

# RANCANG BANGUN SISTEM *TRAFFIC COUNTING* DAN DETEKSI KECEPATAN KENDARAAN BERBASISKAN WIRELESS SENSOR NETWORK (WSN)

Sumartini Dana<sup>1</sup> dan Indranata Panggalo<sup>2</sup>

## **Abstrak :**

Permasalahan lalu lintas telah menjadi sebuah masalah yang sangat serius bagi perkotaan. Jaringan jalan yang tidak memiliki konsep jaringan yang memadai, mengakibatkan pilihan rute menuju suatu kawasan menjadi terbatas sehingga beban jalan-jalan tertentu menjadi sedemikian padatnya. Hal ini diperparah dengan jumlah kendaraan yang terus meningkat

Pembuatan sistem untuk klasifikasi, counting kendaraan dan kecepatan rata-rata menggunakan AMR sensor, seluruh pemrosesan dilakukan di mikrokontroler. Metode yang digunakan untuk klasifikasi kendaraan, menggunakan *decision tree* dengan algoritma pembangunnya C4.5. Data result dikirimkan ke komputer menggunakan media *wireless zigbee*, secara *point to point*, tiap 5 menit. *Interface* sederhana di komputer akan menampilkan total volume kendaraan, jumlah klasifikasi kendaraan berdasarkan group dan kecepatan rata-rata. Hasil pengujian sistem untuk klasifikasi kendaraan, berdasarkan besarnya persentasi *error* dibandingkan dengan perhitungan secara manual. Sedangkan untuk pengiriman data, berdasarkan *delay* per jarak dan jarak maksimal pengiriman data yang diterima dengan baik.

**Kata Kunci :** *Traffic Counting, Deteksi Kecepatan Kendaraan, Wireless Sensor Network*

## **I. PENDAHULUAN**

### **1.1 Latar Belakang**

Permasalahan lalu lintas telah menjadi sebuah masalah yang sangat serius bagi perkotaan. Jaringan jalan yang tidak memiliki konsep jaringan yang memadai, mengakibatkan pilihan rute menuju suatu kawasan menjadi terbatas sehingga beban jalan-jalan tertentu menjadi sedemikian padatnya. Hal ini diperparah dengan jumlah kendaraan yang terus meningkat.

Berbagai upaya telah dilakukan oleh pemerintah untuk mengatasi/mengurangi kemacetan, baik dengan efisiensi terhadap jalan yang sudah ada ataupun penambahan model transportasi umum. Proses efisiensi penggunaan

jalan, salah satunya dengan manajemen lalu lintas terhadap infrastruktur yang ada mulai dari pengaturan rute satu jalur, jalur berbayar dan sebagainya. Dalam hal pengaturan rute, dinas terkait akan menempatkan petugas untuk memantau/menghitung tingkat kepadatan lalu lintas di tempat tertentu, sehingga bisa didapatkan sebuah data yang valid untuk pengambilan keputusan selanjutnya, baik itu pengaturan rute maupun penentuan durasi lampu hijau di *traffic light*. Untuk memperoleh data volume kendaraan di jalan raya, sering masih dilakukan secara manual. Metode perhitungan dengan cara ini akan sangat melelahkan bagi petugas dan kemungkinan duplikasi data dapat terjadi.



Salah satu poin penting dalam pengaturan lalu lintas tersebut adalah di persimpangan [1], atau bagaimana cara mengelola lalu lintas di persimpangan yaitu menggunakan *traffic light*. Pada beberapa system *traffic light*, lama penyalaaan lampu hijau dan merah masih statis, dimana saat waktu padat kendaraan maupun sepi kendaraan adalah sama. Pada kota-kota besar telah menggunakan *Automatic Traffic Light Control System (ATCS)*, dimana lama penyalaaan lampu hijau secara otomatis berdasarkan kepadatan lalu lintas. Tingkat kepadatan lalu lintas dihitung secara otomatis menggunakan sensor atau *camera counting*.

Berbagai metode digunakan dalam pembuatan sistem deteksi kendaraan secara otomatis, metode tersebut dibagi dalam dua group yaitu *intrusive* maupun *non-intrusive* [7]. Sensor yang *intrusive*, biasanya langsung dipasang dibadan jalan raya, dengan membuat lubang sebagai tempat peletakan alat/sensor. Contoh yang *intrusive*: *inductive loops*, *magnetometers*, *micro-loop probes*, *pneumatic road tubes*, *piezoelectric*, *weigh-in-motion sensors* dll. Metode *non-intrusive*, tidak dipasang/ditanam di jalan raya tetapi disisi jalan, sehingga mudah dalam instalasi dan perawatan. contohnya : *video image processing*, *acoustic sensor*, *microwave radar*, *ultrasonic sensor*, *active* dan *passive sensor*, *magnetic sensor* dll.

Dalam implementasinya, perlu ada pertimbangan yang matang dalam pemilihan metode, salah satu pertimbangannya yaitu mengenai sumber energinya (listrik), contohnya untuk perhitungan kendaraan berbasis kamera, memerlukan sumber energy yang besar, baik untuk power kamera itu sendiri maupun pencahayaan (malam hari). Pertimbangan yang lain adalah media komunikasi antara sensor dengan pusat sebagai pengolah data. Sehingga dalam penelitian ini, penulis ingin membuat sebuah *embedded system* untuk mengajukan suatu alternative dalam menghitung jumlah kendaraan sekaligus kecepatan kendaraan. System ini menggunakan sensor *anisotropic magneto-resistive (AMR)* dengan media

komunikasi antara sensor dengan *server/komputer* berbasis *wireless*, yaitu menggunakan *zigbee* (802.15.4).

