

## 2. DASAR TEORI

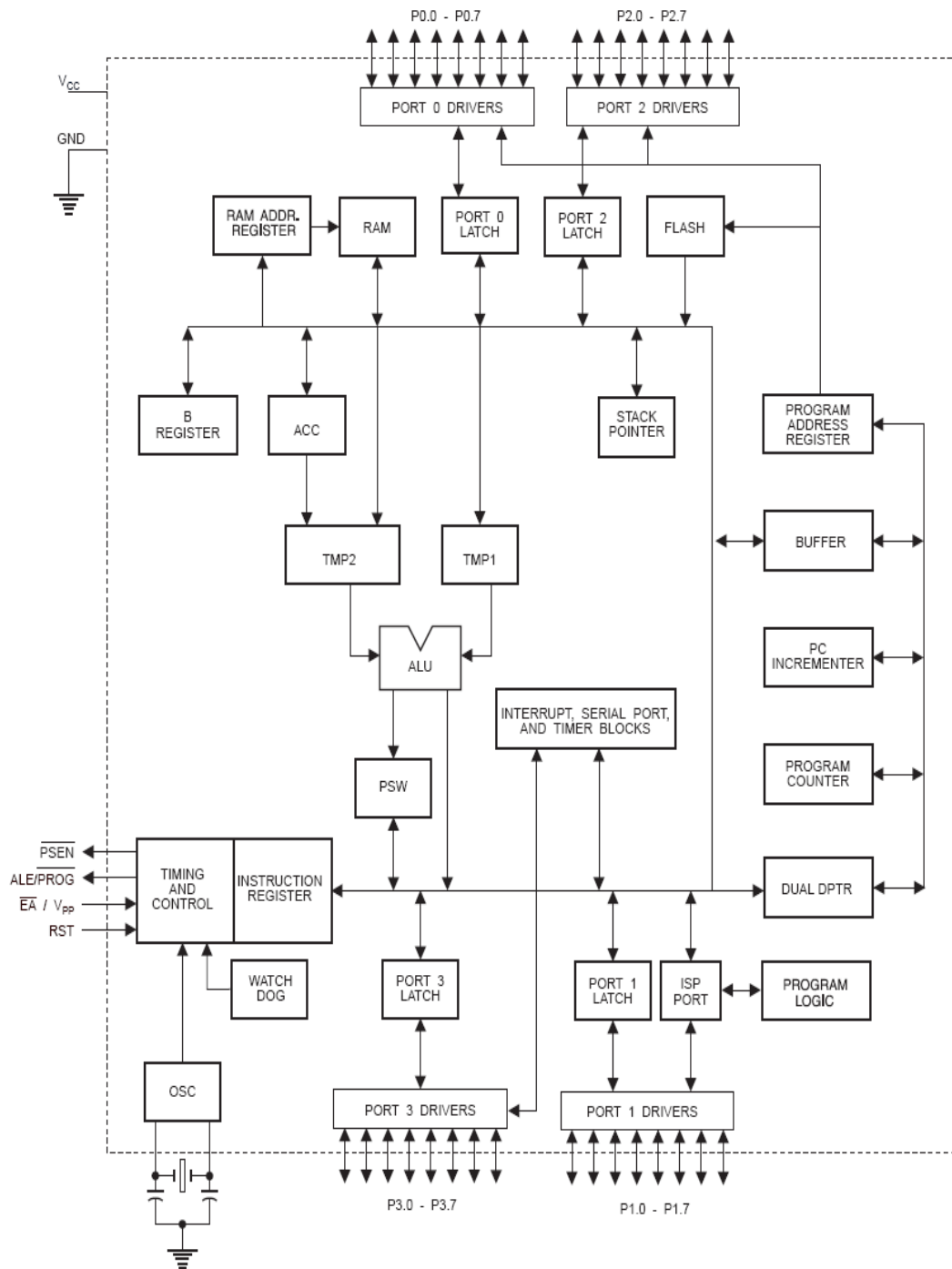
### 2.1. AT89S51

AT89S51 dari Atmel dipercayakan sebagai mikrokontroler dalam robot amfibi ini. IC mikrokontroler yang dipakai pada tugas akhir ini yaitu seri AT89 dari ATMEL yang merupakan varian dari keluarga MCS-51. Pembahasan selanjutnya akan mengacu pada tipe 89S51, yang diharapkan dapat menjadi dasar pengetahuan untuk mempelajari tipe-tipe mikrokontroler lainnya.

#### 2.1.1. Arsitektur AT89S51

AT89S51 adalah mikrokontroler CMOS *8-bit* yang hemat energi dan berkemampuan tinggi dengan *4K bytes ISP flash memory* yang dapat diprogram. Perangkat ini diproduksi menggunakan teknologi *high-density nonvolatile memory* milik ATMEL dan kompatibel dengan standar industri untuk instruksi 89S51 dan *pinout*-nya. *Flash memory* dalam *chip* memungkinkan program ditulis ke dalam *chip* dengan cara ISP ataupun dengan cara konvensional.

Dengan menggabungkan *8-bit* CPU yang serbaguna dan *ISP Flash* yang dapat diprogram ke dalam *chip* yang sama, AT89S51 menjadi mikrokontroler hebat yang memberikan solusi yang sangat fleksibel dan murah bagi banyak aplikasi kontrol. AT89S51 menyediakan banyak fitur, antara lain: *Flash memory* sebesar *4K bytes*, RAM sebesar *128 bytes*, *32* jalur I/O, *Watchdog Timer*, dua *data pointers*, dua *16-bit timer/counters*, arsitektur dua tingkat *five-vectors interrupt*, *full duplex serial port*, osilator dalam *chip* dan *clock circuitry*.

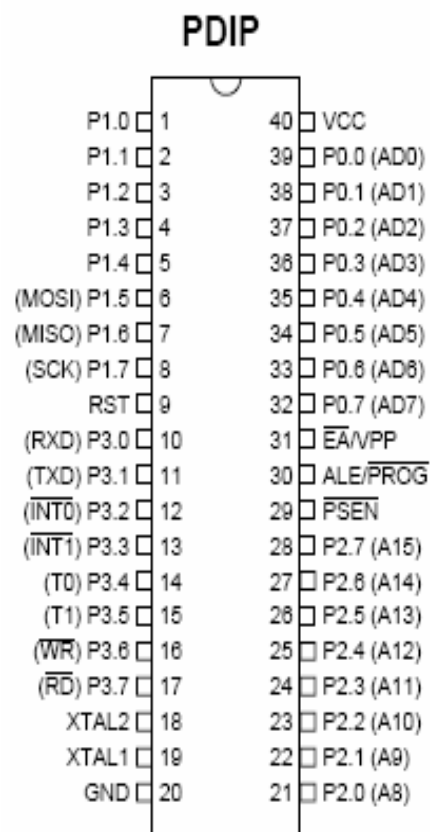


Gambar 2.1. Blok Diagram AT89S51

Sumber: Atmel Corporation. *AT89S51 Datasheet*. 26 Januari 2005. p. 3.  
 <[http://www.atmel.com/dyn/resources/prod\\_documents/doc2487.pdf](http://www.atmel.com/dyn/resources/prod_documents/doc2487.pdf)>

### 2.1.2. Konfigurasi *Pin* AT89S51

*Pin* adalah kaki fisik dari sebuah AT89S51. IC ini memiliki beberapa *pin*, ada yang berfungsi sebagai jalur *input/output* (I/O), ada yang berfungsi sebagai jalur kontrol dan ada juga yang berfungsi sebagai *address bus* atau *data bus*. Jadi secara keseluruhan mikrokontroler AT89S51 mempunyai 40 kaki, 32 kaki diantaranya adalah kaki untuk keperluan *port* paralel. Satu *port* paralel terdiri dari 8 kaki, dengan demikian 32 kaki tersebut membentuk 4 buah *port* paralel, yang masing-masing dikenal sebagai *port* 0, *port* 1, *port* 2, *port* 3. Berikut gambar konfigurasi *pin* pada AT89S51.



Gambar 2.2. Konfigurasi *Pin* AT89S51

Sumber: Atmel Corporation. *AT89S51 Datasheet*. 26 Januari 2005. p. 2.  
<[http://www.atmel.com/dyn/resources/prod\\_documents/doc2487.pdf](http://www.atmel.com/dyn/resources/prod_documents/doc2487.pdf)>

#### 2.1.2.1.VCC

Suplai tegangan. *Pin* ini dihubungkan dengan sumber tegangan antara +4,0 Volt hingga +5,5 Volt.

#### 2.1.2.2. GND

*Ground. Pin* ini dihubungkan dengan *ground* pada rangkaian.

#### 2.1.2.3. Port 0

*Port 0* merupakan salah satu *port* yang berfungsi sebagai *general purpose I/O* dengan lebar 8 *bit* bertipe *open drain bidirectional*. *Port 0* terdiri dari P0.0 hingga P0.7. Sebagai *port* keluaran, masing-masing kaki dapat menyerap arus (*sink*) delapan masukan TTL (sekitar 3,8 mA). Pada saat '1' dituliskan ke kaki-kaki *port 0* ini, maka kaki-kaki *port 0* dapat digunakan sebagai masukan-masukan berimpedansi tinggi. *Port 0* juga dapat dikonfigurasi sebagai *bus* alamat/data bagian rendah (*low byte*) selama proses pengaksesan memori data dan program eksternal. Jika digunakan dalam mode ini *port 0* memiliki *pull-up* internal.

*Port 0* juga menerima kode-kode yang dikirimkan kepadanya selama proses pemrograman dan mengeluarkan kode-kode selama proses verifikasi program yang telah tersimpan dalam *flash*. Dalam hal ini dibutuhkan *pull-up* eksternal selama proses verifikasi program.

#### 2.1.2.4. Port 1

*Port 1* merupakan salah satu *port* yang berfungsi sebagai *general purpose I/O* dengan lebar 8 *bit*. *Port 1* terdiri dari P1.0 hingga P1.7. *Port 1* merupakan *port I/O* dua arah yang dilengkapi dengan *pull-up* internal. Penyangga keluaran *port 1* mampu memberikan/menyerap arus empat masukan TTL (sekitar 1,6 mA). Jika '1' dituliskan ke kaki-kaki *port 1*, maka masing-masing kaki akan di-*pulled high* dengan *pull-up* internal sehingga dapat digunakan sebagai masukan. Sebagai masukan, jika kaki-kaki *port 1* dihubungkan ke *ground* (di-*pulled low*), maka masing-masing kaki akan memberikan arus (*source*) karena di-*pulled high* secara internal. *Port 1* juga menerima alamat bagian rendah (*low byte*) selama pemrograman dan verifikasi *flash*. *Port 1* mempunyai fungsi alternatif untuk digunakan pemrograman secara langsung dalam sistemnya. Berikut tabel fungsi alternatif dari *port 1*.

Tabel 2.1. Fungsi Alternatif *Port 1*

Port Pin	Alternate Functions
P1.5	MOSI (used for In-System Programming)
P1.6	MISO (used for In-System Programming)
P1.7	SCK (used for In-System Programming)

Sumber: Atmel Corporation. *AT89S51 Datasheet*. 26 Januari 2005. p. 4.  
 <[http://www.atmel.com/dyn/resources/prod\\_documents/doc2487.pdf](http://www.atmel.com/dyn/resources/prod_documents/doc2487.pdf)>

#### 2.1.2.5. *Port 2*

*Port 2* merupakan salah satu *port* yang berfungsi sebagai *general purpose* I/O dengan lebar 8 *bit*. *Port 2* terdiri dari P2.0 hingga P2.7. Selain sebagai jalur I/O, *port 2* juga berfungsi sebagai *high byte address*.

