

Sistem Pencacah berbasis Arduino-RFID

Eddy Nurraharjo

Program Studi Teknik Informatika, Universitas Stikubank Semarang

Email: eddynurraharjo@gmail.com

Abstrak

Sistem deteksi RFID memiliki beberapa kelebihan untuk proses pengambilan data uniknya secara otomatis atau pengenalannya terkait dengan obyek tertentu, dibandingkan dengan penggunaan sistem lainnya, sehingga kesempatan penelitian saat ini adalah mencoba untuk melakukan penambahan fitur terhadap sebuah penelitian sebelumnya sistem berbasis Arduino dengan menggunakan sensor frekuensi radio, berupa fitur pencacah. Implementasi penelitian ini dengan menggunakan RFID RC522, yang merupakan modul siap pakai dan kompatibel dengan Arduino UNO R3.

Sistem pencacahan ini dirancang sedemikian rupa dengan menggunakan masukan dari 2 tipe *tag* yaitu sebuah kartu kosong dan sebuah gantungan kunci. Sistem diharapkan akan mampu memberikan deteksi terhadap ketiga masukan *tag* tersebut, dan divisualisasikan keluarannya berupa LED.

Pada hasil realisasinya, sistem pencacah terhadap perangkat identifikasi frekuensi radio ini dapat dirakit dengan baik dengan beberapa perangkat *library* dari Arduino, yaitu *AddicoreRFID.h*, yang akan digunakan sebagai modul pengendali untuk perangkat RFID, dan *SPI.h* yang digunakan untuk melakukan komunikasi secara serial melalui *port* USB. Sistem pencacahan identifikasi dibentuk dengan menggunakan fungsi *IF ...*, yang akan menentukan kriteria masukan *tag* dan memberikan batasan jumlah cacahan, namun hingga penelitian ini masih diperlukan fitur dengan kemampuan maksimal dalam menangani sejumlah data yang lebih besar.

Sistem dasar ini telah mampu untuk memberikan hasil pencacahan identifikasi dari kedua tipe *tag* yaitu masing-masing adalah "BLANK MAN" dan "BLUE KEY".

Kata kunci: arduino, mikrokontroler, sistem kendali

PENDAHULUAN

Kemampuan teknologi RFID dengan keunggulannya dalam proses pendataan dibandingkan dengan teknologi lainnya mengalami perkembangan dalam pemanfaatannya. Fitur yang pernah dicoba sebelumnya adalah upaya untuk mendeteksi *tag* RFID menggunakan modul RFID-RC522 yang terhubung dengan koneksi pengolah utamanya adalah arduino UNO R3. Penelitian sebelumnya memperoleh hasil berupa data yang terdeteksi terhadap klasifikasi terhadap perangkat RFID yang tersedia, yaitu berupa kartu kosong dan gantungan kunci.

Sistem tersebut telah berjalan dengan baik namun terkendala terhadap perulangan dalam hal penggunaan perangkat tersebut pada periode tertentu. Hal ini mengakibatkan adanya duplikasi data yang tidak bermanfaat bagi sistem. Untuk itu diperlukan fitur tambahan pembatas terhadap setiap hasil pencacahan dari perangkat RFID yang tersedia.

Klasifikasi dilakukan terlebih dahulu dengan menangkap dan menentukan data *default* yang telah tersedia pada perangkat identifikasi. Kemudian hasil pengenalan data tersebut akan menentukan jumlah cacahannya. Hasil cacahan ini pun diperikan upaya untuk membatasinya agar tidak mempengaruhi batas yang diperlukan dan batas yang semestinya, sehingga akan meningkatkan kinerja sistem. Berasal dari harapan terhadap sebuah model ini maka muncullah ide dengan mengangkat tema Sistem Pencacahan berbasis Arduino-RFID.

PERUMUSAN MASALAH

Perumusan masalah yang terkait dengan tema tersebut adalah penerapan teknik pencacahan terhadap data identifikasi berbasis frekuensi radio dari perangkat RFID, dan memberikan pembatasan terhadap pengelolaan data yang diperlukan.

BATASAN MASALAH

Batasan masalah penelitian ini adalah:

1. Penggunaan modul Arduino UNO R3 yang simpel dan murah, sebagai pemroses data.
2. Penggunaan pembangkit sinyal frekuensi radio dengan tipe RFID-RC522 sebagai pemberi data masukannya.

