

BAB III

METODE PENELITIAN DAN PERANCANGAN SISTEM

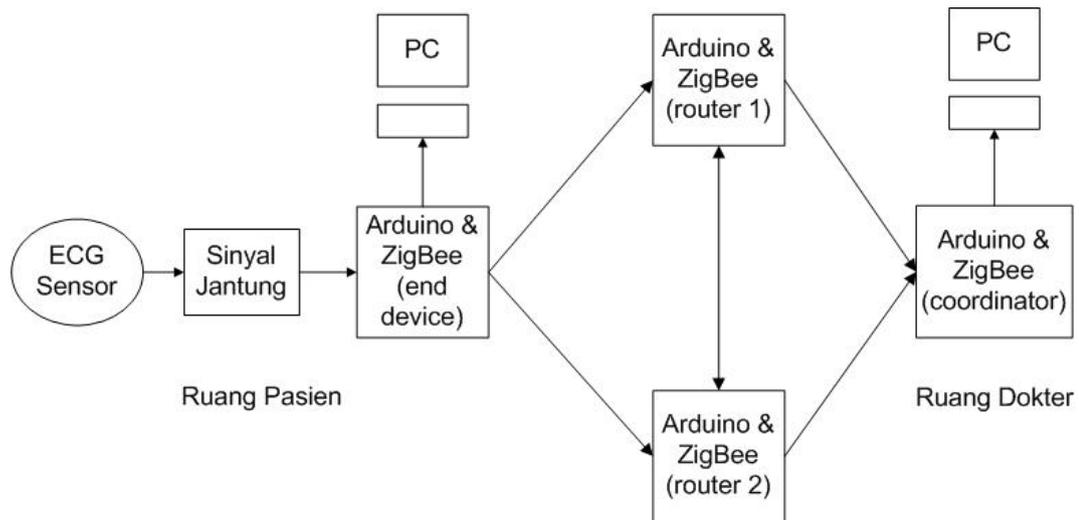
3.1 Metode Penelitian

Metode penelitian yang digunakan pada perancangan ini adalah perancangan sistem wireless sensor network dengan menggunakan ZigBee sebagai media pengiriman sinyal. Zigbee series 2 modul RF dirancang untuk beroperasi dalam protokol ZigBee dengan biaya yang murah dan jaringan sensor nirkabel menggunakan daya yang rendah (Faludi, 2011). Modul ini beroperasi pada frekuensi 2.4 GHz dan memiliki jarak jangkauan indoor 133 ft atau 40 meter dan jarak jangkauan outdoor 400 ft atau 120 meter tanpa adanya halangan (Inc, XBee Series 2 OEM RF Modules, 2007).

Dengan terbatasnya kemampuan ZigBee untuk mentransmisikan sinyal tersebut berdasarkan kebutuhan arsitektur bangunan yang ada maka penulis membuat sebuah topologi jaringan mesh dalam penerapan penelitian ini. Dengan menggunakan topologi jaringan mesh diharapkan dapat mengatasi masalah pengiriman sinyal ECG yang terkendala dengan bentuk arsitektur bangunan ataupun jarak ruangan.

3.2 Model Perancangan

Pada perancangan ini penulis menggambarkan perancangan sistemnya seperti pada gambar berikut ini.



Gambar 3.1. Gambar Perancangan Sistem

Pada perancangan sistem ini menggunakan topologi mesh seperti terlihat pada gambar 3.1. Model topologi mesh ini digunakan agar node *end device* dapat mengirimkan sinyal jantung kepada node coordinator melalui *node router 1* atau *node router 2* tanpa ada kegagalan karena kendala infrastruktur gedung ataupun jarak kemampuan pengiriman data oleh *ZigBee*. selain itu topologi ini memiliki fleksibilitas yang lebih besar dibanding topologi lainnya. Jika sebuah *router* tertentu gagal, maka jaringan dapat merekonstruksi jalur alternatif melalui *router* lain dalam jaringan.

Jumlah *node* yang digunakan pada perancangan kali ini berjumlah 4 *node*, yaitu *node coordinator*, *router 1,2* dan *node end device*. Masing-masing *node* memiliki tugas berbeda-beda seperti berikut ini :

1. *Node end device*

Pada *node* ini, *node* bertanggung jawab sebagai pencatat hasil rekaman sinyal jantung pada pasien, dan mengirimkan data pada *node coordinator* melalui *node router 1,2* sesuai dengan protokol yang telah dibuat melalui modul arduino.

2. *Node router 1*

Pada *node* ini, hasil pengiriman sinyal yang dikirimkan oleh *node end device* apabila dikirimkan ke *node router* ini maka akan diteruskan langsung menuju ke *node coordinator* atau apabila tidak memungkinkan maka akan diteruskan menuju *node router 2* untuk diteruskan ke *node coordinator*.

