

**IMPLEMENTASI PROTOKOL ROUTING FLOODING PADA
WIRELESS SENSOR NETWORK DENGAN MENGGUNAKAN
MEDIA KOMUNIKASI RF**

SKRIPSI

KEMINATAN TEKNIK KOMPUTER

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:

Panji Putera Dwi Saadilah Harun

NIM: 125150300111011



**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2016**

PENGESAHAN

IMPLEMENTASI PROTOKOL *ROUTING FLOODING* PADA *WIRELES SENSOR NETWORK* MENGGUNAKAN MEDIA KOMUNIKASI RF

SKRIPSI

KEMINATAN TEKNIK KOMPUTER

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:

Panji Putera Dwi Saadilah Harun

NIM: 125150300111011

Skripsi ini telah diuji dan dinyatakan lulus pada

15 Agustus 2016

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

Sabriansyah Rizqika Akbar, S.T, M.Eng.

NIK: 19820809 201212 1 004

Aswin Suharsono, S.T, M.T

NIK: 201106 840919 1 001

Mengetahui

Ketua Jurusan Teknik Informatika

Tri Astoto Kurniawan, S. T, M. T, Ph.D

NIP: 19710518 200312 1 001

PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 15 Agustus 2016

Panji Putera Dwi Saadilah Harun

NIM: 125150300111011

KATA PENGANTAR

Puji syukur kami panjatkan kehadirat Allah SWT yang mana atas limpahan rahmat, taufik serta hidayah-Nya, skripsi ini dapat terselesaikan. Skripsi yang berjudul **“Implementasi Protokol Routing Flooding Menggunakan Media Komunikasi RF pada Wireless Sensor Network”** ini disusun untuk memenuhi sebagian persyaratan memperoleh gelar Sarjana Komputer.

Penulis menyadari bahwa penyusunan skripsi ini terlepas dari bantuan berbagai pihak. Oleh karena itu penulis mengucapkan terima kasih kepada pihak-pihak yang telah bersedia untuk memberikan bantuan demi kelancaran penyusunan skripsi ini diantaranya:

1. Bapak Sabriansyah Rizqika Akbar, S.T, M.Eng. dan Aswin Suharsono, S.T, M.T. selaku dosen pembimbing skripsi yang telah dengan sabar membimbing dan mengarahkan penulis sehingga dapat menyelesaikan skripsi ini.
2. Bapak Tri Astoto Kurniawan, S. T, M. T, Ph.D selaku Ketua Program Studi Teknik Informatika Program Teknologi Informasi dan Ilmu Komputer.
3. Bapak dan Ibu serta seluruh keluarga besar yang saya cintai atas segala dukungan dan semangat serta tiada henti-hentinya memberikan do'a demi terselesaikannya skripsi ini.
4. Seluruh civitas akademik Informatika Universitas Brawijaya yang telah banyak memberi bantuan dan dukungan selama penulis menempuh studi di Informatika Universitas Brawijaya dan selama penyelesaian skripsi ini.
5. Rekan-rekan Sistem Komputer 2012 yang selalu memberikan motivasi dalam menyelesaikan skripsi ini.

Pada akhirnya, penulis menyadari bahwa skripsi ini masih belum sempurna. Oleh karena itu, penulis sangat mengharapkan kritik dan saran yang membangun. Penulis berharap semoga skripsi ini dapat bermanfaat bagi pengembangan ilmu pengetahuan dan teknologi.

Malang, 15 Agustus 2016

Panji Putera Dwi Saadilah Harun

Kaaa.conk@gmail.com

ABSTRAK

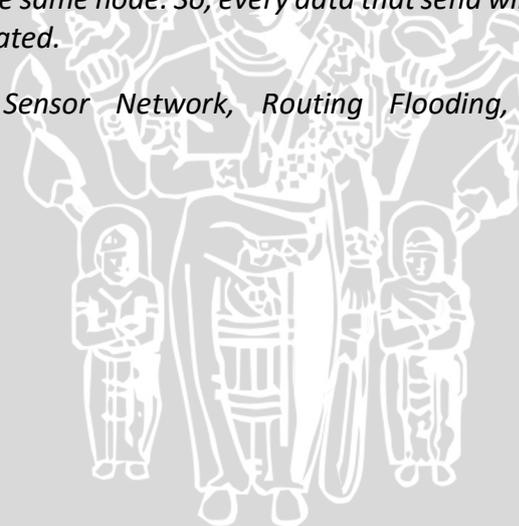
Teknologi dari tahun ke tahun mengalami perkembangan. Dari berbagai sektor, teknologi sudah mulai diterapkan. Teknologi dapat membantu proses pengerjaan di bidang masing-masing sesuai kebutuhan. Salah satu teknologi yang berkembang adalah teknologi berbasis *wireless sensor network* (WSN). Teknologi WSN memiliki banyak keuntungan, baik dari segi harga, aplikasi, serta keberadaan barang di pasaran. Untuk lebih memaksimalkan kinerja WSN, digunakan beberapa protokol routing salah satu contohnya adalah routing *flooding*. Routing *flooding* merupakan algoritma routing dimana data akan disebar ke seluruh alamat tanpa terkecuali, hal ini diharapkan agar data tetap sampai pada tujuan walaupun harus membanjiri jaringan dengan data yang sama. Jika salah satu node mengalami masalah, maka ada node lain yang memungkinkan untuk bisa mengirimkan data hingga sampai ke tujuan. Pengimplementasian routing *flooding* pada penelitian ini menggunakan mikrokontroler berbasis Arduino dan media komunikasi yang digunakan adalah nRF24L01 yang sudah terintegrasi dengan Arduino. Setiap node terdiri atas satu buah Arduino Nano dan satu buah nRF24L01. Setiap node akan menyimpan alamat semua node yang berada pada satu frekuensi. Hasil penelitian yang didapatkan yaitu data akan melewati seluruh node yaitu dengan cara memberikan nilai berupa alamat yang telah dilewati oleh data pada masing-masing node. Setiap pengiriman akan diberikan dihitung dan setiap data yang telah melewati node tidak akan dikirimkan kembali ke node sebelumnya agar tidak terjadi tabrakan data. Sehingga pengiriman secara *flooding* akan mendapatkan hasil yang maksimal tanpa harus terjadi duplikasi pengiriman.

Kata Kunci: Wireless Sensor Network, Routing Flooding, Komunikasi Radio Frekuensi

ABSTRACT

Technology is improved every year. Technology is applied in a lot of fields. Technology can help in the process in every sector according to in every case. One of the technology is Wireless Sensors Networks (WSN). WSN Technology has many advantages, both in price, application, and goods in the market. To maximize WSN performance, we used some routing protocol for example flooding algorithm. Algorithm flooding is a routing algorithm which data will be sent to all address, it's expected that data arrive in destination even though the data will flood the network with the same data. If one of a node has a problem, the others node will send the data until the data arrive at the destination. Implementation routing flooding in this research using microcontroller based on Arduino and communication that used is nRF24L01 which integrated with Arduino. Each node using one Arduino and one nRF24L01. Each node will save all of the address in one frequency. The result in this research every data can through all the nodes which every node will give it addresses after data arrive in every node. Every data that send to all node will not resend to the address that has marked in data, for not making a collision in the same node. So, every data that send will get the best result without getting duplicated.

Keywords: *Wireless Sensor Network, Routing Flooding, Radio Frequency Communication*



DAFTAR ISI

| | |
|---|-----------|
| PENGESAHAN..... | ii |
| PERNYATAAN ORISINALITAS | iii |
| KATA PENGANTAR..... | iv |
| ABSTRAK | v |
| ABSTRACT | vi |
| DAFTAR ISI..... | vii |
| DAFTAR TABEL | xi |
| DAFTAR GAMBAR..... | xii |
| BAB 1 PENDAHULUAN..... | 1 |
| 1.1 Latar belakang..... | 1 |
| 1.2 Rumusan masalah..... | 2 |
| 1.3 Tujuan | 2 |
| 1.4 Manfaat | 2 |
| 1.5 Batasan masalah | 2 |
| 1.6 Sistematika pembahasan | 3 |
| BAB 2 LANDASAN KEPUSTAKAAN | 5 |
| 2.1 Wireless Sensor Network | 5 |
| 2.1.2 Hierarchical-based..... | 7 |
| 2.2 Arduino | 9 |
| 2.3 NRF24L01..... | 10 |
| 2.4 Protokol Flooding..... | 11 |
| 2.4.1 Implosion | 11 |
| 2.4.2 Overlap | 12 |
| 2.4.3 Resource Blindness | 13 |
| BAB 3 METODOLOGI | 13 |
| 3.1 Studi Literatur | 14 |

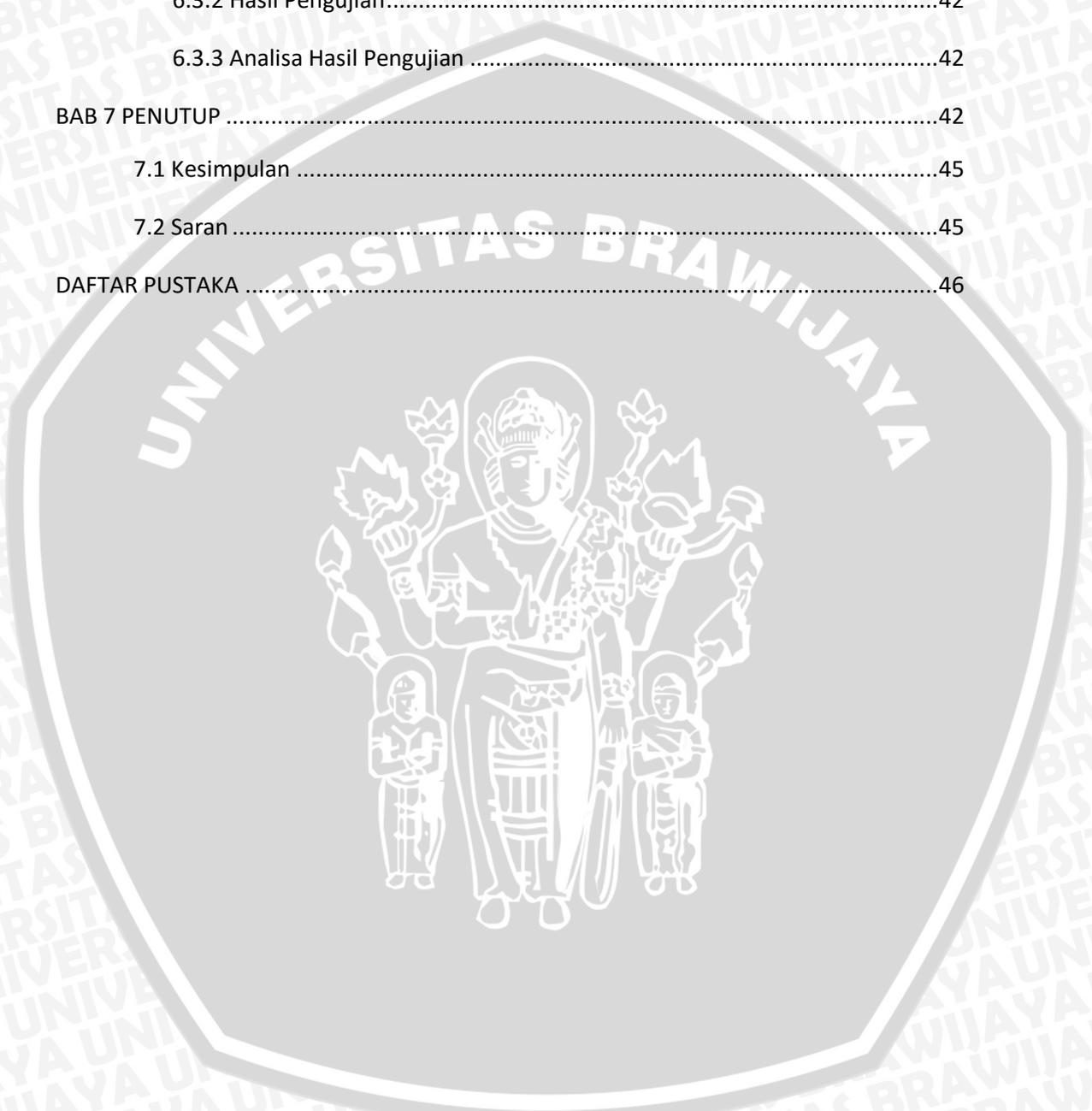
| | |
|---|-----------|
| 3.2 Analisis Kebutuhan..... | 15 |
| 3.2.1 Kebutuhan Pengguna..... | 15 |
| 3.2.2 Kebutuhan Sistem..... | 15 |
| 3.2.3 Kebutuhan Perangkat Keras..... | 15 |
| 3.2.4 Kebutuhan Perangkat Lunak..... | 16 |
| 3.3 Perancangan Sistem..... | 16 |
| 3.3.1 Perancangan Rangkaian Node..... | 16 |
| 3.3.2 Perancangan Topologi Node..... | 16 |
| 3.3.3 Perancangan skema komunikasi untuk protokol routing <i>flooding</i> | 18 |
| 3.4 Implementasi..... | 19 |
| 3.5 Pengujian dan Analisis..... | 19 |
| 3.6 Kesimpulan..... | 20 |
| BAB 4 Rekayasa Kebutuhan..... | 21 |
| 4.1 Gambaran Umum..... | 21 |
| 4.1.1 Tujuan..... | 21 |
| 4.1.2 Ruang Lingkup..... | 21 |
| 4.1.3 Referensi..... | 21 |
| 4.1.4 Sistematika..... | 22 |
| 4.2 Deskriptif umum..... | 22 |
| 4.2.1 Perspektif Produk/Sistem..... | 22 |
| 4.2.2 Kegunaan..... | 22 |
| 4.2.3 Karakteristik Pengguna..... | 23 |
| 4.2.4 Lingkungan Operasi..... | 23 |
| 4.2.5 Batasan Perancangan dan Implementasi..... | 23 |
| 4.2.6 Asumsi dan Ketergantungan..... | 23 |
| 4.3 Kebutuhan Antarmuka Eksternal..... | 23 |



| | | |
|---|---|-----------|
| 4.3.1 | Kebutuhan Antarmuka Pengguna | 23 |
| 4.3.2 | Kebutuhan Perangkat Keras | 24 |
| 4.3.3 | Kebutuhan Perangkat Lunak | 24 |
| 4.4 | Kebutuhan Fungsional | 24 |
| 4.4.1 | Sistem dapat melakukan <i>routing</i> sesuai dengan <i>routing flooding</i> | 24 |
| 4.4.2 | Sistem melakukan kegiatan pengiriman dan penerimaan paket data .. | 24 |
| 4.4.3 | Sistem menampilkan hasil pengiriman dan penerimaan paket data | 25 |
| 4.5 | Kebutuhan Non-Fungsional | 25 |
| 4.5.1 | Kebutuhan Performansi | 25 |
| BAB 5 PERANCANGAN DAN IMPLEMENTASI | | 26 |
| 5.1 | Perancangan Sistem | 26 |
| 5.1.1 | Perancangan Perangkat Lunak | 26 |
| 5.1.2 | Perancangan Perangkat Keras | 26 |
| 5.1.3 | Perancangan Skema Protokol Routing Flooding | 27 |
| 5.2 | Implementasi Sistem | 30 |
| 5.2.1 | Spesifikasi Perangkat Keras | 30 |
| 5.2.2 | Spesifikasi Perangkat Lunak | 32 |
| 5.2.3 | Batasan Implementasi | 32 |
| 5.2.4 | Impelementasi Protokol <i>Routing Flooding</i> | 33 |
| BAB 6 PENGUJIAN DAN ANALISA | | 38 |
| 6.1 | Pengujian Pengiriman dan Penerimaan Data | 38 |
| 6.1.1 | Pelaksanaan Pengujian | 38 |
| 6.1.2 | Hasil Pengujian | 38 |
| 6.1.3 | Analisa Hasil Pengujian | 39 |
| 6.2 | Pengujian Paket Data | 40 |
| 6.2.1 | Pelaksanaan Pengujian | 40 |
| 6.2.2 | Hasil Pengujian | 40 |

