

# **IMPLEMENTASI PROTOKOL ROUTING SPIN PADA WIRELESS SENSOR NETWORK MENGGUNAKAN MEDIA KOMUNIKASI RF**

**SKRIPSI**

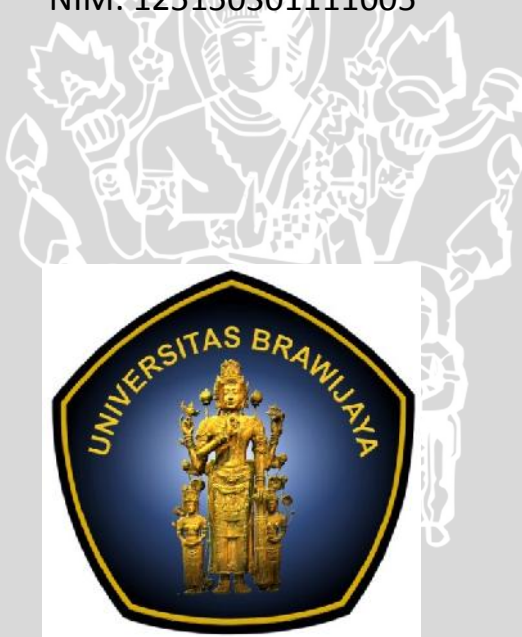
**KEMINATAN TEKNIK KOMPUTER**

Untuk memenuhi sebagian persyaratan  
memperoleh gelar Sarjana Komputer

Disusun oleh:

Ike Mia Wulandari S.

NIM: 125150301111005



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

## PENGESAHAN

IMPLEMENTASI PROTOKOL ROUTING SPIN PADA WIRELESS SENSOR NETWORK  
MENGUNAKAN MEDIA KOMUNIKASI RF

SKRIPSI

KEMINATAN TEKNIK KOMPUTER

Diajukan untuk memenuhi sebagian persyaratan  
memperoleh gelar Sarjana Komputer

Disusun Oleh :

Ike Mia Wulandari S.

NIM: 125150301111005

Skripsi ini telah diuji dan dinyatakan lulus pada  
30 Juni 2016

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

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

NIP: 19820809 201212 1 004

Aswin Suharsono, S.T, M.T.

NIK: 201102 840919 1 001

Mengetahui

Pjs Ketua Program Studi Teknik Informatika

Issa Arwani, S.Kom, M.Sc

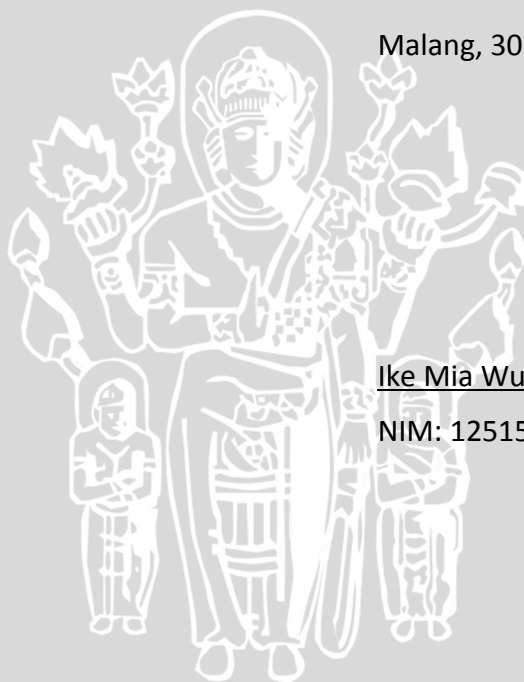
NIP: 19830922 201212 1 003

## 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, 30 Juni 2016



Ike Mia Wulandari S.

NIM: 125150301111005

## KATA PENGANTAR

Puji syukur kehadiran Tuhan YME yang telah melimpahkan rahmat, taufik serta hidayah-Nya sehingga laporan skripsi yang berjudul “Implementasi Protokol Routing SPIN pada Wireless Sensor Network Menggunakan Media Komunikasi RF” ini dapat terselesaikan.

Penulis menyadari bahwa skripsi ini tidak akan berhasil tanpa bantuan dari beberapa pihak. Oleh karena itu, penulis ingin menyampaikan rasa hormat dan terima kasih kepada:

1. Bapak Sabriansyah Rizqika Akbar, S.T, M.Eng. dan Bapak Aswin Suharsono, S.T, M.T. selaku dosen pembimbing skripsi penulis yang dengan sabar membimbing dan mengarahkan penulis sehingga dapat menyelesaikan skripsi ini.
2. Bapak Issa Arwani, S.Kom, M.Sc selaku ketua Program Studi Teknik Informatika.
3. Kedua Orang Tua dan seluruh keluarga besar atas segala nasehat, kasih sayang dan kesabaran dalam membesarkan dan mendidik penulis, serta yang senantiasa tiada henti-hentinya memberikan doa dan semangat demi terselesaikannya skripsi ini.
4. Seluruh sahabat saya Lilis Suryani, M. Rizky Firman Habibie, Ulfa Kurniawati, Ardy Novian Erwanda, dan Panji Putera terimakasih untuk kebersamaannya selama ini yang selalu mendukung penulis dalam menyelesaikan skripsi.
5. Singgih Adisaputra teman dekat penulis terimakasih atas perhatiannya selama ini yang selalu mendukung, menemani dan membantu sampai terselesaikannya skripsi ini.
6. Sahabat kos 41A Kertosentono Dinar Ayu K.P, Nur Fitria Rohmah, Putri Dwi Rachmawati, Herlin Kiswanti serta Ibu Hanifah dan Bapak Gapung selaku ibu dan bapak kos penulis, terimakasih atas kebersamaannya selama ini atas do’a dan dukungan yang tiada henti-hentinya.
7. Seluruh civitas akademika Fakultas Ilmu Komputer Universitas Brawijaya yang telah banyak memberi bantuan dan dukungan selama penulis menempuh studi Informatika di Universitas Brawijaya dan selama penyelesaian skripsi ini.

Penulis menyadari bahwa dalam penyusunan skripsi ini masih banyak kekurangan, sehingga saran dan kritik yang membangun sangat penulis harapkan. Akhir kata penulis berharap skripsi ini dapat membawa manfaat bagi semua pihak yang menggunakannya.

Malang, 30 Juni 2016

Penulis

125150301111005@mail.ub.ac.id

## ABSTRAK

Pada masa modern saat ini, teknologi dan komunikasi di bidang internet telah berkembang secara signifikan dan menjadi kebutuhan sehari – hari, selain itu pertukaran dan penyebaran data secara nirkabel atau wireless merupakan suatu media yang saat ini paling banyak digunakan dalam jaringan komunikasi. Salah satu penerapan teknologi secara *wireless* adalah sistem monitoring, yaitu dengan melakukan pengiriman data antar node WSN yang dapat diterapkan dengan berbagai macam protokol *routing*.

Algoritma SPIN (*Sensor Protocol for Information via Negotiation*) merupakan salah satu protokol *routing* yang bersifat *data-centric* (melakukan negosiasi sebelum proses transmisi data dilakukan). Ada tiga jenis proses pengiriman data pada *routing* SPIN, yaitu ADV, REQ dan DATA. Penerapan protokol routing SPIN pada WSN ini merupakan salah satu cara untuk mengoptimalkan kinerja jaringan sensor, karena penggunaan energi terbanyak pada tiap node terjadi saat node tersebut bertukar informasi dalam jaringan. Node sensor akan melakukan negosiasi dengan satu sama lain sebelum proses transmisi data. Hasil implementasi menunjukkan bahwa protokol routing SPIN cocok diterapkan pada WSN, setelah dilakukan 10 kali pengiriman data yang dikirimkan berhasil dikirim dan sesuai dengan alamat tujuan, namun semakin banyak node akan mempengaruhi kegagalan proses pengiriman data. Sedangkan saat dilakukan 20 kali pengiriman didapatkan delay yang cenderung stabil.

Kata kunci: Routing, WSN, SPIN, ADV, REQ, DATA

## ABSTRACT

*In today's era, internet communication technology has evolved significantly and become the daily needs, moreover the reciprocity and diffusion of wireless data is currently the most widely used in communication networks. One application of wireless technology is the monitoring system by transferring data between WSN nodes that can be applied to a range of routing protocols.*

*Algorithm SPIN (Sensor Protocol for Information via Negotiation) is a routing protocol that is data-centric (negotiate before the data transmission process is done). There are three types of data transmission on the routing process SPIN, namely ADV, REQ and DATA. The implementation of routing protocols in WSN SPIN This is one way to optimize the performance of sensor networks, because of the use of the most energy on each node occurs when the nodes in the network exchange information. The sensor nodes will negotiate with each other prior to the data transmission process. The implementation results showed that SPIN routing protocol is compatible to the WSN, after 10 times the data transmission is sent successfully delivered and in accordance with the destination, but the more nodes will affect the data transmission process failure. Meanwhile, when administered 20 times the delivery delay cenderug obtained stable.*

**Keywords:** Routing, WSN, SPIN, ADV, REQ, DATA

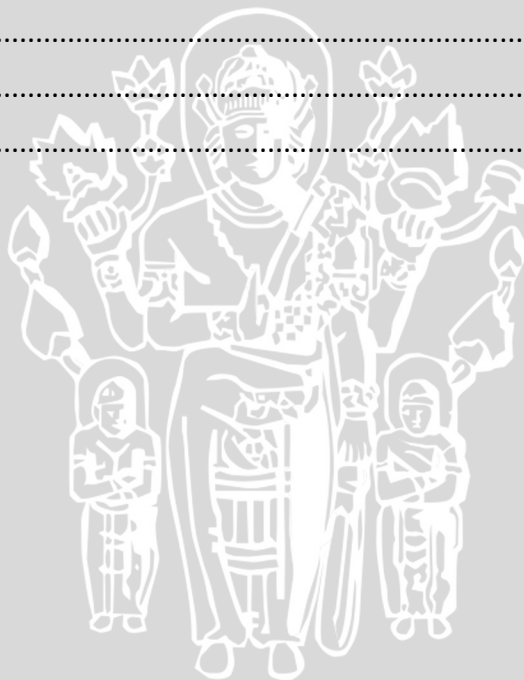


## DAFTAR ISI

|                                               |      |
|-----------------------------------------------|------|
| PENGESAHAN .....                              | ii   |
| PERNYATAAN ORISINALITAS .....                 | iii  |
| KATA PENGANTAR.....                           | iv   |
| ABSTRAK.....                                  | v    |
| ABSTRACT.....                                 | vi   |
| DAFTAR TABEL.....                             | x    |
| DAFTAR GAMBAR.....                            | xi   |
| DAFTAR LAMPIRAN .....                         | xiii |
| BAB 1 PENDAHULUAN.....                        | 1    |
| 1.1 Latar belakang.....                       | 1    |
| 1.2 Rumusan masalah.....                      | 2    |
| 1.3 Tujuan .....                              | 2    |
| 1.4 Manfaat .....                             | 2    |
| 1.4.1 Bagi Penulis .....                      | 2    |
| 1.4.2 Bagi Pengguna/ Pembaca .....            | 2    |
| 1.5 Batasan Masalah.....                      | 2    |
| 1.6 Sistematika pembahasan.....               | 3    |
| BAB 2 LANDASAN KEPUSTAKAAN .....              | 4    |
| 2.1 Wireless Sensor Network (WSN) .....       | 4    |
| 2.2 Radio Frequency .....                     | 5    |
| 2.3 Arduino Nano.....                         | 6    |
| 2.4 Nrf24L01 .....                            | 7    |
| 2.5 SPIN .....                                | 9    |
| BAB 3 METODOLOGI .....                        | 11   |
| 3.1 Alur Metode Penelitian.....               | 11   |
| 3.2 Studi Literatur .....                     | 12   |
| 3.3 Analisa Kebutuhan Sistem .....            | 12   |
| 3.4 Perancangan Sistem dan Diagram Blok ..... | 13   |
| 3.5 Implementasi .....                        | 14   |

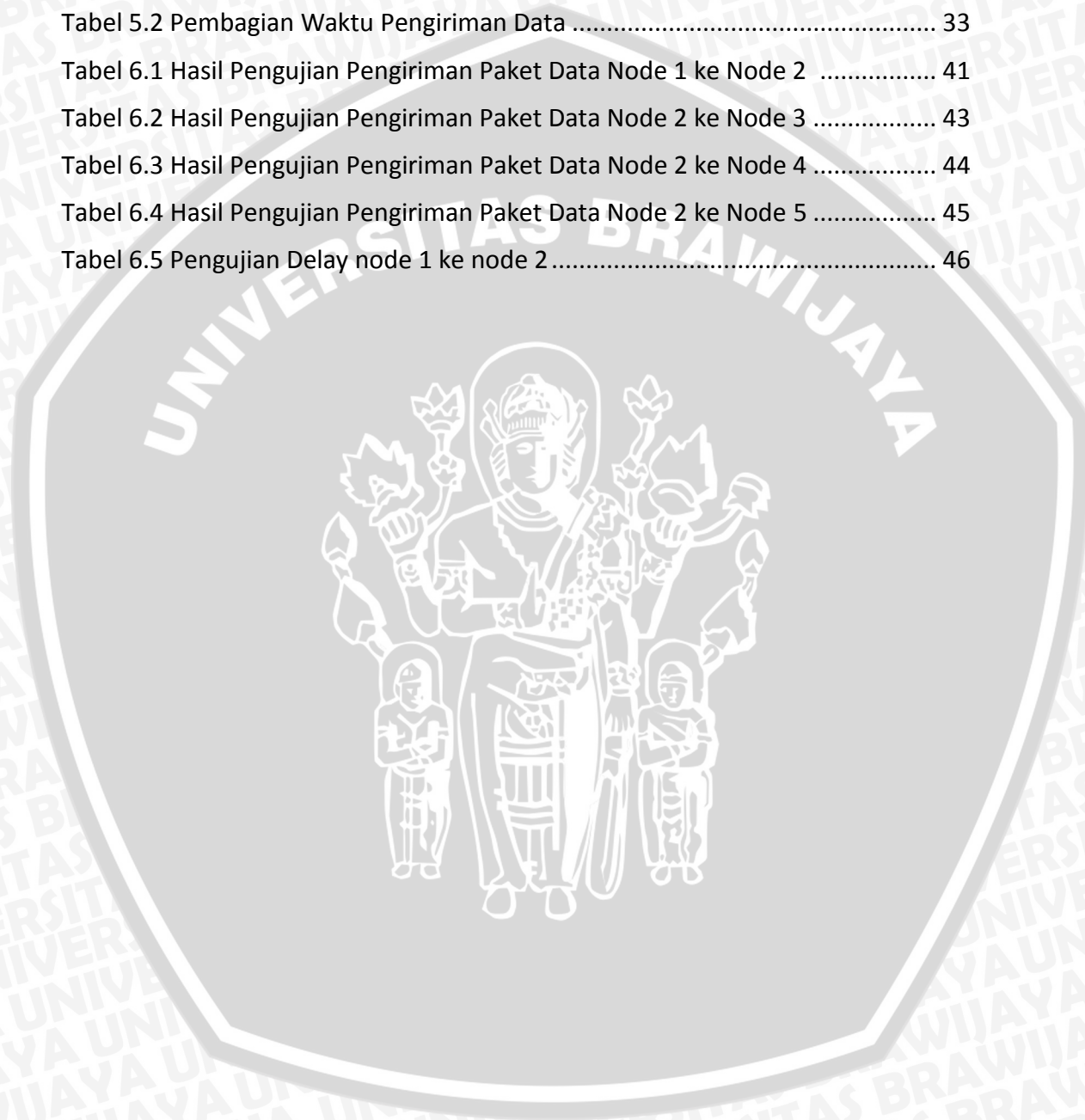
|                                                       |           |
|-------------------------------------------------------|-----------|
| 3.5.1 Implementasi Sensor Node .....                  | 14        |
| 3.6 Pengujian dan Analisis .....                      | 15        |
| 3.7 Kesimpulan dan Saran .....                        | 15        |
| <b>BAB 4 REKAYASA KEBUTUHAN .....</b>                 | <b>17</b> |
| 4.1 Pendahuluan .....                                 | 17        |
| 4.1.1 Tujuan .....                                    | 17        |
| 4.1.2 Ruang Lingkup .....                             | 17        |
| 4.1.3 Istilah .....                                   | 17        |
| 4.1.4 Referensi .....                                 | 18        |
| 4.1.5 Sistematika .....                               | 18        |
| 4.2 Deskripsi Umum .....                              | 18        |
| 4.2.1 Perspektif Sistem .....                         | 18        |
| 4.2.2 Kegunaan .....                                  | 19        |
| 4.2.3 Karakteristik Pengguna .....                    | 19        |
| 4.2.4 Lingkungan Operasi Sistem .....                 | 19        |
| 4.2.5 Batasan Sistem .....                            | 19        |
| 4.2.6 Asumsi dan Ketergantungan Sistem .....          | 19        |
| 4.3 Spesifikasi Kebutuhan .....                       | 19        |
| 4.3.1 Kebutuhan Pengguna .....                        | 20        |
| 4.3.2 Kebutuhan Fungsional .....                      | 20        |
| 4.3.3 Kebutuhan Komunikasi .....                      | 21        |
| 4.3.4 Kebutuhan Sistem .....                          | 21        |
| 4.3.5 Kebutuhan Perangkat Keras .....                 | 22        |
| 4.3.6 Kebutuhan Perangkat Lunak .....                 | 22        |
| <b>BAB 5 PERANCANGAN DAN IMPLEMENTASI .....</b>       | <b>24</b> |
| 5.1 Perancangan Sistem .....                          | 24        |
| 5.1.1 Perancangan Perangkat Keras SPIN pada WSN ..... | 24        |
| 5.1.2 Perancangan Perangkat Lunak SPIN pada WSN ..... | 26        |
| 5.1.3 Perancangan Topologi SPIN .....                 | 29        |
| 5.2 Implementasi .....                                | 30        |
| 5.2.1 Lingkungan Implementasi .....                   | 30        |
| 5.2.2 Batasan Implementasi .....                      | 31        |

|                                                              |           |
|--------------------------------------------------------------|-----------|
| 5.2.3 Implementasi Perangkat Keras SPIN pada WSN .....       | 31        |
| 5.2.4 Implementasi Perangkat Lunak SPIN pada WSN .....       | 31        |
| 5.2.5 Implementasi Komunikasi Data antar Node .....          | 32        |
| <b>BAB 6 PENGUJIAN DAN HASIL .....</b>                       | <b>39</b> |
| 6.1 Pengujian Node Sensor ke PC (Komputer) .....             | 39        |
| 6.2 Pengujian Fungsional .....                               | 41        |
| 6.2.1 Pengujian Pengiriman Data Node 1 ke node 2 .....       | 41        |
| 6.2.2 Pengujian Pengiriman data node 2 ke node 3, 4, 5 ..... | 42        |
| 6.3 Pengujian Delay .....                                    | 46        |
| <b>BAB 7 PENUTUP .....</b>                                   | <b>49</b> |
| 7.1 Kesimpulan .....                                         | 49        |
| 7.2 Saran .....                                              | 49        |
| <b>DAFTAR PUSTAKA.....</b>                                   | <b>50</b> |
| <b>LAMPIRAN .....</b>                                        | <b>52</b> |



## DAFTAR TABEL

|                                                                        |    |
|------------------------------------------------------------------------|----|
| Tabel 4.1 Tabel Daftar Istilah .....                                   | 17 |
| Tabel 5.1 Pin sambungan nRF24L01 pada Arduino nano .....               | 25 |
| Tabel 5.2 Pembagian Waktu Pengiriman Data .....                        | 33 |
| Tabel 6.1 Hasil Pengujian Pengiriman Paket Data Node 1 ke Node 2 ..... | 41 |
| Tabel 6.2 Hasil Pengujian Pengiriman Paket Data Node 2 ke Node 3 ..... | 43 |
| Tabel 6.3 Hasil Pengujian Pengiriman Paket Data Node 2 ke Node 4 ..... | 44 |
| Tabel 6.4 Hasil Pengujian Pengiriman Paket Data Node 2 ke Node 5 ..... | 45 |
| Tabel 6.5 Pengujian Delay node 1 ke node 2 .....                       | 46 |