3. *Node router 2*

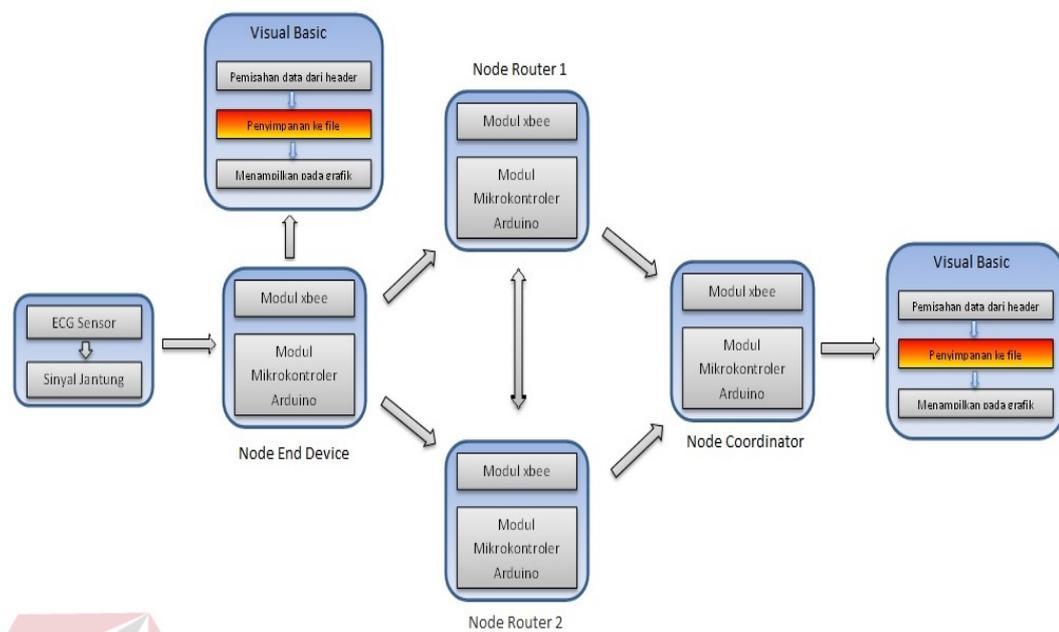
Pada *node* ini, hasil pengiriman sinyal yang dikirimkan oleh *node end device* apabila dikirimkan ke *node router* ini maka akan diteruskan langsung menuju ke *node coordinator* atau apabila tidak memungkinkan maka akan diteruskan menuju *node router 1* untuk diteruskan ke *node coordinator*.

4. *Node coordinator*

Node coordinator bertanggung jawab atas penerimaan data yang telah dikirimkan oleh *node end device* sensor jantung. Data yang diterima *node* ini masih belum diolah, tapi data yang diterima sesuai dengan protokol pengiriman data. Pada *node coordinator* data langsung diolah dan di tampilkan dalam PC dengan menggunakan software visual basic.

3.3 Perancangan Sistem

Adapun blok diagram perancangan sistem ditunjukkan pada gambar 3.2 berikut ini.



Gambar 3.2. Blok Diagram Sistem

Dalam tugas akhir ini, penulis hanya akan memfokuskan penjelasan data yang dikirim dari *node end device* ke *node coordinator*, dan juga hasil unjuk kerja jaringan topologi mesh pada transmisi sinyal auskultasi jantung dari *node end device* ke *node coordinator*.

3.4 Perancangan perangkat keras

3.4.1 Perancangan sensor jantung

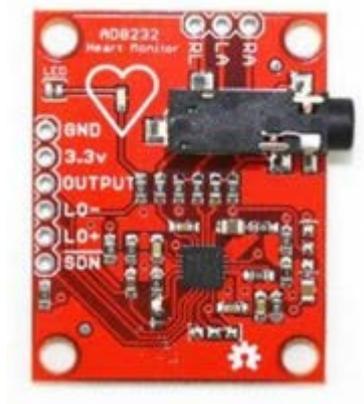
Untuk dapat mendeteksi adanya detak jantung pasien secara elektronik, maka dibutuhkan sensor. Sensor yang digunakan pada penelitian transmisi sinyal auskultasi jantung ini adalah dengan menggunakan modul ECG (*Heart Rate Monitor*) yang disambungkan dengan mikrokontroler arduino untuk bisa mengambil sinyal auskultasi jantung dengan menggunakan elektroda jepit yang

dipasangkan pada tubuh pasien dengan menjepitkan pada kedua pergelangan tangan dan pergelangan kaki kanan pasien.

Heart Rate Monitor adalah modul yang digunakan untuk mengukur aktivitas listrik jantung. aktivitas listrik ini dapat memetakan sebagai ECG atau elektrokardiogram dan output sebagai pembaca analog. ECG bisa sangat bising, Lead Tunggal AD8232 *Heart Rate Monitor* bertindak sebagai op amp untuk membantu mendapatkan sinyal yang jelas dari PR dan Interval QT lebih mudah.

AD8232 adalah sebuah blok pengkondisian sinyal terintegrasi untuk ECG dan aplikasi pengukuran biopotential lainnya. Hal ini dirancang untuk mengekstrak, memperkuat, dan menyaring sinyal biopotential kecil dihadapan kondisi bising, seperti yang dibuat oleh gerakan atau penempatan elektroda terpencil.

Heart Rate Monitor memiliki sembilan koneksi dari IC yang dapat Anda solder pin, kabel, atau konektor lain untuk. SDN, LO +, lo-, OUTPUT, 3.3V, GND memberikan pin penting untuk operasi monitor ini dengan Arduino atau mikrokontroler lainnya. Juga terdapat pin lain yaitu RA (Right Arm), LA (Left Arm), dan RL (Right Leg) pin untuk disambungkan dengan Elektroda Jepit. Selain itu, ada lampu indikator LED yang akan berdenyut dengan irama jantung berdetak.

