

BAB II

TINJAUAN PUSTAKA

2.1 Mikrokontroler AVR

Kata AVR dapat berarti berasal dari singkatan *Alf and Vegard RISC* sesuai dengan nama penggagas pertama. Saat ini ada yang menggunakan singkatan dari *Advanced Virtual RISC*.

Menurut Syahrul (2012:2) Mikrokontroler AVR yang menggunakan teknologi *Reduce Instruction Set Computer* (RISC) dan berarsitektur Harvard ini pertama kali dikembangkan pada tahun 1996 oleh dua orang mahasiswa *Norwegian Institute of Technology* yaitu Alf-Egil Bogen dan Vegard Wollan yang kemudian dikembangkan lebih lanjut oleh perusahaan Atmel.

Seri pertama mikrokontroler AVR yang dikeluarkan adalah mikrokontroler 8-bit dengan nama AT90S8515, dengan konfigurasi pin yang sama dengan mikrokontroler 8051, termasuk bus alamat dan bus data yang termultipleks.

Mikrokontroler AVR mempunyai set instruksi yang lebih sedikit dan mode pengalamatannya yang juga sederhana. Dalam AVR dengan arsitektur RISC 8-bit, semua instruksi berukuran 16-bit dan sebagian besar dieksekusi dalam 1 siklus *clock* kecuali instruksi pencabangan yang membutuhkan 2 siklus *clock*. Berbeda dengan mikrokontroler MCS-51 misalnya, yang instruksinya bervariasi antara 8-bit sampai 32-bit dan dieksekusi selama 1 sampai 4 siklus mesin, di mana 1 siklus mesin membutuhkan 12 periode *clock*.

Dalam perkembangannya, AVR dibagi menjadi sedikitnya enam kelas yaitu kelompok ATtiny, AT90Sxx, ATmega, AT90CAN, AT90PWM, dan AT86RFxx. Pada dasarnya yang membedakan masing-masing kelas adalah kapasitas memori, *peripheral*, fungsinya dan beberapa fitur tambahan lainnya. Sedangkan dari segi arsitektur dan set instruksi yang digunakan hampir sama.

2.1.1 Arsitektur AVR ATmega8535

Mikrokontroler ATmega8535 merupakan mikrokontroler 8-bit teknologi CMOS dengan konsumsi daya rendah yang berbasis arsitektur *enhanced RISC* AVR. Dengan eksekusi instruksi yang sebagian besar hanya menggunakan satu siklus *clock*, ATmega8535 mencapai *throughput* sekitar 1 MIPS per MHz yang mengizinkan perancang sistem melakukan optimasi konsumsi daya versus kecepatan pemrosesan.

Posesor AVR menggabungkan set instruksi yang kaya dengan 32 *register* umum (*general purpose registers, GPRs*). Ke semua 32 *register* tersebut dikoneksikan langsung dengan *Arithmetic Logic Unit (ALU)*, mengizinkan dua register independen untuk diakses dalam satu instruksi yang dieksekusi dalam satu siklus *clock*. Arsitektur yang dihasilkan adalah arsitektur yang kode operasinya lebih efisien serta pencapaian *throughput* nya hingga sepuluh kali lebih cepat daripada mikrokontroler *Complex Instruction Set Computer (CISC)* konvensional.

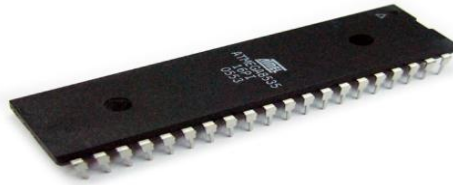
ATmega8535 menyediakan fitur-fitur: 8K byte memori *In-System Programmable Flash* dengan kemampuan *Read-While-Write*, 512 byte EEPROM, 512 byte SRAM, 32 saluran I/O untuk keperluan umum, 32 register GPR, tiga buah *flexible Timer/Counter* dengan *compare mode*, interupsi internal dan eksternal, *serial programmable USART*, satu byte diarahkan untuk *Two-wire Serial Interface*, 8-kanal ADC 10-bit dengan *optional differential input stage* dengan *programmable gain* untuk kemasan TQFP, sebuah *programmable Watchdog Timer* dengan *Internal Oscillator*, sebuah *SPI port*, dan enam *software selectable power saving modes*. *Idle mode* menghentikan CPU sementara mengizinkan SRAM, *Timer/Counter*, *SPI port*, dan sistem interupsi untuk kontinuitas operasi. *Power-down mode* menghemat isi-isi *register* tetapi *freezes the Oscillator*, melumpuhkan semua fungsi chip lainnya sehingga interupsi berikutnya atau Reset perangkat keras. Pada *Power-save mode*, timer asinkron tetap beroperasi, mengizinkan pemakai untuk tetap menjaga basis waktu sambil device lainnya sedang tidur. *ADC Noise Reduction mode* menghentikan CPU dan semua modul I/O kecuali *timer* asinkron dan ADC, untuk mengurangi *switching*

noise selama konversi ADC berlangsung. Pada *Standby mode*, osilator kristal/osilator resonator tetap berjalan sementara *device* lainnya sedang tidur. Hal ini mengizinkan *start-up* yang sangat cepat yang dikombinasikan dengan konsumsi daya rendah. Pada *Extended Standby mode*, osilator utama dan *timer* asinkron tetap berjalan.