TUJUAN DAN MAFAAT PENELITIAN

Adapun tujuan dalam penelitian ini adalah:

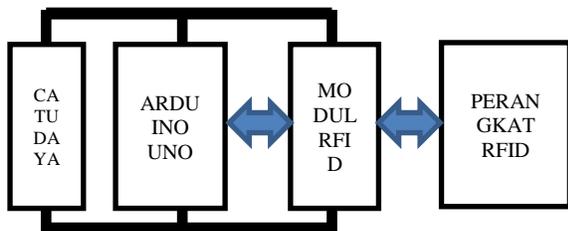
1. Rancang bangun sistem pencacah berbasis Arduino UNO-RFID
2. Implementasi logika fungsi *if* ...

METODE PENELITIAN

Upaya yang dilakukan untuk memperoleh data adalah:

1. Studi literatur, untuk mengetahui karakter perangkat RFID, Arduino dan pembangkit frekuensi radio RFID-RC522
2. Prinsip waterfall, yang dimulai dari persiapan bahan dan data, pemrograman, implementasi, uji coba, perbaikan dan hasil akhir.

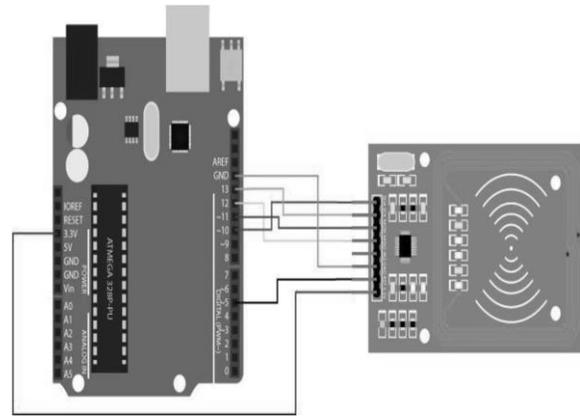
HASIL DAN PEMBAHASAN



Gambar 1. Blok Diagram Sistem

Blok diagram di atas mewakili sistem yang akan dirancang pada kesempatan ini, dan Arduino sebagai komponen mikrokontroler diberikan tegangan dan arus dari sebuah sumber catu daya, baik dengan menggunakan Adaptor maupun dengan menggunakan port USB komputer atau laptop. Catu daya ini akan memberikan sumber tenaga pula untuk modul RFID

Implementasi dari rangkaian di atas telah dirangkai sedemikian rupa dengan memberikan tegangan berasal dari port USB pada terminal komputer laptop serta melakukan koneksi I/O port pada Arduino dengan modul RFID-RC522, seperti pada tabel di bawah ini.



Gambar 2. Rangkaian Implementasi Sistem

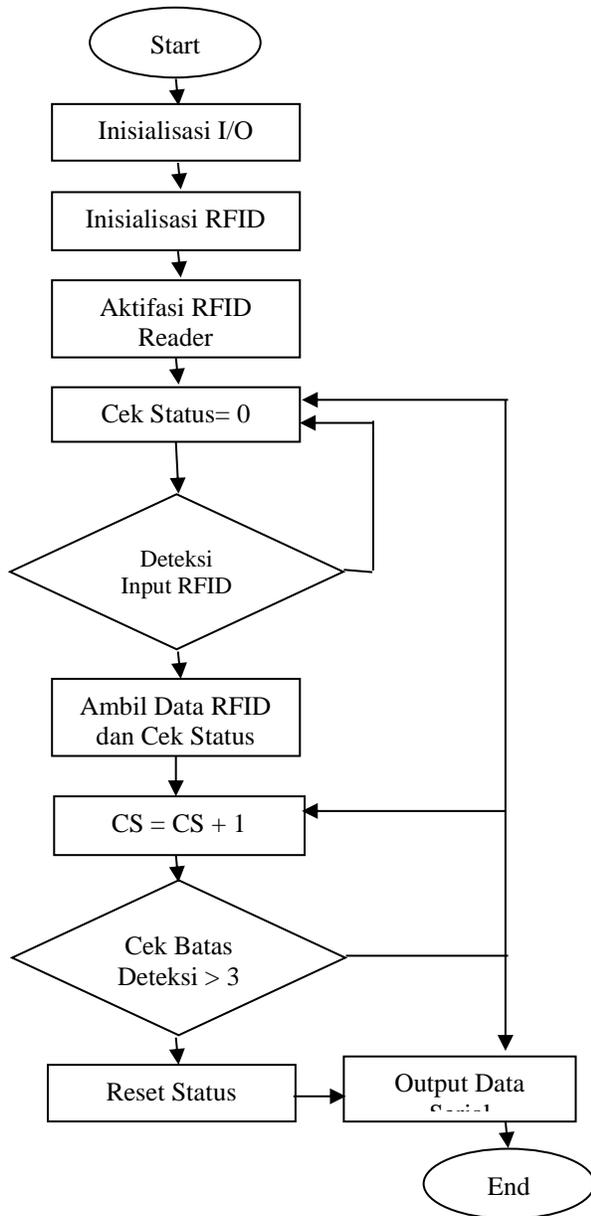
Tabel 1. Koneksi port RFID dan terminal Arduino

