

## BAB 2

### STUDI AWAL

#### 2.1.Deskripsi Karya Sejenis

##### 2.1.1. Pemanfaatan Sistem RFID sebagai Pembatas Akses Ruangan

Pada penelitian ini, sistem RFID (*Radio Frequency Identification*) dimanfaatkan sebagai kartu identifikasi personal pada sistem akses ruangan. Keberadaan sistem ini ditujukan untuk menjaga keamanan dan privasi ruangan dari seseorang yang tidak memiliki otoritas untuk memasuki ruangan tersebut. Melalui perancangan dan implementasi sistem akses ruangan ini, dilakukan evaluasi sistem kerja kunci elektrik berbasis komponen *solenoid* serta jarak dan posisi optimal pembacaan RFID *tag* guna memberikan kenyamanan pada pengguna ketika mengakses ruangan. RFID *tag* yang dipergunakan dalam sistem ini berbentuk kartu tipe EM4001 dan menyimpan kode unik yang digunakan sebagai identifikasi personal. Kode ini dibaca oleh RFID *reader* tipe ID-12 dan divalidasi otoritasnya dengan mikrokontroler ATMega32 untuk mengatur sistem kerja kunci elektrik yang dirancang sendiri menggunakan *solenoid*. Sistem ini dilengkapi pula dengan sistem *database* untuk pencatatan pengguna yang mengakses ruangan.

Dari hasil pengujian, seluruh (100%) RFID *tag* dapat dikenali oleh RFID *reader* dalam tiga posisi berbeda dengan jarak optimal sejauh 5 cm (vertikal) dan 2 cm (horisontal). Keseluruhan (100%) aktifitas pengguna yang mengakses

ruangan dapat tercatat dalam sistem database. Hasil penelitian ini dapat disimpulkan bahwa teknologi RFID dapat digunakan secara nyaman dan aman sebagai alternatif sistem identifikasi personal untuk sistem akses ruangan. (Hendi Hadian Rachmat, Gilbert Allegro Hutabarat, Teknik Elektro ITENAS, 2014)

### **2.1.2. Kunci Pintu Otomatis Berbasis RFID Bagi Tuna Netra**

Ketertarikan peneliti pada teknologi RFID dan kesulitan para penderita tuna netra untuk mendeteksi lubang kunci pada pintu, membuat peneliti berfikir untuk mengaplikasikan teknologi RFID ini untuk kepentingan para tuna netra. RFID yang disebut *Reader* mempunyai fungsi mendeteksi objek yang akan masuk pada pintu, dan *Tag* RFID adalah sebagai objeknya. Saat objek ini terdeteksi, kunci pintu akan terbuka dan saat pintu ditutup kunci secara otomatis akan mengancing. Peneliti membuat kunci pintu otomatis yang berbasiskan RFID ini agar mempermudah para penderita tuna netra dalam menjaga privasi dan keamanannya. (Syamsul Akbar, Teknik Elektro UMY, 2010)

## **2.2. Dasar Teori**

### **2.2.1. NFC (Near Field Communication)**

NFC berbeda dari teknologi komunikasi nirkabel lainnya seperti kode QR. NFC (*Near Field Communication*) merupakan seperangkat teknologi komunikasi nirkabel yang dikembangkan dari teknologi RFID (*Radio Frequency Identification*).

Tabel 2.1 menunjukkan perbedaan diantara 2 jenis teknologi nirkabel, yaitu kode QR dan NFC. Kedua teknologi tersebut memiliki fungsi yang mirip dalam berbagi data dan informasi. Kode QR cukup mengambil gambar dari kode untuk

melakukan transfer data sedangkan NFC untuk keamanan dalam mengakses data, NFC lebih mudah dan cepat dengan cara menyentuh perangkat secara bersamaan. Oleh karena itu, kode QR akan menggunakan waktu sedikit lebih banyak dalam mengakses data dibandingkan NFC. Keuntungan lain yang membuat NFC sebagai teknologi yang diandalkan adalah NFC dapat melakukan komunikasi dua arah sedangkan kode QR hanya dapat melakukan komunikasi satu arah.

**Tabel 2.1 Perbedaan kode QR, dan NFC**

<b>Spesifikasi</b>	<b>Kode QR</b>	<b>NFC</b>
<b>Kecepatan transfer data</b>	-	424 Kbps
<b>Keamanan akses data</b>	Mengambil gambar, semua orang dapat mengakses data	Menyentuh perangkat
<b>Jenis Komunikasi</b>	Satu arah	Dua arah

NFC dibuat berdasarkan pada standar RFID untuk mengembangkan komunikasi radio yang dapat melakukan komunikasi dua arah diantara dua perangkat, dimana sistem koneksi sebelumnya menggunakan *smart card* dengan komunikasi hanya satu arah. Standar yang sudah digunakan tersebut adalah untuk menjamin bahwa semua bentuk teknologi NFC diatur untuk saling berinteraksi dengan perangkat lain dan agar dapat bekerja dengan perangkat terbaru nantinya. Terdapat 2 standar utama pada NFC, yaitu ISO/IEC14443 yang digunakan sebagai *tag* NFC untuk menyimpan informasi dan ISO/IEC 18000-3 yang digunakan untuk komunikasi RFID sebagai perangkat *reader* NFC.

ISO/IEC 18000-3 adalah standar internasional untuk semua perangkat NFC yang dioperasikan pada frekuensi 13,56 MHz untuk berkomunikasi dengan kartu tipe A dan tipe B. Jaraknya adalah kurang lebih 4 cm untuk dapat berkomunikasi dengan perangkat lain.

NFC melakukan pertukaran data dengan melibatkan sebuah perangkat aktif dan perangkat pasif. Perangkat aktif sebagai *reader/writer* memancarkan medan elektromagnetik yang dapat ditangkap oleh antena pada perangkat pasif. Kemudian, pada saat adanya medan elektromagnetik arus listrik di alirkan ke perangkat pasif sebagai catu daya mikroprosesor. Apabila perangkat pasif telah memperoleh catu daya yang cukup untuk bekerja, selanjutnya perangkat pasif akan merespon perintah yang diberikan oleh perangkat aktif. Pada tahap ini memungkinkan terjadinya pengiriman konten yang disimpan dalam memori perangkat pasif. Data yang disimpan pada perangkat pasif dapat berupa gambar, teks, kode atau nomor telepon. Kapasitas data yang disimpan pada perangkat pasif adalah sekitar 48 *bytes* sampai 9 *kilobytes*. Untuk kecepatan transfer data hingga 424 kilobit per detik. Sifat dari perangkat pasif atau *tags* adalah dapat ditulis kembali dan dibaca, juga memiliki fitur sebuah UID (*Unique Identifier*). (Mohamad Syafiq, Electrical Engineering Universiti Teknologi Malaysia, 2013)

#### 2.2.1.1. Antena NFC

Antena NFC bekerja pada frekuensi rendah yang mana panjang gelombangnya lebih panjang dan perangkat harus lebih besar. Walaupun begitu, antena NFC tetap dapat diatur agar cocok dengan sebuah perangkat kecil seperti telepon genggam yang ditunjukkan pada Gambar 2.1. Antena adalah sebuah

pasangan induktif yang memiliki induktans yang besar besar sekali sehingga dapat menghasilkan sebuah medan magnet yang kuat di sekitar perangkat. Antena bekerja pada frekuensi 13.56 MHz dan memiliki panjang gelombang 22 meter.