*Port 2* merupakan *port* I/O dua arah yang dilengkapi dengan *pull-up* internal. Penyangga keluaran *port 2* mampu memberikan/menyerap arus empat masukan TTL (sekitar 1,6 mA). Jika '1' dituliskan ke kaki-kaki *port 2*, maka masing-masing kaki akan di-*pulled high* dengan *pull-up* internal sehingga dapat digunakan sebagai masukan. Sebagai masukan, jika kaki-kaki *port 2* dihubungkan ke *ground* (di-*pulled low*), maka masing-masing kaki akan memberikan arus (*source*) karena di-*pulled high* secara internal.

*Port 2* akan memberikan *byte* alamat bagian tinggi (*high byte*) selama pengambilan instruksi dari memori program eksternal dan selama pengaksesan memori data eksternal yang menggunakan perintah dengan alamat 16-*bit* (misalnya: MOVX @DPTR). Dalam aplikasi ini, jika ingin mengirimkan '1', maka digunakan *pull-up* internal yang sudah disediakan. Selama pengaksesan memori data eksternal yang menggunakan perintah dengan alamat 8-*bit* (misalnya: MOVX @Ri), *port 2* akan mengirimkan isi dari SFR P2. *Port 2* juga menerima alamat bagian tinggi selama pemrograman dan verifikasi *flash*.

#### 2.1.2.6. *Port 3*

*Port 3* merupakan salah satu *port* yang berfungsi sebagai *general purpose* I/O dengan lebar 8 *bit*. *Port 3* terdiri dari P3.0 hingga P3.7. Fungsi lain dari *port 3* terdapat pada tabel 2.2.

*Port 3* merupakan *port I/O* dua arah yang dilengkapi dengan *pull-up* internal. Penyangga keluaran *port 3* mampu memberikan/menyerap arus empat masukan TTL (sekitar 1,6 mA). Jika '1' dituliskan ke kaki-kaki *port 3*, maka masing-masing kaki akan di-*pulled high* dengan *pull-up* internal sehingga dapat digunakan sebagai masukan. Sebagai masukan, jika kaki-kaki *port 3* dihubungkan ke *ground* (di-*pulled low*), maka masing-masing kaki akan memberikan arus (*source*) karena di-*pulled high* secara internal.

Tabel 2.2. Fungsi Alternatif *Port 3*

Port Pin	Alternate Functions
P3.0	RXD (serial input port)
P3.1	TXD (serial output port)
P3.2	$\overline{\text{INT0}}$ (external interrupt 0)
P3.3	$\overline{\text{INT1}}$ (external interrupt 1)
P3.4	T0 (timer 0 external input)
P3.5	T1 (timer 1 external input)
P3.6	$\overline{\text{WR}}$ (external data memory write strobe)
P3.7	$\overline{\text{RD}}$ (external data memory read strobe)

Sumber: Atmel Corporation. *AT89S51 Datasheet*. 26 Januari 2005. p. 5.  
<[http://www.atmel.com/dyn/resources/prod\\_documents/doc2487.pdf](http://www.atmel.com/dyn/resources/prod_documents/doc2487.pdf)>

*Port 3* sebagaimana *port 1*, memiliki fungsi-fungsi alternatif, antara lain sinyal-sinyal kontrol (P3.6 dan P3.7), bersama-sama dengan *port 2* (P2.6 dan P2.7) selama pemrograman dan verifikasi *flash*.

#### 2.1.2.7. RST (*Reset*)

*Pin* ini berfungsi sebagai *input* untuk melakukan *reset* terhadap AT89S51. Jika RST bernilai *high* selama minimal 2 *machine cycle*, AT89S51 akan di-*reset* dan nilai *internal register* akan dikembalikan seperti keadaan awal pada saat 89S51 baru mulai bekerja. Kaki ini akan berkondisi '1' pada 98 periode osilator setelah waktu *watchdog* habis. *Bit* DISRTO dalam SFR AUXR (alamat 8E<sub>H</sub>) dapat digunakan untuk mematikan fitur ini. Pada kondisi *default* dari *bit* DISRTO, fitur *RESET HIGH* diaktifkan.

#### 2.1.2.8. ALE/PROG (*Address Latch Enable/Program*)

*Address Latch Enable* (ALE) berfungsi sebagai *demultiplexer* pada saat *port 0* bekerja sebagai *multiplexed address/data bus*. Pada paruh pertama *memory cycle*, *pin* ALE mengeluarkan sinyal *latch* yang menahan alamat ke *external register*. Pada paruh kedua *memory cycle*, *port 0* akan digunakan sebagai *data bus*. *Pin* ini juga berfungsi sebagai masukan pulsa program atau PROG selama pemrograman *flash*.

Frekuensi pulsa ALE adalah sebesar  $1/6$  dari frekuensi *on-chip oscillator* dan dapat digunakan untuk *external clocking/timing*. Fungsi lain dari *pin* ini adalah sebagai pulsa *input* pada saat pemrograman *internal program memory*. Sebagai catatan, ada satu pulsa yang dilompati selama pengaksesan memori data eksternal.

#### 2.1.2.9. EA/VPP

*External Access* (EA) merupakan *pin* yang berfungsi sebagai *input* kontrol. Jika EA bernilai *low* (dihubungkan ke *ground*), maka program hanya akan dijalankan dari *external program memory* (0000<sub>H</sub> hingga FFFF<sub>H</sub>). Jika EA bernilai *high* (dihubungkan ke +5 V), maka program akan dijalankan dari *internal program memory* terlebih dahulu. Fungsi lain dari *pin* ini adalah sebagai jalur tegangan *input* +12 V pada saat pemrograman *flash*, khususnya untuk tipe mikrokontroler 12 volt Vpp (yaitu dengan kode IC AT89S51 xxxx, sedangkan yang berkode AT89S51 xxxx-5, menggunakan tegangan 5 volt Vpp).

### 2.1.3. *Timer dan Counter*

AT89S51 ini juga mempunyai 2 buah fitur *Timer/Counter*. Fungsi *Timer/Counter* dipilih dengan mengontrol *bit-bit* C/T dalam *Special Function Register* (TMOD). Masing-masing *Timer/Counter* memiliki empat mode dan dapat dikombinasikan.

#### 2.1.3.1. Prinsip *Timer dan Counter*

Pada dasarnya *Timer dan Counter* merupakan sistem yang sama-sama menambahkan dirinya sendiri hingga *overflow*. *Timer* memanfaatkan frekuensi

osilator untuk bertambah tiap *machine cycle*. *Counter* memanfaatkan sumber dari luar AT89S51. *Counter* akan bertambah, jika terdapat transisi dari nilai '1' ke nilai '0' pada *pin* T0 (P3.4) atau T1 (P3.5). *Timer* dapat digunakan untuk menghitung suatu periode waktu antara kejadian, sebagai jarak waktu antara kejadian, dan menghasilkan *baud rate* untuk komunikasi serial. *Counter* dapat digunakan untuk menghitung jumlah munculnya suatu kejadian.

### 2.1.3.2. *Timer Register*

*Register* yang digunakan untuk mengatur *timer/counter* terdapat pada *Timer Mode* (TMOD) dan *Timer Control* (TCON).

MSB				LSB			
GATE	C/T	M1	M0	GATE	C/T	M1	M0
<i>Timer/Counter 1</i>				<i>Timer/Counter 0</i>			

Gambar 2.3. Alokasi *Bit* TMOD

Sumber: Atmel Corporation. *8051 Microcontrollers Hardware Manual*.  
26 Januari 2005. p. 2-45.  
<[http://www.atmel.com/dyn/resources/prod\\_documents/doc4316.pdf](http://www.atmel.com/dyn/resources/prod_documents/doc4316.pdf)>

Tabel 2.3. *Timer Mode*

Simbol	Deskripsi
GATE	Pemilih <i>external</i> atau <i>internal control</i>
C/T	Pemilih <i>Timer</i> atau <i>Counter</i>
M1	Pemilih Mode <i>Timer/Counter</i>
M0	Pemilih Mode <i>Timer/Counter</i>

Sumber: Atmel Corporation. *8051 Microcontrollers Hardware Manual*.  
26 Januari 2005. p. 2-45.  
<[http://www.atmel.com/dyn/resources/prod\\_documents/doc4316.pdf](http://www.atmel.com/dyn/resources/prod_documents/doc4316.pdf)>

Berikut ini adalah penjelasan masing-masing *bit* TMOD:

- GATE

Jika GATE dan TRx (TR0 atau TR1 pada TCON) diberi nilai '1' oleh *user*, maka *Timer/Counter* hanya beroperasi jika *pin* INTx bernilai *high*.

Jika GATE bernilai '0', maka *Timer/Counter* hanya beroperasi jika TRx bernilai '1'.

- C/T  
C/T harus diberi nilai '1' oleh program *user* untuk menjalankan mode *counter* dan diberi nilai '0' untuk menjalankan mode *timer*.
- M1 & M0  
M1 dan M0 merupakan dua *bit* pemilih mode operasi *timer/counter*.

Tabel 2.4. Mode Operasi *Timer/Counter*

M1	M0	Mode Operasi	
0	0	0	<i>Timer/Counter, 13 bit</i>
0	1	1	<i>Timer/Counter, 16 bit</i>
1	0	2	<i>Timer/Counter, 8 bit Auto Reload</i>
1	1	3	<i>Split Timer Mode untuk Timer/Counter 0</i>
1	1	3	<i>Timer/Counter 1 berhenti</i>

Sumber: Atmel Corporation. *8051 Microcontrollers Hardware Manual*.  
26 Januari 2005. p. 2-45.  
<[http://www.atmel.com/dyn/resources/prod\\_documents/doc4316.pdf](http://www.atmel.com/dyn/resources/prod_documents/doc4316.pdf)>

MSB							LSB
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0

Gambar 2.4. Alokasi *Bit* TCON

Sumber: Atmel Corporation. *8051 Microcontrollers Hardware Manual*.  
26 Januari 2005. p. 2-47.  
<[http://www.atmel.com/dyn/resources/prod\\_documents/doc4316.pdf](http://www.atmel.com/dyn/resources/prod_documents/doc4316.pdf)>

Tabel 2.5. *Timer Control*

<i>Bit</i>	<i>Alamat Bit</i>	<i>Simbol</i>	<i>Deskripsi</i>
TCON.7	8F <sub>H</sub>	TF1	<i>Timer/Counter 1 Overflow Flag</i>
TCON.6	8E <sub>H</sub>	TR1	<i>Timer 1 Run Control Bit</i>
TCON.5	8D <sub>H</sub>	TF0	<i>Timer/Counter 0 Overflow Flag</i>
TCON.4	9C <sub>H</sub>	TR0	<i>Timer 0 Run Control Bit</i>
TCON.3	8B <sub>H</sub>	IE1	<i>External Interrupt 1 Edge Flag</i>
TCON.2	8A <sub>H</sub>	IT1	<i>External Interrupt 1 Type Control Bit</i>
TCON.1	89 <sub>H</sub>	IE0	<i>External Interrupt 0 Edge Flag</i>
TCON.0	88 <sub>H</sub>	IT0	<i>External Interrupt 0 Type Control Bit</i>

Sumber: Atmel Corporation. *8051 Microcontrollers Hardware Manual*.  
26 Januari 2005. p. 2-47.  
<[http://www.atmel.com/dyn/resources/prod\\_documents/doc4316.pdf](http://www.atmel.com/dyn/resources/prod_documents/doc4316.pdf)>

Berikut ini adalah penjelasan masing-masing *bit* TCON yang berkaitan dengan *timer/counter*:

- TF1  
TF1 diberi nilai '1' secara *hardware* saat nilai *Timer/Counter* 1 (TH1 dan TL1) mengalami *overflow*. TF1 akan diberi nilai '0' secara *hardware* saat mikrokontroler melompat ke *Interrupt Service Routine*.
- TR1  
TR1 harus diberi nilai '1' oleh program *user* untuk menjalankan *Timer* 1.
- TF0  
TF0 diberi nilai '1' secara *hardware* saat nilai *Timer/Counter* 0 (TH0 dan TL0) mengalami *overflow*. TF0 akan diberi nilai '0' secara *hardware* saat mikrokontroler melompat ke *Interrupt Service Routine*.
- TR0  
TR0 harus diberi nilai '1' oleh program *user* untuk menjalankan *Timer* 0.

