

BAB 2 LANDASAN KEPUSTAKAAN

Pada Bab 2 Landasan Kepustakaan terdapat tinjauan pustaka dari penelitian sebelumnya dan landasan teori dari berbagai sumber pustaka yang terkait dengan teori dan metode yang digunakan dalam penelitian.

2.1 Tinjauan Pustaka

Pada Tabel 2.1 dibawah dijelaskan penelitian sebelumnya yang terkait dengan skripsi ini.

Tabel 2.1 Penelitian Terkait

No	Judul, Tahun	Penulis	Persamaan	Perbedaan	
				Penelitian Sebelumnya	Dalam Skripsi
1.	<i>A Distributed Multi-channel Reader Anti-collision Algorithm for RFID environment</i> , 2015	Haidar Safa, Wassim El-Hajj, Christine Meguerditchian	Penelitian ini dilakukan untuk mengatasi <i>reader-to-reader collision</i>	Menggunakan algoritma <i>multi-channel</i> , dan dilakukan dengan simulasi	Menggunakan algoritma <i>Distributed Color Selection</i> dan diterapkan pada Arduino Uno
2.	Implementasi Algoritma <i>Slotted ALOHA</i> Sebagai <i>Anti-collision</i> Dalam Mengidentifikasi <i>Multiple RFID Tag</i> Pada <i>RFID Reader</i> , 2016	Vynska Amalia Permadi	Penelitian ini diterapkan untuk mengatasi <i>collision</i> , dan diterapkan pada Arduino	Mengatasi <i>tag-to-tag collision</i> menggunakan RFID 125KHz	Mengatasi <i>reader-to-reader collision</i> menggunakan RFID 13.56 MHz
3.	Simulasi Dan Analisis Sistem RFID Menggunakan <i>Muti Channel Anti Collision</i> Protokol, 2011	M. Rizky Rustam	Penelitian ini dilakukan untuk mengatasi <i>reader-to-reader collision</i> , dan pengujian berdasarkan <i>throughput</i> dan efisiensi	Menggunakan algoritma <i>multi-channel</i> dan dilakukan dengan simulasi MATLAB	Menggunakan algoritma <i>Distributed Color Selection</i> dan diterapkan pada Arduino Uno

2.2 Dasar Teori

Pada dasar teori ini akan dijelaskan mengenai teori-teori tentang komponen dan teknologi dari penelitian ini, yaitu RFID, Mikrokontroler Arduino Uno, Algoritma *Anti-Collision*, dan Algoritma *Distributed Color Selection*.

2.2.1 Radio Frequency Identification (RFID)

RFID merupakan salah satu teknologi *Automatic Identification System (AIS)*. Prinsip kerjanya memiliki beberapa kesamaan dengan teknologi AIS lainnya yaitu *smart card*. Sama seperti pada *smart card*, data identifikasi disimpan pada suatu perangkat elektronik yang disebut *transponder*. Perbedaan dari kedua sistem ini terletak pada catu daya dan media pertukaran data. Pada *smart card* pertukaran data dilakukan dengan menggunakan medan magnet atau gelombang elektromagnetik. Sedangkan RFID menggunakan gelombang radio dan teknik radar yang memiliki kemampuan untuk mendeteksi sumber energi dari pemancar sekitarnya (Finkenzeller, 2003). Pada Tabel 2.2 menjelaskan perbandingan antara RFID dengan AIS lainnya.

Tabel 2.2 Perbedaan beberapa perangkat AIS

System Parameters	Barcode	OCR	Voice recog.	Biometry	Smart Card	RFID system
Typical data quantity (bytes)	1-100	1-100	-	-	16-64k	16-64k
Data density	Low	Low	High	High	Very high	Very high
Machine readability	Good	Good	Expensive	Expensive	Good	Good
Readability by people	Limited	Simple	Simple	Difficult	Impossible	Impossible
Influence of dirt/damp	Very high	Very high	-	-	Possible (contacts)	No influence
Influence of (opt.) covering	Total failure	Total failure	-	Possible	-	No influence
Influence of direction and position	Low	Low	-	-	Undirectional	No influence
Degradation/wear	Limited	Limited	-	-	Contacts	No influence
Purchase cost/reading electronics	Very low	Medium	Very high	Very high	Low	Medium
Operating costs (e.g. printer)	Low	Low	None	None	Medium (contacts)	None

