

## FAKTOR YANG MEMPENGARUHI KUALITAS JARINGAN SENSOR NIRKABEL PADA DESAIN MANAJEMEN ENERGI PADA GEDUNG

G. Erwin S.M<sup>1</sup>, Wirawan<sup>2</sup>

<sup>1,2</sup> Lab. Komunikasi Multimedia, T. Elektro, FTI, ITS-Surabaya  
Kampus ITS – Sukolilo – Surabaya - 60111

email : <sup>1</sup>godlieverwin@yahoo.com , <sup>2</sup>wirawan@ee.its.ac.id

### Abstrak

Untuk penghematan energi listrik, salah satu cara yang bisa dilakukan adalah dengan mengatur konsumsi arus untuk peralatan listrik. Jaringan sensor nirkabel (Wireless Sensor Network / WSN) dapat dimanfaatkan untuk keperluan manajemen energi berskala besar maupun kecil secara otomatis. Dengan adanya pengontrolan otomatis ini, manajemen energi menjadi efektif sehingga menekan biaya rekening listrik serta membuat rumah menjadi lebih nyaman dan aman. Dalam penelitian ini didesain suatu sistem jaringan sensor nirkabel yang dapat memantau, mendeteksi dan mengontrol kinerja peralatan listrik yang berkaitan dengan pencahayaan, suhu, kelembaban, pergerakan serta keamanan dalam sebuah bangunan. Jaringan yang dibuat, kemudian diuji kualitas penginderaan, pengiriman data serta konsumsi arus selama masa pemantauan.. Dari hasil pengujian, ternyata posisi sensor dan jumlah hop mempengaruhi kinerja jaringan.

**Kata kunci :** Jaringan, sensor, node, gateway, kontrol.

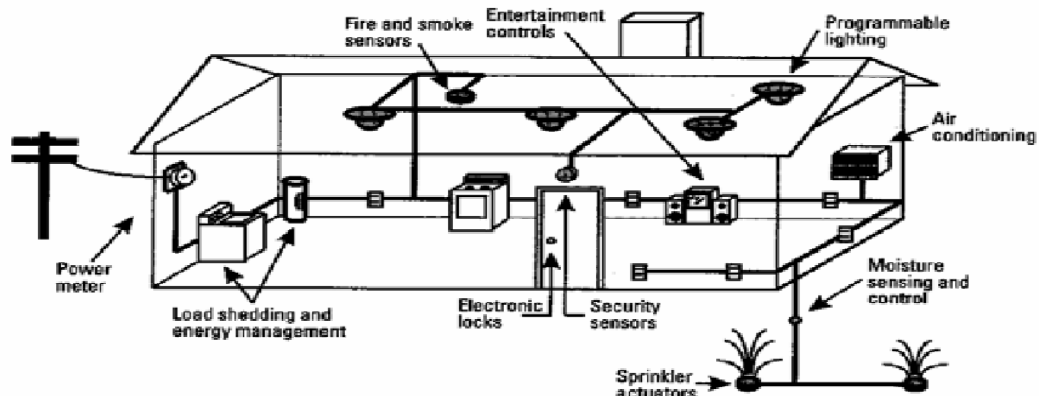
### 1. PENDAHULUAN

Indonesia sedang mengalami krisis energi listrik. Permintaan sambungan listrik yang meningkat akibat perkembangan ekonomi, industry dan populasi penduduk, kadang-kadang tidak dapat langsung dipenuhi oleh instansi yang berwenang. Ditengah keterbatasan suplai energi listrik dan seruan pemerintah untuk menghemat energi listrik, masih sering terjadi peristiwa yang tidak diinginkan seperti kebakaran yang diakibatkan oleh kesalahan atau kelalaian dalam pengoperasian peralatan listrik yang berlebihan serta penggunaan yang tidak terkontrol, pemborosan energi listrik dengan percuma yang menyebabkan tagihan listrik melonjak, kondisi dalam ruangan yang kurang nyaman akibat suhu serta intensitas cahaya yang kurang pas serta pencurian karena kondisi rumah yang sepi pada kompleks perumahan. Hal-hal diatas menimbulkan tantangan yaitu bagaimana membuat suatu gedung atau rumah menjadi lebih "cerdas" dan mampu mengontrol keamanan rumah serta penggunaan peralatan listrik dalam lingkungannya sehingga tempat tersebut menjadi aman dan nyaman dengan mengimplementasikan teknologi jaringan sensor nirkabel. Augusto & Nugent (2006). Dengan menggunakan teknologi sensor yang masih konvensional, masih sering ditemui kesulitan dalam mengumpulkan data. Kelebihan jaringan sensor nirkabel adalah memberikan kemudahan dalam sistem pengontrolan, pemantauan suatu lingkungan secara terus menerus, ada kemudahan dalam proses instalasi, pengukuran data lapangan, penambahan dan penempatan posisi *node* sensor.