#### 2.1.3.3. Mode Operasi

Seperti yang sudah dibahas sebelumnya, *Timer/Counter* memiliki 4 mode operasi. Setiap mode memiliki karakteristik tersendiri. Pemakai harus benar-benar memahami masing-masing mode agar dapat memilih mode yang tepat dalam program yang dibuat.

##### a. MODE 0

Mode 0 adalah *Timer/Counter* 13 *bit*. Pada mode 0, *register* TLx (TL0 atau TL1) hanya digunakan 5 *bit* terendah saja sedangkan *register* THx (TH0 atau TH1) tetap selebar 8 *bit*.

TLx akan bertambah hingga bernilai 1F<sub>H</sub>. Saat ada perubahan nilai TLx dari 1F<sub>H</sub> ke 00<sub>H</sub>, THx akan bertambah 1. Nilai maksimal THx dan TLx adalah FF1F<sub>H</sub> (THx = FF<sub>H</sub> dan TLx = 1F<sub>H</sub>). *Overflow* akan terjadi jika ada perubahan dari FF1F<sub>H</sub> ke 0000<sub>H</sub>. Nilai THx dan TLx dapat diubah oleh *user* setiap saat dalam program.

### b. MODE 1

Mode 1 pada dasarnya serupa dengan mode 0. Namun pada mode 1 semua *bit* TLx digunakan, sehingga mode 1 merupakan *Timer/Counter 16 bit*.

TLx akan bertambah hingga bernilai FF<sub>H</sub>. Saat ada perubahan nilai TLx dari FF<sub>H</sub> ke 00<sub>H</sub>, THx akan bertambah 1. Nilai maksimal THx dan TLx adalah FFFF<sub>H</sub> (THx = FF<sub>H</sub> dan TLx = FF<sub>H</sub>). *Overflow* akan terjadi jika ada perubahan dari FFFF<sub>H</sub> ke 0000<sub>H</sub>. Nilai THx dan TLx dapat diubah oleh *user* setiap saat dalam program.

### c. MODE 2

Mode 2 adalah *Timer/Counter 8 bit* dengan fasilitas *auto reload*. TLx bertindak sebagai *Timer/Counter 8 bit*. Sedangkan THx berisi suatu nilai tertentu. *Auto reload* adalah fasilitas dimana nilai TLx setelah *overflow* tidak kembali ke 00<sub>H</sub> namun nilai TLx akan diambil dari nilai THx.

Misalkan THx berisi 47<sub>H</sub> dan TLx berisi FF<sub>H</sub>. Jika ada *overflow* pada TLx, nilai TLx akan berubah dari FF<sub>H</sub> ke 47<sub>H</sub>, sesuai dengan nilai THx. Proses tersebut dikerjakan secara otomatis secara *hardware* sehingga *user* tidak perlu memeriksa apakah nilai TLx sudah *overflow* dan tidak perlu mengisi ulang dengan nilai yang baru. Mode 2 ini juga digunakan untuk menghasilkan *baud rate*.

### d. MODE 3

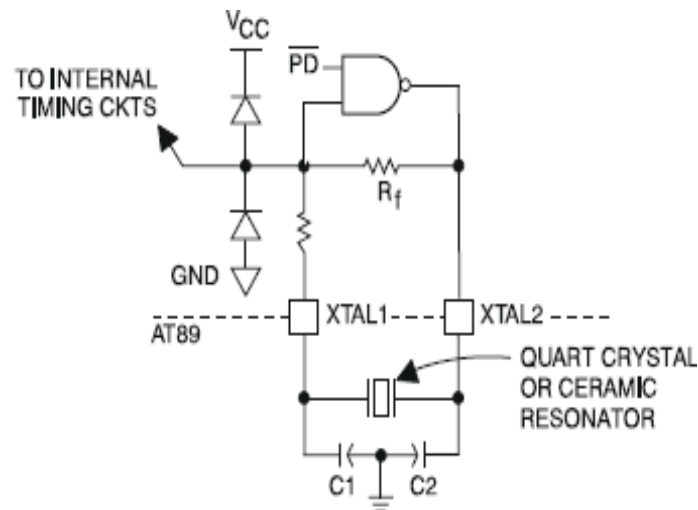
Pada mode 3, *Timer/Counter 0* akan menjadi dua *Timer/Counter 8 bit*, sedangkan *Timer/Counter 1* akan berhenti. TL0 akan menjadi *Timer/Counter 8 bit* yang dikendalikan oleh *bit* kontrol *Timer/Counter 0* (meliputi GATE, C/T, TR0, INT0 dan TF0). TH0 akan menjadi *timer 8 bit* (bukan *counter*) yang dikendalikan oleh *bit* kontrol *Timer/Counter 1* (meliputi TR1 dan TF1). Jadi TR1 mengendalikan TH0 dan *Timer/Counter 1* secara bersamaan.

#### 2.1.4. Clock

IC mikrokontroler AT89S51 memiliki dua *pin* (XTAL1 dan XTAL2) yang berhubungan dengan sistem *clock*. Sumber *clock* dapat berasal dari *on-chip oscillator* atau dari *external clock generator*.

a. *On-chip oscillator*

*On-chip oscillator* digunakan dengan dua kapasitor dan sebuah *quartz crystal* atau *ceramic resonator*. Rangkaiannya terdapat pada gambar 2.5.

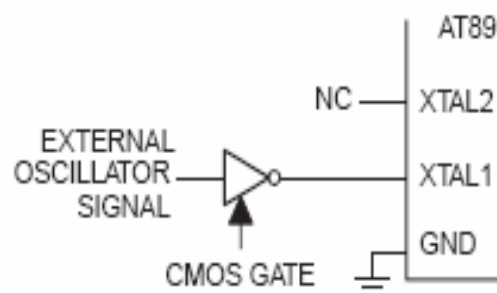


Gambar 2.5. Rangkaian XTAL dengan *On-Chip Oscillator*

Sumber: Atmel Corporation. *AT89S51 Datasheet*. 26 Januari 2005. p. 29.  
<[http://www.atmel.com/dyn/resources/prod\\_documents/doc2487.pdf](http://www.atmel.com/dyn/resources/prod_documents/doc2487.pdf)>

b. *External clock*

Selain dengan *on-chip oscillator*, IC mikrokontroler AT89S51 juga dapat menggunakan *external clock generator*. Sumber *clock* luar ini dihubungkan dengan XTAL1, sedangkan XTAL2 tidak terhubung kemanapun juga. Untuk lebih jelasnya dapat dilihat pada gambar 2.6.



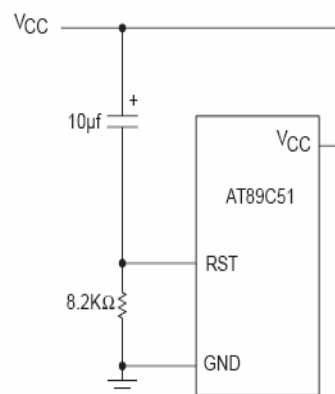
Gambar 2.6. Penggunaan *External Clock*

Sumber: Atmel Corporation. *AT89S51 Datasheet*. 26 Januari 2005. p. 31.  
<[http://www.atmel.com/dyn/resources/prod\\_documents/doc2487.pdf](http://www.atmel.com/dyn/resources/prod_documents/doc2487.pdf)>

### 2.1.5. Reset

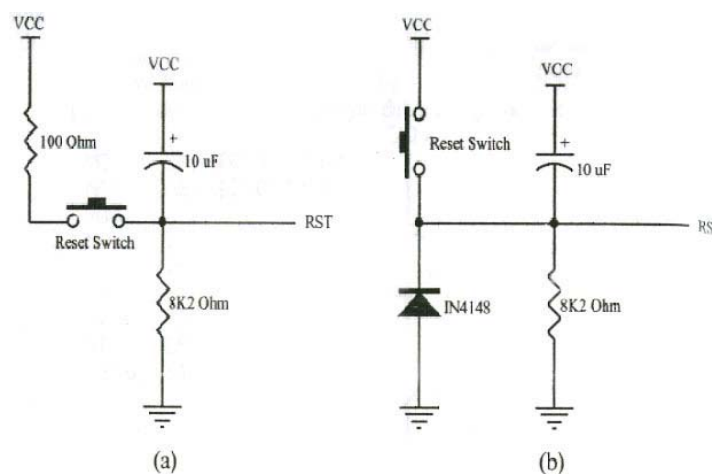
Proses *reset* merupakan proses untuk mengembalikan sistem ke kondisi semula. *Reset* tidak mempengaruhi *internal program memory*. *Reset* terjadi jika *pin* RST bernilai *high* selama minimal 2 *machine cycle* lalu kembali bernilai *low*. Ada dua macam proses *reset* yaitu *Power-on reset* dan *Manual reset*.

*Power-on reset* merupakan proses *reset* yang berlangsung secara otomatis pada saat sistem pertama kali diberi daya. Proses ini mempengaruhi semua *register* dan *internal data memory*. Untuk mendapatkan proses ini, maka *pin* RST harus diberi tambahan rangkaian seperti pada gambar 2.7.



Gambar 2.7. Rangkaian *Power-On Reset*

Jika diinginkan, *pin* RST juga dapat diberi rangkaian *manual reset*. Beberapa rangkaian yang umum digunakan terdapat pada gambar 2.8. Pemberian rangkaian ini membuat sistem dapat di-*reset* oleh *user* setiap saat dengan menekan tombol *reset*.



Gambar 2.8. Rangkaian *Manual Reset*

### 2.1.6. Kumpulan Instruksi

89S51 memiliki instruksi dengan jumlah yang cukup banyak. Instruksi-instruksi tersebut dapat dimasukkan ke dalam beberapa golongan.

#### 2.1.6.1. Boolean Instruction

*Boolean Instruction* mencakup instruksi-instruksi yang hanya melibatkan 1 bit saja. Instruksi ini menggunakan *Carry flag* atau *register 1 bit* lainnya sebagai *operand*.