**Gambar 2.1 Antena NFC**

Sumber: Mohamad Syafiq, Electrical Engineering, Universiti Teknologi Malaysia, 2013

Penggunaan NFC dapat dikembangkan dengan memodifikasi struktur dari antena. Antena NFC dibuat dengan tembaga melingkar yang dapat menghasilkan medan magnet. Putaran tembaga tersebut memberikan sebuah medan magnet yang kuat. Putaran tembaga adalah sebuah induktor yang dapat menginduksikan arus ke *tag* NFC. Ini terjadi ketika perangkat pada jarak yang berdekatan dan kedua perangkat saling berkomunikasi satu sama lain. (Mohamad Syafiq, Electrical Engineering Universiti Teknologi Malaysia, 2013).

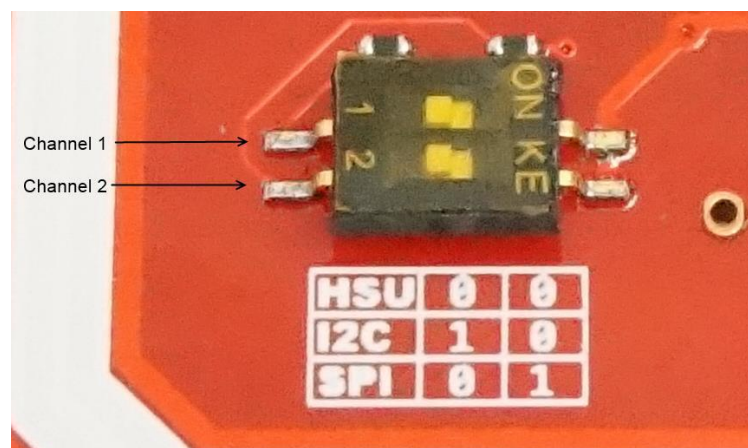
#### 2.2.1.2. NFC Reader

NFC *reader* digunakan untuk membaca isi informasi dari sebuah *tag* NFC. NFC *reader* yang akan digunakan adalah tipe NXP PN532 hasil produksi dari

Elechouse. NFC *reader* dengan tipe NXP PN532 memiliki fitur-fitur sebagai berikut :

- 1) Dapat menggunakan antarmuka I2C, SPI dan HSU (*High Speed UART*).
- 2) Terbuat dari PCB antena dengan jarak komunikasi 5 cm hingga 7 cm.
- 3) Bekerja sebagai RFID *reader* atau *writer*.
- 4) Bekerja sebagai kartu 14443-A atau sebuah kartu virtual.
- 5) Kompatibel dengan Arduino
- 6) Mendukung NFC dengan Android pada telepon genggam.
- 7) Berukuran kecil 42,7 mm x 40,4 mm x 4 mm.
- 8) Tegangan 5 volt untuk I2C dan UART, sedangkan 3,3 Volt untuk SPI.

Mode I2C dikonfigurasi sebagai mode *default*, tetapi antarmuka dapat diubah dengan mengatur *pads* seperti yang ditunjukkan pada Gambar 2.2. (PN532 NFC RFID Module User Guide)



**Gambar 2.2 Pengaturan antarmuka NFC**

Sumber: PN532 NFC RFID Module User Guide

Pengaturan *pads* dilakukan untuk dapat menggunakan antarmuka yang diinginkan, baik itu SPI, I2C atau HSU, seperti yang ditunjukkan pada Tabel 2.2

**Tabel 2.2 Pengaturan *pads* NFC**

<b>Antarmuka yang digunakan</b>	<b>Channel 1</b>	<b>Channel 2</b>
HSU	Off	Off
I2C	On	Off
SPI	Off	On

#### 2.2.1.3. *Tags* NFC

*Tag* NFC merupakan alat untuk penyimpanan informasi yang apabila didekatkan dengan *reader* NFC maka informasi dapat diakses oleh *Reader* NFC. Ada empat tipe *tag* NFC yang sudah sering digunakan. Pada Tabel 2.3 menunjukkan perbedaan dari tipe *tag* NFC yang mempunyai perbedaan spesifikasi.

**Tabel 2.3 Tipe-tipe *tag* NFC**

	Tipe 1	Tipe 2	Tipe 3	Tipe 4
Produk yang kompatibel	Broadcom Topaz	NXP Mifare Ultralight, NXP Mifare Ultralight C, NXP NTAG203	Sony FeliCa	NXP DeSfire/ NXP SmartMX- JCOP
Ukuran memori	96 Bytes	48 Bytes/144 Bytes	1, 4, 9 Kbytes	4 KBytes/32 KBytes
Harga	Murah	Murah	Mahal	Sedang/Mahal

Akses Data	<i>Read/write</i> atau <i>read-only</i>	<i>Read/write</i> atau <i>read-only</i>	<i>Read/write</i> atau <i>read-only</i>	<i>Read/write</i> atau <i>read-only</i>
------------	--	--	--	--

*Tag* NFC juga memiliki bermacam-macam bentuk dan ukuran seperti yang ditunjukkan pada Gambar 2.4. Perangkat pasif pada NFC sangat sederhana dan antena tetap bisa diletakan pada bentuk yang kecil seperti gantungan kunci, kartu identitas dan lain-lain. Walaupun begitu, *tag* NFC ini tetap aman dan terlindungi untuk penggunaan publik. (Mohamad Syafiq, Electrical Engineering Universiti Teknologi Malaysia, 2013).



