

IMPLEMENTASI ZIGBEE IEEE 802.15.4 UNTUK PEMANTAUAN SUHU DAN KELEMBABAN UDARA

Jusak¹⁾

1) Program Studi/Jurusan Sistem Komputer, STMIK STIKOM Surabaya, email: jusak@stikom.edu

Abstract: Wireless Sensor Networks (WSN) technology has been deployed widely in many countries for real life applications, such as environment monitoring, military as well as health monitoring. However, its implementation in Indonesia is still limited. In this paper, we build and evaluate a wireless sensor networks system and its communication protocol that are used to retrieve data from temperature and humidity sensors. The wireless sensor networks nodes were equipped with radio Xbee Series 2 that were operated based on the Zigbee 802.15.4 protocol standard. It is shown that the transmission of temperature and humidity data can be accomplished well using a communication protocol that was built in the wireless sensor networks system.

Keywords: wireless sensor networks, temperature, humidity, monitoring.

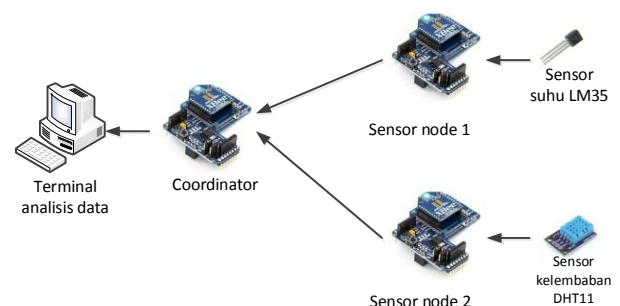
Teknologi Jaringan Sensor Nirkabel (JSN) adalah sebuah teknologi yang merupakan gabungan antara teknologi penginderaan (*sensing*) dengan teknologi jaringan (*networks*). Walaupun aplikasi JSN telah banyak digelar di beberapa negara maju, implementasi teknologi JSN ini masih tergolong baru di Indonesia, padahal teknologi JSN sangat bermanfaat untuk memantau, misalnya lahan pertanian, kebakaran hutan, kondisi lingkungan (Fotue, 2009; Hu, 2011; Khedo, 2010), dan juga kondisi kesehatan seseorang (Ko, 2010). Karena itu potensi dan kebutuhan pemanfaatan teknologi JSN di Indonesia diperkirakan akan besar pada masa yang akan datang.

Dalam penelitian terapan ini, sebuah prototipe sistem JSN akan dibangun untuk melakukan pemantauan terhadap suhu dan kelembaban udara. Dua parameter ini dipilih karena suhu dan kelembaban udara merupakan parameter penting dalam sebagian besar implementasi JSN di lapangan. Misalnya, untuk pemantauan kondisi udara, untuk pemantauan tanaman, dan lain sebagainya. Fokus utama pada penelitian ini adalah perencanaan protokol komunikasi di dalam sistem JSN agar masing-masing *node* dapat terintegrasi dan berkomunikasi sebagai satu kesatuan sistem. Sistem jaringan JSN yang akan dibangun terdiri atas beberapa sensor *node* dengan menggunakan papan mikrokontroler *Arduino Uno* (McRoberts, 2010)) dan radio Xbee sebagai perangkat transmisi data yang berasal dari *node* JSN menuju ke *coordinator*.

JSN dibangun dengan berdasarkan standar internasional Zigbee 802.15.4 sebagai protocol komunikasi. Standar Zigbee beroperasi pada frekuensi 2,4 GHz (LAN/MAN Standards Committee, 2011). Keuntungan menggunakan standar Zigbee antara lain: Zigbee membutuhkan daya rendah (misalnya, Zigbee Series 2 yang diimplementasi pada radio Xbee menggunakan daya rendah sebesar 1mW). Dengan konsumsi daya rendah tersebut, Zigbee sangat berguna untuk mempertahankan umur hidup yang panjang dari *node* JSN (Gholamzadeh, 2008; Faludi, 2011).