Tabel 2.6. Daftar Instruksi *Boolean*

Instruksi		Penjelasan Singkat
CLR	C	Memberi nilai '0' pada <i>bit</i>
CLR	<i>Bit</i>	
SETB	C	Memberi nilai '1' pada <i>bit</i>
SETB	<i>Bit</i>	
CPL	C	Komplemen <i>bit</i>
CPL	<i>Bit</i>	
ANL	C, <i>bit</i>	Operasi logika AND antar <i>bit</i>
ANL	C, <i>bit</i>	
ORL	C, <i>bit</i>	Operasi logika OR antar <i>bit</i>
ORL	C, <i>bit</i>	
MOV	C, <i>bit</i>	Mengisi nilai dari <i>bit</i> ke <i>bit</i>
MOV	<i>Bit</i> , C	
JC	Rel	Lompat ke alamat tertentu jika <i>Carry flag</i> = '1'
JNC	Rel	Lompat ke alamat tertentu jika <i>Carry flag</i> = '0'
JB	<i>Bit</i> , rel	Lompat ke alamat tertentu jika <i>bit</i> = '1'
JNB	<i>Bit</i> , rel	Lompat ke alamat tertentu jika <i>bit</i> = '0'
JBC	<i>Bit</i> , rel	Lompat ke alamat tertentu jika <i>bit</i> = '1' lalu mengisi <i>bit</i> dengan nilai '0'

### 2.1.6.2. Branch Instruction

*Branch Instruction* mencakup instruksi-instruksi yang melakukan proses perpindahan alamat. Instruksi-instruksi tersebut antara lain: pemanggilan rutin (*call*) dan lompat (*jump*).

Tabel 2.7. Daftar Instruksi Percabangan

Instruksi	Penjelasan Singkat
ACALL Addr11	Memanggil sub rutin pada alamat tertentu
LCALL Addr16	
RET	Keluar dari subrutin
RETI	Keluar dari <i>interrupt</i>
AJMP Addr11	Lompat ke alamat tertentu
LJMP Addr18	
SJMP Rel	
JMP @A+DPTR	
JZ Rel	Lompat ke alamat tertentu jika <i>Accumulator</i> = 00 <sub>H</sub>
JNZ Rel	Lompat ke alamat tertentu jika <i>Accumulator</i> tidak bernilai 00 <sub>H</sub>
CJNE A, direct, rel	Membandingkan kedua <i>operand</i> dan melompat ke alamat tertentu jika kedua <i>operand</i> tidak sama
CJNE A, #data, rel	
CNJE @Ri, #data, rel	
DJNZ Rn, rel	Mengurangi <i>operand</i> dan melompat ke alamat tertentu jika <i>operand</i> tidak bernilai 00 <sub>H</sub>
DJNZ Direct, rel	
NOP	Tidak ada operasi

### 2.1.6.3. Data Transfer Instruction

*Data Transfer Instruction* mencakup instruksi-instruksi yang melakukan proses pemindahan atau pertukaran data yang melibatkan *register* 8 bit atau 16 bit. Instruksi-instruksi tersebut antara lain: MOV, PUSH, POP, dan XCH.

Tabel 2.8. Daftar Instruksi Pemindahan Data

Instruksi	Penjelasan Singkat
MOV <i>A, source</i>	Mengisi nilai <i>operand</i> kedua ( <i>source</i> ) ke dalam <i>operand</i> pertama ( <i>destination</i> )
MOV <i>A, #data</i>	
MOV <i>Dest, A</i>	
MOV <i>Dest, source</i>	Mengisi nilai <i>operand</i> kedua ( <i>source</i> ) ke dalam <i>operand</i> pertama ( <i>destination</i> )
MOV <i>Dest, #data</i>	
MOV <i>DPTR, #data16</i>	
MOVC <i>A, @A+DPTR</i>	Mengisi nilai dari <i>program memory</i> ke dalam <i>Accumulator</i>
MOVC <i>A, @A+PC</i>	
MOVX <i>A, @Ri</i>	Mengisi nilai dari <i>external data memory</i>
MOVX <i>A, @DPTR</i>	
MOVX <i>@Ri, A</i>	
MOVX <i>@DPTR, A</i>	
PUSH <i>Direct</i>	Mengisi nilai ke dalam <i>stack</i>
POP <i>Direct</i>	Mengambil nilai dari <i>stack</i>
XCH <i>A, source</i>	Menukar nilai kedua <i>operand</i>
XVHD <i>A, @Ri</i>	Menukar 4 <i>bit</i> terendah dari kedua <i>operand</i>

## 2.2. Receiver/Decoder RXD-433-KH

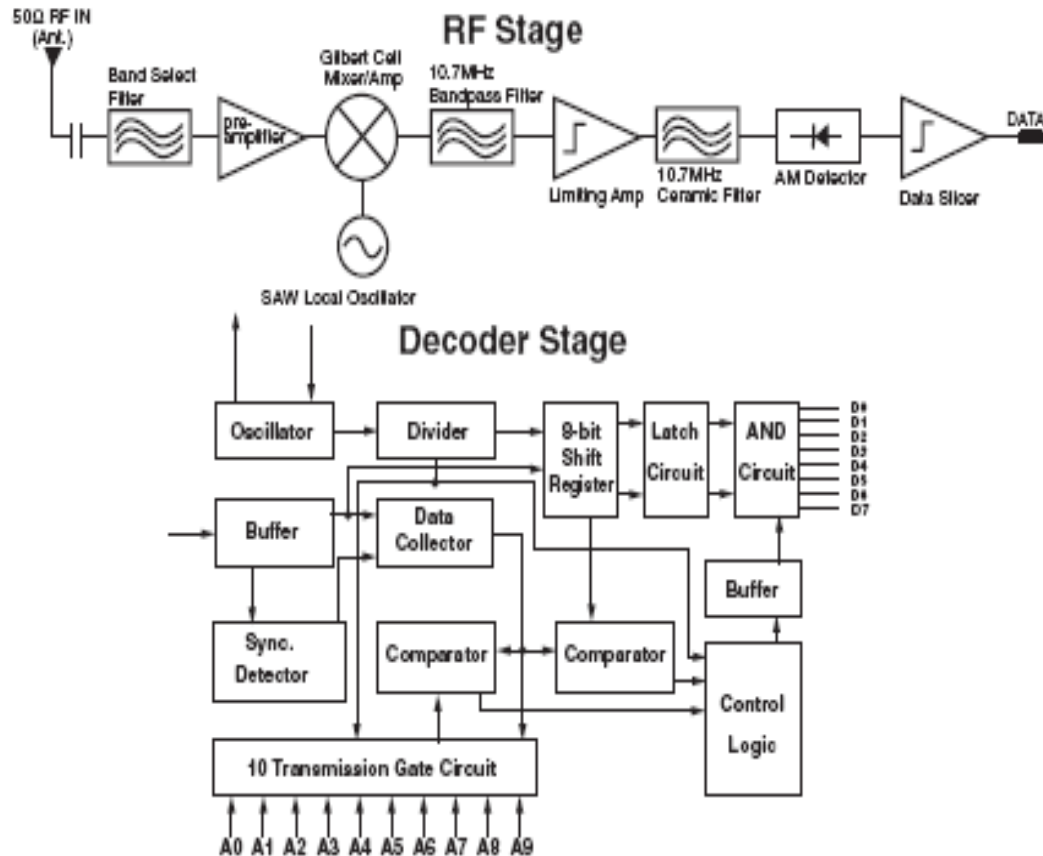
Seri KH ini ideal untuk digunakan dalam aplikasi OEM seperti *remote control/command* dan *keyless entry*. Ini dikombinasikan dengan penerima RF (*Radio Frequency*) dengan *on board decoder*.

Ketika dipasangkan sesuai dengan modul seri KH *transmitter/encoder* sangat memungkinkan untuk digunakan dalam membentuk hubungan *wireless* dan memungkinkan untuk mentransfer status dari 8 input paralel selama pada jarak 90 meter untuk merek yang sama. Tipe ini juga dilengkapi dengan 10 *tri-state address* yang menyediakan  $3^{10}$  kemungkinan *address* yang berbeda untuk alasan keamanan dan keunikan.

### 2.2.1. Penjelasan Modul

Modul ini mengkombinasikan *receiver* Linx seri LC dengan IC *decoder* dalam bentuk SMD *package*. Modul ini ideal digunakan untuk *remote control* dan aplikasi-aplikasi *command*. Kemampuan jarak jangkauannya antara *transmitter* dan *receiver* adalah sejauh 90 meter, dengan catatan modul ini juga dipasangkan

dengan modul *transmitter* dari Linx juga. Modul ini juga disediakan  $3^{10}$  atau 59,049 kombinasi *address* yang berbeda-beda. Berikut adalah blok diagram dari modul ini.

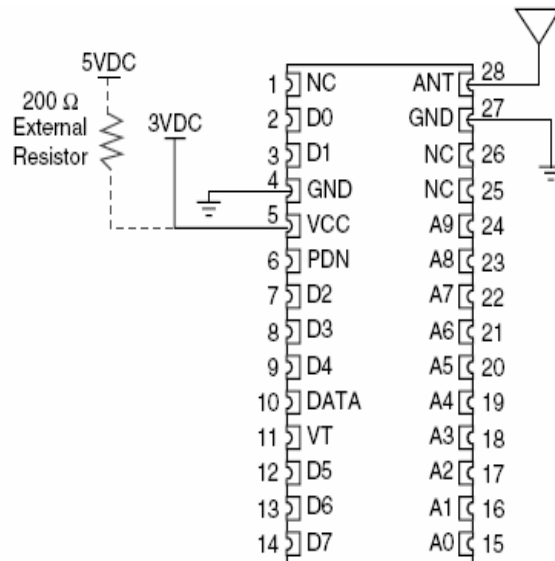


Gambar 2.9. Blok Diagram *Receiver* Seri KH

Sumber: Linx Technologies. *RXD-433-KH*. 10 Januari 2006. p. 5.  
[http://www.linxtechnologies.com/documents/kh\\_rxd\\_manual.pdf](http://www.linxtechnologies.com/documents/kh_rxd_manual.pdf)

### 2.2.2. Aplikasi

Modul ini mampu beroperasi pada suhu 25°C pada 3 V<sub>DC</sub> atau pada 5 V<sub>DC</sub> dengan menambahkan resistor 200 ohm.



Gambar 2.10. Aplikasi Sirkuit *Receiver*

Sumber: Linx Technologies. *RXD-433-KH*. 10 Januari 2006. p. 3.  
 <[http://www.linxtechnologies.com/documents/kh\\_rxd\\_manual.pdf](http://www.linxtechnologies.com/documents/kh_rxd_manual.pdf)>

### 2.2.3. Deskripsi *Pin*

1	NC	ANT	28
2	D0	GND	27
3	D1	NC	26
4	GND	NC	25
5	VCC	A9	24
6	PDN	A8	23
7	D2	A7	22
8	D3	A6	21
9	D4	A5	20
10	DATA	A4	19
11	VT	A3	18
12	D5	A2	17
13	D6	A1	16
14	D7	A0	15

Gambar 2.11. *Pinout Receiver* Seri KH

Sumber: Linx Technologies. *RXD-433-KH*. 10 Januari 2006. p. 4.  
 <[http://www.linxtechnologies.com/documents/kh\\_rxd\\_manual.pdf](http://www.linxtechnologies.com/documents/kh_rxd_manual.pdf)>

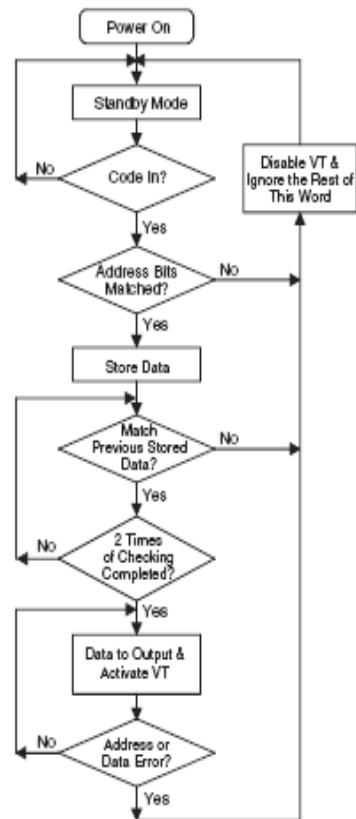
Tabel 2.9. Deskripsi *Pin Transmitter* Seri KH