## 1.2 Perumusan Masalah

Berdasarkan latar belakang maka dirumuskan masalah sebagai berikut :

1. Bagaimana merancang dan membuat sebuah sistem yang dapat menghitung volume kendaraan dan kecepatan rata-rata secara otomatis, efisien dengan sumber energi yang kecil (*batterai*) menggunakan AMR Sensor type HMC1001, mikrokontroler ATmega 16.
2. Bagaimana mengirimkan data dari mikrokontroler ke server/komputer menggunakan jaringan *wireless (zigbee/ 802.15.4)*.
3. Membuat tampilan sederhana pada komputer, yang menampilkan volume kendaraan dan kecepatan rata-rata dalam waktu tertentu.

## 1.3 Tujuan Penelitian

Tujuan penelitian ini adalah dapat mengembangkan kinerja sistem menghitung volume kendaraan dan kecepatan rata-rata kendaraan secara otomatis yang efisien dengan sumber energi dari *batterai* menggunakan AMR sensor dan mikrokontroler, dengan media komunikasi ke server/komputer secara *wireless*. Diharapkan hasil penelitian ini juga dapat dijadikan pertimbangan dinas/operator terkait, sehingga meringankan pekerjaan mereka untuk menghitung jumlah kendaraan dan menghitung kecepatan rata-rata kendaraan persatuan waktu secara otomatis. Dan juga sebagai sistem pemberi *input* data bagi *ATCS* secara *real time* sehingga durasi penyalaaan lampu hijau dan merah pada system *traffic light* yang ada menjadi dinamis.

## 1.4 Batasan Masalah

- Proses *counting* kendaraan dan kecepatan rata-rata dilakukan di mikrokontroler. Hasil dikirimkan secara *wireless* ke komputer tiap 5 menit.

- Sistem bekerja untuk pemantauan pada satu lajur saja dan pemasangan sensor diatas jalan (dibawah kendaraan).

## II. TINJAUAN PUSTAKA

### 2.1 Anisotropic Magneto-Resistive (AMR) Sensor

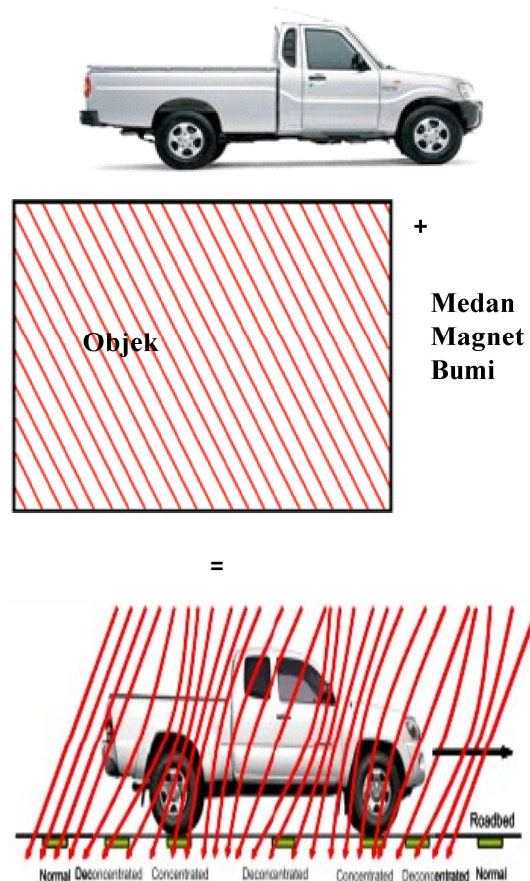
Sensor AMR merupakan salah satu jenis sensor yang baik untuk melakukan penginderaan terhadap perubahan medan magnet bumi, baik itu pengukuran posisi linear dan sudut dan juga perpindahan pada medan magnet bumi. Sensor AMR terbuat dari besi nikel-(*Permalloy*), *thin film* diendapkan pada *wafer silikon* dan berpola sebagai *strip resistif* [4]. Sifat *thin film* ini menyebabkan sensor dapat mengubah resistensi 2-3% dengan adanya medan magnet. Konfigurasi resistor dalam sensor dibuat terhubung membentuk jembatan *wheatstone* sehingga besar dan arah medan magnet pada sumbu tunggal dapat diukur. Dan elemen sensitive digabungkan kedalam jembatan untuk meningkatkan sensitifitas pengukuran, sehingga sensor akan segera bereaksi menyalurkan arus terhadap perubahan medan magnet.

Kemampuan sensor dengan konsumsi daya kecil, biaya yang rendah, sensitivitas tinggi, ukuran kecil, kekebalan terhadap *noise*, mudah dirakit dan kehandalannya membuat sensor ini memiliki banyak keunggulan. Sehingga sensor ini banyak digunakan dalam aplikasi :

- Kontrol *Railroad Crossing* untuk kereta api.
- *Drive Through Retail (Banking, Fast-Food)*
- Pintu otomatis / gerbang otomatis
- Monitoring lalu lintas (kecepatan, arah)
- Deteksi parker
- *Parking Meters*

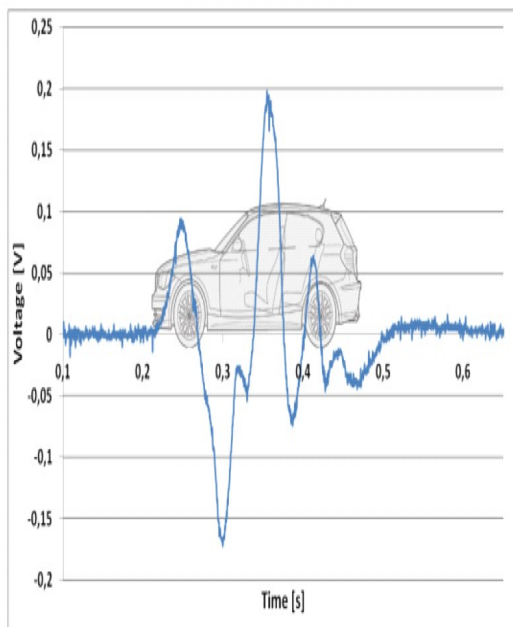
Umumnya kendaraan dibuat dari bahan logam, sehingga saat kendaraan melewati wilayah tertentu maka kendaraan tersebut akan melakukan turbulensi pada medan magnet bumi [5]. Pada Gambar 2.1, menunjukkan sebuah flux medan magnet bumi yang akan terlewati oleh sebuah objek (mobil), ketika objek tersebut memasuki area medan magnet maka flux medan terbut akan mengalami turbulensi, perubahan

turbulensi ini yang akan dibaca oleh sensor. Perubahan turbulensi medan magnet bumi hanya dapat dipengaruhi oleh objek yang mengandung logam contoh mobil, motor, kereta api dan pesawat.



