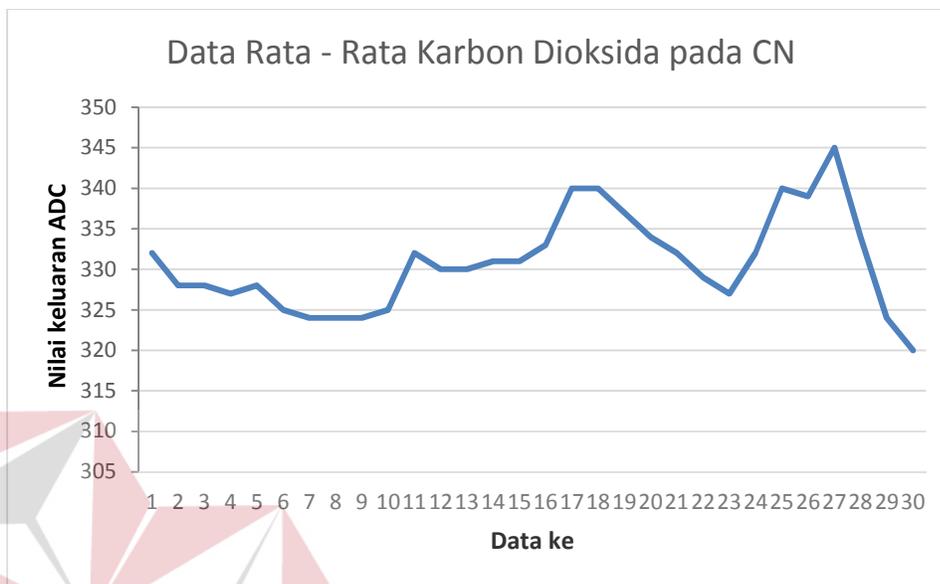


Gambar 4.36 Data rata-rata CO pada CN 1 untuk PN

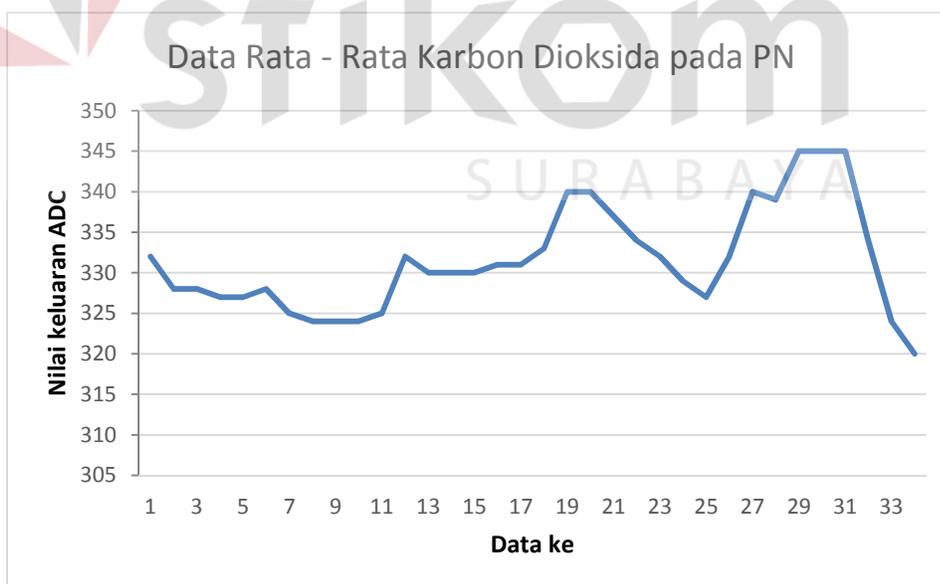


Gambar 4.37 Data rata-rata CO pada PN untuk CN 1

b. CN 2 ke PN

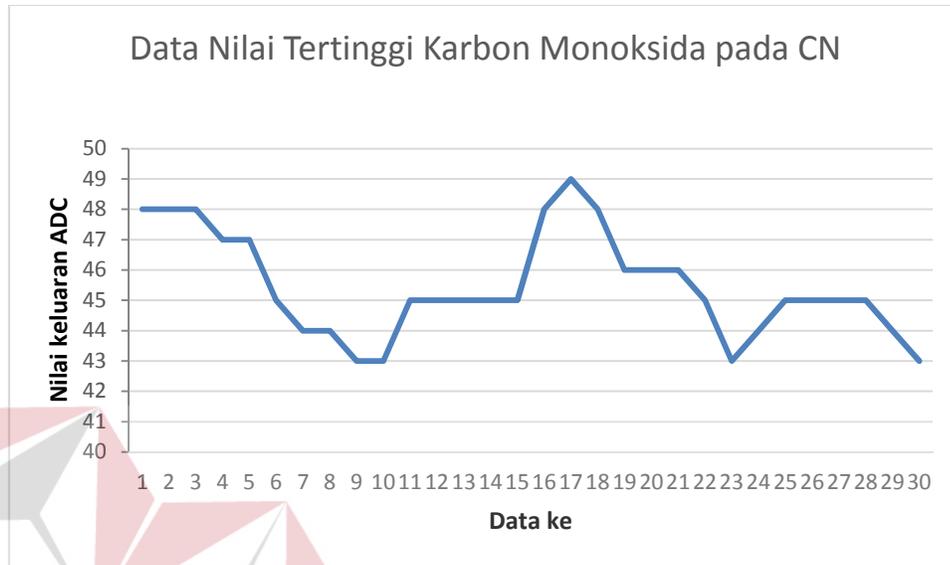


Gambar 4.38 Data rata-rata CO₂ pada CN 2 untuk PN