<i>Pin #</i>	Name	Penjelasan
1	N/C	<i>Pin</i> ini tidak memiliki koneksi, hanya untuk mendukung fisik modul.
2,3,7,8, 9,12,13, 14	D0-D7	<i>Data Input Lines</i> . Pada kondisi <i>valid transmission pin-pin</i> ini akan menyesuaikan alamat pada <i>transmitter</i> jika terjadi kecocokan maka <i>pin-pin</i> ini akan mengeluarkan data yang sama seperti dikirimkan oleh <i>transmitter</i> .
4	GND	<i>Analog Ground</i> .
5	VCC	<i>Supply Voltage</i> .
6	PDN	<i>Power Down</i> . Menurunkan <i>pin</i> ini pada kondisi <i>low</i> akan membuat <i>receiver</i> bekerja pada kondisi arus kecil, dengan demikian modul tidak dapat menerima sinyal pada kondisi ini.
10	DATA	<i>Output data dari encoder</i>
11	VT	<i>Valid Transmission</i> . <i>Pin</i> ini akan menjadi <i>high</i> ketika transmisi yang <i>valid</i> telah diterima.
15-24	A0-A9	<i>Address Lines</i> .
25	N/C	<i>Pin</i> ini tidak memiliki koneksi, hanya untuk mendukung fisik modul.
26	N/C	<i>Pin</i> ini tidak memiliki koneksi, hanya untuk mendukung fisik modul.
27	GND	<i>Analog Ground</i> .
28	RF IN	50-ohm RF <i>input</i> .

#### 2.2.4. Teori Pengoperasian

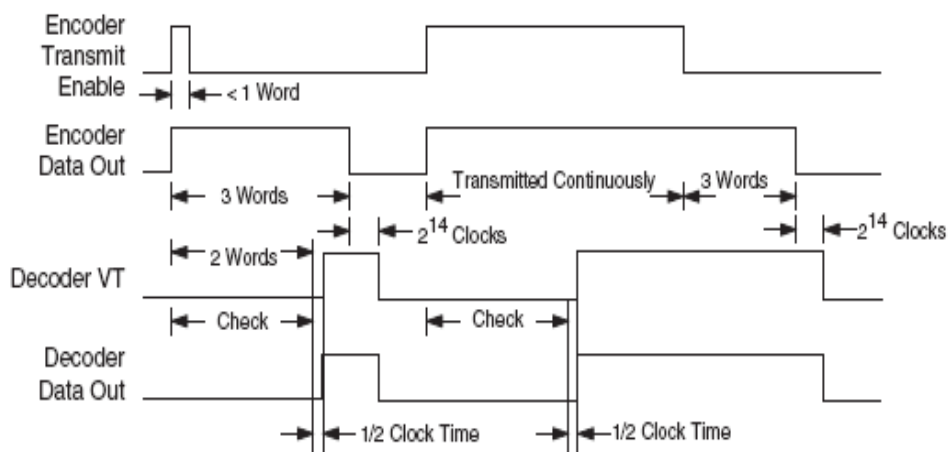
*Decoder* menerima data yang ditransmisikan oleh *encoder* dan menafsirkan 10 *bit-bit* pertama adalah sebagai kombinasi alamat dan 8 *bit-bit* terakhir adalah sebagai data yang akan diambil oleh modul ini. *Decoder* akan terlebih dahulu memeriksa kebenaran alamat yang diterima sebanyak 2 kali. Jika alamat yang diterima cocok dengan alamat yang dimiliki oleh *decoder*, maka 8 *bit-bit* data akan direplikasikan pada *pin-pin output*. Dan *pin* VT akan menjadi *high* untuk mengindikasikan bahwa data telah *valid* diterima. Proses ini akan berlangsung sampai pengkodean alamat berubah, atau tidak ada sinyal lagi yang diterima. *Pin* VT akan di-*set high* hanya ketika proses transmisi berlangsung *valid*. Jika tidak demikian, maka *pin* VT akan selalu berada pada kondisi *low*.

Berikut ini adalah *flowchart* dan *timing diagram* dari proses yang terjadi pada *decoder*.



Gambar 2.12. *Flowchart Decoder*

Sumber: Linx Technologies. *RXD-433-KH*. 10 Januari 2006. p. 6.  
 <[http://www.linxtechnologies.com/documents/kh\\_rxd\\_manual.pdf](http://www.linxtechnologies.com/documents/kh_rxd_manual.pdf)>



Gambar 2.13. *Timing Diagram Decoder*

Sumber: Linx Technologies. *RXD-433-KH*. 10 Januari 2006. p. 5.  
 <[http://www.linxtechnologies.com/documents/kh\\_rxd\\_manual.pdf](http://www.linxtechnologies.com/documents/kh_rxd_manual.pdf)>

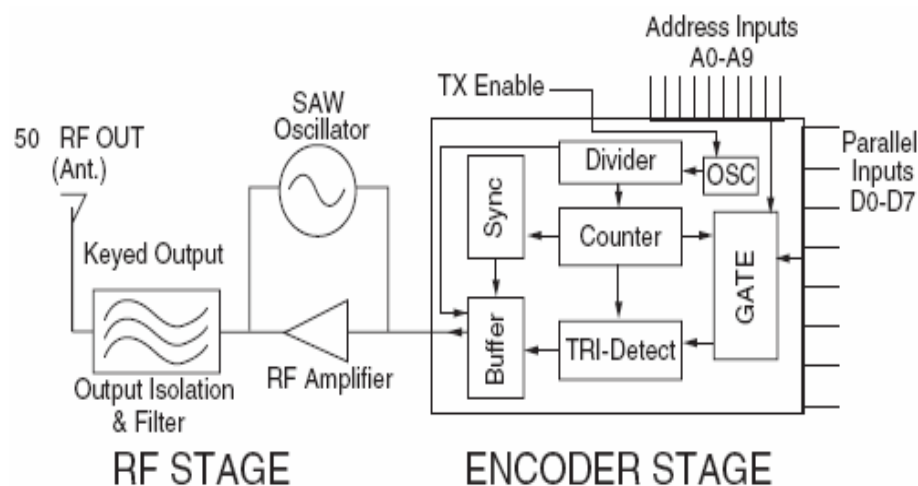
### 2.3. Transmitter/Encoder TXE-433-KH

Seri KH ini ideal untuk digunakan dalam aplikasi OEM seperti *remote control/command* dan *keyless entry*. Ini dikombinasi dengan pengirim RF (*Radio Frequency*) dengan *on board decoder*.

Ketika dipasangkan sesuai dengan modul seri KH *receiver/decoder* sangat memungkinkan untuk digunakan dalam membentuk hubungan *wireless* dan memungkinkan untuk mentransfer status dari 8 input paralel selama pada jarak 90 meter untuk merek yang sama. Tipe ini juga dilengkapi dengan 10 *tri-state address* menyediakan  $3^{10}$  *address* yang berbeda untuk alasan keamanan dan keunikan.

#### 2.3.1. Penjelasan Modul

Modul *transmitter/encoder* ini mengkombinasikan *Surface Acoustic Wave* (SAW) berbasis *transmitter* dengan *on-board encoder*. Kemampuan jarak jangkauannya antara *transmitter* dan *receiver* adalah sejauh 90 meter, dengan catatan modul ini juga dipasangkan dengan modul *transmitter* dari Linx juga. Modul ini juga disediakan  $3^{10}$  atau 59,049 kombinasi *address* yang berbeda-beda. Modul ini mampu menerima sampai dengan 8 *bit* paralel input data. Berikut adalah blok diagram dari modul ini:

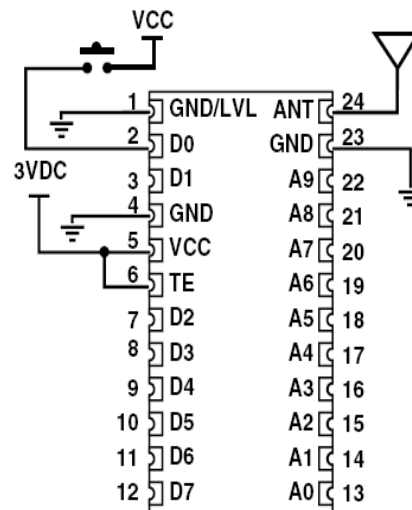


Gambar 2.14. Blok Diagram *Transmitter* Seri KH

Sumber: Linx Technologies. *TXE-433-KH*. 10 Januari 2006. p. 5.  
[http://www.linxtechnologies.com/documents/kh\\_txe\\_manual.pdf](http://www.linxtechnologies.com/documents/kh_txe_manual.pdf)

### 2.3.2. Aplikasi

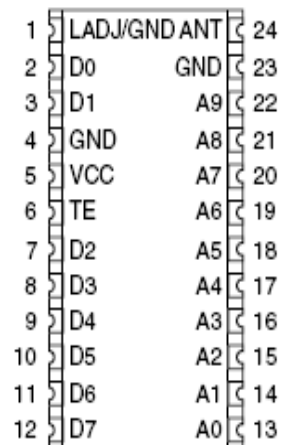
Modul ini mampu beroperasi pada suhu 25°C pada 3 V<sub>DC</sub> atau pada maksimal 5 V<sub>DC</sub>.



Gambar 2.15. Aplikasi Sirkuit *Transmitter*

Sumber: Linx Technologies. *TXE-433-KH*. 10 Januari 2006. p. 3.  
<[http://www.linxtechnologies.com/documents/kh\\_txe\\_manual.pdf](http://www.linxtechnologies.com/documents/kh_txe_manual.pdf)>

### 2.3.3. Deskripsi *Pin*



Gambar 2.16. *Pinout Transmitter* Seri KH

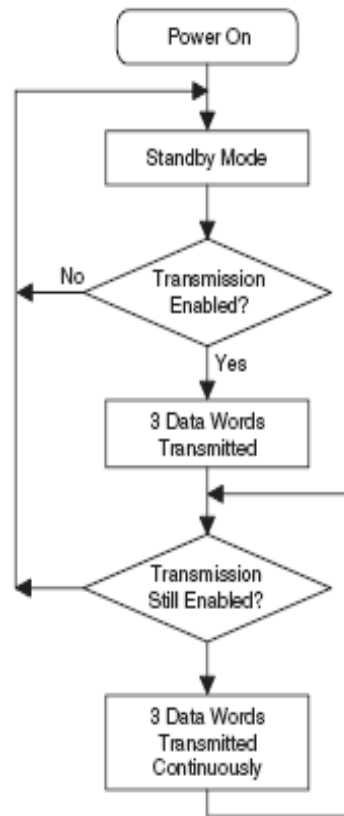
Sumber: Linx Technologies. *TXE-433-KH*. 10 Januari 2006. p. 4.  
<[http://www.linxtechnologies.com/documents/kh\\_txe\\_manual.pdf](http://www.linxtechnologies.com/documents/kh_txe_manual.pdf)>

Tabel 2.10. Deskripsi *Pin Transmitter* Seri KH

<i>Pin #</i>	Name	Penjelasan
1	GND/LADJ	<i>Level Adjust.</i> <i>Pin</i> ini dapat digunakan untuk mengatur level keluaran <i>power</i> dari <i>transmitter</i> ini. Bila <i>pin</i> ini terhubung langsung dengan GND akan memberi level <i>power</i> terbesar. Sedangkan jika ditambahkan resistor terhadap GND, maka akan menurunkan level <i>power</i> -nya.
2,3 & 7 - 12	D0-D7	<i>Data Input Lines.</i> Ketika TE menjadi <i>high</i> , dan modul akan mereplikasikan data dari <i>pin-pin</i> ini untuk ditransmisikan dan dikirimkan kepada <i>receiver</i> .
4	GND	<i>Analog Ground.</i>
5	VCC	<i>Supply Voltage.</i>
6	TE	<i>Transmit Enable Line.</i> Ketika <i>pin</i> ini <i>high</i> , maka data pada <i>pin</i> D0-D7 akan dikirimkan dalam bentuk paket sebanyak 3 kali pengiriman.
13-22	A0-A9	<i>Address Lines.</i> Kombinasi <i>pin-pin</i> ini harus sama dengan kombinasi <i>pin-pin address</i> yang ada di <i>receiver</i> (agar proses transmisi dapat diterima).
23	N/C	<i>Pin</i> ini tidak memiliki koneksi, hanya untuk mendukung fisik modul.
24	ANT	50-ohm RF <i>output</i> .