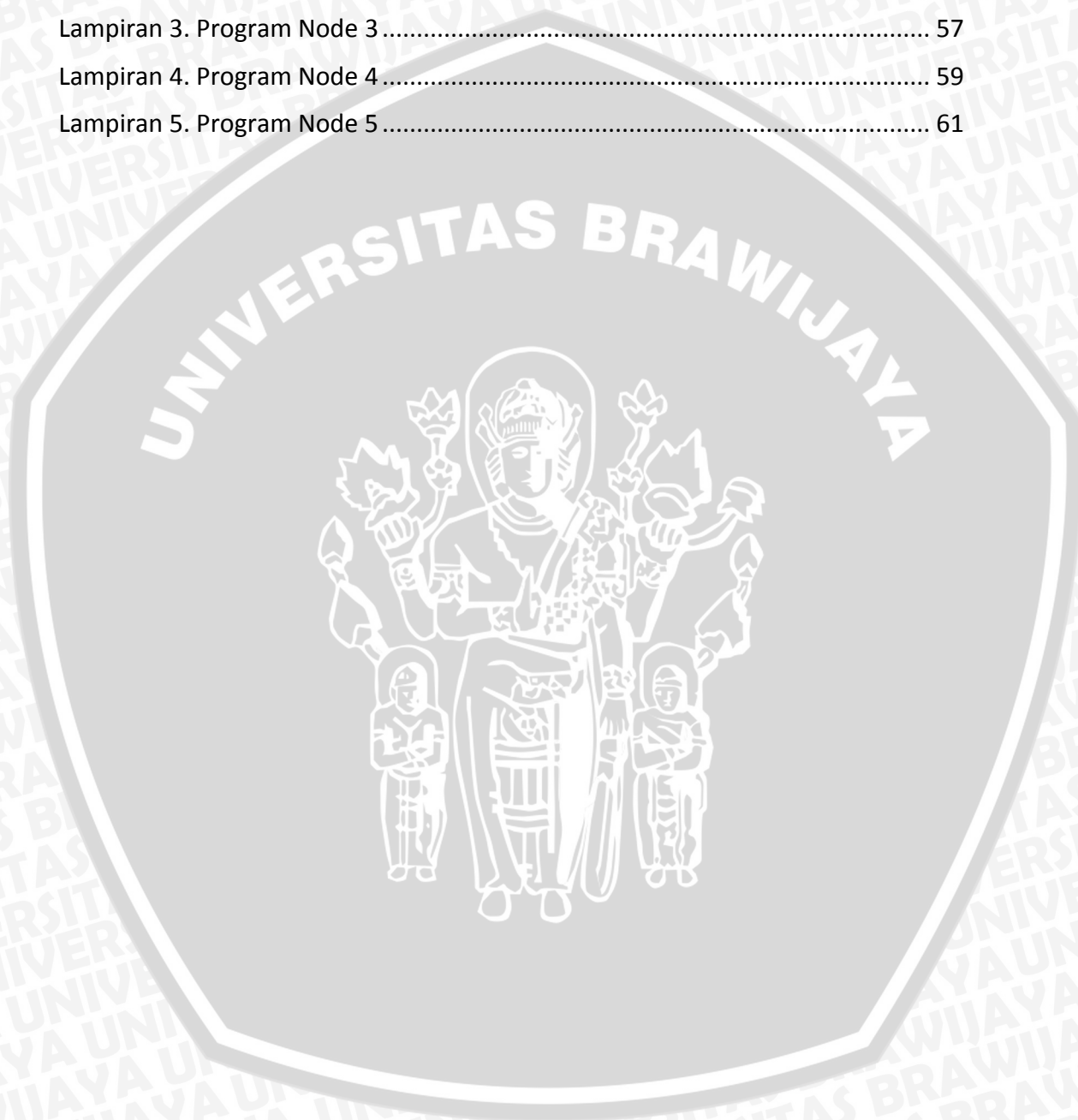
## DAFTAR GAMBAR

|                                                                         |    |
|-------------------------------------------------------------------------|----|
| Gambar 2.1 WSN Kategori 1 .....                                         | 5  |
| Gambar 2.2 WSN Kategori 2 .....                                         | 5  |
| Gambar 2.3 Arduino Nano .....                                           | 6  |
| Gambar 2.4 Modul NRF24L01 .....                                         | 8  |
| Gambar 2.5 Protokol SPIN - PP .....                                     | 9  |
| Gambar 3.1 Diagram Alir .....                                           | 11 |
| Gambar 3.2 Diagram Blok Analisa Kebutuhan Sistem .....                  | 13 |
| Gambar 3.3 Diagram Blok Sistem Keseluruhan .....                        | 14 |
| Gambar 3.4 Proses Routing SPIN .....                                    | 15 |
| Gambar 4.1 Diagram Blok Kebutuhan Perangkat Keras .....                 | 22 |
| Gambar 5.1 Perancangan Node .....                                       | 25 |
| Gambar 5.2 Perancangan Rangkaian Mikrokontroler dengan Modul NR24L01 .. | 25 |
| Gambar 5.3 Flowchart Pengiriman Data node 1 ke node 2 .....             | 26 |
| Gambar 5.4 Flowchart Pengiriman Data node 2 ke node 3 .....             | 27 |
| Gambar 5.5 Flowchart Pengiriman Data node 2 ke node 4 .....             | 28 |
| Gambar 5.6 Flowchart Pengiriman Data node 2 ke node 5 .....             | 29 |
| Gambar 5.7 Perancangan Topologi Node .....                              | 30 |
| Gambar 5.8 Implementasi Perangkat Keras SPIN pada WSN .....             | 31 |
| Gambar 5.9 Library yang Dibutuhkan pada Implementasi routing SPIN ..... | 32 |
| Gambar 5.10 Diagram Blok Komunikasi Data .....                          | 32 |
| Gambar 5.11 Node 1 mengirim ADV meta-data ke node 2 .....               | 33 |
| Gambar 5.12 Node 2 menerima meta-data dari node 1 .....                 | 34 |
| Gambar 5.13 Node 2 melakukan REQ ke node 1 .....                        | 34 |
| Gambar 5.14 Node 1 menerima REQ dari node 2 .....                       | 34 |
| Gambar 5.15 Node 1 mengirim DATA ke node 2 .....                        | 35 |
| Gambar 5.16 Node 2 menerima DATA dari node 1 .....                      | 35 |
| Gambar 5.17 Node 2 mengirim pesan ADV meta-data ke node 3,4,5 .....     | 35 |
| Gambar 5.18 Node 3 menerima meta-data dari node 2 .....                 | 36 |
| Gambar 5.19 Node 4 menerima meta-data dari node 2 .....                 | 36 |
| Gambar 5.20 Node 5 menerima meta-data dari node 2 .....                 | 37 |

|                                                                            |    |
|----------------------------------------------------------------------------|----|
| Gambar 5.21 Node 2 menerima REQ dari node 3 dan 4 .....                    | 37 |
| Gambar 5.22 Node 5 tidak meminta data .....                                | 37 |
| Gambar 5.23 Node 2 mengirimkan DATA ke node 4 dan 5 .....                  | 37 |
| Gambar 5.24 Node 3 menerima DATA dari node 2 .....                         | 38 |
| Gambar 5.25 Node 4 menerima DATA dari node 2 .....                         | 38 |
| Gambar 6.1 Hubungan Node Sensor dengan PC .....                            | 39 |
| Gambar 6.2 Dua Indikator yang menyatakan Node Sensor tidak bermasalah..... | 40 |
| Gambar 6.3 Hubungan Keseluruhan Sistem .....                               | 40 |
| Gambar 6.4 Hasil Pengujian Pengiriman Paket Data di Node 1.....            | 42 |
| Gambar 6.5 Hasil Pengujian Pengiriman Paket Data di Node 2.....            | 42 |
| Gambar 6.6 Hasil Pengujian Pengiriman Paket Data di Node 2.....            | 43 |
| Gambar 6.7 Hasil Pengujian Pengiriman Paket Data di Node 3.....            | 44 |
| Gambar 6.8 Hasil Pengujian Pengiriman Paket Data di Node 4.....            | 45 |
| Gambar 6.9 Hasil Pengujian Pengiriman Paket Data di Node 5.....            | 46 |
| Gambar 6.10 Grafik Hasil Pengujian Delay di Node 1.....                    | 48 |
| Gambar 6.11 Grafik Hasil Pengujian Delay di Node 2.....                    | 48 |

## DAFTAR LAMPIRAN

|                                  |    |
|----------------------------------|----|
| Lampiran 1. Program Node 1 ..... | 52 |
| Lampiran 2. Program Node 2 ..... | 54 |
| Lampiran 3. Program Node 3 ..... | 57 |
| Lampiran 4. Program Node 4 ..... | 59 |
| Lampiran 5. Program Node 5 ..... | 61 |



## BAB 1 PENDAHULUAN

### 1.1 Latar belakang

Pada masa modern saat ini, teknologi dan komunikasi di bidang internet telah berkembang secara signifikan dan menjadi kebutuhan sehari – hari, selain itu pertukaran data secara nirkabel atau wireless merupakan suatu media yang saat ini paling banyak digunakan dalam komunikasi. (Aziz, Soebarto, & Akbar, 2014)

Jaringan Sensor Nirkabel mempunyai banyak node yang tersebar untuk memantau wilayah yang besar. Untuk bisa beradaptasi dan bekerja pada lingkungan dengan kondisi tertentu, node-node tersebut biasanya memiliki topologi yang tidak tetap. Hal ini dapat dilakukan dengan mengukur suhu tekanan, kelembaban, suara, getaran dll. (Sharma, Mittal, & Rathi, 2014)

Sensor node tidak hanya berkomunikasi antara satu sama lain, tetapi juga berkomunikasi dengan Base Station (BS) menggunakan radio nirkabel. Hal ini memungkinkan node untuk menyebarkan data sensor dalam proses analisis dan penyimpanan. (Sharma, Sehwat, & Jyoti, 2012)

Pada jaringan sensor nirkabel ada banyak kendala seperti keterbatasan resource energi, desain, keamanan, dll. (Akkaya & Younis, 2005). Penggunaan energi terbanyak pada tiap node terjadi pada saat node tersebut bertukar informasi dalam jaringan. Semakin jauh node tersebut mengirimkan data, maka akan dibutuhkan energi yang semakin besar. (Perillo & Heinzelman, 2005). Oleh karena itu, saat ini sudah banyak dikembangkan berbagai algoritma protokol routing untuk mengatasi masalah tersebut. Beberapa contoh algoritma yang dikembangkan yaitu LEACH, SPIN, GBR, PEGASIS, dll.

Algoritma SPIN (Sensor Protocol for Information via Negotiation) merupakan protokol routing yang bersifat data-centric. (Basagni, Distributive and Mobility-Adaptive Clustering for Multimedia Support in Multi-hop Wireless Networks, 1999). SPIN merupakan pengembangan dari protokol flooding yang digunakan untuk mencegah terjadinya tumpang tindih dan ledakan data. Keuntungan dari protokol ini adalah setiap node sumber akan melakukan negosiasi dengan node tujuan sebelum proses transmisi data dilakukan yaitu dengan mengirimkan meta-data terlebih dahulu sebelum mengirimkan data aktual, karena jika node tujuan tidak menginginkan data tersebut, maka node sumber tidak akan mengirimkannya, sehingga akan mengurangi masuknya data yang berlebihan.

Pengembangan dan analisis algoritma SPIN sebelumnya sudah banyak dilakukan, sebagai contoh adalah dilakukan analisis modifikasi Algoritma SPIN dengan penambahan VC table di tinyos, ini dilakukan untuk mengurangi masalah kinerja yang ada pada algoritma SPIN seperti “blindly forward” dan “data uncessible”. Dengan penambahan VC Table, node tidak perlu mengirimkan informasi data secara terus menerus dan membuang data yang sama, selain itu node akan mengetahui ke node mana dia harus mengirimkan data. (Mayasari, Munadi, & Sumaryo, 2013)

Pada permasalahan tersebut, maka akan di lakukan implementasi protokol routing menggunakan algoritma SPIN pada lingkungan *Wireless Sensor Network*. Node sensor tersebut akan menggunakan arduino nano dan menggunakan komunikasi radio frekuensi (RF) sebagai komunikasi pengiriman dan penerima data.

## 1.2 Rumusan masalah

1. Bagaimana menerapkan proses routing SPIN pada *wireless sensor network* ?
2. Bagaimana mekanisme pertukaran informasi menggunakan protokol routing SPIN pada *wireless sensor network*?
3. Bagaimana mengetahui performa routing SPIN pada *wireless sensor network*?

## 1.3 Tujuan

1. Mengimplementasikan algoritma routing SPIN pada *wireless sensor network*.
2. Mengetahui mekanisme pertukaran informasi menggunakan protoko routing SPIN pada *wireless sensor network*.
3. Mengetahui performa dari protokol routing SPIN pada *wireless sensor network*.

## 1.4 Manfaat

### 1.4.1 Bagi Penulis

Sebagai pengaplikasian dari teori dan praktikum mata kuliah yang telah dipelajari selama mengikuti pendidikan di Jurusan Informatika Keminatan Teknik Komputer Fakultas Ilmu Komputer Universitas Brawijaya dengan mengimplementasikan Protokol Routing Spin Pada *Wireless Sensor Network*.

### 1.4.2 Bagi Pengguna/ Pembaca

Penelitian ini diharapkan dapat bermanfaat sebagai referensi dalam pengaplikasian algoritma routing pada *wireless sensor network*.

## 1.5 Batasan Masalah

1. Komunikasi pengiriman data menggunakan komunikasi RF.
2. Pengujian hanya mengimplementasikan algoritma SPIN secara umum.
3. Menggunakan mikrokontroler Arduino nano, dan menggunakan bahasa pemograman bahasa C.
4. Windows 8.1 sebagai sistem operasi.
5. Saat proses pengiriman data antar node berlangsung, kemungkinan kecil akan terjadi kegagalan pengiriman paket data (*packet loss*).
6. Ukuran data yang dikirimkan 4 byte dan ukuran meta-data 1 byte.
7. Pengujian pada sistem ini hanya mengacu pada pengiriman pesan ADV, REQ dan DATA.
8. Menggunakan lima buah node arduino nano dan 5 buah NRF24L01.
9. Menggunakan WSN kategori 2 (C2WSNs).

## 1.6 Sistematika pembahasan

Sistematika pembahasan dari penyusunan implementasi routing protokol SPIN pada wireless sensor *network* akan disusun sebagai berikut:

### **BAB I : PENDAHULUAN**

Bab I membahas mengenai latar belakang masalah, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan masalah, sistematika pembahasan.

### **BAB II : LANDASAN KEPUSTAKAAN DAN DASAR TEORI**

Bab II membahas mengenai dasar teori yang didapat dari sumber-sumber yang relevan untuk digunakan sebagai referensi dalam pembuatan proposal skripsi implementasi routing protokol SPIN pada *wireless sensor network*.

### **BAB III : METODOLOGI PENELITIAN**

Bab III membahas mengenai metode penelitian terkait dengan implementasi routing protokol SPIN pada *wireless sensor network*.

### **BAB IV : REKAYASA KEBUTUHAN**

Bab IV membahas mengenai rekayasa kebutuhan antara lain pendahuluan, deskripsi umum dan spesifikasi kebutuhan.

### **BAB V : PERANCANGAN DAN IMPLEMENTASI**

Bab V membahas mengenai perancangan sistem yang terdiri dari perancangan perangkat lunak dan perangkat keras yang digunakan. Implementasi dari sistem yaitu mengimplementasikan routing SPIN pada *wireless sensor network*.

### **BAB VI : PENGUJIAN**

Bab VI membahas mengenai proses dan hasil pengujian sistem sesuai dengan metode yang telah direalisasikan, sehingga sistem dapat diimplementasikan dengan baik.

### **BAB VII: PENUTUP**

Bab VII membahas mengenai kesimpulan dan saran yang didapatkan dari rumusan masalah dan tujuan yang dilakukan dengan menganalisis hasil pegujian. Serta pengembangan dari Implementasi routing SPIN pada *Wireless Sensor Network*.

## BAB 2 LANDASAN KEPUSTAKAAN

### 2.1 Wireless Sensor Network (WSN)

*Wireless Sensor Network* adalah sebuah jaringan nirkabel yang terdiri dari suatu perangkat komunikasi sensor dan digunakan untuk memantau kondisi fisik suatu lingkungan tertentu pada lokasi yang berbeda antara sensor dan pemroses datanya (Maribun Sibarani, 2008). Jaringan komunikasi *wireless* sensor biasanya digunakan pada industri ataupun aplikasi komersial lainnya yang kesulitan dengan pemasangan sistem perkabelan. Area penggunaan dari *wireless* sensor ini adalah seperti monitor tingkat polusi atau kontaminasi udara, pengendali reaktor nuklir, sistem pendeteksi kebakaran atau semburan panas bumi dan monitoring kelembaban pada tanaman tertentu di bidang pertanian, dll.

Beberapa karakteristik dari *wireless sensor network* diantaranya adalah:

1. Power (Daya) yang terbatas bisa diolah atau disimpan.
2. Mampu bertahan pada lingkungan yang sulit dijangkau dan di kontrol secara terus menerus.
3. Kemampuan untuk mengatasi kesalahan *node*
4. Dapat digunakan untuk kondisi dan pemrosesan data secara *mobile*
5. Mempunyai topologi jaringan yang tidak tetap atau berubah-ubah.
6. Penyebarannya dapat berkembang pada skala yang besar.

Saat *Wireless Sensor Network* sudah dilengkapi dengan perangkat komunikasi nirkabel (*transceiver* radio) dan sumber energi yang biasanya berupa baterai. Ada empat komponen dasar dalam jaringan sensor:

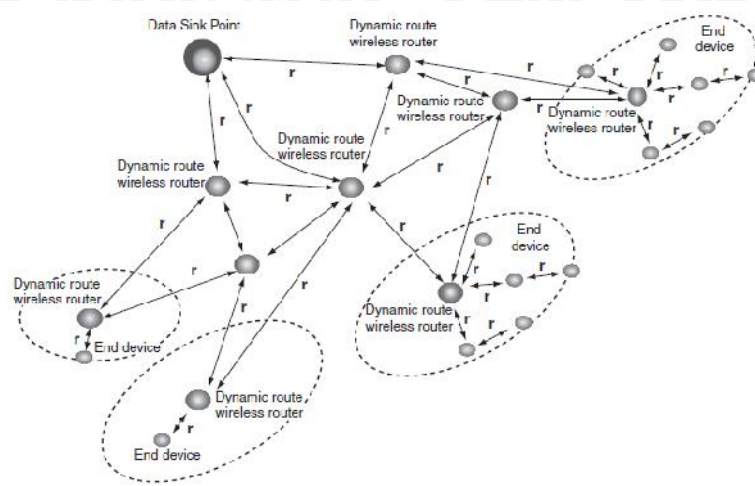
- a) Perakitan sensor secara distribusi atau lokal
- b) Jaringan interkoneksi
- c) Titik pusat dari informasi clustering
- d) Satu set sumber daya komputasi pada titik pusat (atau di luar) untuk menangani korelasi data, *status query*, dan data mining.

Terdapat 2 kategori dalam WSN, yaitu:

- 1) WSN Kategori 1 (C1WSNs)

Pada kategori 1, jaringan menggunakan topologi mesh dengan jaringan radio multihop di kalangan atau diantara node pada WSN, dan memanfaatkan routing dinamis pada *wireless* dan kabel jaringan.

C1WSNs merupakan jaringan dimana perangkat akhir (*end device*) yaitu sensor, diijinkan untuk lebih dari satu radio hop yang jauh dari *forwarding node*. Berikut gambar topologi kategori 1 seperti pada Gambar 2.1 :

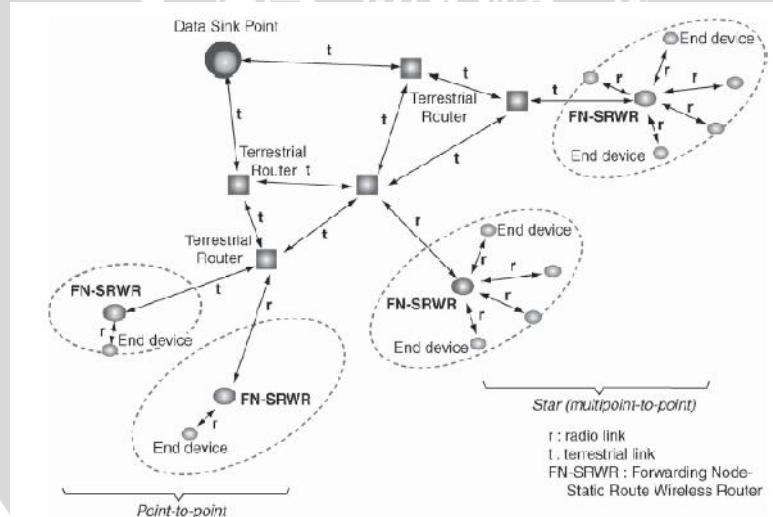


**Gambar 2.1 WSN Kategori 1**

## 2) WSN Kategori 2 (C2WSNs)

Pada kategori 2, jaringan menggunakan topologi star (*point-to-point* atau *multipoint-to-point*) dengan jaringan radio *single-hop*, memanfaatkan routing statis melalui jaringan nirkabel.

C2WSNs merupakan jaringan dimana perangkat akhir (*end device*) yaitu sensor, hanya terdapat satu radio hop yang jauh dari *forwarding node*. Berikut gambar topologi kategori 2 seperti pada Gambar 2.2 :



**Gambar 2.2 WSN Kategori 2**

## 2.2 Radio Frequency

Radio frekuensi merupakan sinyal arus bolak-balik frekuensi tinggi yang lewat pada suatu konduktor tembaga dan kemudian dipancarkan ke udara melalui suatu antenna (Grace G.P Usmany dan Bobi Kurniawan Soegoto, 2013). Radio frekuensi merupakan gelombang elektromagnetik yang digunakan oleh sistem komunikasi untuk mengirim informasi melalui udara dari satu titik ke titik lain. Radio frekuensi

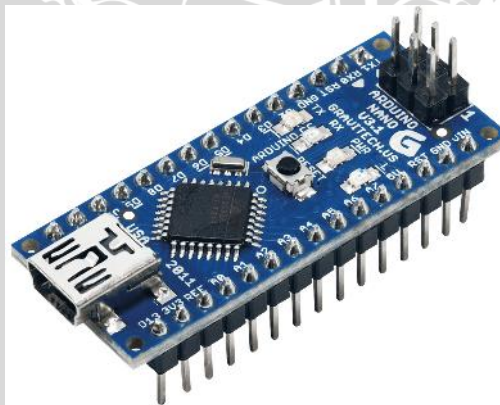
disebut juga arus bolak-balik (AC) yang terus berubah antara tegangan positif dan negatif.

Panjang gelombang  $\lambda$  adalah jarak antara dua puncak berurutan (puncak) atau dua palung berturut-turut (lembah) dari pola gelombang. Dengan kata sederhana, panjang gelombang adalah jarak satu siklus dari sinyal RF yang sebenarnya.

