MENGENAL MIKROKONTROLER AVR ATMega16

Mokh. Sholihul Hadi
m_sholihul_hadi@yahoo.com

AVR merupakan seri mikrokontroler CMOS 8-bit buatan Atmel, berbasis arsitektur *RISC (Reduced Instruction Set Computer*). Hampir semua instruksi dieksekusi dalam satu siklus *clock*. AVR mempunyai 32 register general-purpose, timer/counter fleksibel dengan mode *compare*, *interrupt internal* dan *eksternal*, serial UART, *programmable Watchdog Timer*, dan *mode power saving*, ADC dan PWM internal. AVR juga mempunyai In-*System Programmable Flash on-chip* yang mengijinkan memori program untuk diprogram ulang dalam sistem menggunakan hubungan serial SPI. ATMega16.

ATMega16 mempunyai *throughput* mendekati 1 MIPS per MHz membuat disainer sistem untuk mengoptimasi konsumsi daya versus kecepatan proses.

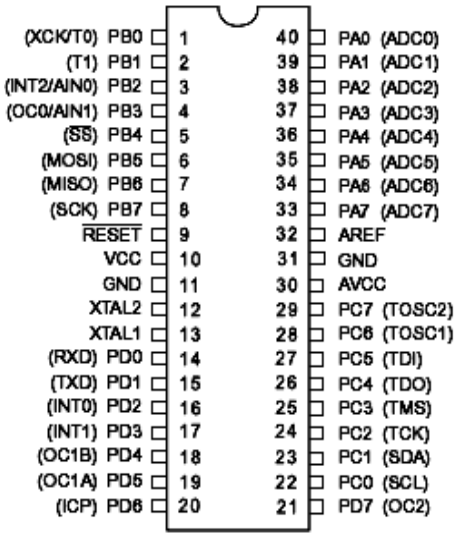
Beberapa keistimewaan dari AVR ATMega16 antara lain:

1. *Advanced RISC Architecture*
   - *130 Powerful Instructions – Most Single Clock Cycle Execution*
   - *32 x 8 General Purpose Fully Static Operation*
   - *Up to 16 MIPS Throughput at 16 MHz*
   - *On-chip 2-cycle Multiplier*
2. *Nonvolatile Program and Data Memories*
   - *8K Bytes of In-System Self-Programmable Flash*
   - *Optional Boot Code Section with Independent Lock Bits*
   - *512 Bytes EEPROM*
   - *512 Bytes Internal SRAM*
   - *Programming Lock for Software Security*

3. *Peripheral Features*
   - *Two 8-bit Timer/Counters with Separate Prescalers and Compare Mode*
   - *Two 8-bit Timer/Counters with Separate Prescalers and Compare Modes*

- *One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode*
- *Real Time Counter with Separate Oscillator*
- *Four PWM Channels*
- *8-channel, 10-bit ADC*
- *Byte-oriented Two-wire Serial Interface*
- *Programmable Serial USART*

4. *Special Microcontroller Features*
   - *Power-on Reset and Programmable Brown-out Detection*
   - *Internal Calibrated RC Oscillator*
   - *External and Internal Interrupt Sources*
   - *Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby and Extended Standby*
5. *I/O and Package*
   - *32 Programmable I/O Lines*
   - *40-pin PDIP, 44-lead TQFP, 44-lead PLCC, and 44-pad MLF*
6. *Operating Voltages*
   - *2.7 - 5.5V for Atmega16L*
   - *4.5 - 5.5V for Atmega16*



Gambar 1 Pin-pin ATMega16 kemasan 40-pin

Pin-pin pada ATMega16 dengan kemasan 40-pin DIP (*dual in-line package*) ditunjukkan oleh gambar 1. Guna memaksimalkan performa, AVR menggunakan arsitektur *Harvard* (dengan memori dan bus terpisah untuk program dan data).