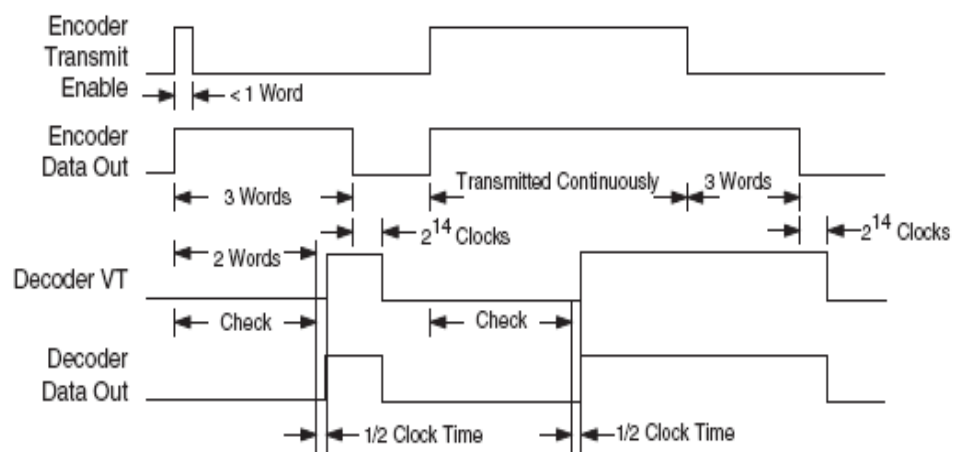
#### 2.3.4. Teori Pengoperasian

Pada saat *pin* TE (*Transmit Enable*) berada pada kondisi *high*, maka *encoder* akan aktif. *Encoder* akan mendeteksi data dan alamat, kemudian diubah menjadi transmisi 3-*word* dan akan terus terjadi proses ini (hingga *pin* TE kembali menjadi *low*). *Encoder* menghasilkan paket data serial yang digunakan untuk memodulasi *transmitter*. Sinyal yang ditransmisikan mungkin diterima juga oleh modul lain yang menggunakan frekuensi sama, namun modul *receiver* lain harus memeriksa *bit-bit* alamat yang dikirimkan oleh *transmitter*. Berikut ini adalah *flowchart* dan *timing diagram* dari proses yang terjadi pada *encoder*.



Gambar 2.17. *Flowchart Encoder*

Sumber: Linx Technologies. *TXE-433-KH*. 10 Januari 2006. p. 6.  
 <[http://www.linxtechnologies.com/documents/kh\\_txe\\_manual.pdf](http://www.linxtechnologies.com/documents/kh_txe_manual.pdf)>



Gambar 2.18. *Timing Diagram Encoder*

Sumber: Linx Technologies. *TXE-433-KH*. 10 Januari 2006. p. 6.  
 <[http://www.linxtechnologies.com/documents/kh\\_txe\\_manual.pdf](http://www.linxtechnologies.com/documents/kh_txe_manual.pdf)>

## 2.4. Modul Ultrasonik PING)))

### 2.4.1. Penjelasan Modul

Ping))) produk dari Parallax ini adalah modul ultrasonik yang berfungsi untuk mengetahui jarak tanpa kontak dengan obyek yang diukur. Kemampuan modul ini berada pada jarak 3 cm (1,2 inches) sampai dengan 3 meter (3,3 yard). Modul ini beroperasi pada tegangan suplai 5 V<sub>DC</sub>.



Gambar 2.19. PING))) Ultrasonik

### 2.4.2. Teori Pengoperasian

Sensor Ping))) mendeteksi jarak obyek dengan cara memancarkan gelombang ultrasonik (40 kHz) selama  $t_{BURST}$  (200  $\mu$ s) kemudian mendeteksi pantulannya. Sensor Ping))) memancarkan gelombang ultrasonik sesuai dengan kontrol dari mikrokontroler pengendali (pulsa *trigger* dengan  $t_{OUT}$  min. 2  $\mu$ s). Gelombang ultrasonik ini melalui udara dengan kecepatan 344 meter per detik, mengenai obyek dan memantul kembali ke sensor. Ping))) mengeluarkan pulsa *output high* (pada *pin* SIG) setelah memancarkan gelombang ultrasonik. Dan setelah gelombang pantulan terdeteksi, maka Ping))) akan membuat *output low* (pada *pin* SIG). Lebar pulsa *high* ( $t_{IN}$ ) akan sesuai dengan lama waktu tempuh gelombang ultrasonik untuk 2 kali jarak ukur dengan obyek. Maka jarak yang diukur adalah  $[(t_{IN} \text{ s} \times 344 \text{ m/s}) \div 2]$  meter. Berikut ini adalah analogi cara kerja ultrasonik dalam mengukur jarak.



Seri LM35 adalah sensor temperatur yang presisi, dimana tegangan *output*-nya adalah linier terhadap satuan derajat Celcius. LM35 tidak memerlukan pengkalibrasian dari peralatan luar lainnya untuk mendapatkan nilai akurasi. LM35 memiliki kemampuan pengukuran suhu antara  $-55^{\circ}\text{C}$  sampai dengan  $150^{\circ}\text{C}$ . Setiap kenaikan 10.0 mV, maka suhu akan naik sebesar  $1^{\circ}\text{C}$ . LM35 hanya menyedot arus 60  $\mu\text{A}$  dari suplainya. Dengan kondisi demikian, maka LM35 sangat sedikit mengeluarkan panas. Sehingga hampir tidak ada *self-heating* (kurang dari  $0,1^{\circ}\text{C}$  pada pengukuran suhu udara) yang mempengaruhi akurasi pengukurannya.

## 2.6. ADC0809

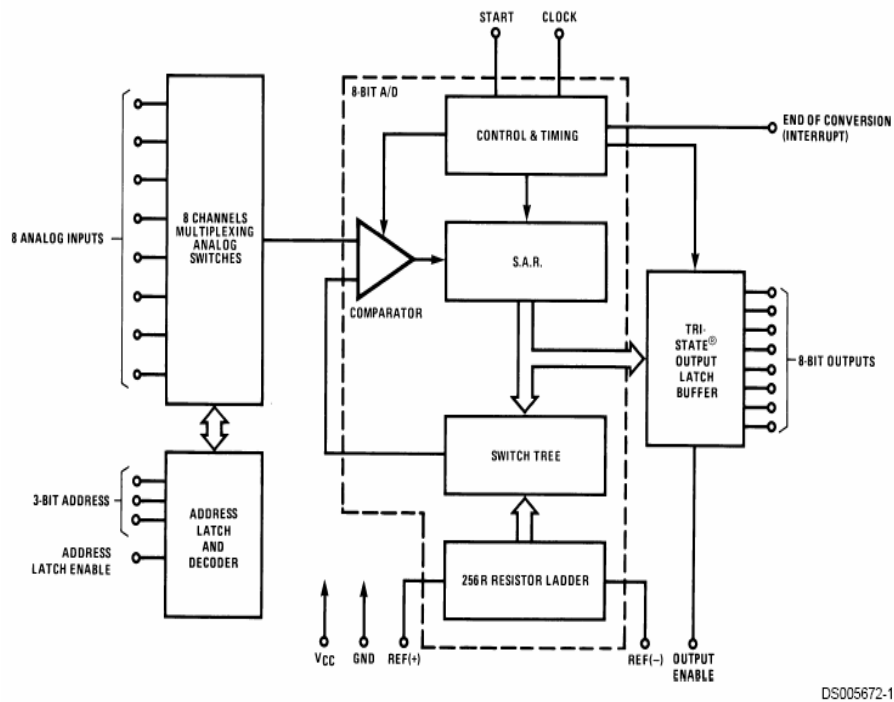
ADC0809 adalah komponen monolitik dari CMOS yang mampu mengubah 8 *bit* data *analog* menjadi *bit-bit* data digital, dengan *multiplexer* 8 *channel* dan cocok dikontrol oleh mikrokontroler. *Multiplexer* 8 *channel* mampu mengakses setiap *pin* dari 8 *pin input analog* secara langsung. *Multiplexer* 8 *channel* dapat diatur dengan mengubah kombinasi logika *address*-nya pada ketiga *pin address*-nya. ADC0809 memiliki kecepatan tinggi, akurasi yang tinggi dan mengkonsumsi *power* yang kecil. ADC0809 ini beroperasi pada tegangan suplai  $5V_{\text{DC}}$ . ADC tipe ini mampu menerima *input-an analog* antara  $0V_{\text{DC}}$  sampai dengan  $5V_{\text{DC}}$ , pada tegangan suplai  $5V_{\text{DC}}$ , dengan kemampuan waktu konversi adalah 100  $\mu\text{s}$ .

Berikut adalah tabel kombinasi *address logic* ADC0809

Tabel 2.11. Kombinasi *Address Logic* ADC0809

<i>Channel Analog</i>	<i>Pin Address</i>		
	C	B	A
IN0	L	L	L
IN1	L	L	H
IN2	L	H	L
IN3	L	H	H
IN4	H	L	L
IN5	H	L	H
IN6	H	H	L
IN7	H	H	H

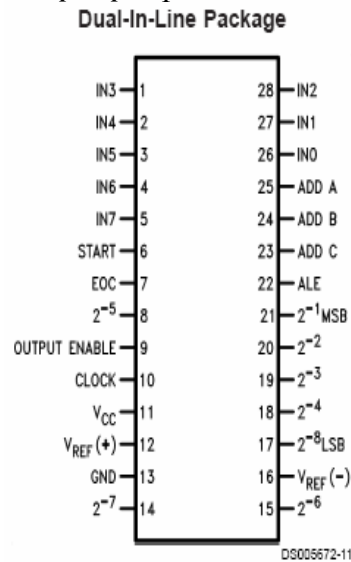
Berikut adalah blok diagram ADC0809.