METODE

Sistem JSN yang akan dibangun memiliki 2 buah sensor *node* dan sebuah *coordinator* yang masing-masing memiliki fungsi dan spesifikasi teknis sebagaimana terlihat dalam Tabel 1.



Gambar 1. Arsitektur JSN

Arsitektur JSN yang digunakan dalam penelitian ini tersusun dalam rangkaian dua buah sensor *node* dan

sebuah *coordinator*. Lihat dalam Gambar 1. Sensor *node 1* dan sensor *node 2* berfungsi sebagai pengumpul data, yang mana sensor *node 1* bertugas untuk mengambil data analog yang berasal dari sensor suhu, sedangkan sensor *node 2* bertugas untuk mengambil data analog yang berasal dari sensor kelembaban ruangan.

Tabel 1. Spesifikasi Sensor *Node*

Nama	Fungsi	Spesifikasi
Sensor <i>Node 1</i>	End-device/Router untuk membaca data yang berasal dari sensor suhu.	Arduino Uno SMD, Xbee shield dan radio Xbee Series 2.
Sensor <i>Node 2</i>	End-device/Router untuk membaca data yang berasal dari sensor kelembaban udara.	Arduino Uno R3, Xbee shield dan radio Xbee PRO Series 2.
Coor-dinator	Coordinator	Arduino Uno SMD, Xbee shield dan radio Xbee Series 2.

Sensor *Node 1*

Sensor *node 1* menggunakan mikrokontroler Arduino Uno SMD untuk mengambil data analog yang berasal dari sensor suhu LM35. Masukan data analog melalui pin A0 pada modul Arduino. Sensor suhu LM35 sendiri merupakan sensor suhu dalam bentuk *Integrated Circuit* (IC) yang memiliki presisi cukup baik dengan keluaran yang proporsional terhadap suhu dalam satuan °C.

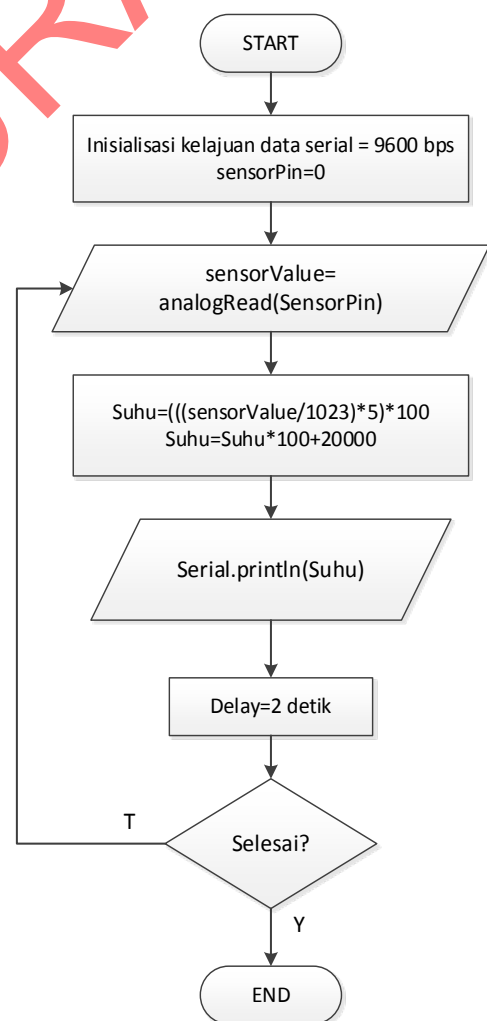
Sensor *Node 2*

Sensor *node 2* menggunakan modul mikrokontroler Arduino Uno R3 untuk mengambil data analog yang berasal dari modul sensor kelembaban DHT11. Masukan data analog melalui pin A0 pada modul Arduino. Modul DHT11 sebenarnya merupakan

modul yang terdiri atas 2 sensor, yaitu sensor kelembaban dan sensor suhu, namun pada penelitian ini kami hanya memanfaatkan fungsi sensor kelembaban saja. Sensor DHT11 berfungsi untuk melakukan pengukuran kelembaban terhadap udara disekitarnya. Sensor DHT11 ini merupakan sensor kelembaban kapasitif.