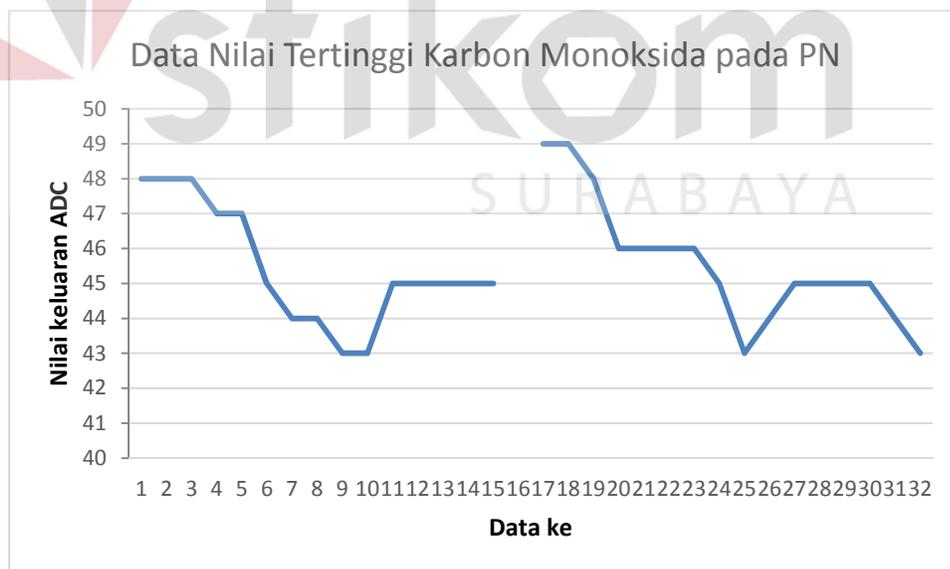


Gambar 4.39 Data rata-rata CO₂ pada PN untuk CN 2

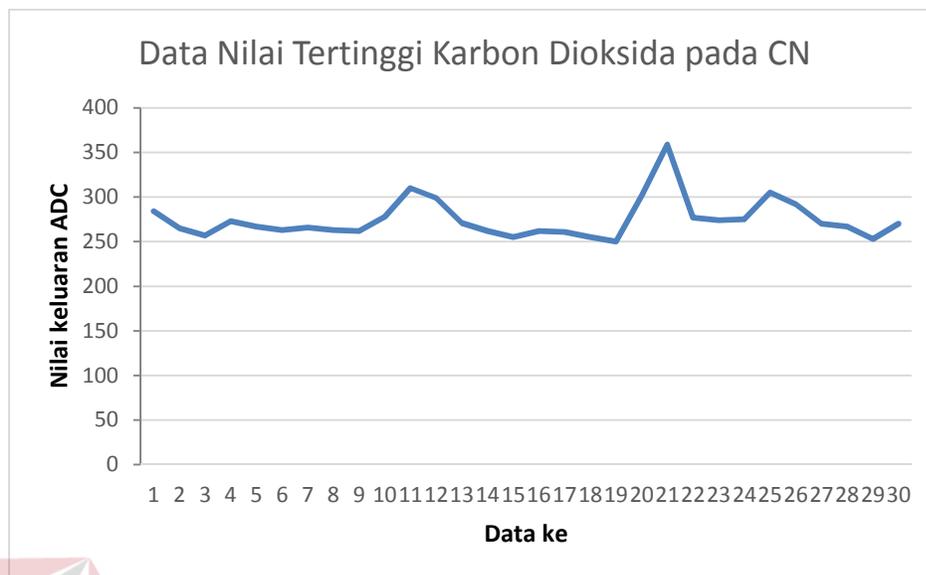
c. CN 3 ke PN



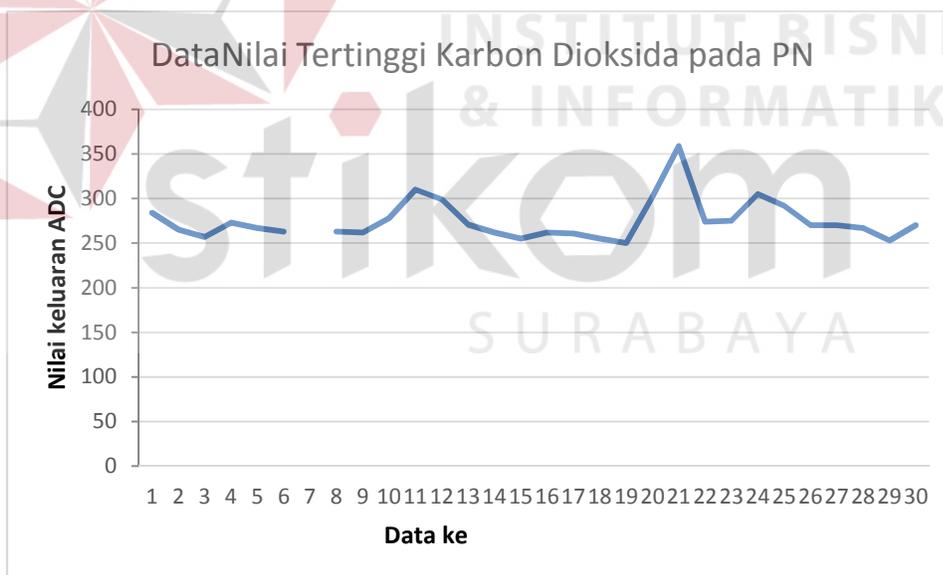
Gambar 4.40 Data tertinggi CO pada CN 3 untuk PN