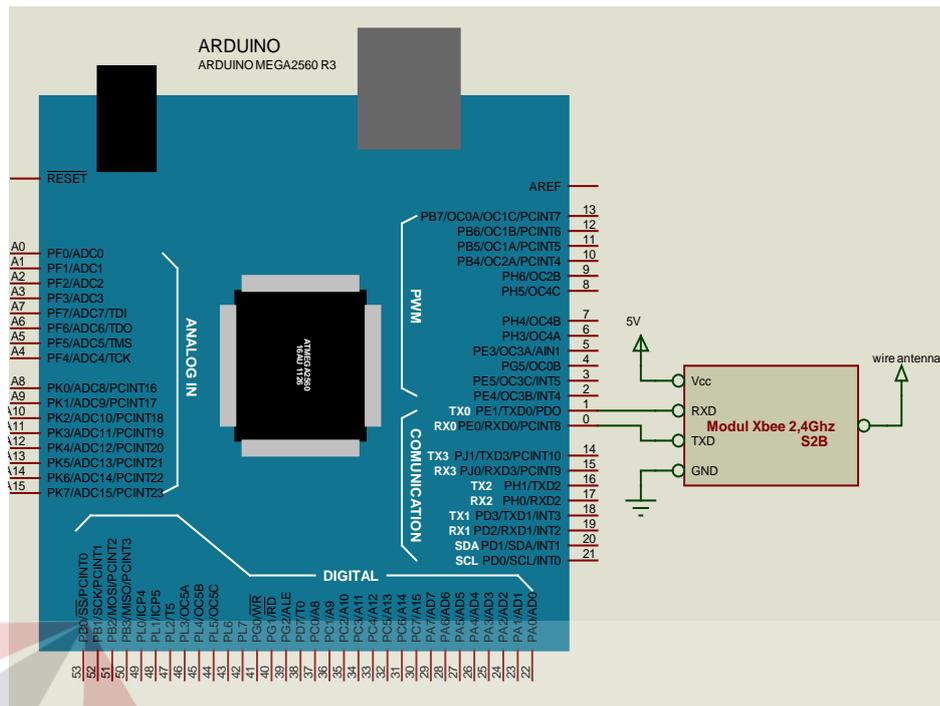


Gambar 3.3. *Heart Rate Monitor (Modul ECG)*

3.4.2 Perancangan rangkaian Xbee Zigbee S2B

Agar modul arduino dapat berkomunikasi secara serial *wireless* dengan perangkat lain, maka dibutuhkan rangkaian *wireless* yang dalam perancangan ini menggunakan modul Zigbee S2B. modul zigbee dapat berkomunikasi *wireless* dan diakses menggunakan komunikasi serial TTL (*Time to Live*). Adapun port serial yang digunakan untuk pengendalian dan pembacaan modul Xbee adalah TX0 dan RX0 pada modul arduino sebagaimana ditunjukkan pada gambar 3.4:

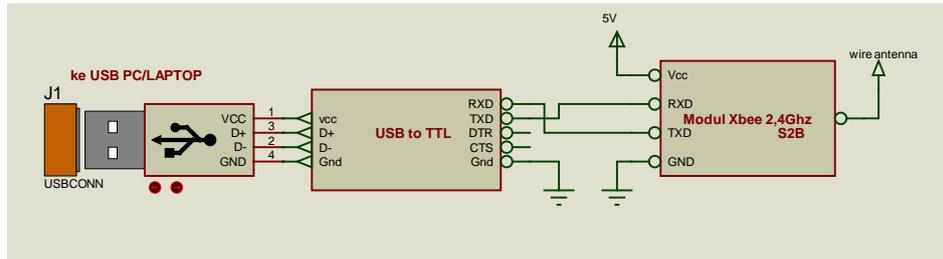
SURABAYA



Gambar 3.4 Hubungan Rangkaian Xbee dan arduino

3.4.3 Perancangan rangkaian USB to serial Xbee

Untuk dapat menerima data serial hasil pengiriman dari *node* 1 dan *node* 2 atau perangkat *wireless* Xbee pengirim, maka pada bagian penerima juga dibutuhkan Xbee. Sementara itu agar hasil pembacaan dan pengiriman data pada Xbee dapat diproses menggunakan laptop atau PC, maka dibutuhkan konverter USB to serial. Untuk itu pada perancangan ini digunakan modul USB to serial Xbee yang difungsikan khusus untuk menjembatani antarmuka UART antara komputer dengan Xbee. Adapun rangkaian modul USB to serial Xbee ditunjukkan pada gambar 3.5 :



Gambar 3.5 Rangkaian modul USB to Serial Xbee pada *base station*

Sementara itu bentuk fisik dari modul USB to serial Xbee ditunjukkan pada gambar 3.6



Gambar 3.6 modul USB to serial Xbee

3.4.4 Arduino 2560

Pada gambar 3.2 terdapat 2 arduino 2560 yang memiliki fungsi yang sama yaitu membaca sensor yang memiliki nilai analog, pembacaan data dilakukan dengan cara inputan yang berasal dari sensor diletakkan pada PORT A0, untuk membaca nilai dari sinyal analog tersebut digunakan fungsi *ReadAnalog* didalam modul arduino.

Pada modul arduino juga dilakukan pemberian tanda pada data yang akan ditransmisikan. Adapun format pengiriman data seperti terlihat pada gambar di bawah ini.

Format Pengiriman Data



Gambar 3.7 Format Pengiriman Data

Berikut penjelasan dari gambar 3.7 :

1. % : penanda awal pengiriman data
2. DATA : data sinyal auskultasi jantung yang dikirimkan
3. # : penanda akhir pengiriman data

Hal tersebut dibuat untuk memudahkan dalam pengolahan data pada saat penerimaan data pada *coordinator*. Selanjutnya arduino mengirimkan informasi yang dipancar melalui pemancar data zigbee.

3.4.5 Xbee

Untuk mengirimkan data dari *node end device* ke *coordinator* diperlukan sebuah pemancar data. dalam penelitian ini penulis menggunakan Xbee Series 2 untuk pemancar data. Konfigurasi yang dilakukan pada Xbee sangat penting, agar data dapat dikirimkan ke alamat yang sesuai.

Untuk mengkonfigurasi Xbee tersebut dibutuhkan sebuah *software*. *Software* yang biasa digunakan untuk mengkonfigurasi Xbee salah satunya ialah X-CTU.