Coordinator

Coordinator menggunakan mikrokontroler Arduino Uno SMD dan radio Xbee Series 2. Tidak ada modul sensor yang dilekatkan pada *coordinator*. Tugas dari *coordinator* adalah menerima data yang berasal dari Sensor *node 1* dan Sensor *node 2*. Selanjutnya data yang berasal dari kedua sensor dikirimkan ke komputer melalui kabel USB yang terhubung pada mikrokontroler.



Gambar 2. Diagram Alir Perangkat Lunak Pada Sensor *Node 1*.

HASIL DAN PEMBAHASAN

Komunikasi antara radio Xbee dengan papan mikrokontroler Arduino pada kedua sensor *node* dan *coordinator* menggunakan komunikasi serial dengan kecepatan perpindahan data yang bervariasi. Akan tetapi dalam penelitian ini digunakan kecepatan perpindahan data tetap sebesar 9.600 *bit per second* (bps).

Diagram alir untuk perangkat lunak yang ditanam pada mikrokontroler Arduino untuk sensor *node* 1 ditunjukkan dalam Gambar 2.

Seperti terlihat dalam Gambar 2 komunikasi serial pada mikrokontroler pada sensor *node* 1 akan berkomunikasi dengan kelajuan sebesar 9.600 bps. Mikrokontroler menggunakan komunikasi serial ini untuk melakukan transmisi dan menerima data melalui radio Xbee. Selanjutnya variable *sensorPin* diset menjadi 0 yang menunjukkan bahwa data suhu akan diambil dari pin analog 0 pada mikrokontroler.

Pembacaan nilai dari sensor suhu LM35 pada waktu tertentu dilakukan melalui perintah `analogRead(sensorPin)`, dan hasilnya diletakkan dalam sebuah variabel bernama *sensorValue*. Selanjutnya nilai suhu dalam derajat Celsius didapatkan melalui rumusan (McRoberts, 2010):

$$T = \left(\left((sV/1023) \times 5 \right) \times 100 \right) \dots\dots\dots(1)$$

yang mana:

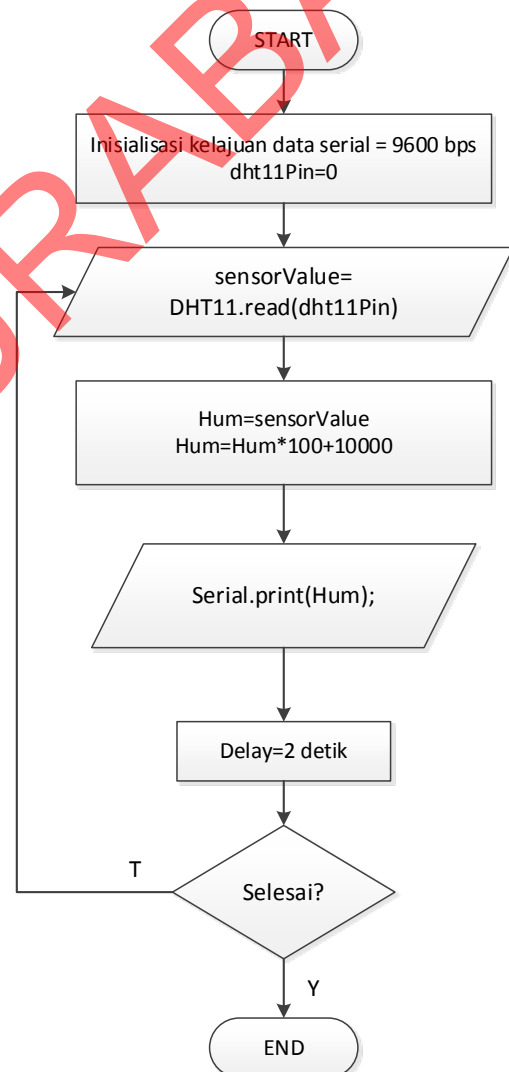
T = suhu dalam °C.

sV = nilai pembacaan sensor analog (variabel *sensorValue*).