### 2. TINJAUAN PUSTAKA

#### 2.1 Konservasi Energi dan Penghematan

Krisis global dibidang energi dan finansial tengah melanda hampir semua negara. Menurut Porter ([www.smarthouse.com.au](http://www.smarthouse.com.au)), hampir 70% dari rata-rata pengeluaran rumah tangga di Amerika adalah untuk pembayaran rekening listrik khususnya untuk pemakaian lampu dan pengaturan suhu. Dengan adanya pengaturan penggunaan energi seperti pada Gambar 1, bukan saja keadaan rumah menjadi nyaman dan aman tetapi sudah memperhatikan isu *global warming*.



Gambar 1: Model rumah dengan pengontrol otomatis ([www.freescale.com/zigbee](http://www.freescale.com/zigbee))

Pemilik rumah dapat mengatur penggunaan energi listrik dengan bantuan jaringan sensor nirkabel. Saat penggunaan energi listrik mencapai titik maksimal dari yang telah ditentukan, secara otomatis jaringan sensor akan mengirim data ke *gateway* untuk diproses dan kontroler akan menurunkan nilai cahaya, temperatur dan kelembaban ruangan. Sensor gerakan yang berada diluar dan didalam ruangan untuk mencegah tamu tak diundang tetap aktif saat peralatan listrik yang lain diturunkan kinerjanya. Lampu adalah salah satu penyerap energi terbesar yang bisa diatur untuk menghemat energi dan biaya. Jaringan sensor nirkabel dapat dirancang untuk mengatur dengan tepat sesuai keinginan secara manual atau otomatis semua peralatan listrik yang akan dikontrol sehingga dapat menekan penggunaan energi dan biaya.

## 2.2 Skenario Manajemen Energi dan Keamanan

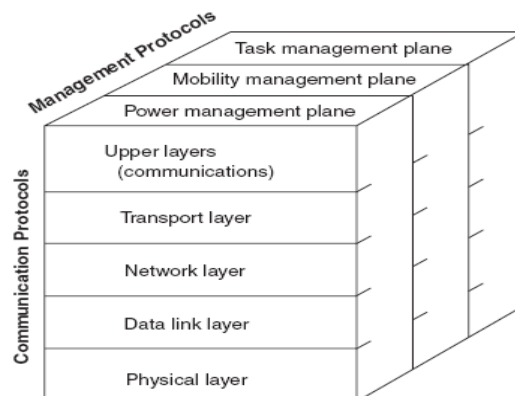
Tujuan desain manajemen energi ini selain untuk penhematan juga untuk digitalisasi semua perangkat listrik dalam rumah serta menghubungkan rumah dengan komputer dan telepon. Sistem ini mengatur :

- **Pencahayaannya** : Lampu dalam desain dibuat supaya hanya akan menyala bila dibutuhkan dan akan mati sendiri bila tidak ada kegiatan dalam ruangan, kecuali diseting untuk keperluan tertentu. Bila ada kegiatan dalam ruangan pada siang hari, maka sistem akan menghidupkan motor untuk menggerakkan gordena membuka, dan apabila dirasa belum cukup terang, maka lampu akan menyala dikontrol *dimmer*.
- **Pemanas dan AC** : Saat tidak ada orang dalam rumah, pemanas ruangan atau AC akan mati dengan sendirinya. AC dan pemanas akan beroperasi untuk memenuhi kondisi yang telah diatur.
- **Monitoring** : penggunaan energi yang berlebihan dari peralatan listrik dalam rumah dapat dideteksi dan peralatan yang beroperasi berlebihan dimatikan bila tidak dibutuhkan lagi.
- **Keselamatan dan keamanan** : Desain ini menawarkan sistem keamanan dan keselamatan yang lebih baik dari sistem alarm tradisional. Dengan mengintegrasikan semua alat listrik dalam satu jaringan pengontrolan, saat terjadi bencana seperti kebakaran, yang bisa dilakukan sistem ini bukan hanya membunyikan alarm tapi juga menyalakan lampu sehingga penghuni dapat menemukan jalan keluar yang paling aman, membuka pintu dan jendela secara otomatis sehingga ventilasi udara menjadi lancar dan gas berbahaya bisa cepat keluar, mematikan semua alat listrik dan panel listrik utama sehingga mencegah hubungan arus pendek serta menghubungi nomor telepon pemadam kebakaran dan tetangga untuk mencari pertolongan secepatnya. semua peralatan.

Pengontrolan sederhana ini membantu penggunaan energi yang tidak perlu serta menghemat pengeluaran untuk tagihan listrik serta membuat rumah menjadi nyaman dan aman.