Pada *On-chip ISP Flash* mengizinkan memori program untuk deprogram ulang dalam system melalui sebuah antarmuka SPI serial, dengan menggunakan *programmer* memori *nonvolatile* konvensional, atau dengan sebuah *On-chip Boot program* yang sedang beroperasi pada AVR *core*. *Boot program* dapat menggunakan antarmuka apapun untuk mengunduh program aplikasi kedalam *Application Flash memory*. Perangkat lunak pada *Boot Flash section* akan tetap beroperasi ketika *Application Flash Section* di-*'update'*, menyediakan operasi *Read-While-Write* yang sebenarnya. Dengan menggabungkan sebuah CPU 8-bit RISC dengan *In-System Self-Programable Flashc* pada sebuah *chip* monolithic, Atmel ATmega8535 merupakan mikrokontroler yang tangguh yang menyediakan fleksibilitas tinggi dan solusi biaya efektif untuk berbagai aplikasi control *embedded*.

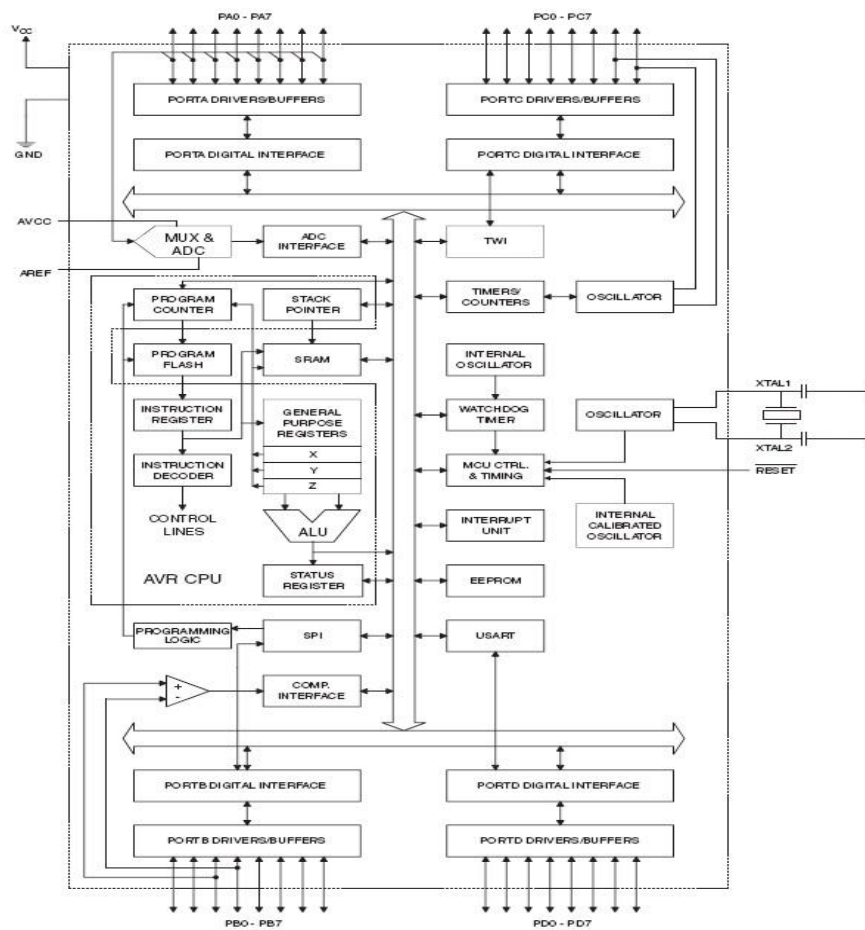
Beberapa fitur utama yang tersedia pada ATmega8535 adalah:

1. *Port* I/O 32 bit, yang dikelompokkan dalam: PortA, PortB, PortC, dan PortD
2. *Analog to Digital Converter* 10-bit sebanyak 8 input
3. *Timer/Counter* sebanyak 3 buah
4. CPU 8 bit yang terdiri dari 32 *register*
5. *Watchdog Timer* dengan osilator internal
6. SRAM sebesar 512 byte
7. Memori Flash sebesar 8 Kbyte dengan kemampuan *read while write*
8. Interrupt internal maupun eksternal
9. *Port* komunikasi SPI
10. EEPROM sebesar 512 byte yang dapat diprogram saat operasi
11. *Analog Comparator*
12. Komunikasi serial standar USART dengan kecepatan maksimal 2,5 Mbps
13. Frekuensi *clock* maksimum 16 MHz



Gambar 2.1 Mikrokontroler ATmega8535

(Sumber: <http://www.duniaelektronika.net/mikrokontroler-atmega8535-sistem-minimum/>)

