

## BAB II

### TINJAUAN PUSTAKA

#### 2.1 Studi Pustaka

Pada perancangan sebelumnya yaitu dari penelitian dengan judul “Sistem Pengkondisi Suhu dan Kelembaban Udara pada Ruang Berbasis Mikrokontroler AT89C51” (Oleh Pandu Mayor Hermawan, 2001), terdapat perbedaan yang signifikan dari perancangan “Sistem Monitoring dan Pengendali Suhu Terpusat pada Banyak Inkubator Berbasis Mikrokontroler dan Personal komputer (PC)” yang dibuat. Pada dasarnya, alat yang dibuat menggunakan sensor suhu LM35 sebagai pengindera suhu sama halnya dengan perancangan sebelumnya. Letak perbedaannya yaitu pada perancangan yang dibuat sekarang ini dapat mengendalikan dan memonitoring suhu dari banyak ruangan (inkubator) kemudian ditampilkannya pada sebuah PC.

Dalam perancangan sebelumnya menggunakan mikrokontroler AT89C51 sebagai pengolah data keseluruhan sistem. Sedangkan alat yang dibuat sekarang menggunakan mikrokontroler tipe ATmega8535 sebagai pengolah datanya. Dalam hal instruksi, terdapat perbedaan, yaitu mikrokontroler AT89C51 mempunyai waktu untuk eksekusi instruksi yang lebih lambat dibandingkan dengan mikrokontroler ATmega8535. Dimana, AT89C51 memerlukan 12 siklus clock untuk melaksanakan satu siklus instruksi, sedangkan ATmega8535 hanya membutuhkan 1 siklus saja dalam melakukan eksekusi intruksi.

Demikian juga dalam perancangan sebelumnya, hanya dapat mengendalikan suhu dan kelembaban dalam satu ruangan saja dan menampilkan data suhu pada sebuah LCD. Sedangkan pada perancangan yang dibuat dapat mengendalikan dan memonitoring suhu lebih dari satu ruangan dan menampilkannya pada PC, dan juga dapat mendeteksi keberadaan orang yang ada didalam ruangan tersebut sehingga dapat mengaktifkan dan menonaktifkan sistem secara otomatis.

Pada studi pustaka ini, ada beberapa landasan teori dari komponen utama yang digunakan, yaitu

## **2.2 Mikrokontroler**

Perkembangan teknologi telah maju dengan pesat dalam perkembangan dunia elektronika, khususnya dunia mikroelektronika. Penemuan silikon menyebabkan bidang ini mampu memberikan sumbangan yang amat berharga bagi perkembangan teknologi modern. Atmel sebagai salah satu vendor yang mengembangkan dan memasarkan produk mikroelektronika telah menjadi suatu teknologi standar bagi para desainer elektronika masa kini.

Mikrokontroler yang digunakan dalam perancangan tugas akhir ini yaitu menggunakan ATmega8535 yang merupakan generasi AVR (*Alf and Vegard's Risc Processor*) dari vendor ATMEL. Mikrokontroler AVR memiliki arsitektur RISC 8 bit, dimana semua instruksi dikemas dalam kode 16-bit (*16-bits words*) dan sebagian besar instruksi dieksekusi dalam 1 (satu) siklus clock. Gambar 2.2 memperlihatkan arsitektur yang dimiliki mikrokontroler ATmega8535.

### 2.2.1 Arsitektur Mikrokontroler ATmega8535

Adapun blok diagram dari Arsitektur ATmega8535 :



**Gambar 2.1** Blok Diagram Fungsional ATmega8535

Dari gambar blok diagram tersebut dapat dilihat bahwa ATmega8535 memiliki bagian sebagai berikut :

1. Saluran I/O sebanyak 32 buah, yaitu Port A, Port B, Port C, dan Port D.
2. ADC 10 bit sebanyak 8 saluran.
3. Tiga buah *Timer/Counter* dengan kemampuan pembandingan.
4. CPU yang terdiri atas 32 buah register.
5. *Watchdog Timer* dengan osilator internal.
6. SRAM sebesar 512 byte.
7. Memori *Flash* sebesar 8 kb dengan kemampuan *Read While Write*.
8. Unit interupsi internal dan eksternal.
9. Port antarmuka SPI.
10. EEPROM sebesar 512 byte yang dapat diprogram saat operasi.
11. Antarmuka komparator analog.
12. Port USART untuk komunikasi serial.

### 2.2.2 Fitur ATmega8535

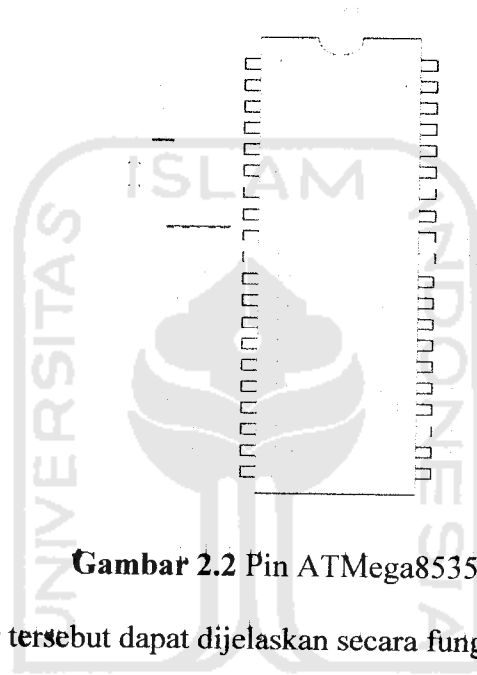
Kapabilitas detail dari ATmega8535 adalah sebagai berikut :

1. Sistem mikroprosesor 8 bit berbasis RISC dengan kecepatan maksimal 16 MHz.
2. Kapabilitas memori flash 8 KB, SRAM sebesar 512 byte, dan EEPROM sebesar 512 byte.
3. ADC internal dengan fidelitas 10 bit sebanyak 8 *channel*.