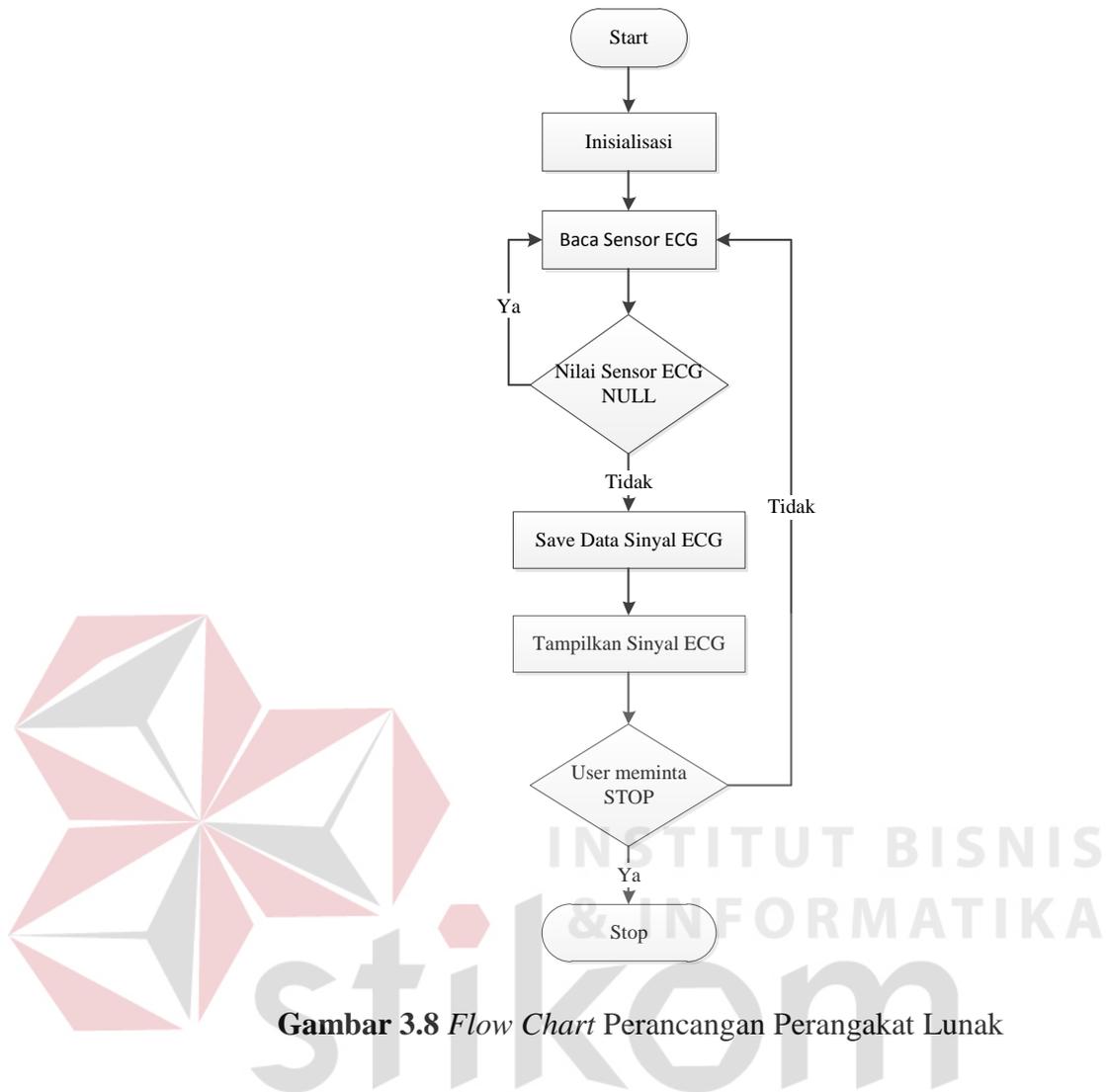
Xbee dikonfigurasi untuk menjadi *Znet 2.5 Router/end device* dalam mode AT untuk Xbee yang terdapat pada *node end device* dan *node router 1,2*

serta *Znet 2.5 coordinator* dalam mode AT untuk *node coordinator*. Dalam mengkonfigurasi Xbee *series 2* hal yang terpenting ialah mengisi nilai PAN ID.

Langkah pertama untuk dapat berkomunikasi dalam satu jaringan, maka PAN ID antar Xbee harus diisi dengan nilai yang sama. Selanjutnya perangkat Xbee akan membangun jaringan secara otomatis. *Node coordinator* akan memulai jaringan ZigBee dengan melakukan pencarian PAN ID pada area tersebut dan saluran yang tersedia, kemudian *router* atau *end device* yang berada disekitarnya dapat bergabung dengan *node coordinator*. setelah itu akan terbentuk jaringan ZigBee dengan topologi *Mesh* dimana *node end device* dapat mengirimkan data menuju *node coordinator* secara langsung ataupun melalui *node router* 1,2 secara otomatis yang memungkinkan data itu terkirim melalui jalur manapun.

3.4.6 Rancangan Perangkat Lunak

Perancangan perangkat lunak bertujuan untuk menampilkan sinyal ECG secara visual dan menyimpan data sinyal ECG ke dalam *file* dengan format “.txt”. adapun perancangan perangkat lunak seperti terlihat pada *flowchart* seperti gambar 3.8