## 2.3 Jaringan Sensor Nirkabel

Secara umum, jaringan sensor nirkabel (*Wireless Sensor Network*) digambarkan sebagai penyebaran suatu kelompok sensor pintar (*smart sensors*) berukuran kecil secara luas dimana masing-masing titik sensornya memiliki kemampuan untuk merasakan (*sensing*), mengambil data *sensing*, memproses dan berkomunikasi serta melakukan fungsi pengawasan (*monitoring*) terhadap keadaan suatu lingkungan *indoor* maupun *outdoor* secara kolektif. Jaringan sensor nirkabel merupakan generasi baru dari sistem sensor (*sensory system*) yang mempunyai keterbatasan energi, pemrosesan data yang lambat, kapasitas penyimpanan data yang kecil serta dioperasikan dalam suatu lingkungan dalam waktu yang lama. Para peneliti telah mencoba mengembangkan beberapa protokol baru yang khusus didesain untuk jaringan sensor nirkabel, yang bersifat *energy awareness*. Fokus penelitian lebih kepada protokol routing, karena adanya perbedaan yang mendasar antara jaringan biasa dibandingkan dengan jaringan sensor (baik arsitektur jaringan maupun aplikasi). Gambar 2 berikut menunjukkan model protokol generik yang dapat digunakan untuk mendeskripsikan jalur komunikasi didalam Jaringan sensor nirkabel. Willey & Sons (2007).



Gambar 2: Protokol generik dari Jaringan sensor nirkabel

Tabel 1. Komparasi spesifikasi beberapa teknologi *wireless*

Spesifikasi	GPRS/GSM 1xRTT/CDMA	IEEE 802.11b/g	IEEE 802.15.1	IEEE 802.15.4
Nama market standart	2.5G/3G	Wi-Fi	Bluetooth	Zigbee
Jenis jaringan	WAN / MAN	WLAN / Hot spot	PAN	WSN
Aplikasi	Suara dan data pada area luas	Aplikasi Enterprise (VoIP)	Pengganti kabel	Monitoring dan pengaturan
BW (Mbps)	0.064-0.128	11-54	0.7	0.020-0.25
Jarak transmisi	3000 kaki	1-300 kaki	1-30 kaki	1-300kaki
Faktor desain	Kualitas transmisi	Pendukung	Pemakaian sederhana	Reliabilitas, dan power

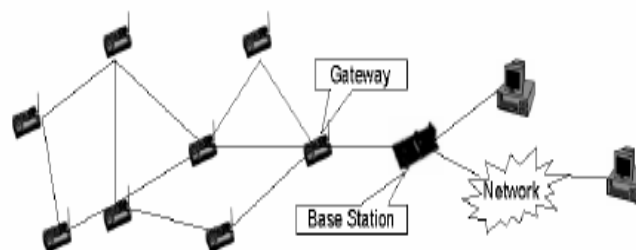
Beberapa versi komersial dari penerapan teknologi transmisi data nirkabel ditujukan untuk aplikasi dari jaringan sensor nirkabel. Kebanyakan dari sistem tersebut mengikuti spesifikasi standart dari IEEE 802.15.4 dan *Zigbee* untuk aplikasi pada lingkup jaringan area personal nirkabel (*WPAN-Wireless Personal Area Network*) IEEE (2003). Untuk komparasi dengan teknologi wireless lainnya dapat dilihat pada Tabel 1.

### 3. METODE PENELITIAN

Metode yang dipakai dalam penelitian ini adalah menentukan skenario manajemen energi, instalasi *hardware* dan *software*, melakukan pengamatan dan pengambilan data serta menganalisa data dengan rincian sebagai berikut :

#### 3.1 Pemilihan Topologi Sistem

Ada 6 topologi dasar jaringan komunikasi yang dikenal adalah koneksi jenis *star*, *ring*, *bus*, *tree*, *fully connected* dan *mesh*. Pada topologi *mesh*, *node* tersebar dan komunikasi data antar *node* dikirim bertahap melalui *node* terdekatnya. Semua *node* pada jaringan ini sudah ada identitas pengenalnya. Karl & Willig (2005). Berdasarkan pertimbangan diatas, maka dipilih topologi *mesh*, sebab cocok untuk jaringan dengan *node* yang banyak atau yang tersebar di lingkungan yang luas. Komunikasi antar *node* melalui banyak jalur (*multihop*). Keuntungan dari jaringan ini, walaupun semua *node* mempunyai identitas masing-masing dan mempunyai kemampuan komputasi yang sama, tapi ada satu *node* yang bisa diset sebagai *node* kepala (*group leader*) yang mempunyai kemampuan tambahan yang tidak dimiliki *node* lain. Apabila *node* kepala mengalami gangguan atau rusak, fungsinya bisa diambil alih oleh *node* yang lain. Secara umum, topologi jaringan sensor nirkabel terdiri dari dua atau tiga jaringan *node* yang terhubung secara *multihop* seperti pada Gambar 3. Data hasil penginderaan mengalami proses awal pada *node*, kadang disimpan sebelum dikirimkan ke *gateway* yang mempunyai fitur lebih lengkap untuk pemrosesan data yang kemudian dikirim ke *base station*.



Gambar 3: Topologi WSN untuk pengumpulan data (Hermann C., 2005)

#### 3.2 Hardware

Dalam rancangan ini, perangkat keras yang digunakan dibagi menjadi 3 bagian yaitu sensor dan *interface*, *server* serta kontroler. Berikut adalah perincian perangkat keras dan spesifikasinya umumnya.