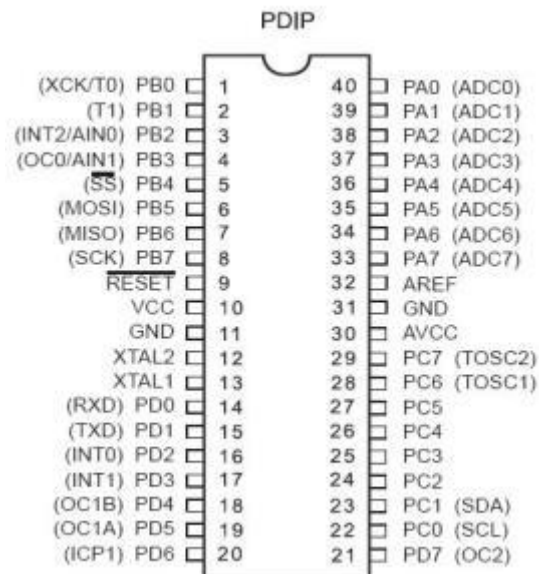


Gambar 2.2 Diagram Blok Arsitektur Mikrokontroler AVR ATmega8535

(Sumber: http://ayazblogz.blogspot.com/2011_04_01_archive.html)

2.1.2 Konfigurasi Pin ATmega8535

Konfigurasi pin mikrokontroler AVR ATmega8535 untuk 40 pin DIP (*dual in line package*) ditunjukkan pada gambar 2.3



Gambar 2.3 Konfigurasi Pin ATmega8535

(Sumber: <https://fahmizaleeits.wordpress.com/tag/fungsi-pin-atmega-8535/>)

Untuk dapat memahami lebih jauh tentang konfigurasi pin ATmega8535 maka pada Tabel 1.1 diberikan deskripsi kaki-kaki ataupun pin ATmega8535.

Tabel 2.1 Deskripsi pin AVR ATmega8535

No.Pin	Nama Pin	Keterangan
10	VCC	Catu daya
11	GND	Ground
40→33	PortA: PA0 – PA7 (ADC0 – ADC7)	Port I/O dua arah dilengkapi <i>internal pull-up resistor</i> . Port ini juga dimultipleks dengan masukan analog ke ADC 8 kanal
1→7	PortB: PB0 – PB7	Port I/O dua arah dilengkapi <i>internal pull-up resistor</i> . Fungsi lain dari port ini masing-masing adalah:

		<u>Port Pin</u> <u>Fungsi lain</u> PB0 T0 (timer/counter0 external counter input) PB1 T1 (timer/counter1 external counter input) PB2 AIN0 (analog comparator positive input) PB3 AIN1 (analog comparator positive input) PB4 SS (SPI slave select input) PB5 MOSI (SPI bus master output/slave input) PB6 MISO (SPI bus master input/slave output) PB7 SCK (SPI bus serial clock)
22→29	PortC: PC0 – PC7	Port I/O dua arah dilengkapi <i>internal pull-up resistor</i> . Dua pin yaitu PC6 dan PC7 berfungsi sebagai osilator eksternal untuk timer/counter2
14→21	PortD: PD0 – PD7	Port I/O dua arah dilengkapi <i>internal pull-up resistor</i> . Fungsi lain dari <i>port</i> ini masing-masing adalah: <u>Port Pin</u> <u>Fungsi lain</u> PD0 RXD (UART <i>input line</i>) PD1 TXD (UART <i>output line</i>) PD2 INT0 (<i>External interrupt 0 input</i>) PD3 INT1 (<i>External interrupt 1 input</i>) PD4 OC1B (<i>timer/counter1 output compareB match output</i>) PD5 OC1A (<i>timer/counter1 output compareA match output</i>) PD6 ICP (<i>timer/counter1 iutput capture pin</i>) PD7 OC2 (<i>timer/counter2 output compare match output</i>)
9	RESET	Masukan <i>reset</i> . Sebuah reset terjadi jika pin ini diberi logika <i>low</i> melebihi periode minimum yang diperlukan.
13	XTAL1	Masukan ke <i>inverting oscillator amplifier</i> dan masukan ke rangkaian <i>internal clock</i>
12	XTAL2	Keluaran dari <i>inverting oscillator amplifier</i>
30	AVCC	Catu daya untuk <i>port A</i> dan ADC
31	AGND	<i>Analog ground</i>
32	AREF	Referensi masukan analog untuk ADC

(sumber: Syahrul. MIKRIKONTROLER AVR ATmega8535. 2012)