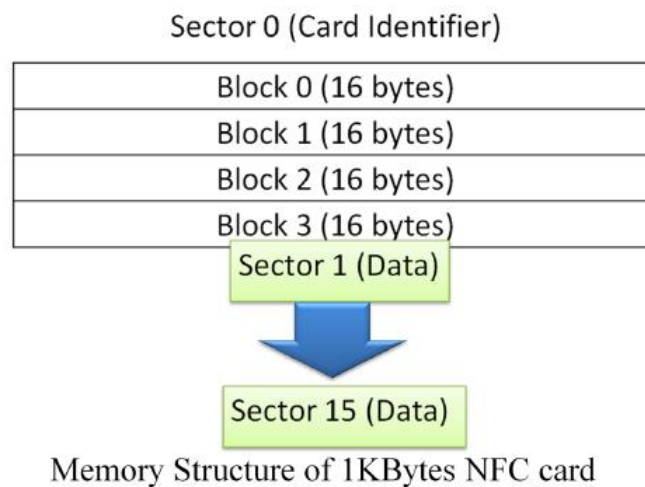
**Gambar 2.3 *Tag* NFC**

Sumber: Mohamad Syafiq, Electrical Engineering, Universiti Teknologi Malaysia, 2013

#### 2.2.1.4. Struktur Memori Kartu NFC

Memori pada kartu NFC ini terbagi menjadi blok-blok dan dikelompokkan ke dalam sektor-sektor. Contohnya, 1 *KBytes* dari memori itu memiliki 16 sektor, dimana setiap sektor terdiri dari 4 blok, dan setiap blok terdiri dari 16 *bytes*. Dengan mengalikkan semua angkanya, maka akan diperoleh memori 1 *KBytes*. Gambar 2.4 adalah ilustrasi blok memori dari kartu NFC. (Mohamad Syafiq, Electrical Engineering Universiti Teknologi Malaysia, 2013)





**Gambar 2.4 Blok memori kartu NFC**

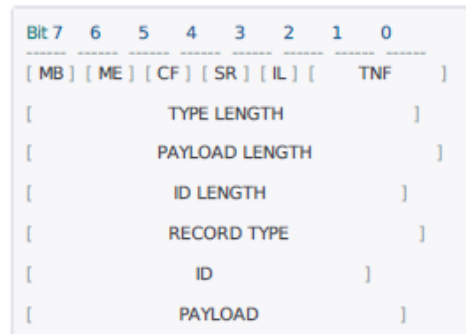
Sumber: Mohamad Syafiq, Electrical Engineering, Universiti Teknologi Malaysia, 2013

Semua sektor terdiri dari 3 buah blok yang masing-masing memiliki 16 *bytes* untuk menyimpan data. Untuk sektor ke-0 hanya terdiri dari 2 buah blok data dan blok *read-only*. Alasan memori terbagi menjadi blok – blok yang berbeda adalah agar memberikan keamanan pada setiap sektor. Apabila ingin mengakses setiap sektor, maka sektor harus di autentikasi sebelum mengakses data. Setelah proses autentikasi sukses, kemudian data pada kartu bisa dibaca, ditulis, dikurang atau ditambah. Metode ini memberikan keamanan akses data disetiap sektor dari pengguna yang tidak dikehendaki. Tetapi, struktur ini hanya ada pada 1 *KBytes Mifare Card Classic* dan tidak sama dengan tipe *Mifare card* lainnya seperti *ultralight Mifare Card*. (Mohamad Syafiq, Electrical Engineering Universiti Teknologi Malaysia, 2013)

#### 2.2.1.5. Format NDEF

Memori pada kartu NFC terdiri dari informasi yang menawarkan untuk melakukan pertukaran data diantara NFC *reader*. Proses pertukaran data membutuhkan sebuah format yang dapat mengenali tipe data yang disimpan pada

kartu NFC. Gambar 2.5 menunjukkan rincian format yang diperlukan oleh NFC reader. (Mohamad Syafiq, Electrical Engineering Universiti Teknologi Malaysia, 2013)



**Gambar 2.5 Rincian data pada sector**

Sumber: Mohamad Syafiq, Electrical Engineering, Universiti Teknologi Malaysia, 2013

Gambar 2.5 menjelaskan fungsi dari 7 bytes pertama sebelum data yang disebut dengan *header*. Tabel 2.4 menunjukkan fungsi dari *header* data.

**Tabel 2.4 Rincian header**

Sumber: Mohamad Syafiq, Electrical Engineering, Universiti Teknologi Malaysia, 2013

Header	Function	Value
MB:	represent as data begin	0 and 1
ME:	represent as data end	0 and 1
CF	represent as chunk flag	0 and 1
SR	represent as short message	0 and 1
IL	ID length	0 and 1
TNF:	represent a type name format field	0 and 1
Type length	data type length in byte	8bits/0-255
Payload length	the length of data based on SR	Depend on SR If SR 1, 0-255bytes If SR 0, 4 bytes
ID length	length of ID	8bits/0-255
Record type	type of data	8bits/0-255
ID: identity	identity of the card	8bits/0-255
Payload	The exact data length	8bits/0-255

### 2.2.2. Arduino Uno

Arduino adalah sebuah nama produk desain sistem minimum mikrokontroler yang di buka secara bebas. Kelebihan dari arduino adalah Arduino mempunyai bahasa pemrograman sendiri, pemrograman yang digunakan adalah bahasa C yang telah dipermudah dengan fungsi-fungsi yang sederhana sehingga pemula pun bisa mempelajarinya dengan cukup mudah. Arduino juga memiliki program yang namanya *boot loader* yang sudah di tanam pada mikrokontrolernya, *boot loader* ini sendiri berfungsi untuk menjembatani antara *software compiler* arduino dengan mikrokontrolernya yang berfungsi untuk mengontrol dalam bentuk yang kecil. Di sini mikrokontroler memiliki memori sendiri, serta proses-proses yang dapat berdiri sendiri, sehingga ketika dihubungkan dengan input dan output yang lain, pengguna juga dapat mengontrol alat tersebut, (M. Irfan Rischi, 2014).

Arduino UNO adalah sebuah papan mikrokontroler yang didasarkan pada ATmega328. Arduino UNO mempunyai 14 pin data *input/output* (6 di antaranya dapat digunakan sebagai keluaran PWM), 6 pin masukan analog, sebuah osilator Kristal 16 MHz, sebuah koneksi USB, sebuah *power jack*, sebuah ICSP *header* dan sebuah tombol *reset*. Arduino UNO mudah sangat mudah untuk dihubungkan ke sebuah komputer dengan sebuah kabel USB. (M. Irfan Rischi, 2014).

Arduino Uno berbeda dari semua Arduino yang telah dibuat sebelumnya, Arduino UNO tidak menggunakan *chip driver* FTDI USB-*to*-*serial*. Sebaliknya, fitur-fitur Atmega16U2 (Atmega8U2 sampai ke versi R2) diprogram sebagai sebuah pengubah USB ke serial. Revisi 2 dari Arduino Uno mempunyai sebuah resistor yang menarik garis 8U2 HWB ke *ground*, yang membuatnya lebih mudah

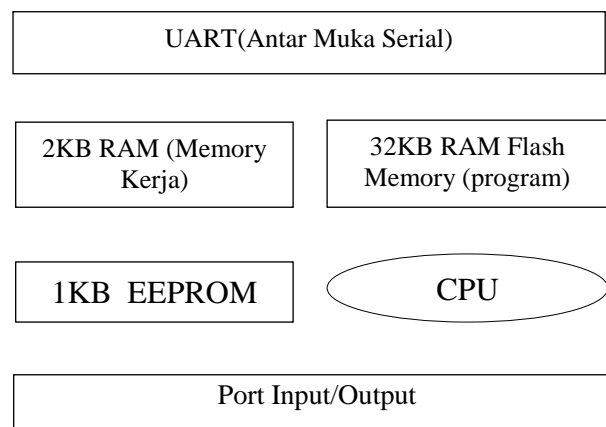
untuk diletakkan ke dalam DFU mode. Revisi 3 dari Arduino Uno memiliki fitur-fitur baru seperti yang ditunjukkan pada Tabel 2.5.