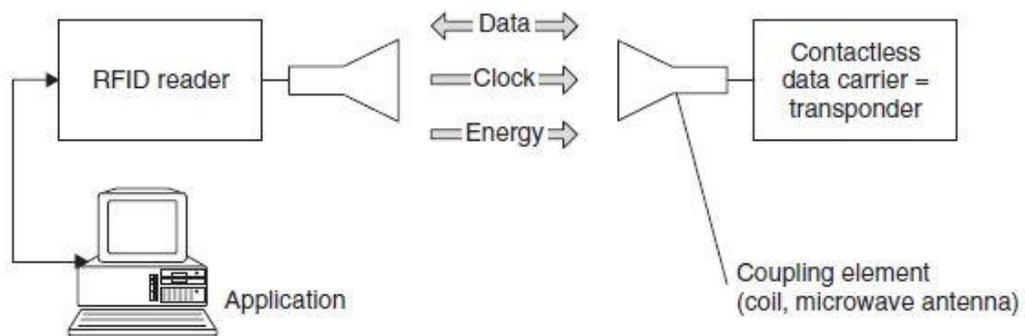
Unauthorised copying/modification	Slight	Slight	Possible* (audio tape)	Impossible	Impossible	Impossible
Reading speed (including handling data of carrier)	Low ~4s	Low ~3s	Very low >5s	Very low >5-10s	Low ~4s	Very fast ~0.5s
Maximum distance between data carrier and reader	0-50cm	<1cm Scanner	0-50cm	Direct contact**	Direct contact	0-5m, microwave

Sumber: (Finkenzeller, 2003)

Sistem RFID terdiri atas 2 komponen yaitu:

1. *Transponder/Tag*. Merupakan Perangkat yang diletakkan atau ditanamkan pada objek yang akan diidentifikasi.
2. *Interrogator/Reader*. Merupakan perangkat yang berfungsi menangkap data identifikasi suatu objek.

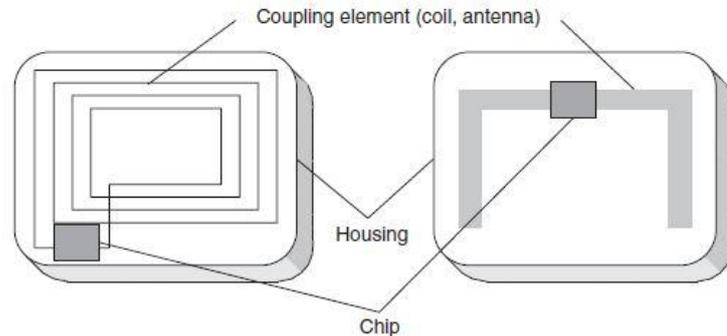
Pada Gambar 2.1 menjelaskan ilustrasi komunikasi dasar antara RFID *reader* dan *tag*. RFID *reader* merupakan sebuah modul *radio frequency* (RF) yang terdiri dari *transmitter* dan *receiver*, *control unit*, dan *coupling element*. Beberapa RFID *reader* juga memiliki interface tambahan, seperti: RS 232, RS 485, dan lain-lain. *Interface* tersebut digunakan untuk meneruskan data yang diterima ke sistem lainnya, seperti: PC, *robot control system*, dan lain-lain.



Gambar 2.1 Komponen dan Ilustrasi Komunikasi Dasar pada Sistem RFID

Sumber: (Finkenzeller, 2003)

Dan pada Gambar 2.2 dijelaskan lebih rinci tentang *tag* RFID. *Tag* merupakan perangkat yang memiliki data identifikasi pada sistem RFID. *Tag* terdiri dari *coupling element* dan *microchip* elektronik.