Hal ini sangat penting untuk memahami bahwa ada hubungan terbalik antara panjang gelombang dan frekuensi. Tiga komponen dari hubungan terbalik ini adalah frekuensi ( $f$ , diukur dalam hertz, atau Hz), panjang gelombang  $\lambda$ , diukur dalam meter, atau m), dan kecepatan cahaya ( $c$ , yang merupakan nilai konstan 300.000.000 m / detik). Rumus referensi berikut menggambarkan hubungan:  $\lambda = c / f$  dan  $f = c / \lambda$ . Penjelasan sederhana adalah bahwa semakin tinggi frekuensi sinyal RF, semakin kecil panjang gelombang dari sinyal. Semakin besar panjang gelombang dari sinyal RF, semakin rendah frekuensi sinyal.

### 2.3 Arduino Nano

Arduino Nano adalah board berbasis mikrokontroler Atmel ATmega168 atau ATmega328. Board ini memiliki 14 pin *digital input/output* masing-masing mempunyai tegangan 5 volt, 6 pin dapat digunakan sebagai output PWM dengan menggunakan fungsi `pinMode()`, `digitalRead` dan `digitalWrite()`. Setiap pin dapat menerima atau memberikan arus maksimum 40 mA dan memiliki resistor pull-up internal sebesar 20-50 KOhm, 8 pin analog, 16 Mhz osilator Kristal serta tombol reset. Pin-pin ini berisi semua yang diperlukan untuk mendukung mikrokontroler. Untuk *upload* Arduino Nano ini bisa menggunakan FTDI USB to TTL.



**Gambar 2.3 Arduino Nano**  
**Sumber: [www.arduino.cc](http://www.arduino.cc)**

Berdasarkan Gambar 2.3 diatas, arduino nano memiliki fitur-fitur seperti dibawah ini.

1. **Serial 0 (RX) dan 1 (TX)**, pin ini terhubung ke pin yang sesuai dari chip FTDI USB -TTL Serial dan digunakan untuk mengirimkan (TX) TTL data serial dan menerima (RX).

2. **External Interrupt** Pin 2 dan pin 3, pin ini dapat dikonfigurasi untuk memicu sebuah interupsi pada perubahan nilai, mulai nilai yang rendah, meningkat atau menurun.
3. **PWM** Pin 3, 5, 6, 9, 10, dan 11, pin ini mempunyai output PWM 8-bit dengan fungsi analogWrite().
4. **SPI** Pin 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK), pin ini yang mendukung komunikasi SPI yang tersedia pada hardware.
5. **LED** Pin 13, pin ini tersedia secara built-in pada papan Arduino Nano. LED terhubung ke pin digital 13. Led akan menyala ketika pin diset bernilai HIGH, dan akan padam ketika pin diset bernilai LOW.

Arduino Nano memiliki 8 pin sebagai input analog yang diberi label A0 sampai dengan A7 dan masing-masing mempunyai resolusi 10 bit atau 1024 nilai yang berbeda. Pin Analog 6 dan 7 tidak dapat digunakan sebagai pin digital. Selain itu juga, beberapa pin memiliki fungsi yang dikhususkan, yaitu:

1. **GND** adalah pin Ground atau negatif.
2. **Vin** pin ini digunakan apabila ingin memberikan power langsung ke board Arduino dengan rentang tegangan yang disarankan 7V-12V.
3. **Pin 5V** Ini adalah pin output dimana pada pin tersebut tegangan 5V yang telah melalui regulator.
4. **3V3** adalah pin output dengan tegangan 3V yang telah melalui regulator.
5. **I<sup>2</sup>C** pin A4(SDA) dan A5(SCL), pin ini mendukung komunikasi I<sup>2</sup>C (TWI) menggunakan Wire Library.
6. **AREF** Referensi tegangan untuk input analog yang digunakan dengan fungsi analogReference().
7. **RESET** merupakan jalur LOW yang digunakan untuk menghidupkan ulang (reset) mikrokontroler.

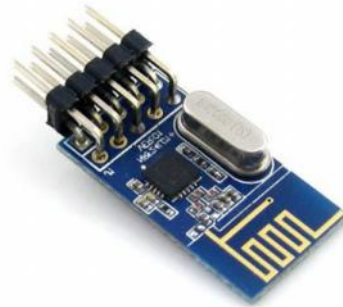
Beberapa kelebihan yang dimiliki oleh Arduino adalah sebagai berikut:

1. Sudah ada bootloader yang menangani upload program dari komputer jadi tidak perlu perangkat chip programmer.
2. Memiliki sarana komunikasi USB yang dapat digunakan untuk pengguna yang tidak memiliki port serial/RS323.
3. Bahasa pemrograman yang mudah dan dilengkapi dengan kumpulan library yang lengkap.
4. Memiliki shield atau modul siap pakai yang ditancapkan pada board arduino nano. Misal shield GPS, Ethernet, SD Card, dll.

## 2.4 Nrf24L01

nRF24L01 adalah sebuah modul komunikasi jarak jauh dengan memanfaatkan pita gelombang RF 2.4GHz *Industrial, Scientific and Medical* (ISM) yang mempunyai tegangan sebesar 5V DC, memiliki 8 buah pin diantaranya VCC (3.3V DC), GND, CE, CSN, MOSI, MISO, SCK, IRQ dan untuk berkomunikasi

menggunakan antarmuka SPI. Pada Gambar 2.6 berikut merupakan bentuk dari Modul NRF24L01 :



**Gambar 2.4 Modul nRF24L01**

**Sumber : [www.arduinor.com](http://www.arduinor.com)**

Modul nRF24L01 dapat digunakan untuk pembuatan piranti permainan, mainan anak-anak, peripheral PC dan alat lainnya. Selain itu juga memiliki true ULP solution yang dapat membuat daya baterai tahan lama berbulan-bulan sampai bertahun-tahun.

**a. Spesifikasi pada modul NRF24L01**

- Worldwide 2.4GHz ISM band operation
- Jangkauan data di udara yaitu 250kbps, 1bps and 2Mbps
- Ultra low power operation
- 11.2Ma TX at 0dBm output power
- 13.5Ma RX at 2Mbps air data rate
- Power turun pada 900Na
- 26Ma In standby-I
- On chip voltage regulator
- 1.9 to 3.6V supply range
- On-air compatible in 259kbps dan 1Mbps dengan Nrf2401A, Nrf2402, Nrf24E1 dan Nrf24E2
- 6 data pipe
- $\pm 60$ ppm 16 MHz crystal
- 5V input
- Compact 20 pin 4x4mm QFN package

**b. Implementasi Modul NRF24L01**

- Wireless PC Peripherals
- Mouse, keyboard dan remot kontrol
- Pengontrol Game
- Remote control RF untuk pengguna elektronika
- Rumah dan automasi komersil
- Jaringan sensor dengan daya rendah
- Active RFID
- Asset tracking sistems

**c. Fitur-Fitur pada Modul NRF24L01**

1. Radio
2. Transmitter

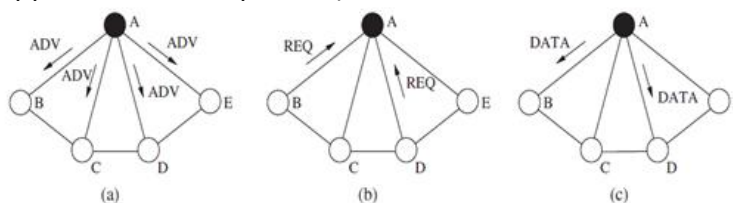
3. Receiver
4. IRF Synthesizer
5. Enhanced ShockBurst™:
6. Power Management :
7. Host Interface

## 2.5 SPIN

SPIN (Sensor Protocol for Information via Negotiation) merupakan protokol *routing* yang bersifat *data-centric* (Basagni, Distributive and Mobility-Adaptive Clustering for Multimedia Support in Multi-hop Wireless Networks, 1999). Node sensor akan melakukan negosiasi dengan satu sama lain sebelum melakukan proses transmisi data aktual menggunakan meta-data. Meta-data merupakan gambaran dari suatu data yang dikumpulkan oleh sensor node, selain itu ukuran meta-data tersebut tidak boleh melebihi data aktualnya. SPIN menggunakan tiga jenis pesan ADV, REQ, dan DATA dalam proses pengirimannya. ADV digunakan untuk mengiklankan data baru, REQ digunakan untuk meminta data, dan DATA adalah pesan itu sendiri. Ketika sebuah node mendapatkan data baru, node tersebut akan melakukan advertisement (ADV) dengan mengirimkan meta-data ke node tetangganya, saat pesan ADV diterima, node tetangga akan melakukan pengecekan apakah data tersebut sudah diterima atau tidak sebelumnya, jika sudah diterima maka node tersebut tidak akan melakukan request data. Jika tidak maka node tetangga akan mengirimkan pesan permintaan (REQ) ke node pengirim dan meminta untuk dikirimkan data baru, dan setelah itu node pengirim mengirimkan pesan DATA yang merupakan data lengkap dari suatu meta-data, dan setelah itu proses akan selesai. SPIN dirancang untuk mengatasi masuknya data yang berlebihan yang diselesaikan dengan negosiasi terlebih dahulu, sehingga energy yang dikeluarkan lebih efisien. SPIN mencakup banyak protokol diantaranya adalah SPIN-PP, SPIN-EC, SPIN-BC, dan SPIN RL.

### a. SPIN-PP

Protokol ini dirancang untuk berkomunikasi dari satu titik ke titik lain. Pada Gambar 2.8 menunjukkan komunikasi antara dua node yang saling berkomunikasi tanpa harus mengganggu node lainnya. Dalam hal ini dilakukan negosiasi terlebih dahulu antara node pengirim dan node penerima, proses 3-way *handshake* juga berlaku pada protokol SPIN-PP, ini dilakukan untuk menghindari terjadinya kelebihan data. (Basagni, Distributive and Mobility-Adaptive Clustering for Multimedia Support in Multi-hop, 1999)



Gambar 2.5 Protokol SPIN-PP

(a) Advertisement Phase, (b) Request Phase, and (c) Data Transmission

Sumber : (Sharma, Mittal, & Rathi, 2014)

**b. SPIN-EC**

Protokol SPIN-EC tidak berbeda jauh dengan SPIN-PP yaitu berkomunikasi menggunakan 3-way handshake. Namun pada SPIN-EC terdapat parameter konservasi energi atau penambahan energi di dalamnya. (L.Li & Wu, 2007).

Pada saat node penerima sudah mendapatkan ADV node akan memeriksa apakah penggunaan energi yang digunakan di atas ambang batas atau dibawah ambang batas. Jika masih di atas ambang batas node akan melakukan proses pengiriman seperti yang dilakukan SPIN-PP, namun jika energi sudah dibawah ambang batas node tidak akan mengirim pesan REQ dan tidak akan terjadi proses komunikasi.

**c. SPIN-BC**

Protokol ini dirancang untuk saluran penyiaran node dimana jaringan berkomunikasi menggunakan satu saluran yang sama. (Hassanein & Luo, 2006). Pada saat node mengirimkan pesan ke node tujuan tetapi jalur tersebut sedang digunakan, maka node pengirim harus menunggu sampai jalur tersebut sudah siap untuk digunakan sebelum mencoba untuk melakukan pengiriman kembali sehingga akan menghemat penggunaan energi. Keuntungan dari SPIN-BC adalah ketika node mengirimkan pesan ke alamat yang broadcast, pesan ini akan sampai ke semua tetangga node yang lain hanya dengan menggunakan satu transmisi.

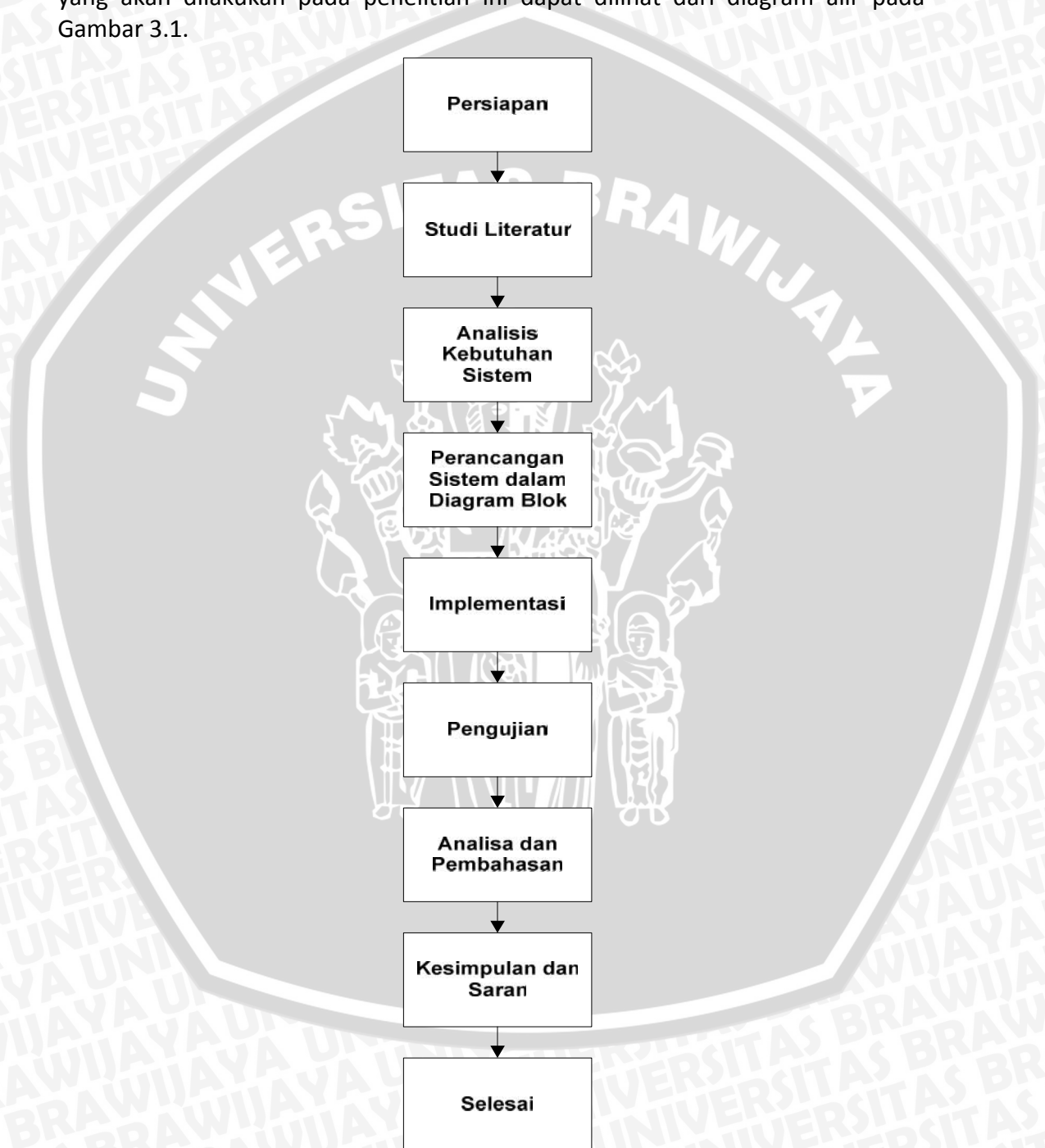
**d. SPIN-RL**

SPIN-RL cara kerjanya sama dengan SPIN-BC, tetapi lebih handal. Protokol ini melakukan penambahan dan perubahan yang ada pada pada SPIN-BC. Setiap node akan menerima ADV dan meminta REQ, jika dalam jangka waktu tertentu node tidak menerima data yang diminta maka node tersebut akan mengirimkan permintaan lagi. Setelah data dikirimkan, node tersebut akan menunggu dalam waktu tertentu sebelum melakukan proses REQ kembali untuk data yang sama.

## BAB 3 METODOLOGI

### 3.1 Alur Metode Penelitian

Pada bab ini akan dijelaskan tentang metode penelitian yang akan digunakan dalam tahapan-tahapan penelitian dan penyusunan skripsi. Metode penelitian yang akan dilakukan pada penelitian ini dapat dilihat dari diagram alir pada Gambar 3.1.



Gambar 3.1 Diagram Alir

### 3.2 Studi Literatur

Pada tahap studi literatur membahas mengenai penjelasan dasar teori yang digunakan untuk menunjang penulisan skripsi yang didapat dari buku, e-book, jurnal, artikel dan dokumentasi penelitian sebelumnya. Berikut adalah dasar teori yang digunakan untuk menunjang penulisan skripsi:

1. Pengenalan Routing Protokol SPIN (*Sensor Protocol for Information Negotiation*)

Studi literatur mengenai pengenalan SPIN (*Sensor Protocol for Information Negotiation*) ini dilakukan dengan mencari beberapa referensi dari jurnal, buku dan berbagai sumber yang terkait dengan algoritma routing spin dan cara kerja dari protokol Spin.

2. *Wireless Sensor Network*

Studi literatur mengenai *Wireless Sensor Network* dilakukan dengan mencari beberapa referensi dari jurnal, buku dan penelitian terkait dengan sistem secara umum. Pada tahap ini mempelajari bagaimana node pada WSN bekerja sesuai komponen penyusunnya.

3. Arduino

Studi literatur mengenai mikrokontroler arduino dibutuhkan sebagai referensi yang digunakan sebagai bahan implementasi protokol routing SPIN.

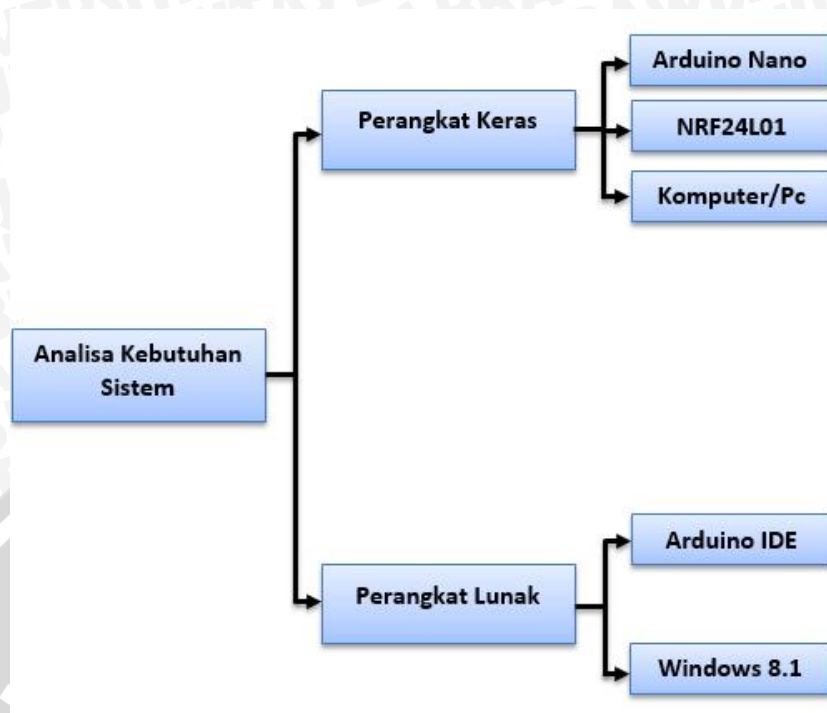
- a. Pengenalan Arduino
- b. Arduino IDE
- c. Pemrograman Arduino

4. NRF24L01

Studi literatur mengenai NRF24L01 dibutuhkan sebagai referensi yang digunakan sebagai bahan implementasi modul pengiriman data secara wireless.

### 3.3 Analisa Kebutuhan Sistem

Analisis kebutuhan bertujuan untuk mendapatkan semua kebutuhan pada penelitian. Pada analisis kebutuhan yang dilakukan adalah menentukan perangkat keras, perangkat lunak yang digunakan dalam perancangan dan implementasi sistem. Dengan demikian diharapkan dapat mempermudah dalam mendesain sistem. Diagram blok analisa kebutuhan sistem dapat dilihat pada Gambar 3.2 :



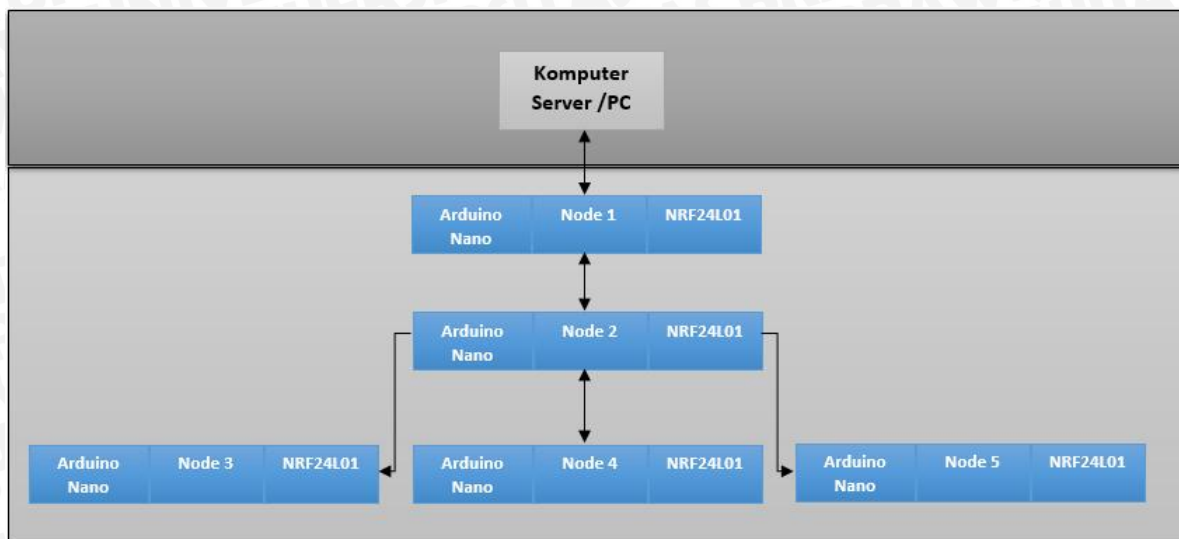
**Gambar 3.2 Diagram Blok Analisa Kebutuhan Sistem**

### 3.4 Perancangan Sistem dan Diagram Blok

Pada tahap perancangan sistem ini, di buat dalam bentuk diagram blok didalam diagram blok tersebut terdapat bagian-bagian dari sistem yang di buat. Yang digambarkan dalam bentuk yang sederhana seperti diagram alur berjalannya suatu sistem.