**Tabel 2.5 Fitur – fitur Arduino Uno**

<b>Fitur</b>	<b>Fungsi</b>
Pin input/output digital 14 buah	Sebagai input atau output, dapat diatur oleh program. Khusus untuk 6 buah pin 3, 5, 6, 9, 10 dan 11 dapat juga berfungsi sbagai pin analog output dimana tegangan outputnya dapat diatur. Nilai sebuah pin output analog dapat diprogram antar 0 – 255, dimana hal itu mewakili nilai tegangan 0 – 5 Volt.
USB	Dapat memuat program dari komputer ke dalam <i>microcontroller</i> dengan menggunakan komunikasi serial, serta memberikan catu daya ke rangkaian Arduino.
Sambungan SV1	Sambungan atau <i>jumper</i> untuk memilih sumber daya rangkaian dari sumber eksternal atau dari USB. Sambungan ini tidak diperlukan lagi dalam Arduino versi terakhir karena pemilihan sumber daya eksterna atu USB dilakukan secara otomatis.
Q1 – Kristal ( <i>Quartz Crystal Oscillator</i> )	Menghasilkan detak-detak yang dikirim kepada <i>microcontroller</i> agar melakukan sebuah operasi untuk setiap detaknya. Kristal ini dipilih yang berdetak 16 juta kali per detik atau 16 MHz.
Tombol Reset S1	Untuk mereset rangkaian sehingga program akan mulai dari awal, bukan untuk menghapus program atau mengosongkan <i>microcontroller</i> .
<i>In-CircuitSerial Programming (ICSP)</i>	Port ini digunakan untuk memprogram <i>microcontroller</i> secara langsung, tanpa melakukan <i>bootloader</i> . Umumnya pengguna Arduino tidak melakukan ini sehingga ICSP tidak terlalu dipakai walaupun disediakan.
IC 1 - <i>Microcontroller</i>	Komponen utama dari rangkaian Arduino, di dalamnya terdapat CPU, ROM dan RAM.

X 1 – Sumber daya eksternal	Jika hendak di supply dengan sumber daya eksternal, rangkaian Arduino dapat diberikan tegangan DC antara 7 – 12 V.
Pin input analog 6 buah (0-5)	Dapat membaca tegangan yang dihasilkan oleh sensor analog, seperti sensor suhu. Program dapat membaca nilai sebuah pin input antara 0 – 1023, dimana hal itu mewakili tegangan 0 – 5 Volt.

Arduino UNO dan versi 1.0 akan menjadi referensi untuk versi-versi Arduino selanjutnya. Arduino UNO adalah sebuah seri terakhir dari board Arduino USB dan model referensi untuk papan Arduino. Konfigurasi dari Arduino Uno ditunjukkan pada Gambar 2.6.



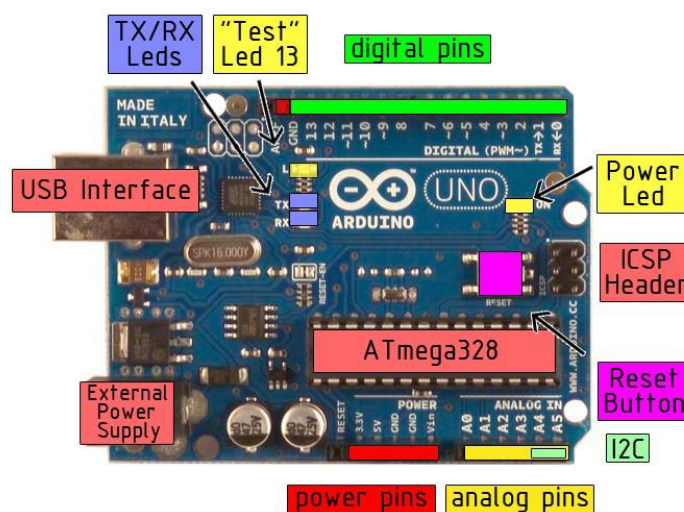
**Gambar 2.6 Konfigurasi Arduino Uno**

- 1) *Universal Asynchronous Receiver/Transmitter (UART)* adalah antar muka yang digunakan untuk komunikasi serial seperti pada RS-232, RS-422 dan RS-485.
- 2) 2KB RAM pada memory kerja bersifat *volatile* (hilang saat daya dimatikan), digunakan oleh variable-variabel di dalam program.

- 3) 32KB RAM flash memory bersifat *non-volatile*, digunakan untuk menyimpan program yang dimuat dari komputer. Selain program, flash memory juga menyimpan *bootloader*. *Boot loader* adalah program inisialisasi yang ukurannya kecil, dijalankan oleh CPU saat daya dihidupkan. Setelah *boot loader* selesai dijalankan, berikutnya program di dalam RAM akan dieksekusi.
- 4) 1KB EEPROM bersifat *non-volatile*, digunakan untuk menyimpan data yang tidak boleh hilang saat daya dimatikan. Tidak digunakan pada papan Arduino.
- 5) *Central Processing Unit (CPU)*, bagian dari *mikrokontroler* untuk menjalankan setiap instruksi dari program.
- 6) Port input/output, pin-pin untuk menerima data (input) digital atau analog, dan mengeluarkan data (output) digital atau analog.

Kemudian bagian-bagian dari papan Arduino akan ditunjukkan pada Gambar

2.7.



**Gambar 2.7 Papan Arduino Uno**

Sumber: M. Irfan Rischi, 2014

Selain berfungsi sebagai penghubung untuk pertukaran data, kabel USB ini juga akan mengalirkan arus DC 5 volt kepada papan Arduino sehingga praktis tidak diperlukan sumber daya dari luar. Saat mendapat suplai daya, lampu LED indikator daya pada papan Arduino akan menyala menandakan bahwa ia siap bekerja. Pada papan Arduino Uno terdapat sebuah LED kecil yang terhubung ke pin digital no 13. LED ini dapat digunakan sebagai output saat seorang pengguna membuat sebuah program dan membutuhkan sebuah penanda dari jalannya program tersebut. (M. Irfan Rischi, 2014).