Gambar 2.2 Layout Transponder

Sumber: (Finkenzeller, 2003)

RFID dapat diproduksi dalam berbagai format agar sesuai dengan lingkungan aplikasi yaitu: RFID Aktif dan Pasif. Namun pada dasarnya terdiri dari komponen yang sama. Dalam Tabel 2.3 dijelaskan klasifikasi RFID Aktif dan Pasif.

Tabel 2.3 Klasifikasi RFID

	RFID Aktif	RFID Pasif
Sumber Daya Tag	Ada di dalam <i>tag</i>	Daya dikirimkan dari <i>reader</i> melalui frekuensi radio
Baterai Tag	Ya	Tidak
Ketersediaan Daya Tag	Continuous	Hanya ketika berada dalam jangkauan <i>reader</i>
Kekuatan sinyal yang dibutuhkan dari <i>reader</i> ke <i>tag</i>	Low	High (Harus menyediakan daya untuk <i>tag</i>)
Ketersediaan kekuatan sinyal dari <i>tag</i> ke <i>reader</i>	High	Low
Jarak Komunikasi	Long Range (100m atau lebih)	Short atau very short range (3m atau kurang)
Kumpulan Multi-<i>tag</i>	<ul style="list-style-type: none"> - 1000 static <i>tag</i> dari satu <i>reader</i> - 20 mobile <i>tag</i> dengan kecepatan 100mph 	<ul style="list-style-type: none"> - Ratusan <i>tag</i> dengan maksimal jarak 3 meter dari satu <i>reader</i> - 20 mobile <i>tag</i> dengan kecepatan 3

		mph atau lebih lambat
Kemampuan Sensor	Kemampuan untuk terus menerus memantau dan merekam input sensor	Kemampuan untuk membaca dan transfer nilai sensor hanya ketika <i>tag</i> mendapat daya dari <i>reader</i>
Penyimpanan Data	Besar, read/write penyimpanan data (128KB) dengan kemampuan pencarian data dan akses yang canggih	Kecil, read/write penyimpanan data (e.g. 128 bytes)

Sumber: (AUTO-ID, 2002)

RFID terpusat pada komunikasi nirkabel dan memanfaatkan gelombang radio yang merupakan bagian dari spektrum elektromagnetik. Pada umumnya RFID beroperasi pada empat pita frekuensi secara global. Tabel 2.4 merangkum karakteristik masing-masing pita frekuensi dengan aplikasinya.

Tabel 2.4 Karakteristik RFID Berdasarkan Pita Frekuensi

Band	LF (Low Frequency)	HF (High Frequency)	UHF (Ultra High Frequency)	Microwave
Frekuensi	30 – 300KHz	3 – 30 MHz	300 MHz – 3 GHz	2 – 30 GHz
Frekuensi yang digunakan RFID	125 – 134 KHz	13.56 MHz	433MHz, 860 – 960 MHz	2.45 GHz~
Jarak Komunikasi	< 0.5m	≤ 1.5m	433MHz: ≤ 100m 860 – 960MHz: 0.5m~5m	≤ 10m
Data Transfer Rate	< 1kbit/s	± 25kbit/s	± 30kbit/s	≤ 100kbit/s
Karakteristik	- Jangkauan pendek - Data	- jangkauan menengah - Data rate	- Jangkauan luas - Data rate	- Jangkauan luas - Data rate

	rate rendah - Menembus air, tapi tidak logam	menengah - Menembus air, tapi tidak logam	tinggi - Tidak bisa menembus air dan logam	tinggi - Tidak bisa menembus air dan logam
Aplikasi	Animal ID, Access Control, Car Immobiliser	Smart Labels, Contactless Travel Cards, Access & Security, Apparel	Logistics, Baggage Handling, Electronic Toll Collection	Electronic Toll Collection

Sumber: (iEMobile, 2012)

2.2.2 Arduino Uno

Arduino Uno adalah *board mikrokontroler* berbasis *ATmega328*. Uno memiliki 14 *pin digital input / output* (dimana 6 dapat digunakan sebagai *output PWM*), 6 *input analog*, *resonator keramik 16 MHz*, koneksi USB, *jack listrik*, *header ICSP*, dan tombol *reset*. Arduino Uno berisi semua yang diperlukan untuk mendukung *mikrokontroler*, sumber daya cukup dihubungkan menggunakan *power USB* (jika terhubung ke komputer dengan kabel USB) dan juga dengan adaptor AC/DC atau baterai.