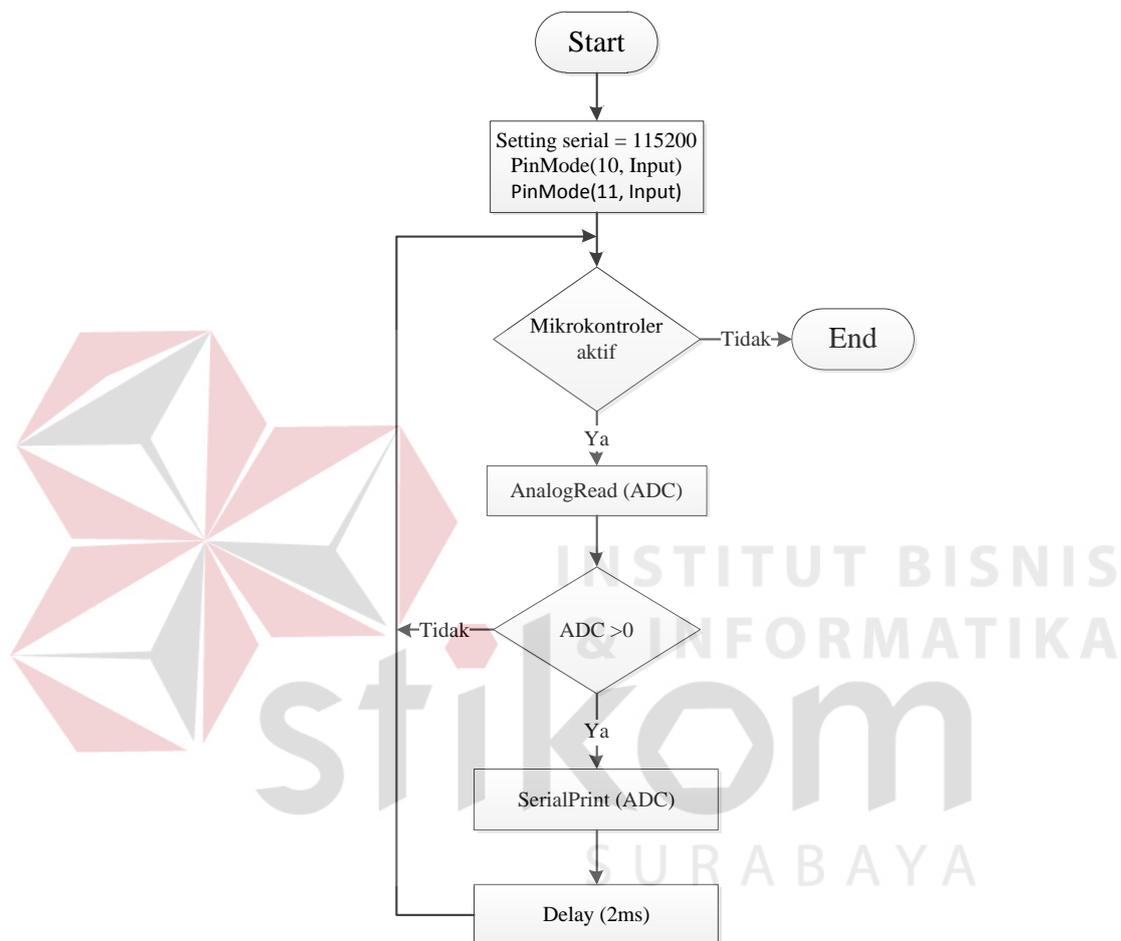


Gambar 3.8 Flow Chart Perancangan Perangkat Lunak

Pada gambar di atas merupakan alur untuk kerja dari perangkat lunak, pembacaan sensor ECG adalah untuk mengambil sinyal dari perangkat ECG untuk ditampilkan secara visual ke layar PC dan menyimpan data sinyal ECG ke dalam database. Jika nilai sensor NULL maka program akan membaca sensor ECG lagi sampai mendapatkan nilai, jika sensor tidak NULL maka nilai tersebut akan diproses untuk disimpan terlebih dahulu kedalam database kemudian dimunculkan secara visual ke layar PC. Program akan mengulang terus menerus selama *User* tidak meminta stop, jika *User* tidak meminta stop maka sinyal ECG yang didapat dan disimpan akan semakin banyak.

3.4.7 Program Membaca Sensor ECG

Diagram alir untuk mengambil sinyal ECG terhadap tubuh pasien berdasarkan pembacaan sensor ECG dan dikirim ke PC untuk ditampilkan dan disimpan data sinyal ECG terdapat pada Gambar 3.9.



Gambar 3.9 Diagram alir Pengiriman Data Auskultasi Jantung

Pada dasarnya konsep dari sensor jantung adalah menerima setiap suara,, kemudian melakukan pengambilan tegangan yang keluar dari tubuh melalui sensor ECG dan data sensor ECG tersebut diolah pada mikrokontroller menjadi data digital melalui pin ADC. Setelah itu dikirim ke Komputer, maka dibuatlah algoritma seperti gambar 3.9. selain dikirim dan ditampilkan pada komputer maka modul Xbee yang juga terpasang pada arduino dan telah di konfigurasi sebagai

end device juga akan mengambil nilai pada serial port kemudian mengirimkan data hasil auskultasi jantung kepada *node coordinator* secara wireless. Lalu kemudian hasil pengiriman data bisa di simpan dan ditampilkan juga pada komputer yang terdapat pada *node coordinator*. Berikut potongan program pembacaan sensor ECG serta mengirim data tersebut ke Komputer :

```

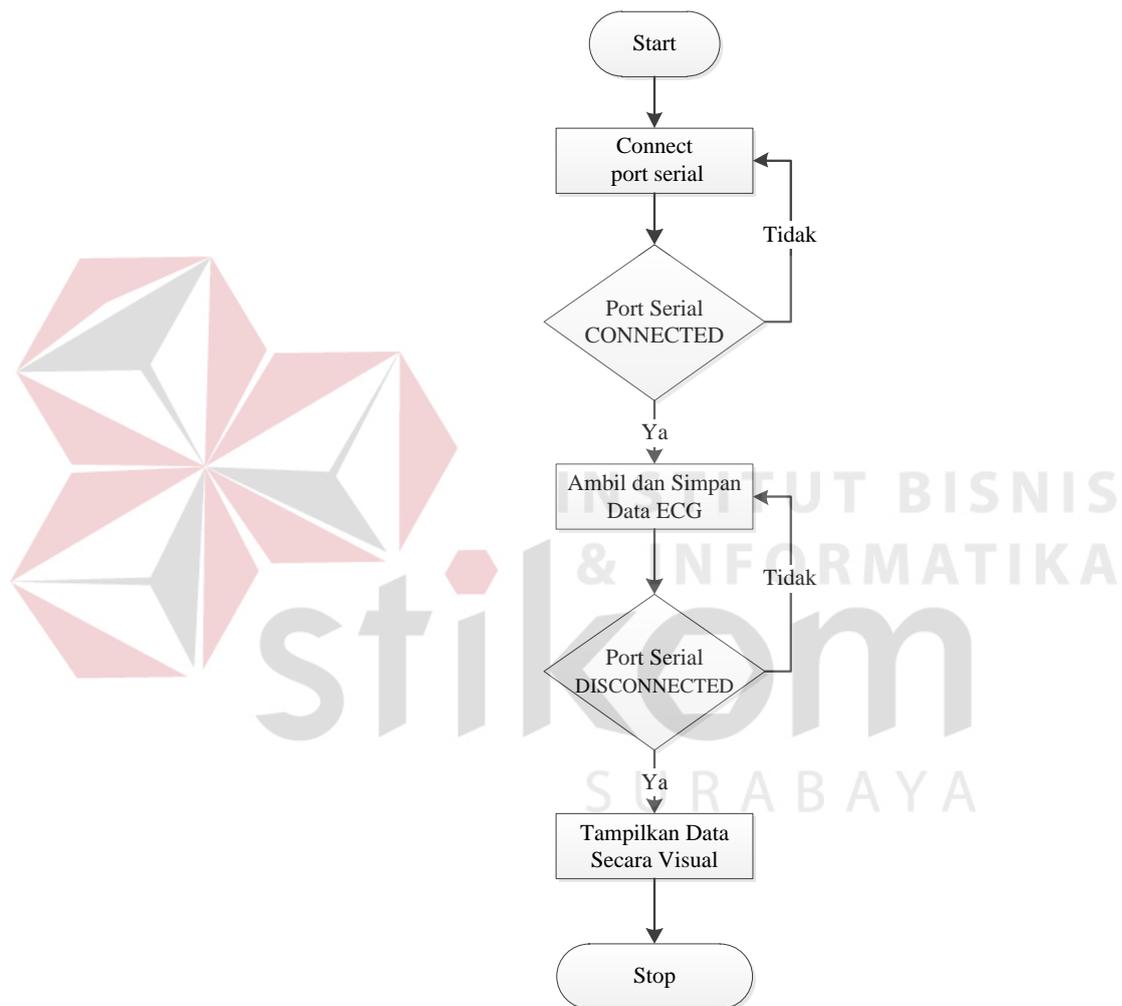
void setup() {
  // initialize the serial communication:
  Serial.begin(115200);
  pinMode(10, INPUT); // Setup for leads off detection LO +
  pinMode(11, INPUT); // Setup for leads off detection LO -
}
void loop() {
  if((digitalRead(10) == 1)|| (digitalRead(11) == 1)){
    Serial.print("%");
    Serial.print(analogRead(A0));
    Serial.println("#");
  }
  else{
    // send the value of analog input 0:
    Serial.print("%");
    Serial.print(analogRead(A0));
    Serial.println("#");  }
  //Wait for a bit to keep serial data from saturating
  delay(2);
}