Gambar 4.41 Data tertinggi CO pada PN untuk CN 3



Gambar 4.42 Data tertinggi CO₂ pada CN 4 untuk PN



Gambar 4.43 Data tertinggi CO₂ pada PN untuk CN 4

Tabel 4.3 Hasil analisa data masing-masing CN ke PN

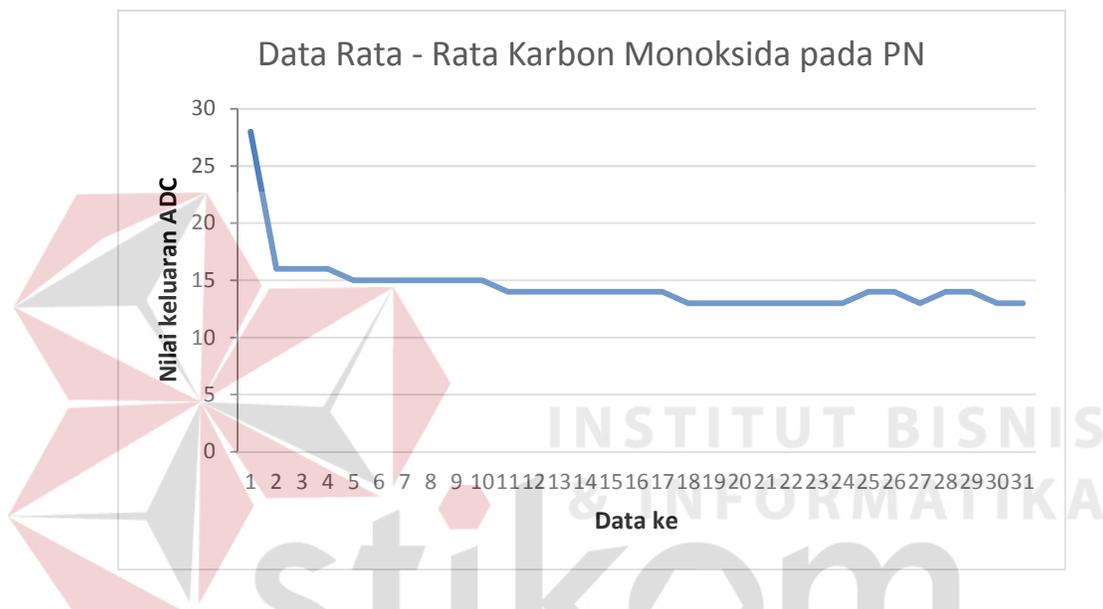
Node Asal	Node Tujuan	Delay	Paket Loss
CN 1	PN	4 detik	3.33%
CN 2	PN	4 detik	13.33%
CN 3	PN	5 detik	10%
CN 4	PN	6 detik	10%
Rata – rata		4.75 detik	9.16%

Dari analisa data dari CN ke PN dapat dilihat pada tabel di atas bahwa paket *loss* kecil yaitu dengan rata – rata sebesar 9.16%. yang di dapat dari penambahan paket *loss* dari CN ke CH ditambahkan dengan paket *loss* dari CH ke PN kemudian hasil penambahan tersebut dibagi 30 kemudian hasilnya dikalikan 100%. Dari hasil rata – rata tersebut hal ini menunjukkan bahwa paket *loss* yang didapat termasuk dalam kategori bagus, sesuai dengan teori pada tabel 2.3. Komunikasi antara CN sampai dengan PN ini menggunakan xbee. Delay pengirimanpun juga memiliki waktu yang *relative* sama, yaitu antara 4 - 5 detik dan untuk perhitungan *delay*pun sama dengan menghitung paket *loss* dengan menambahkan *delay* dari CN ke CH ditambahkan dengan *delay* dari CH ke PN . Hal ini menunjukkan bahwa waktu penerimaan data masih bisa ditoleransi, mengingat lama waktu pengiriman setiap data adalah 1 menit, sehingga *delay* tidak terlihat signifikan oleh *user*.

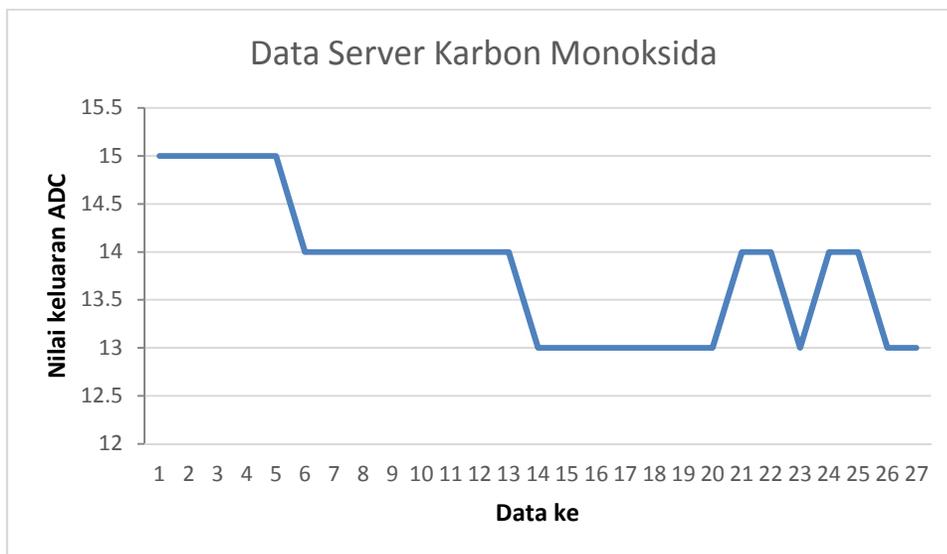
Pada grafik gambar 4.36 sampai gambar 4.43 terdapat perbedaan grafik antara grafik pengiriman data dan grafik penerimaan data, hal ini dikarenakan terdapat *noise* pada saat data dikirim. Dan apabila dilihat dari grafik yang tampil, *noise* tersebut mengakibatkan pengulangan penerimaan data yang sama. Hal tersebut

dapat dilihat pada gambar 4.16 sampai dengan 4.19 terdapat pada *text highlight* berwarna kuning berupa data *double* , sehingga mengakibatkan grafik penerimaan dan pengiriman berbeda.