Pada tahap penggabungan diagram blok masing-masing blok atau perangkat keras yang digunakan digabungkan menjadi sebuah sistem yang nantinya akan difungsikan sebagaimana tujuan dari sistem tersebut, sehingga sistem dapat berjalan. Diantaranya penggabungan Arduino dan NRF24L01 sebagai node wsn yang dihubungkan dengan PC, selanjutnya pengiriman data dari node pengirim ke node penerima akan ditampilkan di serial monitor.

Diagram blok sistem terdiri dari node WSN yang mencakup node A, node B, node C, node D dan Node E. Node A akan mengirimkan pesan ke node B, setelah itu node B akan mengirimkan pesan yang sudah diterima ke node C,D,E. Pengiriman pesan ini memanfaatkan media komunikasi wireless (NRF24L01) seperti Gambar 3.3 :



**Gambar 3.3 Diagram Blok Sistem Keseluruhan**

Diagram blok ini menjelaskan sistem secara keseluruhan yang terdiri dari dua bagian utama, yaitu bagian node sensor dan komputer server. Bagian yang akan dirancang adalah node sensor, yaitu dengan menggunakan lima node untuk melakukan pengiriman data antar node. Node tersebut terdiri dari Arduino nano dan NRF24L01.

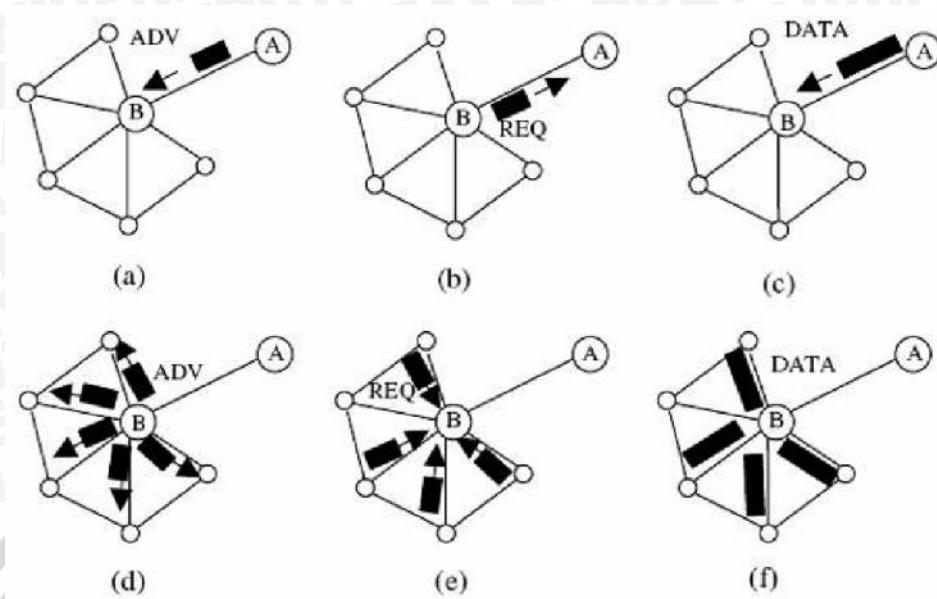
Bagian selanjutnya adalah komputer server yang berfungsi untuk mengatur dan menentukan proses pengiriman data antar node dan berapa data yang akan dikirimkan ke node yang dituju. Komputer server berfungsi untuk melakukan pengiriman data yang sudah diprogram sesuai dengan algoritma SPIN di Arduino IDE. Algoritma SPIN adalah proses pengiriman data secara wireless dengan melakukan negosiasi terlebih dahulu, node pengirim akan melakukan negosiasi apakah node yang dituju membutuhkan data atau tidak, jika data tersebut dibutuhkan maka node pengirim akan mengirimkan data yang diinginkan. Untuk mengirim data secara nirkabel digunakan arduino nano dan NRF24L01.

### 3.5 Implementasi

Implementasi sistem dilakukan dengan mengacu pada perancangan sistem yang sudah dibuat sebelumnya. Implementasi yang akan dilakukan adalah sebagai berikut :

#### 3.5.1 Implementasi Sensor Node

Sensor node berfungsi sebagai penerima dan pengirim data. Sensor node akan mengirim ADV ke node tetangganya yang lain, kemudian node tetangga tersebut akan membalas dengan mengirim REQ jika node tersebut membutuhkan data yang di ADV oleh node pengirim. Setelah itu node pengirim akan mengirimkan pesan DATA ke node tetangga yang mengirimkan pesan REQ. Pada Gambar 3.4 menunjukkan proses pengiriman data dari node pengirim ke node penerima.



**Gambar 3.4 Proses Routing SPIN**

**Sumber :** (Anisi, Abdullah, Razak, & Ngadi, 2012)

Seperti yang dapat dilihat pada gambar node A mengirimkan pesan ADV ke node B, kemudian node B membalas dengan mengirimkan REQ ke node A. Setelah itu node A akan mengirimkan pesan DATA ke node B. Selanjutnya node B yang sudah memiliki data yang diinginkan akan membroadcast pesan ADV ke node tetangganya apakah membutuhkan pesan yang dimiliki oleh node B, jika pesan tersebut dibutuhkan maka node tetangga tersebut akan mengirimkan pesan REQ kemudian node B akan mengirimkan pesan DATA ke node tetangga yang membutuhkan data tersebut.

### 3.6 Pengujian dan Analisis

Dari data yang sudah didapat dari pengujian sistem, akan dilakukan analisa dan pembahasan untuk mengetahui hasil dari pengujian untuk menarik kesimpulan dari penelitian yang dilakukan. Analisa hasil digunakan untuk mengetahui kelayakan dari sistem yang sudah dibuat. Pengujian yang akan dilakukan meliputi:

1. Pengujian node sensor ke PC
2. Pengujian Fungsional Sistem

### 3.7 Kesimpulan dan Saran

Kesimpulan dan saran didapat berdasarkan dari hasil pengujian setelah melakukan perancangan, implementasi, pengujian dan analisis pada sistem yang telah dibuat. Isi dari kesimpulan diharapkan dapat menjadi acuan pada penelitian selanjutnya dan digunakan sebagai pengembangan dari pemilihan routing SPIN. Selain itu, pada akhir penulisan terdapat saran yang bertujuan untuk memperbaiki

kesalahan-kesalahan yang terjadi dan menyempurnakan penulisan serta untuk memberikan pertimbangan atas pengembangan penelitian selanjutnya.



## BAB 4 REKAYASA KEBUTUHAN

### 4.1 Pendahuluan

Pada bab ini menjelaskan secara rinci implementasi protokol routing SPIN pada *Wireless Sensor Network* dengan menggunakan media komunikasi *radio frequency* berdasarkan format *Software Requirements Specification* IEEE 830 tahun 1998. Bab ini meliputi tujuan, fungsi, ruang lingkup sistem, batasan sistem bekerja, dan reaksi sistem dengan faktor eksternal yang ditujukan bagi para pengembang selanjutnya.

Sistem ini menggunakan lima buah node dan proses pengiriman datanya memerlukan negosiasi terlebih dahulu antara node pengiriman dan node penerima sesuai dengan algoritma SPIN.

#### 4.1.1 Tujuan

Tujuan dari bab ini untuk menjelaskan secara rinci sistem penelitian yang dimulai dari tujuan dan apa yang dibutuhkan sistem, serta bagaimana sistem akan bekerja. Sehingga akan didapat dokumentasi yang jelas mengenai alur dan pembuatan sistem yang nantinya akan ditujukan bagi para pengembang sistem selanjutnya.

#### 4.1.2 Ruang Lingkup

*Wireless Sensor Network* merupakan sebuah jaringan nirkabel yang terdiri dari suatu perangkat komunikasi sensor dan digunakan untuk memantau kondisi fisik suatu lingkungan tertentu. Sistem ini merupakan penerapan dari protokol routing SPIN pada lingkungan WSN yang dalam proses pengirimannya membutuhkan negosiasi terlebih dahulu sebagai syarat pengiriman data lebih lanjut. Penerapan protokol ini berupa lima buah node dimana pengiriman datanya menggunakan media komunikasi radio frequency (RF). Secara spesifik sistem ini ditujukan untuk monitoring jaringan nirkabel dan untuk menjamin bahwa tidak ada data yang berlebihan dikirim ke seluruh jaringan sehingga dapat diterapkan pada lingkungan WSN.

#### 4.1.3 Istilah

Tabel 4.1 Tabel Daftar Istilah

| Term                           | Definition                                                                              |
|--------------------------------|-----------------------------------------------------------------------------------------|
| <i>Wireless Sensor Network</i> | Merupakan sebuah jaringan nirkabel yang terdiri dari suatu perangkat komunikasi sensor. |
| SPIN                           | <i>Sensor Protocol Information via Negotiation</i> merupakan protokol routing yang      |

|                                            |                                                                                                                                             |
|--------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------|
|                                            | menggunakan algoritma negosiasi sebelum data ditransmisikan.                                                                                |
| RF                                         | <i>Radio frequency</i> merupakan suatu gelombang elektromagnetik yang terletak pada kisaran 3kHz sampai 300Hz.                              |
| Node                                       | Merupakan suatu rangkaian elektronik tunggal yang terhubung dalam <i>wireless sensor network</i> .                                          |
| <i>Software Requirements Spesification</i> | Dokumen yang menggambarkan fungsi dari suatu sistem dan menjelaskan kebutuhan sistem secara rinci agar sistem dapat berjalan sesuai tujuan. |

#### 4.1.4 Referensi

*IEEE Std 830 Tahun 1998 IEEE Recommended Practice for Software Requirements Spesification. IEEE Computer Society, 1998.*

#### 4.1.5 Sistematika

Untuk mempermudah memahami pembahasan pada dokumentasi ini, maka dibutuhkan sistematika penulisan yang merupakan kerangka pedoman penulisan dokumentasi. Dokumentasi dibagi menjadi 3 bagian sebagai berikut:

##### 1. Pendahuluan

Pada sub bab pendahuluan berisi tujuan, ruang lingkup, tabel istilah, referensi dan sistematika penulisan dokumen.

##### 2. Deskripsi Umum

Pada sub bab deskripsi umum berisi prespektif sistem, kegunaan sistem, karakteristik pengguna, batasan cara kerja sistem dan asumsi dari sistem.

##### 3. Spesifikasi Kebutuhan

Pada sub spesifikasi kebutuhan berisi kebutuhan fungsional sistem, kebutuhan antarmuka sistem, kebutuhan performasi, kendala dan kebutuhan sistem.

#### 4.2 Deskripsi Umum

##### 4.2.1 Perspektif Sistem

Sistem ini dikatakan bekerja sesuai tujuan apabila sistem dapat melakukan proses routing sesuai dengan algoritma SPIN dengan negosiasi antar node sebelum transmisi data dilakukan dengan melakukan 3 tipe pengiriman yaitu ADV, REQ, dan DATA.

#### 4.2.2 Kegunaan

Dengan adanya penerapan protokol routing SPIN pada *Wireless Sensor Network*, maka dari sisi pengiriman dapat dijamin bahwa tidak ada data yang berlebihan dikirim ke seluruh jaringan karena adanya negosiasi terlebih dahulu sebelum proses transmisi data.

#### 4.2.3 Karakteristik Pengguna

Karakteristik pengguna pada sistem ini terletak pada node pengirim dan penerima dimana node-node tersebut melakukan negosiasi sebelum komunikasi data berlangsung sesuai dengan algoritma SPIN.

#### 4.2.4 Lingkungan Operasi Sistem

1. Lingkungan harus mendukung proses kerja sistem, misalnya lingkungan tersebut tidak boleh basah dikarenakan dapat merusak komponen sistem tersebut dan tidak memiliki frekuensi gelombang yang sama dengan frekuensi gelombang yang digunakan oleh sistem.
2. Lingkungan harus mendukung proses pengiriman data agar data yang dikirimkan tidak hilang (*packetloss*), misalkan jarak antar node bisa dijangkau oleh node tetangga terdekat.

#### 4.2.5 Batasan Sistem

Berikut merupakan batasan yang ada pada sistem antara lain:

1. Komunikasi pengiriman data menggunakan komunikasi RF (*radio frequency*)
2. Proses ADV, REQ, DATA digunakan sebagai acuan topik pada protokol SPIN
3. Node pengirim mengirimkan data sesuai dengan alamat yang sudah ditentukan
4. Pengiriman data suhu akan dijadikan sebagai acuan topik pada routing SPIN

#### 4.2.6 Asumsi dan Ketergantungan Sistem

Berikut merupakan asumsi dan ketergantungan pada sistem antara lain:

1. Pengiriman data dapat berjalan apabila modul wireless (NRF24L01) diterapkan sesuai dengan rancangan rangkaian modul wireless dengan mikrokontroler.
2. Proses pengiriman data oleh node 2 akan diteruskan apabila node 2 menerima data dari node 1.
3. Node 3,4,5 tidak dapat menjangkau node 1 diasumsikan jarak dan sinyal yang tidak terjangkau.

### 4.3 Spesifikasi Kebutuhan

Pada bab ini menjelaskan seluruh kebutuhan sistem agar sistem dapat bekerja sesuai tujuan. Analisa kebutuhan pada sistem ini dibagi menjadi beberapa kebutuhan sebagai berikut:

#### 4.3.1 Kebutuhan Pengguna

Sistem dapat melakukan pengiriman data antara node pengirim dan penerima sesuai dengan algoritma SPIN menggunakan media komunikasi RF. Pengiriman data dilakukan dengan negosiasi antar node sumber dan node tujuan.

#### 4.3.2 Kebutuhan Fungsional

Kebutuhan fungsional dari sistem adalah kebutuhan yang harus terpenuhi agar sistem dapat dikatakan bekerja sesuai tujuan. Berikut merupakan beberapa kebutuhan fungsional pada sistem:

##### 4.3.2.1 Fungsi implementasi protokol routing SPIN pada WSN

1. Penjelasan dan Prioritas  
Fungsi ini mengharuskan sistem dapat melakukan pengiriman data sesuai dengan protokol routing SPIN. Fungsi ini termasuk prioritas tinggi karena skenario pengiriman data tersebut merupakan dasar dari protokol SPIN.
2. Stimulus dan Respon Sistem  
Sistem dapat berjalan sesuai dengan protokol routing SPIN pada saat proses pengiriman data berlangsung.
3. Kebutuhan Fungsional  
Sistem dapat menerapkan proses pengiriman data sesuai dengan protokol routing SPIN pada saat proses pengiriman data berlangsung.  
REQ-1 : Sistem melakukan proses routing  
REQ-2 : menerapkan proses negosiasi antara node sumber dengan node tujuan.

##### 4.3.2.2 Fungsi Node Sumber mengirimkan pesan ADV

1. Penjelasan dan Prioritas  
Fungsi ini mengharuskan sistem berupa node sumber melakukan pengiriman pesan ADV (*advertise*) ke node tujuan atau node tetangganya. Fungsi ini termasuk prioritas tinggi karena merupakan dasar dari proses pengiriman data.
2. Stimulus dan Respon Sistem  
Node sumber dapat mengirimkan pesan ADV berupa meta-data menuju ke node tujuan/tetangga terdekat.
3. Kebutuhan Fungsional  
Node sumber dapat mengirimkan pesan ADV berupa meta-data menuju ke node tujuan/tetangga terdekat sesuai dengan algoritma routing SPIN.  
REQ-1 : node sumber mengirimkan pesan ADV berupa meta-data ke node tetangga  
REQ-2 : pesan ADV dikirim menuju node tujuan melalui komunikasi wireless.

##### 4.3.2.3 Fungsi Node Tujuan mengirimkan REQ

1. Penjelasan dan Prioritas  
Fungsi ini mengharuskan sistem berupa node tujuan melakukan pengiriman pesan REQ (*request*) ke node sumber. Fungsi ini termasuk prioritas tinggi karena merupakan dasar dari proses pengiriman data.

## 2. Stimulus dan Respon Sistem

Node tujuan dapat mengirimkan pesan balasan berupa pesan REQ menuju ke node sumber.

## 3. Kebutuhan Fungsional

Node tujuan dapat mengirimkan pesan balasan berupa REQ menuju ke node sumber sesuai dengan algoritma routing SPIN.

REQ-1 : node tujuan mengirimkan pesan balasan berupa pesan REQ ke node sumber

REQ-2 : pesan REQ dikirim menuju node sumber melalui komunikasi wireless

### 4.3.2.4 Fungsi Node Sumber mengirimka DATA

## 1. Penjelasan dan Prioritas

Fungsi ini mengharuskan sistem berupa node sumber melakukan pengiriman pesan DATA yang diinginkan oleh node tujuan. Fungsi ini termasuk prioritas tinggi karena merupakan dasar dari proses pengiriman data.

## 2. Stimulus dan Respon Sistem

Node sumber dapat mengirimkan DATA yang diinginkan menuju ke node tujuan.

## 3. Kebutuhan Fungsional

Node sumber dapat mengirimkan DATA menuju ke node tujuan sesuai dengan algoritma routing SPIN.

REQ-1 : node tujuan mengirimkan DATA ke node tujuan.

REQ-2 : DATA dikirim menuju node tujuan melalui komunikasi wireless

### 4.3.3 Kebutuhan Komunikasi

Berdasarkan dari kebutuhan fungsional yang harus dipenuhi, maka kebutuhan komunikasi yang terjadi pada sistem ada 2 tahap yaitu node 1 ke node 2, dan node 2 ke node 3,4 dan 5. Komunikasi antar node ini dilakukan dengan menerapkan proses pengiriman pesan ADV, REQ, DATA. Komunikasi ini dilakukan pada channel dan payload yang sudah ditentukan sesuai dengan kebutuhan pesan yang akan dikirim, selain itu juga dilengkapi library khusus yang mendukung proses komunikasi.

### 4.3.4 Kebutuhan Sistem

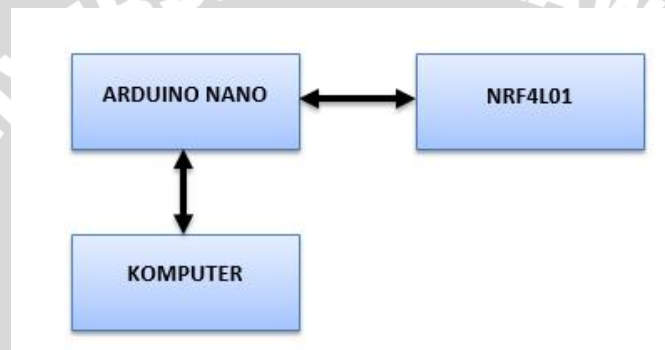
Berdasarkan dari kebutuhan fungsional dan kebutuhan komunikasi, maka sistem membutuhkan beberapa perangkat agar dapat terpenuhi. Pada sistem ini digunakan mikrokontroler arduino nano sebagai node yang digunakan untuk pengiriman dan penerimaan data. Untuk kebutuhan komunikasi secara wireless antara node pengirim dan node penerima sistem ini menggunakan NR24L01 yang merupakan modul *wireless* yang mampu menjalankan komunikasi melalui gelombang radio. Dengan adanya perangkat tersebut maka kebutuhan fungsional dan kebutuhan komunikasi pada sistem dapat terpenuhi dan berjalan sesuai tujuan.

#### 4.3.5 Kebutuhan Perangkat Keras

Kebutuhan perangkat keras pada sistem dibagi menjadi dua bagian. Pertama kebutuhan perangkat keras sistem yaitu kebutuhan perangkat saat bekerja pada kondisi yang sebenarnya, kedua adalah kebutuhan perangkat keras lingkungan development yaitu kebutuhan perangkat pendukung pada saat sistem dibangun.

##### 4.3.5.1 Kebutuhan Perangkat Keras Sistem

Untuk mengirimkan data secara nirkabel antar node WSN dibutuhkan sebuah perangkat keras (hardware) yang mampu mengirimkan, mengolah dan menerima data. Untuk mengolah data yang diterima dan dikirimkan digunakan arduino nano, sedangkan untuk pengiriman data secara nirkabel digunakan NRF24L01. Diagram blok kebutuhan hardware pada sistem seperti ditunjukan pada gambar 4.1 berikut.



**Gambar 4.1 Diagram Blok Kebutuhan Perangkat Keras**

Pada Gambar 4.1 menunjukkan kebutuhan hardware yang digunakan, analisa kebutuhan hardware yang diperlukan pada sistem ini antara lain sebagai berikut:

1. *Arduino Nano* yaitu mikrokontroller yang digunakan sebagai pengolah input dan output data
2. *NRF24L01* yaitu modul komunikasi yang digunakan untuk menerima data secara wireless
3. Komputer/PC yaitu untuk memonitoring jalannya sistem pada saat pertukaran data.

##### 4.3.5.2 Kebutuhan Perangkat Keras Lingkungan Development

Kabel USB digunakan pada saat proses penanaman program yang berfungsi sebagai penghubung komunikasi serial saat proses upload program menuju ke arduino nano.

#### 4.3.6 Kebutuhan Perangkat Lunak

Kebutuhan perangkat lunak pada sistem dibagi menjadi dua bagian. Pertama kebutuhan perangkat lunak sistem yaitu kebutuhan perangkat saat bekerja pada kondisi yang sebenarnya, kedua adalah kebutuhan perangkat lunak lingkungan development yaitu kebutuhan perangkat pendukung pada saat sistem dibangun.