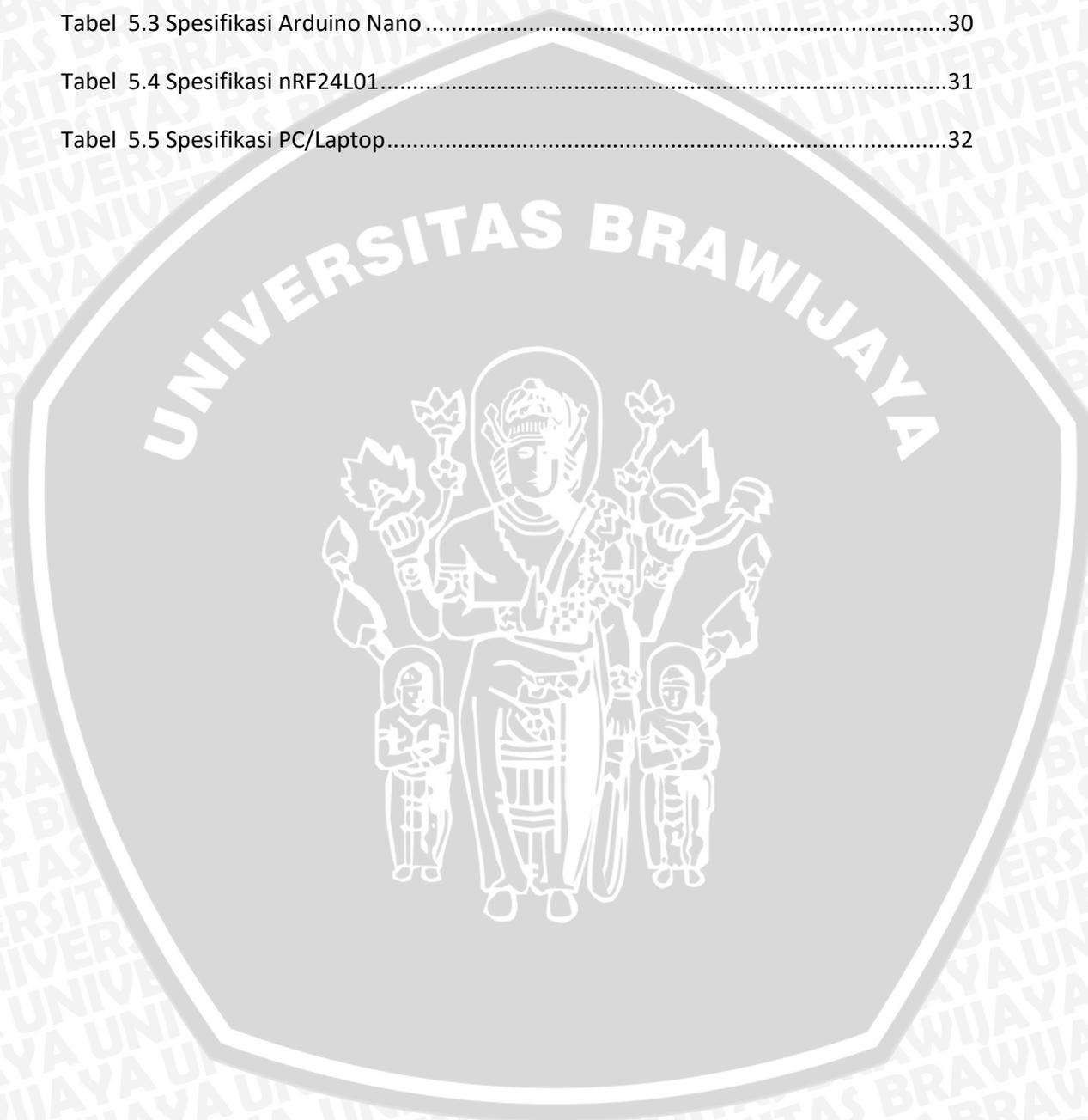


| | |
|--------------------------------------|-----------|
| 6.2.3 Analisa Hasil Pengujian | 41 |
| 6.3 Pengujian Routing Flooding | 41 |
| 6.3.1 Pelaksanaan Pengujian | 41 |
| 6.3.2 Hasil Pengujian..... | 42 |
| 6.3.3 Analisa Hasil Pengujian | 42 |
| BAB 7 PENUTUP | 42 |
| 7.1 Kesimpulan | 45 |
| 7.2 Saran | 45 |
| DAFTAR PUSTAKA | 46 |



DAFTAR TABEL

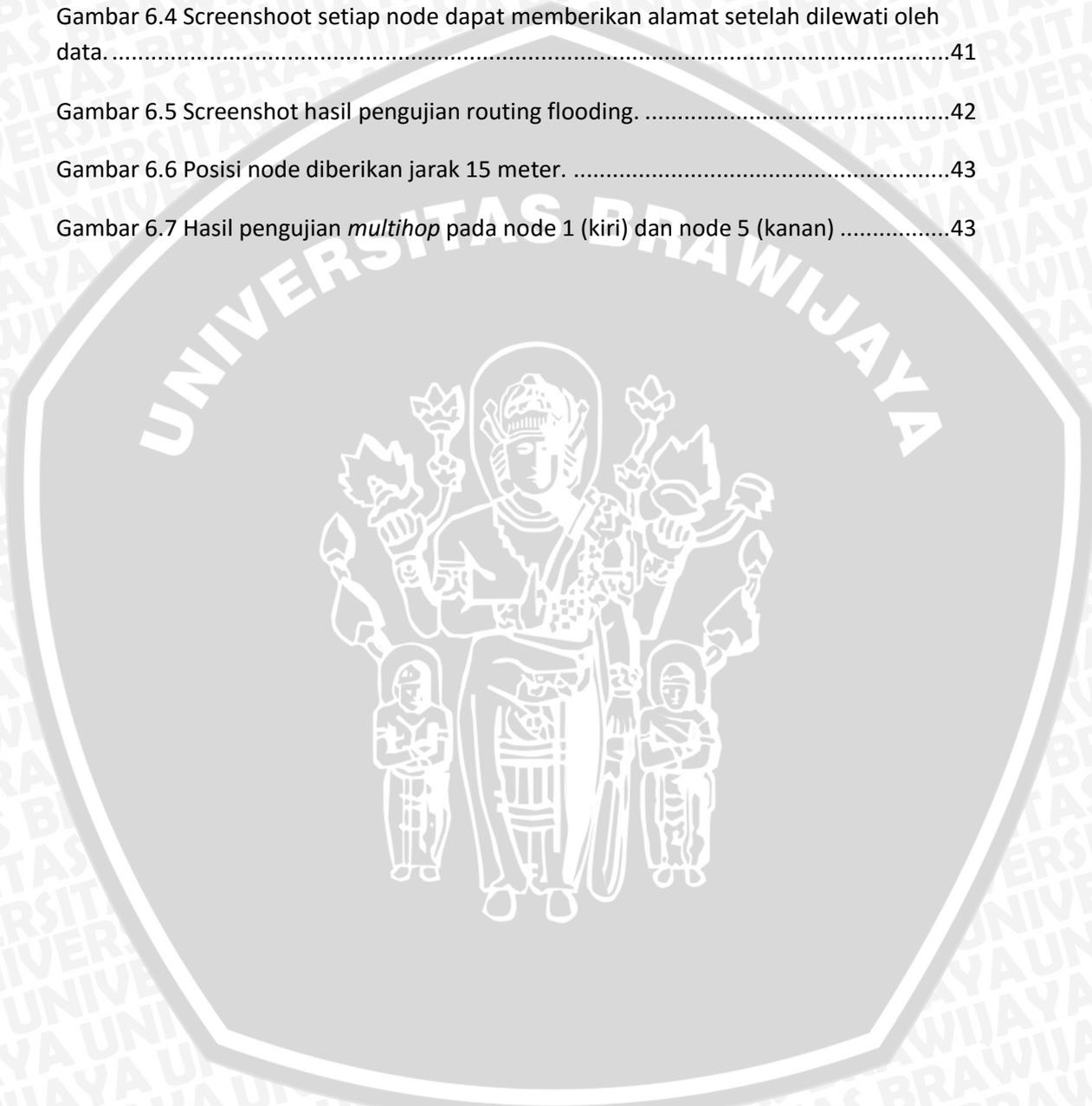
| | |
|--|----|
| Tabel 5.1 Komunikasi Pin Arduino Nano dan Pin nRF24L01 | 26 |
| Tabel 5.2 Daftar Tabel Node Tetangga | 29 |
| Tabel 5.3 Spesifikasi Arduino Nano | 30 |
| Tabel 5.4 Spesifikasi nRF24L01 | 31 |
| Tabel 5.5 Spesifikasi PC/Laptop | 32 |



DAFTAR GAMBAR

| | |
|---|----|
| Gambar 2.1 Topologi pada OMNeT++ | 5 |
| Gambar 2.2 <i>Flat-Bassed Architecture</i> Sumber : (Zheng & Jamalipour, 2009)..... | 7 |
| Gambar 2.3 Topologi jaringan <i>hierarchical</i> | 7 |
| Gambar 2.4 Topologi WSN singlehop..... | 8 |
| Gambar 2.5 topologi WSN multihop | 8 |
| Gambar 2.6 Arduino Nano (tampak atas) (tampak bawah) | 9 |
| Gambar 2.7 NRF24L01 | 11 |
| Gambar 2.8 Mekanisme cara kerja flooding | 11 |
| Gambar 2.9 Implosion pada flooding..... | 12 |
| Gambar 2.10 Masalah Overlapping: node C menerima data Y oleh A dan B | 12 |
| Gambar 3.1 Blok diagram metode penelitian | 14 |
| Gambar 3.2 Diagram Blok Pengiriman dan Penerimaan Antar Node | 17 |
| Gambar 3.3 Alamat pada node-node | 18 |
| Gambar 3.4 Aplikasi arduino untuk membuat code..... | 19 |
| Gambar 5.1 Gambar skematik Arduino dan nRF24L01 | 26 |
| Gambar 5.2 Flowchart sistem routing flooding..... | 28 |
| Gambar 5.3 Topologi yang Digunakan untuk routing flooding | 29 |
| Gambar 5.4 Struktur Data | 30 |
| Gambar 5.5 Implementasi Node dan tersambung ke PC/Laptop | 33 |
| Gambar 5.6 Topologi yang digunakan protokol <i>routing flooding</i> | 34 |
| Gambar 5.7 Rangkaian semua node..... | 34 |
| Gambar 5.8 Impelementasi sistem pada Program Arudino IDE..... | 35 |
| Gambar 5.9 Inisialisasi masing-masing node..... | 36 |
| Gambar 5.10 Inisialisasi Node Tetangga | 36 |
| Gambar 5.11 Format Paket Data..... | 37 |

| | |
|---|----|
| Gambar 6.1 Jalur Routing data..... | 39 |
| Gambar 6.2 Screenshoot pada node 1 saat mengirimkan data dan node 2 saat menerima data..... | 39 |
| Gambar 6.3 Jalur routing pengiriman data | 41 |
| Gambar 6.4 Screenshoot setiap node dapat memberikan alamat setelah dilewati oleh data..... | 41 |
| Gambar 6.5 Screenshot hasil pengujian routing flooding. | 42 |
| Gambar 6.6 Posisi node diberikan jarak 15 meter. | 43 |
| Gambar 6.7 Hasil pengujian <i>multihop</i> pada node 1 (kiri) dan node 5 (kanan) | 43 |



BAB 1 PENDAHULUAN

1.1 Latar belakang

Teknologi terus berkembang dari tahun ke tahun, hal ini tidak dapat dipungkiri oleh siapapun. Teknologi mulai memasuki berbagai sektor, baik pertanian, kesehatan dan masih banyak yang lain. Teknologi sendiri diyakini dapat memudahkan untuk pengerjaan, baik pengerjaan di lapangan maupun di dalam ruangan. Teknologi yang tengah berkembang akhir ini adalah teknologi berbasis *wireless*, yaitu teknologi yang sudah tidak menggunakan kabel (*wire*). *Wireless sensor network* merupakan kumpulan sejumlah node yang diatur dalam sebuah jaringan yang saling bekerjasama. *Wireless sensor network* merupakan node sensor yang kecil terdiri dari sensor (suhu, cahaya, kelembaban, radiasi dan yang lainnya), mikroprosesor, memori, transceiver, dan power supply. (Norouzi & Zaim, 2012)

Wireless sensor network dapat diimplementasikan ke dalam beberapa bidang, peggunaannya juga mudah serta didukung dengan harga yang terjangkau serta banyak di pasaran. Pengaplikasiannya yaitu dengan memasang beberapa node dan sink node yang nantinya node akan mengirimkan data kepada sink node. Setiap node dituntut agar bisa beradaptasi agar mampu bekerja pada kondisi tertentu, node tersebut juga tidak memiliki topologi yang tetap (Sharma, et al., 2014). Pengaplikasian sendiri memerlukan beberapa metode routing yang akan dipakai pada sistem. Protokol routing yang ada pada *wireless sensor network* terdiri atas *flooding routing*, *gossiping routing*, LEACH, SPIN, GBR, PEGASIS dan lain-lain. Algoritma *flooding* merupakan algoritma sederhana yaitu dengan meneruskan data dengan cara membanjiri semua jaringan dengan data yang sama. Algoritma ini dipilih untuk menjaga agar data tidak hilang saat pengiriman, yaitu data akan dikirimkan ke seluruh alamat hingga data sampai pada tujuan. Penerapan algoritma *flooding* yaitu pada sebuah ruangan untuk mengetahui suhu ruangan, dimana setiap ruangan akan mengirimkan data ke node yang lain agar diteruskan hingga mencapai base station, algoritma *flooding* digunakan untuk dapat mengirimkan data ke semua alamat atau ruangan yang bertujuan agar data tetap sampai pada tujuan.

Pada penelitian sebelumnya telah digunakan protokol routing *flooding* pada *wireless sensor network* untuk bencana alam oleh Alma Otero, Monica Huerta, Giovanni Sagbay. Pada penelitian tersebut menggunakan *software* OMNeT++ sebagai simulasi routing *flooding* yang berjudul "*A flooding routing algorithm for a wireless sensor network for seismic events*". Untuk permasalahan tersebut akan digunakan protokol routing menggunakan algoritma *flooding* pada *wireless sensor network*. Node dan sink node akan menggunakan arduino nano dan media komunikasi radio frekuensi menggunakan nrf24L01.

1.2 Rumusan masalah

1. Bagaimana mengimplementasikan protokol routing *flooding* pada *wireless sensor network* menggunakan media komunikasi RF?
2. Bagaimana mekanisme komunikasi data dengan menggunakan protokol *routing flooding*?
3. Bagaimana hasil pengujian protokol routing *flooding* pada *wireless sensor network* dengan menggunakan media komunikasi radio frekuensi?

1.3 Tujuan

Tujuan umum:

Dilakukannya penelitian ini untuk mengimplementasikan dan menganalisis protokol routing dengan algoritma *flooding* pada *wireless sensor network* menggunakan media komunikasi radio frekuensi

Tujuan khusus:

1. Untuk mengetahui perancangan protokol routing *flooding* pada *wireless sensor network* menggunakan media komunikasi RF.
2. Untuk mengetahui mekanisme komunikasi data yang digunakan saat pengiriman data dari node ke sink node.
3. Untuk mengetahui kinerja protokol routing *flooding* pada *wireless sensor network*.

1.4 Manfaat

Penelitian ini diharapkan dapat bermanfaat baik bagi penulis maupun bagi Pengguna, antara lain:

1. Bagi Penulis
 - Mengaplikasikan ilmu yang diperoleh selama mengikuti perkuliahan di Sistem Komputer Universitas Brawijaya
 - Mendapatkan pengetahuan bagaimana mengimplementasikan protokol routing *flooding* pada *wireless sensor network* dengan menggunakan komunikasi frekuensi radio.
2. Bagi Pengguna
 - Mengetahui kinerja protokol *flooding* pada *wireless sensor network* dengan menggunakan media komunikasi radio frekuensi.

1.5 Batasan masalah

Berdasarkan latar belakang dan rumusan masalah yang telah diuraikan, agar pembahasan tidak berkembang menuju permasalahan yang lebih lebar, maka penelitian dibatasi dalam hal:

1. Pengiriman data yang dilakukan oleh tiap-tiap node.

2. Parameter uji yang digunakan adalah paket data yang menyimpan alamat yang dilewati oleh data.
3. Node-node yang dibangun digunakan untuk pengiriman dan penerimaan data.
4. Mikrokontroler yang digunakan adalah Arduino Nano.
5. Mengabaikan duplikasi data.
6. Menggunakan topologi mesh.

1.6 Sistematika pembahasan

Untuk mencapai tujuan yang diharapkan, maka sistematika penulisan yang disusun dalam laporan tugas akhir ini sebagai berikut:

| | | |
|--------------|------------------------------|---|
| BAB 1 | Pendahuluan | Berisi tentang latar belakang masalah, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan penelitian / ruang lingkup penelitian, sistematika penelitian, dan jadwal penelitian. |
| BAB 2 | Dasar Teori | Bab ini membahas mengenai teori-teori yang berkaitan dan menunjang dalam penyelesaian skripsi. Dasar teori yang diambil berasal dari jurnal, buku, dan sumber referensi lainnya yang berhubungan dengan topic yang akan diteliti. |
| BAB 3 | Metodologi Penelitian | Langkah-langkah dalam merancang penerapan wireless sensor network menggunakan topologi mesh. Selain itu, pada bab ini juga akan dibahas tahap-tahap implementasi, analisis dan pengujian sistem yang dirancang |
| BAB 4 | Rekayasa Kebutuhan | Membahas analisis kebutuhan dan perancangan sesuai dengan teori yang ada |
| BAB 5 | Implementasi | Membahas tentang implementasi berdasarkan metodologi yang dibuat |
| BAB 6 | Pengujian | Memuat tentang hasil pengujian dan analisis terhadap sistem yang sudah di implementasikan dan sudah memenuhi rekayasa kebutuhan |

BAB 7

Penutup

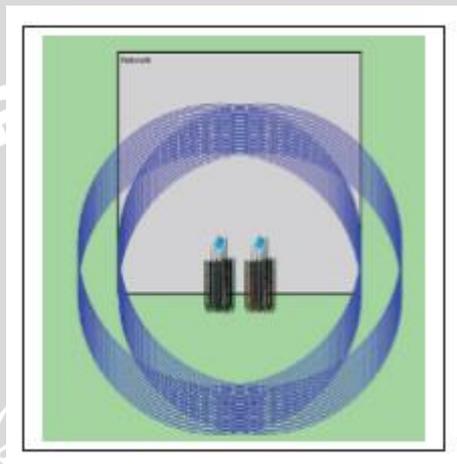
Memuat kesimpulan yang diperoleh dari pembuatan dan pengujian perangkat lunak yang dikembangkan serta saran-saran untuk pengembangan selanjutnya



BAB 2 LANDASAN KEPUSTAKAAN

2.1 Tinjauan Pustaka

Pada penelitian routing flooding sebelumnya masih bersifat simulasi. Yaitu telah dilakukan oleh Alma Otero, Monica Huerta, Giovanni Sagbay. Pada penelitian tersebut menggunakan *software* OMNeT++ sebagai simulasi routing *flooding* yang berjudul “A flooding routing algorithm for a wireless sensor network for seismic events”. Pada penelitian tersebut juga menggunakan beberapa routing lain seperti AODV, DSR, OSLR.