Port sebagai input/output digital

ATMega16 mempunyai empat buah port yang bernama *PortA, PortB, PortC, dan PortD*. Keempat port tersebut merupakan jalur *bi-directional* dengan pilihan *internal pull-up*. Tiap port mempunyai tiga buah register bit, yaitu DDxn, PORTxn, dan PINxn. Huruf 'x'mewakili nama huruf dari port sedangkan huruf 'n' mewakili nomor bit. Bit DDxn terdapat pada I/O address DDRx, bit PORTxn terdapat pada

I/O address PORTx, dan bit PINxn terdapat pada I/O address PINx. Bit DDxn dalam register DDRx (*Data Direction Register*) menentukan arah pin. Bila DDxn diset 1 maka Px berfungsi sebagai pin output. Bila DDxn diset 0 maka Px berfungsi sebagai pin input.Bila PORTxn diset 1 pada saat pin terkonfigurasi sebagai pin input, maka resistor *pull-up* akan diaktifkan. Untuk mematikan resistor *pull-up*, PORTxn harus diset 0 atau pin dikonfigurasi sebagai pin output. Pin port adalah *tri-state* setelah kondisi reset. Bila PORTxn diset 1 pada saat pin terkonfigurasi sebagai pin output maka pin port akan berlogika 1. Dan bila PORTxn diset 0 pada saat pin terkonfigurasi sebagai pin output maka pin port akan berlogika 0. Saat mengubah kondisi port dari kondisi *tri-state* (DDxn=0, PORTxn=0) ke kondisi *output high* (DDxn=1, PORTxn=1) maka harus ada kondisi peralihan apakah itu kondisi *pull-up enabled* (DDxn=0, PORTxn=1) atau kondisi *output low* (DDxn=1, PORTxn=0).

Biasanya, kondisi pull-up enabled dapat diterima sepenuhnya, selama lingkungan impedansi tinggi tidak memperhatikan perbedaan antara sebuah *strong high driver* dengan sebuah *pull-up*. Jika ini bukan suatu masalah, maka bit PUD pada register SFIOR dapat diset 1 untuk mematikan semua *pull-up* dalam semua port. Peralihan dari kondisi *input dengan pull-up* ke kondisi *output low* juga menimbulkan masalah yang sama. Kita harus menggunakan kondisi *tri-state* (DDxn=0, PORTxn=0) atau kondisi output high (DDxn=1, PORTxn=0) sebagai kondisi transisi.

Tabel 1 Konfigurasi pin port

DDxn	PORTxn	PUD (In SFIOR)	I/O	Pull-up	Comment
0	0	X	Input	No	Tri-state (Hi-Z)
0	1	0	Input	Yes	Pxn will source current if ext. pulled low.
0	1	1	Input	No	Tri-state (Hi-Z)
1	0	X	Output	No	Output Low (Sink)
1	1	X	Output	No	Output High (Source)

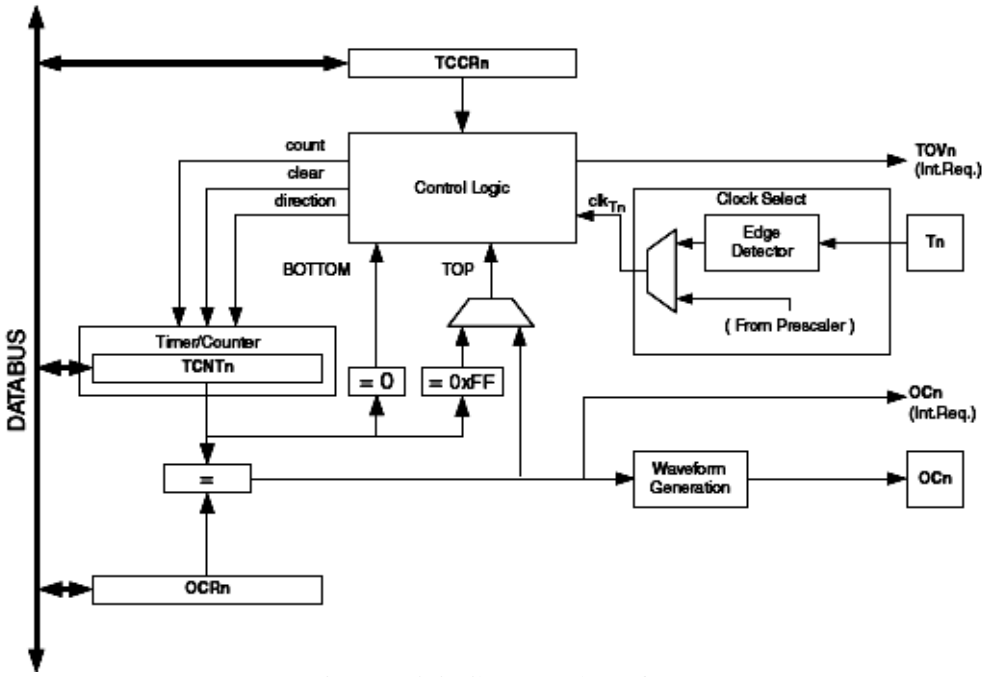
Bit 2 – PUD : *Pull-up Disable*
Bila bit diset bernilai 1 maka *pull-up* pada port I/O akan dimatikan walaupun *register* DDxn dan PORTxn dikonfigurasikan untuk menyalakan *pull-up* (DDxn=0, PORTxn=1).