Arduino Uno berbeda dari semua papan sebelumnya dalam hal tidak menggunakan *FTDI chip driver USB-to-serial*. Sebaliknya, fitur *Atmega16U2* (*Atmega8U2* sampai versi R2) diprogram sebagai konverter *USB-to-serial*. Revisi 2 dari Uno memiliki *resistor pulling 8U2 HWB* yang terhubung ke tanah, sehingga lebih mudah untuk menggunakan mode DFU (www.arduino.cc, 2017).



Gambar 2.3 Arduino Uno

Sumber: (www.arduino.cc, 2017)

Gambar 2.3 adalah bentuk dari *board* Arduino Uno. Arduino Uno dapat disuplai melalui koneksi USB atau dengan sebuah *power supply* eksternal. Sumber daya dipilih secara otomatis. *Supply* eksternal (non-USB) dapat diperoleh

dari sebuah adaptor AC ke DC atau baterai. Adaptor dapat dihubungkan dengan mencolokkan sebuah *center-positive plug* yang panjangnya 2,1 mm ke *power jack* dari *board*. Kabel *lead* dari sebuah baterai dapat dimasukkan dalam *header pin Ground (Gnd)* dan *pin Vin* dari konektor *POWER*.

Board Arduino UNO dapat beroperasi pada sebuah suplai eksternal 6 sampai 20 Volt. Jika disuplai dengan yang lebih kecil dari 7 V, kiranya *pin 5 Volt* mungkin mensuplai kecil dari 5 Volt dan *board* Arduino UNO bisa menjadi tidak stabil. Jika menggunakan suplai yang lebih dari besar 12 Volt, *voltage regulator* bisa kelebihan panas dan membahayakan *board* Arduino UNO. *Range* yang direkomendasikan adalah 7 sampai 12 Volt. Pin-pin dayanya adalah sebagai berikut:

1. VIN. Tegangan *input* ke Arduino *board* ketika *board* sedang menggunakan sumber suplai eksternal (seperti 5 Volt dari koneksi USB atau sumber tenaga lainnya yang diatur). Kita dapat menyuplai tegangan melalui pin ini, atau jika penyuplaian tegangan melalui *power jack*, aksesnya melalui pin ini.
2. 5V. Pin output ini merupakan tegangan 5 Volt yang diatur dari *regulator* pada *board*. *Board* dapat disuplai dengan salah satu suplai dari DC *power jack* (7-12V), USB *connector* (5V), atau pin VIN dari *board* (7-12). Penyuplaian tegangan melalui pin 5V atau 3,3V *membypass regulator*, dan dapat membahayakan *board*. Hal itu tidak dianjurkan.
3. 3V3. Sebuah suplai 3,3 Volt dihasilkan oleh *regulator* pada *board*. Arus maksimum yang dapat dilalui adalah 50 mA.

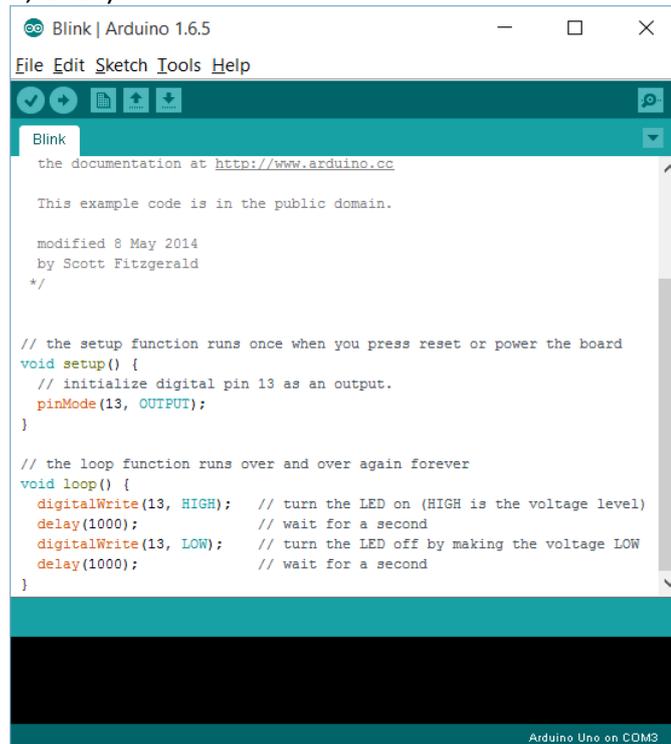
2.2.3 Arduino *Integrated Development Environment (IDE)*