Gambar 2.1 Topologi pada OMNeT++

Pada penelitian ini setiap node akan ditaruh di dasar sebuah gedung bertingkat, dimana saat terjadi gempa setiap lantai akan diberi node untuk menerima data dari node lantai bawah hingga data dikirimkan ke node paling atas. Penelitian ini juga menghitung delay pengiriman data dan juga *packet loss* yang dihasilkan dari routing *flooding*.

2.2 Wireless Sensor Network

Wireless Sensor Network (WSN) merupakan sebuah kumpulan node yang saling terorganisir di dalam sebuah jaringan. Setiap node memiliki kemampuan untuk memproses (satu atau lebih mikrokontroler), memiliki beberapa jenis memori (program, data dan flash memories), memiliki sebuah RF (Radio Frekuensi) (biasanya menggunakan sebuah *omnidirectional antenna*), memiliki sumber tegangan (baterai atau *solar cells*), dan mengakomodasi berbagai macam node dan actuators. Setiap node berkomunikasi secara *wireless* (tanpa kabel) dan bisa mengatur secara sendiri setelah dikerahkan ke mode ad-hoc. (Stankovic, 2006)

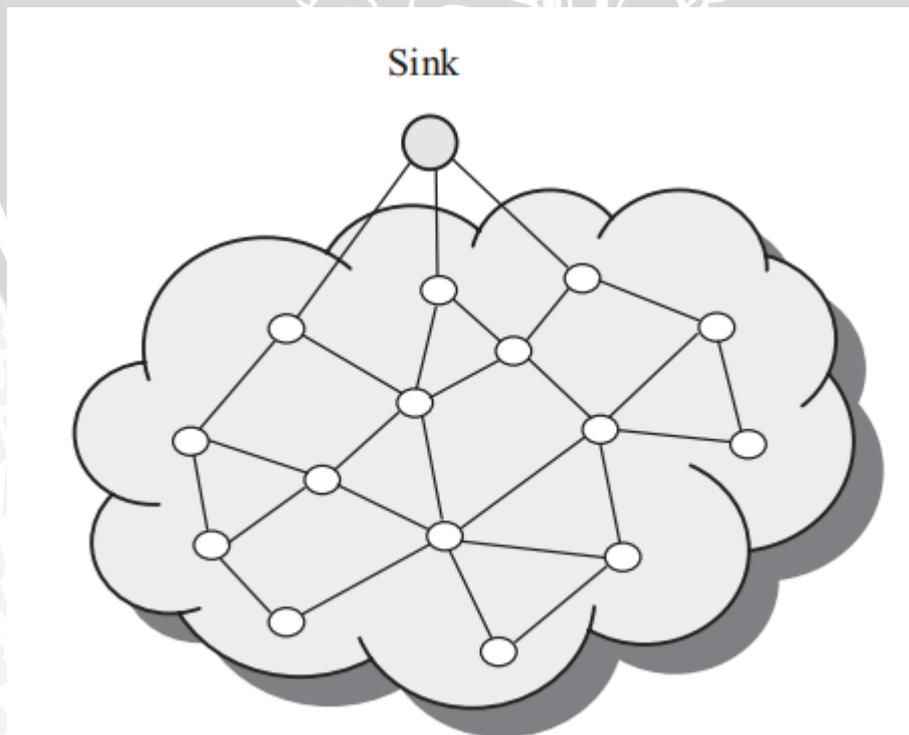
Wireless Sensor Network terdiri dari *low-cost*, *low-power*, dan multifungsi. Node ditempatkan pada tempat yang menarik. Node memiliki ukuran yang kecil, dilengkapi dengan sensor, mikroprosesor, dan RF (Radio Frekuensi), oleh karena

itu setiap node tidak hanya memiliki kemampuan penginderaan tetapi juga memiliki kemampuan mengolah data dan saling berkomunikasi. Setiap node saling berkomunikasi melalui media komunikasi nirkabel dan saling berkolaborasi untuk mencapai sebuah tujuan, misalnya, pemantauan lingkungan, pengawasan lapangan, dan proses pengendalian industri. (Zheng & Jamalipour, 2009)

Wireless Sensor Network memiliki bagian yang dibagi menjadi 4 yaitu sistem sensing, processing, communication dan *power*. Penyatuan sistem haruslah diperhatikan saat awal melakukan perancangan sistem, terutama pada sistem *processing*. Sistem *processing* dianggap sangat penting karena sistem tersebut dapat berpengaruh terhadap performa serta konsumsi energi. Dalam WSN sendiri dapat menggunakan beberapa sensor node. WSN dapat mengimplementasikan sensor node cukup banyak, namun banyak atau sedikitnya node yang digunakan dikembalikan lagi kepada sistem itu sendiri.

2.1.1 Flat based

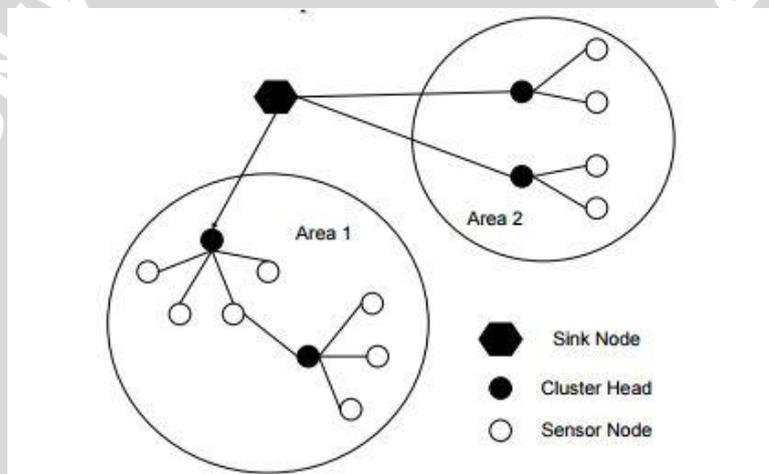
Arsitektur jaringan *Wireless Sensor Network* tradisional pada umumnya mengadaptasi flat structure. Dalam arsitektur flat-based pada umumnya setiap node memiliki peran yang sama dan tidak ada yang terpusat. Hanya terdapat 2 macam node secara fungsional sensor node dan sink node seperti pada gambar 2.1. Pada flat-based pendistribusian informasi dilakukan dengan menggunakan *data-routing centric*, dimana sink akan mentransmitkan *query* ke semua node di dalam satu area yang dipilih dengan membanjiri semua node dan node yang memiliki kriteria yang sama dengan sink yang akan mengirim atau merespon kepada sink node.



Gambar 2.2 Flat-Based Architecture
 Sumber : (Zheng & Jamalipour, 2009)

2.2.2 Hierarchical-based

Arsitektur ini bertujuan untuk mencoba penghematan energi dengan menggunakan node yang dibuat menjadi cluster. Topologi jaringan hierachical dapat dilihat pada gambar 2.2. Topologi jaringan hierachical disusun secara hierachical sehingga terdapat 3 macam node, yaitu *cluster head*, *sink node*, dan *sensor node*. Pada jaringan *hierachical*, *sensor node* yang berada dalam cluster akan mengirimkan informasi ke *cluster head*, *cluster head* bertujuan mengirimkan informasi dan meneruskannya kepada *sink node*. Dalam topologi hierachical memiliki kekurangan karena setiap *cluster head* memiliki tugas menampung dan mengirimkan informasi data dari beberapa *sensor node* yang pimpinnya. Sehingga *cluster head* dapat membebani energi yang digunakan dan jaringan komunikasinya akan padat.

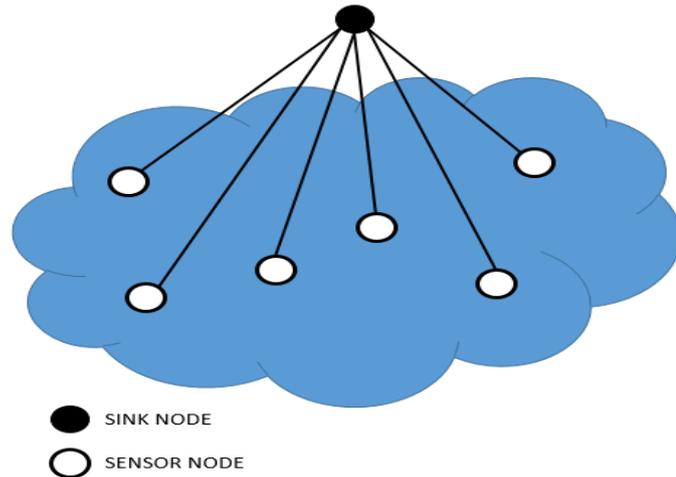


Gambar 2.3 Topologi jaringan hierachical

Sumber : (Zheng & Jamalipour, 2009)

Berikut ini karakteristik jaringan pada *wireless sensor network* berdasarkan node sensor dibagi menjadi dua kategori, yaitu singlehop dan mutihop. Pada topologi singlehop merupakan topologi WSN sederhana, dimana setiap node berkomunikasi langsung dengan gateway atau pengumpul data.

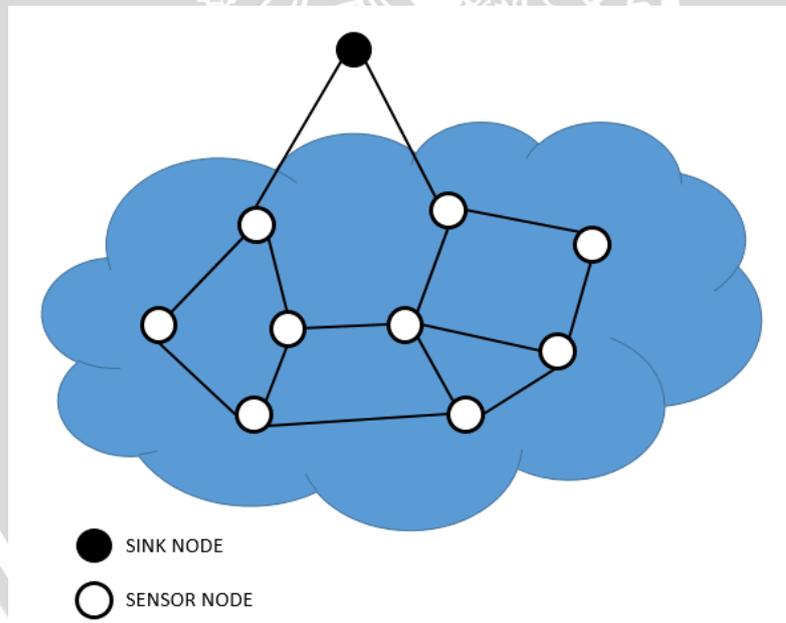




Gambar 2.4 Topologi WSN singlehop

Sumber : (Zheng & Jamalipour, 2009)

Pada topologi jaringan WSN multihop, data akan dikirim melalui sensor terdekat ke sensor terdekat yang lain sampai mencapai ke *sink node*. Pada topologi ini, rute ditentukan oleh protokol routing tertentu, tergantung pada jaringan acak atau terstruktur seperti gambar 2.4 dibawah ini.



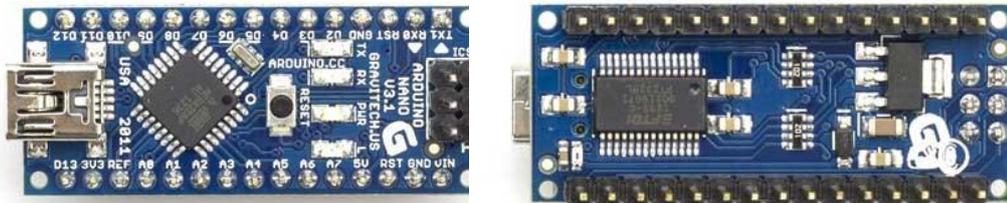
Gambar 2.5 topologi WSN multihop

Sumber : (Zheng & Jamalipour, 2009)

2.3 Arduino

Arduino merupakan salah satu mikrokontroler yang sering kita temukan di pasaran. Selain harganya yang terjangkau, arduino didukung dengan aplikasi yang opensource. Arduino bisa digunakan untuk pengembangan objek interaktif, mengambil data input dari berbagai switch atau sensor, dan mengendalikan berbagai lampu, motor, dan output fisik lainnya. Aplikasi arduino sendiri bisa dijalankan pada Windows, Macintosh OSX, dan Linux. Arduino Nano 3.0 menggunakan Atmega328 sedangkan Arduino nano 2.3 menggunakan Atmega168. Berikut adalah spesifikasi Arduino Nano:

- Microcontroller : Atmel Atmega168 or Atmega328
- Voltage : 5V
- Input Voltage (recommended) : 7-12V
- Input Voltage (limits) : 6-20v
- Digital I/O Pins : 14 (of which 6 provide PWM output)
- Analog input pins: 8
- DC Current I/O Pin = 40mA
- Flash Memory = 32KB(ATmega328) or 32KB(ATmega328)of which 2KB used by bootloader
- SRAM = 2KB (ATmega328) or 1KB (Atmega168)
- EEPROM = 1KB(ATmega328) or 512bytes(ATmega168)
- Clock Speed : 16 Mhz
- Length : 45 mm
- Width : 18mm
- Wight : 5g



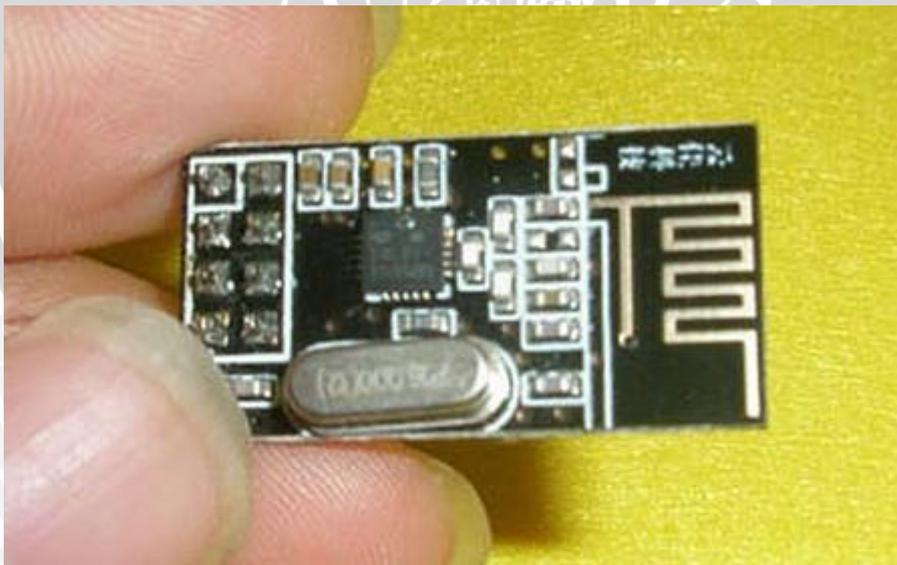
Gambar 2.6 Arduino Nano (tampak atas) (tampak bawah)

Sumber : (<http://www.arduino.cc/en/Main/ArduinoBoardNano>, 2015)

2.4 NRF24L01

Radio merupakan suatu teknologi yang dipakai untuk pengiriman sinyal, yaitu dengan menggunakan cara modulasi dan cara radiasi gelombang elektromagnetik. Gelombang ini melintas dan merambat melalui udara serta bisa juga merambat lewat ruang angkasa yang hampa akan udara, sebab gelombang ini memerlukan medium pengangkut (misalnya seperti molekul udara). Gelombang radio sendiri memiliki frekuensi dari yang paling kecil hingga ke yang paling panjang. NRF24L01 merupakan modul RF yang bekerja pada frekuensi 2.4 GHz. Selain harganya terjangkau, NRF24L01 ini dapat bekerja pada tegangan yang rendah yaitu 5V DC. Modul NRF24L01 ini menggunakan interface SPI yang digunakan untuk berkomunikasi. NRF24L01 memiliki baseband logic Enhanced Shockburst hardware protocol controller. NRF24L01 memiliki true ULP solution, yang memungkinkan daya pada baterai bisa bertahan lebih lama. Modul NRF24L01 memiliki 8 pin yaitu :

1. VCC
2. GND
3. CE
4. CSN
5. MOSI
6. MISO
7. SCK
8. IRQ

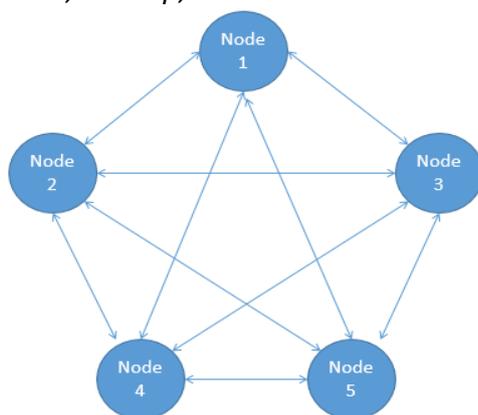


Gambar 2.7 NRF24L01

Sumber : (<http://arduino-info.wikispaces.com/file/view/nRF24L01-3.jpg/243329431/nRF24L01-3.jpg> , 2015)

2.5 Protokol Flooding

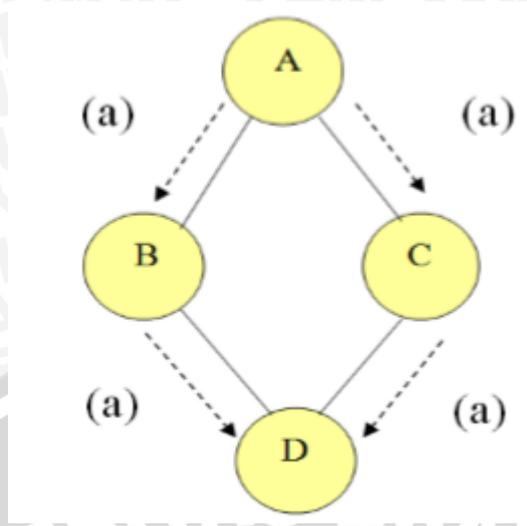
Flooding merupakan routing sederhana dimana setiap paket yang masuk dikirim melalui setiap link keluar kecuali salah satu node tiba pada tujuan. *Flooding* merupakan strategi lama dan sederhana untuk menyebarkan informasi ke jaringan atau untuk mencapai node di lokasi yang tidak diketahui dengan membanjiri seluruh jaringan. (Sharma, et al., 2014). Namun flooding sendiri memiliki beberapa kelemahan yaitu *implosion*, *overlap*, dan *resources blindness*.