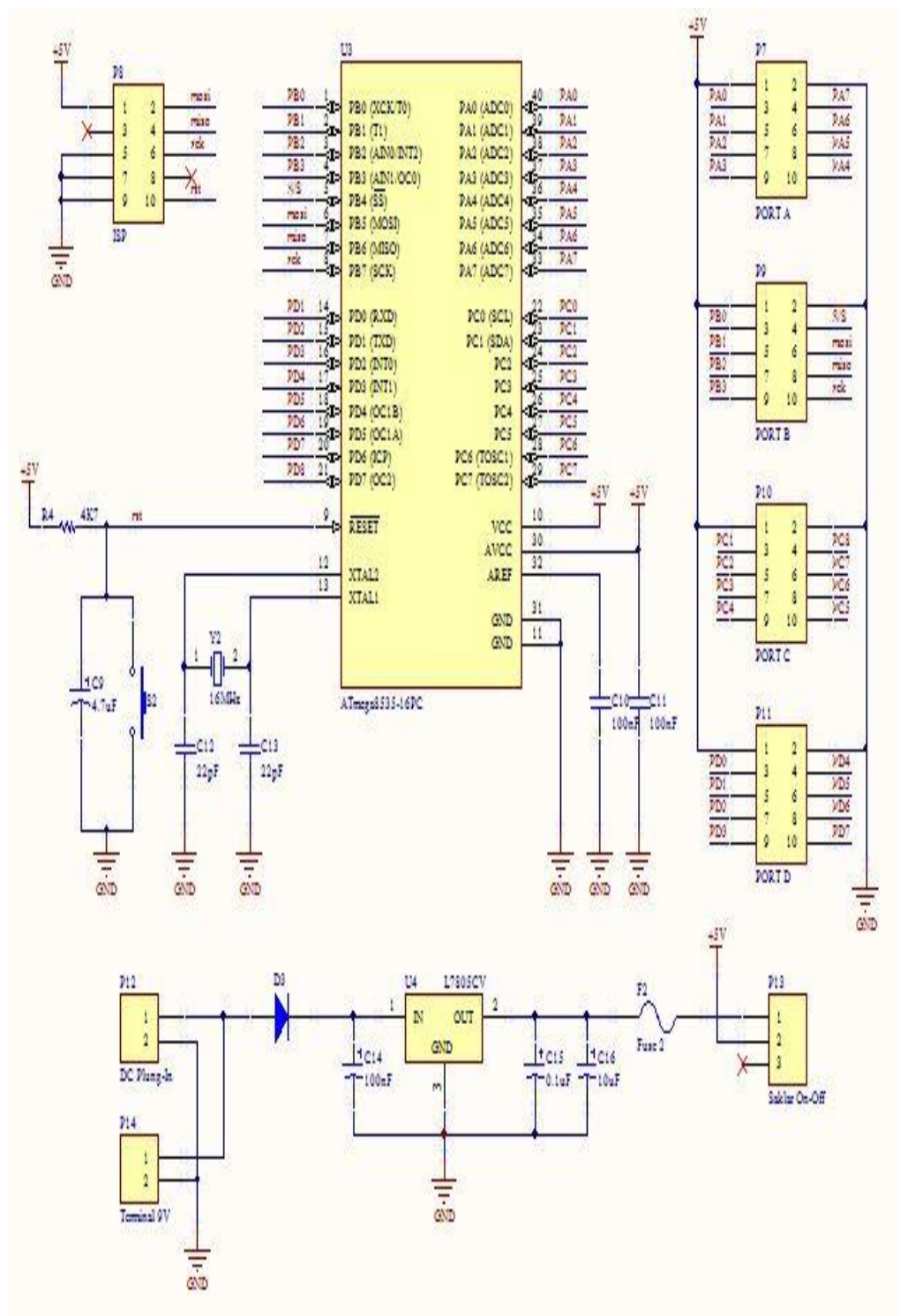
2.1.3 Sistem Minimum ATmega8535

Skematik sistem minimum mikrokontroler adalah rangkaian elektronik minimum yang diperlukan untuk dapat mengoperasikan IC mikrokontroler. Sistem minimum ini kemudian bisa dihubungkan dengan dengan rangkaian lain untuk menjalankan aplikasi tertentu. Mikrokontroler seri ATmega8535 keluarga mikrokontroler AVR, adalah salah satu seri yang banyak digunakan.

Untuk membuat skematik sistem minimum Atmel AVR ATmega8535 diperlukan beberapa komponen yaitu:

- a. IC mikrokontroler ATmega8535
- b. 1 buah XTAL 4 MHz, atau 8 MHz atau 12 MHz, maksimum 16 MHz.
- c. 2 kapasitor 10 pF (C3 dan C4)
- d. 2 kapasitor 100nF (C1 dan C2)
- e. 1 resistor 100 k (R1)
- f. 1 tombol reset *push button*

Skematik sistem minimum ini sudah siap untuk menerima sinyal analog (fasilitas ADC) pada *port A*. Skematiknya diberikan pada Gambar 2.4 berikut.



Gambar 2.4 Skematik sistem minimum mikrokontroler AVR ATmega8535

Sumber: <http://www.duniaelektronika.net/mikrokontroler-atmega8535-sistem-minimum/>

2.2 BASCOM AVR

2.2.1 Pengertian BASCOM AVR

Pemrograman menggunakan BASCOM-AVR adalah salah satu dari sekian banyak bahasa BASIC untuk pemrograman mikrokontroler, misalnya Bahasa Assembly, Bahasa C, dan lain-lain. Penulis menggunakan Bahasa BASIC BASCOM-AVR karena penggunaannya mudah dalam penulisannya, ringkas, cepat dimengerti bagi pemula, dan tidak kalah dengan bahasa BASIC lainnya.