#### 4.3.6.1 Kebutuhan Perangkat Lunak Sistem

Analisa kebutuhan software yang diperlukan pada sistem ini antara lain sebagai berikut:

1. SPIN merupakan protokol yang akan digunakan sebagai metode komunikasi pengiriman data.
2. Mirf library adalah library pada arduino yang digunakan untuk menjalankan pengiriman data secara wireless melalui gelombang radio.
3. Bahasa pemograman c++ adalah bahasa yang digunakan untuk menulis program Node 1 sampai node 5 pada arduino IDE.

#### 4.3.6.2 Kebutuhan Perangkat Lunak Lingkungan Development

Arduino IDE adalah perangkat lunak pendukung yang berfungsi untuk menulis, mengompile dan mengupload program pada mikrokontroler Arduino Nano.



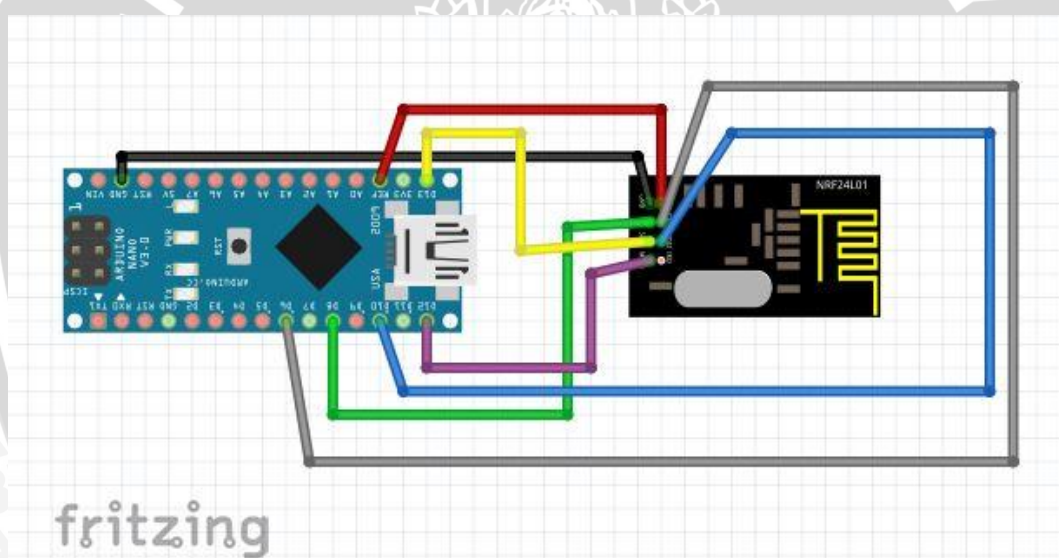
## BAB 5 PERANCANGAN DAN IMPLEMENTASI

### 5.1 Perancangan Sistem

Tahap perancangan berisi tentang analisis perangkat keras dan perangkat lunak yang digunakan untuk membuat sistem tersebut. Ada beberapa tahap di antaranya sebagai berikut:

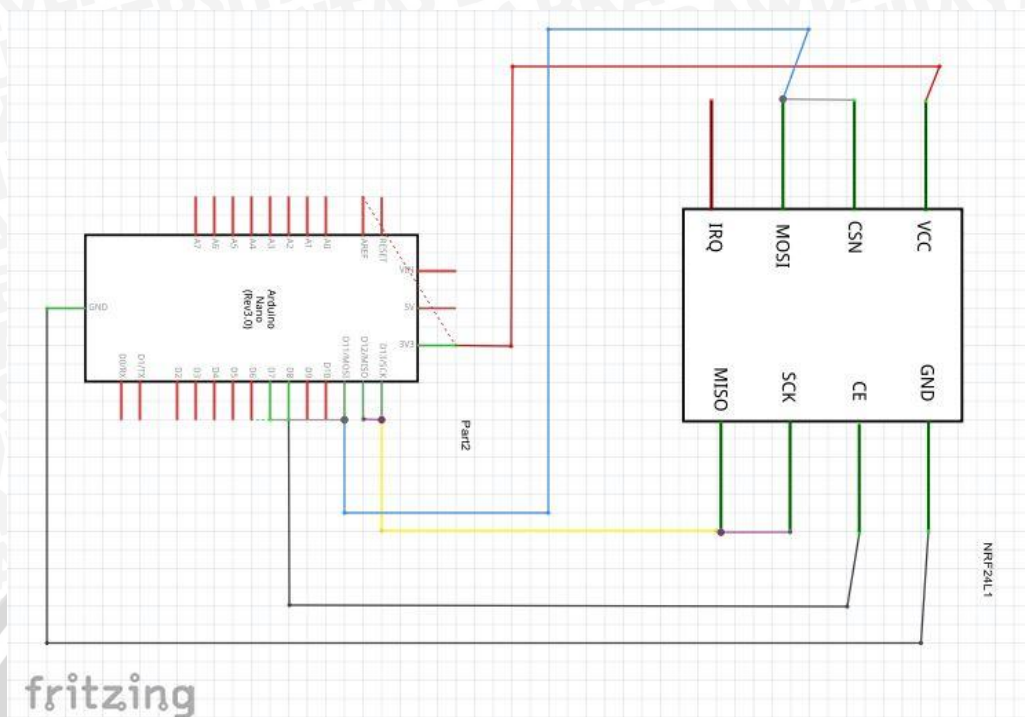
#### 5.1.1 Perancangan Perangkat Keras SPIN pada WSN

Perancangan perangkat keras SPIN pada WSN ini terdiri dari rangkaian mikrokontroler Arduino Nano dengan modul NRF24L01 yang membentuk suatu node. Pertama yang harus dilakukan adalah membuat rangkaian pada mikrokontroler yang bertujuan untuk memasukkan program pada mikrokontroler agar dapat dibaca, diproses dan menghasilkan output yang diinginkan. Kedua, yaitu membuat rangkaian pada NRF24L01 yang berfungsi sebagai modul wireless (pengiriman secara wireless). Pada Gambar 5.1 menunjukkan gambar skematik dari rangkaian mikrokontroler dengan modul NRF24L01.



**Gambar 5.1 Perancangan Node**

Berikut merupakan gambar skematik mikrokontroler dengan modul NRF24L01 dengan menggunakan aplikasi fritzing sehingga dapat mempermudah untuk mengimplementasikan kedalam alat yang sesungguhnya. Gambar perancangan keseluruhan dapat dilihat pada Gambar 5.2



**Gambar 5.2 Perancangan Rangkaian Mikrokontroler dengan Modul NRF24L01**

Pada gambar dapat dilihat bahwa untuk merancang satu buah node dibutuhkan 1 buah mikrokontroler Arduino Nano dan 1 buah NRF24L01. Sedangkan pin pada Arduino yang digunakan antara lain GND, VCC, D8, D7, D11, D12, D13 yang akan disambungkan pada pin nRF24L01. Berikut table 5.1 merupakan tabel pin sambungan nRF24L01 pada Arduino Nano:

**Tabel 5.1 Pin sambungan nRF24L01 pada Arduino nano**

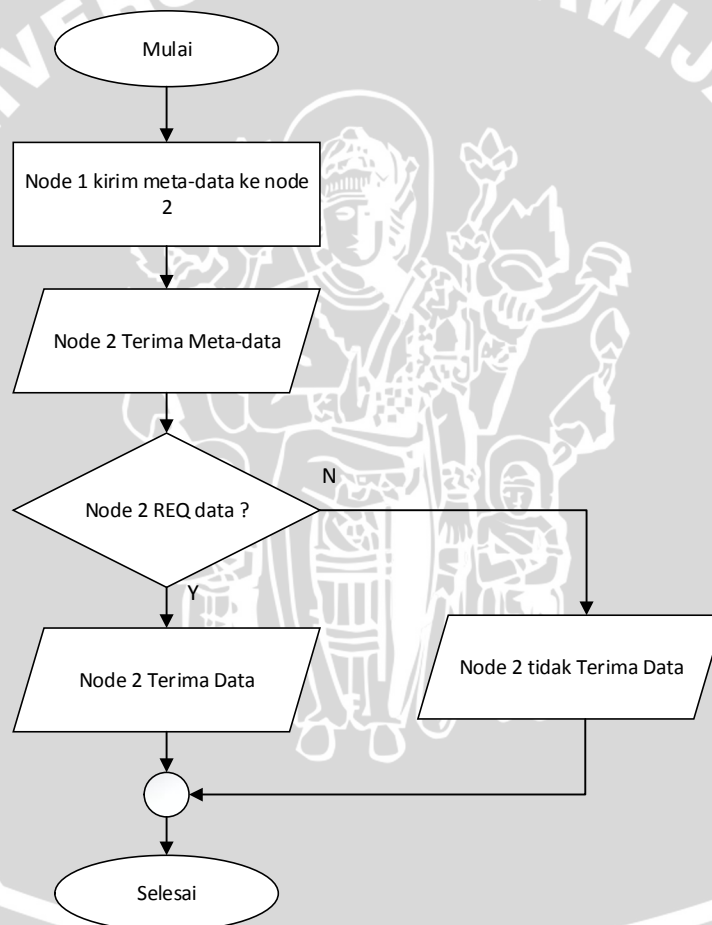
| Pin NRF24L01 | Pin Arduino Nano | Warna   |
|--------------|------------------|---------|
| GND          | GND              | HITAM   |
| VCC 3.3V     | VCC 3.3V         | MERAH   |
| CE           | D8               | HIJAU   |
| CSN          | D7               | ABU-ABU |
| SCK          | D13              | KUNING  |
| MOSI         | D11              | BIRU    |
| MISO         | D12              | UNGU    |
| IRQ          | -                | -       |

### 5.1.2 Perancangan Perangkat Lunak SPIN pada WSN

Perancangan perangkat lunak SPIN pada WSN berupa pengkondisian sistem melalui pemrograman pada arduino sebagai kontroler sistem agar sistem dapat melakukan pengiriman dan penerimaan data sesuai algoritma SPIN. Pemrograman ini ditulis menggunakan bahasa C pada arduino IDE dan decompile, kemudian di upload menuju mikrokontroler arduino melalui serial port pada komputer.

#### 5.1.2.1 Pengiriman data node 1 ke node 2

Pengujian pertama node 1 akan mengirimkan pesan ADV ke node 2 berupa meta-data suhu. Node 2 akan menerima pesan ADV dan akan mengirimkan pesan REQ ke node 1 yang artinya node 2 menginginkan data di node 1, kemudian node 1 akan mengirimkan DATA yang diinginkan oleh node 2. Berikut pada Gambar 5.3 merupakan flowchart pengiriman data node 1 ke node 2.

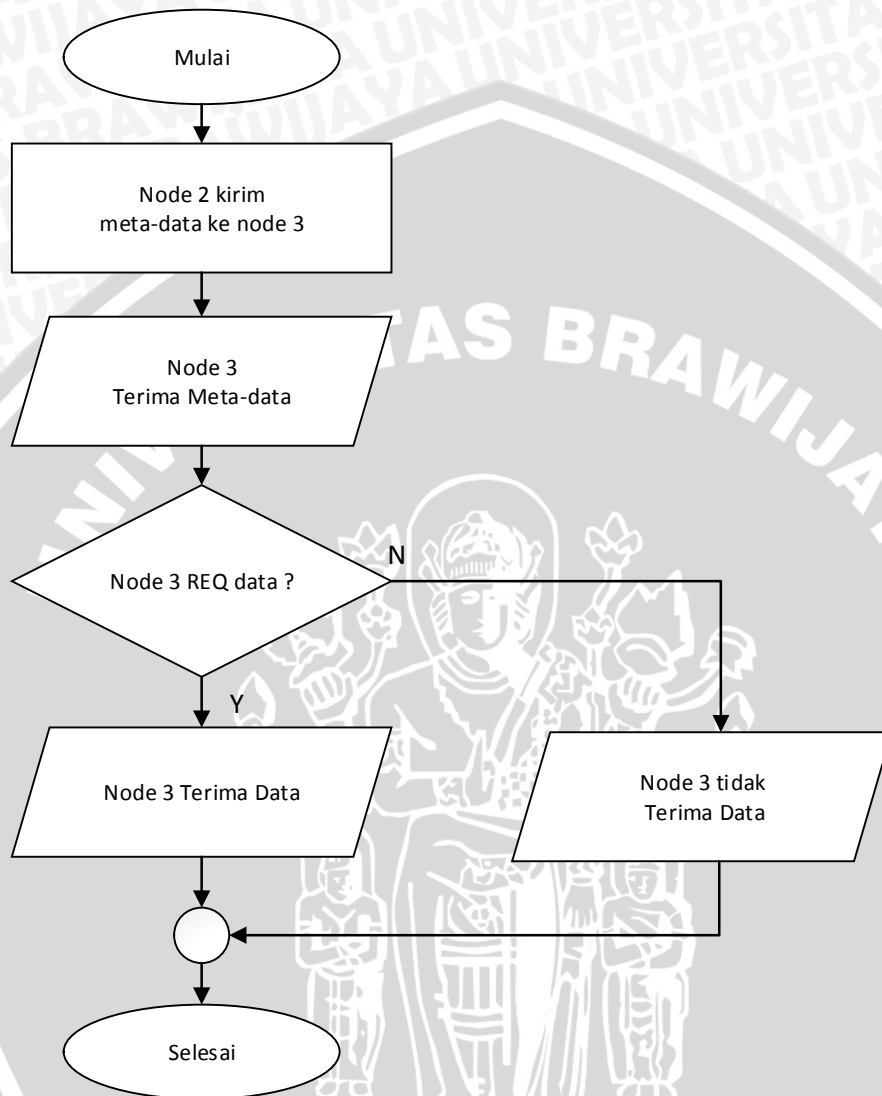


Gambar 5.3 Flow Chart Pengiriman Data node 1 ke node 2

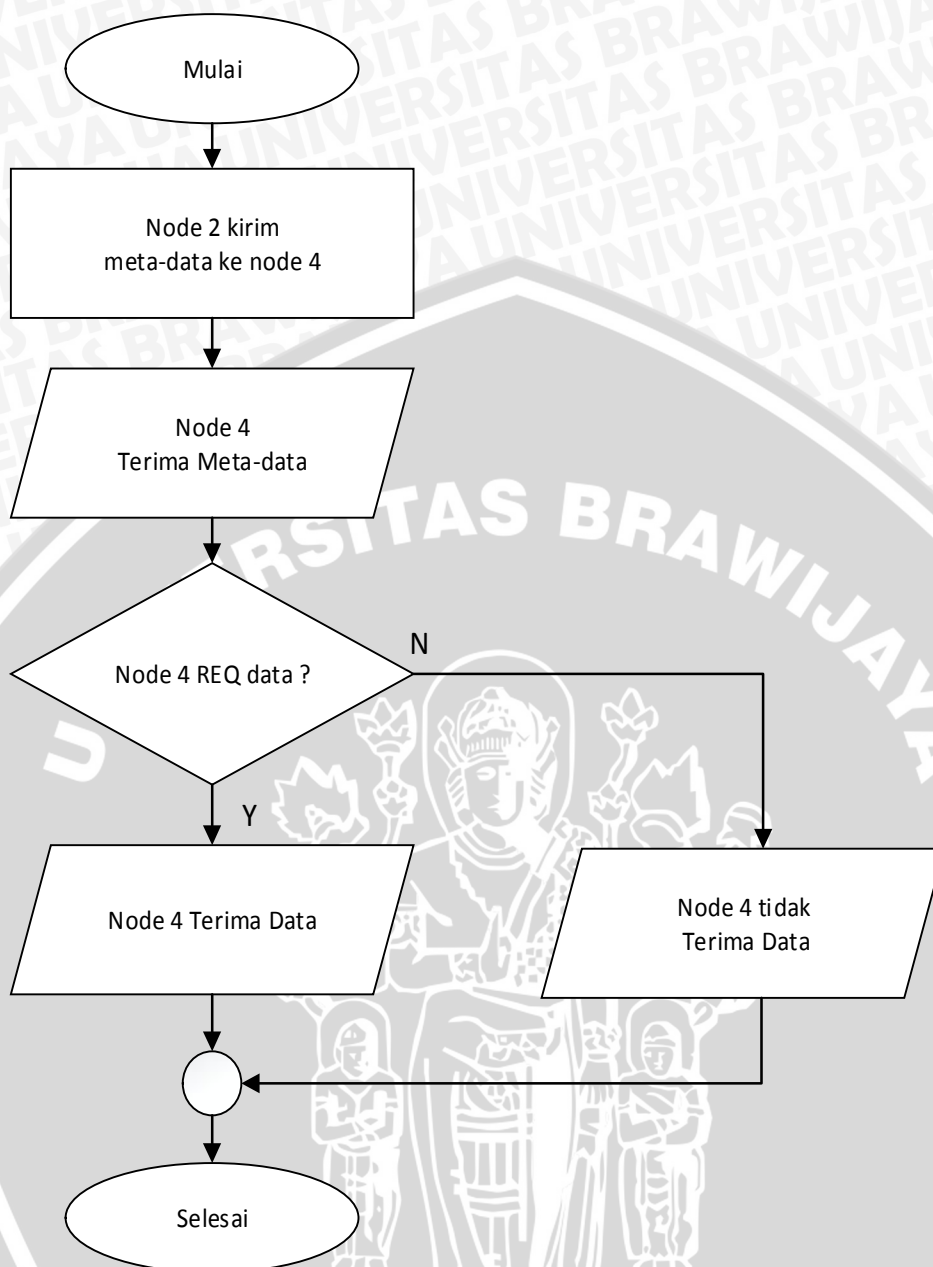
#### 5.1.2.2 Pengujian Pengiriman data node 2 ke node 3, 4, 5

Pengujian pengiriman data dari node 2 ke node 3, 4 dan 5 dilakukan secara bergantian. Node 2 akan melanjutkan proses pengiriman meta-data dengan mengirimkan pesan ADV ke node 3, 4, dan 5. Setelah node 3, 4 dan 5 menerima

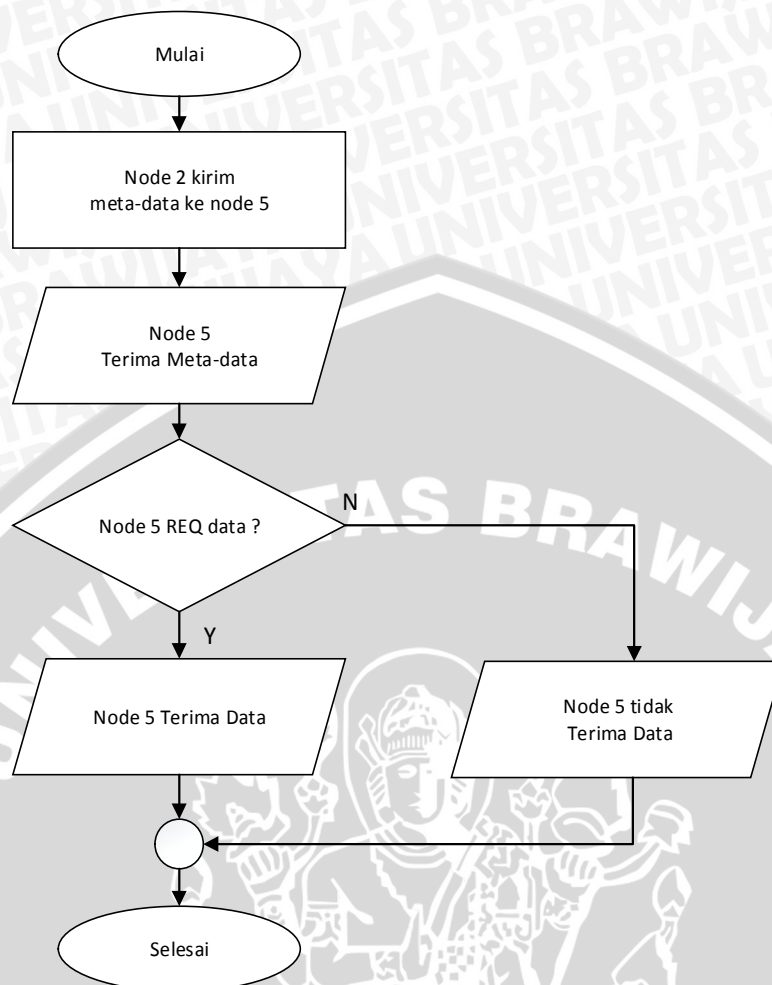
meta-data dari node 2, node 3 dan 4 akan mengirimkan pesan REQ ke node , sedangkan node 5 tidak melakukan request data. Node 2 akan mengirimkan DATA hanya ke node 3 dan 4. Berikut pada Gambar 5.4, 5.5, 5.6 merupakan flowchart pengiriman data node 2 ke node 3, 4 dan 5.