4. Port komunikasi serial (USART) dengan kecepatan maksimal 2,5 Mbps.
5. Enam pilihan mode *sleep* menghemat penggunaan daya listrik.

### 2.2.3 Konfigurasi Pin ATmega8535

Konfigurasi pin ATmega8535 adalah sebagai berikut :



**Gambar 2.2** Pin ATmega8535

Dari gambar tersebut dapat dijelaskan secara fungsional konfigurasi pin ATmega8535 sebagai berikut :

1. VCC merupakan pin yang berfungsi sebagai pin masukan catu daya.
2. GND merupakan pin *ground*.
3. Port A (PA0..PA7) merupakan pin I/O dua arah dan pin masukan ADC.
4. Port B (PB0..PB7) merupakan pin I/O dua arah dan pin fungsi khusus, yaitu Timer/Counter, komparator analog, dan SPI.

5. Port C (PC0..PC7) merupakan pin I/O dua arah dan pin fungsi khusus, yaitu TWI, komparator analog, dan *Timer Oscillator*.
6. Port D (PD0..PD7) merupakan pin I/O dua arah dan pin fungsi khusus, yaitu komparator analog, interupsi eksternal, dan komunikasi serial.
7. RESET merupakan pin yang digunakan untuk mereset mikrokontroler.
8. XTAL1 dan XTAL2 merupakan pin masukan *clock* eksternal.
9. AVCC merupakan pin masukan tegangan untuk ADC.
10. AREF merupakan pin masukan tegangan referensi ADC.

#### 2.2.6 ADC (Analog to Digital Converter) Internal ATmega8535

Mikrokontroler ATmega8535 memiliki fasilitas Analog to Digital Converter yang sudah *built-in* dalam chip. Fitur ADC internal inilah yang menjadi salah satu kelebihan mikrokontroler ATmega8535 bila dibandingkan dengan beberapa jenis mikrokontroler lain. Dengan adanya ADC internal ini kita tidak dapat direpotkan lagi dengan kompleksitas hardware saat membutuhkan proses pengubahan sinyal dari analog ke digital seperti yang harus dilakukan jika kita memakai komponen IC ADC eksternal.

Spesifikasi dari ADC internal ATmega8535 antara lain :

- a) Resolusi 10 bit
- b) Tegangan catu antara 4,5 5,5 Volt
- c) Tegangan input 0-VCC
- d) Waktu konversi 65-260 uS

- e) Tegangan referensi ADC internal 2,56 Volt
- f) Mode konversi *free running* dan *single conversion*
- g) Interrupt on ADC Conversion Complete
- h) 2 kanal input dengan penguatan 10 dan 200 x (differential)

ADC ini menggunakan dua mode, yaitu *single conversion* dan *free running* untuk melakukan konversi sinyal analog yang masuk ke pin input menjadi sinyal digital. Bilangan biner yang mewakili sinyal analog pada suatu harga tertentu dikenal dengan nama *cuplikan* (sample) sedangkan frekuensi untuk mencuplik disebut laju cuplikan (sampling rate).

#### ❖ Resolusi

Resolusi merupakan perubahan yang dapat terjadi pada keluaran digital sebagai hasil perubahan pada input analog. Proses perubahan data analog menjadi data digital tergantung dari jumlah bit pada suatu ADC yang digunakan. Dimana resolusi adalah besaran terkecil (analog) yang masih dapat dikonversikan menjadi satuan digital.

$$\text{Resolusi} = \left( \frac{1}{2^n} \right) \times V_{\text{ref}} \quad (2.1)$$

Dengan  $n$  : Banyaknya bit ADC

$V_{\text{ref}}$  : Tegangan referensi yang digunakan

#### ❖ Akurasi

Akurasi merupakan spesifikasi yang menunjukkan suatu ADC untuk mengkonversi suatu input analog. Semakin tinggi akurasi yang dimiliki ADC maka

keluaran ADC ini mempunyai besaran yang semakin mendekati nilai yang sebenarnya.

$$\text{NilaiADC} = \frac{V_{in}}{r} \quad (2.2)$$

Dengan : Nilai ADC : Hasil konversi ADC

$V_{in}$  : Tegangan masukan ADC (Volt)

$r$  : Resolusi (Volt)

### 2.2.7 USART

*Universal Synchronous and Asynchronous Serial Receiver and Transmitter* (USART) juga merupakan salah satu mode komunikasi serial yang dimiliki oleh ATmega8535. USART merupakan komunikasi yang memiliki fleksibilitas tinggi, yang dapat digunakan untuk melakukan transfer data baik antar mikrokontroler maupun dengan modul-modul eksternal termasuk PC yang memiliki fitur UART.

