BAB II

DASAR TEORI

2.1 Tinjauan Aturan Terkait Air Bawah Tanah

Air bawah tanah sering disebut dengan air tekanan yaitu air yang tersimpan dalam lapisan tanah. Air bawah tanah adalah air sumur gali dan air sumur bor. Air tanah adalah air yang terdapat dalam lapisan tanah atau bebatuan di bawah permukaan tanah. Air tanah merupakan salah satu sumber daya air yang keberadaannya terbatas dan kerusakannya dapat mengakibatkan dampak yang luas serta pemulihannya sulit dilakukan.

Selain air sungai dan air hujan, air tanah juga mempunyai peranan yang sangat penting terutama dalam menjaga keseimbangan dan ketersediaan bahan baku air untuk kepentingan rumah tangga (domestik) maupun untuk kepentingan industri. Dibeberapa daerah, ketergantungan pasokan air bersih dan air tanah telah mencapai ± 70%.

Setiap daerah memiliki peraturan tentang perpajakan yang diberlakukan oleh pemerintah daerah tersebut. Peraturan yang mendukung adanya tata kelola tentang lingkungan, konsumsi dan biaya penggunaan air tanah didukung oleh SE Gubernur no. 37/SE/2011, Keputusan Gubernur DKI No 42 Tahun 2011 tentang Perubahan atas Keputusan Gubernur DKI No 88 Tahun 1999 tentang Petunjuk Pelaksana Penyelenggaraan dan Pemungutan Pajak Air Bawah Tanah dan Air Permukaan di DKI Jakarta. Mengenai pemungutan pajak dari penggunaan air tanah sudah diaturkan

dalam Peraturan Daerah Nomor 6 Tahun 2010, tentang Ketentuan Umum Pajak Daerah yang sudah tercantum kriteria pemungutan pajak penggunaan atau pemanfaatan air tanah tersebut.

Wajib Pajak Air Tanah adalah orang pribadi atau badan yang melakukan pengambilan dan/atau pemanfaatan air tanah. Dasar pengenaan Pajak Air Tanah adalah nilai perolehan air tanah, nilai perolehan air tanah sebagaimana disebutkan dalam ayat (1), dinyatakan dalam rupiah yang dihitung dengan mempetimbangkan factor-faktor berikut :

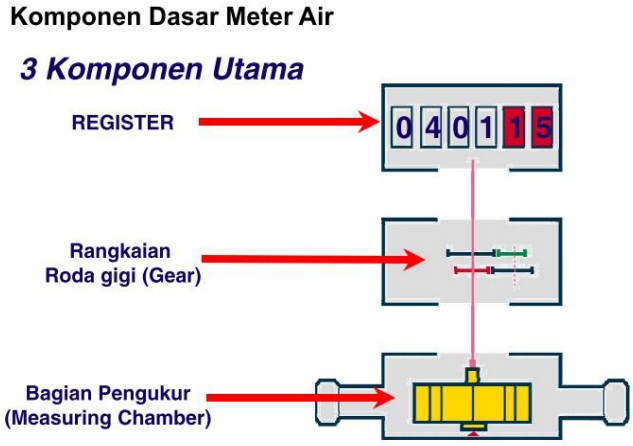
- Jenis sumber air
- Lokasi sumber air
- Tujuan pengambilan dan/atau pemanfaatan air
- Volume air yang diambil dan/atau dimanfaatkan
- Kualitas air
- Tingkat kerusakan lingkungan yang disebabkan oleh pengambilan atau pemanfaatan air tanah.

2.2 *Flow meter*

Flowmeter adalah alat yang digunakan untuk mengukur massa atau laju aliran volumetrik cairan atau gas. Sebelum menetapkan *flowmeter*, juga dianjurkan untuk menentukan apakah aliran informasi akan lebih berguna jika disajikan dalam unit massa atau volumetrik.

Tidak semua *fluida* yang berpindah dinamakan *fluida* bergerak. Yang dimaksud *fluida* bergerak adalah jika *fluida* tersebut bergerak lurus terhadap sekitar. Aliran *fluida* dikatakan aliran garis lurus apabila aliran *fluida* yang mengalir mengikuti suatu garis (lurus melengkung) yang jelas ujung pangkalnya. Aliran garis lurus juga disebut aliran berlapis atau aliran laminar (laminar flow). Kecepatan- kecepatan partikel di tiap titik pada garis arus, searah dengan garis singgung di titik itu. Dengan demikian garis arus tidak pernah berpotongan. Pada *fluida* yang tak termampatkan, hasil kali antara kelajuan aliran *fluida* dan luas penampangnya selalu tetap. Jadi A.v = konstan, atau disebut debit (Q). Debit adalah volume *fluida* (m3) yang mengalir melewati suatu penampang dalam selang waktu tertentu.

Prinsip kerja *flowmeter* menggunakan laju air sebagai penggerak dari mekanikal yang ada di dalam meter dan terhubung kepada angka register meter. Kecepatan pada air secara spesifik di konversi menjadi volume yang terbaca pada register meter, Prinsip kerja dari *flowmeter* digambarkan pada gambar 2.1 :



Gambar 2.1 Prinsip kerja *flowmeter*

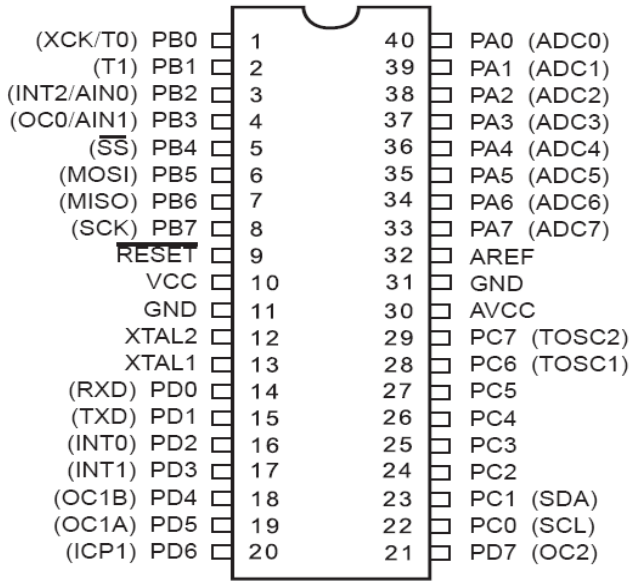
2.3 Mikrokontroler AVR ATMega 8535

ATMega 8535 merupakan salah satu mikrokontroler 8 *bit* buatan Atmel untuk keluarga AVR yang diproduksi secara masal pada tahun 2006. Karena merupakan keluarga AVR, maka ATMega 8535 juga menggunakan arsitektur RISC *(Reduced Intruction Set Computing)*.