**Gambar 5.4 Flow Chart Pengiriman Data node 2 ke node 3**



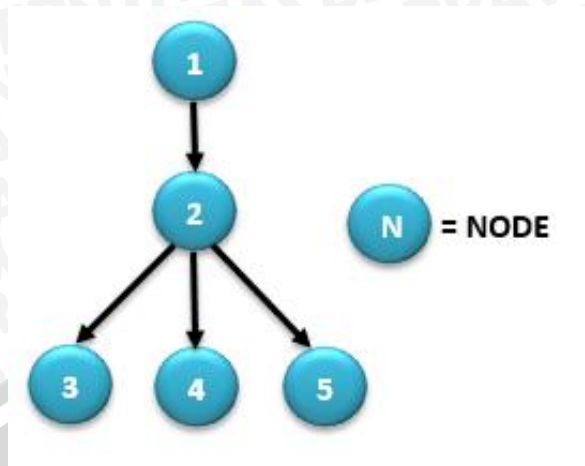
**Gambar 5.5 Flowchart Pengiriman Data node 2 ke node 4**



**Gambar 5.6 Flowchart Pengiriman Data node 2 ke node 5**

### 5.1.3 Perancangan Topologi SPIN

Perancangan topologi yang akan di implementasikan pada protokol routing SPIN adalah topologi tree. Node a diletakan dipuncak hirarki karena node a berfungsi mengirim hasil proses data dari *input* menuju ke node b dengan menggunakan proses negosiasi. Node b sebagai node penerima data yang dikirimkan oleh node a, kemudian node b akan menentukan data yang ingin diterima. Node b juga sebagai node pengirim data ke node c, d dan e sehingga topologi yang digunakan adalah topologi yang ditunjukkan pada Gambar 5.7:



Gambar 5.7 Perancangan Topologi Node

## 5.2 Implementasi

Pada bab ini membahas mengenai tahapan implementasi routing protokol SPIN pada wireless sensor network menggunakan media komunikasi RF yang telah didapatkan dari analisis kebutuhan dan proses perancangan perangkat lunak dan perangkat keras. Implementasi terdiri atas penjelasan lingkungan implementasi, batasan implementasi, dan implementasi sistem.

### 5.2.1 Lingkungan Implementasi

Lingkungan implemetasi dari proses routing protokol SPIN pada *wireless sensor network* meliputi spesifikasi lingkungan perangkat keras (*hardware*) dan spesifikasi lingkungan perangkat lunak (*software*) dan bahasa pemograman yang digunakan.

#### 5.2.1.1 Spesifikasi Lingkungan Perangkat Keras (*Hardware*)

Lingkungan perangkat keras yang digunakan dalam mengimplementasikan routing protokol Spin pada wireless sensor network yaitu arduino nano, NRF24L01, komputer/PC.

#### 5.2.1.2 Spesifikasi Lingkungan Perangkat Lunak (*Software*)

Lingkungan perangkat lunak yang digunakan dalam mengimplementasikan routing protokol Spin pada *wireless sensor network* yaitu arduino IDE sebagai media pembuatan program.

#### 5.2.1.3 Bahasa Pemograman

Bahasa pemograman yang digunakan dalam mengimplementasikan routing protokol Spin pada *wireless sensor network* yaitu bahasa pemograman c pada arduino.

### 5.2.2 Batasan Implementasi

Beberapa batasan dalam implementasi routing protokol Spin pada wireless sensor network sebagai berikut:

1. Algoritma Spin diimplementasikan untuk menerapkan routing protokol SPIN pada *wireless sensor network* menggunakan media komunikasi RF
2. Pengiriman dan penerimaan data melalui wireless sensor network
3. Input data berasal dari user
4. Node sumber mengirimkan meta-data berukuran 1 byte yang berisi meta-data suhu dan Data aktual suhu sebesar 4 byte.

### 5.2.3 Implementasi Perangkat Keras SPIN pada WSN

Implementasi perangkat keras SPIN pada WSN dilakukan dengan menggabungkan semua perangkat keras (hardware) sesuai dengan perancangan yang telah dilakukan sebelumnya. Dari hasil perancangan, node SPIN pada WSN terdiri dari rangkaian mikrokontroler dengan modul wireless. Mikrokontroler yang digunakan adalah Arduino Nano, dan modul wireless yang digunakan adalah NRF24L01. Implementasi node SPIN pada WSN ini dapat dilihat pada Gambar 5.8:



Gambar 5.8 Implementasi Perangkat Keras SPIN pada WSN

### 5.2.4 Implementasi Perangkat Lunak SPIN pada WSN

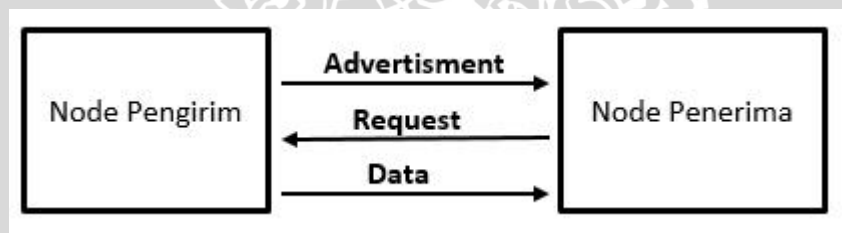
Implementasi perangkat lunak SPIN pada WSN dilakukan melalui pemrograman pada Arduino IDE, dibutuhkan beberapa library agar sistem dapat berfungsi. Untuk dapat menggunakan modul wireless NRF24L01 dibutuhkan library SPI.h, Mirf.h, nRF24L01.h, dan MirfHardwareSpiDriver.h seperti yang ditunjukkan Gambar 5.9:



**Gambar 5.9 Library yang Dibutuhkan pada Implementasi routing SPIN**

### 5.2.5 Implementasi Komunikasi Data antar Node

Pada implementasi komunikasi data, hal yang di implementasikan adalah proses komunikasi data antar node yaitu node pengirim dan node penerima. Data yang dikirimkan dari node pengirim adalah data yang diinputkan dari arduino IDE. Pada tahap ini node pengirim akan melakukan proses negosiasi dengan node penerima sebelum melakukan pengiriman data seperti pada Gambar 5.10:



**Gambar 5.10 Diagram Blok Komunikasi Data**

Dalam proses pengiriman data, terbagi menjadi beberapa sesi pengiriman yang terjadi antara node pengirim dan node penerima yaitu:

1. Node pengirim akan mengirimkan pesan ADV ke node penerima yang berisi meta-data, yaitu berupa gambaran dari sebuah data aktual yang nantinya akan dikirimkan ke node penerima.
2. Apabila node penerima menginginkan data tersebut, maka node penerima akan membalas dengan mengirimkan pesan REQUEST.
3. Kemudian node pengirim akan mengirimkan pesan DATA yang diinginkan node penerima, yaitu berupa data aktual atau data lengkap dari meta-data yang dikirim oleh node pengirim.

Setelah melengkapi modul library yang dibutuhkan untuk modul wireless, implementasi sistem dilanjutkan dengan format pengiriman data sesuai dengan algoritma SPIN. Berikut Tabel 5.2 merupakan tabel pembagian waktu pengiriman data node 1 – node 5.

**Tabel 5.2 Pembagian Waktu Pengiriman Data**

| Pembagian Waktu Pengiriman Data                                                                                                                                                                                                          |                                                                                                                              |                                                                                                                                                                                                                          |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 0 – 5 ( <i>millisecond</i> )                                                                                                                                                                                                             | 5,5 ( <i>ms</i> ) – 1,1 ( <i>second</i> )                                                                                    | 1,1 <i>second</i> – selesai                                                                                                                                                                                              |
| <ul style="list-style-type: none"> <li>- Node 1 mengirim ADV ke node 2</li> <li>- Node 2 mengirim REQ ke node 1</li> <li>- Node 2 menerima DATA dari node 1</li> <li>- Node 2 mengirim ADV berupa meta-data ke node 3,4 dan 5</li> </ul> | <ul style="list-style-type: none"> <li>- Node 3 mengirim REQ ke node 2</li> <li>- Node 2 menerima REQ dari node 3</li> </ul> | <ul style="list-style-type: none"> <li>- Node 4 mengirim REQ ke node 2</li> <li>- Node 2 menerima REQ dari node 4</li> <li>- Node 2 mengirim DATA ke node 3 dan 4</li> <li>- Node 5 tidak ingin menerima data</li> </ul> |

Berikut Gambar 5.11 – 5.25 merupakan potongan script pengiriman data dari node 1 sampai node 5.



```

node1 | Arduino 1.0.6
File Edit Sketch Tools Help

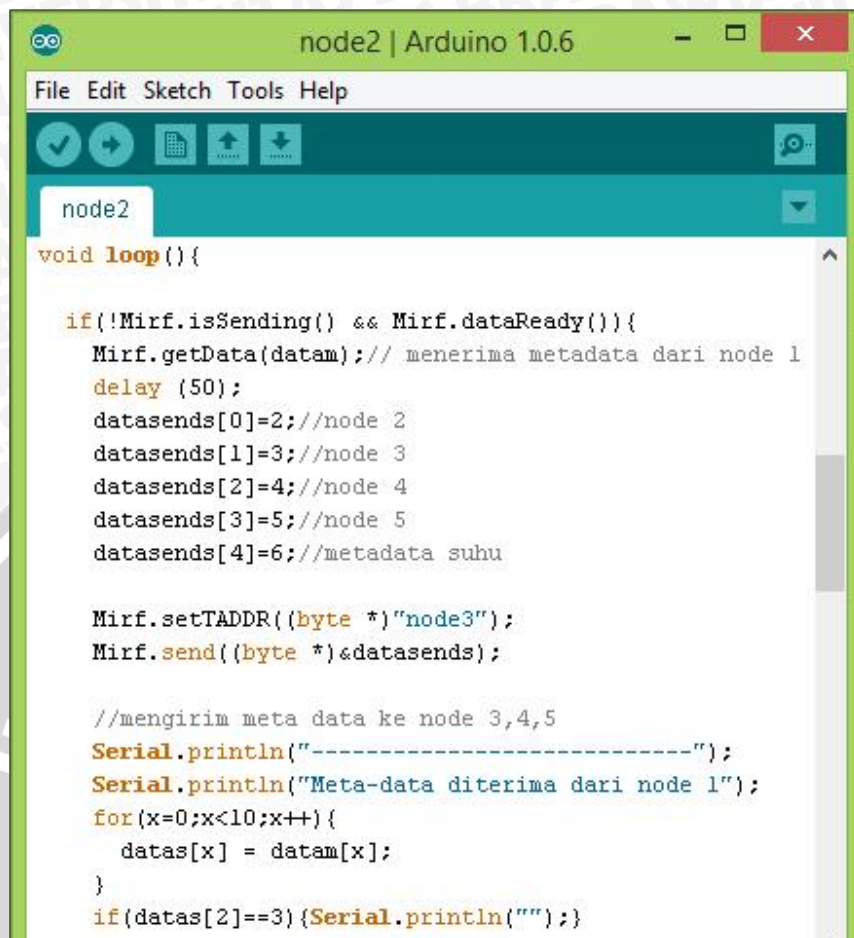
node1

void loop(){

  delay(10000);
  datasends[0]=1;//node 1
  datasends[1]=2;//node 2
  datasends[2]=3;//metadata suhu

  Mirf.setTADDR((byte *)"node2");
  Mirf.send((byte *)&datasends);
  Serial.println("-----");
  Serial.println("Kirim metadata suhu");
}
  
```

**Gambar 5.11 Node 1 mengirim ADV meta-data ke node 2**



```

void loop(){

  if(!Mirf.isSending() && Mirf.dataReady()){
    Mirf.getData(datam); // menerima metadata dari node 1
    delay (50);
    datasends[0]=2; //node 2
    datasends[1]=3; //node 3
    datasends[2]=4; //node 4
    datasends[3]=5; //node 5
    datasends[4]=6; //metadata suhu

    Mirf.setTADDR((byte *)"node3");
    Mirf.send((byte *)&datasends);

    //mengirim meta data ke node 3,4,5
    Serial.println("-----");
    Serial.println("Meta-data diterima dari node 1");
    for(x=0;x<10;x++){
      datas[x] = datam[x];
    }
    if(datas[2]==3){Serial.println("");}
  }
}

```

Gambar 5.12 Node 2 menerima meta-data dari node 1



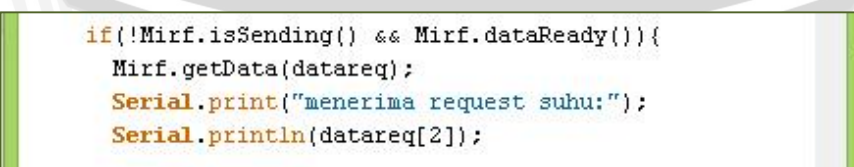
```

if(datas[1]==2 && datas[2]==3){
  Serial.println("Siap kirim Request ke node 1");
  datasendc[0]=datam[1]; //node tujuan node 2
  datasendc[1]=datam[0]; // node asal node 1
  datasendc[2]=1; //minta suhu

  Mirf.setTADDR((byte *)"node1");
  Mirf.send((byte *)&datasendc);
  Serial.println("Request dikirim ke node 1");
  Serial.println("-----");
}

```

Gambar 5.13 Node 2 melakukan REQ ke node 1



```

if(!Mirf.isSending() && Mirf.dataReady()){
  Mirf.getData(datareq);
  Serial.print("menerima request suhu:");
  Serial.println(datareq[2]);
}

```

Gambar 5.14 Node 1 menerima REQ dari node 2

```
if(datareq[2]==1 && datareq[1]==1){
    Serial.println("siap mengirim data suhu");

    datasendsuhu[0]=datasends[0];
    datasendsuhu[1]=datasends[1];
    for(int x=0;x<datasuhu.length();x++){
        datasendsuhu[x+2]=datasuhu[x];
    }
    delay(50);
    Mirf.setTADDR((byte *)"node2");
    Mirf.send((byte *)&datasendsuhu);
    Serial.println("Data berhasil dikirim");
}
```

**Gambar 5.15 Node 1 mengirim DATA ke node 2**

```
if(!Mirf.isSending() && Mirf.dataReady()){
    Mirf.getData(data);
    Serial.println("Terima data node 1");
    Serial.print(data[2]);
    Serial.println(data[3]);
    Serial.println("-----");
}
```

**Gambar 5.16 Node 2 menerima DATA dari node 1**

Pada Gambar 5.11 - 5.16 menunjukkan proses pengiriman data node 1 ke node 2, pertama node 1 mengirimkan pesan ADV ke node 2 yang berisi meta-data yaitu meta-data suhu. Kemudian node 2 menerima meta-data dan mengirimkan pesan REQ ke node 1, setelah itu node 1 menerima pesan request dan mengirimkan DATA yang diinginkan node 2.

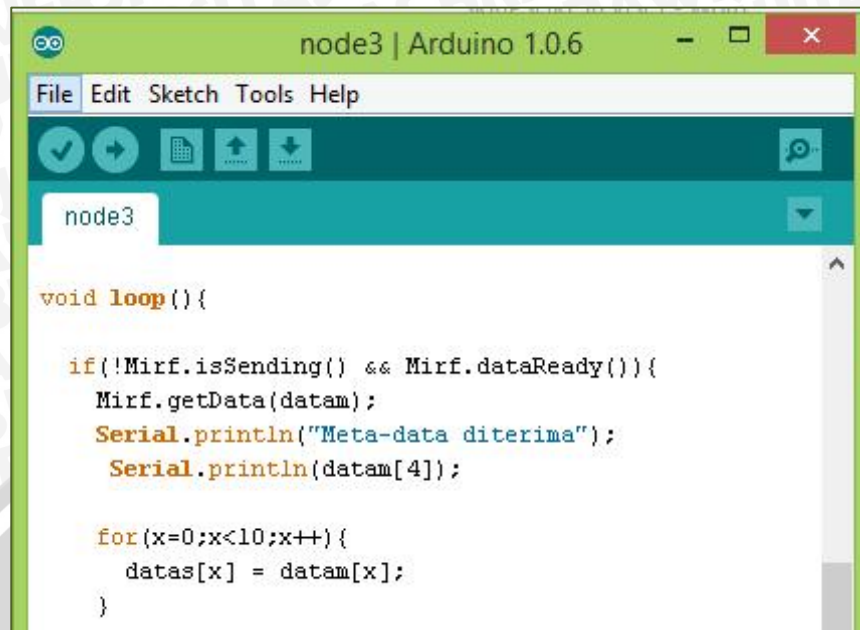
Setelah node 1 selesai mengirimkan data, node 2 kemudian meneruskan pesan ADV ke node 3, 4 dan 5. Berikut pada Gambar 5.17 – 5.25 merupakan script pengiriman data node 2 ke node 3, 4 dan 5.

Pada implementasi ini node 5 tidak melakukan request, namun tetap menerima pesan ADV yang berisi meta-data suhu yang dikirimkan oleh node 2. Seperti yang ditujukan pada Gambar 5.22.

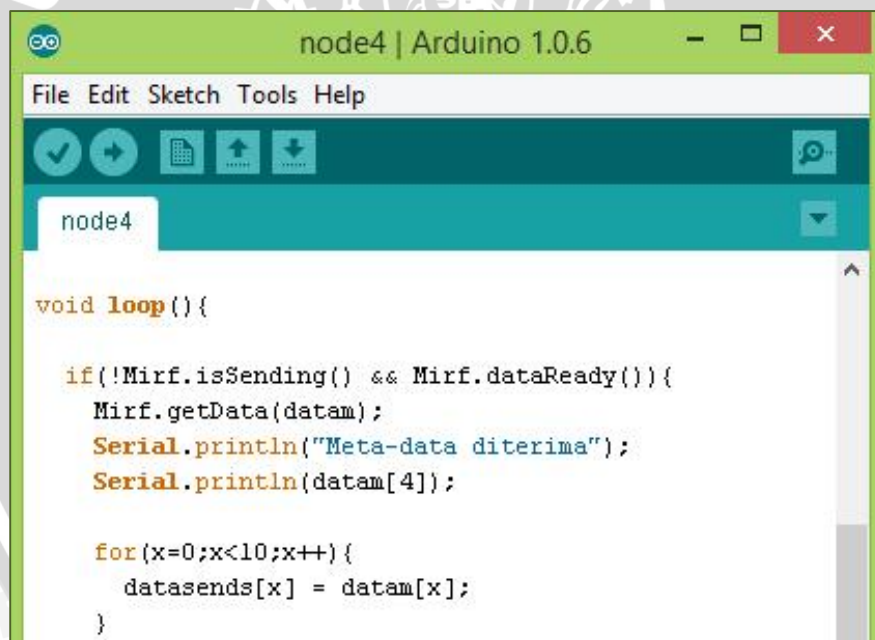
```
delay (50);
datasends[0]=2;//node 2
datasends[1]=3;//node 3
datasends[2]=4;//node 4
datasends[3]=5;//node 5
datasends[4]=6;//metadada suhu

Mirf.setTADDR((byte *)"node3");
Mirf.send((byte *)&datasends);
Mirf.setTADDR((byte *)"node4");
Mirf.send((byte *)&datasends);
Mirf.setTADDR((byte *)"node5");
Mirf.send((byte *)&datasends);
Serial.println("Meta-data dikirim ke node 3,4,5");
```

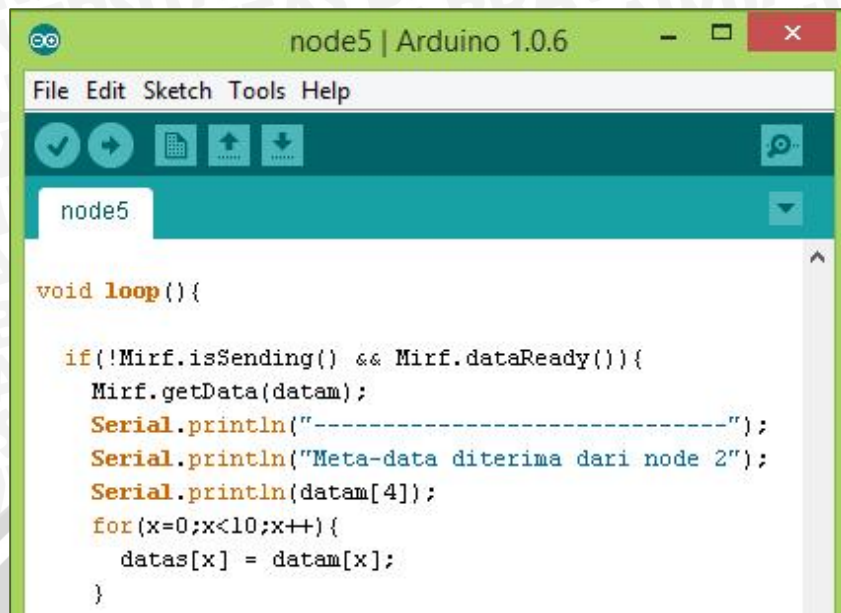
**Gambar 5.17 Node 2 mengirim pesan ADV meta-data ke node 3,4,5**



Gambar 5.18 Node 3 menerima meta-data dari node 2



Gambar 5.19 Node 4 menerima meta-data dari node 2



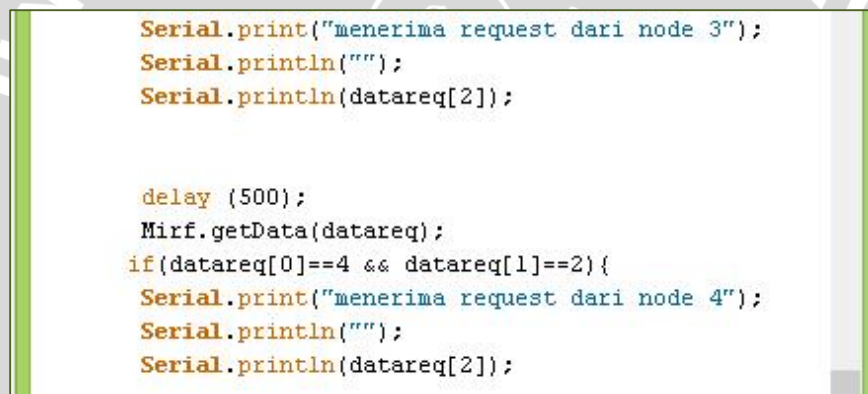
```

void loop(){

  if(!Mirf.isSending() && Mirf.dataReady()){
    Mirf.getData(datam);
    Serial.println("-----");
    Serial.println("Meta-data diterima dari node 2");
    Serial.println(datam[4]);
    for(x=0;x<10;x++){
      datas[x] = datam[x];
    }
  }
}

```

Gambar 5.20 Node 5 menerima meta-data dari node 2



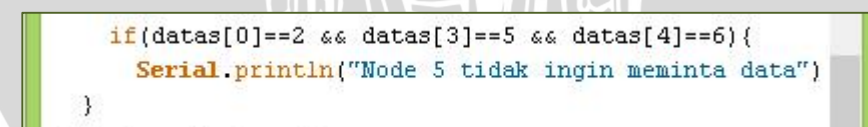
```

Serial.print("menerima request dari node 3");
Serial.println("");
Serial.println(datareq[2]);

delay (500);
Mirf.getData(datareq);
if(datareq[0]==4 && datareq[1]==2){
  Serial.print("menerima request dari node 4");
  Serial.println("");
  Serial.println(datareq[2]);
}

```

Gambar 5.21 Node 2 menerima REQ dari node 3 dan 4



```

if(datas[0]==2 && datas[3]==5 && datas[4]==6){
  Serial.println("Node 5 tidak ingin meminta data")
}

```

Gambar 5.22 Node 5 tidak meminta data



```

Serial.println("kirim data suhu ");
Mirf.setTADDR((byte *)"node3");
Mirf.send((byte *)&data);
Mirf.setTADDR((byte *)"node4");
Mirf.send((byte *)&data);
}}
}}
}

```

Gambar 5.23 Node 2 mengirimkan DATA ke node 4 dan 5

```
if(!Mirf.isSending() && Mirf.dataReady()){
  Mirf.getData(data);
  Serial.println("Terima data");
  Serial.print(data[2]);
  Serial.println(data[3]);
  Serial.println("-----");
}
```

Gambar 5.24 Node 3 menerima DATA dari node 2

```
if(!Mirf.isSending() && Mirf.dataReady()){
  Mirf.getData(data);
  Serial.println("Terima data");
  Serial.print(data[2]);
  Serial.println(data[3]);
  Serial.println("-----");
}
```

Gambar 5.25 Node 4 menerima DATA dari node 2



## BAB 6 PENGUJIAN DAN HASIL

Bab ini membahas mengenai tahapan pengujian dan analisis dari sistem yang sudah dibuat. Proses pengujian terdapat tiga pengujian yaitu pengujian packet delivery ratio pada node tujuan, pengujian real time sistem, dan pengujian packet loss data yang terjadi selama komunikasi.