Arduino *Integrated Development Environment (IDE)* adalah *software* yang dibuat dengan menggunakan Bahasa pemrograman *java* berdasarkan pengolahan *avr-gcc* dan perangkat lunak *open source* lainnya. Arduino IDE terdiri dari:

1. Editor program. Merupakan sebuah window yang memungkinkan pengguna menulis dan mengubah program dalam Bahasa pemrograman.
2. *Verify/Compiler*. Merupakan sebuah modul yang mengubah kode program menjadi kode biner. Sebuah *mikrokontroler* tidak akan bias memahami Bahasa pemrograman, karena yang dipahami oleh mikrokontroler adalah kode biner.
3. *Uploader*. Merupakan sebuah modul yang digunakan untuk membuat kode biner dari computer ke dalam memori mikrokontroler didalam *board* Arduino.

Pada Gambar 2.4 merupakan tampilan dari Arduino IDE. Arduino IDE bisa dijalankan di computer dengan berbagai macam *platform* karena berbasis *java*. *Source* program yang dibuat untuk aplikasi mikrokontroler menggunakan

Bahasa pemrograman C/C++ dan dapat digabungkan dengan *assembly* (www.arduino.cc, 2017).

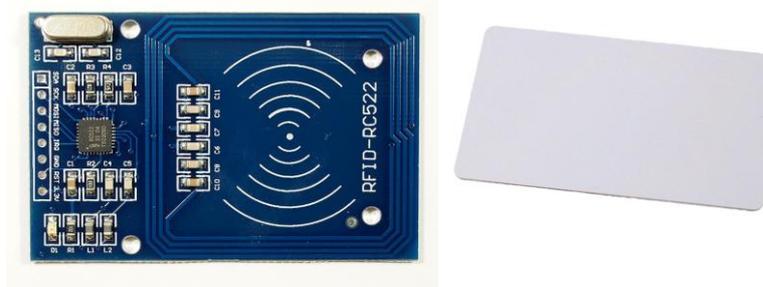


```
Blink | Arduino 1.6.5
File Edit Sketch Tools Help
Blink
the documentation at http://www.arduino.cc
This example code is in the public domain.
modified 8 May 2014
by Scott Fitzgerald
*/
// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin 13 as an output.
  pinMode(13, OUTPUT);
}
// the loop function runs over and over again forever
void loop() {
  digitalWrite(13, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000);           // wait for a second
  digitalWrite(13, LOW); // turn the LED off by making the voltage LOW
  delay(1000);           // wait for a second
}
Arduino Uno on COM3
```

Gambar 2.4 Arduino IDE
Sumber: (www.arduino.cc, 2017)

2.2.4 RFID MFRC522

MFRC522 adalah IC *reader/writer* yang terintegrasi tinggi untuk komunikasi *contactless* pada frekuensi 13.56 MHz. *Reader* MFRC522 mendukung ISO/IEC 14443 A / MIFARE dan NTAG. Pemancar internal MFRC522 mampu mengelola antena *reader/writer* yang dirancang untuk berkomunikasi dengan *transponder* atau *tag* ISO/IEC 14443 A/MIFARE tanpa sirkuit aktif tambahan (Semiconductors, 2016). Pada Gambar 2.5 merupakan bentuk dari modul RFID *reader* MFRC522 beserta *tag* nya (www.hotmcu.com, 2017)