USART memungkinkan transmisi data baik secara *synchronous* maupun *asynchronous* sehingga dengan demikian USART pasti kompatibel dengan UART. Pada ATmega8535, secara umum pengaturan mode komunikasi baik *synchronous* maupun *asynchronous* adalah sama. Perbedaannya hanyalah terletak pada sumber clock saja. Jika pada mode *asynchronous* masing-masing peripheral memiliki sumber clock sendiri maka pada mode *synchronous* hanya ada satu sumber clock yang digunakan secara bersama-sama. Dengan demikian secara hardware untuk mode *asynchronous* hanya membutuhkan 2 pin yaitu TXD dan RXD sedangkan untuk



mode *synchronous* harus 3 pin yaitu TXD, RXD dan XCK. Untuk mengatur mode dan prosedur komunikasi USART dilakukan melalui register UCSRA, UCSRB, UCSRC, UBRRH, UBRRL, dan UDR.

### 2.3 Antar Muka RS-232

Komunikasi atau *transfer* data antara mikrokontroler dengan memori atau piranti lain dapat dilakukan secara serial dan paralel. Transmisi serial berarti mengirimkan data satu bit dalam satu satuan waktu melewati satu jalur, berlawanan dengan pengiriman secara paralel yang mengirimkan sejumlah bit sekaligus dalam satu satuan waktu sehingga memerlukan banyak jalur.

Transmisi data serial hanya memakai sebuah jalur untuk pengiriman data artinya pengiriman data dilakukan dengan mengirimkan satu persatu bit dalam satu satuan waktu. Sedangkan dalam transmisi paralel, beberapa buah bit dikirimkan sekaligus dalam satu satuan waktu. Hal ini akan mempercepat proses pengiriman data atau menaikkan *baud rate* pengiriman.

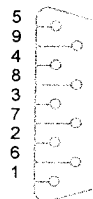
Keuntungan pengiriman data secara serial adalah jumlah kabel yang digunakan lebih sedikit, jika komunikasi paralel minimal memerlukan delapan kabel plus *ground*, maka komunikasi serial hanya butuh dua kabel plus *ground*, selain itu jangkauan panjang kabel lebih jauh dibandingkan paralel karena port serial mengirimkan logika 1 dengan kisaran tegangan -3 volt hingga -15 volt dan logika 0 sebagai +3 volt hingga +15 volt, sehingga kehilangan daya karena panjangnya kabel bukan masalah utama. Bandingkan hal ini dengan port paralel yang menggunakan

level TTL yang berkisar dari 0 volt untuk logika 0 dan +5 volt untuk logika 1. Kerugian dari pengiriman data secara serial adalah kecepatan pengiriman data relatif lebih lambat dan port serial lebih sulit ditangani karena peralatan yang dihubungkan ke port serial harus berkomunikasi menggunakan transmisi serial sedangkan data di komputer diolah secara paralel.

### 2.3.1 Standar RS-232

RS 232 merupakan antar muka peralatan terminal data dengan peralatan komunikasi data menggunakan data biner secara serial. Peralatan terminal data (*Data Terminal Equipment / DTE*) merupakan komputer, sedangkan peralatan komunikasi data (*Data Communication Equipment / DCE*) merupakan modem. Antara komputer dengan modem level sinyal data yang disalurkan pada kabelnya adalah level RS-232.

Pada perkembangannya DCE tidak hanya berupa modem atau perangkat komunikasi, tetapi bisa berupa instrumentasi seperti pH meter, timbangan dan sebagainya. Prinsip pengiriman data secara serial adalah data yang dikirim suatu terminal akan diterima oleh terminal lain demikian juga sebaliknya. Gambar 2.3 merupakan konfigurasi konektor DB-9 pada komunikasi serial.



**Gambar 2.3** Konektor DB-9

Tabel 2.1 menerangkan kegunaan pin konektor DB-9 dalam proses komunikasi data secara serial.

**Tabel 2.1** Konfigurasi Pin Serial

<b>Pin DB9</b>	<b>Nama</b>	<b>Kepanjangan</b>
3	TD	<i>Transmit Data</i>
2	RD	<i>Receive Data</i>
7	RTS	<i>Request To Send</i>
8	CTS	<i>Clear To Send</i>
6	DSR	<i>Data Set Ready</i>
5	SG	<i>Signal Ground</i>
1	CD	<i>Carrier Detect</i>
4	DTR	<i>Data Terminal Ready</i>
9	RI	<i>Ring Indicator</i>

### 2.3.2 Pengkonversi Level Tegangan TTL ke Level Tegangan RS-232

Untuk mengkonversi level tegangan TTL ke level tegangan RS-232 digunakan IC MAX 232. Tegangan tingkat RS-232 sangat jauh berbeda dengan tingkat TTL. Jika TTL bekerja dengan tegangan antara 0 sampai 5 volt, dengan tegangan sekitar 0 volt dianggap sebagai logika '0' dan tegangan disekitar 5 volt sebagai logika '1', sedangkan untuk tingkat RS232 tegangan kerjanya antara -15

sampai +15 volt dan cara menerjemahkan logika '0' dan '1' -nya yang sangat berbeda.

Untuk itu diperlukan piranti khusus yang digunakan untuk melakukan konversi tingkat tegangan TTL dan RS-232. IC MAX 232 ini dapat digunakan dalam dua arah konversi tegangan yaitu dari level tegangan RS-232 ke level tegangan TTL atau sebaliknya. Gambar 2.4 memperlihatkan konfigurasi pin dari IC MAX 232.