Secara umum bahasa yang digunakan untuk pemrograman mikrokontroler adalah bahasa tingkat rendah yaitu bahasa assembly, dimana setiap mikrokontroler memiliki bahasa-bahasa pemrograman yang berbeda-beda. Karena banyaknya hambatan dalam penggunaan bahasa assembly ini maka mulai dikembangkan compiler/penerjemah untuk bahasa tingkat tinggi yang banyak dikembangkan adalah BASIC dan Bahasa C.

Pada perancangan alat ini, penulis menggunakan compiler BASCOM AVR (Basic Compiler AVR), dengan pertimbangan bahwa compiler ini cukup lengkap karena dilengkapi simulator untuk LCD, dan monitor untuk komunikasi serial. Selain itu, bahasa Basic jauh lebih mudah dipahami dibandingkan dengan bahasa pemrograman yang lainnya. Sebuah mikrokontroler dapat bekerja bila didalam mikrokontroler tersebut terdapat sebuah program yang berisikan instruksi-intruksi yang akan digunakan untuk menjalankan sistem mikrokontroler tersebut. Pada prinsipnya program pada mikrokontroler dijalankan secara bertahap. Maksudnya, pada program itu sendiri terdapat beberapa set instruksi yang mana tiap instruksi itu dijalankan secara bertahap atau berurutan.

Basic Compiler (BASCOM-AVR) merupakan software compiler (penerjemah untuk bahasa tingkat tinggi) dengan menggunakan bahasa basic yang dibuat untuk melakukan pemrograman chip-chip mikrokontroler tertentu, salah satunya ATmega8535.

Tabel 2.2 Intruksi Dasar BASCOM AVR

Intruksi	Keterangan
DO....LOOP	Perulangan
GOSUB	Memanggil prosedur
IF....THEN	Percabangan
FOR.....NEXT	Perulangan
WAIT	Waktu tanda detik
WAITMS	Waktu tanda mili detik
WAITUS	Waktu tanda micro detik
GOTO	Loncat ke alamat memori
SELECT....CASE	Percabangan

(Sumber: <http://dheni-yulistianto.blogspot.com/2013/07/pengertian-bascom-avr.html>)

2.2.2 Operasional

2.2.2.1 Tipe Data

Tipe data adalah jangkauan dari suatu variabel atau konstanta. Tipe data tersebut adalah:

Tabel 2.3 Tipe Data BASCOM AVR

Tipe Data	Kapasitas (Byte)	Jangkauan Nilai
Bit	1/8	0 dan 1
Byte	1	0 dan 255
Integer	2	-32.768 to 32.767
Word	2	0 to 65.535
Long	4	-2.147.483.648 to 2.147.483.647
Single	4	$1,5 \times 10^{-45}$ to $3,4 \times 10^{38}$
Double	8	5×10^{-324} to $1,7 \times 10^{308}$
String	254	-

(Sumber: Setiawan, Afrie. 2011. 20 Aplikasi Mikrokontroler ATmega8535 dan ATmega16 Menggunakan BASCOM-AVR)

2.2.2.2 Variabel

Variabel ditulis pada text program untuk menyimpan suatu pemrosesan data. Variabel dideklarasikan jika akan digunakan dengan mengacu pada aturan aturan sebagai berikut:

1. Dimulai dengan huruf,
2. Nama variabel tidak boleh lebih dari satu,
3. Tidak menggunakan spasi,
4. Maksimum 32 karakter, dan
5. Tidak menggunakan karakter khusus yang digunakan oleh program BASCOM AVR.

Variabel dideklarasikan dengan cara:

```
Dim <NamaVariabel> As <TipeData>
```

Contohnya:

```
Dim Putaran As Single
```

```
Dim Kecepatan As Integer
```

2.2.2.3 Konstanta

Konstanta adalah pendeklarasian suatu nama tetapi bernilai tetap. Konstanta dideklarasikan dengan cara:

```
Dim <NamaKonstanta> As Const <NilaiKonstanta>
```

Contohnya:

```
Dim Penjumlah As Const 14           'Penjumlah bernilai 14
```

2.2.2.4 Alias

Alias digunakan untuk mempermudah penulisan program.

Contoh:

```
Relay_1 Alias PORTA.1           'Nama dari PORTA.1 adalah Relay_1
```

2.2.2.5 Aritmatik dan Rasional

1. Data Aritmatik

Tabel 2.4 Data Aritmatik pada BASCOM AVR

Operasi	Keterangan
+	Penjumlahan
-	Pengurangan
/	Pembagian
*	Perkalian
%	Hasil sisa pembagian

(Sumber: Setiawan, Afrie. 2011. 20 Aplikasi Mikrokontroler ATMega8535 dan ATMega16 Menggunakan BASCOM-AVR)

2. Data Rasional

Tabel 2.5 Data Rasional pada BASCOM AVR

Operasi	Contoh	Keterangan
=	$X=Y$	Sama dengan
<>	$X<>Y$	Tidak sama dengan
>	$X>Y$	Lebih besar dari...
<	$X<Y$	Lebih kecil dari...
>=	$X>=Y$	Lebih besar atau sama dengan
<=	$X<=Y$	Lebih kecil atau sama dengan

