

## BAB 6 PENGUJIAN DAN ANALISIS

Bab ini membahas tentang hasil pengujian dan analisis sistem yang telah direalisasikan.

### 6.1 Pengujian dan Analisis

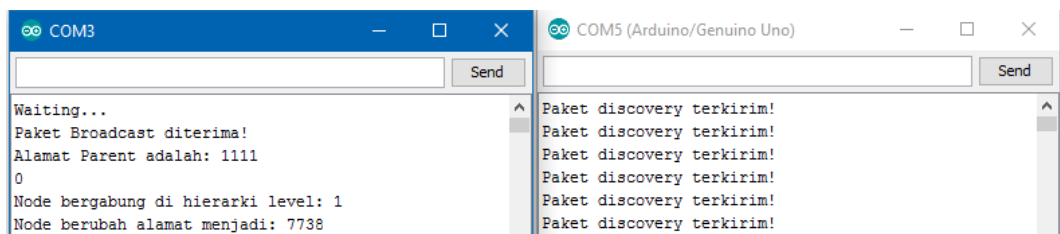
Pengujian dilakukan dengan menyalakan 2 *node client* yang terkoneksi dengan *node base* secara *wireless* menggunakan modul komunikasi NRF24L01. Adapun kebutuhan yang perlu diuji dijelaskan pada sub bab 6.1.1, sub bab 6.1.2, dan sub bab 6.1.3 beserta prosedur pengujiannya.

#### 6.1.1 Pengujian Protokol Penyetaraan Waktu TPSN

Pada penelitian ini digunakan TPSN sebagai protokol untuk sinkronisasi waktu antar node. Proses sinkronisasi waktu pada TPSN dibagi menjadi dua tahapan, *Discovery Phase* dan *Synchronization Phase*. Untuk prosedur pengujian, pertama pada sisi *node base* sebagai *parent* dinyalakan terlebih dahulu dengan membuka *Serial Monitor* pada aplikasi Arduino. Kedua, pada sisi *node client* dinyalakan dengan membuka *serial monitor*. Namun untuk sisi *client*, *serial monitor* dibuka secara bergantian setelah salah satu *client* telah sukses mendapatkan waktu yang setara dengan *node base*. Lalu, dilanjutkan menyalakan *node client* lainnya.

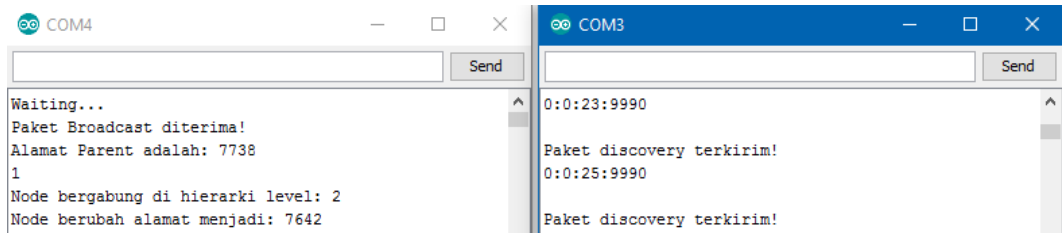
##### 1. *Discovery Phase*

Pada tahapan ini setiap *node client* akan mendapatkan alamat baru secara acak yang terdiri dari 4 karakter angka (bilangan bulat). Prosedur pengujian *Discovery Phase* dapat dilihat pada **Gambar 6.1**. Hasil pengujian dapat dilihat pada **Tabel 6.1**.



**Gambar 6.1** Tampilan *Discovery Phase Node Base Ke Node Client 1*

**Gambar 6.1** di atas, merupakan tampilan *serial monitor node client 1* (kiri) dan *node base* (kanan), dimana *node base* membroadcast paket *discovery* dan berhasil diterima oleh salah satu *node client*. *Node client* tersebut berhasil mendapatkan level *node* dan alamat berupa 4 digit karakter angka. Alamat ini difungsikan sebagai pembeda untuk tiap *node client* yang aktif.



**Gambar 6.2** Tampilan *Discovery Phase Node Client 1* ke *Node Client 2*

**Gambar 6.2** di atas, adalah tampilan *discovery phase* dari *node client 1* (kanan) kepada *node client 2* (kiri). *Node client 1* yang awalnya telah mendapatkan level *node* dan alamat dari *node base* menjadi *node parent* agar *node client 2* sehingga membentuk sebuah level *node* hierarki dengan didapatkannya level *node* dan alamat berbeda.

Pengujian tahapan *discovery phase* dilakukan sebanyak 10 kali dan hasilnya bisa dilihat pada **Tabel 6.1** berikut.