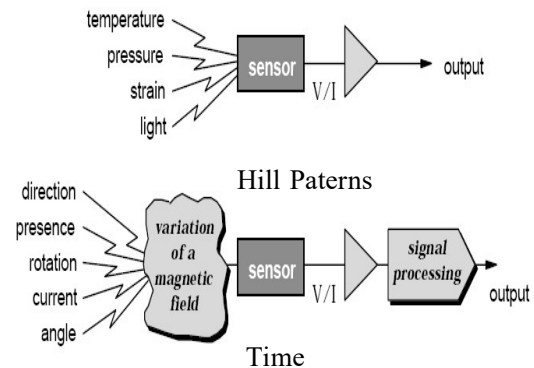
**Gambar 2.1 :** Medan Magnet Bumi Terpengaruh oleh Mobil [4]

Pada Gambar 2.2, merupakan ilustrasi dari sinyal *output* dari sensor AMR. Pada saat kendaraan memasuki area *coverage sensor*, maka sensor akan mengeluarkan *output* sinyal analog dengan nilai amplitude voltage tertentu (berbanding lurus dengan besar objek) terhadap waktu. Sinyal analog dari sensor akan didigitasi pada ADC (Gambar 2.3) untuk mendapatkan parameter tegangan tertentu, selanjutnya diproses pada MCU/Komputer.

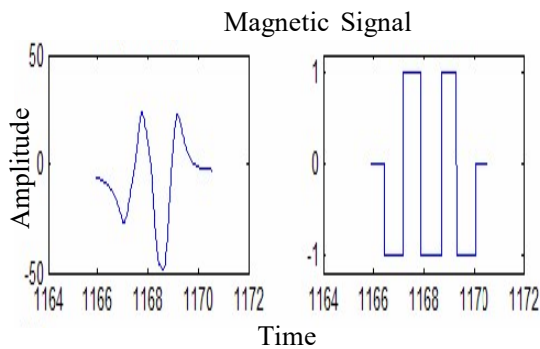


**Gambar 2.2 :** Kurva Output Sensor AMR [8]

parameter-parameter tegangan yang berubah tersebut, sinyal keluaran membutuhkan beberapa pemrosesan sinyal untuk menerjemahkan *output* sensor ke dalam nilai parameter yang diinginkan. Selanjutnya seluruh data digitasi tersebut diproses (MCU/Komputer) sesuai dengan tujuannya dengan algoritma tertentu untuk pemanfaatan selanjutnya.



**Gambar 2.4 :** Conventional vs. Magnetic Sensing [12]



**Gambar 2.3 :** Output Digitasi Kurva di ADC [8]

Pada Gambar 2.4, memberikan ilustrasi perbedaan antara pengukuran menggunakan sensor lain (suhu, tekanan, ketegangan, cahaya) dengan sensor magnetik. *Output* dari sensor konvensional akan langsung melaporkan parameter yang diinginkan. Berbeda dengan sensor magnetik, sensor ini mendeteksi secara tidak langsung dengan mengambil perubahan parameter. Setelah sensor merasakan adanya perubahan medan magnet dan mengeluarkan

## 2.2 Mikrokontroler ATmega16

AVR merupakan seri mikrokontroler *Complementary Metal Oxide Semiconductor* (CMOS) 8-bit buatan Atmel berbasis arsitektur RISC (*Reduced Instruction Set Computer*). Hampir semua instruksi pada program dieksekusi dalam satu siklus *clock*. AVR mempunyai 32 register *general-purpose, timer/counter* fleksibel dengan mode *compare*, interupsi *internal* dan *eksternal*, serial UART, *programmable Watchdog Timer*, *power saving mode*, ADC dan PWM. AVR pun mempunyai *In-System Programmable* (ISP) *Flash on-chip* yang memungkinkan memori program untuk diprogram ulang (*read/write*) dengan koneksi secara serial yang disebut *Serial Peripheral Interface* (SPI).

AVR memiliki beberapa keunggulan yaitu kecepatan dalam mengeksekusi program yang lebih cepat, karena sebagian besar instruksi dieksekusi dalam 1 siklus *clock* (lebih cepat dibandingkan mikrokontroler keluarga MCS 51

yang memiliki arsitektur *Complex Intrukstion Set Compute*), mempunyai *throughput* mendekati 1 *Millions Instruction Per Second* (MIPS) per MHz, sehingga membuat konsumsi daya menjadi rendah terhadap kecepatan proses eksekusi perintah.

Beberapa fitur dari AVR ATMEGA16 antara lain:

1. Mikrokontroler AVR 8 bit yang memiliki kemampuan tinggi dengan konsumsi daya rendah
2. Arsitektur RISC dengan *throughput* mencapai 16 MIPS pada frekuensi 16 MHz
3. Memiliki kapasitas *Flash* memori 16 Kbyte, EEPROM 512 Byte dan SRAM 1 Kbyte
4. Saluran I/O sebanyak 32 buah, yaitu *Port A*, *Port B*, *Port C* dan *Port D*
5. CPU yang terdiri dari 32 buah *register*
6. Unit interupsi dan eksternal
7. *Port* USART untuk komunikasi serial
8. Fitur *peripheral*

- Tiga buah *Timer/Counter* dengan kemampuan perbandingan (*compare*)
  - Dua buah *Timer/Counter* 8 bit dengan *Prescaler* terpisah dan *Mode Compare*
  - Satu buah *Timer/Counter* 16 bit dengan *Prescaler* terpisah, *Mode Compare* dan *Mode Capture*