Gambar 2.5 RFID MFRC522 *reader* dan *tag* nya
Sumber: (www.hotmcu.com, 2017)

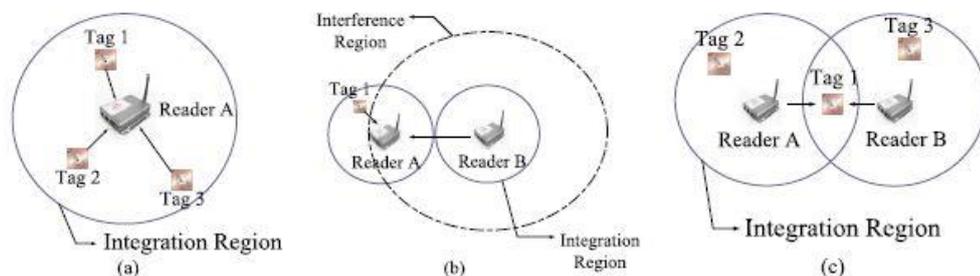
Berikut merupakan spesifikasi dari RFID *reader* MFRC522:

1. Nama modul : MFRC522

2. *Working current* : 13 – 26mA / DC 3.3V
3. *Standby current* : 10 – 13mA / DC 3.3V
4. *Sleep current* : < 80uA
5. *Peak current* : < 20mA
6. Pita Frekuensi : 13.56MHz
7. Jarak Baca : 0 ~ 60mm
8. Protokol : SPI
9. Kecepatan komunikasi data : 10Mbit/s Max
10. Tipe *tag* yang mendukung : Mifare1 S50, Mifare1 S70, Mifare UltraLight, Mifare Pro, Mifare Desfire
11. Ukuran : 40mm x 60mm
12. Temperatur kerja : -20 – 80 derajat
13. Temperatur penyimpanan : -40 – 85 derajat
14. Kelembapan : kelembapan relevan 5% - 95%

2.2.5 Collision & Anti-Collision

Dalam sistem RFID transmisi yang terjadi secara simultan dapat menyebabkan tabrakan data/*collision*, hal ini disebabkan karena *reader* dan *tag* beroperasi pada saluran yang sama. Pada umumnya ada dua jenis *collision* yang memungkinkan terjadi pada sistem RFID, yaitu: *tag collision (tag-to-tag collision)* dan *reader collision*. Dalam *reader collision* dibagi menjadi dua yaitu *reader-to-tag collision*, dan *reader-to-reader collision*.



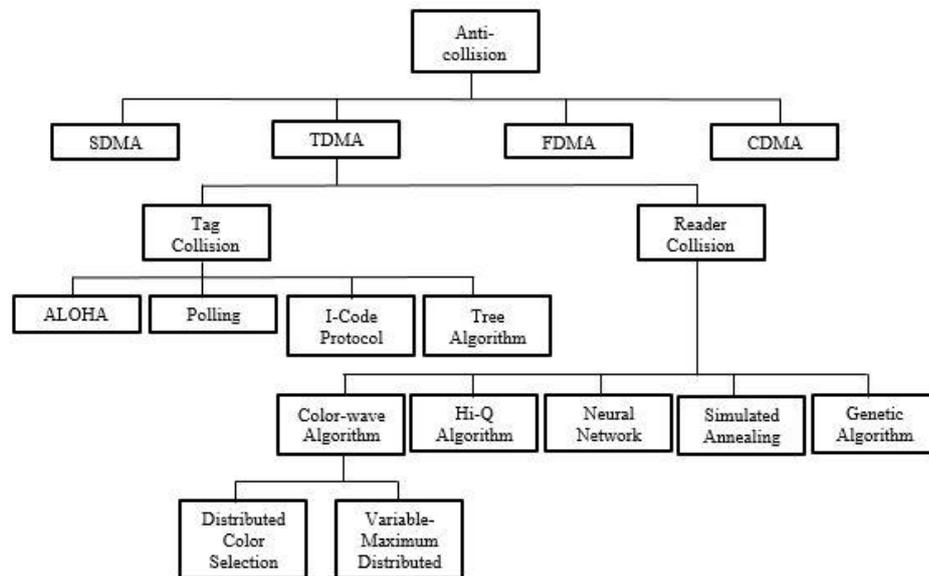
Gambar 2.6 Jenis Collision pada sistem RFID