Timer

Timer/counter adalah fasilitas dari ATMega16 yang digunakan untuk perhitungan pewaktuan. Beberapa fasilitas *chanel* dari timer counter antara lain: *counter channel* tunggal, pengosongan data timer sesuai dengan data pembanding, bebas -*glitch*, tahap yang tepat *Pulse Width Modulation (PWM),* pembangkit frekuensi, *event counter external.*.

Gambaran Umum

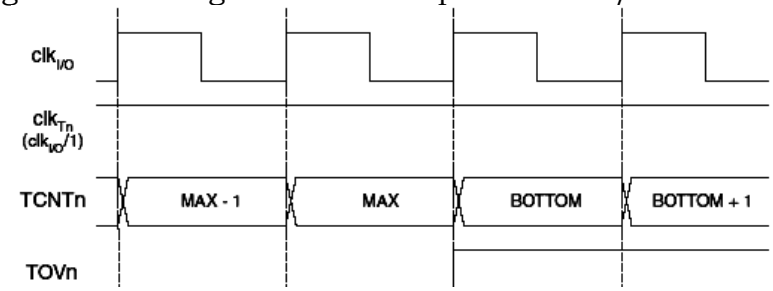
Gambar diagram *block* timer/counter 8 bit ditunjukan pada gambar 2. Untuk penempatan pin I/O telah di jelaskan pada bagian I/O di atas. CPU dapat diakses register I/O, termasuk dalam pin-pin I/O dan bit I/O. *Device* khusus register I/O dan lokasi bit terdaftar pada deskripsi timer/counter 8 bit.



Gambar 2 Blok diagram timer/counter

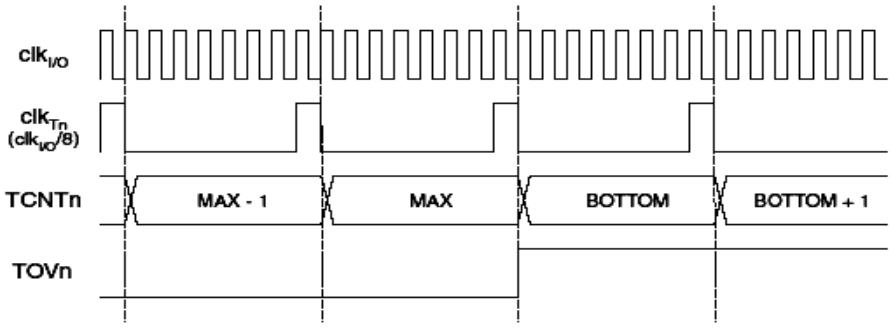
Timing Diagram Timer/Counter

Timer/counter didesain sinkron *clock* timer (clkT0) oleh karena itu ditunjukkan sebagai sinyal *enable* clock pada gambar 3. Gambar ini termasuk informasi ketika *flag interrupt* dalam kondisi set. Data timing digunakan sebagai dasar dari operasi timer/counter.



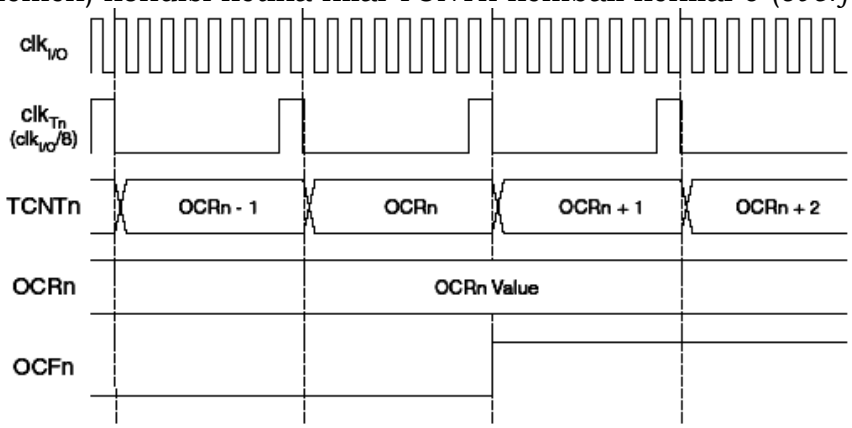
Gambar 3 Timing diagram timer/counter, tanpa prescaling