Gambar 2.8 Mekanisme cara kerja flooding

2.5.1 Implosion

Dalam *flooding*, sebuah node selalu mengirim data ke node tetangga meskipun tetangga tersebut telah mendapatkan data dari sumber yang lain. Hasilnya terjadi masalah yaitu implosion (ledakan) seperti pada ilustrasi dibawah. Node A mulai mengirimkan data dengan metode *flooding* ke kedua tetangganya yaitu B dan C. Kedua node akan mendapatkan data dari A dan kemudian mengirimkan kembali ke tetangganya yaitu ke D. Dengan demikian keduanya akan mengirimkan data yang sama kepada D.

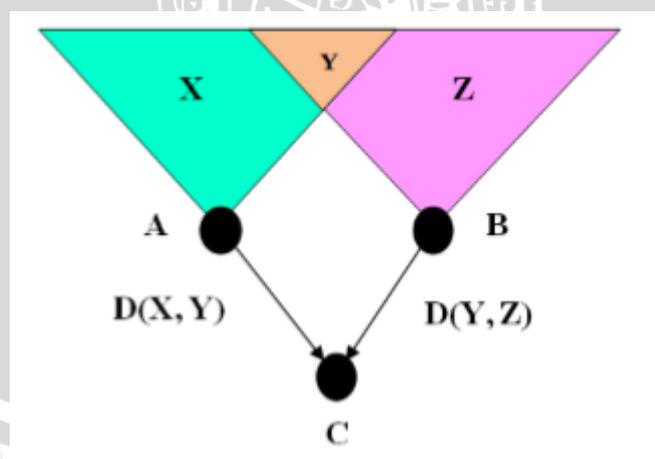


Gambar 2.9 Implosion pada flooding

Sumber : (Sharma, et al., 2014)

2.5.2 Overlap

Node sensor sering menutupi area dari node lain dimana akan melakukan tumpukan karena node saling berkumpul dan saling menumpuk data. Gambar dibawah ini menggambarkan apa yang terjadi ketika dua node (A dan B) saling mengumpulkan data yang bertumpukan lalu mengirimkan dan mengirimkan data ke tetangga bersama mereka (C). Pada algoritma *flooding* ini, penggunaan energi dan bandwidth dengan mengirimkan dua salinan data dari sepotong data ke node yang sama.



Gambar 2.10 Masalah Overlapping: node C menerima data Y oleh A dan B

Sumber : (Sharma, et al., 2014)

2.5.3 Resource Blindness

Algoritma *flooding* mengabaikan keterbatasan sumber daya dari masing-masing node. Node tidak dapat memodifikasi operasi mereka sendiri berdasarkan jumlah energi mereka hadapi. Jadi node dengan sumber daya yang sedikit dapat cepat habis (Sharma, et al., 2014).

2.5.4 Mirf

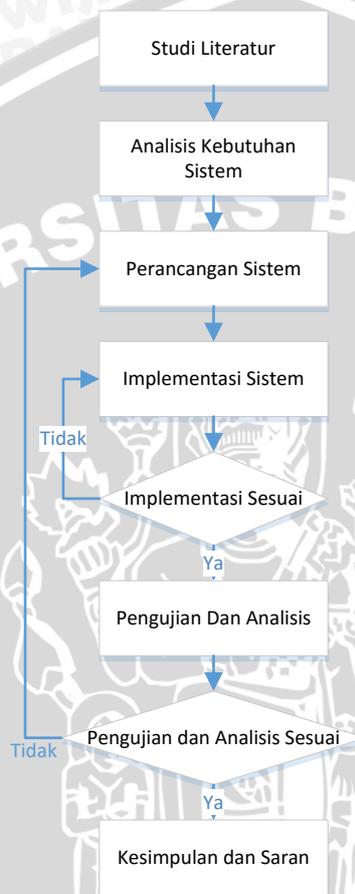
Mirf merupakan salah satu library yang ada pada Arduino, pada dasarnya Mirf digunakan untuk implementasi dan transmisi informasi dari satu Arduino yang bekerja pada satu lingkungan sensor ke *base station*. Untuk mempermudah proses pengiriman informasi, Mirf menggunakan beberapa metode pendekatan yaitu:

- Data yang ditransmisikan oleh sensor berupa string ASCII sederhana
- Setiap karakter ASCII yang ditransmisikan secara individual (tidak efisien namun benar-benar sederhana)
- *Base station* hanya menjadi pipa data yang diterima melalui port serial ke PC
- PC menjalankan perangkat lunak (misalnya berdasarkan *Processing*) untuk melakukan pemeriksaan kesalahan dan menampilkan data yang diterima

Sumber: (<https://arduino-info.wikispaces.com/nRF24L01-Mirf-Examples>).

BAB 3 METODOLOGI

Dalam perancangan sistem terdapat proses-proses yang saling berkaitan agar menciptakan suatu perancangan yang terstruktur dengan baik. Pada bab ini akan dijelaskan langkah-langkah yang dilakukan dalam analisis dari sistem yang dirancang.



Gambar 3.1 Blok diagram metode penelitian

3.1 Studi Literatur

Studi literatur bertujuan untuk mencari dasar-dasar teori yang dapat dijadikan bahan referensi dalam merancang sistem. Berbagai artikel maupun penelitian yang berkaitan dengan perancangan sistem dianalisis agar dapat menciptakan sistem yang handal diantaranya:

- a. Komunikasi radio frekuensi nRF24L01
- b. Routing protokol *Flooding* pada *wireless sensor network*.
- c. Sistem Arduino Nano

3.2 Analisis Kebutuhan

Analisis kebutuhan akan mengidentifikasi perangkat-perangkat yang akan digunakan dalam membangun dan mengimplementasikan sistem. Perangkat yang digunakan harus sesuai dengan fungsi yang akan dijalankan dalam kerja oleh sistem sehingga dapat mempermudah dalam perancangan. Komponen yang dibutuhkan dalam perancangan dan implementasi adalah:

- Kebutuhan fungsional
 1. Sistem dapat melakukan routing sesuai dengan algoritma *flooding*
 2. Sistem dapat melakukan pengiriman data dan penerimaan data
 3. Sistem menampilkan hasil pengiriman dan penerima data.
- Kebutuhan non fungsional
 1. Proses pengiriman dan meneruskan data akan berhenti apabila data sudah sampai pada tujuan
 2. Komunikasi radio frekuensi

3.2.1 Kebutuhan Pengguna

Kebutuhan pengguna dalam sistem ini adalah sistem dapat melihat hasil protokol *routing flooding* pada *wireless sensor network* dengan menggunakan media komunikasi RF.

3.2.2 Kebutuhan Sistem

Dalam perancangan dan impleementasi protokol *routing flooding*, adapun kebutuhan sistem sebagai berikut:

- a. Sistem dapat melakukan pengiriman dan penerimaan data.
- b. Sistem menampilkan hasil pengiriman dan penerimaan data.
- c. Proses pengiriman dan meneruskan paket akan berhenti apabila data sudah sampai di tujuan atau telah melampaui jumlah node.

3.2.3 Kebutuhan Perangkat Keras

Dalam mengimplementasikan protokol routing *flooding*, kebutuhan perangkat keras yang dibutuhkan adalah sebagai berikut.

1. Arduino Nano digunakan pada setiap node untuk menampilkan hasil pengujian di PC/laptop.

2. NRF240L01 merupakan transmitter frekuensi radio yang dipasang disetiap node untuk melakukan pengiriman dan penerimaan informasi data.
3. Laptop/PC digunakan untuk memberikan inputan dan mengirim sketch Arduino serta melihat tampilan serial.
4. Jumper digunakan untuk menghubungkan antar perangkat nRF24L01 dan Arduino Nano
5. Kabel USB digunakan untuk menghubungkan Arduino Nano ke Laptop/PC.

3.2.4 Kebutuhan Perangkat Lunak

Selain perangkat keras, dalam mengimplementasikan protokol *routing gossiping* perangkat lunak sangat diperlukan untuk dapat melakukan implementasi tersebut. Adapun penggunaan perangkat lunak sebagai berikut.

- a. Arduino IDE digunakan untuk mengirim *sketch* kedalam perangkat Arduino Nano.
- b. Windows 8.1 merupakan sistem operasi yang digunakan untuk menjalankan program Arduino IDE.

3.3 Perancangan Sistem

Dalam mengimplementasikan protokol *routing flooding* ini ada beberapa perancangan yang harus dilakukan. Perancangan rangkaian node, perancangan topology node, dan perancangan skema komunikasi.

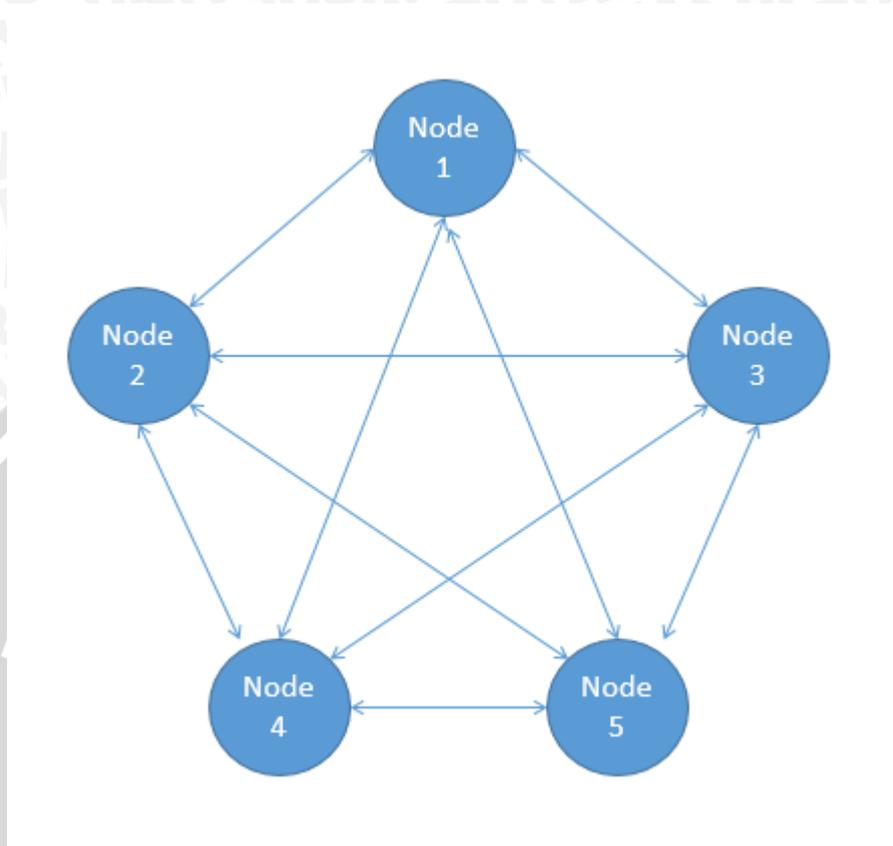
3.3.1 Perancangan Rangkaian Node

Dalam perancangan node akan menggunakan Arduino Nano sebagai mikrokontroler dan nRF24L01 sebagai media komunikasi radio frekuensi yang digunakan. nRF24L01 akan dihubungkan pada Arduino Nano. Arduino Nano akan diberikan *power* untuk dapat menjalankan sistem Arduino tersebut. Power yang digunakan berasal dari PC/Laptop dan adaptor dari USB HUB. PC/Laptop digunakan untuk dapat memberikan inputan dan menampilkan hasil dari serial *monitor* Arduino.

3.3.2 Perancangan Topologi Node

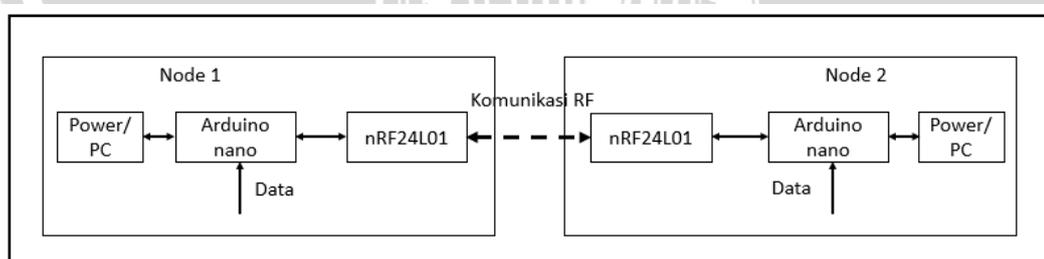
Topologi yang digunakan dalam mengimplementasikan protokol *routing flooding* adalah topologi mesh dan penempatan node akan diletakkan secara acak. Topologi ini digunakan karena pengiriman data secara *flood* sehingga untuk

penentuan node tetangga ditetapkan secara statis, atau di deklarasikan di setiap node .Illustrasi topologi yang digunakan seperti pada gambar dibawah ini.



Gambar 3.1 Topologi yang digunakan routing flooding.

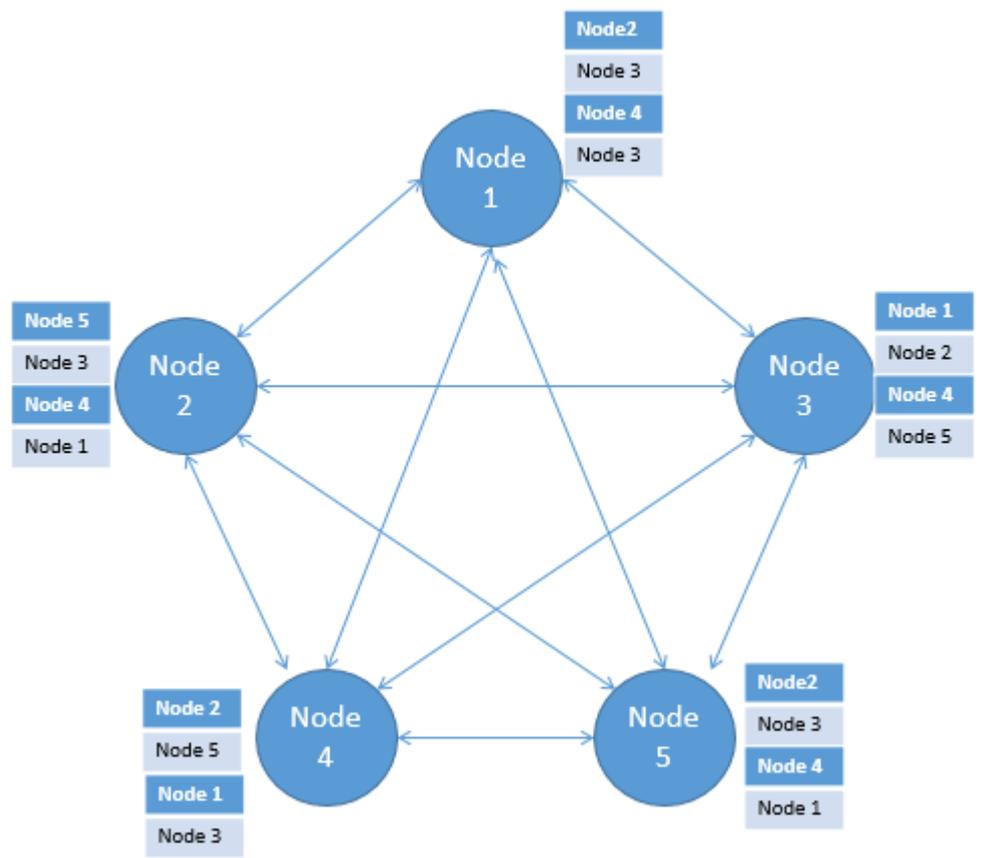
Dalam pengiriman dan penerimaan data menggunakan nRF24L01, komunikasi antar node dapat digambarkan dengan diagram blok pada gambar 3.4.



Gambar 3.2 Diagram Blok Pengiriman dan Penerimaan Antar Node

3.3.3 Perancangan skema komunikasi untuk protokol routing flooding

Skema protokol routing yang digunakan pada topologi ini menggunakan 4 tahap. Tahap pertama yaitu dengan menentukan alamat node tetangga di setiap node, penentuan alamat sendiri dideklarasikan di setiap node, yaitu semua node memiliki informasi alamat pada satu wilayah. Tahap kedua yaitu pengiriman data berdasarkan masukan dari keyboard PC/Laptop, nantinya data akan dikirimkan secara flood ke seluruh node. Tahap ketiga yaitu setiap node akan memberikan tanda bahwa data telah melewati suatu node dengan memberikan tanda pada array berupa angka yang menunjukkan alamat node yang sudah dilewati.