- *Sensorboard* : Tipe MTS420, produksi *Crossbow Technology*. *Sensorboard* ini dapat mengindera adanya cahaya, temperatur, kelembaban, tekanan serta gerakan (akselerometer) sesuai jumlah sensor yang *onboard*.

MTS240 juga mempunyai modul GPS. *Sensorboard* dipasang pada *mote* Micaz MPR2400 yang akan memancarkan data sensing menuju *gateway*.

- *Mote Processor* : Prosesor Atmel ATmega 128L, Program *Flash Memory* 128 Kbyte, *Measurement Flash* 512 Kbyte, komunikasi serial UART, 10 bit ADC (0-3 volt), pemakaian arus 8 mA (mode aktif), < 15  $\mu$ A mode *sleep*. *Transceiver Radio* : frekuensi kerja 2400 MHz – 2483.5 MHz, kecepatan pengiriman data 250 kbps, daya kirim -24 dbm - 0 dbm, sensitivitas penerima -90 dbm (min), -94 dbm (typ), pemakaian arus 19.7ma (mode terima), 11ma (pengiriman -10dbm), 14ma (pengiriman -5dbm), 17.4dbm (pengiriman 0dbm), 20  $\mu$ A (mode *idle*), 1  $\mu$ A (mode *sleep*). Elektromekanik : Baterai 2xAA, sumber daya lain 2.7 V – 3.3 V, *User Interface* LED (merah, kuning, hijau), ukuran 58x32x7 (tidak termasuk baterai), berat 18 gram (tidak termasuk baterai), 51-pin konektor ekspansi.
- *Gateway* : *Interface Node* ; konektor ekspansi 51-pin. *interface* UART, *user interface* LED (merah, kuning, hijau). *Interface LAN* ; *internal web server*, *serial login*, *telnet login*, *password* keamanan, layanan pengunci. *Interface jaringan ethernet* : konektor RJ45, berbasis protokol IEEE 802.3, kecepatan data 10 MBps (10 Base-T). Pemrograman dalam sistem (*In-System Programming*) ; Protokol UISP (*UART in-system programming*), LED indikator ISP aktif, *RESET node*. Fisik : Dimensi 11.76cm x 5.82cm x 2.54cm, sumber daya PoE 5V. *Gateway* digunakan untuk melewatkan paket dari jaringan sensor kepada *server*. Tipe MIB 600 untuk komunikasi jaringan sensor dan *interface Ethernet Programming Board* (EPRB) untuk komunikasi *server*. Produksi *Crossbow Technology*. Bentuk fisik dari *sensorboard*, *mote* dan *gateway* dapat dilihat pada Gambar 4. *Sensorboard* yang ditempelkan diatas *mote* nantinya disebut *node*.

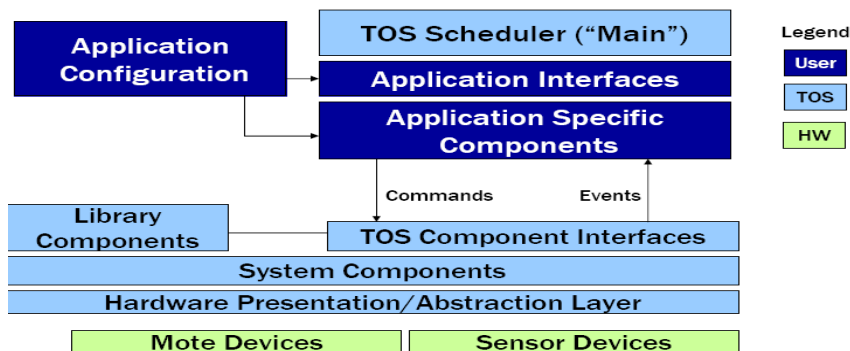


Gambar 4: *Gateway* (MIB 600), *Mote* (MPR 2400) dan *sensorboard* MTS240

- *Server* : Prosesor Intel Pentium Dual Core CPU T2390 @ 1.86 GHz, *Memory* DDR2 1 GB, *Network Interface Card* (NIC) Realtek RTL8186/8111PCI-E Gigabit Ethernet.
- *Kontroler* : *Minimum system microcontroller* DT-51, *switch panel*, *dimmer*, *motor stepper* DC 12V.

### 3.3 Software

Setiap sensor *node* membutuhkan sistem operasi untuk mengatur *hardware* dari sensor agar dapat berinteraksi dengan *software* aplikasinya. Dan di setiap sistem operasi menggunakan bahasa pemrograman yang beragam. Pada desain ini, digunakan sistem operasi *TinyOS* dan bahasa pemrograman *NesC*. *TinyOS* merupakan sistem operasi *open-source* yang didesain khusus untuk jaringan sensor nirkabel. *TinyOS* memiliki arsitektur berbasis komponen yang mendukung adanya inovasi dan implementasi jaringan sensor nirkabel, dengan cara meminimalisir ukuran kode yang dibutuhkan, karena komponen sensor memiliki memori yang sangat terbatas. *TinyOS* memiliki model pemrograman berbasis komponen yaitu *NesC*. Layaknya sistem operasi lainnya, *TinyOS* mengorganisir komponen perangkat lunaknya dalam beberapa lapisan. Lapisan paling bawah berkaitan dengan perangkat keras, dan lapisan paling tinggi adalah aplikasi yang digunakan. Ilustrasi lapisan *TinyOS* dapat dilihat pada Gambar 5. *Crossbow Technology* (2005).