RFID-RC522 Module	Arduino Uno
1 - SDA	Digital 10
2 - SCK	Digital 13
3 - MOSI	Digital 11
4 - MISO	Digital 12
5 - IRQ	--unconnected--
6 - GND	Gnd
7 - RST	Digital 5
8 - 3.3V	3.3v

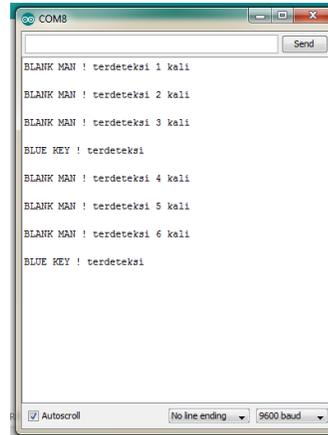
Koneksi dilakukan dengan memberikan *jumper* berupa kabel sesuai dengan daftar tabel koneksi port RFID dan terminal Arduino

Program untuk sistem pencacahan ini dilakukan sebagai upaya untuk mengetahui konsep pencacahan dengan menggunakan port I/O sebagai masukan. Penelitian sebelumnya sudah pernah dilakukan oleh tim untuk mengetahui *interfacing system* dengan menggunakan modul RFID, tanpa pencacahan perangkat RFID.

Koneksi Arduino dan modul RFID-RC522 ini meminta sebuah *library* yaitu *AddicoreRFID.h* dan *SPI.h*. *AddicoreRFID* akan digunakan untuk melakukan pendeteksian terhadap tipe dan identitas dari perangkat RFID yang akan diamati yaitu kartu kosong (BLANK MAN) dan gantungan kunci biru (BLUE KEY). Adapun hasil dalam pengujian sementara dengan dan tanpa *counter system* ini dapat dilihat berikut ini.



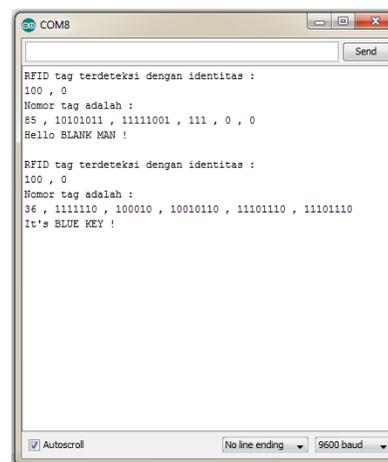
Gambar 3. Algoritma Program



Gambar 4. Tampilan Layar Monitoring dengan Sistem Counter

Kode-kode di atas diberikan sebagai hasil uji untuk pendeteksian salah satu perangkat RFID, yang pada kesempatan penelitian ini menggunakan dua model yaitu gantungan kunci biru (diindikasikan dengan “BLUE KEY”) dan kartu kosong (diindikasikan dengan “BLANK MAN”).

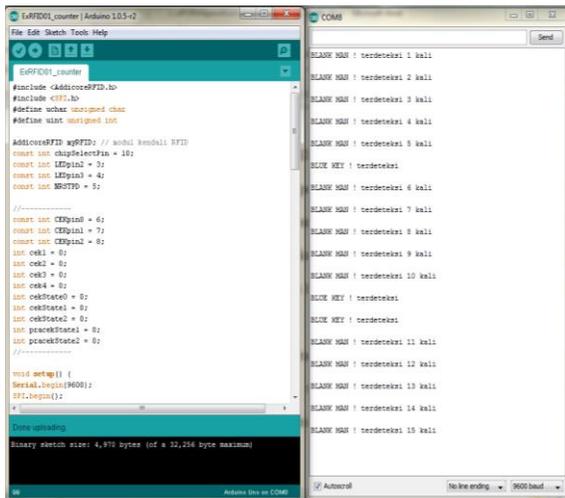
Pengenalan terhadap perangkat RFID (BLANK MAN dan BLUE KEY) ini dilakukan dengan menempatkan *library* AddicoreRFID.h pada awal program. File ini akan mengambil segenap identitas yang diperlukan dari perangkat RFID, yang kemudian dilanjutkan dengan penentuan algoritma dan variabel bantu terkait dengan identitas tersebut.