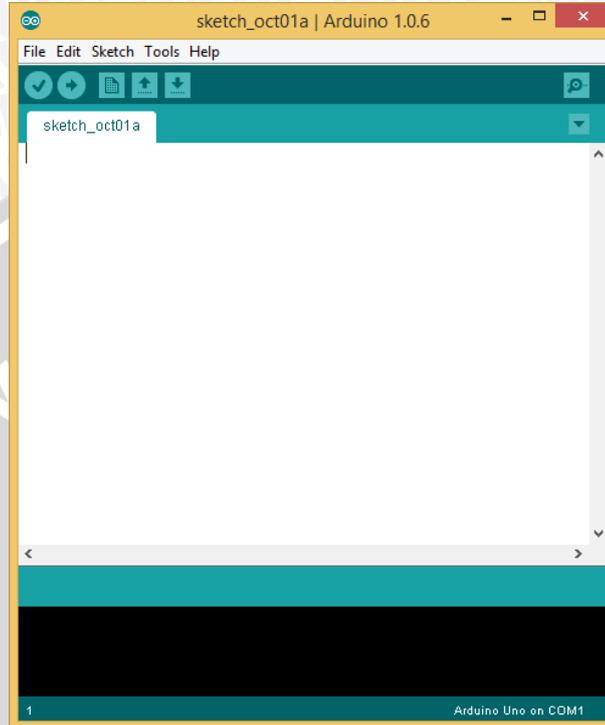


Gambar 3.3 Alamat pada node-node

Tahap ke 4 yaitu setelah node sampai di node tetangga dan telah diberi tanda maka setiap node akan menampilkan darimana data berasal dan node mana saja yang telah dilewati. Pengiriman data akan menggunakan 6 byte, dimana pada 1 byte pertama akan digunakan sebagai header yaitu berupa data. Byte selanjutnya berisikan node yang telah dilewati oleh data.

3.4 Implementasi

Implementasi dilakukan berdasarkan perancangan yang telah dibuat. Implementasi pertama yang dilakukan adalah membuat code program untuk setiap node pada aplikasi arduino seperti pada gambar 3.5.



Gambar 3.4 Aplikasi arduino untuk membuat code

Kode program akan dibuat sesuai skema routing yang telah dibuat, ini dilakukan agar node-node berjalan sesuai topologi dan sistem yang telah dirancang. Apabila kode program semua node sudah dibuat, maka kode program akan di upload ke semua node. Semua node menggunakan media komunikasi RF yaitu NRF24L01 untuk bisa saling berkomunikasi, sehingga sensor node dapat terhubung ke *sink node*.

3.5 Pengujian dan Analisis

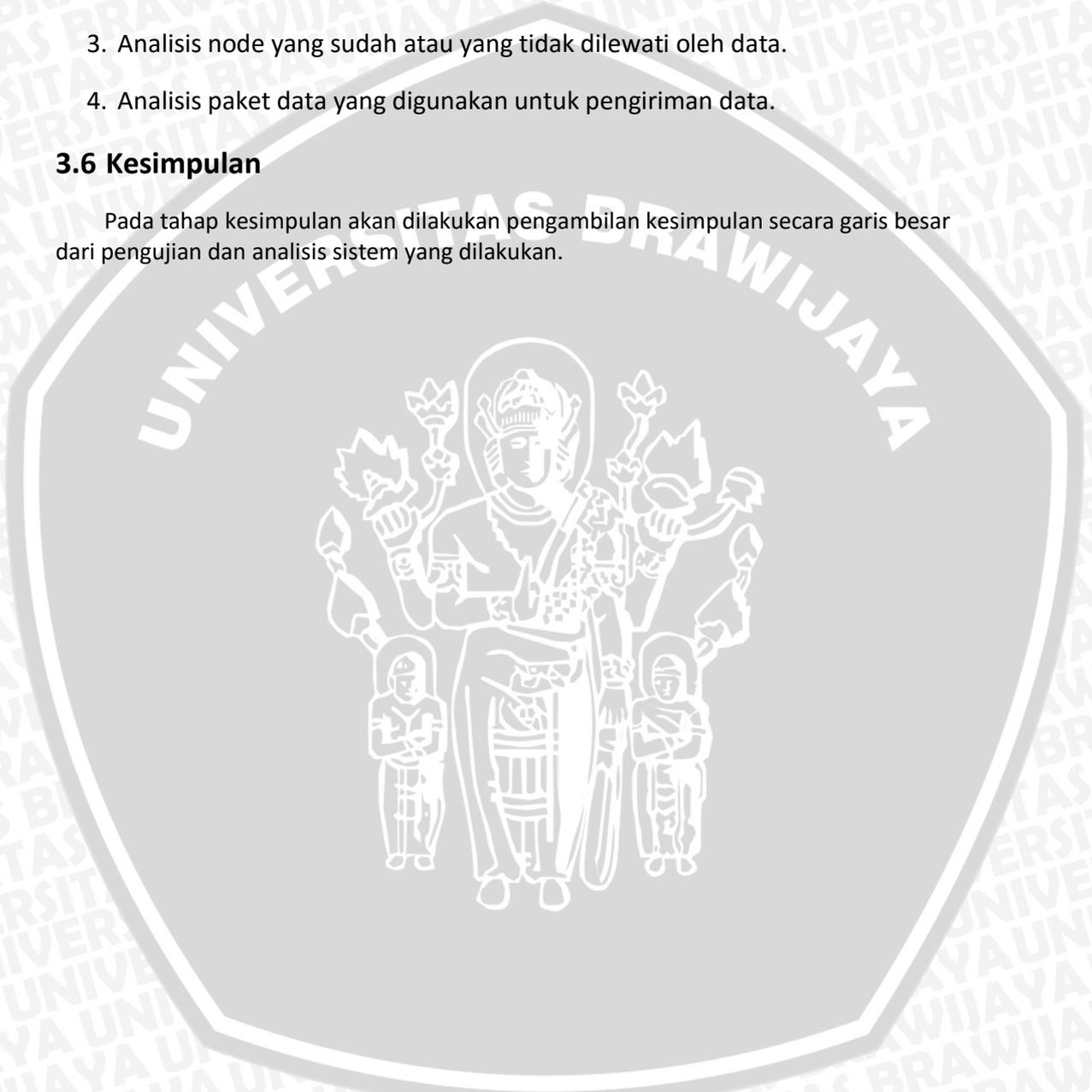
Pengujian dilakukan pada setiap node yang telah diimplementasikan. Pengujian dilakukan pada setiap node yang terdapat arduino nano dan nRF4L01. Salah satu node akan dipilih untuk pengiriman pertama dan satu node lagi untuk menjadi target tujuan penerimaan data terakhir. Dimana nanti hasilnya akan ditampilkan pada Layar laptop/PC.

Analisis yang dilakukan adalah sebagai berikut:

1. Analisis semua node yang dibutuhkan dan output yang dihasilkan oleh semua node.
2. Analisis protokol routing *flooding* pada *wireless sensor network* menggunakan media komunikasi radio frekuensi.
3. Analisis node yang sudah atau yang tidak dilewati oleh data.
4. Analisis paket data yang digunakan untuk pengiriman data.

3.6 Kesimpulan

Pada tahap kesimpulan akan dilakukan pengambilan kesimpulan secara garis besar dari pengujian dan analisis sistem yang dilakukan.



BAB 4 REKAYASA KEBUTUHAN

4.1 Gambaran Umum

Bab ini bertujuan untuk menjelaskan secara rinci tentang rekayasa kebutuhan yang harus dipenuhi untuk perancangan hingga implementasi sistem. Sehingga diharapkan implementasi protokol *routing flooding* pada *wireless sensor network* menggunakan media komunikasi radio frekuensi dapat berjalan dengan baik. Pada gambaran umum sistem terdapat tujuan, ruang lingkup, referensi dan sistematika.

4.1.1 Tujuan

Pada sub bab ini menjelaskan secara rinci tentang Sistem protokol *routing flooding* pada *wireless sensor network* dengan menggunakan media komunikasi radio frekuensi (RF) berdasarkan format pada *Software Requirement Specification* (SRS) IEEE 830. Pada bab ini akan dijelaskan tentang tujuan dari sistem, *interface* sistem, fungsi dari sistem, dan batasan tentang system yang akan dibuat.

Sistem ini bertujuan untuk membantu pengguna dalam pengiriman paket data hingga sampai ke tujuan dengan melakukan pengiriman ke seluruh node yang akan dilewati oleh data. Sistem dilengkapi dengan serial monitor yang akan menampilkan informasi tentang paket data yang telah dikirim dan telah melewati beberapa node.

4.1.2 Ruang Lingkup

Pada sistem dibuat bertujuan untuk melakukan implementasi protokol *routing flooding* pada *wireless sensor network* dengan menggunakan media komunikasi radio frekuensi (RF). Protokol *routing flooding* merupakan *traditional routing* pada *wireless sensor network* yaitu algoritma yang mengirimkan data dengan cara meneruskan data ke node selanjutnya, yaitu dengan membanjiri jaringan dengan sebuah data, nantinya data dari node pertama akan sama dengan data di node terakhir. Protokol *routing flooding* diimplementasikan dengan menggunakan node *wireless sensor network* dengan media komunikasi antar node berupa media komunikasi radio frekuensi.

4.1.3 Referensi

IEEE. IEEE Std 830-1998 IEEE *Recommended Practice for Software Requirements*

Specifications. IEEE Computer Society, 1998. Merupakan standar penulisan untuk rekayasa kebutuhan.

4.1.4 Sistematika

Pada sub bab ini akan menjabarkan secara deskriptif hal-hal yang akan ditulis secara garis besar untuk mengetahui pembahasan yang ada, maka perlu adanya kerangka dalam penulisan *software requirement specification*. Kerangka tersebut dibagi menjadi 3 bagian, diantaranya:

1. Pendahuluan

Pada bagian ini terdiri dari tujuan, ruang lingkup, istilah, referensi dan sistematika penulisan.

2. Deskripsi Umum

Bagian ini terdiri dari perspektif produk, fungsi produk karakteristik pengguna, batasan-batasan, asumsi dan ketergantungan.

3. Spesifikasi kebutuhan

Bagian ini terdiri dari kebutuhan fungsional, kebutuhan antarmuka eksternal, kebutuhan performansi, kendala desain, atribut dan kebutuhan lain .

4.2 Deskriptif umum

Pada deskripsi umum bertujuan untuk menjelaskan tentang rekayasa kebutuhan yang harus dipenuhi untuk perancangan hingga implementasi sistem. Sehingga *routing flooding* dapat berjalan dengan baik.

4.2.1 Perspektif Produk/Sistem

Sistem ini bekerja dengan menggunakan rangkaian node, node akan dibuat dari rangkaian *wireless sensor network* yang terhubung dengan modul radio frekuensi. Node akan berhubungan satu sama lain sesuai dengan topologi yang telah dirancang sebelumnya. Sistem dari setiap node yaitu mengirim dan menerima data menggunakan protokol *routing flooding*. Setiap hasil pengiriman dan penerimaan data akan ditampilkan pada PC/Laptop melalui serial monitor dari aplikasi Arduino IDE.

4.2.2 Kegunaan

Protokol *routing flooding* akan mengirimkan data sampai ke tujuan yaitu dengan membanjiri jaringan dengan data yang sama. Setiap node akan mengirimkan data ke seluruh alamat yang ada pada satu jaringan, nantinya setiap node akan meneruskan kembali data ke semua alamat dengan pengiriman unicast hingga data sampai pada tujuan. Setiap data yang sudah melewati alamat node, maka data tersebut tidak akan dikirim kembali ke alamat tadi, untuk mengurangi benturan data.

4.2.3 Karakteristik Pengguna

Pengguna/*user* pada implementasi sistem dapat bertindak di setiap node, dimana pengguna dapat melakukan *monitoring* node dengan menggunakan serial monitor pada aplikasi Arduino IDE.

4.2.4 Lingkungan Operasi

Persyaratan kebutuhan lingkungan yang mendukung kebutuhan sistem untuk melakukan operasi yaitu:

1. penempatan setiap node pada lingkungan harus terlindungi dari gangguan alam seperti hujan. Dikarenakan hujan dapat menyebabkan komponen pada sistem menjadi rusak.
2. Channel yang digunakan untuk komunikasi radio frekuensi harus menggunakan channel yang sama.

4.2.5 Batasan Perancangan dan Implementasi

Beberapa batasan perancangan dan implementasi adalah sebagai berikut.

1. Sistem akan berjalan apabila semua node siap untuk digunakan.
2. Sistem mengirimkan data keseluruhan alamat node.
3. Sistem bekerja apabila node terhubung dengan node lain sesuai topologi yang sudah dirancang sebelumnya.
4. Belum memperhitungkan *delay* dan *throughput* pada paket data.

4.2.6 Asumsi dan Ketergantungan

Asumsi dan ketergantungan sebagai persyaratan sistem antara lain modul *transceiver* radio frekuensi dapat mengirim dan menerima paket data sesuai skematik dan tidak adanya kerusakan pada hardware tersebut, dan library yang digunakan adalah Mirf.

4.3 Kebutuhan Antarmuka Eksternal

Pada sub bab ini akan dijelaskan kebutuhan antarmuka eksternal seperti kebutuhan antarmuka pengguna, kebutuhan perangkat keras, dan kebutuhan perangkat lunak.

4.3.1 Kebutuhan Antarmuka Pengguna

Sistem ini akan melakukan pengiriman dan penerimaan data pada node menggunakan protokol *routing flooding* pada node *wireless sensor network* dengan menggunakan media komunikasi radio frekuensi. Proses pengiriman dan penerimaan data akan ditampilkan di serial monitor pada Arduino IDE.

4.3.2 Kebutuhan Perangkat Keras

Setiap node memiliki perangkat keras sebagai berikut:

- Mikrokontroller yang digunakan adalah Arduino Nano.
- NRF24L01 merupakan *transmitter* frekuensi radio yang dipasang di setiap node untuk melakukan pengiriman dan penerimaan informasi data.
- PC/Laptop digunakan untuk mengirim sketch Arduino dan menampilkan serial monitor.
- Jumper digunakan untuk menghubungkan perangkat Arduino Nano dan nRF24L01.
- Kabel USB digunakan untuk menyambungkan Arduino Nano ke PC/Laptop.

4.3.3 Kebutuhan Perangkat Lunak

- Arduino IDE digunakan untuk mengirim sketch ke dalam perangkat Arduino Nano dan menampilkan hasil serial monitor dari setiap node.
- Windows 8.1 merupakan sistem operasi yang digunakan untuk menjalankan program Arduino IDE.
- Mirf merupakan *library* yang digunakan untuk menjalankan algoritma *routing flooding*.

4.4 Kebutuhan Fungsional

Kebutuhan fungsional merupakan pernyataan layanan sistem yang harus disediakan dalam sebuah sistem. Ada beberapa kebutuhan fungsional dalam sistem tersebut antara lain:

4.4.1 Sistem dapat melakukan *routing* sesuai dengan *routing flooding*

Sistem harus dapat melakukan protokol *routing flooding* yaitu sistem mampu mengirimkan dan meneruskan data dengan melakukan pengiriman data ke semua alamat pada satu jaringan.

4.4.2 Sistem melakukan kegiatan pengiriman dan penerimaan paket data

Sistem harus dapat melakukan kegiatan pengiriman dan penerimaan paket data karena dalam protokol *routing flooding* terjadi proses pengiriman dan

penerimaan paket data dengan menggunakan modul *transceiver wireless* nRF24L01.

4.4.3 Sistem menampilkan hasil pengiriman dan penerimaan paket data

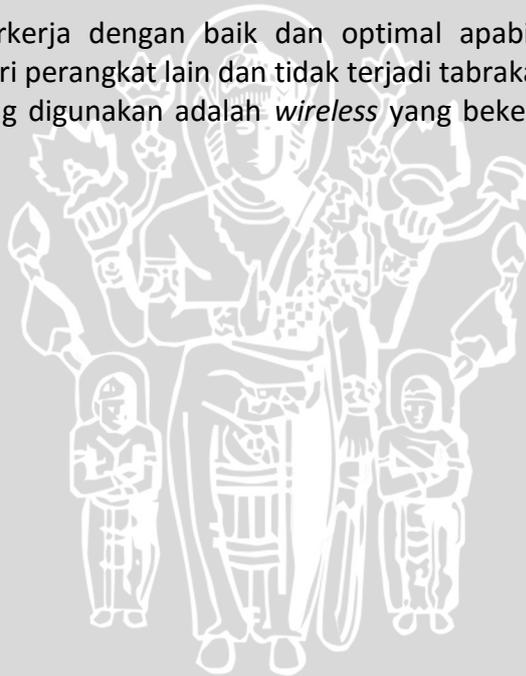
Sistem harus dapat menampilkan hasil pengiriman dan penerimaan paket data sesuai dengan keadaan sebenarnya. Menampilkan hasil pengiriman dan penerimaan paket data yang bertujuan agar dapat melihat jalannya sistem pada setiap node melalui serial monitor pada Arduino IDE.