**Gambar 2.4** Susunan Pin IC MAX 232

#### **2.4 Sensor Suhu LM35**

Sensor suhu LM35 adalah komponen elektronika yang memiliki fungsi untuk mengubah besaran suhu menjadi besaran listrik dalam bentuk tegangan. Sensor Suhu LM35 yang dipakai dalam penelitian ini berupa komponen elektronika yang diproduksi oleh *National Semiconductor*. LM35 memiliki keakuratan tinggi dan kemudahan perancangan jika dibandingkan dengan sensor suhu yang lain, LM35 juga mempunyai keluaran impedansi yang rendah dan linieritas yang tinggi sehingga dapat dengan mudah dihubungkan dengan rangkaian kendali khusus serta tidak memerlukan penyetelan lanjutan.

Meskipun tegangan sensor ini dapat mencapai 30 volt akan tetapi yang diberikan kesensor adalah sebesar 5 volt, sehingga dapat digunakan dengan catu daya tunggal dengan ketentuan bahwa LM35 hanya membutuhkan arus sebesar 60  $\mu\text{A}$  hal ini berarti LM35 mempunyai kemampuan menghasilkan panas (*self-heating*) dari sensor yang dapat menyebabkan kesalahan pembacaan yang rendah yaitu kurang dari 0,5  $^{\circ}\text{C}$  pada suhu 25  $^{\circ}\text{C}$  .

**Gambar 2.5** Sensor Suhu LM35

Gambar diatas menunjukkan bentuk dari LM35 tampak depan dan tampak bawah. 3 pin LM35 menunjukan fungsi masing-masing pin diantaranya, pin 1 berfungsi sebagai sumber tegangan kerja dari LM35, pin 2 atau tengah digunakan sebagai tegangan keluaran atau  $V_{OUT}$  dengan jangkauan kerja dari 0 Volt sampai dengan 1,5 Volt dengan tegangan operasi sensor LM35 yang dapat digunakan antar 4 Volt sampai 30 Volt. Keluaran sensor ini akan naik sebesar 10 mV setiap derajat *celcius* sehingga diperoleh persamaan sebagai berikut

$$V_{LM35} = \text{Temp} \times 10 \text{ mV} \quad (2.3)$$

Dengan :

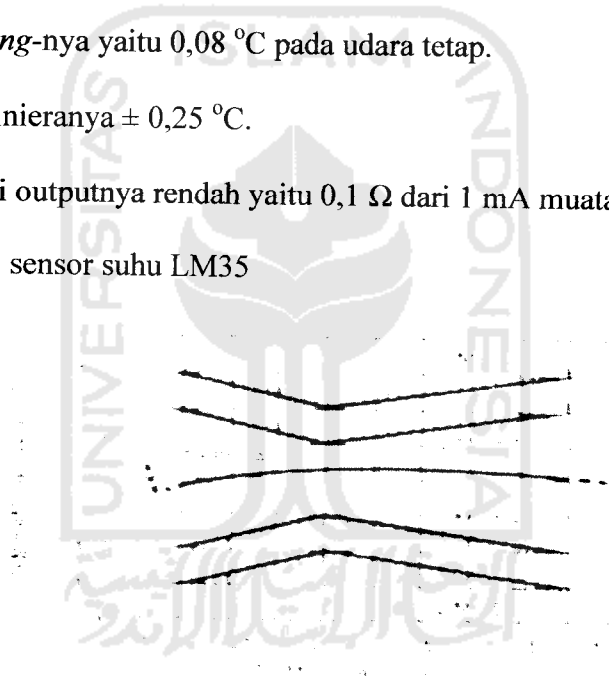
$V_{LM35}$  : Tegangan output dari sensor LM35 (Volt)

Temp : Besarnya suhu yang dibaca sensor ( $^{\circ}\text{C}$ )

Spesifikasi dari Sensor Suhu LM35:

- Dikalibrasi secara langsung ke celcius
- Faktor skala linier +10 mV/°C
- Jaminan akurasinya 0,5 °C pada suhu +25 °C
- Rata-rata temperaturnya antara -55 °C sampai 150 °C.
- Dioperasikan pada 4 volt sampai 30 volt.
- Arus yang dibawa kurang dari 60 uA
- *Self-heating*-nya yaitu 0,08 °C pada udara tetap.
- Ketidak linieranya  $\pm 0,25$  °C.
- Impedansi outputnya rendah yaitu 0,1  $\Omega$  dari 1 mA muatan

Karakteristik dari sensor suhu LM35



**Gambar 2.6** Karakteristik Sensor LM35

## 2.5 Cahaya Inframerah

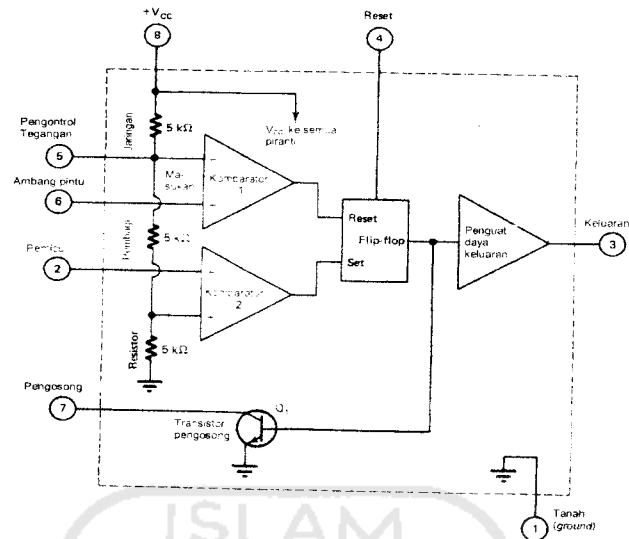
Cahaya inframerah merupakan cahaya yang tidak tampak. Jika di lihat dengan spektroskop cahaya, maka radiasi cahaya inframerah akan nampak pada

spektrum elektromagnet dengan panjang gelombang diatas panjang gelombang cahaya merah yaitu  $10^{-3}$  m sampai  $7,8 \times 10^{-7}$  m dengan frekuensi pada range  $10^{12}$  Hz sampai  $10^{14}$  Hz. Dengan panjang gelombang ini maka cahaya inframerah ini akan tidak nampak oleh mata, namun radiasi panas yang ditimbulkannya masih terasa atau dideteksi.