Setiap 14 pin masukan digital dapat digunakan sebagai masukan atau keluaran dengan menggunakan *pinMode()*, *digitalWrite()*, dan *digitalRead()* pada program. Setiap pin bisa memberikan atau menerima maksimal 40 mA, dan memiliki resistor *pull-up* internal sekitar 20 sampai 50 Kohm. Beberapa pin memiliki fungsi khusus, yaitu :

- 1) *Serial* : pin 0 (RX) dan pin 1 (TX) digunakan untuk menerima dan mengirim data dengan *serial*. Dua pin ini terhubung dengan pin yang sama pada Atmega8U2 USB ke TTL *chip serial*.
- 2) Interupsi eksternal : pin 2 dan pin 3 dapat diatur menjadi pemicu sebuah interupsi pada logika 0, atau mengubah logika.
- 3) PWM : pin 3, 5, 6, 9, 10 dan 11 menyediakan 8bit keluaran PWM dengan menggunakan fungsi *analogWrite()*.
- 4) SPI : pin 10 (SS), 11 (MOSI), 12(MISO) dan 13 (SCK) dapat digunakan sebagai pin komunikasi SPI.

- 5) LED : pin 13 memiliki sebuah LED yang apabila diberikan logika 1 pada pin 13 maka LED akan menyala, dan jika logika 0 maka LED akan mati.

Arduino juga memiliki 6 buah pin masukan analog, setiap pin memiliki resolusi 10 bit. Selain itu, pada pin masukan analog beberapa pin memiliki fungsi khusus, yaitu pin 4 (SDA) dan pin 5 (SCL) yang dapat digunakan untuk antarmuka I2C. Terdapat pin AREF yang merupakan tegangan referensi untuk masukan analog, pin *reset* untuk memberikan logika 0 pada mikroprosesor, dan mengulang program yang sedang dijalankan bukan menghapus program pada mikroprosesor. (M. Irfan Rischi, 2014).

#### 2.2.2.1. *Software* Arduino

Sehubungan dengan pembahasan untuk saat ini *software* Arduino yang akan digunakan adalah *driver* dan IDE, walaupun masih ada beberapa *software* lain yang sangat berguna selama pengembangan Arduino. IDE Arduino adalah *software* yang sangat canggih ditulis dengan menggunakan Java,. IDE Arduino terdiri dari:

- 1) *Editor* program, sebuah window yang memungkinkan pengguna menulis dan mengedit program dalam bahasa *Processing*. *Processing* adalah bahasa pemrograman tingkat tinggi yang dialektanya sangat mirip dengan C++ dan Java, sehingga pengguna yang sudah terbiasa dengan kedua bahasa tersebut tidak akan menemui kesulitan dengan *Processing*, (Djuandi, 2011).
- 2) *Compiler*, sebuah modul yang mengubah kode program (bahasa *Processing*) menjadi kode biner. Bagaimanapun sebuah mikrokontroler tidak akan bisa



memahami bahasa *Processing*. Yang bisa dipahami oleh mikrokontroler adalah kode biner. Itulah sebabnya *compiler* diperlukan dalam hal ini.

- 3) *Uploader*, sebuah modul yang memuat kode biner dari komputer ke dalam memori di dalam papan Arduino.

#### 2.2.2.2. *Reset Otomatis (Software)*

Dari pada mengharuskan sebuah penekanan fisik dari tombol reset sebelum sebuah penguploadan, Arduino Uno didesain pada sebuah cara yang memungkinkannya untuk *direset* dengan *software* yang sedang berjalan pada komputer yang sedang terhubung. Salah satu garis kontrol aliran hardware (DTR) dari ATmega8U2/16U2 dihubungkan ke garis reset dari ATmega328 melalui sebuah kapasitor 100 nanoFarad. Ketika saluran ini dipaksakan (diambil rendah), garis reset jatuh cukup panjang untuk mereset chip. *Software* Arduino menggunakan kemampuan ini untuk memungkinkan kita untuk mengupload kode dengan mudah menekan tombol upload di *software* Arduino. Ini berarti bahwa *bootloader* dapat mempunyai sebuah batas waktu yang lebih singkat, sebagai penurunan dari DTR yang dapat menjadi koordinasi yang baik dengan memulai pengunggahan. (M. Irfan Rischi, 2014).

Arduino Uno berisikan sebuah jalur yang dapat dihapus untuk mencegah reset otomatis. tombol pada salah satu sisi dari jalur dapat disolder bersama untuk mengaktifkan kembali. Tombol itu diberi label "RESET-RN" agar dapat menonaktifkan *reset* otomatis dengan menghubungkan sebuah resistor 110 ohm dari tegangan 5 volt ke garis *reset*. (M. Irfan Rischi, 2014).

### 2.2.2.3. Proteksi Arus lebih USB (*Universal Serial Bus*)

Arduino UNO mempunyai sebuah sekering *reset* yang memproteksi *port* USB komputer dari hubungan pendek dan arus lebih. Walaupun sebagian besar komputer menyediakan proteksi internal sendiri, sekering menyediakan sebuah proteksi tambahan. Jika lebih dari 500 mA diterima *port* USB, sekering secara otomatis akan memutuskan koneksi sampai hubungan pendek atau kelebihan beban hilang. (M. Irfan Rischi, 2014).

### 2.2.2.4. Komunikasi

Arduino Uno memiliki sejumlah fasilitas untuk berkomunikasi dengan komputer, Arduino lain, atau mikrokontroler lain. ATmega328 ini menyediakan UART TTL (5V) komunikasi serial, yang tersedia pada pin digital 0 (RX) dan 1 (TX). Sebuah ATmega16U2 pada saluran board ini komunikasi *serial* melalui USB dan muncul sebagai saluran virtual untuk perangkat lunak pada komputer. *Firmware* Arduino menggunakan USB *driver* standar COM dan tidak ada *driver* eksternal yang dibutuhkan. Perangkat lunak Arduino termasuk layar *serial* yang memungkinkan data sederhana yang akan dikirim ke Arduino. LED RX dan TX pada papan Arduino akan berkedip ketika data sedang dikirim melalui *chip* USB-*to-serial* dan koneksi USB ke komputer. ATmega328 ini juga mendukung komunikasi I2C (*Inter Integrated Circuit*), SPI (*Serial Peripheral Interface*). Fungsi ini digunakan untuk melakukan komunikasi *inteface* pada sistem. (M. Irfan Rischi, 2014).

### 1) Komunikasi *Serial*

Komunikasi *serial* merupakan komunikasi data dengan pengiriman data satu per satu pada satuan waktu. Transmisi data pada komunikasi *serial* dilakukan per bit. Kelebihan dari komunikasi *serial* dibandingkan komunikasi paralel adalah jalur data yang dibutuhkan hanya dua, yaitu jalur *Transmitter* (Tx) dan jalur *Receive* (Rx), selain itu kelebihan lainnya adalah komunikasi data dapat dilakukan dalam jarak yang cukup jauh dengan jumlah kabel *serial* lebih sedikit. Kekurangan dari komunikasi *serial* adalah waktu yang diperlukan untuk pengiriman dan penerimaan data lebih lama. Komunikasi *serial* pada umumnya memiliki dua mode :

#### a) Sinkron

Pada mode sinkron data dikirim bersamaan dengan sinyal *clock*, hal ini menyebabkan antara satu karakter dengan karakter lainnya memiliki jeda waktu yang sama.