4.5 Kebutuhan Non-Fungsional

Kebutuhan non-fungsional merupakan batasan layanan atau fungsi yang ditawarkan oleh sistem. Beberapa kebutuhan non-fungsional dalam sistem tersebut antara lain :

4.5.1 Kebutuhan Performansi

Sistem dapat bekerja dengan baik dan optimal apabila tidak terdapat gangguan frekuensi dari perangkat lain dan tidak terjadi tabrakan kanal data yang digunakan. Modul yang digunakan adalah *wireless* yang bekerja pada frekuensi 2.4GHz.



BAB 5 PERANCANGAN DAN IMPLEMENTASI

Pada bab ini akan menjelaskan tentang perancangan sistem pada software dan hardware, serta Implementasi software dan hardware yang dibutuhkan oleh sistem.

5.1 Perancangan Sistem

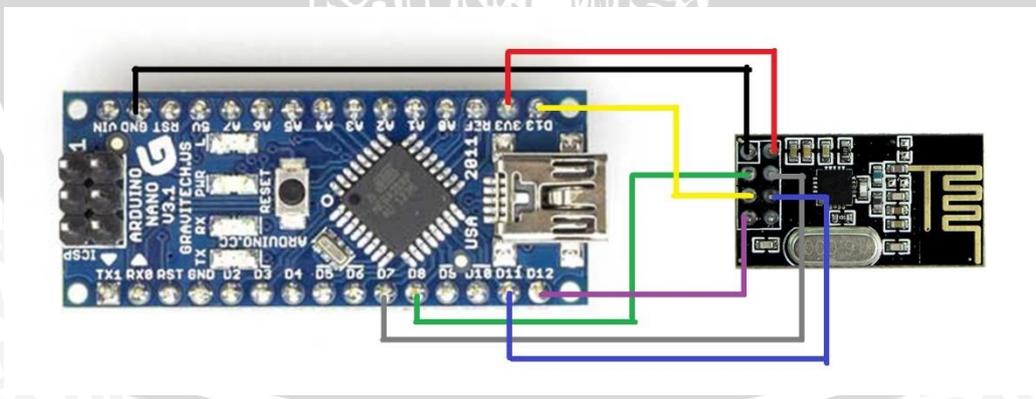
Pada Sub bab ini menjelaskan tentang perancangan sistem yang terdiri dari perancangan perangkat keras, perangkat lunak, serta perancangan skema protokol *routing flooding*.

5.1.1 Perancangan Perangkat Lunak

Perancangan perangkat lunak pada sistem dilakukan pada mikrokontroler di setiap node. Hal ini untuk memasukkan *source code* dari PC/Laptop ke dalam mikrokontroler yang nantinya dikomunikasikan melalui nRF24L01. *Source Code* menggunakan Bahasa C yang digunakan pada aplikasi Arduino IDE dan menggunakan library modul *Mirf*.

5.1.2 Perancangan Perangkat Keras

Perancangan perangkat keras pada sistem dilakukan di masing-masing node. Setiap node menggunakan 1 buah mikrokontroler (Arduino Nano) dan 1 buah radio frekuensi (nRF24L01). Mikrokontroler dan radio frekuensi dihubungkan sesuai dengan pin yang ada pada nRF24L01 yang selanjutnya dihubungkan ke PC/Laptop untuk diberi *source code*. Berikut adalah tabel pin pada Arduino Nano dan nRF24L01.



Gambar 5.1 Gambar skematik Arduino dan nRF24L01

Tabel 5.1 Komunikasi Pin Arduino Nano dan Pin nRF24L01

| NRF24L01 | Arduino Nano |
|----------|--------------|
| GND | GND |
| VDD | 3,3V |

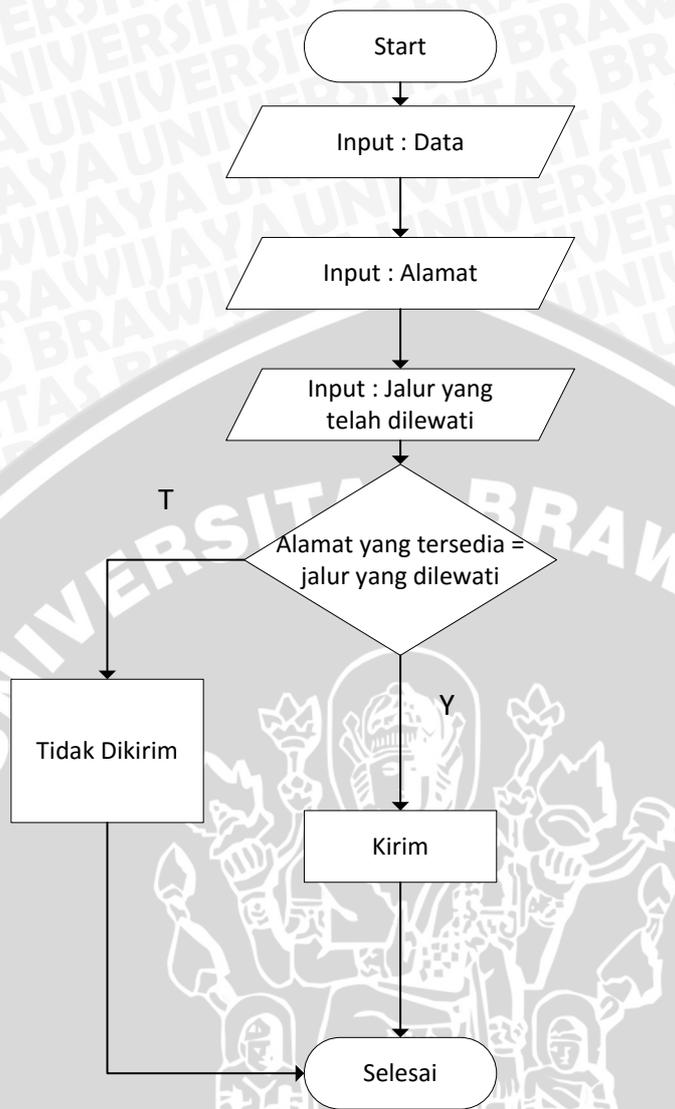
| | |
|-------|-----|
| CE/SS | D9 |
| CSN | D10 |
| SCK | D13 |
| MOSI | D11 |
| MISO | D12 |

5.1.3 Perancangan Skema Protokol Routing Flooding

Perancangan skema protokol *routing flooding* menggunakan beberapa tahapan, yaitu protokol *routing flooding*, penggunaan topologi, penentuan node tetangga, format data yang akan dikirim dan diterima.

a. Protokol Routing Flooding

Routing Flooding merupakan suatu protokol yang digunakan untuk pengiriman data dimana data tersebut akan di sebar dalam sebuah jaringan, yaitu sesuai dengan node sebelah (tetangga) yang nantinya meneruskan hingga menuju node terakhir. Pengiriman data bersifat *broadcast* sesuai dengan alamat tujuan (node tetangga) yang dimiliki oleh setiap node. Pada *routing flooding* data yang sudah dikirim tidak akan dikirimkan kembali ke node sebelumnya, hal ini dapat menyebabkan tabrakan dan duplikasi data, sehingga pada perancangannya sistem akan melakukan *broadcast* pada node tujuan dengan tetap memeriksa data yang diterima dari node lain apakah telah melewati node tersebut, jika data dari node lain belum melewati node tersebut maka data akan diberi tanda berupa nomor node tersebut kemudian diteruskan ke node tujuan (node tetangga), untuk pengiriman selanjutnya sistem akan memeriksa alamat tujuan apakah sama dengan alamat terakhir yang dilewati oleh data, jika alamat terakhir sama dengan alamat tujuan maka data tidak akan dikirimkan ke alamat tujuan tersebut, dan jika alamat terakhir pada data tidak sama dengan alamat tujuan maka data akan dikirimkan ke alamat tujuan tersebut. Hal ini digunakan untuk menghindari duplikasi data pada setiap node. Berikut adalah flowchart *routing flooding*.

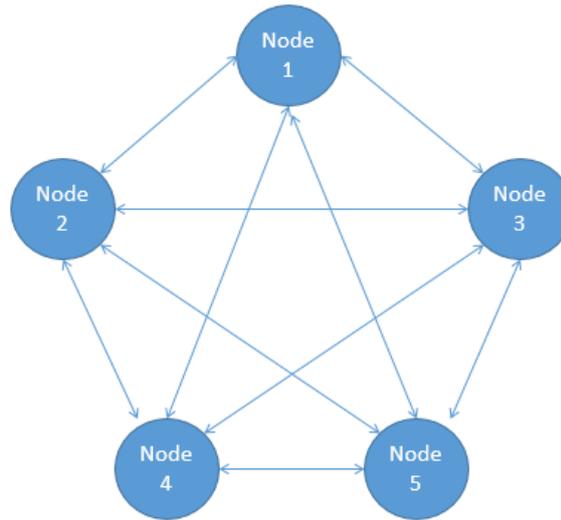


Gambar 5.2 Flowchart sistem routing flooding.

b. Topologi

Topologi yang digunakan pada routing flooding yaitu topologi berbentuk Mesh. Karena topologi ini dipilih agar data dapat berjalan melewati beberapa node. Untuk node sendiri akan dirancang sesuai dengan topologi seperti pada gambar dibawah ini.





Gambar 5.3 Topologi yang Digunakan untuk routing flooding.

c. Node Tetangga

Penentuan node tetangga di masing-masing node yaitu dengan meyertakan semua alamat yang ada pada semua jaringan di masing-masing node.

Tabel 5.2 Daftar Tabel Node Tetangga

| Node | Node Tetangga | | | | |
|------|---------------|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 |
| 1 | ✓ | ✓ | ✓ | ✓ | ✓ |
| 2 | ✓ | ✓ | ✓ | ✓ | ✓ |
| 3 | ✓ | ✓ | ✓ | ✓ | ✓ |
| 4 | ✓ | ✓ | ✓ | ✓ | ✓ |
| 5 | ✓ | ✓ | ✓ | ✓ | ✓ |

d. Format Data

Format data yang dikirimkan sebesar 13 byte. Setiap byte terdiri dari array yang berisi data dan alamat yang dilewati oleh data. Array akan terisi dari array 0 [0] yang berisi tentang ID Pengiriman. Array 1 [1] berisi data yang akan dikirim. Array ke 2 [2] berisi alamat tujuan pertama yang akan



dilewati oleh data setelah melewati alamat. Array ke 3 [3] berisi alamat kedua yang diisi setelah data melewati alamat pertama. Array ke 4 [4] akan berisi alamat ketiga setelah data melewati alamat kedua. Array ke 5[5] akan berisi alamat ke empat setelah data melewati alamat ketiga. Array ke 6[6] akan berisi alamat ke enam setelah data melewati alamat kelima. Jika array belum melewati alamat berikutnya maka array akan bernilai 0. Sehingga format data yang dikirimkan setelah data melewati keseluruhan node yaitu[ID pengiriman] [data] [alamat1] [alamat2] [alamat3] [alamat4] [alamat5].



Gambar 5.4 Struktur Data

5.2 Implementasi Sistem

Pada bab ini akan dilakukan Implementasi Sistem setelah perancangan perangkat keras, lunak, dan skema protokol *routing flooding* terpenuhi dan siap untuk saling dihubungkan untuk membentuk sistem. Implementasi Sistem meliputi spesifikasi perangkat keras, spesifikasi perangkat lunak, implementasi skema protokol *routing flooding*, dan batasan implementasi.

5.2.1 Spesifikasi Perangkat Keras

Pada spesifikasi perangkat keras yaitu perangkat keras yang akan digunakan di dalam membangun sistem. Perangkat keras yang digunakan yaitu Arduino Nano, radio frekuensi (RF) nRF24L01, PC/Laptop.

a. Arduino Nano

Arduino Nano digunakan sebagai kendali utama dari tiap-tiap node, yang akan mengolah data baik dari node lain maupun dari PC/Laptop. Berikut adalah spesifikasi dari Arduino Nano.

Tabel 5.3 Spesifikasi Arduino Nano

| | |
|---------------------------------|------------------------------|
| Microcontroller | Atmel ATmega168 or ATmega328 |
| Operating Voltage (logic level) | 5V |
| Input Voltage (recommended) | 7-12V |
| Input Voltage (limits) | 6-10V |

| | |
|------------------------|---|
| Digital I/O Pins | 14 (of which 6 provide PWM output) |
| Analog Input Pins | 8 |
| DC Current per I/O pin | 40 mA |
| Flash Memory | 16 KB (ATmega168) or 32 KB (ATmega328) of which 2 KB used by bootloader |
| SRAM | 1 KB (ATmega168) or 2KB (ATmega328) |
| EPROM | 512 bytes (ATmega168) or 1 KB (ATmega328) |
| Clock Speed | 16MHz |
| Dimensions | 0.73"x1.70" |
| Length | 45 mm |
| Weigth | 5 g |

b. NRF24L01

NRF24L01 merupakan radio frekuensi(RF) modul yang dapat bekerja pada frekuensi 2,4GHz. NRF24L01 digunakan untuk proses komunikasi node, yang akan mengirimkan dan yang menerima data dari node-node lain. Berikut adalah spesifikasi dari nRF24L01.

Tabel 5.4 Spesifikasi nRF24L01

| | |
|---|-----------|
| Minimum supply voltage | 1.9 V |
| Maximum output power | 0 dBm |
| Maximum data rate | 2000 kbps |
| Supply current in TX mode @ 0dBm output power | 11.3 mA |
| Supply current in RX mode @ 2000 kbps | 12.3 mA |
| Sensitivity @ 1000 kbps | -85 dBm |
| Supply current in Power Down mode | 900 nA |

c. PC/Laptop

PC/Laptop digunakan untuk memasukkan *source code* ke Arduino serta untuk menampilkan serial monitor dari program aplikasi Arduino IDE. Berikut ini adalah spesifikasi laptop yang digunakan.

Tabel 5.5 Spesifikasi PC/Laptop

| | |
|---------------------|--|
| Operating System | Windows 8.1 Pro 64-bit (6.3, Build 9600) |
| Language | English (Regional Setting: English) |
| System Manufacturer | ASUSTeK COMPUTER INC. |
| System Model | X455LJ |
| BIOS | X455LJ.202 |
| Processor | Intel® Core™ i3-5010 CPU @ 2.10GHz (4 CPUs), ~2.1GHz |
| Memory | 4096MB RAM |
| Page file | 3554MB used, 3466MB available |
| DirectX Version | DirectX11 |

5.2.2 Spesifikasi Perangkat Lunak

Spesifikasi perangkat lunak yang digunakan adalah Arduino IDE dan Windows 8.1. Sistem Operasi yang digunakan adalah Windows 8.1 Pro, untuk kestabilan program Arduino yang akan dipakai. Arduino IDE yang digunakan versi 1.68 dengan menggunakan Bahasa pemrograman C.

5.2.3 Batasan Implementasi

Pada pengimplementasian sistem ada beberapa hal yang dijadikan sebagai batasan dari sistem yaitu sebagai berikut :

1. Penentuan node tujuan diinisialisasikan di masing-masing node yaitu semua node pada satu jaringan diinisialisasikan di masing-masing node.
2. Implementasi dikatakan berjalan jika semua node sudah siap atau sudah dalam keadaan hidup.

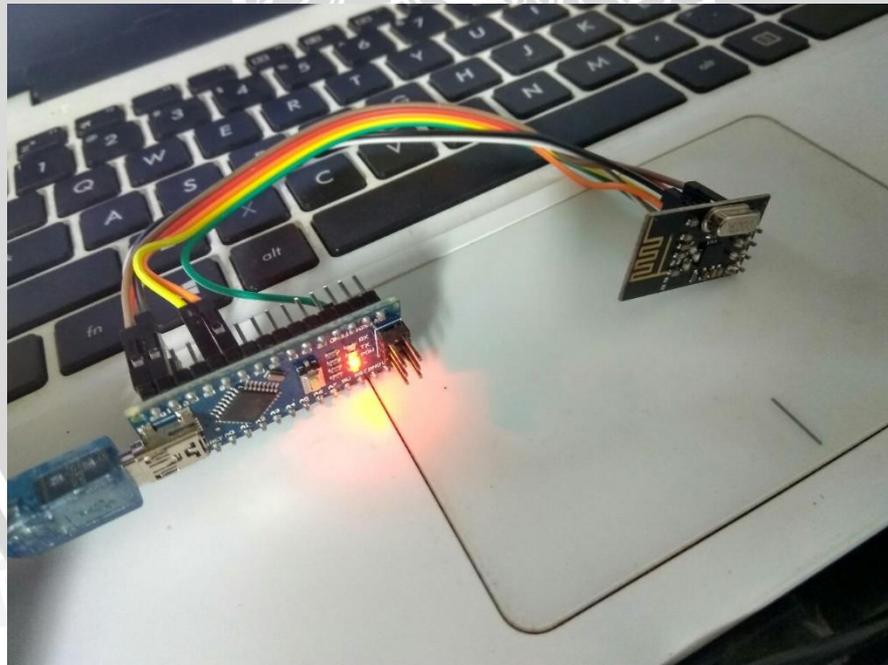
3. Pengiriman data berupa inputan yang dikirimkan dari node 1 (node pertama) melalui serial monitor.
4. Untuk penampilan data yang diterima di masing-masing node akan ditampilkan pada PC/Laptop yang sudah terhubung dengan masing-masing node.

5.2.4 Impelementasi Protokol *Routing Flooding*

Pada Implementasi skema protokol *routing flooding* akan dirancang setelah semua kebutuhan terpenuhi. Perancangan *routing flooding* disesuaikan dengan perangkat keras dan perangkat lunak yang akan dipakai.