Cahaya inframerah walaupun mempunyai panjang gelombang yang sangat panjang, tetapi tidak dapat menembus bahan-bahan yang tidak dapat melewatkan cahaya yang nampak sehingga cahaya inframerah tetap mempunyai karakteristik seperti halnya cahaya yang nampak oleh mata

### **2.5.1 IC Pembangkit Gelombang**

IC NE/SE 555 adalah piranti multiguna yang telah secara luas digunakan. Piranti ini dapat digunakan sebagai astable multivibrator. Rangkaian khusus ini dapat dibuat dengan komponen dan daya yang minimal. Rangkaian dapat dengan mudah dibuat dan sangat reliabel. Rangkaian internal IC NE555 biasanya dilihat sebagai blok-blok. Dalam hal ini chip memiliki 2 komparator, sebuah bistable flip-flop, sebuah pembagi resistif, sebuah transistor pengosongan dan sebuah keluaran. Gambar berikut memperlihatkan blok fungsional IC NE555.

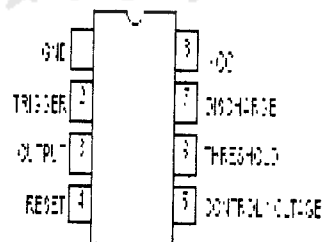


**Gambar 2.7** Blok Diagram IC NE555

Rangkaian IC NE555 dikemas dalam bentuk DIP (*Dual in Line Package*) 8 pin.

Secara garis besar memiliki 4 komponen internal yaitu :

- Output drive
- Pengosongan transistor
- Pembanding (komparator)
- Flip-flop pengendali



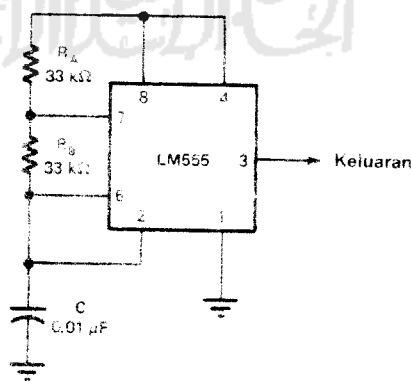
**Gambar 2.8** Konfigurasi Pin IC NE555



Fungsi masing-masing pin IC NE555 adalah sebagai berikut :

- Pin 1 : ground (pentanahan)
- Pin 2 : sebagai pemicu (trigger)
- Pin 3 : sebagai keluaran (output)
- Pin 4 : sebagai reset
- Pin 5 : mengendalikan tingkat keluaran tegangan picu dengan tegangan ambang, memodulasi bentuk gelombang keluaran, melewati gangguan riak tegangan yang mungkin muncul dari catu daya bila dihubungkan dengan kapasitor ke ground.
- Pin 6 : terminal ambang
- Pin 7 : sebagai pengosongan muatan yang dihasilkan selama proses
- Pin 8 : sebagai catu daya (VCC)

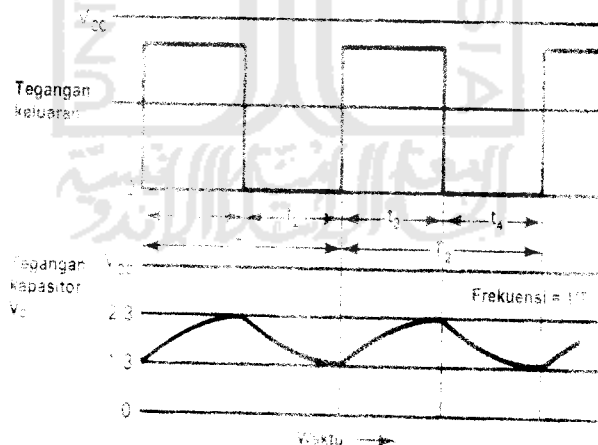
Rangkaian astable dibuat dengan mengubah susunan resistor dan kapasitor luar pada IC 555. Ada dua buah resistor  $R_a$  dan  $R_b$  serta satu kapasitor eksternal yang diperlukan seperti terlihat pada gambar berikut :



**Gambar 2.9** Rangkaian Astable Multivibrator

Saat daya mula-mula diberikan, kapasitor akan terisi melalui  $R_a$  dan  $R_b$ . Ketika tegangan pada pin 6 ada sedikit kenaikan di atas dua pertiga  $V_{cc}$ , maka terjadi perubahan kondisi pada komparator 1. ini akan mereset flip-flop dan keluaran akan bergerak ke positif. Keluaran (pin 3) bergerak ke "tanah" dan basis Q1 berprategangan maju. Q1 mengosongkan C lewat  $R_b$  ke "tanah".