**Tabel 6.1** Hasil Pengujian *Discovery Phase* Kedua *Node Client*

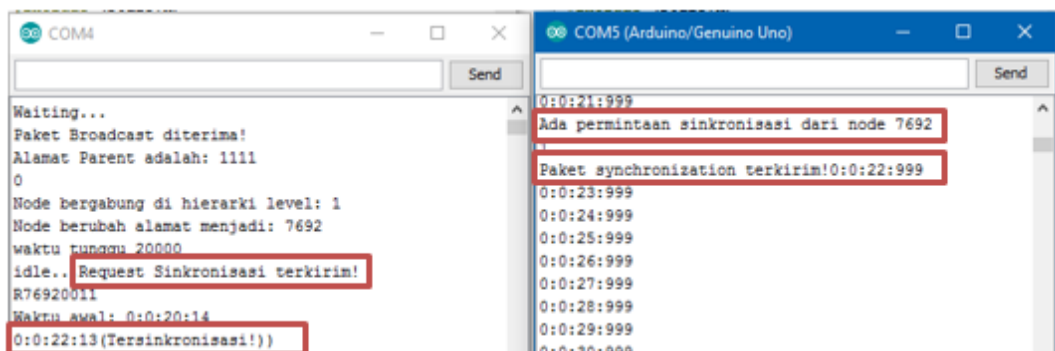
Pengujiann Ke-	Level Node	Alamat Parent	Alamat Node
1	1	1111	7739
	2	7739	7625
2	1	1111	7698
	2	7698	7518
3	1	1111	7672
	2	7672	7618
4	1	1111	7644
	2	7644	7654
5	1	1111	7642
	2	7642	7656
6	1	1111	7590
	2	7590	7697
7	1	1111	7592
	2	7592	7613
8	1	1111	7652
	2	7652	7697
9	1	1111	7684
	2	7684	7587

Pengujiann Ke-	Level Node	Alamat Parent	Alamat Node
10	1	1111	7654
	2	7654	7648

Dari hasil 10 kali pengujian bisa disimpulkan bahwa tahapan *Discovery Phase* pada *node client* bisa berjalan dengan baik. *Node client* bisa mendapatkan paket *discovery* dari *node base* dan mendapat level serta alamat. Proses aktivasi *node client* dilakukan secara bergantian, agar didapatkan level yang membentuk topologi hierarki berdasarkan konsep metode penyetaraan waktu TPSN.

## 2. *Synchronization Phase*

Tahapan pengujian ini bertujuan untuk mengamati keberhasilan masing-masing *node client* dalam melakukan proses sinkronisasi waktu terhadap *node base*. Awalnya satu *Node client* akan mengirim permintaan penyetaraan waktu terlebih dulu ke *node base* sebagai *parent*. Kemudian *Node base* menerima permintaan itu dan membalas permintaan penyetaraan waktu tersebut dengan mengirim waktu *node base* yang sedang berjalan. Prosedur pengujian *Synchronization Phase* dilihat di Gambar 6.3 dan hasil pengujian dijelaskan di Tabel 6.2.



**Gambar 6.3 Tampilan *Synchronization Phase Node Client 1***

Setelah sebelumnya *node client 1* (kiri) berhasil mendapatkan level *node* serta alamat dari *node base*, *node client 1* lalu melakukan permintaan sinkronisasi waktu kepada *node base* sebagai *parent* atau *root* dari *node client 1*. **Gambar 6.3** menunjukkan *node client 1* telah berhasil memiliki waktu yang sama dengan *node base*.

```

request synchronization from: 8533
0:0:24:961
Balasan Sinkron 500:00:24:96309216
0:0:24:999
0:0:25:999
0:0:26:999
0:0:27:999
0:0:28:999
0:0:29:999
0:0:30:999
0:0:32:0
0:0:32:999
0:0:33:000

```

```

Dari paket: 500:00:19:3570113554
Ukuran: 32 byte 256 bit
Delay total: 158, delay propagasi: 323
Jadwal: 4
Slot: 2
jadwal terpakai : 0

Request Sinkronisasi terkirim!
Alamat Node: 8533
Waktu awal: 0:0:31:102
0:0:24:733
(Tersinkronisasi!)

```

**Gambar 6.4** Tampilan *Synchronization Phase Node Client 2*

Setelah sebelumnya *node client 1* (kanan) berhasil sinkronisasi waktu dengan *node base*, lalu *node client 2* (kiri) akan melakukan permintaan sinkronisasi waktu kepada *node client 1* sebagai *parent*-nya. **Gambar 6.4** menunjukkan bahwa *node client 2* telah berhasil mempunyai waktu yang setara dengan *node client 1*.

Hasil pengujian tahapan *synchronization phase* bisa dilihat pada **Tabel 6.2** berikut.