Data pada variable suhu yang memiliki nilai desimal sebanyak 2 tersebut selanjutnya dinormalisasi dengan mengalikan data tersebut dengan bilangan 100, sehingga menjadi sebuah angka tanpa nilai pecahan. Selanjutnya untuk tujuan memberikan identifikasi unik bahwa data ini berasal dari sensor *node* 1, angka 20.000 ditambahkan pada data tersebut. Sehingga data sensor *node* 1 yang terkirim melalui radio Xbee selalu ditandai dengan angka awal 2.

Proses transmisi data dilakukan dengan memberikan perintah `Serial.print(Suhu)`. Untuk proses pengambilan data sensor berikutnya diberikan waktu tunda sebesar 2 detik. Apabila proses pembacaan data masih akan dilakukan lagi, maka program akan melakukan perulangan menuju ke pembacaan data sensor sampai proses pembacaan sensor suhu dianggap selesai (terminasi oleh pengguna atau catu daya dari sensor *node* dicabut/habis).

Diagram alir untuk perangkat lunak yang ditanam pada mikrokontroler Arduino untuk sensor *node* 2 ditunjukkan dalam Gambar 3.



Gambar 3. Diagram Alir Perangkat Lunak Pada Sensor *Node* 2

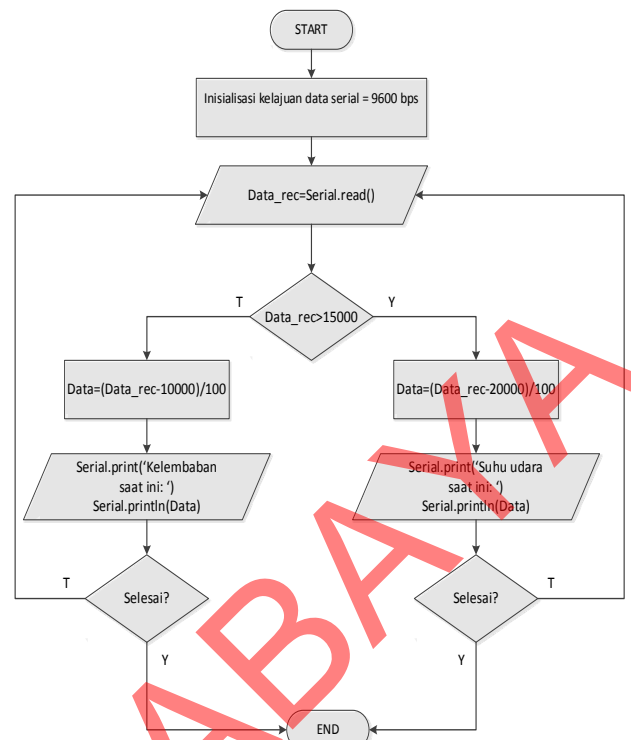
Seperti terlihat dalam Gambar 3 komunikasi serial pada mikrokontroler pada sensor *node* 2 akan

berkomunikasi dengan kelajuan sebesar 9.600 bps. Mikrokontroler menggunakan komunikasi serial ini untuk melakukan transmisi dan menerima data melalui radio Xbee. Selanjutnya variable `dht11Pin` diset menjadi 0 yang menunjukkan bahwa data kelembaban akan diambil dari pin analog 0 pada mikrokontroler.

Pembacaan nilai dari sensor suhu pada waktu tertentu dilakukan melalui perintah `dht11.read(dht11Pin)`, dan hasilnya diletakkan dalam sebuah variabel bernama `sensorValue`. Data pada variable `hum` yang memiliki nilai pecahan sebanyak 2 angka tersebut dinormalisasi dengan mengalikan data tersebut dengan bilangan 100 sedemikian sehingga menjadi sebuah angka tanpa pecahan. Selanjutnya untuk mengidentifikasi bahwa data ini berasal dari sensor *node* 1, maka ditambahkan angka 10.000 pada data tersebut. Sehingga data sensor *node* 1 yang terkirim melalui radio Xbee selalu ditandai kode unik dengan angka awal 1.