(Sumber: Setiawan, Afrie. 2011. 20 Aplikasi Mikrokontroler ATMega8535 dan ATMega16 Menggunakan BASCOM-AVR)

2.2.2.6 Pengolahan bilangan Logika

Tabel 2.6 Pengolahan bilangan Logika

Operasi	Contoh	Keterangan
AND	&B10 AND &B01 = &B10	Operasi AND
OR	&B1000 OR &B0111 = &B1111	Operasi OR
NOT	NOT &B11 = &B00	Operasi NOT

(Sumber: Setiawan, Afrie. 2011. 20 Aplikasi Mikrokontroler ATmega8535 dan ATmega16 Menggunakan BASCOM-AVR)

2.2.3 Macam-macam Perintah

2.2.3.1 IF – THEN

Perintah IF-THEN digunakan untuk menguji suatu keadaan benar atau salah dan menentukan tindakan yang sesuai dengan keinginan.

Perintahnya:

If <Keadaan> Then <Perintah>	'1 baris perintah
End If	
If <Keadaan> Then	'lebih dari 1 baris perintah
<Perintah_1>	
<Perintah_2>	
<Perintah_n>	
End If	

Contohnya:	
If Saklar_1 = 1 Then Lampu_1 = 1	'1 baris perintah
End If	
If Saklar_1 = 1 Then	'lebih dari 1 baris perintah
Lampu_1 = 1	
Lampu_2 = 1	
Lampu_3 = 1	
End If	

2.2.3.2 IF-THEN-ELSE

Perintah IF-THEN-ELSE digunakan untuk menguji dua keadaan (benar ataupun salah) dan menentukan tindakan yang sesuai dengan keinginan.

Perintahnya:

```

If <Keadaan> Then
    <Perintah_1>
Else
    <Perintah_2>
End If

Contohnya:
    If Saklar_1 = 1 Then
        Lampu_1 = 1
    Else
        Lampu_1 = 0
    End If

```

2.2.3.3 IF-THEN-ELSEIF

Perintah IF-THEN-ELSEIF digunakan untuk menguji lebih dari satu keadaan dan menentukan tindakan yang sesuai dengan keinginan.

Perintahnya:

```

If <Keadaan_1> Then
    <Perintah_1>
Elseif <Keadaan_2> Then
    <Perintah_2>
Elseif <Keadaan_3> Then
    <Perintah_3>
    .....
    .....
End If

Contohnya:
    If Saklar_1 = 1 Then
        Lampu_1 = 1
    Elseif Saklar_2 = 1 Then
        Lampu_2 = 1
    Elseif Saklar_3 = 1 Then
        Lampu_3 = 1
    .....

```

2.2.3.4 SELECT-CASE

Perintah SELECT-CASE digunakan untuk pengujian keadaan yang banyak sehingga penulisan program menjadi lebih sederhana.

Perintahnya:

```

Select case <Nama_variabel>
  Case 1 : <Perintah_1>
  Case 2 : <Perintah_2>
  Case 3 : <Perintah_3>
  .....
  .....
End Select

```

Contohnya:

```

Select case temperatur
  Case 1 : kipas_berputar_200RPM
  Case 2 : kipas_berputar_800RPM
  Case 3 : kipas_berputar_1400RPM
  .....
  .....
End Select

```

2.2.3.5 DO-LOOP

Perintah DO-LOOP merupakan perintah untuk perulangan yang digunakan untuk melakukan perulangan program selama suatu kondisi telah terpenuhi.

Perintahnya:

```

Do
  <Pernyataan>
Loop
Contohnya:
  Do
    If Saklar_1 = 1 Then
      Lampu_1 = 1
    Elseif Saklar_2 = 1 Then
      Lampu_2 = 1
    Elseif Saklar_3 = 1 Then
      Lampu_3 = 1
    .....
    .....
  End If

```

2.2.3.6 FOR-NEXT

Perintah FOR-NEXT merupakan perintah untuk perulangan yang digunakan untuk melakukan perulangan program sesuai dengan jumlah dan tingkat perulangannya.

Perintahnya:

```

For <Variabel=Nilai_awal> To <Nilai_akhir> <Selisih_pertambahan>
    <Pernyataan>
Next
Contohnya:
    For temperatur=35 To 42 2
        Kipas_berputar_1400RPM
    Next

```

2.2.3.7 WHILE-WEND

Perintah WHILE-WEND merupakan perintah untuk perulangan yang akan melakukan perulangan apabila keadaan yang diminta telah terpenuhi.

Perintahnya:

```

While <Keadaan>
    <Perintah>
Wend
Contohnya:
    While temperature>35
        Kipas_berputar_1400RPM
    Wend

```

2.2.3.8 GOSUB

Perintah GOSUB merupakan perintah untuk lompatan yang akan melakukan lompatan ke label yang ditunjuk dan kembali ke tempat semula setelah melakukan perintah pembacaan program dengan menambahkan "Return".