Sesuai dengan gambar 4 timing diagram timer/counter dengan *prescaling* maksudnya adalah counter akan menambahkan data counter (TCNTn) ketika terjadi pulsa *clock* telah mencapai 8 kali pulsa dan sinyal *clock* pembagi aktif *clock* dan ketika telah mencapai nilai maksimal maka nilai TCNTn akan kembali ke nol. Dan kondisi *flag timer* akan aktif ketika TCNTn maksimal.



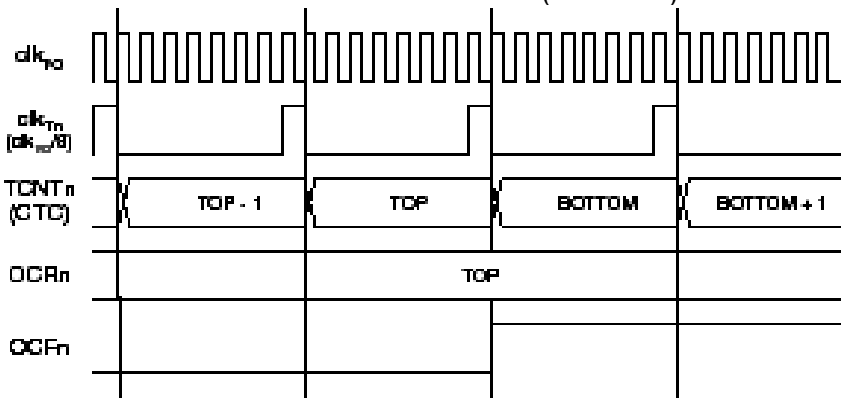
Gambar 4 Timing diagram timer/counter, dengan prescaling

Sama halnya timing timer diatas, timing timer/counter dengan seting OCFO timer mode ini memasukan data ORCn sebagai data input timer. Ketika nilai ORCn sama dengan nilaiTCNTn maka pulsa *flag timer* akan aktif. TCNTn akan bertambah nilainya ketika pulsa *clock* telah mencapai 8 pulsa. Dan kondisi *flag* akan berbalik (komplemen) kondisi ketika nilai TCNTn kembali kenilai 0 (*overflow*).



Gambar 5 Timing diagram timer/counter, menyeting OCFO, dengan pescaler (fclk_I/O/8)

Ketika nilai ORCn sama dengan nilai TCNTn maka pulsa *flag timer* akan aktif. TCNTn akan bertambah nilainya ketika pulsa *clock* telah mencapai 8 pulsa. Dan kondisi *flag* akan berbalik (komplemen) kondisi ketika nilai TCNTn kembali kenilai 0 (overflow).



Gambar 6 Timing diagram timer/counter, menyeting OCFO, pengosongan data timer sesuai dengan data pembanding,dengan pescaler (fclk_I/O/8)

Deskripsi Register Timer/Counter 8 bit

Bit	7	6	5	4	3	2	1	0	
	FOC0	WGM00	COM01	COM00	WGM01	CS02	CS01	CS00	TCCR0
Read/Write	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	

Gambar 7 Regiter timer counter 8 bit

Bit 7 – FOCO : perbandingan kemampuan output

FOCO hanya akan aktif ketika spesifik-spesifik bit WGM00 tanpa PWM mode. Adapun untuk meyakinkan terhadap kesesuaian dengan *device-device* yang akan digunakan, bit ini harus diset nol ketika TCCRO ditulisi saat mengoperasikan mode PWM. Ketika menulisi logika satu ke bit FOCO, dengan segera dipaksakan untuk disesuaikan pada unit pembangkit bentuk gelombang. Output OCO diubah disesuaikan pda COM01: bit 0 menentukan pengaruh daya pembanding.