#### b) Asinkron

Mode asinkron ini pengiriman data dikirim tanpa sinyal *clock*/sinkronasi sinyal *clock*. Oleh karena itu pada mode asinkron *Transmitter* yang mengirimkan data harus menyetujui suatu standar *Universal Asynchronous Receive Transmit* (UART) sehingga komunikasi data dilakukan dengan suatu standar yang telah disepakati antara *Transmitter* dan *Receiver*. Dalam pengaturan UART terdapat perintah-perintah yang berguna sebagai pengaturan yaitu *start* bit, data bit, *parity* bit dan juga *stop* bit. Dibawah ini akan dijelaskan mengenai perintah-perintah diatas :

- a) *Start Bit* merupakan penanda awal dimana akan dilakukan suatu proses pengiriman bit data.
- b) *Data Bit* merupakan data yang akan dikirim.
- c) *Parity Bit* berfungsi sebagai “*flag*”, atau bisa dikatakan sebagai penanda.
- d) *Stop Bit* berguna sebagai penanda proses pengiriman bit data telah selesai.

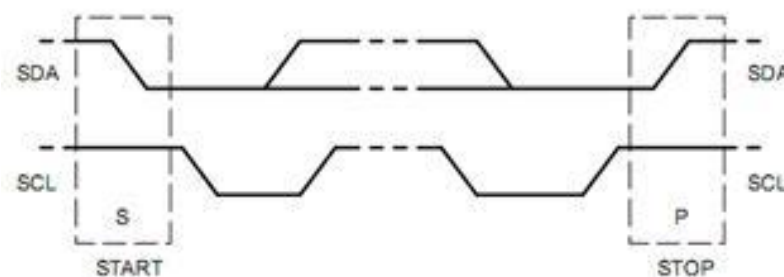
Dalam pengiriman data secara digital terdapat dua buah ukuran yang penting untuk diketahui, yaitu *Bit Rate* dan *Baud Rate*. Perbedaan antara *Bit Rate* dan *Baud Rate* yaitu :

- a) *Bit Rate* adalah jumlah dari bit yang terkirim atau diterima per satuan waktu (*second*).
- b) *Baud Rate* banyaknya perubahan data yang terjadi per satuan waktu. Pada komunikasi serial umumnya jumlah data yang dikirim adalah satu bit *start*, delapan bit data, dan satu bit *stop* sehingga dalam satu *frame* data terdapat sepuluh bit dengan *baud rate* 9600. (M. Irfan Rischi, 2014).

## 2) I2C (*Inter Integrated Circuit*)

I2C (*Inter Integrated Circuit*) adalah standar komunikasi serial dua arah menggunakan dua saluran yang didesain khusus untuk mengirim maupun menerima data. Sistem I2C (*Inter Integrated Circuit*) terdiri dari SCL (*Serial Clock*) dan SDA (*Serial Data*) yang membawa informasi data antara I2C (*Inter Integrated Circuit*) dengan pengontrolnya, (Purnomo Sejati, 2011).

Piranti yang dihubungkan dengan sistem I2C (*Inter Integrated Circuit*) Bus dapat dioperasikan sebagai *master* dan *slave*. *Master* adalah piranti yang memulai transfer data pada I2C (*Inter Integrated Circuit*) Bus dengan membentuk sinyal *start*, mengakhiri transfer data dengan membentuk sinyal *stop* dan membangkitkan sinyal *clock*. *Slave* adalah piranti yang diamati oleh *master*. Sinyal *start* merupakan sinyal untuk memulai semua perintah, didefinisikan sebagai perubahan tegangan SDA (*Serial Data*) dari “1” menjadi “0” pada saat SCL (*Serial Clock*) “1”. Sedangkan sinyal *stop* merupakan sinyal untuk mengakhiri semua perintah, didefinisikan sebagai perubahan tegangan SDA (*Serial Data*) dari “0” menjadi “1” pada saat SCL (*Serial Clock*) “1”. Kondisi sinyal *start* dan *stop* ditunjukkan pada Gambar 2.11.

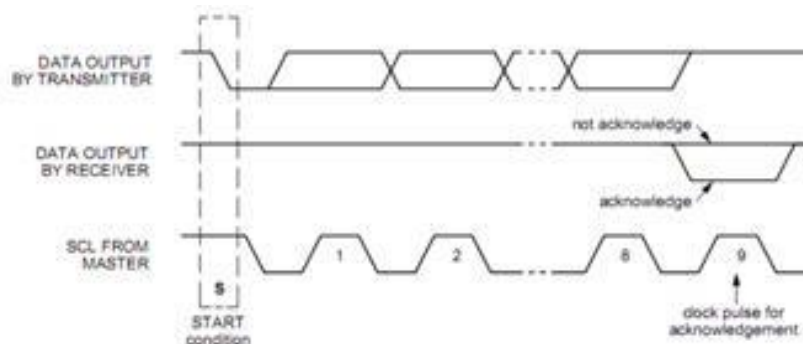


**Gambar 2.8** Kondisi sinyal *start* dan *stop*

Sumber: Purnomo Sejati, 2011

Sinyal dasar yang lain dalam I2C (*Inter Integrated Circuit*) Bus adalah sinyal *acknowledge* yang disimbolkan dengan ACK. Setelah transfer data oleh *master* berhasil diterima oleh *slave*, *slave* akan menjawabnya dengan mengirim sinyal *acknowledge*, yaitu dengan membuat SDA (*Serial Data*) menjadi “0” selama siklus *clock* ke 9. Ini menunjukkan bahwa *slave* telah menerima 8 bit data dari

*master*. Pada Gambar 2.12 menunjukkan kondisi sinyal *acknowledge*. (Purnomo Sejati, 2011).



**Gambar 2.9 Sinyal ACK dan NACK**

Sumber: Purnomo Sejati, 2011

Dalam melakukan transfer data pada I2C (*Inter Integrated Circuit*) *Bus*, ada beberapa tata cara yang telah ditetapkan, yaitu :

- a) Transfer data hanya dapat dilakukan ketika *Bus* tidak dalam keadaan sibuk.
- b) Selama proses transfer data, keadaan data pada SDA (*Serial Data*) harus stabil selama SCL (*Serial Clock*) dalam keadaan tinggi. Keadaan perubahan "1" atau "0" pada SDA (*Serial Data*) hanya dapat dilakukan selama SC (*Serial Clock*) dalam keadaan rendah. Jika terjadi perubahan keadaan pada SDA (*Serial Data*) pada saat SCL (*Serial Clock*) dalam keadaan tinggi, maka perubahan itu dianggap sebagai sinyal *start* atau sinyal *stop*.