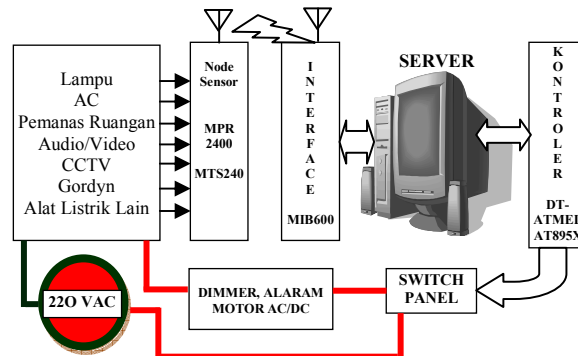


Gambar 5: Lapisan TinyOS

Selain *TinyOS*, ada beberapa software pendukung yang dipakai yaitu *Lantronix Device Installer*, *Cygwin*, *Serial Forwarder* (*Message Center* dan *Listen*) dan *SensorViz* yang akan menjadi GUI.

### 3.4 Desain Sistem Manajemen Energi

Pada penelitian ini, kinerja dari sistem jaringan sensor nirkabel yang dibuat untuk manajemen energi listrik diujicobakan dalam suatu ruangan untuk pemantauan dan pengontrolan cahaya, temperatur, kelembaban serta pergerakan dalam ruangan. Gambar 6 menunjukkan blok sistem manajemen energi.



Gambar 6. Blok sistem manajemen energi

Cara kerja dari desain dan jaringan yang dibangun adalah sebagai berikut :

- *Sensorboard* mengumpulkan data berupa intensitas cahaya, temperatur, kelembaban, dan pergerakan objek dalam ruangan.
- *Mote* kemudian mengirimkan data *sensing* ke *gateway*.
- *Gateway* mengolah data *sensing* dan mengirimkan ke *server*.
- *Server* memproses data dari *gateway* untuk ditampilkan. Bila sensor melaporkan parameter yang melewati batasan yang ditentukan, server memberi perintah kepada kontroler.
- Kontroler mengendalikan *switch* untuk menaikkan atau menurunkan kinerja peralatan listrik.

### 3.5 Pengujian Awal Jaringan Sensor

Sebelum dioperasikan, *gateway* diberi IP menggunakan *Lantronix Device Installer*, software yang digunakan untuk manajemen perangkat *gateway* (MIB600). Software ini dapat digunakan untuk memberikan alamat IP pada *gateway*, mengatur *baud rate gateway*, bahkan mengatur *port* yang ada pada *gateway* serta mampu melakukan pengaturan *hardware* baik secara grafis ataupun melalui *mode text*. Pada desain ini diberikan alamat IP 192.168.1.5/24 pada *gateway*. Setelah itu *mote* diinjekt dengan software untuk menentukan fungsinya sebagai *node* atau *base station* dan diberi IP address untuk bisa berhubungan dengan *gateway* dan *server*. Dalam pengujian ini, 4 buah *mote* difungsikan sebagai *node* dan sebuah *mote* difungsikan sebagai *base station*. Caranya yaitu dengan menempelkan *mote* ke soket yang ada pada *gateway* yang dihubungkan dengan *server* lewat konektor *ethernet* untuk diinjekt. Injeksi dilakukan dengan bantuan software *Cygwin* (*Cygnus Linux for Windows*), yaitu program aplikasi berbasis *Linux* yang bekerja dalam *Windows*. *Mote base station* ditempelkan pada *gateway* sedangkan *sensor board* ditempelkan pada *mote* untuk menjadi *node*. *Node* kemudian disebarkan di beberapa titik dalam ruangan dan diaktifkan untuk mulai mengumpulkan data *sensing*. Data *sensing* kemudian dipancarkan ke *mote base station* yang diintegrasikan dengan *gateway* untuk diolah dan dikirim ke *server* untuk ditampilkan. Pembacaan dan penampilan paket data di *server* menggunakan program *Serial Forwarder*. Program ini bertugas meneruskan data *sensing* ke *server* melalui port *ethernet* dan harus dijalankan sebelum *tools MessageCenter* diaktifkan. Untuk pengamatan dan penyimpanan paket data, digunakan dua *tools* yaitu *Listen* yang bekerja pada jendela *Cygwin* dan *MessageCenter* yang bekerja pada jendela *Java*. Data yang dihasilkan kedua *tools* tadi masih berupa deretan bilangan *Hexadecimal*. *Tools* lain yang bekerja dalam jendela *Cygwin* dan mirip *Listen* adalah *Xlisten*. *Tools* ini mampu mengkonversi data *hexadecimal* menjadi *decimal* dan data *engineer unit* yang lebih mudah dimengerti. Manajemen tugas sensor, teknik komunikasi data antar sensor dan *gateway* serta *monitoring* jaringan dan pengamatan data diatur dengan *TinyOS*.