Untuk menentukan berapa timer value yang harus diisikan ke TCNT agar menghasilkan waktu timer yang diinginkan, dapat dihitung dengan persamaan sbb:

Timer 8 bit:  $TCNT = (FF+1) - ((Ttimer \times fCLK)/N)$

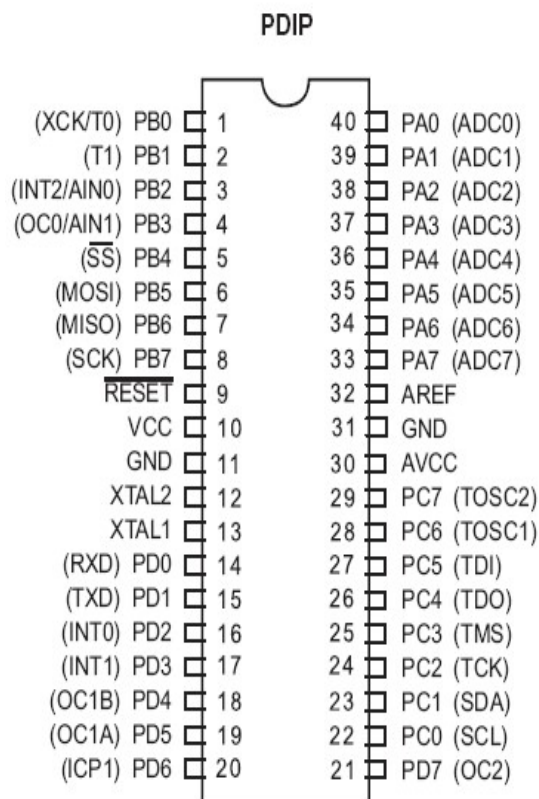
Timer 16 bit:  $TCNT = (FFFF+1) - ((Ttimer \times fCLK)/N)$

- *Real Time Counter* dengan *Oscillator* tersendiri
- Empat kanal PWM
- 8 kanal ADC
  - 8 *Single-ended Channel* dengan keluaran hasil konversi 8 dan 10 resolusi (register ADCH dan ADCL)
  - 7 *Diferrential Channel* hanya pada kemasan *Thin Quad Flat Pack* (TQFP)

➢ 2 *Differential Channel* dengan *Programmable Gain*

- Antarmuka *Serial Peripheral Interface* (SPI) *Bus*
- *Watchdog Timer* dengan *Oscillator Internal*
- *On-chip Analog Comparator*

9. *Non-volatile program memory*



Gambar 2.5 : Konfigurasi Kaki (pin) ATMEGA16 [18]

2.3 Zigbee (802.15.4)

Zigbee merupakan padanan kata Zig dan Bee. Zig berarti gerakan zig-zag dan Bee berarti lebah. Zigbee memiliki sifat komunikasi yang mirip dengan komunikasi diantara lebah yang melakukan gerakan-gerakan tidak menentu dalam menyampaikan informasi adanya madu ke lebah yang satu ke lebah yang lainnya . Spesifikasi Zigbee adalah untuk protokol komunikasi tingkat tinggi yang menggunakan radio digital yang kecil dan berdaya kecil.

ZigBee mengacu pada standard IEEE 802.15.4 (2003) yang berhubungan dengan *wireless personal area networks (WPANs)*. Contoh WPANs antara lain *wireless headphones* yang terhubung dengan telepon genggam melalui radio jarak dekat.

Teknologi yang memenuhi spesifikasi ZigBee dimaksudkan untuk membuat lebih simpel dan tidak lebih mahal dari WPANs lain, seperti *Bluetooth*. Zigbee difokuskan pada penggunaan *radio frequency* yang membutuhkan kecepatan transfer rendah, hemat daya, dan jaringan yang aman.

Tujuan utama penggunaan Zigbee adalah untuk penggunaan umum, tidak mahal, dan dapat mengatur jaringan yang dapat digunakan untuk pengendali industri, *embedded sensing*, kumpulan data medis, sistem alarm keamanan, dan lain-lain. Contohnya antara lain :

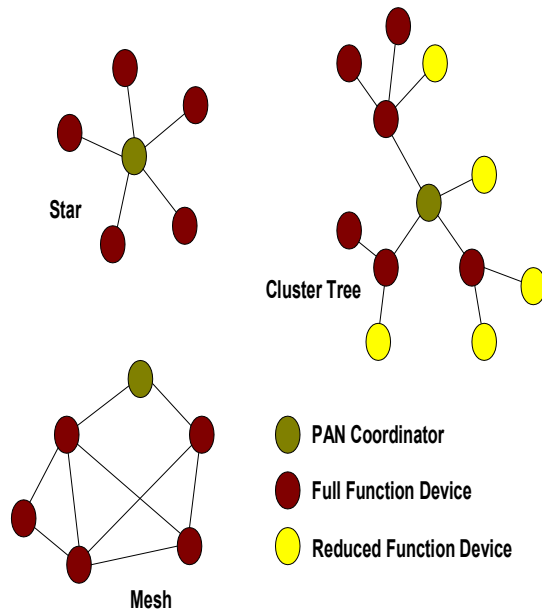
- *Home entertainment* dan kontrol : sistem penerangan cerdas, kontrol suhu dan keamanan.
- Layanan bergerak : *m-payment*, monitoring dan kontrol.
- Bangunan komersial: monitoring energi, penerangan dan kontrol akses.
- Pabrik/industri : kontrol proses, manajemen aset, manajemen energi.

Gambar 2.6 adalah gambar zigbee merk Xbee pro series 2, yang merupakan produk dari Digi.



**Gambar 2.6 :** XBee Seri 2 Wire Antenna

Zigbee memiliki 3 topologi model jaringan yaitu topologi *Star*, *Mesh (Peer to Peer)* serta *Cluster Tree*. Gambar 2.8, jenis topologi jaringan pada zigbee.