AVR memiliki keunggulan dibandingkan dengan mikrokontroler lain, keunggulan mikrokontroler AVR yaitu memiliki kecepatan dalam mengeksekusi program yang lebih cepat, karena sebagian besar instruksi dieksekusi dalam 1 siklus *clock* (lebih cepat dibandingkan mikrokontroler keluarga MCS51 yang memiliki arsitektur *Complex Instruction Set Computer* (CISC)). ATMega 8535 memiliki fitur-fitur, antara lain :

1. Sistem mikroprosesor berbasis RISC dengan kecepatan maksimal 16 Mhz,
2. Memiliki *memory flash* untuk program sebesar 8 Kb,
3. SRAM sebesar 512 *byte*,
4. EEPROM sebesar 512 *byte*,
5. 2 buah *Timer / Counter* 8 *bit* dan 1 buah *Timer / Counter* 16 *bit*,
6. 4 *Channel* PWM,
7. 8 *Channel* ADC dengan resolusi 10 *bit*,
8. Komunikasi USART.

2.3.1 Konfigurasi Pin



Gambar 2.2 Konfigurasi Pin ATMega 8535

Gambar diatas merupakan konfigurasi pin ATMega 8535 dengan kemasan 40 *pin* DIP (*Dual In-line Package*). Berikut penjelasan fungsi-fungsi pin ATMega 8535 :

1. VCC merupakan *pin* yang berfungsi sebagai *pin* masukan catu daya.
2. GND merupakan *pin ground*.
3. PORTA (PA0..PA7) merupakan *pin* I/O dua arah dan *pin* masukan ADC.
4. PORTB (PB0..PB7) merupakan *pin* I/O dua arah dan *pin* fungsi khusus.
5. PORTC (PDC..PC7) merupakan *pin* I/O dua arah dan *pin* fungsi khusus, yaitu *timer/counter, comparator analog,* dan SPI.
6. PORTD (PD0..PD7) merupakan *pin* I/O dua arah dan *pin* fungsi khusus, yaitu *comparator analog*, interupsi eksternal, dan komunikasi serial.
7. RESET merupakan *pin* yang digunakan untuk me-*reset* 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.

2.3.2 Dasar I/O

ATMega 8535 memiliki 4 buah PORT I/O dinamakan PORTA, PORTB, PORTC, dan PORTD. Masing-masing PORT memiliki 3 *register* PORTx, DDRx, dan PINx (x= urutan, misalnya PORTA, DDRA, PINA). Masing-masing *register* dapat diatur dan dibaca tiap *bit*-nya. DDRx merupakan *register* pengaturan arah PORT, sedangkan PORTx digunakan untuk mengeluarkan dan PINx untuk membaca dari PORT.

Tabel 2.1 Pengaturan PORT

	DDR *bit* = 1	DDR *bit* = 0
PORT *bit* = 1	*Output High*	*Input Pull-up*
PORT *bit* = 0	*Output Low*	*Input floating (Tri-state)*

2.3.3 ADC

ATMega 8535 telah dilengkapi dengan 8 *channel* 10-*bit Analog to Digital Converter* (ADC). Rangkaian internal ADC ini memiliki catu daya tersendiri yaitu kaki AVCC. Data pembacaan ADC terdapat pada *register* ADCH dan ADCL.

Sedangkan untuk pengaturan digunakan *register* ADMUX, ADCSRA, dan SFIOR.

Tabel 2.2 ADMUX (ADC *Multiplexer Selection Register*)

Bit	7	6	5	4	3	2	1	0	
	REFS1	REFS0	ADLAR	MUX4	MUX3	MUX2	MUX1	MUX0	ADMUX
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Keterangan :

1. REFS1 : 0 (*Reference Selection Bits*), untuk mengatur sumber tegangan referensi untuk ADC.
2. ADLAR (*ADC Left Adjust Result*), untuk format penyimpanan data ADC.
3. MUX4 : 0 (*Analog Channel and Gain Selection Bits*), untuk memilih saluran pembacaan ADC.

Tabel 2.3 ADCSRA (ADC *Control and Status Register A*)

Bit	7	6	5	4	3	2	1	0	
	ADEN	ADSC	ADATE	ADIF	ADIE	ADPS2	ADPS2	ADPS0	ADCSRA
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Keterangan :

1. ADEN (ADC *Enable*).
2. ADCS (ADC *Start Convesion*). Di-*set* untuk mulai konversi ADC. *Bit* ini akan aktif selama ADC sibuk dan akan bernilai 0 jika telah selesai dikonversi.
3. ADATE (ADC *Auto Trigger Enable*). Mengatur aktivasi picu otomatis operasi ADC. Pengaturan sumber pemicuan terdapat pada *register* SFIOR.
4. ADIF (ADC *Interrupt Enable*).
5. ADPS2 : 0 (ADC *Prescaler Select*). Mengatur *prescaler* untuk ADC.