Pada pengetesan koneksi jaringan, data *sensing* dimunculkan lewat *tools Serial Forwarder* dengan mengetikkan perintah pada jendela *cygwin*

```
$ java net.tinyos.sf.SerialForwarder -comm  
network@192.168.1.5:10002
```

artinya *Serial Forwarder* mengambil paket *sensing* dari *gateway* pada alamat 192.168.1.5 port 10002 dan diteruskan ke *server* port 9001.

Kemudian untuk memunculkan jendela *Listen*, ketikkan perintah

```
$ java net.tinyos.tools.Listen
```

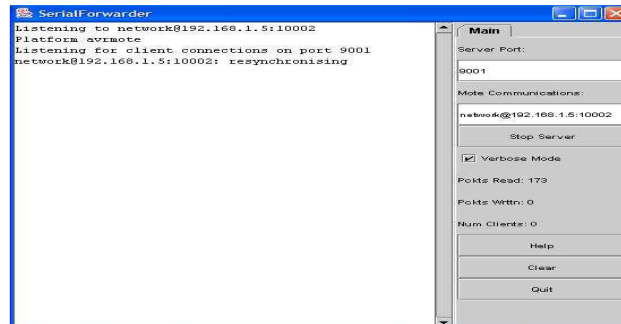
Untuk memanggil *Message Center*, ketikkan perintah

**\$ java net.tinyos.mcenter.MessageCenter**

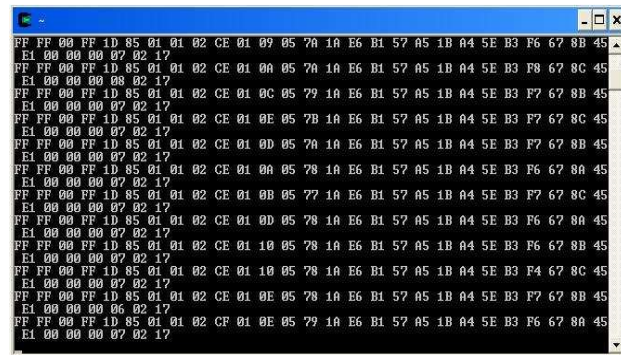
Jendela *Xlisten* yang menghasilkan data yang lebih mudah dibaca dipanggil dengan perintah

**\$ xlisten -p -c -i=192.168.1.5:10002**

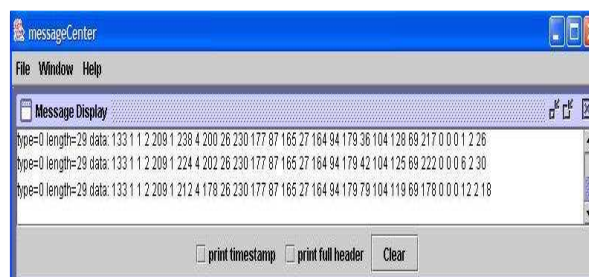
Gambar 7 dan 8 menampilkan jendela *Xlisten* serta foto pengujian jaringan sensor.



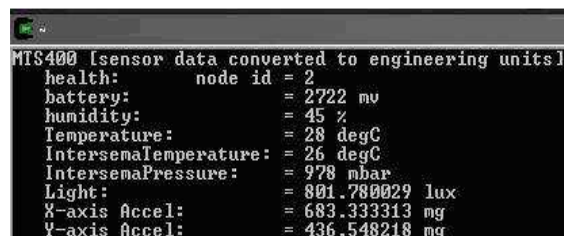
Gambar 7. Tampilan *Serial Forwarder*



Gambar 8. Tampilan *Listen*



Gambar 9. Tampilan *Message Center*



Gambar 10. Tampilan *Xlisten*



Gambar 11. Pengujian Sensor

### 3.5.1 Spesifikasi Kondisi Lingkungan Bangunan Tempat Pengujian

Pada percobaan minimum sistem, dengan menggunakan *TinyOS* dilakukan pemrograman kondisi ideal dalam sebuah ruangan keluarga dengan skenario sebagai berikut :