Sumber: (Tang, et al., 2011)

Pada Gambar 2.6 (a) menjelaskan bagaimana terjadinya *tag-to-tag collision*. *Collision* terjadi ketika ada beberapa *tag* yang berada pada *interrogation zone* dari satu *reader* kemudian mengirimkan data secara simultan/bersamaan. Sehingga *reader* tidak dapat membaca *tag* manapun. Adapun solusi yang ditawarkan untuk mengatasi *tag-to-tag collision*, salah satunya menggunakan algoritma ALOHA. Pada Gambar 2.6 (b) menjelaskan bagaimana terjadinya

reader-to-reader collision. Ketika ada dua *reader* saling berdekatan maka *reader* akan saling terganggu oleh sinyal *reader* yang lain. Transmisi *reader B* dapat mempengaruhi respon dari *Tag1* terhadap *reader A*. dalm kasus ini *reader A* tidak dapat membaca *tag* manapun. Sedangkan pada Gambar 2.6 (c) menjelaskan terjadinya *reader-to-tag collision*. Hal ini akan terjadi bila *tag (Tag1)* diposisikan dalam *interrogation zone* yang saling tumpang tindih (*overlapping*) dari dua *reader*, *tag* tersebut tidak dapat membedakan antara sinyal *reader A* maupun *reader B*. Berbeda dengan *reader-to-tag collision*, disini *reader* hanya tidak bisa membaca *tag* yang berada pada area yang terjadi *overlapping*. Namun *reader* masih dapat membaca *tag* lain yang berada pada *interrogation zone* nya masing-masing, *Tag2* dan *Tag3* berhasil dibaca (Tang, et al., 2011).

Dengan masalah yang terjadi pada sistem RFID tersebut maka diperlukan suatu algoritma yang mampu mengatasi *collision*, yang biasa disebut algoritma *anti-collision*. Pada algoritma *anti-collision* didasari oleh beberapa protokol *multiple acces* diantaranya *Space Division Multiple Access (SDMA)*, *Time Division Multiple Access (TDMA)*, *Code Division Multiple Acces (CDMA)*, *Frequency Division Multiple Acces (FDMA)*. Pada Gambar 2.7 dijelaskan prosedur dari keempat dasar protoKol *multiple access* beserta dengan pengembangannya.



Gambar 2.7 Prosedur Algoritma *Anti-collision*

Sumber: (Shih, et al., 2006)

Pada TDMA ini berkaitan dengan teknik dimana secara kronologis kapasitas *channel* dibagi-bagi. Prosedur TDMA sangat luas dibidang sistem radio, dan menjadi dasar sebagian besar kelompok algoritma *anti-collision*. Yang menjadi penyebab terjadinya *tag collision* adalah ketika beberapa *tag* yang secara bersamaan berada pada *interrogation zone* suatu *reader* dan membuat *reader* bingung dalam mengidentifikasinya. Ada beberapa algoritma yang telah ada dalam mengatasi *tag collision*, dan yang paling sering digunakan adalah algoritma ALOHA. Sedangkan pada *reader collision*, *reader* dapat mengganggu pengoperasian *reader* yang lain ketika berada pada jarak yang berdekatan. Untuk

masalah ini ada beberapa algoritma yang dapat digunakan, salah satunya adalah algoritma *Colorwave* (Shih, et al., 2006). Dalam algoritma *Colorwave* ada dua kategori yaitu *Distributed Color Selection* (DCS) dan *Variable-Maximum Distributed Color Selection* (VDCS), algoritma DCS akan digunakan dalam penelitian ini untuk mencari solusi yang optimal dalam mengatasi *reader collision*.