ATMega 8535 mengubah tegangan atau data analog ke digital dengan rumus sebagai berikut :

$$ADC = Vin\ x \frac{n}{Vref} \quad \text{..... (2-1)}$$

Keterangan :

ADC : hasil konversi data analog ke digital

Vin : tegangan input analog

n : jumlah resolusi (8 *bit* atau 10 *bit*)

Vref : tegangan referensi

2.3.4 *Timer/Counter*

ATMega 8535 memiliki 3 buah *Timer/Counter*, yaitu *Timer/Counter0* (8-*bit*), *Timer/Counter1* (16-*bit*) dan *Timer/Counter2* (8-*bit*). Penjelasan untuk masing-masing *Timer/Counter* akan dijelaskan berikut ini :

2.3.4.1 *Timer/Counter0* dan *Timer/Counter2*

Timer/Counter0 dan *Timer/Counter2* adalah 8-*bit Timer/Counter* yang serbaguna. Kedua *timer/Counter* tersebut memiliki fitur antara lain :

1. 1 *channel counter*,
2. *timer* dinolkan (*clear*) pada *match-compare* (*auto reload*),
3. *glitch-free, phase correct pulse width modulator* (PWM),

4. *counter* yang di-*trigger* dari luar (hanya pada *timer/counter0*),
5. 10-*bit clock prescaler*,
6. Interupsi pada saat *overflow* (TOVn) dan *match compare* (OCFn).

Register TCNTn (*Timer/Counter Register*) merupakan *register* yang menyimpan nilai dari *timer/counter*. *Register* OCRn (*Output Compare Register*) merupakan *register* pembanding, jika nilai OCRn = TCNTn maka terjadi *Compare Match*. *Compare Match* dapat digunakan untuk menghasilkan pulsa pada *pin* OCRn. Pengaturan *timer/counter0* dan *timer/counter2* dilakukan melalui *register Timer/Counter Control Register* (TCCRn).

Tabel 2.4 TCCRn (*Timer/Counter Control Register* n)

Bit	7	6	5	4	3	2	1	0	
	FOCn	WGMn0	COMn1	COMn0	WGMn1	CSn2	CSn1	CSn0	TCCRn
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Keterangan :

1. FOCn (*Force Output Compare*)

   Memaksakan operasi *Compare Match*. Hanya aktif pada *mode* non-PWM FOCn tidak akan memicu terjadinya interupsi atau men-*clear*-kan *timer* pada *mode* CTC.

2. WGMn1:0 (*Waveform Generation Mode*)

   Bit-bit ini mengontrol kenaikkan dari *counter*, sumber dari nilai maksimal dari *counter* dan tipe *timer/counter* yang akan digunakan.

3. COMn1:0 (*Compare Match Output*)

Bit-bit ini mengontrol *pin Output Compare* n (OCn). Apabila kedua *bit* tersebut bernilai 0 maka OCn akan berfungsi sebagai *pin* biasa. Apabila salah satu *bit* bernilai 1 maka fungsi dari OCn bergantung pada pengaturan pada WGMn1:0.

4. CSn2:0 (*Clock Select*)

   Bit-bit ini mengontrol sumber *clock* yang akan digunakan dan *prescaler*-nya. Terdapat perbedaan knfigurasi CSn2:0 untuk kedua *timer/counter*. Sumber *clock* untuk *timer/counter*2 *default*-nya terhubung dengan MCU *clock* (clk). *Timer/counter*2 dapat menggunakan *clock* eksternal yang terhubung dengan *pin* TOSC1 dan TOSC2 dan mengaktifkan *bit* AS2 pada *register Asynchronous Status Register* (ASSR).

2.3.4.2 *Timer/Counter*1

*Timer/counter*1 merupakan *timer/counter* 16-*bit* yang memungkinkan pewaktuan yang lebih akurat. Fitur dari *Timer/Counter*1 antara lain :

1. 16-*bit* (memungkinkan untuk 16-*bit* PWM), 2 buah *compare unit*,
2. 2 buah *register* pembanding,
3. 1 buah *input capture unit* dengan *noise canceller*,
4. *clear timer* pada *match compare* (*auto reload*),
5. *glitch-free*, *Phase Correct Pulse Width Modulator* (PWM),
6. perioda PWM yang dapat diubah-ubah,
7. generator frekuensi,

8. *counter* yang di-*trigger* dari luar,
9. 4 buah pemicu interupsi, (TOV1, OCF1A, OCF1B, and ICF1).

Register TCNT1H:TCNT1L (*Timer/Counter*1 *Register*) merupakan register 16-*bit* yang menyimpan nilai dari *timer/counter1*. *Timer/counter1* memiliki 2 *register* pembanding OCR1AH:OCR1AL dan OCR1BH:OCR1BL (*Output Compare Register*) merupakan *register* pembanding 16-*bit*.

Selain diatas *timer/counter1* juga memiliki *register* ICR1H:ICR1L (*Input Capture Register*) yang mencacah kejadian pada *pin* ICP1. Pengaturan *timer/counter1* dilakukan 2 buah *register Timer/Counter1 Control Register* (TCCR1A dan TCCR1B).

Tabel 2.5 ECCR1A (*Timer/Counter1 Control Register* A)

Bit	7	6	5	4	3	2	1	0	
	COM1A1	COM1A0	COM1B1	COM1B0	F0C1A	F0C1B	WGMI1	WGMI0	TCCR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Keterangan :

1. COM1A1 : 0 (*Compare Match Output Mode* untuk *channel* A)
2. COM1B1: 0 (*Compare Match Output* Mode untuk *channel* B)
3. FOC1A : (*Force Output Compare* untuk *channel* A)
4. FOC1B : (*Force Output Compare* untuk *channel* B)

Tabel 2.6 TCCR1B (*Timer/Counter1 Control Register* B)

Bit	7	6	5	4	3	2	1	0	
	ICNC1	ICES1	-	WGMI3	WGMI2	CSI2	CSI1	CSI0	TCCR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Keterangan :

1. WGMI 3:0 (*Waveform Generation Mode*)
2. ICNC1 (*Input Noise Canceler*), diberi nilai 1 untuk mengaktifkan *noise canceler* untuk masukkan *pin* ICP1.
3. ICES (*Input Capture Edge Select*), mengontrol jenis pemicuan yang akan mengaktifkan *capture* pada *pin* ICP1 (0 = *falling edge*, 1 = *rising edge*)
4. CS12:0 (Clock Select), *bit-bit* ini mengontrol sumber *clock* yang akan digunakan dan *prescaler*-nya. Terdapat perbedaan konfigurasi t CSn2:0 untuk kedua *timer/counter*.