- Suhu dalam ruangan misalkan diset berkisar antara 25-30°C. Apabila data sensing menunjukkan suhu ruangan diatas 30°C karena ada peningkatan aktifitas dalam ruangan, semakin banyak orang didalam ruangan atau semakin banyak alat listrik dalam ruangan itu yang dioperasikan, maka pengontrol *menswitch* AC agar mengeluarkan suhu yang lebih dingin lagi. Bila data sensing menunjukkan suhu ruangan dibawah 25°C karena sedikit aktifitas dalam ruangan atau sedang musim dingin, maka pengontrol *menswitch* pemanas ruangan untuk mengeluarkan suhu yang lebih hangat lagi.
- Luminan cahaya dalam sebuah ruang keluarga pada jam 06.00-09.00 misalkan sebesar 3.000-4.000 cd/m<sup>2</sup>, jam 09.01-15.00 sebesar 5.000-20.000 cd/m<sup>2</sup>, jam 15.01-17.00 sebesar 5.000-6.000 cd/m<sup>2</sup>, jam 17.01-21.30 sebesar 5.000-20.000 cd/m<sup>2</sup>, jam 21.31-05.59 sebesar 4.000-15.000 cd/m<sup>2</sup>. Pada jam 18.00-21.20, apabila sensor gerak mendeteksi tidak ada orang dalam ruangan, maka *dimmer* menurunkan luminan cahaya lampu menjadi 1000 cd/m<sup>2</sup>. Bila intensitas cahaya berada dibawah batasan pada jam 06.00-16.30, maka kontroler *menswitch* motor agar memutar naik *gordyn*. Jika luminasi belum mencapai batasan minimal, lampu akan menyala dikontrol *dimmer*. Jika luminasi melebihi batas maksimal, *gordyn* akan diputar turun.
- Kelembaban udara misalkan diatur 18-25% RH /1M<sup>3</sup>. Jika kelembaban kurang dari nilai yang ditentukan, kontroler *menswitch* motor agar membuka jendela, jika diatas nilai yang ditentukan, pemanas ruangan akan dihidupkan.

Selanjutnya dibuat program monitoring data *sensing* untuk kontroler. Program dirancang untuk mengenal batasan-batasan nilai yang bisa diterima yang diinputkan ke kontroler dan akan dipakai untuk mengatur switch bila data *sensing* keluar dari batasan yang sudah ditentukan. Setelah mendapat *trigger* dari pengontrol, *switch* akan mematikan, menghidupkan, menaikkan atau menurunkan kinerja semua alat yang terhubung dengannya. Alat-alat listrik yang bisa dihubungkan dengan *switch* contohnya adalah, motor penggerak untuk *gordyn* dan pintu, AC, fan, *dimmer*, oven, jendela dan lain-lain. Dalam pengembangannya, pemantauan dan pengontrolan juga bisa dilakukan dari jarak jauh melalui telepon seluler atau jaringan internet.

## 4. HASIL DAN PEMBAHASAN

Pengolahan salah satu paket data pada *tools Listen* yang diterima dari *node* ID 1 dalam bentuk *hexadecimal* adalah sebagai berikut :

```
7E 00 11 7D 32 00 00 01 00 8B 00 00 00 00 00 00 00 00 CE 04 00 00 00 00 11 00 00 00 12 00 00 00 13 00 00 00 14 00 00 00 15 00 00 00 16 00 00 00 17 00 00 00 18 00 00 00
```

Struktur data diatas dibagi dalam tiga bagian yaitu :

1. **TinyOS Message** yang terdiri dari *header* pembuka paket yang dikirim seperti alamat tujuan pengiriman pesan, terdiri dari 2 *byte* dengan simbol **7E 00** yang artinya alamat *server*. **Message Type** adalah daftar jenis pesan yang dapat diterima oleh *gateway* dari *node*, dalam desain ini dipakai *multihop* dengan simbol **0x11** atau 17 desimal. Bila jenis data yang dikirim ke *gateway* tidak ada dalam daftar, maka tidak akan diterima dan dihapus. **Group ID** adalah identitas semua *node* yang berada dalam Jaringan sensor nirkabel tersebut dengan simbol **0x7D** atau 125 desimal. Jika ada *intruder* diluar jaringan yang mencoba mengakses *gateway*, tidak akan dilayani. **Message Length** adalah banyaknya *byte* dalam pesan dengan simbol **0x32** atau 50 desimal.
2. **Multihop Message** adalah format pesan yang dikirim memakai protokol *multihop* dengan *header* sebanyak 7 *byte* untuk : **source\_addr** yaitu alamat *gateway* dengan ID 0 yang dituju oleh *node*. Alamat tersebut disimbolkan dengan 2 *byte* yaitu **00 00**. **origin\_addr** adalah *header* yang mempresentasikan alamat *node* pengirim paket. Alamat tersebut disimbolkan dengan 2 *byte* yaitu **10 00** dalam hal ini *node* ID. **seq\_no** adalah *header* yang menunjukkan urutan penerusan pengiriman paket yang disimbolkan dengan **0x00 8B** sepanjang 2 *byte*. **hop\_count** menunjukkan berapa hop yang dilakukan untuk sampai di tujuan, dalam contoh ini disimbolkan dengan **00**, artinya *node* ID 1 berhubungan langsung dengan *gateway*.
3. **Data Message (Surge)** yang berisi paket data hasil *sensing* dan terbagi menjadi **Message Type** yang dipakai untuk sinkronisasi antara *node* dan *gateway* untuk identifikasi tipe pesan pada paket . Disimbolkan dengan **0x00** untuk aplikasi *Surge*. **Sensor Reading** adalah data *sensing* yang dalam contoh paket ini disimbolkan dengan **0x0000**. *Header* yang berisi alamat *parent* tujuan pengiriman data adalah **parent\_addr** sepanjang 2 *byte* dengan simbol **0x0000** ,dalam hal ini *node* ID 1 mengirim data ke *node* ID 0 yaitu *gateway*. **seq\_no** adalah *header* yang menunjukkan urutan penerusan pengiriman paket yang disimbolkan dengan **0x00 00 CE 04** sepanjang 4 *byte* dan bisa bertambah atau berkurang sesuai

dengan kondisi kepadatan *traffick*. Selanjutnya adalah **Data Sensing** dengan maksimum data adalah 80 *byte* karena keterbatasan memori.