Gambar 2.23. Blok Diagram ADC0809

Sumber: National Semiconductors. *ADC0809*. 20 Februari 2006. p. 1.  
<<http://www.national.com/ds/ADC/ADC0809.pdf>>

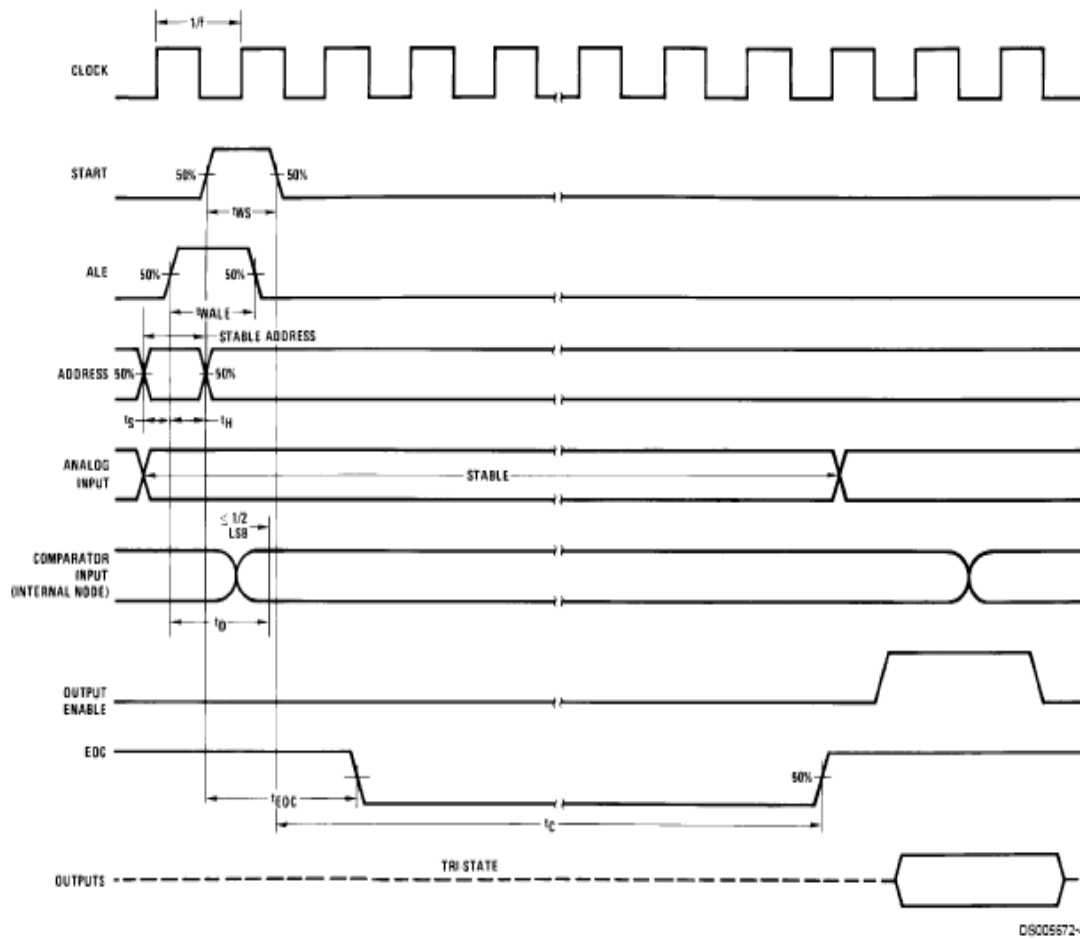
Berikut adalah struktur *pin-pin* pada ADC0809.



Gambar 2.24. Struktur *Pin-pin* ADC0809

Sumber: National Semiconductors. *ADC0809*. 20 Februari 2006. p. 2.  
<<http://www.national.com/ds/ADC/ADC0809.pdf>>

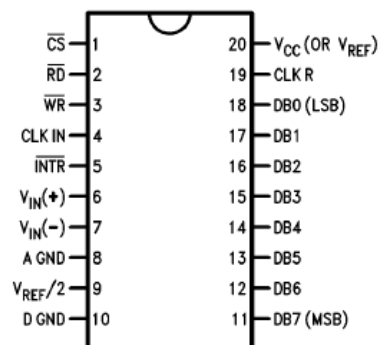
Berikut adalah *timing* diagram pada ADC0809.



Gambar 2.25. *Timing* Diagram ADC0809

Sumber: National Semiconductors. *ADC0809*. 20 Februari 2006. p. 7.  
 <<http://www.national.com/ds/ADC/ADC0809.pdf>>

## 2.7. ADC0804

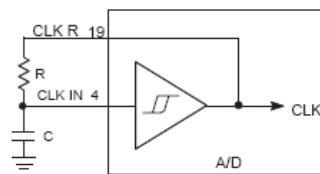


Gambar 2.26. Struktur *Pin-pin* ADC0804

Sumber: National Semiconductors. *ADC0804 Datasheet*. 27 Maret 2006. p. 32.  
 <<http://www.national.com/ds/ADC/ADC0804.pdf>>

ADC0804 adalah komponen CMOS yang berfungsi sebagai konverter data *analog* menjadi data digital. ADC0804 ini mampu menerima *input-an analog* antara  $0V_{DC}$  sampai dengan  $5V_{DC}$ , pada tegangan suplai  $5V_{DC}$ , dengan kemampuan waktu konversi adalah  $100 \mu s$ . ADC0804 menyediakan *pin*  $V_{ref}/2$  untuk tegangan referensi sebagai pembanding *input-an analog*-nya. Jadi jika mengkehendaki tegangan referensinya sebesar  $5V_{DC}$ , maka harus diberikan  $2,5V_{DC}$  pada *pin*  $V_{ref}/2$ . ADC0804 menyediakan *on-chip clock generator*.

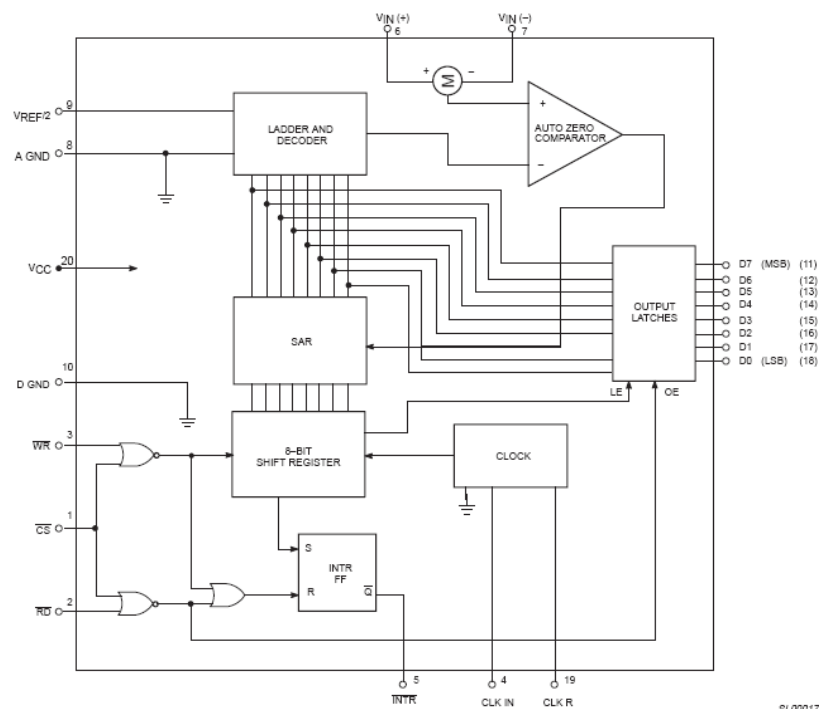
Berikut adalah aplikasi *on-chip clock generator*.



Gambar 2.27. *On-chip Clock Generator* ADC0804

Sumber: National Semiconductors. *ADC0804 Datasheet*. 27 Maret 2006. p. 12.  
<<http://www.national.com/ds/ADC/ADC0804.pdf>>

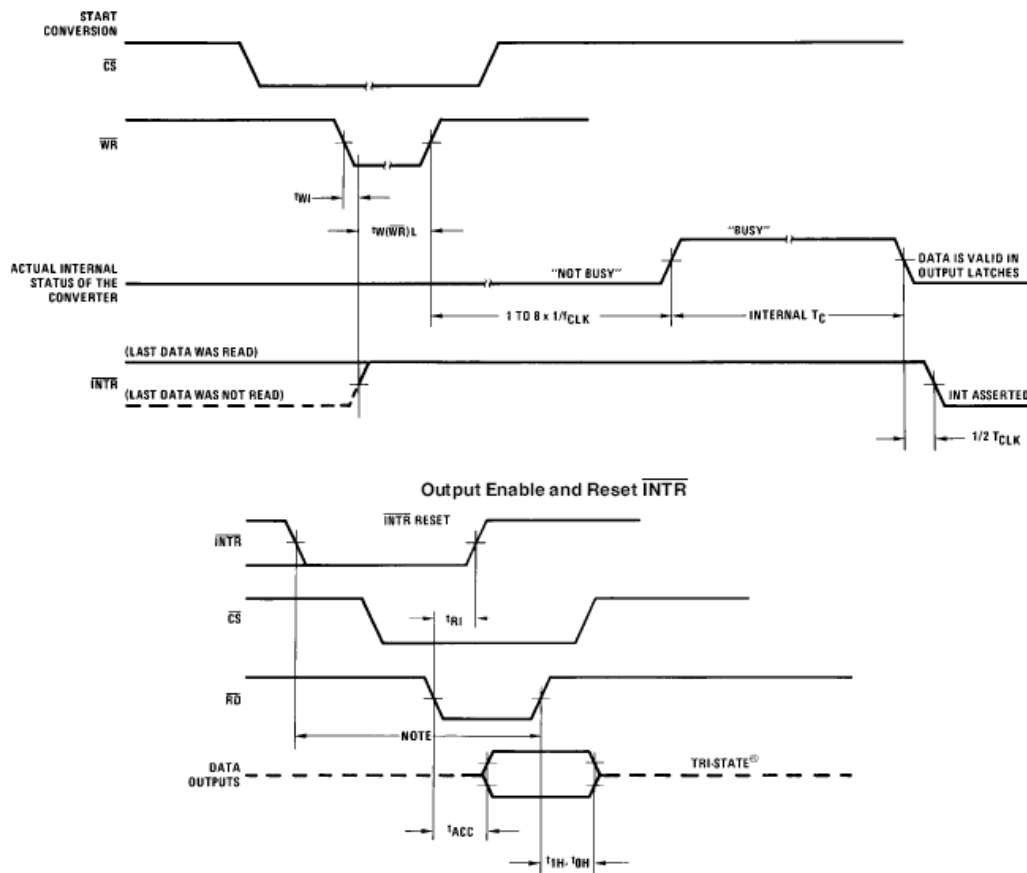
Berikut adalah blok diagram ADC0804.



Gambar 2.28. Blok Diagram ADC0804

Sumber: National Semiconductors. *ADC0804 Datasheet*. 27 Maret 2006. p. 3.  
<<http://www.national.com/ds/ADC/ADC0804.pdf>>

Berikut adalah *timing* diagram ADC0804.



Gambar 2.29. Timing Diagram ADC0804

Sumber: National Semiconductors. *ADC0804 Datasheet*. 27 Maret 2006. p. 10.

<<http://www.national.com/ds/ADC/ADC0804.pdf>>

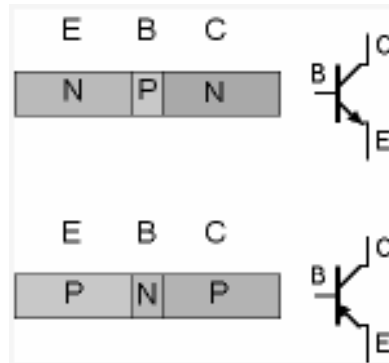
## 2.8. Transistor

Transistor merupakan dioda dengan dua sambungan (*junction*). Sehubungan itu membentuk transistor PNP maupun NPN. Ujung-ujung terminalnya berturut-turut disebut emitor, basis dan kolektor. Basis selalu berada di tengah, diantara emitor dan kolektor. Transistor ini disebut transistor bipolar, karena struktur dan prinsip kerjanya tergantung dari perpindahan elektron di kutub negatif mengisi kekurangan elektron (*hole*) di kutub positif. William Schockley adalah orang yang pertama kali menemukan transistor bipolar pada tahun 1951.