2.3.5 *Interrupt* (Interupsi)

Interupsi adalah kondisi di mana pada saat program utama dieksekusi/dikerjakan oleh CPU kemudian tiba-tiba berhenti untuk sementara waktu karena ada rutin lain yang harus ditangani terlebih dahulu oleh CPU, dan setelah selesai mengerjakan rutin tersebut CPU kembali mengerjakan intruksi pada program utama. ATMega 8535 mempunyai 21 buah sumber interupsi. Untuk mengaktifkan *interrupt bit* I (*Global Interrupt Enable*) pada SREG perlu diaktifkan terlebih dahulu kemudian melakukan pengaturan pada *register* yang bersesuaian untuk masing-masing *interrupt*.

Tabel 2.7 Alamat *Interrupt*

No	Nama	Alamat	Sumber
1	Reset	0x0000	*Reset*
2	INT0	0x0001	*External Interrupt Request 0*
3	INT1	0x0002	*External Interrupt Request 1*
4	TIMER2 COMP	0x0003	*Timer/Counter 2 Compare Match*
5	TIMER2 OVF	0x0004	*Timer/Counter 2 Overflow*
6	TIMER1 CAPT	0x0005	*Timer/Counter 1 Capture Event*
7	TIMER1 COMPA	0x0006	*Timer/Counter 1 Compare Match A*
8	TIMER1 COMPB	0x0007	*Timer/Counter 1 Compare Match B*
9	TIMER1 OVF	0x0008	*Timer/Counter 1 Overflow*
10	TIMER0 OVF	0x0009	*Timer/Counter 0 Overflow*
11	SPI STC	0x000A	*Serial Transfer Complete*
12	USART RXC	0x000B	*USART Rx Complete*
13	USART UDRE	0x000C	*USART Data Register Empty*
14	USART TXC	0x000D	*USART Tx Complete*
15	ADC	0x000E	*ADC Conversion Complete*
16	EE_RDY	0x000F	*EEPROM Ready*
27	ANA_COMP	0x0010	*Analog Comparator*
28	TWI	0x0011	*Two-Wire Serial Interface*
29	INT2	0x0012	*External Interrupt Request 2*
20	TIMER2_COMP	0x0013	*Timer/Counter 0 Compare Match*
21	SPM_RDY	0x0014	*Store Program Memory Ready*

ATMega 8535 memiliki 3 buah sumber interupsi eksternal (INT0, INT1, dan INT2) untuk INT0 dan INT1 pengaturan keadaan yang menyebabkan terjadinya interupsi terdapat pada *register* MCUCR (*MCU Control Register*), sedangkan untuk INT2 terdapat pada MCUCSR (*MCU Control and Status Register*).

Tabel 2.8 MCUCR (*MCU Control Register*)

Bit	7	6	5	4	3	2	1	0	
	SM2	SE	SM1	SM0	ISC11	ISC10	ISC01	ISC00	MCUCR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

ISC01 dan ISC00 merupakan *bit* untuk mengatur kondisi terjadi interupsi untuk INT0, sedangkan ISC10 dan ISC11 merupakan *bit* untuk mengatur kondisi terjadinya interupsi untuk INT1.

Tabel 2.9 Pengaturan Kondisi Terjadinya Interupsi pada INT0

ISC11	ISC10	Deskripsi
0	0	Logika 0 pada INT1 menyebabkan interupsi
0	1	Perubahan logika INT1 menyebabkan interupsi
1	0	Transisi dari 1 ke 0 (*failing edge*) pada INT1 menyebabkan interupsi
1	1	Transisi dari 0 ke 1 (*rising edge*) pada INT1 menyebabkan interupsi

Tabel 2.10 MCUCSR (MCU *Controland Status Register*)

Bit	7	6	5	4	3	2	1	0	
	-	ISC2	-	-	WRDF	BORF	EXTRF	PORF	MCUCSR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Tabel 2.11 Pengaturan Kondisi Terjadinya Interupsi pada INT2 ISC2

ISC2	Deskripsi
0	Transisi dari 1 ke 0 (*failing edge*) pada INT2 menyebabkan interupsi
1	Transisi dari 0 ke 1 (*rising edge*) pada INT2 menyebabkan interupsi

Untuk mengaktifkan masing-masing interupsi selain *bit* 1 pada SREG harus aktif juga dilakukan pengaturan pada *register* GICR (*Global Interrupt Control Register*).

Tabel 2.12 GICR (*Global Interrupt Control Register*)

Bit	7	6	5	4	3	2	1	0	
	INT1	INT0	INT2	-	-	-	IVSEL	IVCE	GICR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Tabel 2.13 *Bit-Bit* Untuk Mengaktifkan Interupsi Eksternal *Bit*

Bit	Deskripsi
INT1	*External Interrupt Request* 1 *Enable* (diberi nilai 1 untuk mengaktifkan)
INT0	*External Interrupt Request* 0 *Enable* (diberi nilai 1 untuk mengaktifkan)
INT2	*External Interrupt Request* 2 *Enable* (diberi nilai 1 untuk mengaktifkan)

2.3.6 Komunikasi Serial ATMega 8535

ATMega 8535 telah dilengkapi *Universal Synchronous and Asynchronous serial Receiver and Transmitter* (USART). Terdapat 2 jenis cara komunikasi USART, yaitu : sinkron dan asinkron. Melalui USART dapat dilakukan komunikasi serial dengan komputer atau perangkat lainnya.