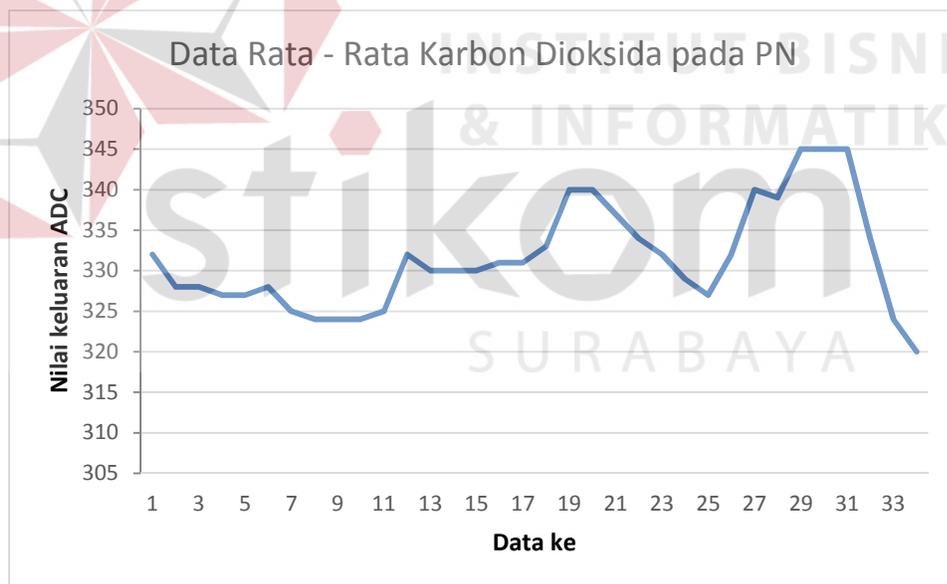
8. Data dari *Parent Node* ke *server*



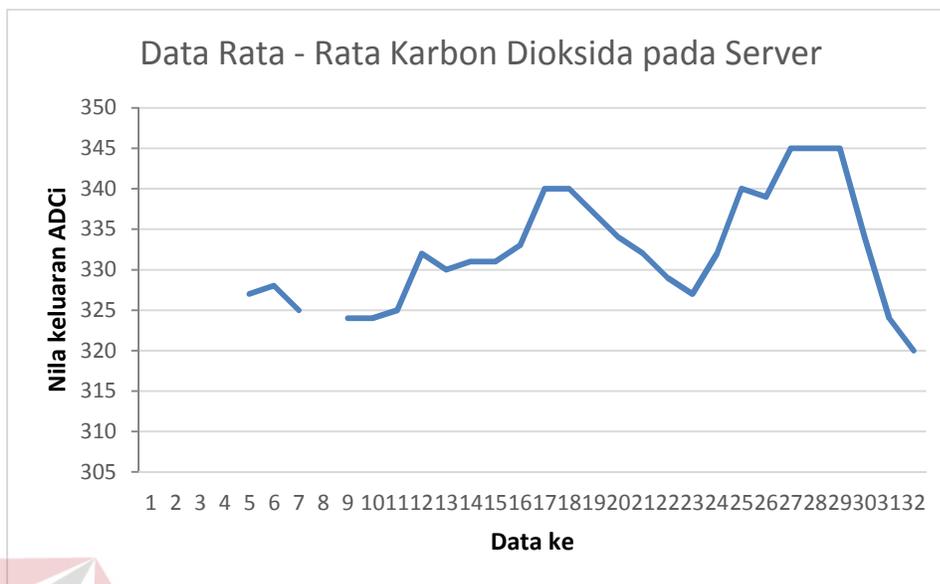
Gambar 4.44 Data rata - rata CO pada PN untuk *server*



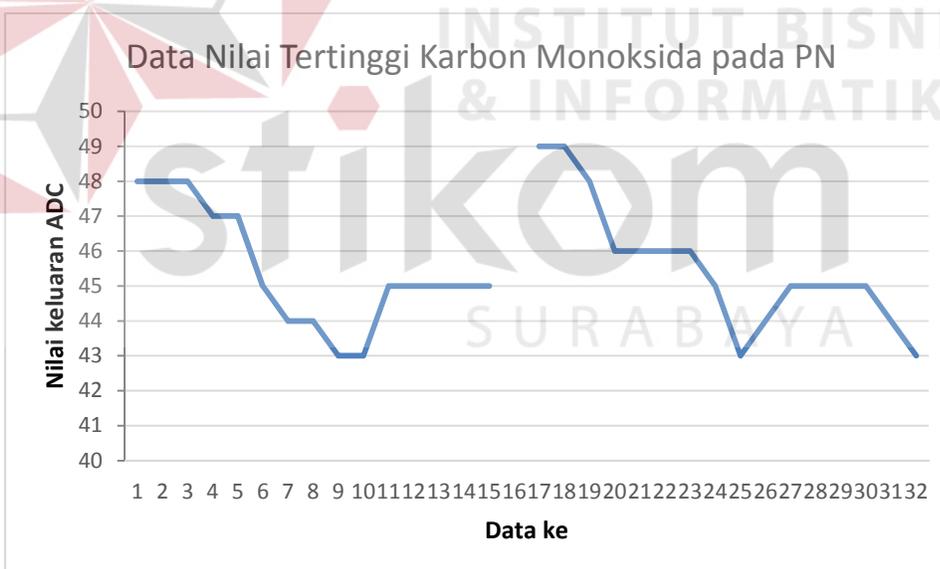
Gambar 4.45 Data rata - rata CO pada *server* untuk PN