Ketika tegangan pada kapasitor C turun sedikit dibawah sepertiga  $V_{cc}$ , ini akan memberikan energi ke komparator 2. antara pemicu (pin 2) dan pin 6 masih terhubung bersama. Komparator 2 menyebabkan tegangan positif ke masukan set dari flip-flop dan memberikan keluaran negatif. Keluaran (pin 3) akan bergerak ke harga  $+V_{cc}$ . Tegangan Q1 berpanjar mundur. Ini akan membuka proses pengosongan (pin 7). C mulai terisi lagi dengan harga berkisar antara sepertiga dan dua pertiga  $V_{cc}$ . Perfhatkan gelombang yang dihasilkan pada gambar berikut :



**Gambar 2.10** Bentuk Gelombang pada Rangkaian Astable Multivibrator

Frekuensi keluaran astable multivibrator dinyatakan sebagai  $f = 1/T$ . Ini menunjukkan sebagai total waktu yang diperlukan untuk pengisian dan pengosongan kapasitor C. Waktu pengisian ditunjukkan oleh jarak  $t_1$  dan  $t_3$ . jika dinyatakan dalam detik  $t_1 = 0,693 (R_A + R_B)C$ . Waktu pengosongan diberikan oleh  $t_2$  dan  $t_4$ . Dalam detik  $t_2 = 0,693 R_B C$ . Dalam satu putaran atau satu periode pengoperasian waktu yang diperlukan adalah sebesar  $T = t_1 + t_2$  atau  $T = t_3 + t_4$ . Dengan menggunakan harga  $t_1$  dan  $t_2$  atau  $t_3$  dan  $t_4$ , maka persamaan frekuensi dapat dinyatakan sebagai :

$$f = \frac{1}{T} = \frac{1,44}{(R_A + 2R_B)C} \quad (2.4)$$

Dengan : f : Frekuensi (Hz)

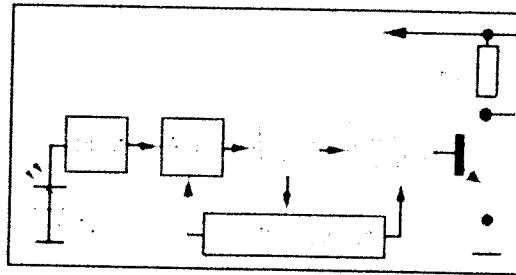
T : Periode (Sekon)

R : Resistansi (Ohm)

C : Kapasitansi (Farad)

### 1.5.2 Penerima Inframerah

Modul penerima TSOP 4838 merupakan suatu modul penerima data melalui gelombang inframerah dengan frekuensi carrier sebesar 38 KHz. Modul penerima inframerah terdiri dari PIN diode sebagai photo detektor dan preamplifier sebagai penguat dalam satu kemasan. Dalam modul ini dilengkapi juga dengan bandpass filter yang hanya dapat melewatkan frekuensi tertentu. Modul ini terbungkus oleh plat yang terhubung dengan ground rangkaian untuk melindungi rangkaian dari interferensi noise. Adapun blok diagram dari modul penerima TSOP 4838



**Gambar 2.11** Blok diagram Modul TSOP 4838

## 2.6 Borland Delphi 7.0

Delphi adalah salah satu pemrograman visual di lingkungan sistem operasi Microsoft Windows, berbasis bahasa PASCAL, sehingga bila kita telah mengetahui konsep dasar dan aturan-aturan yang berlaku dalam pemrograman PASCAL maka untuk masuk ke Delphi akan lebih mudah. Untuk mempermudah pemrogram dalam membuat program aplikasi, Delphi menyediakan fasilitas pemrograman yang sangat lengkap. Fasilitas pemrograman tersebut dibagi dalam dua kelompok, yaitu object dan bahasa pemrograman.

Secara ringkas, object adalah suatu komponen yang mempunyai bentuk fisik dan biasanya dapat dilihat ( visual ). Object biasanya dipakai untuk melakukan tugas tertentu dan mempunyai batasan-batasan tertentu. Sedangkan bahasa pemrograman secara singkat dapat disebut sebagai sekumpulan teks yang mempunyai arti tertentu dan disusun dengan aturan tertentu serta untuk menjalankan tugas tertentu. Delphi menggunakan struktur bahasa pemrograman Object Pascal. Gabungan dari object dan

Jendela IDE Delphi 7.0 mempunyai perangkat-perangkat yang dapat dipergunakan untuk memudahkan seorang programmer dalam membuat program.

Perangkat-perangkat tersebut seperti terlihat pada gambar 2-10 diantaranya adalah:

1. *Main Window*

Jendela utama ini adalah bagian dari IDE yang mempunyai fungsi yang sama dengan semua fungsi utama dari program aplikasi Windows lainnya. Jendela utama Delphi terbagi menjadi tiga bagian, berupa Main Menu, Toolbar dan Component Palette.

2. *Main Menu*

Menu utama pada Delphi memiliki kegunaan yang sama seperti program aplikasi Windows lainnya. Dengan menggunakan fasilitas menu, dapat memanggil atau menyimpan program. Pada dasarnya semua perintah yang diberikan dapat ditemukan pada bagian menu utama ini.

3. *Toolbar*

Dengan *toolbar* dapat melakukan beberapa operasi pada menu utama dengan sebuah klik tunggal. Setiap tombol pada *toolbar* mempunyai sebuah *tooltip* yang berisi informasi mengenai fungsi dari tombol tersebut.