```

Pemrograman Arduino Mega 2560 merupakan sebuah pemrograman modul, maka pemrograman langsung dilakukan pada setiap pin. Pin 10 dan 11 digunakan sebagai filter untuk leads. Jika salah satu dari pin 10 tau 11 mendapatkan input HIGH, maka lead dinyatakan tidak terpasang.

3.4.8 Program Memunculkan Sinyal ECG di Visual Basic

Pada saat memunculkan sinyal ECG secara *visual* terdapat *flowchart* dimana *flowchart* tersebut merupakan alur dari cara untuk memunculkan sinyal ECG secara *visual* di *Visual Basic*. Di bawah ini pada gambar 3.10 merupakan diagram alir dari pada program *Visual Basic*.

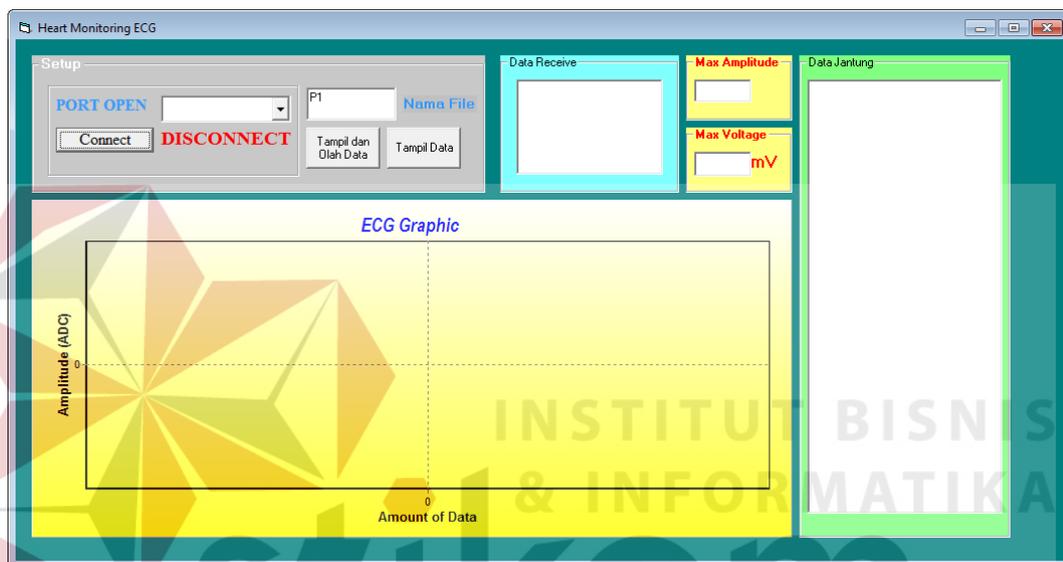


Gambar 3.10 Diagram alir Pengambilan Data ECG

Dari diagram alir di atas, Visual basic pada komputer *end device* berfungsi untuk mengolah data yang dikirimkan oleh mikrokontroler untuk disimpan kemudian ditampilkan secara visualisasi dalam bentuk sinyal. Untuk

memulai pengambilan data adalah harus memilih PORT yang digunakan oleh mikrokontroler, setelah itu koneksikan PORT yang sudah dipilih sebelum. Jika sudah terkoneksi maka pengambilan data ECG langsung disimpan dalam file dalam format “.txt”. Tombol “disconnect” digunakan untuk menghentikan pengambilan data seperti Gambar 3.11.