Komunikasi USART dilakukan melalui *kaki* RXD (PD0) dan TXD (PD1) serta *kaki* XCK (PB0) untuk komunikasi serial sinkron. Untuk mengontrol USART digunakan *register* UCSRA, UCSRB dan UCSRC (USART *Control and Status Register* A< B, C) dan UBRR (USART *Baud Rate Register*), menulis ke UDR akan mengirimkan data sedangkan membaca dari UDR akan mengambil data dari *buffer*.

Tabel 2.14 UCSRA (USART *Control and Status Register* A)

Bit	7	6	5	4	3	2	1	0	
	RXC	TXC	UDRE	FE	DOR	PE	U2X	MPCM	UCSRA
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Keterangan :

1. RXC (USART *Receive Complete*). *Bit* ini akan aktif jika ada data masuk.
2. TXC (USART *Transmit Complete*). *Bit* ini akan aktif pada saat selesai melakukan pengiriman data.

3. UDRE (USART *Data Register Empty*). *Bit* ini menandakan transmitter siap untuk menerima data.
4. FE (*Framing Error*). *Bit* ini akan aktif jika terdapat *error* saat menerima data.
5. DOR (*Data OveRun*). *Bit* ini akan aktf jika ada data yang masuk namun register UDR penuh (belum dibaca).
6. PE (*Parity Error*). *Bit* ini akan aktif jika terdapat *error parity* saat menerima data.
7. U2X (*Double* USART *Transmission Speed*). *Bit* ini digunakan untuk menganda kecepatan *baud rate*.
8. MPCM (*Multi-processor Communication Mode*). *Enable Multi-processor Communication.*

Tabel 2.15 UCSRB (USART *Control and Status Register* B)

Bit	7	6	5	4	3	2	1	0	
	RXCIE	TXCIE	UDRIE	RXIE	TXEN	UCSZ2	RXB8	TXB8	UCSRB
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Keterangan :

1. RXCIE (USART *Receive Complete Interrupt Enable*)
2. TXC (USART *Transmit Complete Interrupt Enable*)
3. UDRIE (USART *Data Register Empty Interrupt Enable*)
4. RXEN (*Receive Enable*)
5. TXEN (*Transmit Enable*)
6. UCSZ2 (*Character Size*)

7. RXB8 (*Receive Data Bit* 8). Menampung *bit* ke 8 pada penerimaan 9-*bit*.
8. TXB8 (*Transmit Data Bit* 8).

Tabel 2.16 UCSRC (USART *Control and Status Register* C)

Bit	7	6	5	4	3	2	1	0	
	URSEL	UMSEL	UPM1	UPM0	USBS	UCSZ1	UC8Z0	UCPOL	UCSRC
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Keterangan :

1. URSEL (*Register Select*). Karena UCSRC memiliki alamat yang sama dengan UBBRH *bit* ini harus aktis untuk menulis ke UCSRC.
2. UMSEL (USART *Mode Select*). Memilih *mode Asynchronous* (0) atau *Synchronous* (1)
3. UPM1:0 (*Parity Mode*).
4. USBS (*Stop Bit Select*). Memilih ukuran *stop bit* (0 = 1-*bit*, 1 = 2-*bit*).
5. UCPOL (*Clock Polarity*). Mengatur pemicuan pengiriman dan penerimaan data pada *mode synchronous*.

Saluran serial pada mikrokontroler terdiri atas dua kaki yaitu RXD dan TXD, RXD berfungsi untuk menerima data dari komputer/perangkat lainnya, TXD berfungsi untuk mengirim data ke komputer atau perangkat lainnya.

Saluran serial pada komputer atau disebut saluran RS-232 memberikan kemudahan untuk komunikasi antara komputer dengan perangkat lainnya. Tidak seperti saluran paralel yang memiliki tingkat tegangan logika (0 – 5 *Volt*), saluran serial memiliki tegangan positf dan negatif.

Standar komunikasi serial untuk komputer ialah RS-232, RS-232 mempunyai standar tegangan yang berbeda dengan saluran serial mikrokontroler, sehingga agar sesuai dengan RS-232 maka dibutuhkan suatu rangkaian *level conditioner*.

2.3.6.1 Komunikasi Serial

Ada 2 macam cara komunikasi serial, yaitu :

1. Komunikasi serial sinkron atau *Universal Synchronous serial Receiver and Transmitter (USART)* dan
2. Komunikasi serial sinkron atau *Universal Asynchronous serial Receiver and Transmitter (UART)*.

Pada komunikasi data serial sinkron, *clock* dikirimkan bersama sama dengan data serial, tetapi *clock* tersebut dibangkitkan oleh masing-masing sisi pengirim maupun penerima. Sedangkan pada komunikasi serial asinkron tidak diperlukan *clock* karena data dikirimkan dengan kecepatan tertentu yang sama baik pada pengirim dan penerima.

Konektor komunikasi serial ini biasa disebut DB9 yang biasa ada pada bagian belakang komputer atau disebut juga *port* serial RS232



Gambar 2.3 Konektor DB9

Tabel 2.17 Keterangan Sinyal dan Kaki Saluran Serial DB9

Nomor Kaki Konektor DB9	Nama Sinyal	Arah Sinyal
1	DCD (*Data Carrier Detect*)	Dari DCE
2	RD (*Received Data*)	Dari DCE
3	TD (*Transmitted Data*)	Dari DTE
4	DTR(*DTE Ready*)	Dari DTE
5	*Ground*	-
6	DSR (DCE *Ready*)	Dari DCE
7	RTS (*Request to Send*)	Dari DTE
8	CTS (*Clear to Send*)	Dari DCE
9	RI (*Ring Indikator*)	Dari DCE