5.2.4.1 Implementasi Node

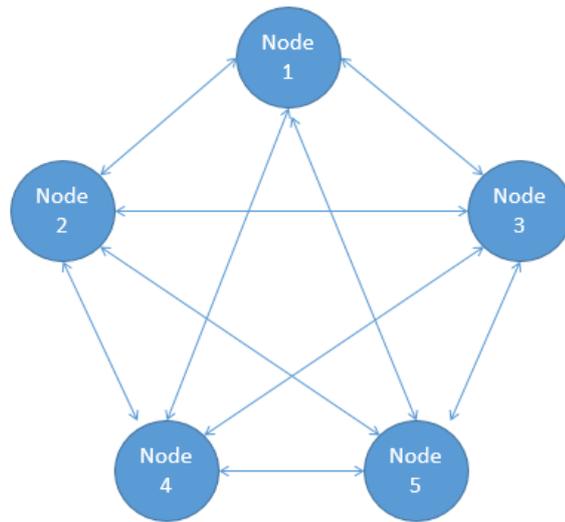
Pada implementasi node yang dibutuhkan adalah perangkat keras yang sesuai pada perancangan perangkat keras yaitu menggunakan Arduino Nano dan nRF24L01. Arduino Nano dan nRF24L01 dihubungkan sesuai dengan pin yang ada pada tabel perancangan perangkat keras. Setelah Arduino Nano dan nRF24L01 sudah terhubung dengan jumper, maka rangkaian tersebut dapat dihubungkan menggunakan USB untuk dihubungkan ke PC/Laptop untuk memasukkan *source code* ke dalam Arduino Nano.



Gambar 5.5 Implementasi Node dan tersambung ke PC/Laptop

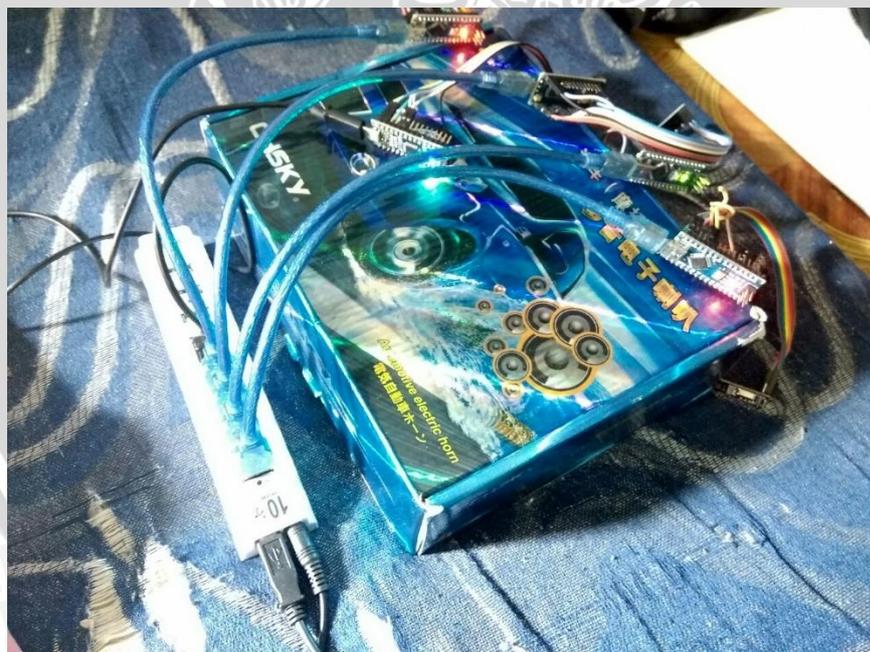
5.2.4.2 Implementasi Topologi

Implementasi topologi yaitu sesuai dengan perancangan topologi. Sesuai dengan perancangan topologi, dari sisi software disusun sesuai gambar dibawah ini.



Gambar 5.6 Topologi yang digunakan protokol *routing flooding*.

Dari sisi hardware, maka node akan diurut secara parallel tanpa harus membentuk seperti pada gambar di atas, hanya urutan pengirimannya disesuaikan dengan gambar di atas. Berikut gambar node sudah siap dijalankan.

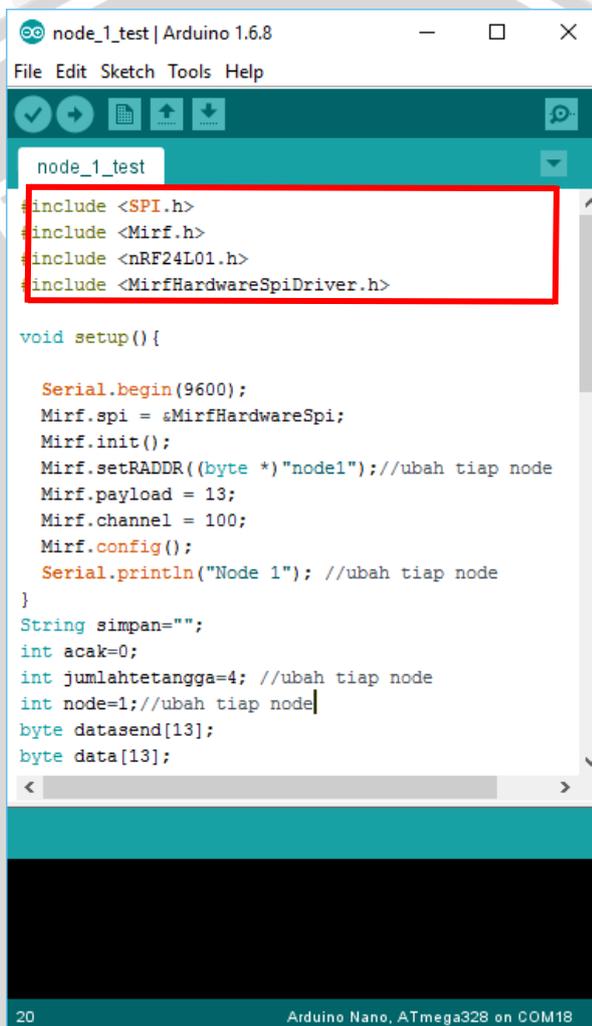


Gambar 5.7 Rangkaian semua node.

5.2.4.3 Implementasi *Routing Flooding*

Implementasi *routing flooding* menggunakan program Arduino IDE yang juga merupakan program untuk memasukkan *source code* ke dalam Arduino Nano.

Pada aplikasi Arduino IDE terdapat library yang digunakan yaitu SPI.h, Mirf.h, nRF24L01.h, MirfHardwareSpiDriver.h. SPI.h (*Serial Peripheral Interface*) merupakan sebuah protokol serial data yang sinkron yang digunakan oleh mikrokontroler untuk saling berkomunikasi baik dengan satu *device* atau dengan banyak *device*. NRF24L01.h merupakan fungsi yang digunakan untuk modul radio frekuensi (nRF24L01) agar dapat melakukan pengiriman dan penerimaan data. Sedangkan Mirf.h dan MirfHardwareSpiDriver.h merupakan fungsi untuk memanggil fungsi lain agar dapat melakukan pengiriman dan penerimaan data melalui radio frekuensi yang digunakan.



```
node_1_test | Arduino 1.6.8
File Edit Sketch Tools Help

node_1_test

#include <SPI.h>
#include <Mirf.h>
#include <nRF24L01.h>
#include <MirfHardwareSpiDriver.h>

void setup() {

  Serial.begin(9600);
  Mirf.spi = &MirfHardwareSpi;
  Mirf.init();
  Mirf.setRADDR((byte *)"node1");//ubah tiap node
  Mirf.payload = 13;
  Mirf.channel = 100;
  Mirf.config();
  Serial.println("Node 1"); //ubah tiap node
}

String simpan="";
int acak=0;
int jumlahtetangga=4; //ubah tiap node
int node=1;//ubah tiap node
byte datasend[13];
byte data[13];

20 Arduino Nano, ATmega328 on COM18
```

Gambar 5.8 Implelementasi sistem pada Program Arudino IDE.

1. Inisialisasi Node

Masing-masing node harus diinisialisasikan terlebih dahulu agar dapat mengirimkan data sesuai dengan tujuannya. Pada masing-masing node di deklarasikan nama node, frekuensi, jumlah node tetangga, dan juga besar paket data yang akan dikirimkan . Pada modul radio frekuensi bisa bekerja

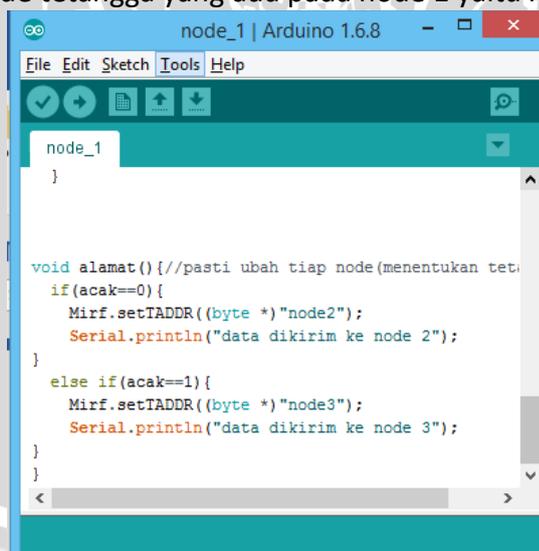
pada frekuensi yang sama maka penentuan frekuensi harus disamakan di masing-masing node.

```
void setup() {
  Serial.begin(9600);
  Mirf.spi = &MirfHardwareSpi;
  Mirf.init();
  Mirf.setRADDR((byte *)"node3");//ubah tiap node
  Mirf.payload = 13;
  Mirf.channel = 100;
  Mirf.config();
  Serial.println("Node 3"); //ubah tiap node
}
String simpan="";
int acak=0;
int jumlahtetangga=4; //ubah tiap node
int node=3;//ubah tiap node
byte datasend[13];
```

Gambar 5.9 Inisialisasi masing-masing node

2. Inisialisasi Node Tetangga

Inisialisasi node tetangga merupakan inisialisasi untuk alamat tujuan pengiriman data. Setiap node memiliki jumlah node tetangga yang sama, sehingga harus diinisialisasikan terlebih dahulu. Untuk pengiriman data dibuat skema bahwa data tidak akan kembali ke alamat sebelumnya, agar tidak terdapat penumpukan dan tabrakan data. Untuk contoh inisialisasi menggunakan node tetangga yang ada pada node 1 yaitu node 2,3,4,5.



```
node_1 | Arduino 1.6.8
File Edit Sketch Tools Help
node_1
}

void alamat()//pasti ubah tiap node(menentukan tet
if(acak==0){
  Mirf.setTADDR((byte *)"node2");
  Serial.println("data dikirim ke node 2");
}
else if(acak==1){
  Mirf.setTADDR((byte *)"node3");
  Serial.println("data dikirim ke node 3");
}
}
```

Gambar 5.10 Inisialisasi Node Tetangga

3. Format Paket Data

Untuk format paket data pada *routing flooding* menggunakan fungsi array, yaitu untuk menyimpan data dan alamat yang sudah dilewati. Pada array 0 diisi oleh ID Pengiriman, array 1 berisi data yaitu data yang dimasukkan dari inputan

pada node 1. Array 2 dan seterusnya diisi oleh alamat node yang telah dilewati oleh node. Saat penerimaan data, akan ditampilkan oleh masing-masing node yaitu ditunjukkan dengan gambar 1. Setiap node yang dilewati oleh data akan mengisi array sesuai dengan berurutan yaitu akan mengecek apakah dari array 0 sampai array 6 bernilai nol, jika nol maka node akan mengisi array yang bernilai nol tersebut ditunjukkan oleh gambar 2. Setelah mengisi alamat, data dapat diteruskan ke node selanjutnya.

```
int node=2;//ubah tiap node
byte datasend[13];
byte data[13];

void loop() {
  if(!Mirf.isSending() && Mirf.dataReady()){
    Mirf.getData(data);
    Serial.println("Data diterima");
    Serial.print(data[0]);
    Serial.print(data[1]);
    Serial.print(data[2]);
    Serial.print(data[3]);
    Serial.print(data[4]);
    Serial.print(data[5]);
    Serial.println(data[6]);
    for(int x=2;x<12;x++){//isi alamat saat sudah
      if (data[x]==0){
        data[x]=node;
        x=12;
      }
    }
    kirim();
    for(int x=0;x<12;x++){//isi alamat saat sudah
      data[x]=0;
    }
  }
}
```

Gambar 5.11 Format Paket Data

BAB 6 PENGUJIAN DAN ANALISA

Pada bab ini dilakukan pengujian dan analisa dari sistem dengan tujuan sesuai dengan implementasi yang sudah dibangun. Pengujian sistem dilakukan untuk mengetahui apakah kebutuhan fungsional dan non-fungsional sudah terpenuhi. Analisa pada sistem dilakukan untuk mendapatkan kesimpulan dari hasil pengujian dari sistem. Pengujian pada sistem antara pengujian pengiriman dan penerimaan data, pengujian paket data, dan pengujian *routing flooding*.

6.1 Pengujian Pengiriman dan Penerimaan Data

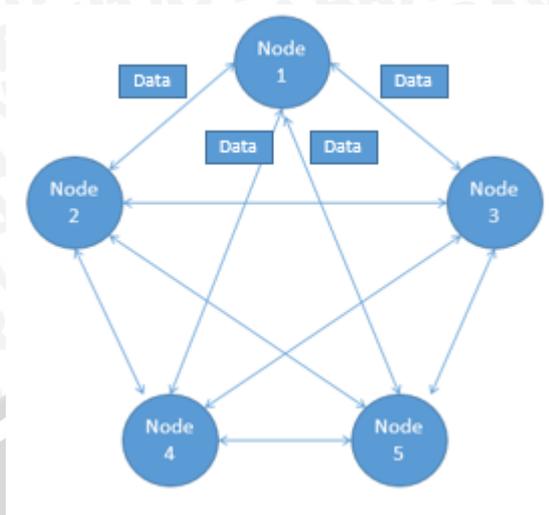
Pada pengujian pengiriman dan penerimaan data bertujuan agar masing-masing node bisa menerima dan meneruskan data. Setiap node akan menerima data dari node sebelumnya dan meneruskan data yang sudah diterima ke node setelahnya.

6.1.1 Pelaksanaan Pengujian

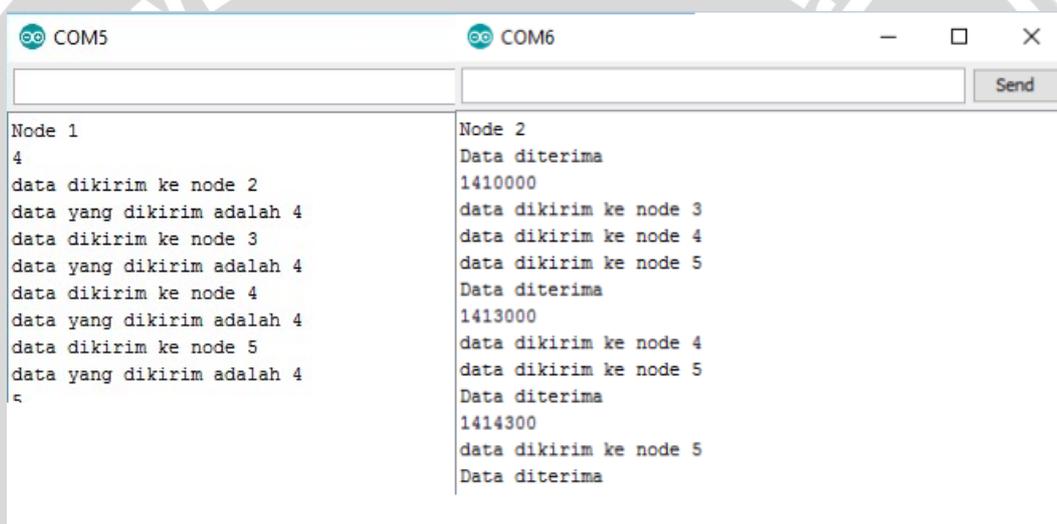
Pelaksanaan pengujian pengiriman dan penerimaan data dilakukan dengan menggunakan seluruh node yang dihubungkan ke PC/Laptop. Node pertama akan melakukan pengiriman pertama berupa data, kemudian pada node selanjutnya akan dilakukan pengecekan bahwa telah menerima data dari node sebelumnya dan akan meneruskan data ke node setelahnya.

6.1.2 Hasil Pengujian

Hasil yang di dapat untuk pengujian pengiriman dan penerimaan paket data, diperoleh dari serial monitor Arduino dari masing-masing node. Setiap node menerima dan mengirimkan data sesuai dengan routing alamat yang sudah ditentukan pada topologi.



Gambar 6.1 Jalur Routing data



Gambar 6.2 Screenshot pada node 1 saat mengirimkan data dan node 2 saat menerima data

6.1.3 Analisa Hasil Pengujian

Dari hasil analisa berdasarkan gambar node 1 dan node 2 setiap node sudah dapat menerima dan mengirimkan data sesuai dengan alamat tujuan. Untuk mengambil contoh diatas, data yang dikirimkan berupa data bilangan 4, yang dikirimkan menuju semua alamat. Pada node 2 melakukan penerimaan data, kemudian node membuka data tersebut yaitu berupa 1410000 yang merupakan 4 adalah data yang dikirim, angka 1 pertama merupakan ID pengiriman yaitu berupa pengiriman pertama sedangkan angka 1 setelah 4 merupakan alamat yang dilewati oleh data yaitu node 1 sedangkan 0 adalah alamat node 2, karena data baru sampai dan belum dikirim maka bernilai 0.