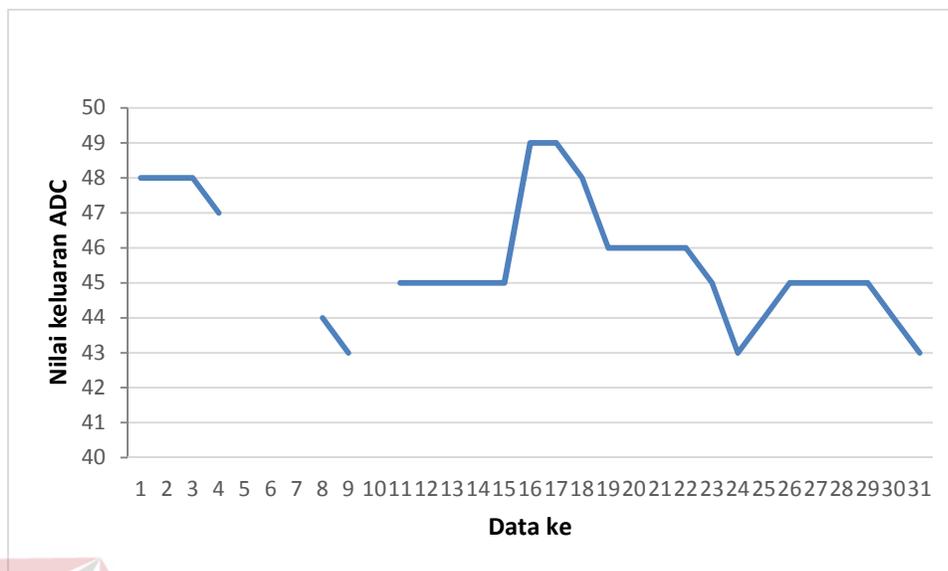
Gambar 4.46 Data rata - rata CO₂ pada PN untuk *server*



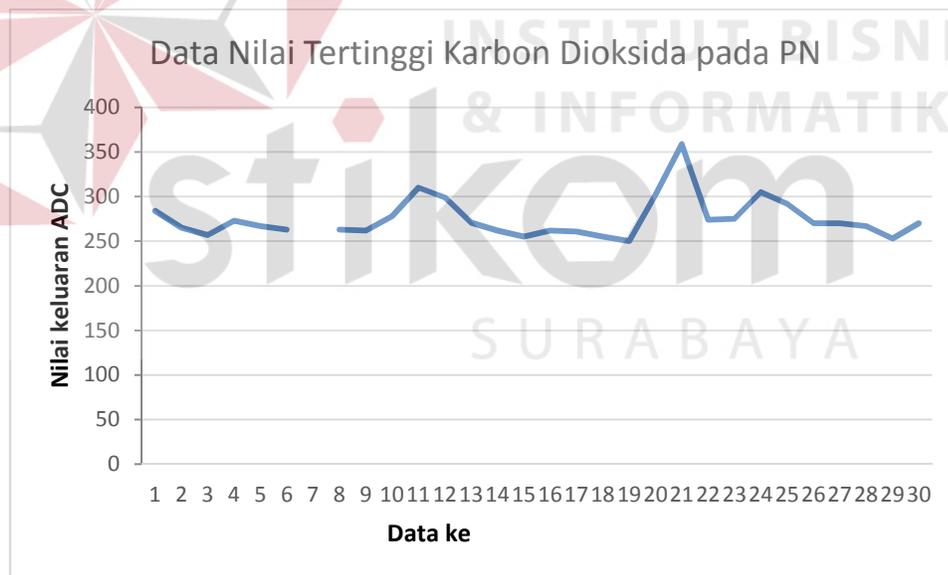
Gambar 4.47 Data rata - rata CO₂ pada *server* untuk PN



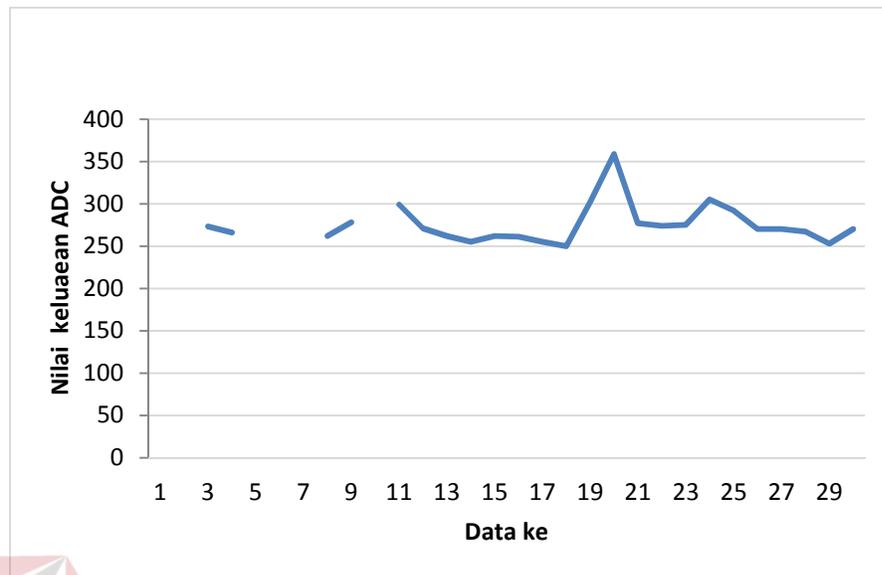
Gambar 4.48 Data tertinggi CO pada PN untuk *server*