**Gambar 2.7 :** Topologi Jaringan pada Zigbee [11]

- Pada topologi *star* komunikasi dilakukan antara perangkat dengan sebuah pusat pengontrol tunggal, disebut sebagai koordinator PAN (*Personal Area Network*).
- Dalam topologi *mesh/peer to peer* juga hanya ada satu koordinator PAN. Berbeda dengan topologi *star*, setiap perangkat dapat berkomunikasi satu sama lain sepanjang ada dalam jarak jangkauannya. *Peer to peer* dapat berupa *ad hoc*, *Self-organizing* dan *self healing*.
- Cluster tree menggunakan gabungan topologi *star* dan topologi *mesh*, dengan menggabungkan keduanya mempunyai keuntungan keandalan tingkat tinggi.

Zigbee dapat melakukan komunikasi dengan 65000 node zigbee dalam waktu yang bersamaan dengan metode komunikasi *multihop ad-hoc* tanpa harus melakukan pengaturan apa pun padanya. Jenis komunikasi dalam bentuk bintang (*star*) maupun pohon (*tree*) dapat dilakukan sesama zigbee tanpa memerlukan *base station* atau *access point*, sehingga dapat melakukan komunikasi secara acak (*mesh network*).

Karena *zigbee* memiliki protocol stack yang sangat sederhana, *zigbee* dapat mengirimkan data sepanjang 127 huruf (127 *byte*) saja. Data sekecil ini sangat bisa dikirim oleh *zigbee* yang memiliki kecepatan pengiriman 250kbps, dan dapat mengurangi beban *host* CPU. Bila pada *bluetooth* minimal dibutuhkan komputer mikro 32 bit, maka dengan *zigbee* hanya membutuhkan komputer mikro 4 atau 8 bit saja. Selain itu, *bluetooth* membutuhkan waktu inisialisasi sistem puluhan detik, sementara *zigbee* hanya membutuhkan 30ms (sedangkan dari kondisi tidur/sleep sampai bangunnya hanya perlu 15 ms) saja. Ini sangat cocok untuk peralatan-peralatan sensor yang membutuhkan operasi kecepatan waktu *ON/OFF*-nya tinggi.

## 2.4 Metode Deteksi Kendaraan

Beberapa penelitian telah dilakukan untuk mendeteksi kendaraan. Pendeteksian kendaraan biasanya untuk *counting*, klasifikasi, *plate recognition*, kecepatan dan *parking*. Khusus penelitian yang sekaligus *counting* dan kecepatan rata-rata pada sebuah lajur, juga telah dilakukan, baik itu yang berbasis kamera maupun sensor dengan metode/algorithm yang berbeda-beda. Model-model tersebut memiliki kelebihan dan kekurangan.

Untuk *counting* dan kecepatan kendaraan yang menggunakan sensor induktif, ada beberapa jenis sensor yang pernah digunakan dalam penelitian tersebut. Berdasarkan sifat masing-masing sensor, maka pemasangan/instalasi di lapangan akan mengikuti karakteristik sensor tersebut. Dibawah ini, penulis sedikit menjelaskan contoh penelitian yang menggunakan *magnetic sensor*.

### 2.4.1 Counting Kendaraan

Dalam penelitian yang dilakukan oleh Sing Yiu Cheung dkk, untuk *counting* kendaraan sangat bagus karena mencapai 99%. Error 1% tersebut karena kendaraan yang datang tidak

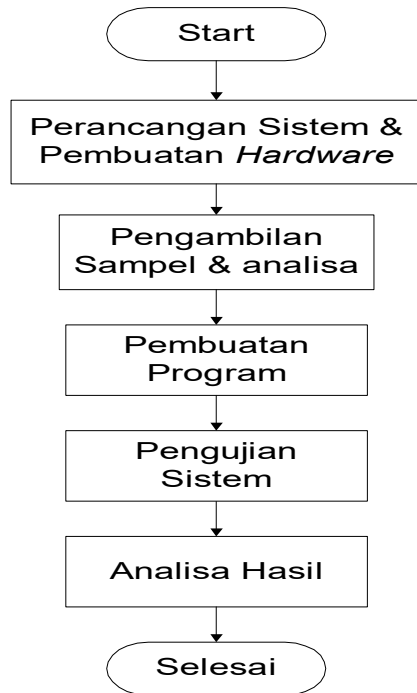
melewati tepat diatas sensor yang dipasang di jalan. Ini menunjukkan bahwa AMR sensor tersebut sangat sensitive, dimana dapat mendeteksi keberadaan kendaraan dengan baik. Tetapi yang menjadi kekurangan dalam penelitian tersebut, tidak menganalisis prediksi jarak maksimal terdekat antara kendaraan yang saling menyusul. Disamping itu jenis sensor yang digunakan dari tipe mempunyai sudut pengambilan magnet pada bidang Z dan X, sedangkan yang akan digunakan oleh penulis dalam penelitian ini hanya bidang Z.

### 2.4.2 Kecepatan Kendaraan

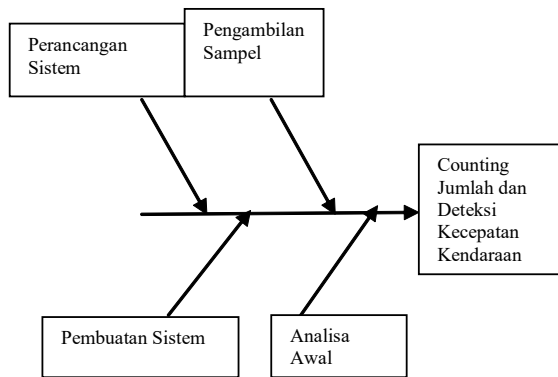
Untuk mendeteksi kecepatan kendaraan menggunakan sensor, maka biasanya menggunakan 2 buah sensor yang dipasang secara garis lurus mengikuti arah jalan, dengan jarak tertentu antara sensor. Saat kendaraan terdeteksi sensor 1 maka perhitungan waktu dimulai. Timing akan berhenti saat kendaraan telah terdeteksi di sensor 2. Berdasarkan perbandingan jarak antar sensor dan waktu yang dibutuhkan kendaraan dari sensor 1 mencapai sensor 2, maka akan didapatkan kecepatan kendaraan tersebut. Pada penelitian terdahulu yaitu *Taghvaeeyan dan Rajamani* [12], jarak antara sensor yaitu 0,9 meter dan *Ondrej Karpis* [2] pada jarak 2 meter. Dalam penelitian ini, penulis akan mencari jarak yang tepat antara sensor agar dapat disesuaikan dgn system penghitungan jumlah kendaraan yaitu mendapatkan jarak maksimal terdekat antara kendaraan yang saling menyusul.