**Tabel 6.2** Hasil Pengujian *Synchronization Phase*

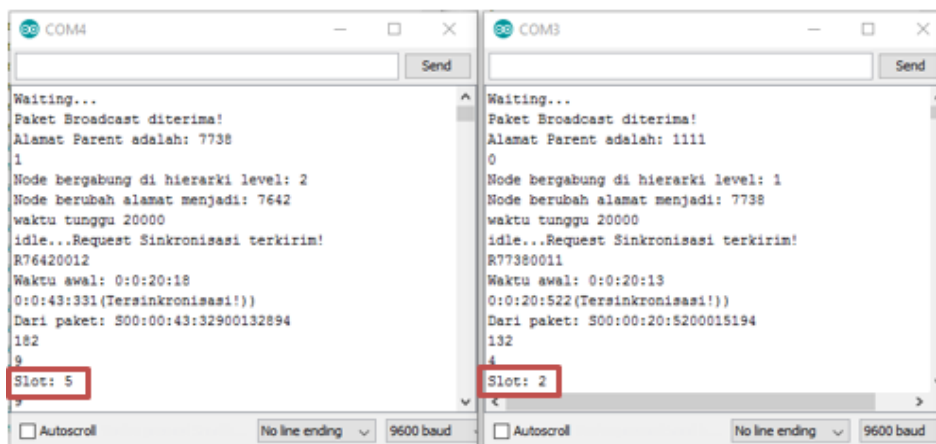
Pengujian Ke-	Level Node	Waktu Awal (j:m:d:md)	Waktu Setara (j:m:d:md)
1	1	0:0:20:13	0:0:20:52
	2	0:0:20:18	0:0:43:33
2	1	0:0:20:14	0:0:22:13
	2	0:0:20:18	0:0:45:93
3	1	0:0:20:14	0:0:20:59
	2	0:0:20:28	0:0:43:00
4	1	0:0:20:13	0:0:21:15
	2	0:0:20:19	0:0:43:56
5	1	0:0:20:13	0:0:21:26
	2	0:0:20:19	0:0:44:29
6	1	0:0:40:12	0:0:40:84
	2	0:0:20:18	0:1:03:16
7	1	0:0:20:13	0:0:20:77
	2	0:0:20:18	0:0:43:62
8	1	0:0:20:13	0:0:21:20

Pengujian Ke-	Level <i>Node</i>	Waktu Awal (j:m:d:md)	Waktu Setara (j:m:d:md)
	2	0:0:20:18	0:0:44:19
9	1	0:0:20:14	0:0:21:35
	2	0:0:20:18	0:0:45:10
10	1	0:0:20:13	0:0:21:30
	2	0:0:20:18	0:0:43:29

Dari hasil 10 kali pengujian *Synchronization Phase* dapat disimpulkan bahwa kedua *node client* telah berhasil mendapatkan waktu yang setara dengan rata-rata waktu yang dibutuhkan untuk proses penyetaraan waktu pada kedua client dibawah 50 detik.

### 6.1.2 Pengujian Protokol TDMA

Pada penelitian ini protokol *TDMA* berfungsi untuk melakukan pengiriman data menurut slot waktu yang telah ditentukan secara *random*. Pada penelitian ini, *node client* menerima slot waktu pengiriman ketika proses *Synchronization Phase*. Slot waktu pengiriman bisa dilihat pada Gambar 6.5.



**Gambar 6.5 Slot Waktu Pengiriman Data *Node Client***

Pengiriman data pada penelitian ini dilakukan sesuai slot waktu yang telah diberikan pada *node client* secara acak. Ada 5 slot waktu yang ditentukan, yang mana masing-masing slot memiliki dua detik waktu pengiriman. Dua detik waktu pengiriman pada tiap slot waktu ditetapkan supaya data yang dikirimkan oleh *node client* tidak terlalu lama.