Berikut adalah tampilan program untuk membaca sinyal ECG :



Gambar 3.11 Tampilan Program ECG di Visual Basic

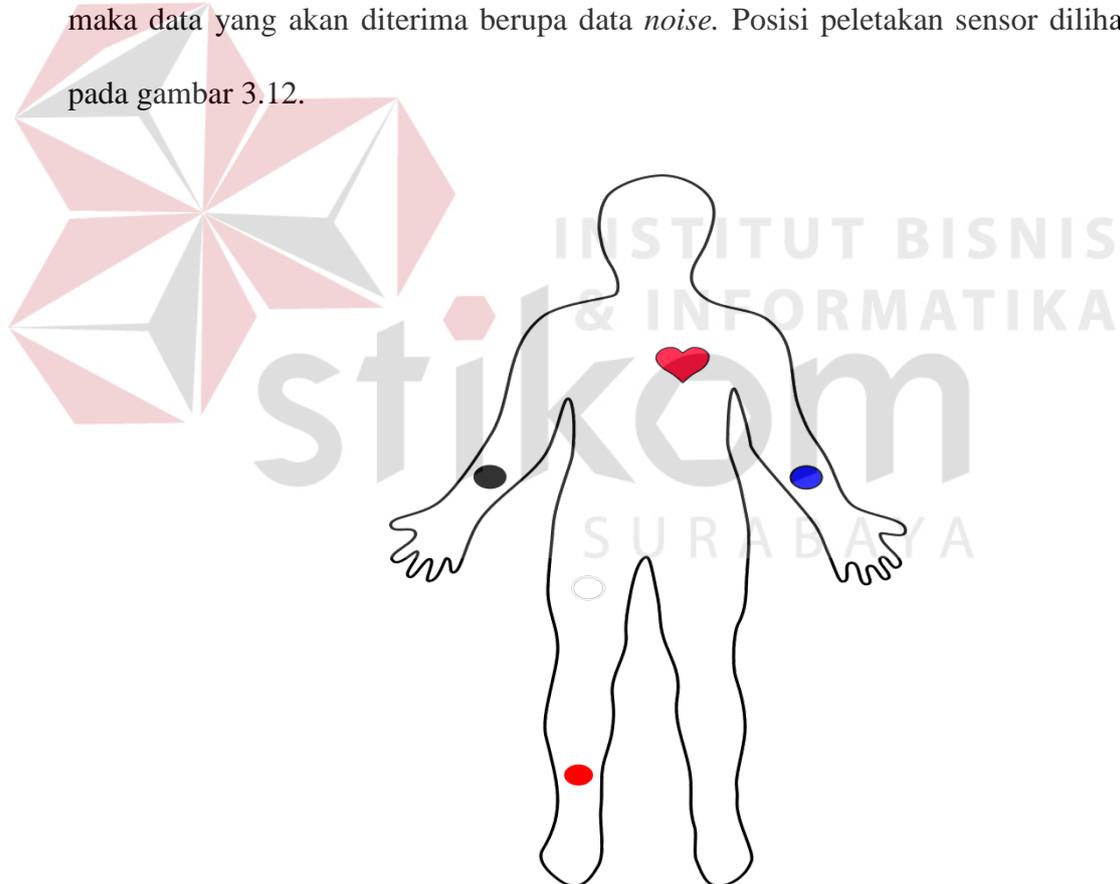
3.5 Metode Analisa

Pada transmisi sinyal auskultasi ini, selain pembuatan algoritma pengiriman data, hal terpenting lainnya adalah analisa dari hasil pengiriman itu sendiri agar dapat diketahui seberapa baik sistem yang telah dibangun.

3.5.1 Peletakan Elektroda Jepit

Dalam transmisi sinyal auskultasi, komponen terpenting adalah data yang diambil dari auskultasi jantung. Maka tahapan yang pertama dilakukan adalah mengambil data auskultasi dengan meletakkan sensor pada pergelangan kedua tangan dan pergelangan kaki kanan.

Posisi jantung manusia adalah pada tulang iga manusia ke 6 di sebelah kiri dada manusia, atau 5 cm di atas ulu hati di sebelah kiri. Peletakan sensor sangat berpengaruh, karena apabila sensor tidak diletakkan pada bagian yang tepat maka data yang akan diterima berupa data *noise*. Posisi peletakan sensor dilihat pada gambar 3.12.



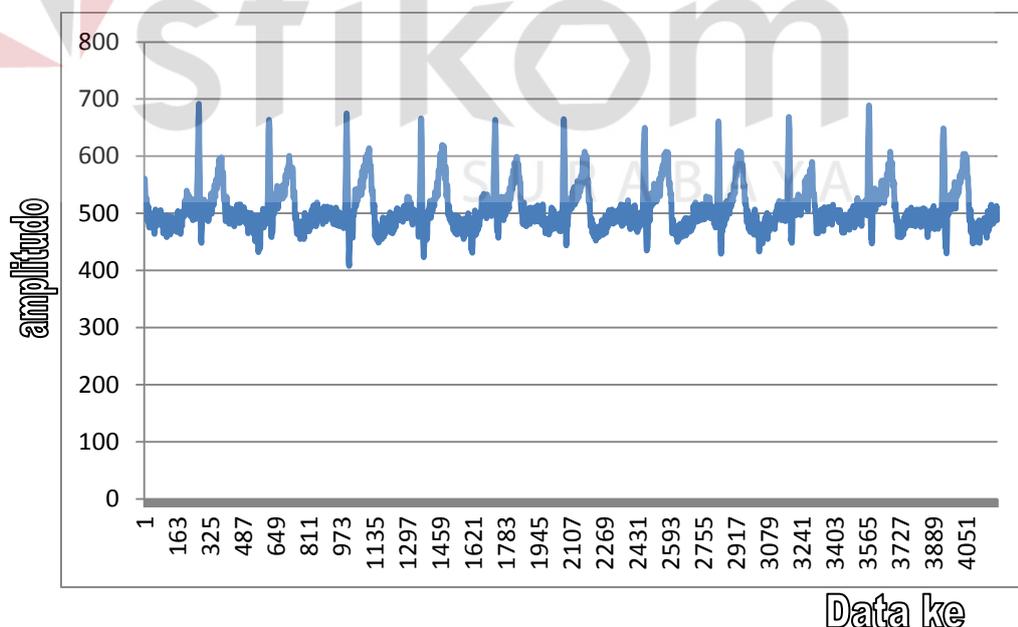
Gambar 3.12 Letak posisi penempatan Elektroda Jepit

3.5.2 Pengambilan Sinyal Auskultasi Jantung

Proses pengambilan data dilakukan saat semua alat terpasang, dan proses pengiriman data berlangsung. Seperti yang dijelaskan di atas, melalui grafik kita dapat melihat apakah sensor sudah berada pada posisi yang tepat.