## III. METODE PENELITIAN

Pembuatan sistem *counting* jumlah kendaraan dan kecepatan rata-rata akan terdiri dari beberapa tahapan. Tahapan tersebut adalah perancangan dan pembuatan sistem, pengambilan sampel dan analisa, pembuatan program, pengujian sistem, analisa hasil. Ke-4 tahapan implementasi ini dapat digambarkan pada gambar 3.1.



Gambar 3.1 : Flowchart Tahapan Implementasi Penelitian



Tahapan implementasi tersebut dapat dirumuskan secara rincinya sebagai berikut:

### 3.1 Perancangan dan Pembuatan Sistem

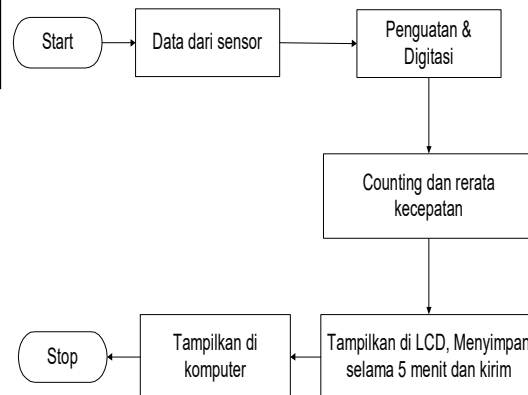
Pada tahapan perancangan, sesuai dengan tema penelitian yaitu untuk membuat sebuah *counting* jumlah kendaraan dan kecepatan rata-rata kendaraan yang efisien, maka penulis harus merancang dengan baik mengenai *hardware*,

aplikasi maupun metode/algorithm yang bekerja dalam sistem. Diharapkan perpaduan ketiga hal tersebut dapat memberikan hasil yang maksimal. Maka langkah pertama dalam tahapan ini yaitu, membaca banyak paper/artikel yang terkait dengan tema penelitian.

Berdasarkan referensi dan konsultasi berbagai pihak, maka penulis dapat mengambil beberapa kesimpulan mengenai komponen-komponen yang akan digunakan, contohnya pemilihan jenis sensor, jenis mikrokontroler, jenis perangkat *wireless*, metode/algorithm klasifikasi kendaraan dan software/pemrograman yang akan digunakan. Penulis membuat kesimpulan secara garis besar mengenai alat yang akan digunakan, yaitu : AMR Sensor, Mikrokontroler ATmega16, LCD 2x16, XBee Series 2, laptop. *Software*/pemrograman adalah bahasa C (untuk mikrokontroler), XCTU (untuk XBee pro), VB6.0 (tampilan di server).

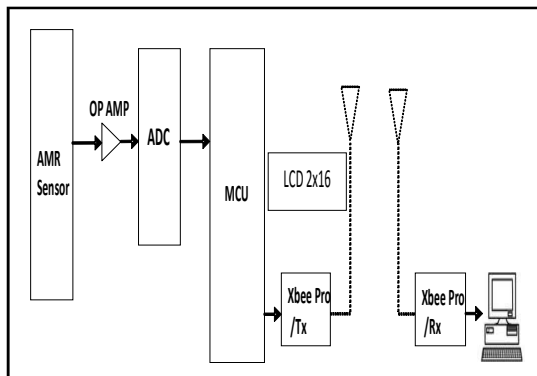
Langkah selanjutnya dalam tahapan ini, yaitu membuat rancangan arsitektur sistem, model rangkaian dan rancangan langkah-langkah pengujian sistem. Gambar 3.2, merupakan gambaran kerja sistem secara garis besar yang menghubungkan seluruh perangkat tersebut.

Gambar 3.2 : Flowchart Kerja Sistem Secara Otomatis



Gambar 3.3 dibawah merupakan gambaran arsitektur sistem secara garis besar yang menghubungkan perangkat *hardware*.



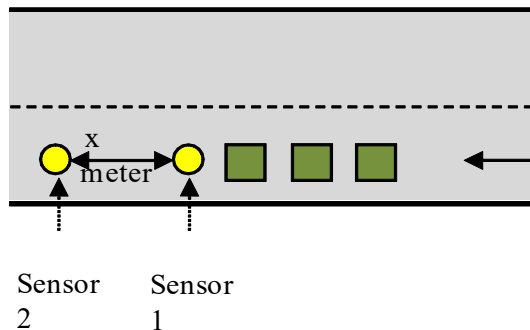


Gambar 3-3 : Arsitektur Sistem

Sensor akan menangkap turbulensi medan magnet akibat adanya kendaraan yang lewat. *Output* sensor, sinyal analog berupa nilai tegangan per satuan waktu. Sinyal yang lemah tersebut akan dikuatkan di *op-amp* dan digitasi di ADC. Selanjutnya data diproses di mikrokontroler untuk melakukan *counting* kendaraan dan menghitung kecepatan kendaraan tersebut. *Result*-nya akan ditampilkan di LCD dan dikirimkan ke komputer menggunakan Xbee Pro.

Dalam perancangan sistem ini, penulis juga akan mempertimbangkan kapan sensor mulai melakukan pengambilan data dari kendaraan yang lewat didepannya dan kapan sensor berhenti melakukan pengambilan data. Oleh karena sifat sensor yang sangat sensitif terhadap perubahan medan magnet bumi, maka sesaat kendaraan memasuki *coverage area* sensor, arus akan mengalir dan terbaca di *pin* mikrokontroler sehingga proses pengambilan data dilakukan. Mikrokontroler akan berhenti melakukan pengambilan data di *pin input* ketika nilai tegangannya bernilai 0 Volt selama 30 ms.