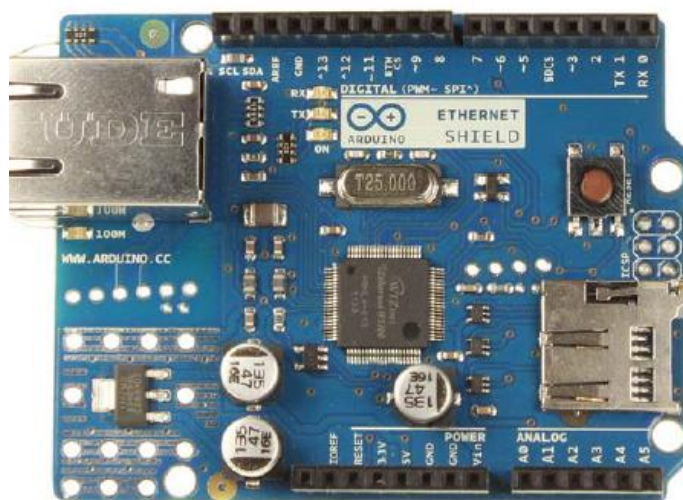
### 2.2.3. Ethernet Shield

*Ethernet shield* digunakan sebagai penghubung Arduino Uno ke internet. Cara penggunaan dengan menghubungkan kabel UTP pada Arduino Uno dan

*ethernet shield*, kemudian mengikuti beberapa instruksi maka Arduino Uno dapat mengakses internet dan *Local Area Network* (LAN). Keduanya biasa digunakan untuk mengirim dan menerima informasi melalui internet.

Kecepatan koneksi *ethernet shield* mencapai 10/100Mb dengan koneksi melalui *port Serial Peripheral Interface* (SPI). Ethernet dapat menghubungkan Arduino Uno ke internet berdasarkan penggunaan Wiznet W5100 sebagai *chip* pada *ethernet shield*. Wiznet W5100 menyediakan sebuah jaringan yang mampu mengatur TCP dan UDP. (M. Irfan Rischi, 2014).

*Ethernet shield* juga memiliki tempat untuk kartu micro-SD yang dapat digunakan untuk menyimpan data - data yang dikirim melalui internet. Selain itu juga memiliki sebuah kontrol *reset*, untuk memastikan bahwa modul *shield* W5100 melakukan *reset* dengan baik saat dinyalakan. *Ethernet shield* bisa juga dihubungkan dengan perangkat elektronik lain melalui pin – pin pada bagian atas dan bawah yang sudah tersedia papan rangkaian. Gambar 2.8 merupakan tampak depan dari *ethernet shield*. (M. Irfan Rischi, 2014).



**Gambar 2.10 Ethernet shield**

Sumber: M. Irfan Rischi, 2014

Adapun fitur - fitur yang dimiliki oleh modul ethernet yaitu sebagai berikut :

- a. Memiliki keluaran *ripple* dan *noise* yang rendah sekitar 100mVpp
- b. Memiliki pengaman untuk beban berlebih dan hubungan arus pendek
- c. Tegangan keluaran sebesar 9 volt
- d. Memiliki efisiensi yang tinggi dalam mengkonversikan DC ke DC
- e. Kecepatan koneksi 10/100Mb
- f. Menggunakan antarmuka SPI untuk koneksi dengan Arduino
- g. Bekerja dengan tegangan masukan 5 volt yang disuplai dari Arduino

Arduino berkomunikasi dengan W5100 dan kartu SD menggunakan antarmuka SPI (*Serial Peripheral Interface*) yang terhubung dengan pin 10, 11, 12 dan 13 pada masukan digital. Maka, pin – pin tersebut tidak dapat digunakan sebagai masukan atau keluaran data. Untuk itu, karena W5100 dan kartu SD menggunakan antarmuka SPI (*Serial Peripheral Interface*), maka hanya bisa menggunakan salah satunya saja. Dengan cara mengeluarkan logika 1 pada pin 4 keluaran digital untuk mematikan kartu SD, sedangkan untuk mematikan W5100 dengan memberikan logika 1 pada pin 10 keluaran digital. Pada saat menjalankan *ethernet shield* yang sudah terhubung ke catu daya dan kabel UTP, maka beberapa LED akan menyala, yang terdiri dari :

- a. PWR : sebagai indikasi *ethernet shield* menyala
- b. LINK : sebagai indikasi adanya hubungan ke jaringan dan berkedip-kedip saat *ethernet* menerima atau mengirim data.
- c. FULLD : sebagai indikasi bahwa sambungan ke jaringan *full duplex*

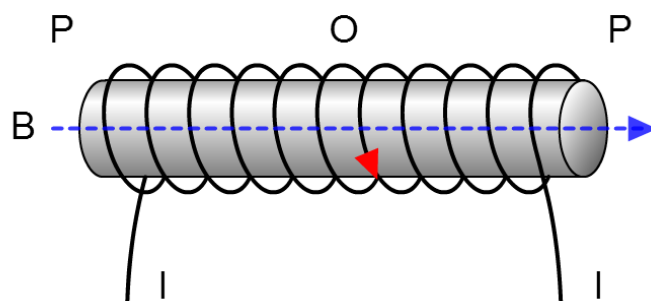


- d. 100M : sebagai indikasi adanya sambungan ke jaringan 100Mb/s
- e. RX : berkedip ketika *ethernet* sedang menerima data (*Receive*)
- f. TX : berkedip ketika *ethernet* sedang mengirim data (*Transmit*)
- g. COLL : berkedip ketika adanya jaringan yang tabrakan

#### 2.2.4. Solenoid

*Solenoid* adalah kumparan kawat berbentuk tabung panjang dengan lilitan yang sangat rapat. Sebuah kawat dibentuk seperti spiral yang disebut sebagai kumparan. Apabila dialiri arus listrik, maka akan berfungsi sebagai magnet batang, (Pustekkom Depdiknas, 2007).

Pada Gambar 2.13 menunjukkan arah dari arus listrik dengan simbol warna merah, dan arah medan magnet setelah dialiri arus listrik.