Data pada masing – masing *end device* dan *coordinator* akan tersimpan pada sebuah file. File inilah yang nantinya digunakan untuk menganalisa seberapa baik sistem dapat mentransmisikan sinyal auskultasi jantung dari *node end device* ke titik *coordinator* secara bersamaan dengan menggunakan topologi mesh.

Pengambilan sinyal jantung dilakukan selama 30 detik untuk mendapatkan hasil transmisi sinyal jantung. Hal ini dikarenakan penerimaan data pada titik *coordinator* lebih lama karena adanya proses pemisahan data yang terjadi pada titik *coordinator*. Contoh sinyal jantung hasil auskultasi terlihat dalam gambar 3.13.



Gambar 3.13 Hasil Sinyal Askultasi Jantung

3.5.3 Analisa Transmisi Sinyal Auskultasi Jantung

Cara menganalisa hasil transmisi jantung adalah dengan memindah data dari file penyimpanan ke file excel. Lalu data dibandingkan antara data inputan dengan data yang berasal dari *receiver*. Posisi data pertama yang sesuai dengan data yang ada pada *transmitter* adalah data yang berhasil dikirim dan diterima. Sehingga dapat diketahui *delay* penerimaan data dari data yang telah dikirim.

Dari semua data yang dikirim terdapat *loss data*, *loss data* dapat diketahui dari data yang tidak sesuai dengan data yang terdapat pada pengiriman, dan dengan mengurangi jumlah data yang terkirim dengan jumlah data yang telah diterima. Maka data yang *loss* akan dapat diketahui.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1													
2	T(S)	T Signal	T(S)	R Signal			legend						
3		560	6,676	560			Start awal data T dan R sama						
4		561		561			Titik awal data T dan R tidak sama atau data hilang						
5		560		560			Jumlah data hilang						
6		559		559									
7		542		542									
8		528		528			Jumlah data hilang =	57					
9		525		525			Jumlah data =	20000					
10		520		520			% data hilang =	0,285					
11		523		523									
12		523		523									
13		520		520									
14		527		527									
15		525		525									
16		513		513									
17		499		499									
18		487		487									
19		486		486									
20		494		494									
21		499		499									
22		503		503									
23		512		512									
24		497		497									
25		483		483									
26		477		477									

Gambar 3.14 Data yang diterima pada waktu 6,676s dari pengiriman

471	471
477	463
480	453
496	481
469	522
467	546
468	572
475	539
494	496
504	461
518	469
474	516
484	494
531	479
543	477
543	465
548	509
560	568
519	609
504	588
482	584
463	581
453	547
481	554
522	453
546	448
	522
	546
	453
	481
	522
	546

Gambar 3.15 Data yang *loss*

Dari pencarian *delay* dan data *loss* seperti pada Gambar 3.14 dan Gambar 3.15 maka dapat dilakukan perhitungan untuk mengetahui berapa *packet loss* yang diterima, berapa lama pengiriman data serta berapa besar *throughput* rata – rata setiap pengiriman data.

a. *Delay*

Setelah menyamakan data antara *transmitter* dan *receiver* maka terdapat selisih urutan antara kedua data tersebut, selisih urutan

tersebutlah yang disebut *delay* dalam pentransmisiian sinyal auskultasi jantung (Gani,2010).

Karena data dikirim setiap 2ms maka jarak antara data tersebut dikalikan dengan waktu pengiriman data, dan akan ditemukan berapa lama data yang dikirimkan oleh *transmitter* diterima oleh *receiver*.

$$Delay = \text{selisih urutan data} \times 0,002$$

b. *Packet Loss*

Pada pencarian *packet loss* seperti pada gambar 3.14 maka akan ditemukan banyak data yang tidak dapat diterima dengan baik oleh *receiver*, jumlah paket yang tidak diterima dengan sempurna tersebut adalah *packet loss* yang digunakan untuk mencari berapa besar persentase data yang hilang.

$$packet\ loss = \frac{\text{jumlah paket hilang}}{\text{jumlah data masuk}} \times 100 \%$$

c. *Througput*

Seperti yang telah dijelaskan pada bab 3, *througput* adalah besar kecepatan data terkirim secara *real*. Maka untuk menemukan *througput* dilakukan dengan cara memasukkan jumlah data diterima selanjutnya dibagi dengan lama waktu pengamatan.

$$Througput = \frac{\text{jumlah data masuk} \times \text{jumlah tiap paket data} \times \text{besar pengiriman data}}{\text{lama pengamatan}}$$

Maksud dari rumus di atas adalah :

- Jumlah data masuk : keseluruhan data yang masuk sebelum melalui proses pengelompokan data
- Jumlah tiap *packet data* : dalam mengirimkan 1 buah paket data terdapat ± 5 *character*
- Besar pengiriman data : sebuah karakter terbentuk dari 10 bit data, yaitu 8 bit untuk setiap *character*, 1 bit pembuka data dan juga 1 bit penutup.

Pengiriman dilakukan dengan *delay* 2ms sesuai dengan ketentuan yang ada, yaitu pengiriman dilakukan minimal dengan 2 kali frekuensi sampling jantung normal (Teori Sampling) (Lynn, 1994).

$$\text{frekuensi sampling} = 2x \text{fn}$$

$$\text{frekuensi jantung}(\text{fn}) = 250\text{Hz} - 500\text{Hz}$$

$$= 2 x 250\text{Hz}$$

$$= 500\text{Hz}$$

$$\text{Perioda Sampling} = \frac{1}{500}$$

$$= 2 \text{ ms}$$

Berdasarkan hasil di atas maka pengiriman data dilakukan setiap frekuensi pengambilan data sebesar 500 pengambilan data per detik.