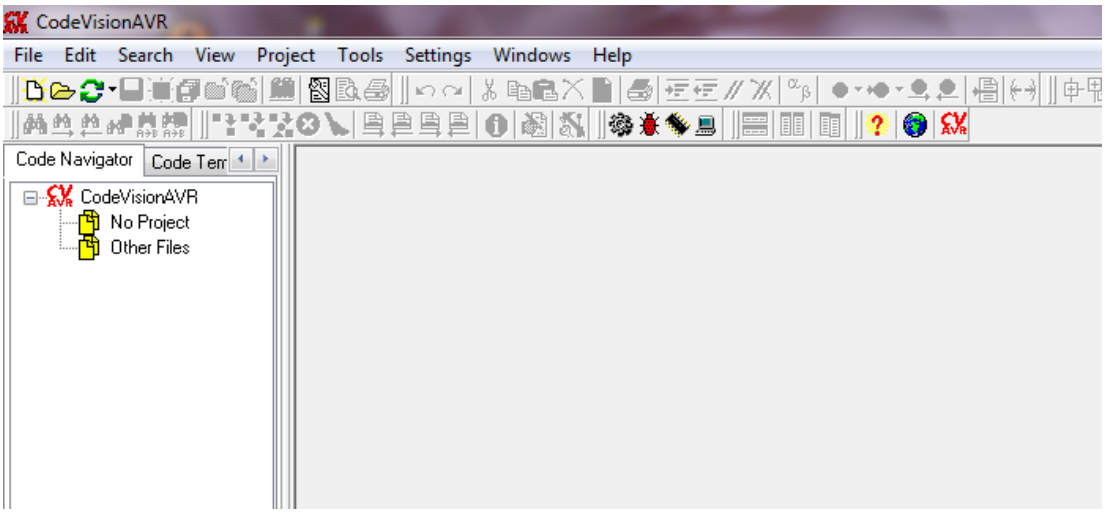
Keterangan mengenai fungsi saluran pada konektor DB9 adalah sebagai berikut:

1. *Data Carrier Detect*, dengan saluran ini DCE memberitahukan ke DTE bahwa pada konektor masukkan ada data masuk.
2. *Receive Data*, digunakan DTE menerima data dari DCE.
3. *Transmitted Data*, digunakan DTE mengirimkan data ke DCE.
4. *Data Terminal Ready*, pada saluran ini DTE memberitahukan kesiapan konektornya.
5. *Signal Ground* merupakan saluran *ground.*
6. DCE *Ready*, sinyal aktif pada saluran ini menunjukkan bahwa DCE sudah siap.
7. *Request To Send*, dengan saluran ini DCE diminta mengirim data oleh DTE.
8. *Clear To Send*, dengan saluran ini DCE memberitahukan bahwa DTE boleh mulai mengirim data.
9. *Ring Indikator*, pada saluran ini DCE memberitahukan ke DTE bahwa sebuah stasiun menghendaki berhubungan dengannya.

2.4 Codevision AVR

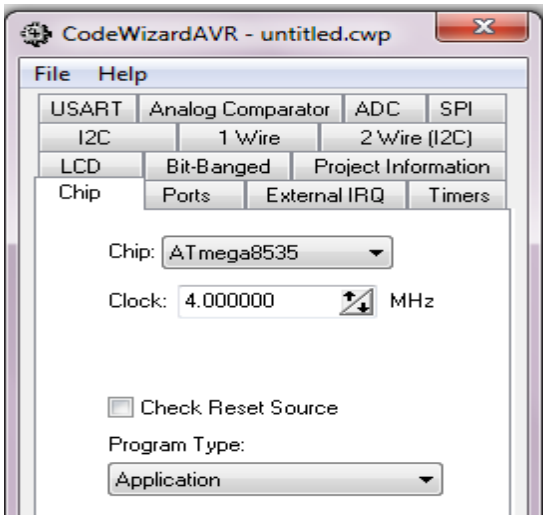
CodevisionAVR merupakan *compiler* bahasa C, didesain khusus untuk mikrokontroler keluarga Atmel AVR. Dapat digunakan pada Windows 98, Me, NT 4, XP, dan Vista. Fungsi utama CodevisionAVR ialah untuk menuliskan program yang akan kita masukkan ke dalam mikrokontroler. Selain dengan CodevisionAVR masih ada jenis lainnya (*compiler* lain) yakni WinAVR.

Selain library standar C, CodeVisionAVR C compiler juga memiliki library lainnya, yaitu *alphanumeric* LCD *modules*, *philips* I2C bus, *national semiconductor* LM75 *temperature sensor*, dan sebagainya.