6.2 Pengujian Paket Data

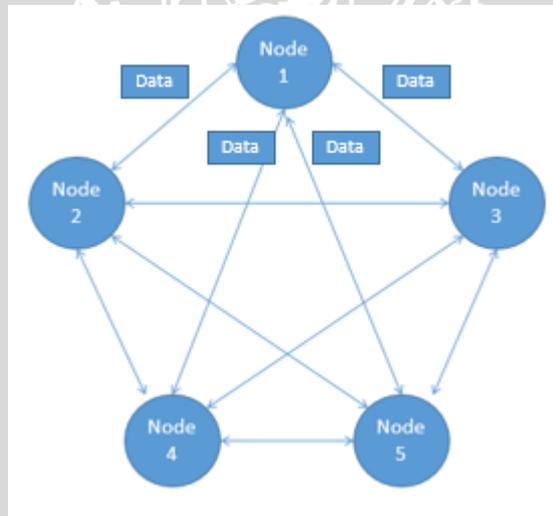
Pada pengujian paket data bertujuan untuk mendapatkan struktur paket data sesuai dengan perancangan paket data. ID pengiriman berada pada array 0, data berada pada array 1, dan array 2 hingga array 6 akan berisi alamat node yang sudah dilewati oleh data. Pada pengujian paket data, setiap node akan menampilkan data sesuai dengan data yang diterima.

6.2.1 Pelaksanaan Pengujian

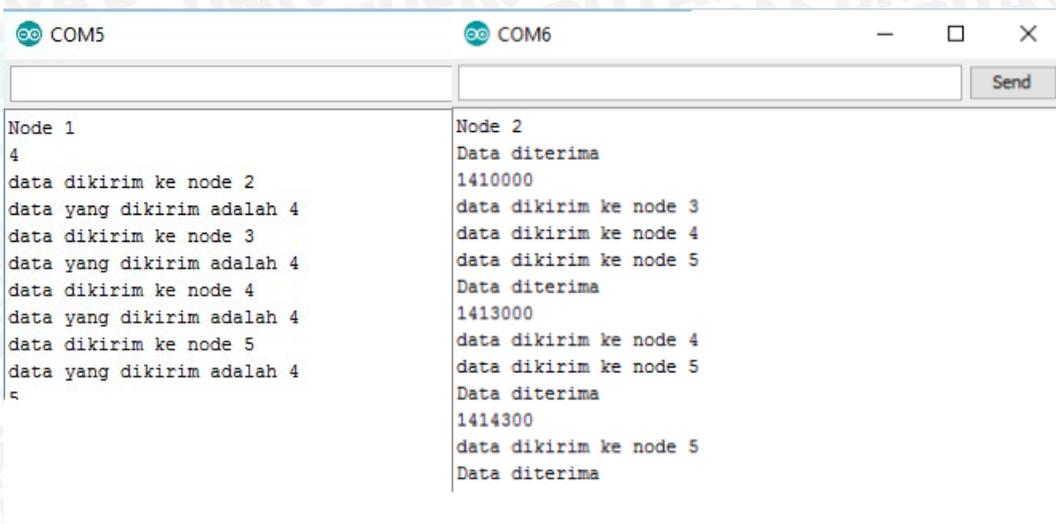
Pelaksanaan pengujian paket data dilakukan di salah satu node setelah node 1 melakukan pengiriman data. Setiap node akan dilihat data yang sudah diterima sebelumnya dan node akan mengisi alamat selanjutnya setelah data sampai pada node tujuan.

6.2.2 Hasil Pengujian

Hasil yang di dapat pada pengujian paket data yaitu setiap node dapat memberikan alamat yang sudah dilewati oleh data. Berikut adalah hasil yang di dapat.



Gambar 6.3 Jalur routing pengiriman data



Gambar 6.4 Screenshoot setiap node dapat memberikan alamat setelah dilewati oleh data.

6.2.3 Analisa Hasil Pengujian

Dari hasil pengujian yang dilakukan untuk pengujian paket data, setiap node memberikan tanda bahwa data telah melewati node tersebut dengan cara menambahkan alamat tersebut ke array yang telah disediakan setelah array 1, karena array 1 berisi data yang dikirimkan. Data berupa inputan yang berasal dari node 1 yang dimasukkan ke array 1, kemudian array 2 akan diisi oleh node 1 berupa nilai 1 sebelum paket dikirimkan. Saat paket dikirimkan dan diterima oleh node 2 maka node 2 akan membaca array 2 dan array 3, kemudian sebelum melakukan pengiriman ke node selanjutnya, maka node 2 akan mengisi node ke 3 yaitu berupa alamat node yang ke 2 begitu juga hingga sampai di node terakhir.

6.3 Pengujian Routing Flooding

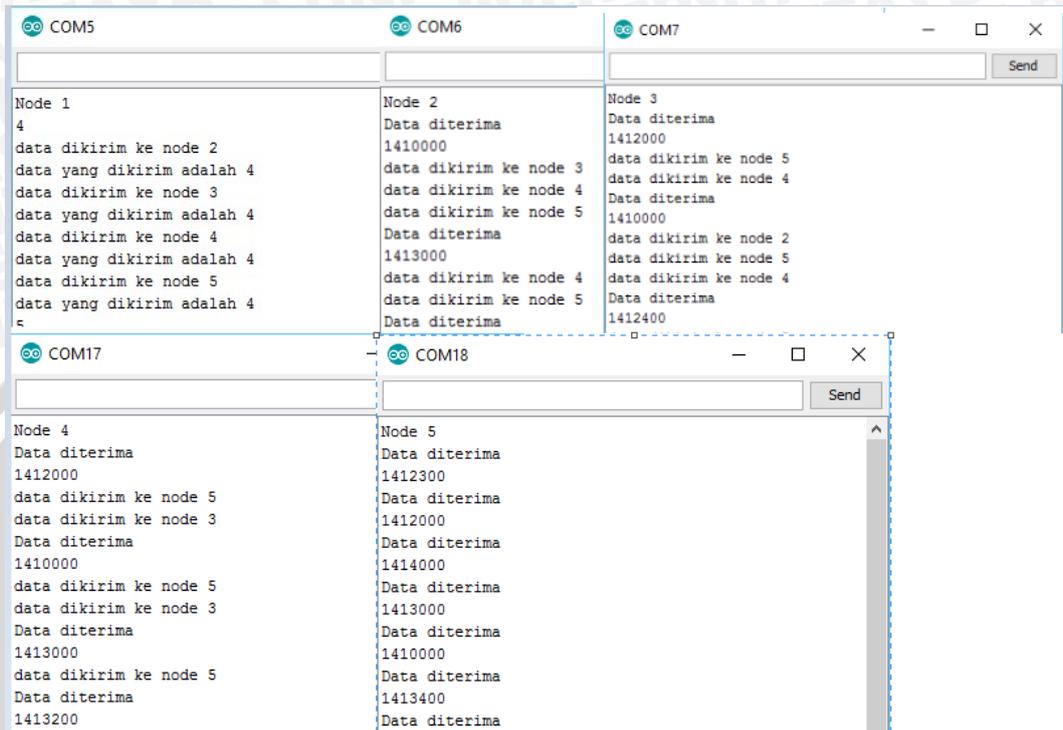
Pada pengujian *routing flooding* bertujuan agar sistem bekerja sesuai dengan kebutuhan sistem, yaitu sesuai dengan implementasi topologi, node, dan struktur paket data. Pada pengujian *routing flooding* diharapkan semua implementasi bisa dijalankan secara bersama dan mendapatkan hasil sesuai dengan implementasi yang sudah dirancang.

6.3.1 Pelaksanaan Pengujian

Pelaksanaan pengujian *routing flooding* dilakukan setelah pengiriman dan penerimaan data, serta pengujian paket data telah bisa dijalankan.

6.3.2 Hasil Pengujian

Hasil yang di dapat dari pengujian routing flooding adalah dari serial monitor Arduino.



Gambar 6.5 Screenshot hasil pengujian routing flooding.

6.3.3 Analisa Hasil Pengujian

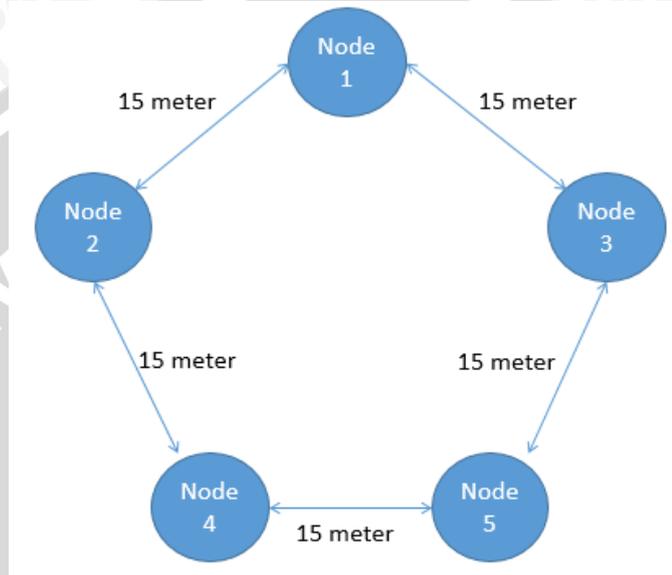
Dari hasil pengujian *routing flooding* setiap node sudah ditentukan *routing* paket masing-masing yaitu dengan memberikan semua alamat di masing-masing node. Setiap node akan memberikan tanda bahwa data telah melewati node tersebut. Untuk data sendiri tidak akan dikirimkan kembali ke alamat sebelumnya karena *routing flooding* pada sistem ini bertujuan agar tidak terjadi tabrakan data, pada proses pengiriman setiap data diberi nilai secara sequensial agar tidak terjadi duplikasi jika data yang dikirimkan sama.

6.4 Pengujian Multihop

Pada pengujian *multihop* bertujuan untuk mengetahui kinerja node router pada protokol *routing flooding*. Pada pengujian ini diharapkan node dapat mengirimkan data hingga tujuan yaitu dengan mengirimkan data melalui node-node yang lain.

6.4.1 Pelaksanaan pengujian

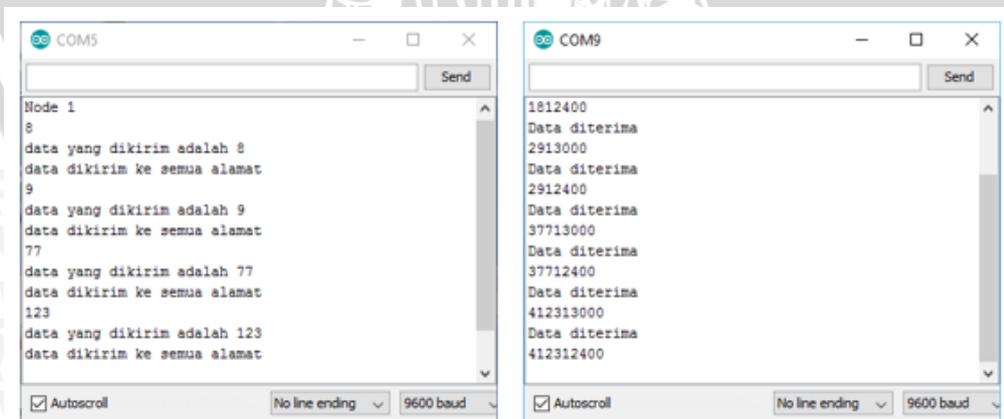
Pelaksanaan pengujian *multihop* dilakukan dengan tetap menggunakan rancangan topologi, hanya saja setiap node diberi jarak 15 meter, hal ini dilakukan untuk menguji node sejauh mana dapat mengirimkan data. Pada pengujian akan dilihat data dari node 1 dan node 5. Pada node 5 akan dilihat bahwa data telah melewati beberapa node.



Gambar 6.6 Posisi node diberikan jarak 15 meter.

6.4.2 Hasil Pengujian

Hasil yang di dapat dari pengujian *multihop* pada serial monitor.



Gambar 6.7 Hasil pengujian *multihop* pada node 1 (kiri) dan node 5 (kanan)

6.4.3 Analisa Hasil Pengujian

Dari hasil pengujian *multihop* didapatkan hasil yaitu pada node 5. Pada node 5 didapatkan 2 data yang sama namun dengan alamat yang berbeda, hal ini didapatkan berdasarkan jarak antar node yang sudah ditentukan yaitu 15 meter. Pada node 1 bertugas untuk mengirimkan data ke seluruh alamat, untuk node 5 menerima data dari semua alamat. Untuk salah satu contoh hasil yang didapat adalah 37713000 dan 3771240. Pada data pertama, node pertama mengirimkan ke semua alamat dan alamat yang menerima yaitu alamat node 3 kemudian node 3 meneruskan data ke node 5. Pada data kedua, node pertama mengirimkan data ke node 2 lalu node 2 meneruskan data ke node 4 hingga mencapai node 5.



BAB 7 PENUTUP

7.1 Kesimpulan

Berdasarkan perancangan, implementasi, serta pengujian yang dilakukan pada sistem dapat diambil kesimpulan:

1. Untuk mengimplementasikan *routing flooding* pada *wireless sensor network* menggunakan media komunikasi radio frekuensi yaitu membutuhkan beberapa hal yaitu: penentuan node tetangga di masing-masing node, penentuan paket data yang akan dikirimkan, serta protokol *routing flooding*.
2. Mekanisme komunikasi data antar node yaitu setiap node akan mengirimkan data sesuai dengan node tetangga yang sudah dideklarasikan di masing-masing node. Setiap node tidak akan mengirimkan data kembali ke alamat sebelumnya agar tidak terjadi tabrakan dan penumpukan data pada node sebelumnya.
3. Dari hasil pengujian *routing flooding* semua node dapat mengirimkan data ke seluruh alamat node tetangga, dan node dapat melakukan pengiriman dan penerimaan data dengan tetap memberikan alamat node yang sudah dilewati oleh data. Sehingga kinerja dari *routing flooding* bisa dikatakan baik karena sedikit mengurangi beban pada jaringan, yaitu data tidak tetap berada pada satu jaringan dan data tidak dikirimkan kembali ke alamat sebelumnya.

7.2 Saran

Saran yang dapat diberikan untuk pengembangan dari implementasi *protokol routing flooding* menggunakan media komunikasi radio frekuensi:

1. Untuk pengembangan selanjutnya, mungkin bisa ditambahkan delay untuk penghitungan kecepatan data yang dapat dikirimkan dari masing-masing node.
2. Untuk pengembangan selanjutnya bisa ditambahkan data sensor untuk pengembangan *routing flooding* lebih lanjut.

DAFTAR PUSTAKA

- Basma M., Sherine M., Hussein S., & Mohammed M.. 2011. *An Optimized Energy-aware Routing Protocol for Wireless Sensor Network*. Tersedia di www.elsevier.com/locate/eij. [Diakses 20 September 2015].
- Sudeep T., Neeraj K., & Joel J.P.C.R . 2015. *A systematic review on heterogeneous routing protocol for wireless sensor network*. Tersedia di www.elsevier.com/locate/jnca. [Diakses 20 September 2015].
- Sheikh F., & Xinrong L. 2014. *Wireless Sensor Network System Design using Raspberry Pi and Arduino for Enviromental Monitoring Application*. Tersedia di www.sciencedirect.com. [Diakses 20 September 2015].
- Arduino, 2015. Product Arduino. [online] Tersedia di: < <http://www.arduino.cc/en/Main/Products>> [Diakses 20 September 2015]
- Kazem S., Daniel M., & Taieb Z. 2007. *WIRELESS SENSOR NETWORK Technology, Protocols, and Applications*. [E-book]. Wiley. [Diakses 20 September 2015].
- NRF24L01, 2015. Nrf24L01-2.4GHz-HowTo [online] Tersedia di : <<https://arduino-info.wikispaces.com/Nrf24L01-2.4GHz-HowTo>> . [Diakses 25 September 2015]
- Ojha T., Misra S. & Raghuvanshi NS., 2015. *Wireless sensor networks for agriculture: The state-of-the-art in practice and future challenges*. Tersedia di <www.elsevier.com/locate/compag>. [Diakses 20 September 2015].
- Norouzi, A. & Zaim, A. H., 2012. An Integrative Comparison of Energy Efficient Routing Protocols in Wireless Sensor Network. *Wireless Sensor Network*, Volume 4, p. 75. Tersedia di < <http://www.scirp.org/journal/PaperInformation.aspx?PaperID=18310> > [Diakses 16 Juni 2016]
- Sharma, K., Neha, M. & Priyanka, R., 2014. Performance Analysis of Flooding and SPIN in Wireless Sensor Networks. *International Journal of Future Generation Communication and Networking* , Volume 7, p. 36. tersedia di < http://www.sersc.org/journals/IJFGCN/vol7_no3/3.pdf > [Diakses 16 Juni 2016]
- Stankovic, J. A., 2006. Wireless Sensor Networks. *Wireless Sensor Networks*, 19 June.p. 20. tersedia di < <https://www.cs.virginia.edu/~stankovic/psfiles/wsn.pdf>> [Diakses 17 Juni 2016]
- Muñoz, J., González, R., Otero, A. & Gazca, L., 2015. *A flooding routing algorithm for a wireless sensor network for seismic events*. Mexico, IEEE. tersedia di <: <https://www.researchgate.net/publication/284030363> > [diakses 18 Juni 2016]

Zheng, J. & Jamalipour, A., 2009. *Wireless Sensor Networks A Networking Perspective*.
s.l.:s.n. tersedia di <<https://www.scribd.com/book/145545064/Wireless-Sensor-Networks-A-Networking-Perspective>> [diakses 18 Juni 2016]