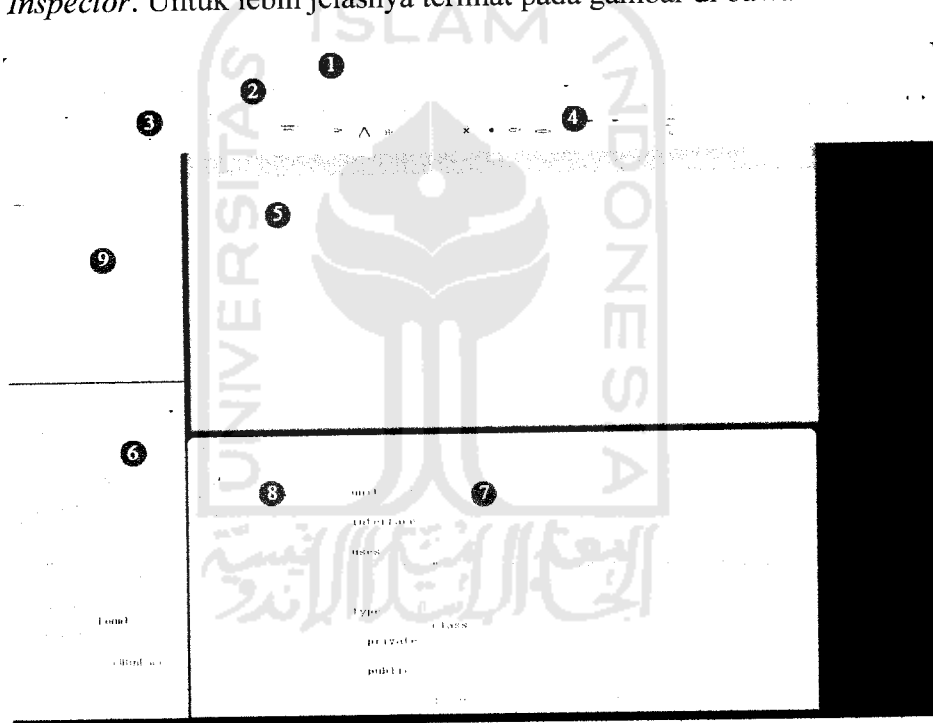
4. *Component Palette*

*Component Palette* merupakan bagian yang digunakan untuk meletakkan berbagai komponen yang sesuai dengan kategorinya. Pada bidang ini semua komponen yang merupakan bawaan dari Delphi, baik berupa komponen visual maupun komponen nonvisual. Komponen-komponen tersebut berguna untuk

bahasa pemrograman ini sering disebut sebagai bahasa pemrograman berorientasi object atau *Object Oriented Programming* ( OOP ).

### 2.6.1 IDE Delphi

Lingkungan pengembangan terpadu atau *Integrated Development Environment* ( IDE ) dalam program Delphi terbagi menjadi enam bagian utama, yaitu *Main Window*, *ToolBar*, *Componen Palette*, *Form Designer*, *Code Editor* dan *Object Inspector*. Untuk lebih jelasnya terlihat pada gambar di bawah ini:



**Gambar 2.12** Lembar Kerja Delphi

IDE merupakan sebuah lingkungan dimana semua tombol perintah yang diperlukan untuk mendesain aplikasi, menjalankan dan menguji sebuah aplikasi disajikan dengan baik untuk memudahkan pengembangan program.

mendesain user interface ( antarmuka pemakai ) dari program yang sedang dibuat. Borland Delphi sendiri memungkinkan untuk menambahkan komponen dari luar, baik yang dibuat sendiri ataupun komponen dari pihak ketiga ( third party ).

#### 5. *Form Designer*

Merupakan sebuah bidang jendela (*window*) yang masih kosong. Pada bidang ini dapat menempatkan komponen-komponen visual dan nonvisual untuk mendesain *user interface* program. Ketika menjalankan Borland Delphi 7, secara otomatis *Form Designer* akan memanggil sebuah form yang bernama Form1.

#### 6. *Object Inspector*

Dengan perangkat ini dapat mengubah *property* dan *event* pada setiap *object* atau komponen. Object atau komponen yang satu dengan yang lain mempunyai *property* dan *event* yang berbeda. Jika menempatkan sebuah komponen pada suatu Form, komponen tersebut akan berisi nilai *default* dari Delphi. Misalnya komponen TButton akan berisi nilai 25 dan 75 untuk *property* height dan width. Nilai-nilai *property* tersebut dapat diubah kemudian. Baik pada saat perancangan *interface* program ( *design time* ) maupun pada saat program berjalan ( *run time* ) dengan menggunakan kode program.

#### 7. *Code Editor*

Pada bidang ini dapat menuliskan kode-kode program dan logika program dalam bahasa Delphi untuk mengatur jalannya program. Antara Form Designer dan Code Editor merupakan dua bagian yang berkaitan, tidak bias hanya mendesain

user interface pada Form Designer dengan melupakan penulisan kode program pada bagian Code Editor.

#### 8. *Code Explorer*

Pada jendela *Code Explorer* ini akan ditampilkan sebuah *type*, *variabel*, dan *routine* yang didefinisikan pada *unit*. Selain itu juga ditampilkan semua *unit* yang digunakan terletak pada *klausula uses*. Untuk *type* yang kompleks seperti kelas, *code explorer* akan menampilkan semua informasi termasuk daftar *field*, *properties* dan *method*.

#### 9. *Object TreeView*

Merupakan sebuah diagram pohon yang menggambarkan hubungan logis antara komponen visual dan nonvisual yang terletak pada form, data module atau frame. Semua *object* yang anda pakai pada form, data module atau frame akan muncul pada *object TreeView*.