Beberapa metode pemasangan sensor, khusus untuk AMR sensor, ada yang ditanam jalan (dibawah kendaraan) dan ada juga yang dipasang ditepi jalan. Berdasarkan referensi, tingkat kebutuhan dan ketersediaan sensor di pasaran Indonesia maka dalam penelitian ini, pemasangan sensor ditempatkan (ditanam) di jalan, Gambar 3.4.



Gambar 3-4 : Pemasangan Sensor di Jalan

Untuk komunikasi data, seperti yang dijelaskan sebelumnya, bahwa media komunikasi yang digunakan adalah *wireless* dari tipe *zigbee* (802.15.4) dengan kemampuan *data rate* 250 Kbps. Model yang akan digunakan adalah produk dari *Digi*, yaitu *Xbee seri 2*, jarak maksimal berdasarkan *data sheet* 200 meter.

Karena masih dalam tahap awal pengujian, sehingga model komunikasi yang digunakan dalam penelitian ini adalah *point to point*, yaitu hanya menggunakan 2 buah *Xbee*. Jika kiranya hasilnya memberikan hasil yang baik, maka dapat digunakan beberapa *Xbee* yang dihubungkan secara *point to multipoint* dari beberapa titik pengukuran, sehingga datanya merupakan akumulasi dari semua jalur pada jalan raya.

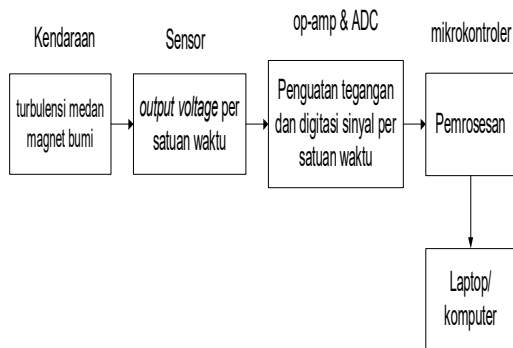
Setelah melakukan perancangan system, maka akan masuk ke tahapan pembuatan system/*hardware* sesuai dengan rancangan. Yaitu melakukan perakitan komponen tersebut menjadi satu kesatuan. Selanjutnya membuat program sederhana untuk melakukan pengujian terhadap *hardware*, apakah telah bekerja dengan baik atau belum, baik itu memunculkan data di LCD dan pengiriman data ke laptop melalui *zigbee*.

### 3.2 Pengambilan sampel

Pengambilan sampel adalah mengambil model gelombang kendaraan dari beberapa jenis kendaraan (sesuai dengan golongan) menggunakan sensor. Data tersebut akan dianalisa secara *offline*, karena data *output* dari



ADC dalam bentuk nilai-nilai tegangan yang membentuk suatu gelombang, maka data tersebut bisa dimasukkan ke *matlab* atau *excel* untuk dilihat pola bentukan gelombangnya. Gambar 3.5, merupakan gambaran proses pengambilan data *acquisition* magnet.



Gambar 3.5 : Flowchart Pengambilan Data Acquisition Magnet

Proses pengambilan sampel ini, akan dibuatkan program yang akan menampilkan data di komputer berupa nilai-nilai *amplitude* saat mendeteksi kendaraan. Dari nilai-nilai tersebut, digunakan sebagai acuan bagi sistem mulai bekerja pada nilai *amplitude* tertentu.

Perhitungan kecepatan, dengan membandingkan antara jarak 2 sensor dan perhitungan estimasi waktu kendaraan dimulai saat terdeteksi di sensor 1 dan *timing*-nya berhenti saat terdeteksi di sensor 2.

### 3.3 Pembuatan Program

Setelah mendapatkan data-data sampel kendaraan (minimal tiap golongan kendaraan memiliki 1 sampel), maka langkah selanjutnya adalah menganalisa data-data primer tersebut dan membuat *decision tree* menggunakan algoritma C4.5 berdasarkan nilai *amplitude peak*, *amplitude mean* dan panjang gelombang. Tujuan penggunaan algoritma ini untuk memudahkan penulis membuat program karena terstruktur, dan juga untuk menghindari *overlapping* data yang dapat menyebabkan pemborosan memori. Setelah membangun *decision tree* tersebut, maka membuat program

menggunakan bahasa C, mengikuti aliran *decision* tersebut dan keseluruhan sistem sesuai dengan rancangan.

### 3.4 Pengujian Sistem

Ada beberapa langkah dalam proses pengujian sistem, yaitu :

1. Pengujian presentasi keberhasilan sistem melakukan klasifikasi, counting dan menghitung kecepatan rata-rata kendaraan. Persentase keberhasilan tersebut dibandingkan dengan pengamatan/perhitungan secara manual. Data hasil klasifikasi dan kecepatan rata-rata akan ditampilkan di layar LCD 2x16, dan juga di layar komputer. Gambar 3.7, contoh tampilan data yang akan ditunjukkan di layar LCD dan komputer.

A	B	C	Vrata
1	2	3	45,5

a) Tampilan di LCD

2. Menghitung berapa waktu yang dibutuhkan, sesaat kendaraan melewati sensor 1 sampai mikroprosesor memberikan hasil. Tujuan pengujian ini untuk mendapatkan jarak minimal antara kendaraan yang saling beriringan.

Klasifikasi, Counting dan	
Gro	1
Gro	2
Gro	3
Volume	6
Kecepatan	45.5

b) Tampilan di Komputer

Gambar 3-7 : Tampilan Sederhana di LCD dan Komputer

3. Pengujian komunikasi data antara mikrokontroler dengan laptop.

Hal ini akan dilakukan dengan mengubah jarak pengiriman antara Xbee Tx dan Xbee Rx. Yaitu sampai pada jarak maksimal Xbee tersebut mampu mengirim sinyal dan menerima dengan baik. Pada pengujian ini yang akan diamati adalah tingkat *delay data* dan *error data*. Untuk *delay*, secara otomatis akan muncul di *interface* pada komputer (waktu pengiriman dan kedatangan data), sedangkan pengujian *error data* dengan membandingkan data di LCD dengan data yang terkirim.