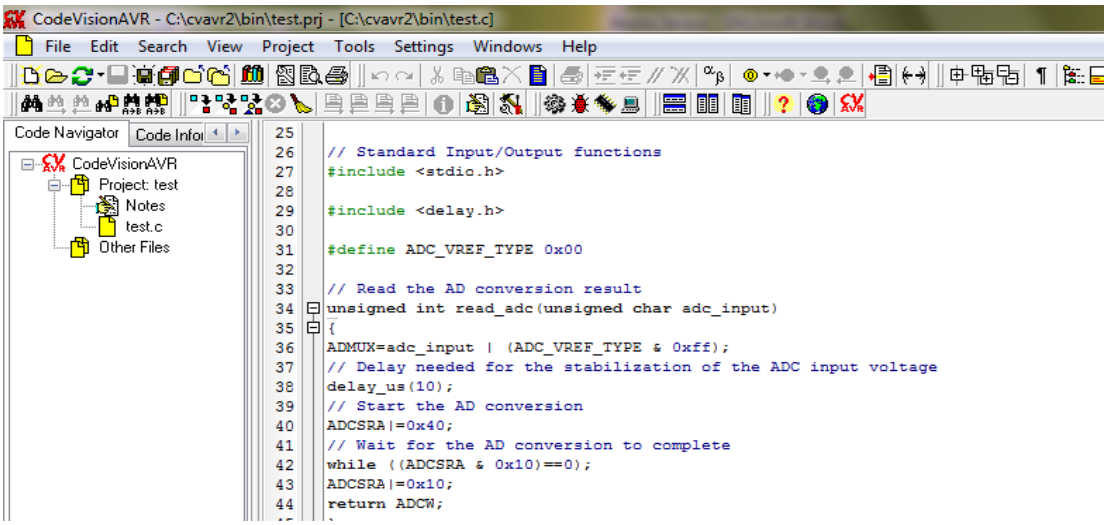
Gambar 2.4 Tampilan Awal CodeVision AVR

CodeVisionAVR memiliki CodeWizardAVR yang memudahkan dalam penulisan program inisialisasi AVR. Dengan cara mengatur konfigurasi dari *Chip*, *PORT*, ADC, USART, *Timer* dan sebagainya.



Gambar 2.5 Tampilan CodeWizard AVR

Setelah mengatur konfigurasi inisialisasi yang akan digunakan, maka secara otomatis program inisilisasi telah tertulis pada CodeVisionAVR.



Gambar 2.6 Program inisialisasi dari konfigurasi CodeVision AVR

Agar program dalam bahasa C dapat ditulis ke dalam ATmega8535, maka terlebih dahulu diubah menjadi *file* yang berekstensi ".hex". CodeVisionAVR telah

menyediakan file yang berekstensi ".hex", walaupun program yang dirancang adalah program yang menggunakan bahasa C.

2.5 Komunikasi Jaringan GSM

Sejarah GSM tidak terlepas dari perkembangan teknologi komunikasi di Eropa. Pada awal tahun 1980, sistem komunikasi selular berkembang dengan pesat dibeberapa negara Eropa. Beberapa diantaranya yaitu sistem C-NET yang dikembangkan di Jerman dan Portugal oleh Siemens, sistem RC-2000 yang dikembangkan di Prancis, sistem NMT yang dikembangkan di Belanda dan Skandinavia oleh Ericsson, serta TACS yang beroprasi di Inggris. Setiap Negara mempunyai sistem komunikasi sendiri yang tidak kompatibel dengan sistem komunikasi di negara lain. Hal ini menyebabkan mobilitas pengguna terbatas pada suatu area sistem teknologi tertentu saja (Fernandez Yogara, 2008).

Pada tahun 1982 dibentuklah sebuah lembaga yang bernama *Groupe Special Mobile* (GSM) yang mengembangkan sistem komunikasi *mobile* untuk masyarakat Eropa secara terpadu. Sistem yang dikembangkan memiliki kualitas suara yang baik, memiliki dukungan jelajah (*roaming*) internasional antar negara Eropa, memiliki dukungan perangkat genggam, dapat dikembangkan untuk layanan fasilitas baru dan memiliki biaya operasional yang murah. Lembaga ini yang mempelopori munculnya teknologi digital selular yang kemudian dikenal dengan nama *Global System for Mobile Communication* (GSM). Lembaga GSM tersebut menyerahkan hasil kerjanya kepada lembaga ETSI (*European Telecommunication Standards Institute*) pada tahun

1989. Hingga saat ini lembaga GSM tersebut lebih populer dengan kepanjangan *Global System for Mobile Communications*.

2.5.1 Komunikasi SMS (*Short Message Sevice*)

Short Message Service (SMS) merupakan salah satu layanan pesan teks yang dikembangkan dan distandarisasi oleh suatu badan yang bernama *European Telecomunication Standards Institute* (ETSI) sebagai bagian dari pengembangan GSM *Phase* 2, yang terdapat pada dokumentasi GSM 03.40 dan GSM 03.38. Fitur SMS ini memungkinkan perangkat Stasiun Seluler Digital (*Digital Cellular Terminal*, seperti *handphone*) untuk dapat mengirim dan menerima pesan-pesan teks dengan panjang sampai dengan 160 karakter melalui jaringan GSM. (ETSI, 1996)SMS dapat dikirimkan ke perangkat Stasiun Seluler Digital lainnya hanya dalam beberapa detik selama berada pada jangkauan pelayanan GSM. Lebih dari sekedar pengiriman pesan biasa, layanan SMS memberikan garansi SMS akan sampai pada tujuan meskipun perangkat yang dituju sedang tidak aktif yang dapat disebabkan karena sedang dalam kondisi mati atau berada di luar jangkauan layanan GSM.

Jaringan SMS akan menyimpan sementara pesan yang belum terkirim, dan akan segera mengirimkan ke perangkat yang dituju setelah adanya tanda kehadiran dari perangkat dijaringan tersebut. Dengan fakta bahwa layanan SMS (melalui jaringan GSM) mendukung jangkauan/jelajah nasional dan internasional dengan waktu keterlambatan yang sangat kecil, memungkinkan layanan SMS cocok untuk dikembangkan sebagai aplikasi-aplikasi seperti: *pager*, *e-mail*, dan notifikasi *voice mail*, serta layanan pesan banyak pemakai (*multiple users)*. Namun pengembangan