Gambar 4.49 Data tertinggi CO pada *server* untuk PN



Gambar 4.50 Data tertinggi CO₂ pada PN untuk *server*



Gambar 4.51 Data tertinggi CO₂ pada *server* untuk PN

Tabel 4.4 Hasil analisa data PN ke *server*

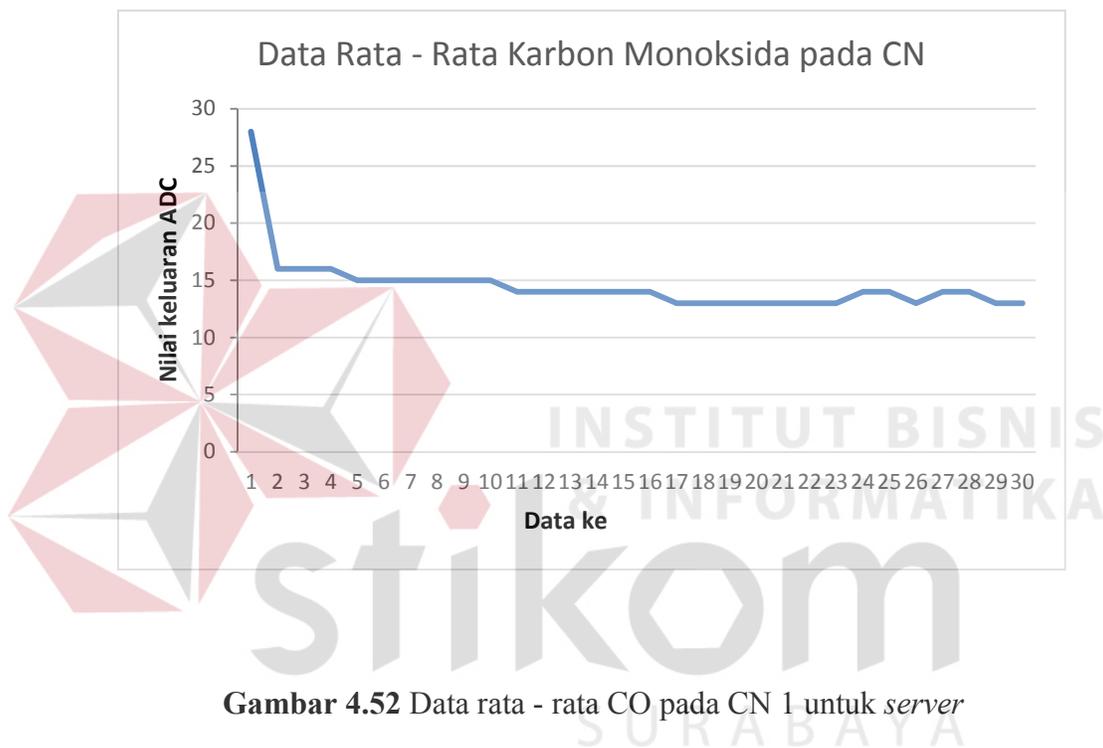
Node Asal	Node Tujuan	Delay	Paket Loss
CN 1 pada PN	server	1 detik	13.33%
CN 2 pada PN	server	0 detik	13.33%
CN 3 pada PN	server	0 detik	10%
CN 4 pada PN	server	3 detik	23.33%
Rata - rata		1 detik	15%

Dari analisa data dari CN pada PN ke *server* dapat dilihat pada tabel di atas bahwa paket *loss* cukup tinggi yaitu dengan rata – rata sebesar 15%. Hal ini menunjukkan bahwa paket *loss* yang didapat termasuk dalam kategori sedang , sesuai dengan teori pada tabel 2.3. Komunikasi antara CN pada PN ke *server* ini menggunakan *wifi*. Meskipun paket *loss* masuk dalam katagori sedang namun *delay*

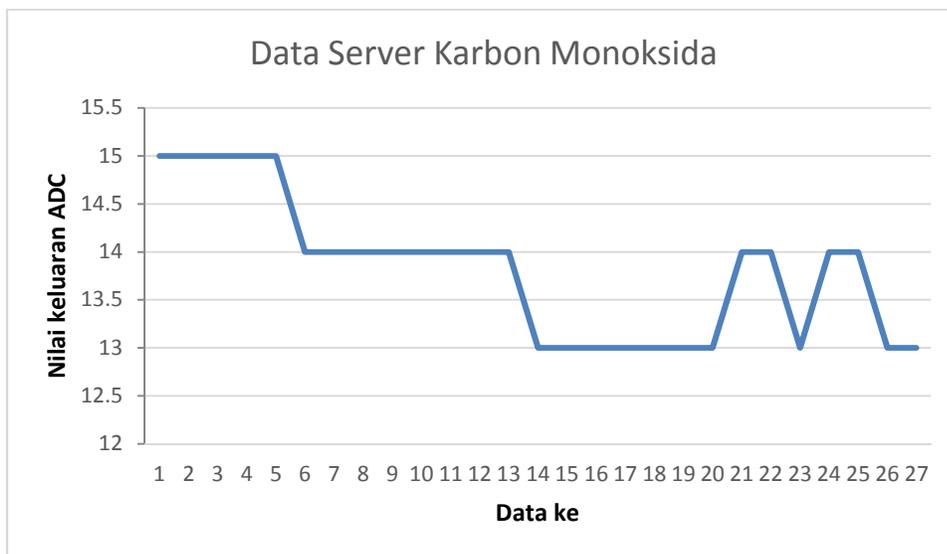
relative kecil yaitu rata – rata 1 detik, dan pada transmisi data CN 2 dan CN 3 pada PN ke *server* mengalami *delay* 0 detik dan itu bisa dikatakan sempurna.