**4.1 Analisa Kualitas Jaringan**

Dalam proses pengetesan dilakukan pengamatan kualitas jaringan dengan cara mengatur posisi sensor pada daerah LOS dengan data seperti pada Tabel 2.

Tabel 2. Hubungan posisi sensor dan *Packet Loss*

Tinggi sensor (m)	Jarak antar sensor (m)	Packet Loss %
0	1-3	0.
	4-7	4 / 24. 5-6 / 0. 7 / 35.
	8-10	100.
0,5	1-13	0.
	14-18	14 / 12. 15-16 / 0. 17 / 20. 18 / 13.
	20-25	100.
1,5	1-20	0.
	21-25	21 / 12. 22-25 / 0.
	26-30	100.

Dari data diatas, terlihat jelas bahwa posisi penempatan sensor dari permukaan lantai serta jarak antar sensor mempengaruhi kualitas jaringan dalam hal penerimaan paket data.

**4.2 Analisa Komsumsi Arus**

Pengamatan terhadap komsumsi arus pada setiap *node* dimaksudkan untuk mengetahui karakteristik komsumsi arus pada komunikasi *multihop*. Setiap *node* dipasang 2 buah *battery* AA 1,5 *volt* dengan arus 4,6 *Ampere*. Semua *Node* mengirimkan paket dengan panjang 80 *byte*. Hasil pengamatan adalah seperti pada Tabel 3. Dari data, terlihat bahwa komsumsi arus pada tiap *node* berbeda. Pada kondisi awal, rata-rata tiap *node* membutuhkan arus sebesar 24,25 mA untuk transmisi data. Setelah beberapa jam beroperasi, *battery* pada *node* 1 mulai kehilangan arus secara signifikan dibanding *battery* pada *node* lain. Kemudian terlihat *drop* dengan cepat, sedangkan pada *node* lain belum mengalami hal serupa. Selang beberapa jam, keadaan ini diikuti oleh *node* 2. *Battery node* 3 mulai *drop* selang beberapa jam dari waktu *drop node* 2. Yang paling irit adalah *battery node* 4. Kondisi ini terjadi karena *node* 1 menerima beban kerja lebih berat dari *node* yang lain yaitu mengambil data *sensing*, menerima data dari *node* 2 berupa data *sensing node* 2,3,dan 4 serta mengirimkan ke *gateway*. Karena data dikirim secara *multihop* maka *node* 4 hanya mengambil data *sensing* dan mengirimkan ke *node* 3 dan seterusnya secara berurutan.

Tabel 3. Komsumsi arus pada *node*

Jam pengukuran	<i>Node</i>			
	1	2	3	4
	Komsumsi arus (mA)			
21.00	24.4	24.35	24.4	25.1
06.00	18.7	23.1	24.1	24.6
07.00	0.5	17.4	23.5	24.1
10.00	0.5	0.6	16.6	23.3
12.00	0.5	0.5	0.6	18.5

**5. KESIMPULAN**

Dalam desain ini, komponen utama yang menjadi ujung tombak adalah sensor *node*, sehingga dalam perancangan harus diperhatikan :

- 1.Posisi dan jarak sensor, karena sangat berpengaruh pada penerimaan data *sensing* yang berkaitan dengan kualitas jaringan dan untuk mencegah *packet loss*.
- 2.Jumlah hop juga dibatasi untuk menghemat penggunaan energi *battery* sehingga umur pemantauan bisa lebih lama serta memperpendek *delay time*.
- 3.Manajemen dan penentuan batasan nilai sensor ideal serta *trigger* pengontrolan diatur lewat *TinyOS*. Kedepannya desain ini perlu dilanjutkan menggunakan komunikasi single hop dengan topologi jaringan selain *mesh* dan menghubungkannya dengan internet serta layanan SMS agar sistem bisa dikontrol dari jauh.

**6. DAFTAR PUSTAKA**

Augusto J.C & Nugent C.D. (2006) : *Designing Smart Homes* , NewYork, Springer.  
 Crossbow Technology, "Tiny OS Overview", www.ce.rit.edu/~fxheec/cisco\_urp/cd\_seminar/Presentations/Day1-All/03\_TinyOS\_Overview.pdf  
 Hermann C. (2005), *Middleware for Supporting Multiple Application In Wireless Sensor Network*, Assignment Thesis, Institute for System Architecture, Technische Universität Dresden  
 IEEE 802.15.4. : *Spesification*, 2003.  
 Karl H. dan Willig A. (2005), *Protocols and architectures for wireless sensor networks*, New York, John Wiley & Sons Inc.  
[www.freescale.com/zigbee](http://www.freescale.com/zigbee)