#### IV. ANALISA HASIL

Pada tahapan ini penulis akan menganalisis data-data hasil pengujian yang disajikan dalam bentuk tabel-tabel. Berdasarkan presentasi keberhasilan sistem, maka penulis dapat membuat kesimpulan mengenai kelebihan/kelemahan sistem dan faktor penyebabnya.

##### 3.6 Spesifikasi Perangkat

Dibawah ini spesifikasi perangkat yang akan digunakan dalam membangun sistem klasifikasi kendaraan yang berbasis *wireless* :

##### 1. Hardware

- AMR Sensor (2 buah)
- Minsys ATmega32 (1 Set)
- Op-amp (2 buah)
- LCD 2x16 (1 buah)
- Xbee USB Adapter (1 buah)
- Xbee Explorer Regulated (1 buah)
- Xbee Pro 63mW Wire Antenna - Series 2B (2 buah)
- Beberapa komponen elektronika pendukung (resistor, transistor kabel, konektor dll)
- Laptop Lenovo, processor centrino, DDR2 2GB.

##### 2. Software/pemrograman

Simulasi ini harus berjalan baik pada spesifikasi software sebagai berikut:

- Sistem Operasi : *Windows7*

- *WinAVR*
- *XCTU*
- *Visual Basic 6.0*

#### V. KESIMPULAN

1. Pembuatan sistem untuk klasifikasi, counting kendaraan dan kecepatan rata-rata menggunakan AMR sensor, seluruh pemrosesan dilakukan di mikrokontroler.
2. Metode yang digunakan untuk klasifikasi kendaraan, menggunakan *decision tree* dengan algoritma pembangunnya C4.5.
3. Data result dikirimkan ke komputer menggunakan media *wireless zigbee*, secara *point to point*, tiap 5 menit.
4. *Interface* sederhana di komputer akan menampilkan total volume kendaraan, jumlah klasifikasi kendaraan berdasarkan group dan kecepatan rata-rata.
5. Hasil pengujian sistem untuk klasifikasi kendaraan, berdasarkan besarnya persentasi *error* dibandingkan dengan perhitungan secara manual. Sedangkan untuk pengiriman data, berdasarkan *delay* per jarak dan jarak maksimal pengiriman data yang diterima dengan baik.

#### DAFTAR PUSTAKA

- [1] Ondrej Karpis, "Sensor for Vehicles Classification", in Proc of the Federated Conference on Computer Science and Information Systems pp. 785–789, 2012.
- [2] Giuseppe Anastasi, Marco Conti, Mario Di Francesco, Vincenzo Neri, "Reliability and Energy Efficiency in Multi-hop IEEE 802.15.4/ZigBee Wireless Sensor Networks", 2010.
- [3] Hang Liu, Shin-Ting (Cindy) Jeng, Yeow Chern Andre Tok, Stephen G. Ritchie, "Commercial Vehicle Classification using Vehicle Signature Data", Paper submitted to the 88th Annual meeting of the Transportation Research Board, January 11-15, 2009, Washington D.C.



- [4] S. Cheung and P. Varaiya, "Traffic Surveillance by Wireless Sensor Networks: Final Report", California PATH Program, Inst. Transp. Stud., Univ. California, Berkeley, UCB-ITS-PRR-2007-4, 2007.
- [5] JUN XING and Qi Zeng, "A Model Based Vehicle Detection and Classification Using Magnetic Sensor Data", Chalmers University of Technology, Göteborg, Sweden, 2007.
- [6] Sing Yiu Cheung, Sinem Coleri, Baris Dunder, Sumitra Ganesh, Chin-woo Tan, Pravin Varaiy, "Traffic Measurement and Vehicle Classification with a Single Magnetic Sensor" For Presentation and Publication 84th Annual Meeting Transportation Research Board, January 2005, Washington, D.C.
- [7] A. Daubaras, M. Zilyys, "Vehicle Detection based on Magneto-Resistive Magnetic Field Sensor", Department of Electronics Engineering, Kaunas University of Technology, 2012.
- [8] P. Ghislain, D. Carona, A. Serrador, P. Jorge, P. Ferreira, J. Lopes, "Wireless Magnetic Based Sensor System For Vehicles Classification", Instituto Superior de Engenharia de Lisboa, Lisbon, Portugal.
- [9] Veronica Sri Moertini, "Pengembangan Skalabilitas Algoritma Klasifikasi C4.5 dengan Pendekatan Konsep Operator Relasi", Disertasi, Institut Teknologi Bandung, 2007.
- [10] Matthew N. Anyanwu, Sajjan G. Shiva, "Comparative Analysis of Serial Decision Tree Classification Algorithms", International Journal of Computer Science and Security, (IJCSS) Volume (3) : Issue (3).
- [11] Faludi, Robert, "Building Wireless Sensor Networks", O'reilly, 2011
- [12] Michael J. Caruso, Lucky S. Withanawasam, "Vehicle Detection and Compass Applications using AMR Magnetic Sensors", Honeywell, SSEC, 12001 State Highway 55, Plymouth, MN USA 55441
- [13] Datasheet : Honeywell, "1- and 2-Axis Magnetic Sensors HMC1001 / 1002", Available: <http://www.datasheetarchive.com/HMC1001-datasheet.html>
- [14] Tempo online (28 Juni 2012) : Pertumbuhan Jalan Kalah Cepat dengan Kendaraan, <http://www.tempo.co>, April 2013.
- [15] Tempo online (24 Maret 2013) : Macet Terus, Jakarta Rugi Rp 65 Triliun per Tahun, <http://www.tempo.co>, April 2013.

LAMPIRAN

Pengujian Alat