### 2.6.2 **Menu Borland Delphi**

#### 1. *Menu File*

Berisi perintah-perintah dasar menu yang sering digunakan yang berhubungan dengan pengoperasian file.

#### 2. *Menu Edit*

Berisi perintah-perintah yang digunakan untuk mentunting teks program dalam jendela code editor, menyunting komponen-komponen yang terletak pada bagian form designer dan beberapa item lainnya.



### 3. *Menu Search*

berisi perintah-perintah yang digunakan untuk menyunting teks program dalam jendela code editor, menyunting komponen-komponen yang terletak pada bagian form designer dan beberapa item lainnya.

### 4. *Menu View*

Berisi perintah yang digunakan untuk menampilkan atau menyembunyikan jendela-jendela tertentu dalam program Delphi.

### 5. *Menu Project*

Berisi perintah-perintah yang berhubungan dengan pengolahan suatu project.

### 6. *Menu Run*

Digunakan untuk menjalankan program dan melihat jalannya program. Dan juga dapat memantau jalannya program dengan memperhatikan procedure yang dijalankan.

### 7. *Menu Component*

Digunakan untuk menambah atau menginstall komponen-komponen baru.

### 8. *Menu Database*

digunakan untuk membuat, mengubah atau melihat database.

### 9. *Menu Tools dan Help*

Menu Tools digunakan untuk mengubah option atau memanggil Database Dekstop dari menu Delphi.

Menu Help digunakan untuk membuka lembar kerja bantu Delphi.



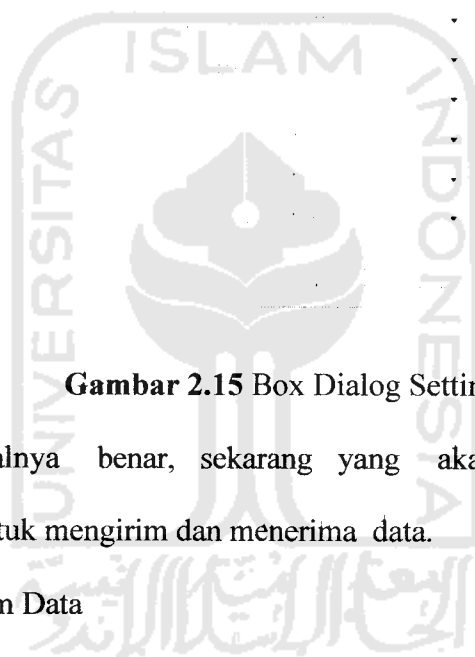
yang berada dibawah ini:

```

procedure TForm1.Button1Click(Sender: TObject);
begin
  Serial1.SerialPortName := 'COM1';
  Serial1.SerialPortSettings := '9600,8,N,1';
  Serial1.Open;
end;

```

Jika program diatas dieksekusi, maka akan tampil dialog box setting serial seperti pada gambar dibawah ini:



**Gambar 2.15** Box Dialog Setting Serial

Setelah setup serialnya benar, sekarang yang akan kita lakukan adalah membuat program untuk mengirim dan menerima data.

#### 1. Program Mengirim Data

Untuk membuat program mengirim data maka perlu dipersiapkan sebuah variabel bisa bertipe string atau integer. Untuk memulai proses pengiriman data maka com serial harus dibuka atau dikoneksikan terlebih dahulu dengan delphinya. Adapun program transmit data seperti yang terlihat dibawah

```

procedure TForm1.Button1Click(Sender: TObject);
begin
  Serial1.SerialPortName := 'COM1';
  Serial1.SerialPortSettings := '9600,8,N,1';
  Serial1.Open;
end;

```

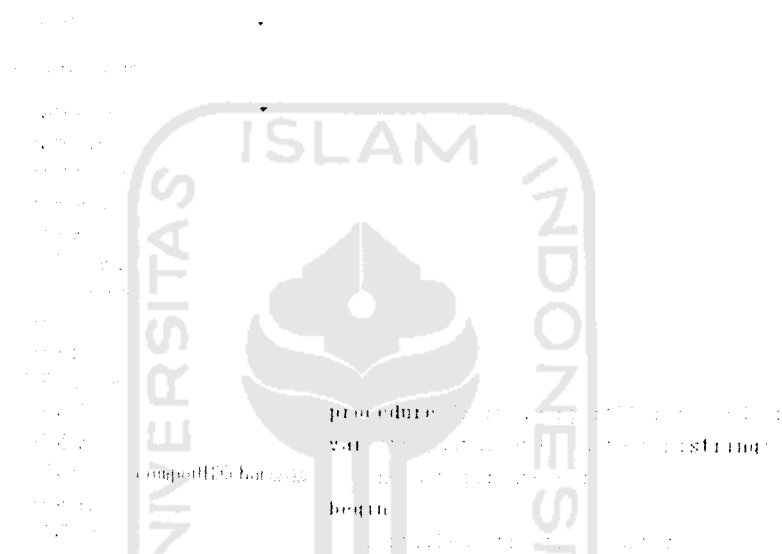
Maksud program :

comport1.Open : open koneksi delphi dengan com serial komputer

comport1.WriteStr(str) : transfer data string (data yang ditransfer bertipe string)

## 2. Program Menerima Data

Untuk membuat program menerima data dapat dilihat pada gambar dibawah :



**Gambar 2.16** Dialog dan Program Menerima Data

Maksud Program :

comport1.ReadStr(str5,count) menerima data dari luar dan dipindahkan ke variabel str5 yang bertipe string.