Perintahnya:


```
Gosub <Nama_label>  
    <Pernyataan>  
<Nama_label>:  
    <Pernyataan>  
Return  
Contohnya:  
    Do  
        If Saklar_1 = 1 Then  
            Lampu_1 = 1  
        ElseIf Saklar_2 = 1 Then  
            Lampu_2 = 1  
        ElseIf Saklar_3 = 1 Then  
            Lampu_3 = 1  
        End If  
        Gosub Pensaklaran  
    Loop  
    Pensaklaran:  
        Lampu_1 = 1  
        Lampu_2 = 1  
        Lampu_3 = 1  
        Lampu_4 = 1  
        Lampu_5 = 1
```

2.2.3.9 GOTO

Perintah GOTO merupakan perintah untuk lompatan yang akan melakukan lompatan ke label yang ditunjuk tanpa kembali lagi ke tempat semula setelah melakukan perintah pembacaan program sehingga tidak menggunakan “Return”.

Perintahnya:

```

Goto <Nama_label>
    <Pernyataan>
<Nama_label>:
    <Pernyataan>
Contohnya:
    Do
        If Saklar_1 = 1 Then
            Lampu_1 = 1
        Elseif Saklar_2 = 1 Then
            Lampu_2 = 1
        Elseif Saklar_3 = 1 Then
            Lampu_3 = 1
        End If
    Goto Pensaklaran
Loop
Pensaklaran:
    Lampu_1 = 1
    Lampu_2 = 1
    Lampu_3 = 1
    Lampu_4 = 1
    Lampu_5 = 1

```

2.2.3.10 EXIT

Perintah EXIT merupakan perintah untuk mengakhiri perulangan DO-LOOP, FOR-NEXT, WHILE-WEND.

Perintahnya:

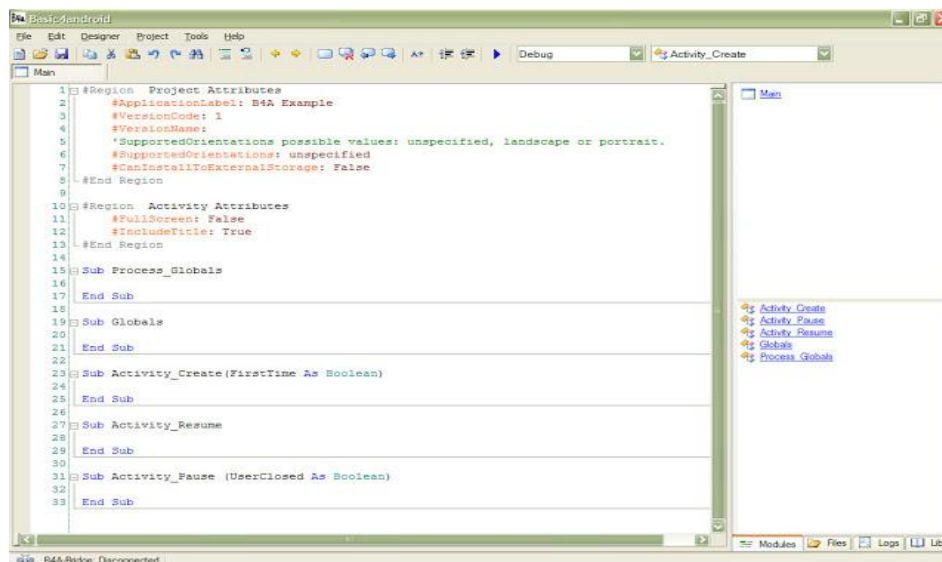
```

    <Pernyataan>
EXIT ...
    Do
        If Saklar_1 = 1 Then
            Lampu_1 = 1
        Elseif Saklar_2 = 1 Then
            Lampu_2 = 1
        End If
    Loop
EXIT Do      ‘untuk perintah “DO-LOOP”
~~~~~
    For temperature=35 To 42 2
        Kipas_berputar_1400RPM
    Next
EXIT For      ‘Untuk perintah “FOR-NEXT”
~~~~~
    While temperature>35
        Kipas_berputar_1400RPM
    Wend
EXIT While    ‘Untuk perintah “WHILE-WEND”

```

2.3 Basic4Android

Basic4android adalah development tool sederhana yang powerful untuk membangun aplikasi Android. Bahasa Basic4android mirip dengan Visual Basic dengan tambahan dukungan untuk objek. Aplikasi Android (APK) yang dicompile oleh Basic4Android adalah aplikasi Android asli dan tidak ada extra runtime seperti di Visual Basic yang ketergantungan file msvbvm60.dll, yang pasti aplikasi yang dicompile oleh Basic4Android adalah NO DEPENDENCIES (tidak ketergantungan file lain). IDE Basic4Android hanya fokus pada development Android.