Gambar 5. Identitas BLANK MAN dan BLUE KEY

Instruksi sistem pencacahan pada pengamatan penelitian ini difokuskan pada upaya untuk mengetahui metode pencacahan yang tepat untuk beberapa masukan pada port I/O Arduino. Masukan port Arduino diupayakan sedemikian rupa dengan menggunakan modul RFID-RC522, yang telah siap pakai dan hanya melakukan koneksi dengan menggunakan kabel *jumper*. Perangkat lunak yang diimplementasikan pada sistem pencacah ini menggunakan satu masukan dan keluaran ada pada tampilan monitor IDE Arduino, dan difungsikan untuk mencacah perangkat RFID dengan kode yang sama.

```
#include <AddicoreRFID.h>
#include <SPI.h>
#define uchar unsigned char
#define uint unsigned int
```



Gambar 6. Hasil Monitoring Program

Librari utama pada program pencacah ini adalah dengan melibatkan AddicoreRFID.h yang menjadi pengendali utama modul RFID-RC522, dan menggunakan variabel myRFID. Modul RFID dikondisikan sedemikian rupa mengikuti standar minimal pada librari AddicoreRFID yaitu penentuan tata letak tiap pin pada modul RFID-RC522 yang terangkai seperti pada Tabel 1 di atas. Kemudian deklarasi terhadap tiap-tiap pin dan variabel yang diperlukan dalam program. Deklarasi juga diperlukan untuk menentukan tipe, status dan konstanta yang sesuai dengan variabel yang dibutuhkan pada pemrograman sistem pencacah berbasis arduino ini.

AddicoreRFID myRFID; // modul kendali RFID

```
const int chipSelectPin = 10;
const int LEDpin2 = 3;
const int LEDpin3 = 4;
const int NRSTPD = 5;
const int CEKpin1 = 7;
const int CEKpin2 = 8;
int cek1 = 0;
int cek2 = 0;
int cek3 = 0;
int cek4 = 0;
int cekState1 = 0;
int cekState2 = 0;
int pracekState1 = 0;
```

Void setup() biasanya digunakan untuk menandai atau mengenalkan atau menginisiasi terminal I/O yang dipakai untuk idnikator seperti LED, sensor, motor dan mungkin penggunaan terminal serial (serial port). Instruksi ini sekaligus memberitahukan kepada Arduino bahwa terminal-terminal tersebut akan digunakan selama program aplikasi dijalankan atau dengan kata lain bahwa terminal tersebut akan disiapkan oleh Arduino dalam menjalankan program. Inisialisasi dilakukan untuk menentukan mode, kondisi dan sifat dari tiap-tiap pin yang digunakan. Dua status yang diperlukan yaitu INPUT, yang digunakan untuk memberikan masukan data ke Arduino, dan OUTPUT, yang digunakan untuk membuat kondisi keluaran dari Arduino, misal dengan memberikan visualisasi LED.

```
void setup() {
  Serial.begin(9600);
  SPI.begin();
```

```
// Set digital pin 10 sebagai OUTPUT dan
terhubung ke RFID/ENABLE pin
pinMode(chipSelectPin,OUTPUT);
pinMode(LEDpin2,OUTPUT);
pinMode(LEDpin3,OUTPUT);
pinMode(CEKpin1,INPUT);
pinMode(CEKpin2,INPUT);
```

```
// Aktifasi RFID reader
digitalWrite(chipSelectPin, LOW);
digitalWrite(LEDpin2, LOW);
digitalWrite(LEDpin3, LOW);
```

```
// Set digital pin 10 , kondisi not Reset dan
Power-down
pinMode(NRSTPD,OUTPUT);
digitalWrite(NRSTPD, HIGH);
myRFID.AddicoreRFID_Init();
}
```