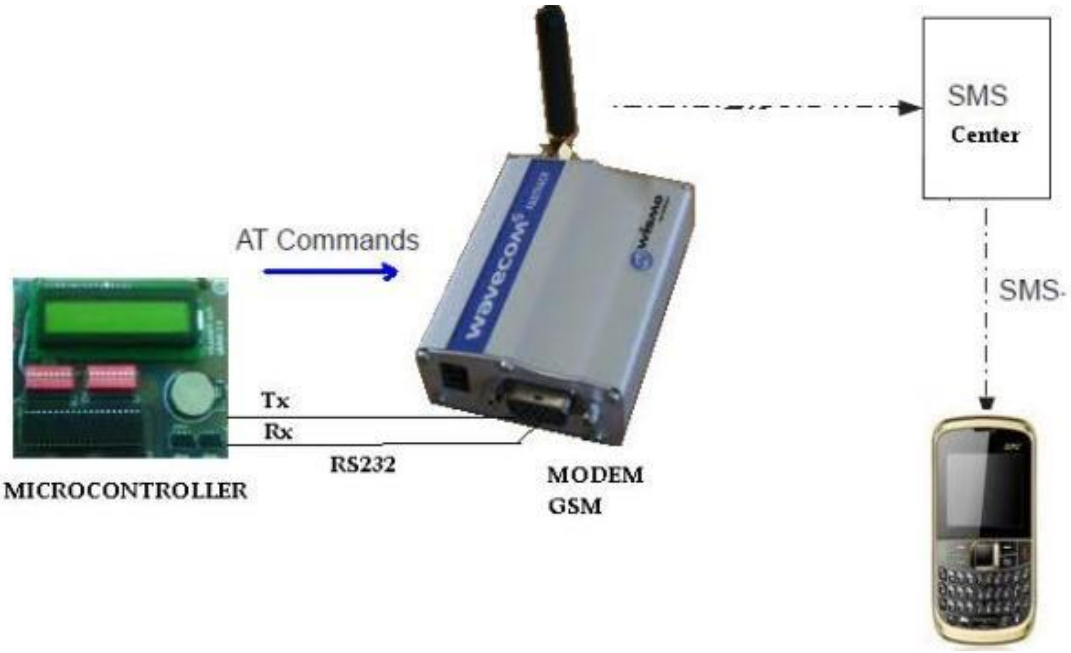
aplikasi tersebut masih bergantung pada tingkat layanan yang disediakan oleh operator jaringan.

2.5.2 Modem GSM

Kebanyakan modem yang digunakan di PC atau laptop dewasa ini adalah dengan menggunakan teknik asynchronous. Asynchronous ini maksudnya bahwa ketika modem ini mengirimkan data tanpa menggunakan clock untuk menyinkronisasikan kegiatan dari kedua sistem yang terhubung. Data dikirim dalam 1 byte yang berada dalam sebuah frame pada satu waktu. Frame tersebut berisikan sebuah start bit, data, dan biasanya satu atau lebih stop bit. Start dan stop bit inilah yang memberitahukan kapan dan dimana data tersebut. Karena fungsi inilah, si penerima akan tahu mana yang data dan mana yang noise, sehingga dapat diketahui mana yang dapat diterima atau tidak.

Modem ini juga bisa menggunakan parity sebagai error detection. Ada dua parity yang digunakan, odd dan even. Jenis modem yang menggunakan parity ini sudah jarang digunakan pada masa sekarang ini. StandarISASI Sistem Transmisi Untuk standarisasi sistem transmisi dari modem, maka dua badan dunia yaitu CCITT (Committee Consultative International Telegraphique et Telephonique) dan ITU (International Telecommunication Union), mengeluarkan sebuah standar yang dinamakan V-dot. Standar ini berhubungan dengan kecepatan kerja modem, tipe kompresi data dan penanganan kesalahan data. Misalnya, V22bis mengacu pada kecepatan modem 2,4 Kbps, V.32 yang diperkenalkan tahun 1984 mengacu pada

kecepatan 9.6 Kbps, V.32bis pada tahun 1991 dengan kecepatan 14.4 Kbps. Standar V.34 pada tahun 1994 memperkenalkan kecepatan 28.8 Kbps, yang pada tahun 1996 diperbaharui dengan V.34+ dengan kecepatan 33.6 Kbps. Kemudian pada tahun 1998 dikembangkan V.90 dari ITU yang mempunyai kecepatan 42 kbps dan disempurnakan kecepatannya menjadi 55.6 Kbps. Penting dicacat bahwa batasan tersebut tidak berhubungan dengan kabel tembaga pada jaringan telepon umum, tetapi berhubungan dengan converter analog ke digital yang dipasang pada jaringan tersebut. Setiap kanal line telepon memiliki 4KHz bandwidth analog, yang setara dengan 64K bit/sec bandwidth digital. Dalam kondisi baik, sebuah line telepon, secara teori, mendukung 64K bit/sec. Prinsip kerja dari Modem GSM digambarkan pada gambar 2.7 :



Gambar 2.7 Prinsip Kerja Modem GSM

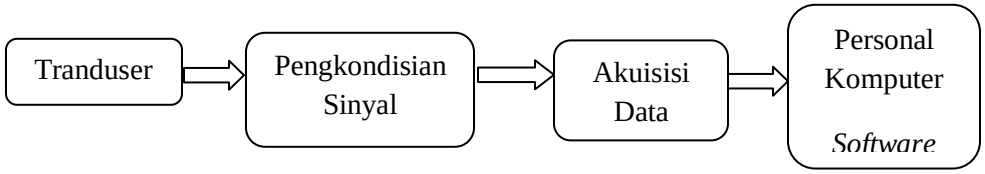
2.6 Akuisisi Data

Akuisisi data adalah suatu proses pengambilan data sampel dari fenomena fisik (suhu, tekanan dan lain-lain) dan mengkonversikan data sampel yang didapat menjadi nilai numerik yang dapat dimanipulasi oleh sebuah komputer.

Sistem akuisisi data adalah kumpulan dari berbagai elemen-elemen elektronik yang saling bekerja sama dengan tujuan melakukan pengumpulan, penyimpanan, pengolahan data, dan penyaluran data untuk dijadikan sebagai suatu bentuk informasi yang berarti.

Elemen-elemen dasar dari sistem akuisisi data berbasis komputer (PC), sebagaimana ditunjukkan pada Gambar 2.6, antara lain :

- Tranduser,
- Pengkondisian sinyal
- Perangkat keras (*hardware*) akuisisi data,
- Sebuah komputer PC, dan
- Perangkat lunak (*software*) yang terkait



Gambar 2.8 Elemen – elemen Sistem Akuisisi Data Berbasis PC