2.2.6 Algoritma *Distributed Color Selection*

Algoritma *Disributed Color Selection* merupakan salah satu bagian dari algoritma *Colorwave*, dimana algoritma *Colorwave* dibagi menjadi dua yaitu *Distributed Color Selection* (DCS) dan *Variable-Maximum Distributed Color Selection* (VDCS). DCS adalah algoritma *anti-collision* berbasis *Time Division Multiple Access* (TDMA) yang menggunakan interval waktu tertentu yang disebut *timeslot* yang diasumsikan sebagai *color*. Dalam penerapannya pada sistem RFID, tiap *reader* memilih *timeslot/color* secara acak dan menyimpannya. *Reader* akan mengantri untuk melakukan permintaan transmisi dan hanya akan melakukan transmisi ketika dalam periode *timeslot/color* yang telah dipilih. Jika terjadi tabrakan diantara satu *reader* dengan yang lain maka permintaan transmisi akan dibuang. Selanjutnya, *reader* akan memilih kembali *timeslot/color* baru secara acak. Hal ini membuat semua *reader* tetangganya juga memilih *timeslot/color* yang baru sampai tidak ada *reader* yang memiliki *timeslot/color* yang sama. Tindakan mengganti dan memilih kembali *timeslot/color* ini disebut dengan istilah "*kick*" (Waldrop, et al., 2003).

Dalam DCS terdapat variabel *timeslot/color* maksimum (*max_colors*) yang merupakan input ke algoritma DCS dan sudah ditetapkan jumlahnya. Setiap *reader* mencatat *timeslot/color* apa saja yang tersedia menjadi (*color_ID*) dan bebas untuk memilih. Sifat terdistribusi dari algoritma ini tidak memerlukan sinkronisasi antara *color_ID* pada *reader* yang berbeda. DCS diimplementasikan dengan tiga *subroutine* terpisah. *Subroutine* pertama mengelola transmisi. *Subroutine* kedua mengelola tabrakan dan pemilihan *timeslot/color* baru. *Subroutine* ketiga mengelola resolusi *kick*, atau komunikasi antar *reader*. Pada Gambar 2.8 dijelaskan secara rinci *subroutine* dari algoritma DCS.

```

DCS Subroutine 1 - Transmission:
• If transmission requested:
  - If (timeslot_ID % max_colors) == current_color
    * then transmit
    * else idle until (timeslot_ID % max_colors) == current_color

DCS Subroutine 2 - Collision:
• If attempted transmission but experienced collision:
  - current_color == random(max_colors)
  - broadcast kick stating new color

DCS Subroutine 3 - Kick resolution:
• If kick received stating current_color
  - randomly change to different color within max_colors

```

Gambar 2.8 Subroutine Algoritma *Distributed Color Selection*
 dSumber: (Waldrop, et al., 2003)

2.2.7 Efisiensi Sistem

Pada proses identifikasi pada sistem RFID *multi-reader* dapat dikatakan “Terdeteksi” jika data yang ada pada *tag* mampu dibaca oleh semua *reader* yang ada dan *reader* yang saling berdekatan tidak mengalami tabrakan antar satu sama lain. Dan proses identifikasi pada sistem RFID *multi-reader* dapat dikatakan “Tidak Terdeteksi” jika data yang ada pada *tag* tidak mampu untuk dibaca oleh semua *reader* yang ada. Jika terdapat suatu keadaan dimana *tag* berhasil dibaca oleh sebagian *reader* saja maka dapat dikategorikan “Tidak Terdeteksi”. Untuk persentase efisiensi merefleksikan kemampuan protokol untuk mendeteksi kemungkinan terjadinya *collision* dan oleh karenanya untuk mencegah transmisi yang tidak diinginkan. Efisiensi dinyatakan dalam persen. Berikut adalah persamaan untuk mengetahui efisiensi sistem (Birari, 2005).

$$Efisiensi (\%) = \frac{Total\ query\ yang\ sukses\ (pada\ semua\ reader)}{Total\ query\ (Sukses + Gagal)} \times 100 \quad (2.1)$$

Sumber: (Birari, 2005)