Sementara itu instruksi `void loop()` merupakan kode instruksi yang akan mengendalikan terminal I/O berkaitan dengan instruksi apa yang akan dilakukan selanjutnya, berikutnya dan seterusnya. Segegap kode program dalam

```
void loop()
{
  uchar status;
  uchar str[MAX_LEN];
  //-----
  cekState0 = digitalRead(CEKpin0);
  cekState1 = digitalRead(CEKpin1);
  cekState2 = digitalRead(CEKpin2);
  if (cekState2 != pracekState2) {
    if (cekState2 == HIGH) {
      cek2 = cek2 + 1;
    }
  }
  pracekState2 = cekState2;
```

Fungsi looping `if ...` digunakan untuk melakukan pengecekan status terhadap pin masukan dari RFID, status awal pin adalah LOW dan akan berganti menjadi HIGH saat modul RFID menerima kode RFID dari perangkat, yang didekatkan pada modul RFID.

Jika kondisi fungsi `if ...` ini dalam kondisi `true`, maka program akan melakukan eksekusi pencacahan yang diambil dan diperbandingkan dengan kondisi sebelumnya. Seandainya terjadi perubahan kondisi masukan maka sistem akan menambahkan nilai 1 dari nilai sebelumnya.

```
//Pencarian tag dan tipe berdasarkan
librari
status =
myRFID.AddicoreRFID_Request(PICC_R
EQIDL, str);

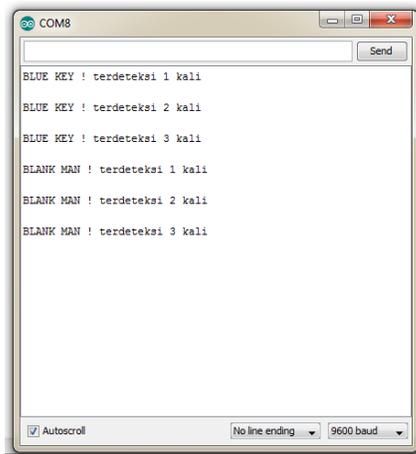
//Metode Anti-collision, nomor seri tag
dalam 4 bytes
status =
myRFID.AddicoreRFID_Anticoll(str);
```

```
if (status == MI_OK)
{
  if(str[0] == 85){
    .....
  }
  Serial.println();
  delay(500);
}
myRFID.AddicoreRFID_Halt();
}
```

Komparasi terhadap data pada perangkat RFID yang akan dideteksi, menggunakan mode string, dimana untuk perangkat kartu kosong dengan identitas BLANK MAN memiliki string 85, sedangkan perangkat gantungan kunci dengan identitas BLUE KEY memiliki nilai string 36. Uji coba pertama ini berhasil melakukan pencacahan terhadap kartu kosong saja sedangkan untuk gantungan kunci hanya dilakukan deteksi saja.

Pengujian yang kedua adalah melakukan pencacahan untuk kedua perangkat RFID yaitu kartu kosong dan gantungan kunci. Modifikasi dilakukan pada bagian `void setup()` dengan tambahan kode berikut ini.

```
if (cekState2 != pracekState2) { // cek
status pin I/O digital BLANK MAN
if (cekState2 == HIGH) {
  cek2 = cek2 + 1;
}
}
pracekState2 = cekState2;
if (cekState1 != pracekState1) { cek status
pin I/O digital BLUE KEY
if (cekState1 == HIGH) {
  cek1 = cek1 + 1;
}
}
pracekState1 = cekState1;
```



Gambar 7. Tampilan Pencacahan untuk BLANK MAN dan BLUE KEY

Adapun model kode untuk pencacahan kedua kartu memerlukan penambahan dan modifikasi pada algoritma perhitungan sebelumnya. Selengkapnya ada di bawah ini.

```

if(str[0] == 85){
  Serial.print("BLANK MAN !");
  Serial.print(" terdeteksi ");
  cek4 = cek2 + 1; // variabel
  pencacah deteksi BLANK MAN
  Serial.print(cek4);
  Serial.println(" kali");
}
else if(str[0] == 36){
  Serial.print("BLUE KEY !");
  Serial.print(" terdeteksi ");
  cek3 = cek1 + 1; // variabel
  pencacah deteksi BLUE KEY
  Serial.print(cek3);
  Serial.println(" kali");
}

```

Prinsip yang sama dilakukan pada BLANK MAN terhadap BLUE KEY, dimana sebelumnya ditentukan identitas dsri kode string 0, yang masing-masing 85 dan 36