Proses transmisi data dilakukan dengan memberikan perintah `Serial.print(Hum)`. Untuk proses pengambilan data sensor berikut diberikan waktu tunda sebesar 2 detik. Apabila proses pembacaan data masih akan dilakukan lagi, maka program akan melakukan looping menuju ke pembacaan data sensor lagi sampai proses pembacaan sensor kelembaban dianggap selesai (terminasi oleh pengguna atau catu daya dari sensor *node* dicabut/habis).

Diagram perangkat lunak yang ada di dalam mikroprosesor (di dalam modul Arduino Uno) pada sisi *coordinator* untuk mengendalikan pembacaan data yang berasal dari sensor *node* 1 dan sensor *node* 2 ditunjukkan dalam Gambar 4.



Gambar 4. Diagram Alir Perangkat Lunak Pada Sisi *Coordinator*

Seperti terlihat dalam Gambar 4 komunikasi serial pada mikrokontroler pada *coordinator* akan berkomunikasi dengan kelajuan sebesar 9.600 bps. Mikrokontroler menggunakan komunikasi serial ini untuk menerima data melalui radio Xbee.

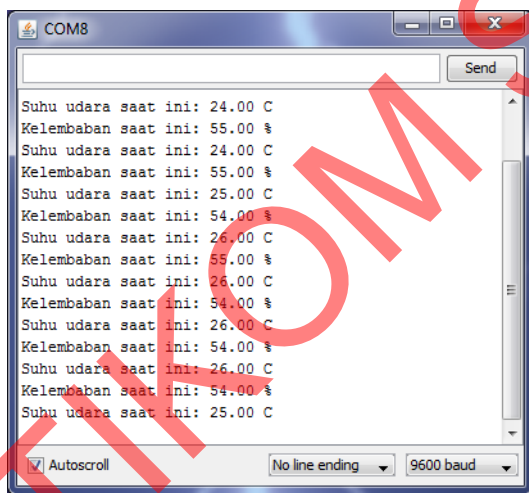
Pembacaan data yang diterima oleh radio XBee pada waktu tertentu dilakukan melalui perintah `Serial.read()`, dan hasilnya diletakkan dalam sebuah variabel bernama `Data_rec`.

Seperti dijelaskan pada bagian sebelumnya (A dan B) bahwa sensor *node* 1 menandai data suhu dengan penambahan angka 20.000 sedangkan sensor *node* 2 menandai data kelembaban dengan penambahan angka 10.000. Karena itu *coordinator* membedakan data yang diterima dengan cara melakukan pengujian apakah data yang diterima lebih besar/kecil dari angka 15.000.

Apabila data yang diterima lebih besar dari 15.000, tentulah data tersebut merupakan data suhu. Maka selanjutnya kebalikan proses pemformatan data yang dilakukan pada sensor *node* 1 dilakukan, yaitu data yang diterima oleh *coordinator* dikurangi dengan angka 20.000 dan dibagi dengan 100 untuk menjadi sebuah angka suhu yang memiliki 2 desimal.

Sebaliknya Apabila data yang diterima lebih kecil dari 15.000, maka dapat dipastikan bahwa data tersebut merupakan data kelembaban yang berasal dari sensor *node* 2. Maka selanjutnya kebalikan proses pemformatan data yang dilakukan pada sensor *node* 2 dilakukan, yaitu data yang diterima oleh *coordinator* dikurangi dengan angka 10.000 dan dibagi dengan 100 untuk menjadi sebuah angka kelembaban yang memiliki 2 desimal.