Gambar 2.5 IDE Basic4Android

(Sumber: [wordpress.https://menyoystoreonline.wordpress.com/tutorial/cara-membuat-aplikasi-android-sendiri/](https://menyoystoreonline.wordpress.com/tutorial/cara-membuat-aplikasi-android-sendiri/))

Basic4android memiliki kekayaan dalam satuan libraries (perpustakaan) yang membuatnya menjadi lebih mudah untuk mengembangkan macam-macam aplikasi Android yang advanced. Library-nya adalah:

SQL databases

GPS

Serial ports (Bluetooth)

Camera

XML parsing
 Web services (HTTP)
 Services (background tasks)
 JSON
 Animations
 Network (TCP and UDP)
 Text To Speech (TTS)
 Voice Recognition
 WebView
 AdMob (ads)
 Charts
 OpenGL
 Graphics
 dan masih banyak lagi

Basic4android termasuk designer GUI untuk aplikasi Android yang powerful dengan dukungan Built-in untuk multiple screens dan orientations, serta tidak dibutuhkan lagi penulisan XML yang rumit.

2.4 LCD (*Liquid Crystal Display*)

Menurut (Andrianto, 2013:77) *Liquid Crystal Display* (LCD) adalah suatu *display* dari bahan cairan *crystal* yang pengoperasiannya menggunakan sistem *dot* matriks. LCD banyak digunakan sebagai *display* dari alat-alat elektronika seperti kalkulator, *Multitester Digital*, jam *digital* dan sebagainya.

LCD dapat dengan mudah dihubungkan dengan mikrokontroler AVR ATmega. LCD yang digunakan dalam proyek ini adalah LCD M1632 yang merupakan modul dengan tampilan 16 x 2, lebar *display* 2 baris 16 kolom, yang memiliki 16 *pin* konektor. LCD berfungsi menampilkan suatu hasil sensor, menampilkan teks, atau menampilkan menu pada aplikasi mikrokontroler.

Menurut (Setiawan, 2010:24) LCD merupakan salah satu perangkat penampil yang sekarang ini mulai banyak digunakan. Penampil LCD mulai dirasakan menggantikan fungsi dari penampil CRT (*Cathode Ray Tube*, yang

sudah berpuluh-puluh tahun digunakan manusia sebagai penampil gambar/teks baik monokrom (hitam dan putih) maupun yang berwarna. Teknologi LCD memberikan keuntungan lebih dibandingkan dengan teknologi CRT, yaitu konsumsi daya relatif kecil, lebih ringan, dan memiliki tampilan yang bagus.



Gambar 2.6 LCD Character 16 x 2 Module

(sumber: elib.unikom.ac.id/download.php?id=9233)

LCD JHD162A memiliki 16 pin (lihat gambar 2.6), yang memiliki deskripsi seperti pada tabel 2.7 dibawah ini :

Tabel 2.7 Deskripsi Pin LCD

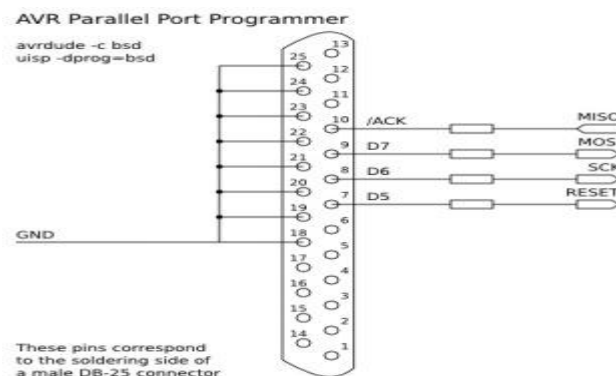
PIN	Name	Function
1	VSS	Ground
2	VCC	+5V
3	VEE	Tegangan Kontras
4	RS	Register Select 0 = Register instruksi 1 = Register Data
5	R/W	<i>Read/ Write</i> , untuk memilih mode tulis atau baca 0 = mode tulis 1 = mode baca
6	E	<i>Enable</i> 0 = <i>enable</i> (mulai menahan data ke LCD) 1 = <i>disable</i>
7	DB0	Data bit 0, LSB
8	DB1	Data bit 1

9	DB2	Data bit 2
10	DB3	Data bit 3
11	DB4	Data bit 4
12	DB5	Data bit 5
13	DB6	Data bit 6
14	DB7	Data bit 7, MSB
15	BPL	<i>Back Plane Light</i>
16	GND	<i>Ground</i>

(sumber: Syahrul. MIKRIKONTROLER AVR ATmega8535. 2012)

2.5 Kabel Downloader

Keunggulan yang dimiliki oleh mikrokontroler AVR dibandingkan dengan mikrokontroler MCS pada cara memasukkan program (upload program) dari komputer ke chip mikrokontroler terletak pada kemudahannya. Mikrokontroler MCS membutuhkan perangkat tambahan berupa downloader yang terdiri dari IC tambahan untuk proses koneksi antara mikrokontroler dengan komputer, sedangkan mikrokontroler AVR dapat langsung diprogram dari port paralel komputer menuju pin mikrokontroler AVR. Untuk konfigurasi pin-pin tersebut adalah: (Afrie Setiawan. 2011. Hal: 10)



Gambar 2.7 Konfigurasi Kabel Downloader mikrokontroler AVR-ATMega8535

(Sumber: <https://fahmizaleeits.wordpress.com/tag/downloader-mikrokontroler/>)