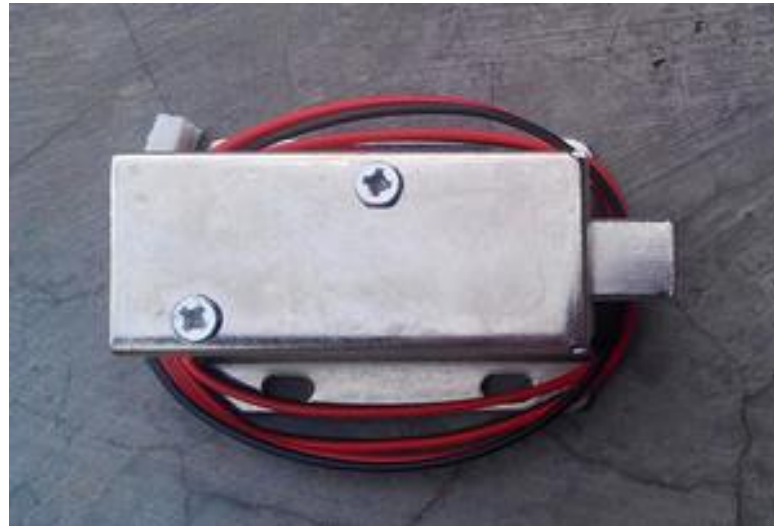


**Gambar 2.11 Arah arus dan medan magnet**

Sumber: Pustekkom Depdiknas, 2007

*Solenoid* memiliki berbagai macam jenis bentuk yang sering digunakan, salah satunya *solenoid door lock* seri LY-03 seperti yang ditunjukkan pada Gambar 2.14. Pada *solenoid* ini, tidak memiliki polaritas seperti motor dan dapat bekerja dengan masukan tegangan 12 Volt. Ketika diberi tegangan masukan sebesar 12 Volt, maka akan ada arus listrik yang mengalir pada lilitan. Maka akan

menyebabkan adanya medan magnet disekitar kumparan, hal itu membuat *plunger* bergerak diantara lilitan tersebut.



**Gambar 2.12 Solenoid door lock seri LY-03**

Sumber: <https://www.bukalapak.com/p/elektronik/lain-lain-208/4scug-jual-solenoid-doorlock-seri-ly-03-dc12v>

#### **2.2.5. RTC (Real Time Clock)**

RTC adalah jenis pewaktu yang bekerja berdasarkan waktu yang sebenarnya atau dengan kata lain berdasarkan waktu nyata yang ada pada jam. Agar dapat berfungsi, pewaktu ini membutuhkan dua parameter utama yang harus ditentukan, yaitu pada saat mulai (*start*) dan pada saat berhenti (*stop*). RTC berbentuk suatu chip (IC) yang memiliki fungsi sebagai penyimpan waktu dan tanggal.

Dalam proses penyimpanannya RTC sendiri memiliki register yang dapat menyimpan data detik, menit, jam, tanggal, bulan dan tahun. RTC ini

memiliki 128 lokasi RAM yang terdiri dari 15 *bytes* untuk data waktu serta kontrol, dan 113 *bytes* sebagai RAM umum.

(<http://mochincorp.blogspot.co.id/2012/10/jam-7-segment-menggunakan-rtc-real-time.html>).

### 2.2.6. Kartu microSD

MicroSD adalah kartu memori *non-volatile* yang dikembangkan oleh SD Card Association yang digunakan dalam perangkat *portable*. Saat ini, teknologi microSD sudah digunakan oleh lebih dari 400 merek produk serta dianggap sebagai standar industri de-facto.

Keluarga microSD yang lain terbagi menjadi SDSC yang kapasitas maksimum resminya sekitar 2GB, meskipun beberapa ada yang sampai 4GB. SDHC (*High Capacity*) memiliki kapasitas dari 4GB sampai 32GB. Dan SDXC (*Extended Capacity*) kapasitasnya di atas 32GB hingga maksimum 2TB. Keberagaman kapasitas seringkali membuat kebingungan karena masing-masing protokol komunikasi sedikit berbeda.

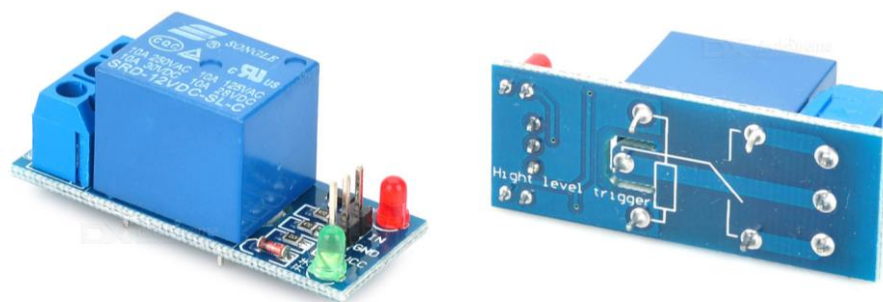
Dari sudut pandang perangkat, semua kartu ini termasuk kedalam keluarga SD. SD adapter memungkinkan konversi fisik kartu SD yang lebih kecil untuk bekerja di slot fisik yang lebih besar dan pada dasarnya ini adalah alat pasif yang menghubungkan pin dari microSD yang kecil ke pin adaptor microSD yang lebih besar.

Protokol komunikasi untuk SDHC/SDXC/SDIO sedikit berbeda dengan microSD yang sudah mapan karena biasanya *host device* keluaran lama tidak bisa mengenali kartu keluaran baru. Kebanyakan masalah mengenai

inkompatibilitas ini dapat diselesaikan dengan *firmware update*.  
(<http://www.harianandroid.com/2014/04/pengertian-dan-fungsi-sd-card.html>)

### 2.2.7. Relay

*Relay* adalah saklar mekanik yang dikendalikan atau dikontrol secara elektronik (elektro magnetik). Saklar pada *relay* akan terjadi perubahan posisi OFF ke ON pada saat diberikan energi elektro magnetik pada armatur *relay* tersebut. *Relay* pada dasarnya terdiri dari 2 bagian utama yaitu saklar mekanik dan sistem pembangkit elektromagnetik (induktor inti besi). Saklar atau kontaktor *relay* dikendalikan menggunakan tegangan listrik yang diberikan ke induktor pembangkit magnet untuk menarik armatur tuas saklar atau kontaktor *relay*. *Relay* memiliki 2 jenis konstruksi, yaitu *relay* elektromekanik posisi *Normally Close* (NC) dan posisi *Normally Open* (NO), berikut pada Gambar 2.15 salah satu bentuk *relay normally close* (NC).  
(<http://elektronika-dasar.web.id/teori-relay-elektro-mekanik>)



**Gambar 2.13 Modul *relay* 5 Volt**

Sumber: <http://elektronika-dasar.web.id/teori-relay-elektro-mekanik>

*Relay* dibutuhkan dalam rangkaian elektronika sebagai eksekutor sekaligus *interface* antara beban dan sistem kendali elektronik yang berbeda sistem *power supply*nya. Secara fisik antara saklar atau kontaktor dengan elektromagnet *relay* terpisah sehingga antara beban dan sistem kontrol terpisah. Bagian utama *relay* elektro mekanik adalah sebagai berikut:

- a. Kumputan elektromagnetik adalah lilitan tembaga untuk menghasilkan medan magnet yang dapat mengendalikan *iron core* menggerakkan *armature*.
- b. Saklar atau kontaktor bagian yang akan terhubung ke beban untuk dialiri arus melalui *armature*.
- c. *Swing Armature* adalah bagian yang digerakkan oleh *iron core* agar terhubung dengan saklar untuk mengalirkan arus ke beban.
- d. *Spring* (Pegas) adalah bagian untuk memudahkan *armature* berubah posisi.