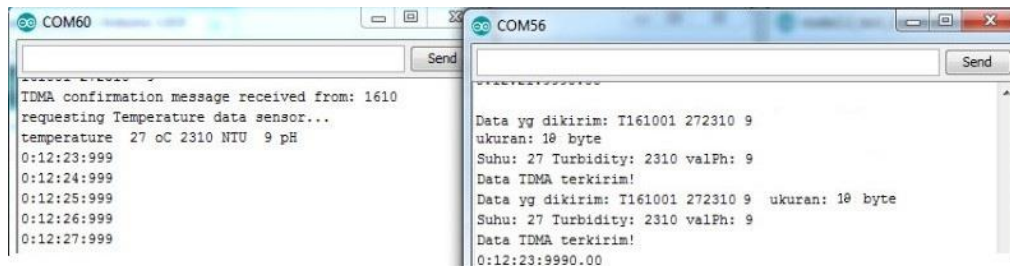
**Tabel 6.3 Slot Waktu Pengiriman**

Pengujian Ke-	Level Node	Waktu Awal (j:m:d:md)	Waktu Setara (j:m:d:md)	Slot Waktu
1	1	0:0:20:13	0:0:21:20	2
	2	0:0:20:18	0:0:44:19	5
2	1	0:0:20:14	0:0:22:13	4
	2	0:0:20:18	0:0:45:93	5
3	1	0:0:40:12	0:0:40:84	2
	2	0:0:20:18	0:1:03:16	5
4	1	0:0:20:13	0:0:21:15	4
	2	0:0:20:19	0:0:43:56	5
5	1	0:0:20:13	0:0:21:26	4
	2	0:0:20:19	0:0:44:29	5
6	1	0:0:20:13	0:0:20:77	4
	2	0:0:20:18	0:0:43:62	5
7	1	0:0:20:14	0:0:20:59	4
	2	0:0:20:28	0:0:43:00	5
8	1	0:0:20:14	0:0:21:35	4
	2	0:0:20:18	0:0:45:10	5
9	1	0:0:20:13	0:0:21:30	2
	2	0:0:20:18	0:0:43:29	5
10	1	0:0:20:12	0:0:20:52	2
	2	0:0:20:18	0:0:43:33	5

**Tabel 6.3** di atas, diketahui bahwa dalam 10 kali pengujian, masing-masing *node client* pada kolam ikan berhasil mendapatkan slot waktu pengiriman. Selanjutnya pengiriman data sensor dikirimkan berdasarkan slot waktu yang telah ditentukan.

### 6.1.3 Pengujian Pengiriman Paket-Paket Data

Pengujian pengiriman paket-paket data oleh node *client* dilakukan berdasarkan batasan penelitian yang telah ditetapkan, yaitu *output* pada *serial monitor* node *base* yang ditampilkan isi paket data yang dikirimkan dan ukuran paket data tiap *node client*. Pengujian ini dilakukan untuk menunjukkan bagaimana keberhasilan pengiriman paket-paket dalam menghindari terjadinya *data collision*.



**Gambar 6.6** Pengujian pengiriman paket-data data

Pada gambar 6.6 paket data *raw* 'T161001 2310 9' dari hasil akuisi data sensor pada air kolam ikan dikirimkan dengan ukuran paket data total sebesar 18 bytes. Tabel 6.4 di bawah ini adalah *sample* ukuran paket-paket data pengiriman hasil pengujian.

**Tabel 6.4** Hasil Pengujian Pengiriman Paket Data

Pengujian ke-	Header			Data Sensor		
	Status (bytes)	Alamat (bytes)	Level (bytes)	pH (bytes)	Suhu (bytes)	Kekeruhan (bytes)
1	1	4	1	4	4	4
2	1	4	1	4	4	4
3	1	4	1	4	4	4
4	1	4	1	4	4	4
5	1	4	1	4	4	4
6	1	4	1	4	4	4
7	1	4	1	4	4	4
8	1	4	1	4	4	4
9	1	4	1	4	4	4
10	1	4	1	4	4	4

Dari **Tabel 6.4** dapat diketahui bahwa dalam 10 kali pengujian dengan dua buah *node client* dari masing-masing kolam ikan. Ukuran paket data header dan data sensor sebesar 18 bytes. Status header pengiriman data sebesar 1 byte

dikarenakan paket data yg dikirimkan hanya berisi panjang 1 karakter berupa nilai 'T'. Alamat header berukuran 4 bytes membutuhkan panjang 4 karakter, level *node* sebesar 1 bytes dikarenakan memiliki panjang data 1 karakter sebesar 1 byte. Nilai pH, suhu dan kekeruhan memiliki ukuran 4 bytes dikarenakan interval data analog yang dikirimkan dari 0 hingga 1024 sebesar 2 karakter.

Total ukuran paket data *raw* akuisisi air kolam ikan adalah 18 bytes atau 144 bit. Dalam implementasi alat ini menggunakan RF *channel* 101. Kecepatan transfer data NRF24L01 sebanding secara linier dengan besar frekuensi channel yang digunakan sebagai contoh 1Mhz memiliki *bandwidth* 1Mbs. Menggunakan perhitungan formula matematis dari datasheet NRF24L01 dapat dihitung waktu pengiriman data antar *node* menggunakan komunikasi NRF24L01 (Nordic, 2008).

$$t = \frac{ukuran\_data\_terkirim[bit]}{RF_{CH}[Mbs]} \quad (6.1)$$

Bila ukuran paket data terkirim sebesar 144 bit dan *channel frekuensi* 101 memiliki *bandwidth* 101Mbs, maka waktu pengiriman data menggunakan komunikasi NRF24L01 sebesar 14  $\mu$ s.