9. Data dari masing – masing *Child Node* ke *server*

a. CN 1 ke *server*

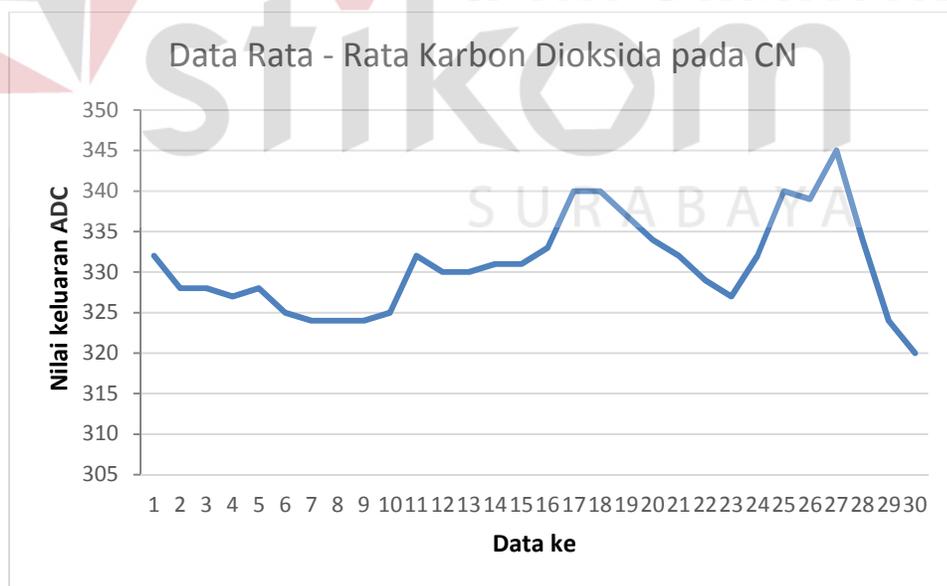


Gambar 4.52 Data rata - rata CO pada CN 1 untuk *server*

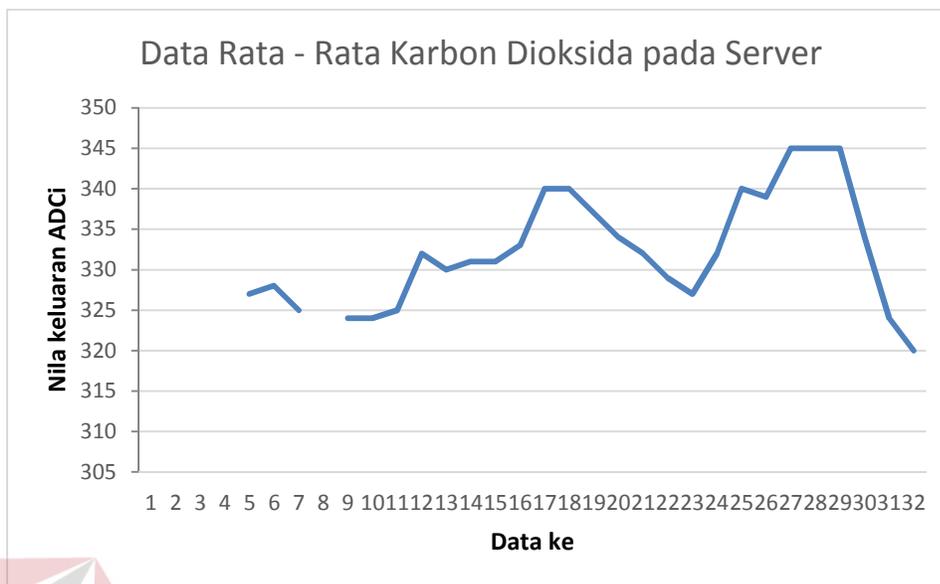


Gambar 4.53 Data rata - rata CO pada *server* untuk CN 1

b. CN 2 ke *server*

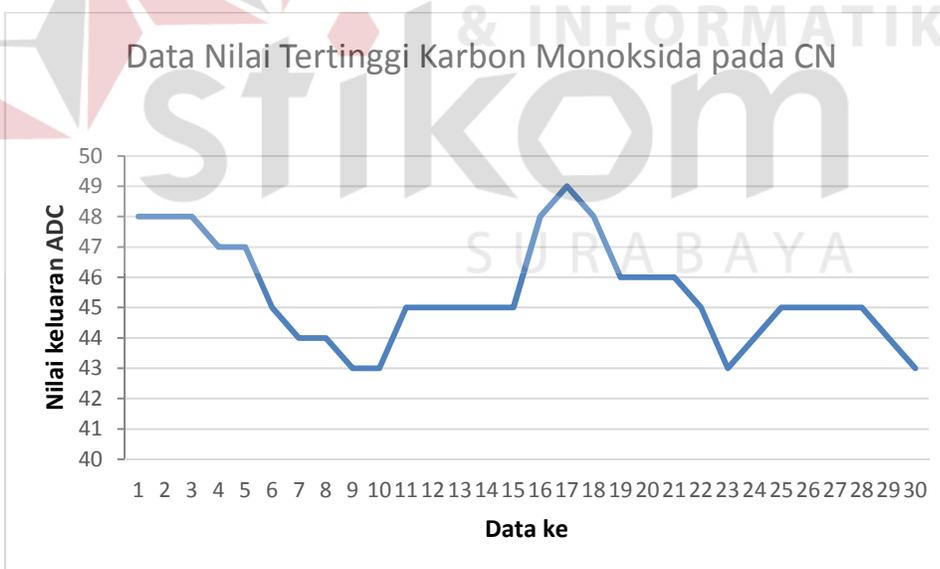


Gambar 4.54 Data rata - rata CO₂ pada CN 2 untuk *server*

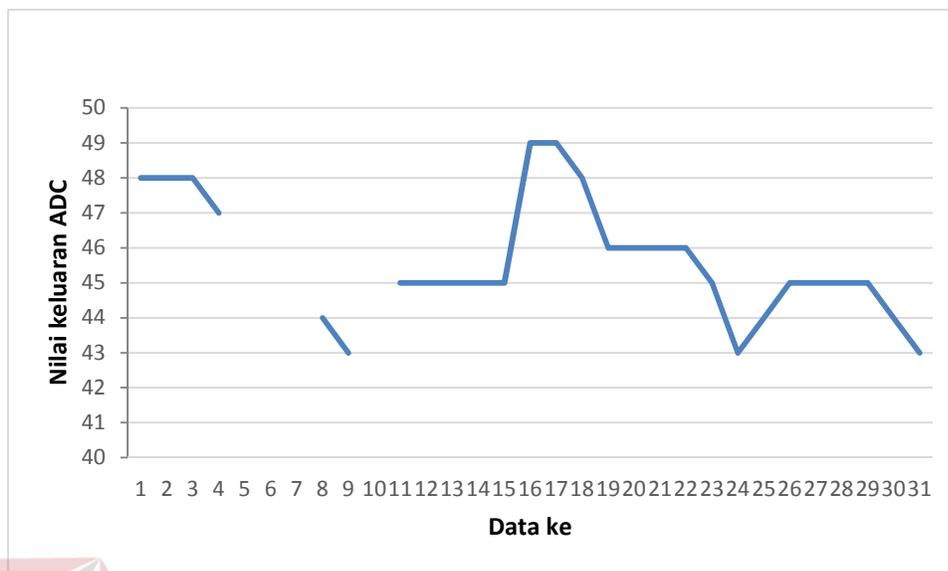


Gambar 4.55 Data rata - rata CO₂ pada *server* untuk CN 2

c. CN 3 ke *server*

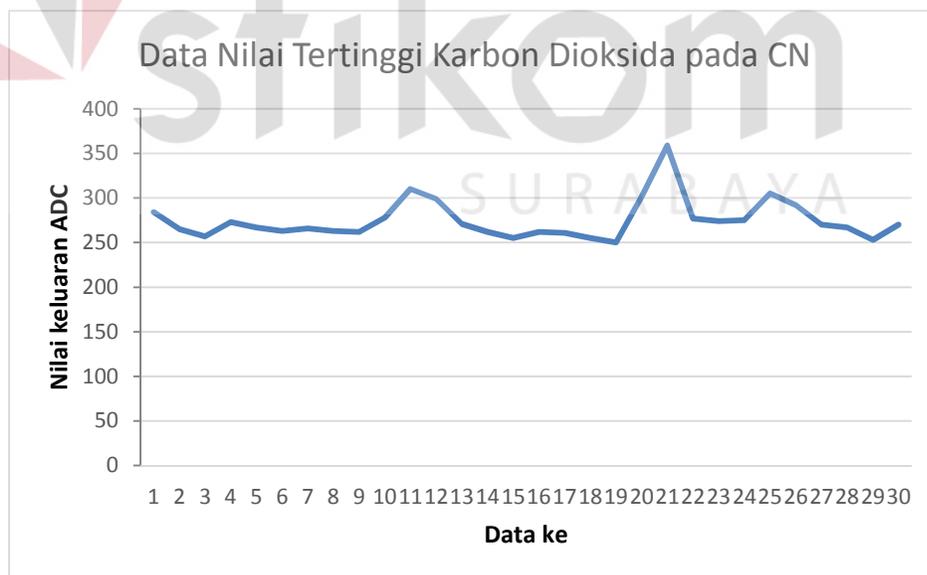


Gambar 4.56 Data tertinggi CO pada CN 3 untuk *server*

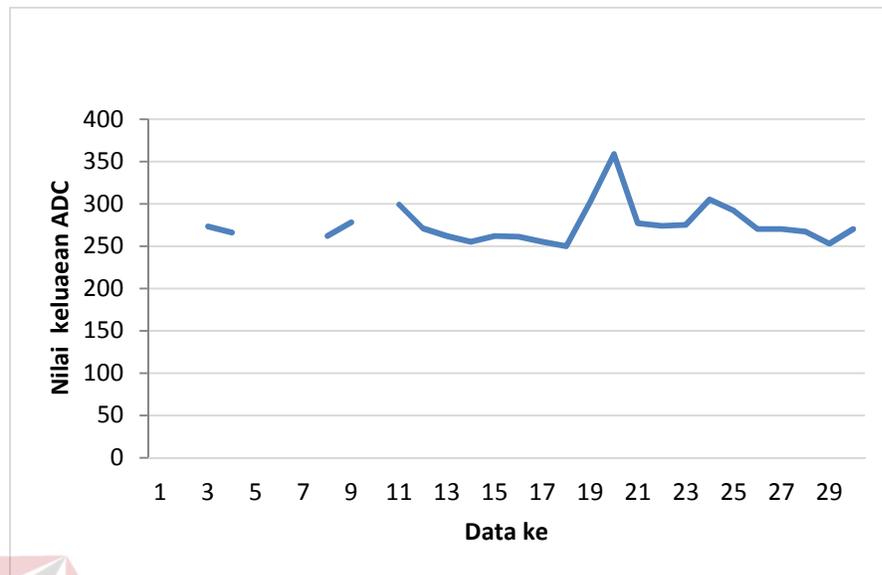


Gambar 4.57 Data tertinggi CO pada *server* untuk CN 3

d. CN 4 ke *server*



Gambar 4.58 Data tertinggi CO₂ pada CN 4 untuk *server*



Gambar 4.51 Data tertinggi CO₂ pada *server* untuk CN 4

Tabel 4.5 Hasil analisa data dari CN ke *server*

Node Asal	Node Tujuan	Delay	Paket loss
CN 1	server	5 detik	16.67%
CN 2	server	4 detik	26.67%
CN 3	server	5 detik	23.33%
CN 4	server	8 detik	33.33%
Rata - rata		5.5 detik	25%

Dari analisa data dari CN ke *server* dapat dilihat pada tabel di atas bahwa paket *loss* dalam katagori buruk sesuai dengan teori pada tabel 2.3. yaitu dengan rata – rata sebesar 25% dengan jumlah *loss* dan *delay* tertinggi pada CN 4 ke *server* yaitu dengan *delay* 8 detik terdapat 33.33% paket *loss*. Hal ini bisa disebabkan karena *node* berada pada daerah dengan *noise* yang relative lebih banyak (banyak lalu lalang kendaraan).

Pada transmisi data dari CN menuju *server* terdapat *delay* yang panjang dikarenakan *delay* merupakan penjumlahan *delay* yang berasal dari CN menuju CH, CH menuju PN, serta *delay* dari PN menuju *server*. Karena terdapat *delay* pada setiap pengiriman dari *node* menuju *node*. Dan begitu juga dengan cara untuk mendapatkan data *loss*.

Delay pengirimanpun juga memiliki waktu yang relative sama, yaitu rata – rata 5.5 detik, hal ini menunjukkan bahwa waktu penerimaan data masih bisa ditoleransi, mengingat lama waktu pengiriman setiap data adalah 1 menit, sehingga *delay* tidak terlihat signifikan oleh *user*.