Fungsi Pembatasan Jumlah Cacahan

Prosedur untuk dapat membatasi jumlah cacahan dilakukan dengan menambahkan fungsi berikut ini.

```

if(cek4 > 3){ // Pembatasan jumlah deteksi
  untuk BLANK MAN

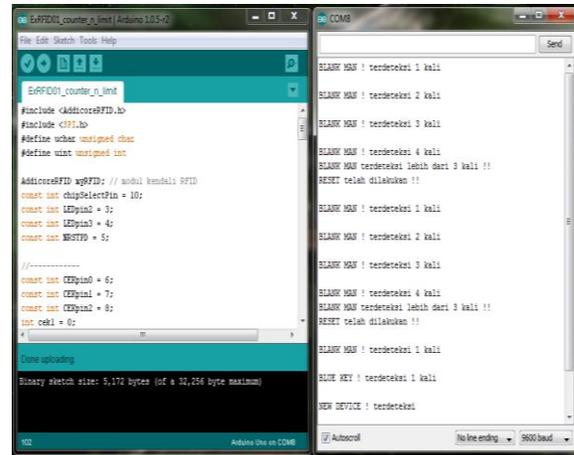
```

```

Serial.println("BLANK MAN terdeteksi
lebih dari 3 kali !!");
cek2 = -1;
Serial.println("RESET telah dilakukan !!")
}

```

Variabel cek4 digunakan untuk mengambil nilai cacahan sebelumnya dengan memberikan batasan nilai 3, dimana jika terdeteksi sebanyak 4 kali, maka sistem akan menginformasikannya melalui layar serial monitor, dan kemudian akan melakukan reset, sehingga deteksi berikutnya akan diberikan nilai awal lagi dari 1. Variabel cek2 diberikan nilai -1 dikarenakan variabel tersebut pada awalnya (saat nilai sama dengan 0) akan memiliki nilai 1, dimana pernyataan fungsi `cek2 = cek2 + 1;`, sehingga untuk memperoleh nilai 0 seperti yang diharapkan, maka diperlukan pengurang -1. Adapun tampilan hasilnya ada di bawah ini.



Gambar 8. Tampilan Pembatasan Pencacahan untuk BLANK MAN

KESIMPULAN

Adapun beberapa kesimpulan yang berhasil diperoleh dalam penelitian ini diantaranya adalah sebagai berikut :

- a. Konsep pencacahan data dapat dilakukan dengan fungsi `if ...`, dengan membandingkan antara kondisi sekarang dengan kondisi sebelumnya, pada sebuah variabel. Jika seandainya kondisi mengalami perubahan status yang semula LOW menjadi HIGH maka prosedur selanjutnya akan memberikan penambahan nilai 1 pada variabel tersebut,

dan kecepatan deteksi bergantung pada nilai *delay()*.

- b. Konsep perancangan perangkat keras sistem pencacah ini hanya menggunakan modul RFID-RC522 yang memiliki keterbatasan jarak deteksi dengan radius kurang lebih hanya 2 cm, terhadap perangkat RFID yang digunakan, yaitu kartu kosong dan gantungan kunci.

DAFTAR PUSTAKA

- Dzjersk, T., 2004, *In Search of Future- Proof RFID*, <http://www.usingrfid.com/features/read.asp?id=5>.
- Eddy Nurraharjo, 2011, "Analisis Model Akuisisi Data Terhadap Piranti Analog to Digital (ADC)", Jurnal Dinamika Informatika Vol. 3 No. 2, Semarang <http://www.unisbank.ac.id/ojs/index.php/fti2/article/view/1312>.
- Eddy Nurraharjo, 2012, "Terminal Port Komputer sebagai Perantara Pemrograman Bahasa Tingkat Tinggi", Jurnal Dinamika Informatika Vol. 17 No. 2, Semarang <http://www.unisbank.ac.id/ojs/index.php/fti1/article/view/1657>.
- Kenzeller, K. F., 1999. *RFID Handbook*, John Wiley & Sons.
- Kinsella, B., 2004. *RFID – It's More than Tags and Standards*, <http://www.usingrfid.com/features/read.asp?id=7>.
- Rush, T., 2003. *RFID in a Nuthshell – a Primer on Tracking Technology*, <http://www.usingrfid.com/features/read.asp?id=2>.