Apabila proses pembacaan data masih akan dilakukan lagi, maka program akan melakukan looping menuju ke pembacaan data lagi sampai proses pembacaan dianggap selesai (terminasi oleh pengguna atau catu daya dari *coordinator* dicabut/habis).



Gambar 5. Tampilan Pembacaan Suhu Dan Kelembaban

Berdasarkan hasil pembacaan suhu dan kelembaban yang diterima oleh *nodecoordinator* didapatkan beberapa kesimpulan terkait dengan sistem JSN yang telah dibangun dalam penelitian ini. Terlihat bahwa komunikasi antara sensor *node* 1 dan *coordinator*, dan begitu juga komunikasi antara sensor *node* 2 dan *coordinator* dapat berjalan dengan baik.

Lihat Gambar 5. Hal ini dibuktikan dengan cara membandingkan data terkirim oleh sensor *node* 1 atau sensor *node* 2 dengan data yang telah diterima oleh *coordinator*. Apabila salah satu sensor *node* gagal berfungsi, misalnya sensor *node* 1 tidak mengirimkan data sama sekali ke *coordinator*, maka *coordinator* hanya menampilkan data yang berasal dari sensor *node* 2 saja.

SIMPULAN

Berdasarkan hasil uji coba dalam penelitian ini didapatkan bahwa sensor *node* yang tersusun atas mikrokontroler Arduino Uno, Xbee shield dan radio Xbee Series 2 dapat melaksanakan fungsi komunikasi dengan baik untuk mengambil dan melakukan transmisi data. Proses transmisi dilakukan pada frekuensi 2,4 GHz sesuai dengan standar Zigbee.

Struktur jaringan sensor nirkabel yang digunakan dalam penelitian ini terdiri atas dua buah sensor *node* yang berfungsi sebagai *node* pengambil data yang berasal dari sensor dan sebuah *node* yang berfungsi sebagai *coordinator*. Sensor mengambil data suhu dan kelembaban udara dalam waktu bersamaan atau berbeda. Struktur semacam ini menyerupai sebuah jaringan dengan topologi star yang mana masing-masing sensor *node* berhubungan dengan sebuah hub (dalam hal ini *nodecoordinator*) sebagai tempat pengumpul data. Proses komunikasi data sensor suhu dan sensor kelembaban udara menuju ke *coordinator* menggunakan sebuah protokol komunikasi yang telah ditanam pada masing-masing *node*. Berdasarkan hasil uji coba, protokol komunikasi juga memiliki kemampuan untuk mengidentifikasi masing-masing sensor *node* yang berbeda di dalam sebuah jaringan JSN.

RUJUKAN

- Ko, J.G., et al. 2010. Wireless Sensor Networks for Healthcare. *Proceeding of the IEEE*, 9(11), pp.1947-1960.
- Faludi, R. 2011. *Building Wireless Sensor Networks*. California: O'Reilly Media.

- Fotue, D., Tanonkou, G., and Engel, T. 2009. An ad-hoc Wireless Sensor Networks with Application to Air Pollution Detection. *Sensor Networks and Applications Conference 2009*, San Francisco, California, pp. 48-53.
- Gholamzadeh, B., and Nabovati, H. 2008. Concept for Designing Low Power Wireless Sensor Network. *World Academy of Science, Engineering and Technology*, 2008(45).
- Hu, S.C., Wang, Y.C., Huang, C.Y., and Seng, Y.C. 2011. Measuring Air Quality in City Areas by Vehicular Wireless Sensor. *The Journal of Systems and Software*, 84 (2011), pp. 2005-2012.
- LAN/MAN Standards Committee. 2011. Part 15.4: Low Rate Wireless Personal Area Network. *IEEE Standard Board*.
- Khedo, K.K., Perseedoss, R., and Mungur, A. 2010. A Wireless Sensor Network Air Pollution Monitoring System. *International Journal of Wireless & Mobile Networks*, 2.2 (2010), pp. 31-45.
- McRoberts, M. 2010. *Beginning Arduino*. New York, USA: Technology in Action.

STIKOM SURABAYA