Pengujian packet delivery ratio pada node tujuan digunakan untuk mengetahui apakah data yang dikirimkan sudah sampai ke node tujuan sesuai data yang dikirim. Pengujian real time sistem digunakan untuk mengetahui apakah waktu dari proses pengiriman data sudah sesuai dengan waktu yang sudah ditentukan oleh user. Dan pengujian packet loss data yang terjadi selama komunikasi digunakan untuk mengetahui apakah data yang dikirimkan oleh node pengirim hilang atau tidak lengkap saat dikirimkan ke node tujuan.

### 6.1 Pengujian Node Sensor ke PC (Komputer)

Pengujian node sensor ke PC dilakukan dengan menghubungkan node sensor yang terdiri dari arduino nano dan nrf24l01 ke komputer melalui kabel USB, seperti yang ditunjukkan Gambar 6.1. Pengujian ini bertujuan untuk memastikan bahwa setiap node sensor yang digunakan tidak bermasalah. Pengujian ini dilakukan dengan menguji satu persatu lima buah node sensor yang akan digunakan.

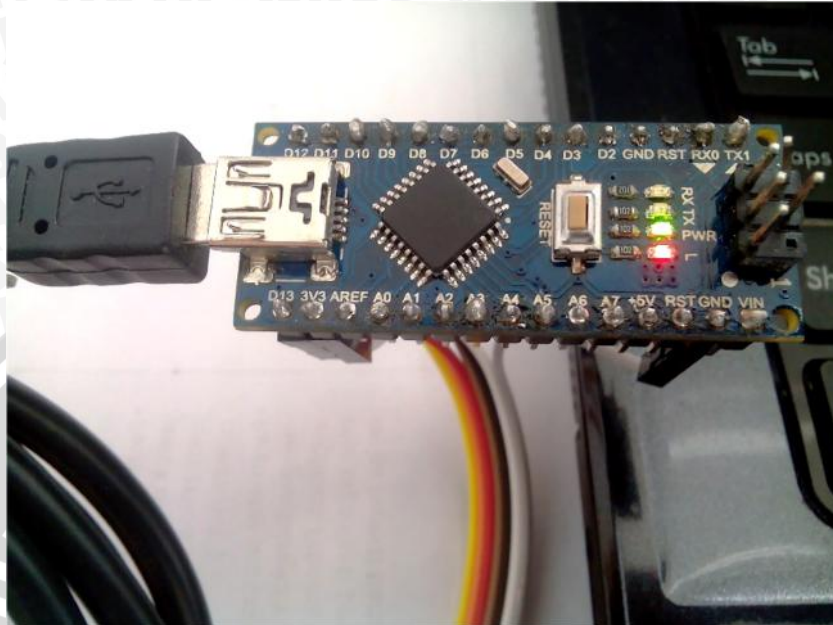


**Gambar 6.1 Hubungan Node Sensor dengan PC**

Apabila node sensor sudah terhubung ke PC dan PC telah dinyalakan, ada dua indikator yang menyatakan bahwa node sensor tidak bermasalah, seperti pada Gambar 6.2.

1. Indikator pertama berupa lampu kecil berlabel PWR/Power yang akan menyala

2. Indikator kedua berupa lampu kecil berlabel L (LED) akan menyala, led ini terhubung ke pin digital 13, ketika pin diset bernilai HIGH, maka LED menyala, dan ketika pin diset bernilai LOW, maka LED padam.



**Gambar 6.2 Dua Indikator yang menyatakan Node Sensor tidak bermasalah**



**Gambar 6.3 Hubungan Keseluruhan Sistem**

Gambar 6.3 merupakan hubungan keseluruhan dari sistem selama pengujian berlangsung.

## 6.2 Pengujian Fungsional

Pengujian Fungsional dilakukan untuk memastikan semua kebutuhan dalam sistem terpenuhi sesuai dengan fungsi yang akan dilakukan oleh sistem. Pengujian fungsional ini akan mengacu pada hasil dari proses yang dilakukan oleh sistem.

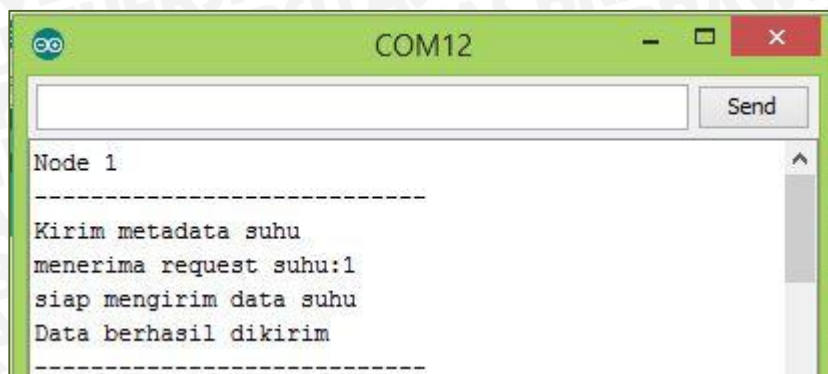
Pengujian yang akan dilakukan adalah pengiriman pesan ADV, REQ, dan DATA yang diimplementasikan pada node sensor. Terdapat lima node sensor yang akan digunakan dalam pengujian ini sesuai dengan topologi yang telah dibuat sebelumnya. Tujuan pengujian fungsional untuk mengetahui apakah data yang dikirimkan sudah benar dan sesuai dengan data yang dibutuhkan dan berjalan sesuai dengan algoritma SPIN.

### 6.2.1 Pengujian Pengiriman Data Node 1 ke node 2

Pengujian pengiriman data dari node 1 ke node 2 yang akan diuji adalah proses negosiasi pengiriman data, yang pertama adalah pengiriman meta-data ke node 2 kemudian pengiriman data. Pada Tabel 6.1 dibawah ini merupakan tabel hasil pengujian pengiriman paket data node 1 ke node 2 selama 10 kali pengiriman.

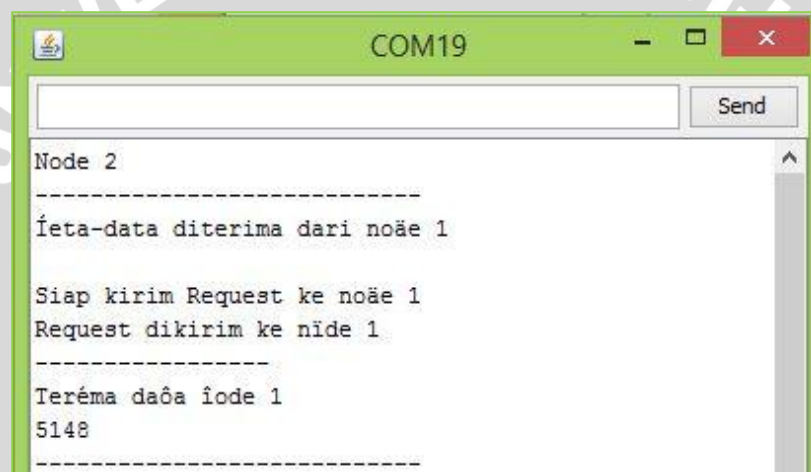
**Tabel 6.1 Hasil Pengujian Pengiriman Paket Data Node 1 ke Node 2**

| PENGIRIMAN ke- | Pengiriman Meta-data | Pengiriman Data | Hasil Pengujian |
|----------------|----------------------|-----------------|-----------------|
| 1              | 6                    | 5148            | OK              |
| 2              | 6                    | 5148            | OK              |
| 3              | 6                    | 5148            | OK              |
| 4              | 6                    | 5148            | OK              |
| 5              | 6                    | 5148            | OK              |
| 6              | 6                    | 5148            | OK              |
| 7              | 6                    | 5148            | OK              |
| 8              | 6                    | 5148            | OK              |
| 9              | 6                    | 5148            | OK              |
| 10             | 6                    | 5148            | OK              |



**Gambar 6.4 Hasil Pengujian Pengiriman Paket Data di Node 1**

Gambar 6.4 merupakan hasil pengujian pengiriman meta data oleh node 1 ke node 2.

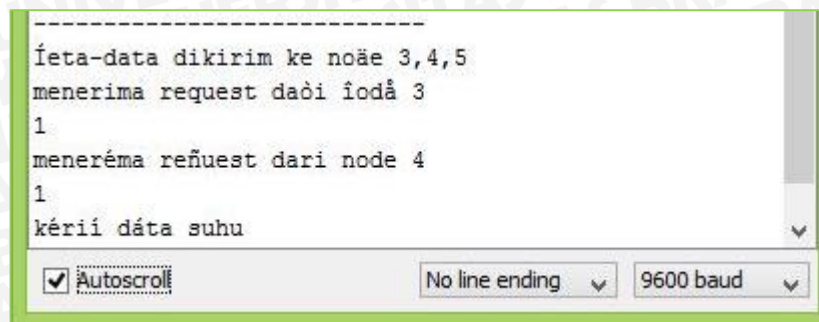


**Gambar 6.5 Hasil Pengujian Pengiriman Paket Data di Node 2**

Gambar 6.5 merupakan hasil pengujian di node 2, menunjukkan meta-data yang dikirim oleh node 1 berhasil dikirim dan data diterima oleh node 2.

### 6.2.2 Pengujian Pengiriman data node 2 ke node 3, 4, 5

Pengujian pengiriman data dari node 2 ke node 3, 4 dan 5 dilakukan secara bergantian. Node 2 akan melanjutkan proses pengiriman meta-data dengan mengirimkan pesan ADV ke node 3, 4, dan 5. Setelah node 3, 4 dan 5 menerima meta-data dari node 2, node 3 dan 4 akan mengirimkan pesan REQ ke node , sedangkan node 5 tidak melakukan request data. Node 2 akan mengirimkan DATA hanya ke node 3 dan 4. Berikut pada Tabel 6.2 , 6.4 dan 6.5 merupakan pengujian pengiriman paket data node 2 ke node 3, 4 dan 5 selama 10 kali pengiriman.

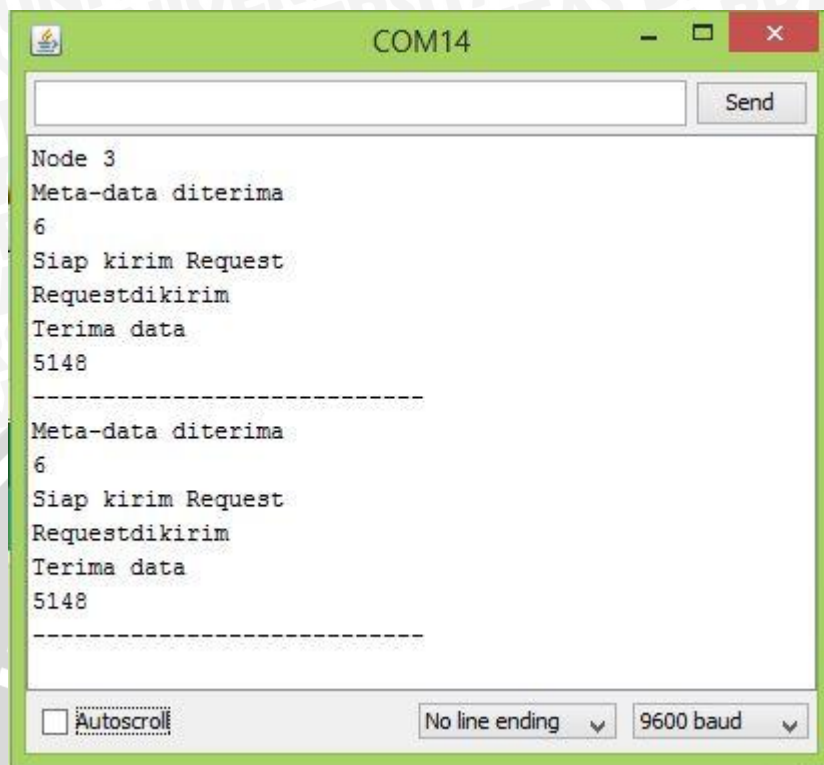


**Gambar 6.6 Hasil Pengujian Pengiriman Paket Data di Node 2**

Gambar 6.6 merupakan hasil pengujian di node 2, menunjukkan pengiriman meta-data yang dikirim ke node 3, 4, 5 dan proses REQ dari node 3 dan 4.

**Tabel 6.2 Pengujian Pengiriman Paket Data Node 2 ke Node 3**

| PENGIRIMAN ke- | Pengiriman Meta-data | Pengiriman Data | Hasil Pengujian |
|----------------|----------------------|-----------------|-----------------|
| 1              | 6                    | 5148            | OK              |
| 2              | 6                    | 5148            | OK              |
| 3              | 6                    | 5148            | OK              |
| 4              | 6                    | 5148            | OK              |
| 5              | 6                    | 5148            | OK              |
| 6              | 6                    | 5148            | OK              |
| 7              | 6                    | 5148            | OK              |
| 8              | 6                    | 5148            | OK              |
| 9              | 6                    | 5148            | OK              |
| 10             | 6                    | 5148            | OK              |

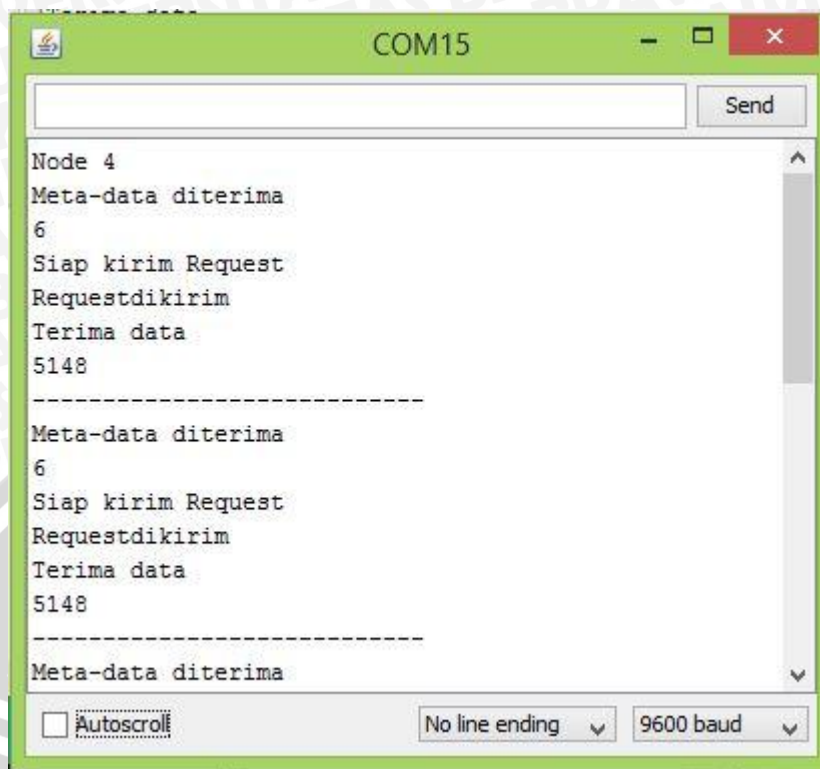


**Gambar 6.7 Hasil Pengujian Pengiriman Paket Data di Node 3**

Gambar 6.7 merupakan hasil pengujian di node 3, menunjukkan meta-data dan DATA berhasil dikirim oleh node 2.

**Tabel 6.3 Pengujian Pengiriman Paket Data Node 2 ke Node 4**

| PENGIRIMAN ke- | Pengiriman Meta-data | Pengiriman Data | Hasil Pengujian |
|----------------|----------------------|-----------------|-----------------|
| 1              | 6                    | 5148            | OK              |
| 2              | 6                    | 5148            | OK              |
| 3              | 6                    | 5148            | OK              |
| 4              | 6                    | 5148            | OK              |
| 5              | 6                    | 5148            | OK              |
| 6              | 6                    | 5148            | OK              |
| 7              | 6                    | 5148            | OK              |
| 8              | 6                    | 5148            | OK              |
| 9              | 6                    | 5148            | OK              |
| 10             | 6                    | 5148            | OK              |

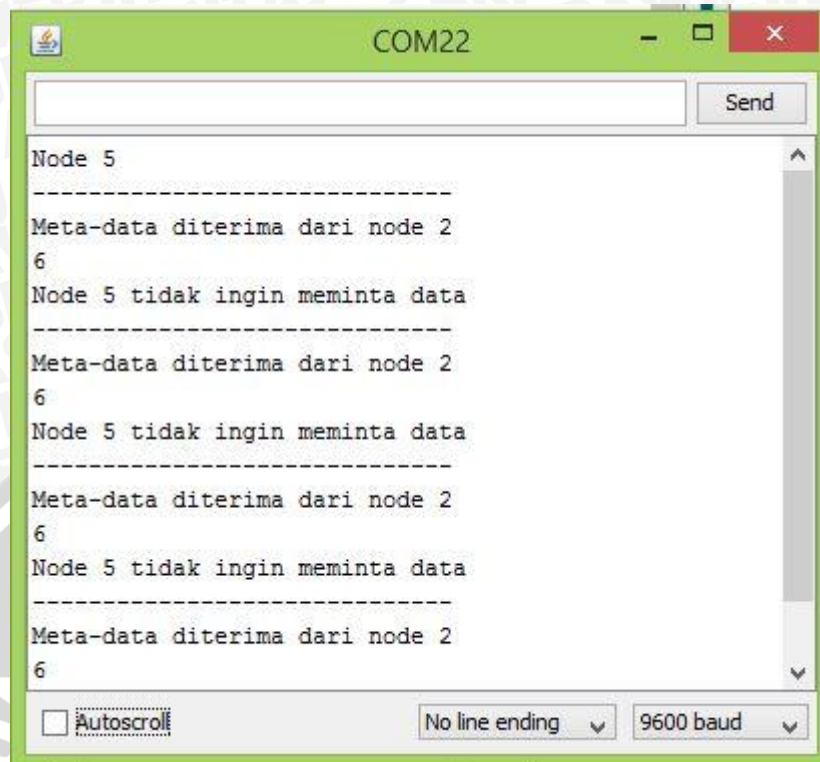


**Gambar 6.8 Hasil Pengujian Pengiriman Paket Data di Node 4**

Gambar 6.8 merupakan hasil pengujian di node 4, menunjukkan meta-data dan DATA berhasil dikirim oleh node 2.

**Tabel 6.4 Pengujian Pengiriman Paket Data Node 2 ke Node 5**

| PENGIRIMAN ke- | Pengiriman Meta-data | Pengiriman Data |
|----------------|----------------------|-----------------|
| 1              | 6                    | Tidak REQ       |
| 2              | 6                    | Tidak REQ       |
| 3              | 6                    | Tidak REQ       |
| 4              | 6                    | Tidak REQ       |
| 5              | 6                    | Tidak REQ       |
| 6              | 6                    | Tidak REQ       |
| 7              | 6                    | Tidak REQ       |
| 8              | 6                    | Tidak REQ       |
| 9              | 6                    | Tidak REQ       |
| 10             | 6                    | Tidak REQ       |



**Gambar 6.9 Hasil Pengujian Pengiriman Paket Data di Node 5**

Gambar 6.9 merupakan hasil pengujian di node 5, menunjukkan pengiriman meta-data yang dikirim oleh node 2 berhasil, namun node 5 tidak ingin melakukan request DATA.

### 6.3 Pengujian Delay

Pengujian delay dilakukan dengan menguji delay milis pengiriman node 1 ke node 2. Pengujian ini berfungsi untuk mengetahui delay pada proses routing SPIN. Kemudian hasil dari pengujian akan digambarkan kedalam sebuah grafik. Adapun pengujian delay masing-masing di uji selama 20 kali pengiriman yang akan dilakukan sebagai berikut:

1. Menghitung delay di node 1 saat melakukan pengiriman meta-data sampai mendapatkan request dari node 2.
2. Menghitung delay di node 2 saat mengirimkan request sampai mendapatkan data dari node 1.