Pada tugas akhir ini difungsikan sebagai *switch*, maka transistor hanya boleh memiliki 2 kondisi, yakni kondisi 'on' atau 'off'. Pada saat transistor pada kondisi 'on' atau kondisi saturasi, besarnya  $V_{ce}$  adalah mendekati nol. Sedangkan

arus yang melalui kolektor adalah maksimum. Pada saat transistor pada kondisi ‘off’ atau kondisi *cut off*, besarnya  $V_{ce}$  adalah maksimum. Sedangkan arus yang melalui kolektor adalah nol.

Berikut adalah 2 jenis transistor:



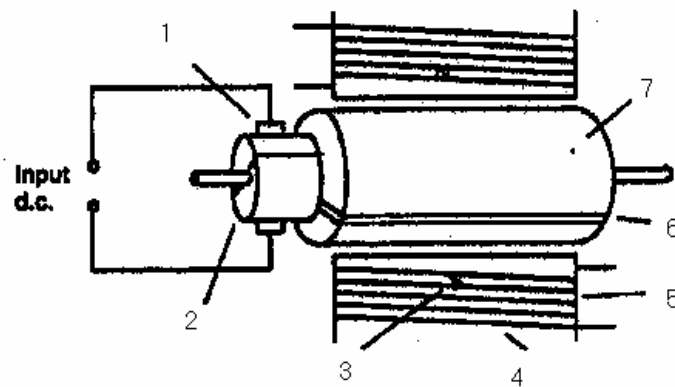
Gambar 2.30. Transistor NPN dan PNP

## 2.9. Motor DC



Gambar 2.31. Motor 24V<sub>DC</sub>

Sebuah motor DC memiliki kumparan-kumparan kawat yang dipancangkan di dalam slot-slot sebuah silinder yang terbuat dari bahan feromagnetik. Silinder ini di beri nama *armature*. *Armature* dipasang pada suatu bentuk dudukan (*bearing*) dan bebas berputar. Dudukan *armature* adalah sebuah medan magnet yang dihasilkan oleh magnet-magnet permanen atau arus yang dialirkan melalui kumparan-kumparan kawat yang dinamakan kumparan medan. Kedua magnet ini, magnet permanen maupun elektromagnet, disebut sebagai *stator* (bagian yang diam). Ketika arus mengalir melalui kumparan *armature*, karena sebuah konduktor berarus yang berada tegak lurus terhadap sebuah medan magnet akan mengalami gaya, gaya-gaya akan bekerja pada kumparan tersebut dan mengakibatkan perputaran (gambar 2.32).



Gambar 2.32. Motor DC

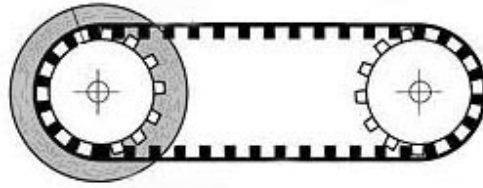
Sumber: Bolton, William. *Programmable Logic Controller (PLC): sebuah pengantar*. (Irzam Harmein, Trans.) Jakarta: Erlangga. 2004. p. 25.

Keterangan elemen dasar dari sebuah motor DC:

1. Sikat
2. Komutator
3. Kutub-kutub kumparan medan
4. Kumparan medan
5. Stator
6. Kumparan *armature* di dalam slot
7. *Armature*

Kecepatan putaran dapat diubah dengan cara mengubah besar arus pada kumparan *armature*. Akan tetapi, karena sumber tegangan tetap biasanya digunakan sebagai *input* ke kumparan, perubahan arus yang diperlukan seringkali diperoleh melalui penggunaan sebuah rangkaian elektronik. Rangkaian ini dapat mengontrol nilai rata-rata tegangan, dengan cara mengubah-ubah interval waktu untuk menghasilkan tegangan DC yang bervariasi, misalnya menggunakan rangkaian PWM (*Pulse Width Modulation*).

Contoh aplikasi dari motor DC adalah sebagai penggerak pada robot amfibi ini. Motor akan membantu roda pada robot amfibi menjadi berputar (sehingga robot amfibi dapat bergerak berpindah tempat sesuai gerakan atau arah putaran yang diberikan oleh motor).



Gambar 2.33. Aplikasi Roda Amphibi

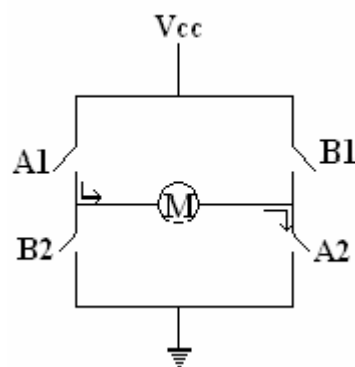
Motor yang digunakan dalam robot amfibi ini memiliki sistem *gearbox*-nya sendiri. *Gearbox* pada motor ini berfungsi untuk membantu kerja motor sehingga dapat memutar beban yang lebih berat.



Gambar 2.34. *Gearbox* Motor DC

## 2.10. H-Bridge

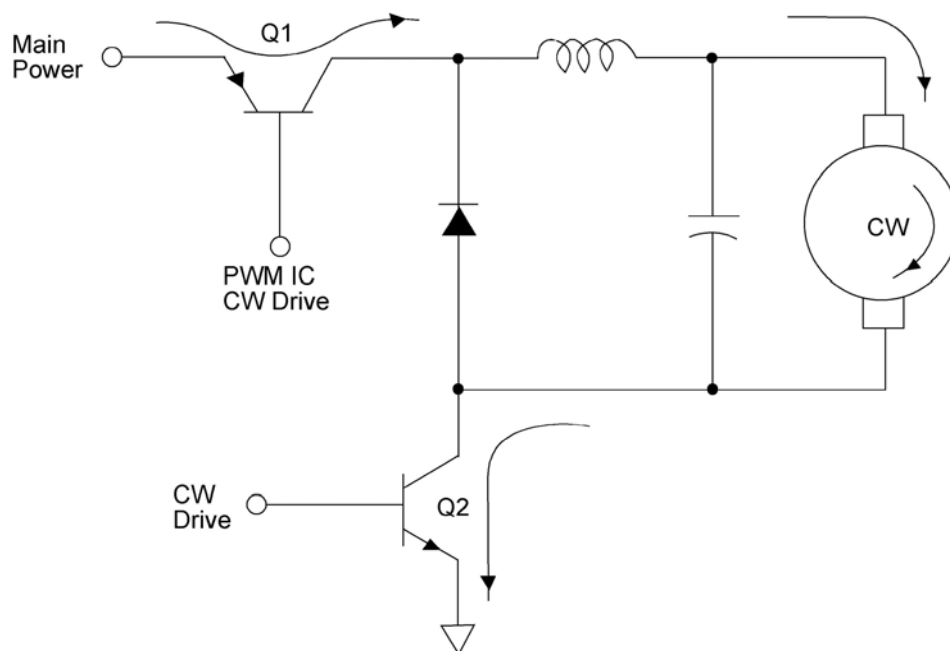
Rangkaian *h-bridge* terdiri dari transistor jenis NPN dan PNP. Rangkaian *h-bridge* tersebut menghasilkan keluaran dengan putaran motor yang dapat berputar bolak-balik, prinsip kerja *h-bridge* secara sederhana dapat dilihat pada gambar 2.35. Bila saklar A1 dan A2 aktif, maka arus akan mengalir dari Vcc menuju A1 dan menggerakkan motor searah jarum jam, lalu arus akan menuju A2. Dan motor akan bergerak dengan arah berlawanan dengan arah jarum jam, bila saklar B1 dan B2 aktif.



Gambar 2.35. *H-Bridge*

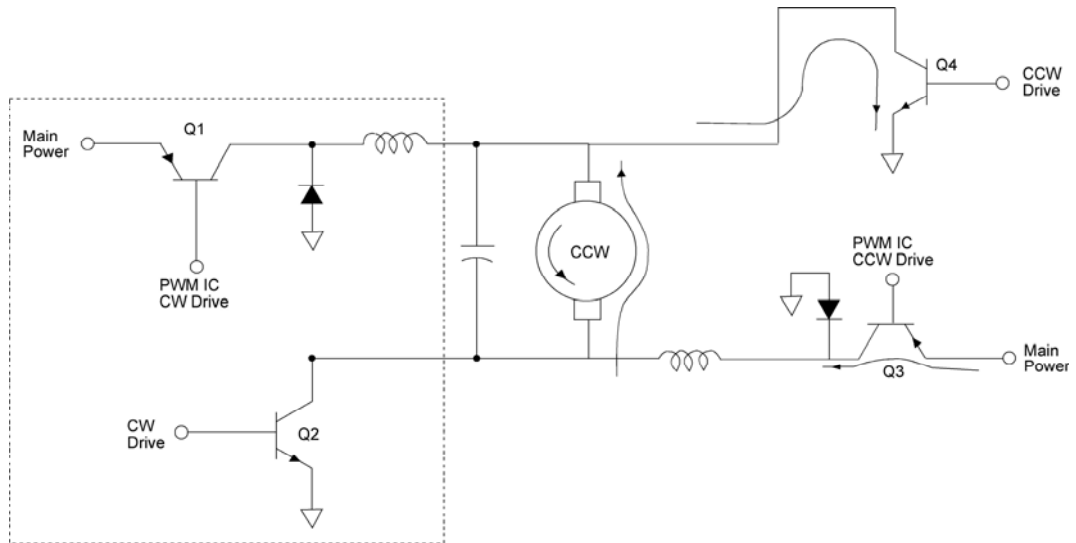
Langkah pertama untuk menghasilkan putaran motor searah jarum jam (*clock wise*) dalam rangkaian *h-bridge* yaitu penempatan hubungan motor dengan transistor *power* NPN. Hal ini diperlihatkan dalam gambar 2.36. Untuk menjalankan motor dengan putaran searah jarum jam, transistor Q2 harus dalam keadaan saturasi (sehingga terjadi pentanahan antara motor dengan komponen filter dalam hal ini adalah kapasitor).

Untuk putaran motor berlawanan dengan arah jarum jam (*counter clock wise*) membutuhkan arus yang mengalir ke atas untuk melewati motor. Hal ini ditandai dengan penambahan dua set transistor (Q3 dan Q4) dan komponen filter. Hal ini ditunjukkan pada gambar 2.37. Sehingga untuk mendapatkan putaran berlawanan dengan arah jarum jam, transistor Q1 dan Q2 yang menghasilkan putaran searah jarum jam harus dalam keadaan mati (*cut off*), sedangkan transistor Q4 harus dalam keadaan aktif.



Gambar 2.36. *H-Bridge Clockwise*

Sumber: Jacob, J. Michael. *Industrial Control Electronics Applications And Design*. Englewood Cliffs, N.J.: Prentice Hall. 1988. p. 458.



Gambar 2.37. *H-Bridge Counter Clockwise*

Sumber: Jacob, J. Michael. *Industrial Control Electronics Applications And Design*. Englewood Cliffs, N.J.: Prentice Hall. 1988. p. 458.