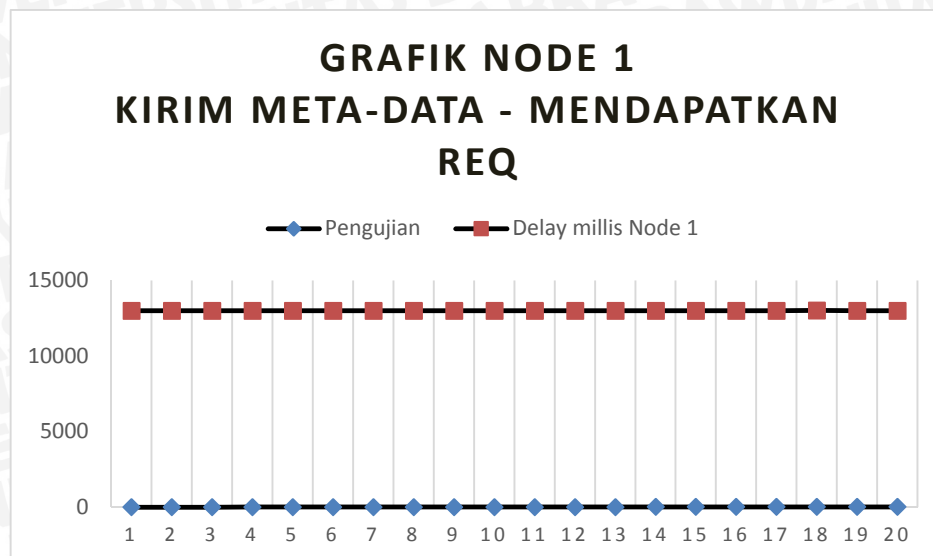
**Tabel 6.5 Pengujian Delay node 1 ke node 2**

| Pengiriman ke- | Delay node 1 ( <i>milisecond</i> ) | Delay node 2 ( <i>milisecond</i> ) |
|----------------|------------------------------------|------------------------------------|
|                | Kirim meta-data – Mendapatkan REQ  | Kirim REQ – Mendapatkan Data       |
| 1              | 13002                              | 13006                              |
| 2              | 13001                              | 13009                              |

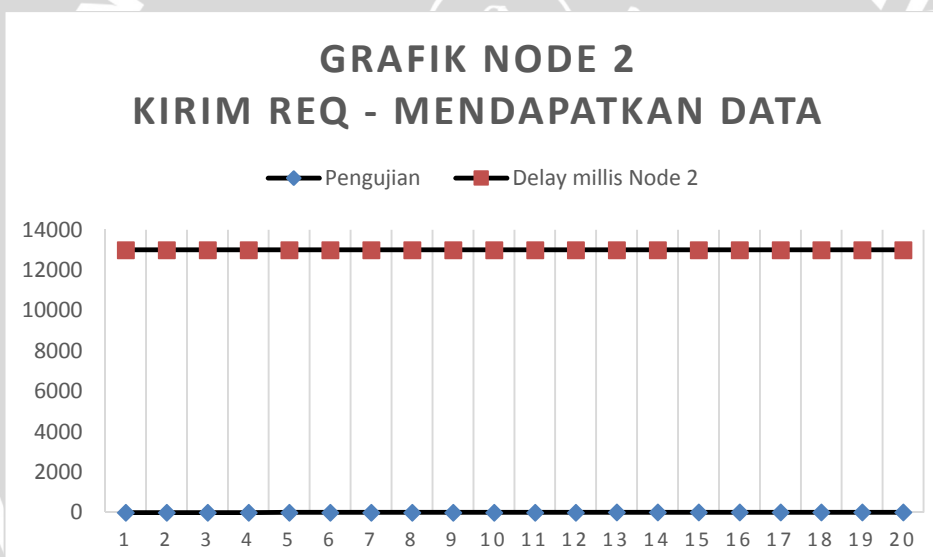
|    |       |       |
|----|-------|-------|
| 3  | 13002 | 13009 |
| 4  | 13002 | 13008 |
| 5  | 13002 | 13009 |
| 6  | 13002 | 13009 |
| 7  | 13002 | 13009 |
| 8  | 13002 | 13008 |
| 9  | 13001 | 13008 |
| 10 | 13002 | 13008 |
| 11 | 13002 | 13009 |
| 12 | 13002 | 13008 |
| 13 | 13001 | 13009 |
| 14 | 13002 | 13008 |
| 15 | 13002 | 13009 |
| 16 | 13002 | 13008 |
| 17 | 13002 | 13010 |
| 18 | 13002 | 13007 |
| 19 | 13002 | 13009 |
| 20 | 13001 | 13008 |

Pada tabel 6.5 menunjukkan delay millis di node 1 dan node 2. Dilihat dari tabel tersebut delay pada node 1 dan node 2 tidak jauh berbeda selama dilakukan 20 kali pengiriman. Ini menunjukkan bahwa pengiriman paket data lancar dan stabil. Delay millis node 1 merupakan lama waktu node 1 mengirimkan meta-data sampai mendapatkan request dari node 2, didapatkan rata-rata 13002 ms (*millisecond*). Sedangkan delay millis di node 2 merupakan lama waktu node 2 mengirimkan request sampai mendapatkan data dari node 1 dan didapatkan delay rata-rata 13008 ms (*millisecond*). Delay proses pengiriman data dari node 1 ke node 2 cukup lama, ini dikarenakan node 1 harus menunggu proses pengiriman data yang dilakukan oleh node 2 ke node 3, 4, 5 sampai selesai. Setelah itu node 1 dapat melakukan pengiriman data kembali

Gambar 6.10 dan 6.11 berikut merupakan grafik node 1 pengiriman meta-data sampai mendapatkan request dari node 2 dan pengiriman request sampai mendapatkan data dari node 1. Grafik menunjukkan bahwa proses pengiriman data berjalan dengan stabil.



Gambar 6.10 Grafik Hasil Pengujian Delay di Node 1



Gambar 6.11 Grafik Hasil Pengujian Delay di Node 2

## BAB 7 PENUTUP

### 7.1 Kesimpulan

Berdasarkan hasil pengujian terhadap pengiriman data yang telah dilakukan dalam implementasi routing protokol SPIN pada WSN dapat ditarik kesimpulan sebagai berikut:

1. Penerapan protokol routing SPIN diimplementasikan pada lima buah node Arduino Nano dimana pengiriman datanya menggunakan *radio frequency* NRF24L01 dan diimplementasikan pada topologi tree. Node 1 akan mengirimkan pesan ADV dalam bentuk meta-data ke node 2 sampai node 2 mendapatkan DATA, setelah itu node 2 akan mengirimkan meta-data ke node 3, 4 dan 5.
2. Proses pertukaran informasi dengan menggunakan protokol routing SPIN adalah dengan melakukan negosiasi terlebih dahulu, pertama node 1 akan mengirimkan pesan ADV berupa meta-data ke node 2 berukuran 1 *byte*, node 2 akan menerima meta-data dan akan melakukan request apabila node 2 menginginkan data tersebut. Setelah itu node 1 akan mengirimkan DATA ke node 2 dengan ukuran 4 *byte*. Node 2 akan meneruskan data tersebut dengan mengirimkan pesan ADV berupa meta-data secara *unicast* ke node 3, 4, 5.
3. Kinerja protokol routing SPIN yang diterapkan pada pengiriman data antar node cukup bagus, dilihat dari hasil pengujian didapatkan delay rata-rata 13 detik (13000 *milisecond*) untuk melakukan proses pengiriman data antara node 1 dan node 2. Delay proses pengiriman data dari node 1 ke node 2 cukup lama, ini dikarenakan node 1 harus menunggu proses pengiriman data yang dilakukan oleh node 2 ke node 3, 4, 5 sampai selesai. Setelah itu node 1 dapat melakukan pengiriman data kembali.

### 7.2 Saran

1. Melakukan pengujian dan analisis lebih lanjut yang dilakukan dengan mengimplementasikan algoritma SPIN yang lebih spesifik, misalnya SPIN-PP, SPIN-BC, SPIN-EC, dan SPIN-RL.
2. Melakukan pengujian lebih lanjut mengenai kinerja protokol routing SPIN yang diterapkan pada *wireless sensor network* dengan menambahkan parameter pengujian yang berbeda.

## DAFTAR PUSTAKA

- Akkaya, K., & Younis, M. (2005). A Survey On Routing Protocols For Wireless Sensor Networks. *Elsevier Ad Hoc Network Journal*, 3, 325-349.
- Anisi, M. H., Abdullah, A. H., Razak, S. A., & Ngadi, M. A. (2012). *An Overview of Data Routing Approaches for Wireless Sensor Networks*. Johor, Malaysia: Department of Computer Systems and Communications.
- Aziz, T. R., Soebarto, A. A., & Akbar, S. R. (2014). *RELAY ROUTING PROTOCOL JARINGAN SENSOR NIRKABEL PADA JARINGAN*.
- Basagni, S. (1999). Distributive and Mobility-Adaptive Clustering for Multimedia Support in Multi-hop Wireless Networks. *Proceedings of Vehicular Technology Conference, VTC*, 2, 889-893.
- Basagni, S. (1999). Distributive and Mobility-Adaptive Clustering for Multimedia Support in Multi-hop. *Proceedings of Vehicular Technology Conference, VTC*, 2, 889-893.
- Departemen, T. P. (2008). *Kamus Bahasa Indonesia*. Jakarta.
- Geetu, & Juneja, S. (2012). Performance Analysis of SPIN and LEACH Routing Protocol in WSN . *International Journal Of Computational Engineering Research*, Vol. 2 (Issue. 5).
- Hassanein, H., & Luo, J. (2006). Reliable Energy Aware Routing in Wireless Sensor Network. *In Second IEEE Workshop on Dependability and Security in Sensor Networks and Systems*, 54-64.
- Jain, V., & Khan, N. A. (2014). Simulation Analysis of Directed Diffusion and SPIN Routing Protocol in Wireless Sensor Network. *IEEE*.
- L.Li, & Wu, F. (2007). Research on SPIN of wireless sensor network. *Computer and Modernization*, 3, 93-96.
- M PATTANI, K., & J CHAUHAN, P. (2015). SPIN PROTOCOL FOR WIRELESS SENSOR NETWORK . *International Journal of Advance Research in Engineering, Science & Technology(IJAREST)*, Volume 2( Issue 5).
- Mayasari, R., Munadi, R., & Sumaryo, S. (2013). ANALISIS MODIFIKASI ALGORITMA SPIN DENGAN PENAMBAHAN VC TABLE. *Tugas Akhir, Magister Elektro Komunikasi, Fakultas Teknik Elektro, Universitas Telkom*.
- Pandey, S., Nagwani, N. K., & Kumar, C. (2015). Cluster based - SPIN Routing Protocol for Wireless. *Indian Journal of Science and Technology*, Vol 8(15).
- Parvin, S., & Rahim, M. S. (2008). Routing Protocols for Wireless Sensor Networks: A Comparative Study. *International Conference on Electronics, Computer and Communication (ICECC)*.
- Patil, H. K., & A.Szygenda, S. (2013). Security for Wireless Sensor Networks using Identity-Based Cryptography. *CRC Press Taylor & Francis Group*.

- Perillo, M. A., & Heinzelman, W. B. (2005). *Wireless Sensor Network Protocol. Department of Electrical and Computer Engineering, University of Rochester.*
- Prerna, & Kumar, S. (2015). *Energy Efficient Clustering Algorithm for SPIN.* India: International Conference on Signal Processing and Integrated Network.
- Priadana, N. M., Soebroto, A. A., & Setyawan, G. E. (2014). *Desain dan Implementasi Wireless Sensor untuk Smart Home Berbasis SMS Gateway.* Malang.
- Renesse, R., & Aghvami, A. (2004). *Formal Verivication of Ad-Hoc Routing Using SPIN Model Checker. IEEE MELECON 2004.*
- Sharma, D., Sehrawat, H., & Jyoti. (2012). *Energy Affiicient M-SPIN Protocol. International Journal of Scientific & Engineering Research, 3(10).*
- Sharma, K., Mittal, N., & Rathi3, P. (2014). *International Journal of Future Generation Communication and Networking. Performance Analysis of Flooding and SPIN in Wireless Sensor Networks, 7(3), 25-36.*
- Tripathi, A., Yadav, N., & Dadhich, R. (2015). *Selection of Node Density with Cluster in SPIN for Data Centric Wireless Sensor Network.* Mumbai, India: International Conference on Technologies for Sustainable Development.
- Tripathi, A., Yadav, N., & Dadhich, R. (2015). *SPIN With Cluster for Data Centric Wireless Sensor Network . India : Fifth International Conference on Advanced & Communcation Technologies .*

## LAMPIRAN

### A.1 PROGRAM NODE

```
#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");
  Mirf.payload = 10;
  Mirf.channel = 100;
  Mirf.config();
  Serial.println("Node 1");
}


String datasuhu="30";
byte datasends[10]=""; // kirim metadata suhu
byte datasendsuhu[10]=""; // digunakan untuk mengirim data asli suhu
byte datareq[10]=""; // save req suhu
int node=1;
int x;


void loop(){

  delay(10000);
  datasends[0]=1; // node 1
  datasends[1]=2; // node 2
  datasends[2]=3; // metadata suhu
```



```
Mirf.setTADDR((byte *)"node2");
Mirf.send((byte *)&datasends);// mengirim meta data
Serial.println("-----");
Serial.println("Kirim metadata suhu");

unsigned long time = millis();
while(millis() - time <= 2000){
if(!Mirf.isSending() && Mirf.dataReady()){
    Mirf.getData(datareq); // terima request
    Serial.print("menerima request suhu:");
    Serial.println(datareq[2]);
    Serial.println(time);//cetak waktu millis

    if(datareq[2]==1 && datareq[1]==1){
        Serial.println("siap mengirim data suhu");

        datasendsuhu[0]=datasends[0];
        datasendsuhu[1]=datasends[1];
        for(int x=0;x<datasuhu.length();x++){
            datasendsuhu[x+2]=datasuhu[x];
        }
        delay(50);
        Mirf.setTADDR((byte *)"node2");
        Mirf.send((byte *)&datasendsuhu);//kirim data suhu
        Serial.println("Data berhasil dikirim");

    }}}
delay (1000);
}
```

## A.2 PROGRAM NODE 2

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

void setup(){
  Serial.begin(9100);
  Mirf.spi = &MirfHardwareSpi;
  Mirf.init();
  Mirf.setRADDR((byte *)"node2");
  Mirf.payload = 10;
  Mirf.channel = 100;
  Mirf.config();
  Serial.println("Node 2");
}

String datasuhu="30";
byte datasends[10]="";// kirim metadata suhu
byte datareq[10]="";// save req suhu
byte datam[10]="";// ambil meta data dari node 1
byte datas[10]="";// penyimpanan meta data
byte data[10]="";// ambil meta data dari node 1 dan save data suhu
int node=2;
byte datasendc[10]="";// payload pengirim request
int x;

void loop(){
  unsigned long time = millis();
  if(!Mirf.isSending() && Mirf.dataReady()){
    Mirf.getData(datam);// menerima metadata dari node 1
    delay (50);
```

```
datasends[0]=2;//node 2
datasends[1]=3;//node 3
datasends[2]=4;//node 4
datasends[3]=5;//node 5
datasends[4]=6;//metadata suhu

Mirf.setTADDR((byte *)"node3");
Mirf.send((byte *)&datasends); //mengirim meta data ke node 3
Serial.println("-----");
Serial.println("Meta-data diterima dari node 1");
for(x=0;x<10;x++){
    datas[x] = datam[x];
}
if(datas[2]==3){Serial.println("");}
Mirf.setTADDR((byte *)"node4");
Mirf.send((byte *)&datasends);// kirim meta data ke node 4
Mirf.setTADDR((byte *)"node5");
Mirf.send((byte *)&datasends);//kirim meta data ke node 5
if(datas[1]==2 && datas[2]==3){
    Serial.println("Siap kirim Request ke node 1");
    datasendc[0]=datam[1];//node tujuan node 2
    datasendc[1]=datam[0];// node asal node 1
    datasendc[2]=1; //minta suhu
    Mirf.setTADDR((byte *)"node1");
    Mirf.send((byte *)&datasendc);// kirim request

    Serial.println("Request dikirim ke node 1");
    Serial.println("-----");
```

```
while(millis() - time <= 2000){  
    if(!Mirf.isSending() && Mirf.dataReady()){  
        Mirf.getData(data);// terima data dari node 1  
        Serial.println("Terima data node 1");  
        Serial.print(data[2]);  
        Serial.println(data[3]);  
        Serial.println(time);  
        Serial.println("-----");  
        delay (500);  
        Serial.println("Meta-data dikirim ke node 3,4,5");  
        Mirf.getData(datareq);// menerima request dari node 3  
        if(datareq[0]==3 && datareq[1]==2){  
            Serial.print("menerima request dari node 3");  
            Serial.println("");  
            Serial.println(datareq[2]);  
            delay (500);  
            Mirf.getData(datareq);// menerima request dari node 4  
            if(datareq[0]==4 && datareq[1]==2){// apabila node tujuan benar node 4  
                dan request suhu berarti kirim data suhu  
                Serial.print("menerima request dari node 4");  
                Serial.println("");  
                Serial.println(datareq[2]);  
  
                Serial.println("kirim data suhu ");  
                Mirf.setTADDR((byte *)"node3");  
                Mirf.send((byte *)&data);//kirim data ke node 3  
                Mirf.setTADDR((byte *)"node4");  
                Mirf.send((byte *)&data);//kirim data ke node 4  
            }  
        }  
    }  
}
```



### A.3 PROGRAM NODE 3

```
#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 *)"node3");
  Mirf.payload = 10;
  Mirf.channel = 100;
  Mirf.config();
  Serial.println("Node 3");
}

byte datam[10]=""; //ambil meta data dari node 2
byte datas[10]=""; //penyimpanan meta data
byte data[10]=""; //ambil data suhu
int node=3;
byte datasendc[10]=""; //payload pengirim request
int x;
void loop(){

  if(!Mirf.isSending() && Mirf.dataReady()){
    Mirf.getData(datam); //terima meta data dari node 2
    Serial.println("Meta-data diterima");
    Serial.println(datam[4]);
    for(x=0;x<10;x++){
      datas[x] = datam[x];
    }
  }
}
```



```
if(datas[0]==2 && datas[1]==3 && datas[4]==6){  
    Serial.println("Siap kirim Request");  
    datasendc[0]=3;//node 3  
    datasendc[1]=2;//node asal node 2  
    datasendc[2]=1;//minta suhu  
  
    delay (550);  
    Mirf.setTADDR((byte *)"node2");  
    Mirf.send((byte *)&datasendc);// kirim request ke node 2  
    Serial.println("Requestdikirim");  
  
    unsigned long time = millis();  
    while(millis() - time <= 2000){  
  
        if(!Mirf.isSending() && Mirf.dataReady()){  
            Mirf.getData(data);// terima data dari node 1  
            Serial.println("Terima data");  
            Serial.print(data[2]);  
            Serial.println(data[3]);  
            Serial.println("-----");  
            return;  
        }  
    }  
}
```

#### A.4 PROGRAM NODE 4

```
#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 *)"node4");
  Mirf.payload = 10;
  Mirf.channel = 100;
  Mirf.config();
  Serial.println("Node 4");
}

byte datam[10]=""; //ambil meta data dari node 2
byte datasends[10]=""; //penyimpanan meta data
byte data[10]=""; //ambil data suhu
int node=4;
byte datareq[10]=""; //payload pengirim request
int x;
void loop(){
  if(!Mirf.isSending() && Mirf.dataReady()){
    Mirf.getData(datam); //terima meta data dari node 2
    Serial.println("Meta-data diterima");
    Serial.println(datam[4]);
    for(x=0;x<10;x++){
      datasends[x] = datam[x];
    }
  }
}
```

```
if(datasends[0]==2 && datasends[2]==4 && datasends[4]==6){  
    Serial.println("Siap kirim Request");  
    datareq[0]=4;//node 4  
    datareq[1]=2;//node asal node 2  
    datareq[2]=1;//minta suhu  
  
    delay (1100);  
    Mirf.setTADDR((byte *)"node2");  
    Mirf.send((byte *)&datareq);// kirim request ke node 2  
    Serial.println("Requestdikirim");  
  
    unsigned long time = millis();  
    while(millis() - time <= 2000){  
  
        if(!Mirf.isSending() && Mirf.dataReady()){  
            Mirf.getData(data);// terima data dari node 1  
            Serial.println("Terima data");  
            Serial.print(data[2]);  
            Serial.println(data[3]);  
            Serial.println("-----");  
            return;  
        }  
    }  
}
```

## A.5 PROGRAM NODE 5

```
#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 *)"node5");
  Mirf.payload = 10;
  Mirf.channel = 100;
  Mirf.config();
  Serial.println("Node 5");
}

byte datam[10]=""; //ambil meta data dari node 2
byte datas[10]=""; //penyimpanan meta data
byte data[10]=""; //ambil data suhu
int node=5;
byte datasendc[10]=""; //payload pengirim request
int x
void loop(){

  if(!Mirf.isSending() && Mirf.dataReady()){
    Mirf.getData(datam);
    Serial.println("-----");
    Serial.println("Meta-data diterima dari node 2");
    Serial.println(datam[4]);
    for(x=0;x<10;x++){
      datas[x] = datam[x];
    }
  }
}
```



```
if(datas[0]==2 && datas[3]==5 && datas[4]==6){  
    Serial.println("Node 5 tidak ingin meminta data");  
}  
//Mirf.payload = 10;  
}  
}
```