Bit 6,3 – WGM01:0: Waveform Generation Mode

Bit ini mengontrol penghitungan yang teratur pada counter, sumber untuk harga counter maksimal (TOP)., dan tipe apa dari pembangkit bentuk gelombang yang digunakan. Mode-mode operasi didukung oleh unit timer/counter sebagai berikut : mode normal, pembersih timer pada mode penyesuaian dengan pembanding (CTC), dan dua tipe mode *Pulse Width Modulation (PWM)*.

Tabel 2 Deskripsi Bit Mode Pembangkit Bentuk Gelombang

Mode	WGM01 (CTC0)	WGM00 (PWM0)	Timer/Counter Mode of Operation	TOP	Update of OCR0	TOV0 Flag Set-on
0	0	0	Normal	0xFF	Immediate	MAX
1	0	1	PWM, Phase Correct	0xFF	TOP	BOTTOM
2	1	0	CTC	OCR0	Immediate	MAX
3	1	1	Fast PWM	0xFF	TOP	MAX

catatan: definisi nama-nama bit CTC0 dan PWM0 sekarang tidak digunakan lagi. Gunakan WGM 01: 0 definisi. Bagaimanapun lokasi dan fungsional dan lokasi dari masing-masing bit sesuai dengan versi timer sebelumnya.

Bit 5:4 – COMO1:0 Penyesuaian Pembanding Mode Output

Bit ini mengontrol pin output *compare* (OCO), jika satu atau kedua bit COM01:0 diset, output OC0 melebihi fungsional port normal I/O dan keduanya terhubung juga. Bagaimanapun, catatan bahwa bit *Direksi Data Register* (DDR) mencocokan ke pin OC0 yang mana harus diset dengan tujuan mengaktifkan. Ketika OC0 dihubungkan ke pin, fungsi dari bit COM01:0 tergantung dari pengesetan bit WGM01:0. Tabel di bawah menunjukkan COM fungsional ketika bit-bt WGM01:0 diset ke normal atau mode CTC (non PWM).

Tabel 3 Mode Output Pembanding, tanpa PWM

COM01	COM00	Description
0	0	Normal port operation, OC0 disconnected.
0	1	Toggle OC0 on compare match
1	0	Clear OC0 on compare match
1	1	Set OC0 on compare match

Tabel 4 menunjukan bit COM01:0 fungsional ketika bit WGM01:0 diset ke mode fast PWM.

Tabel 4 Mode Output Pembanding, Mode fast PWM

COM01	COM00	Description
0	0	Normal port operation, OC0 disconnected.
0	1	Reserved
1	0	Clear OC0 on compare match, set OC0 at TOP
1	1	Set OC0 on compare match, clear OC0 at TOP

Tabel 5 menunjukan bit COM01:0 fungsional ketika bit WGM01:0 diset ke *mode phase correct* PWM.

Tabel 5 Mode Output Pembanding, Mode phase correct PWM

COM01	COM00	Description
0	0	Normal port operation, OC0 disconnected.
0	1	Reserved
1	0	Clear OC0 on compare match when up-counting. Set OC0 on compare match when downcounting.
1	1	Set OC0 on compare match when up-counting. Clear OC0 on compare match when downcounting.

Bit 2:0 – CS02:0 : Clock Select

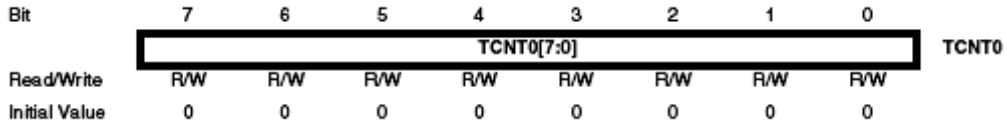
Tiga bit *clock select* sumber *clock* digunakan dengan timer/counter. Jika mode pin *eksternal* digunakan untuk timer counter0, perpindahan dari pin T0 akan memberi clock counter.

Tabel 6 Deskripsi bit clock select

CS02	CS01	CS00	Description
0	0	0	No clock source (Timer/Counter stopped).
0	0	1	$clk_{I/O}$/(No prescaling)
0	1	0	$clk_{I/O}$/8 (From prescaler)
0	1	1	$clk_{I/O}$/32 (From prescaler)
1	0	0	$clk_{I/O}$/256 (From prescaler)
1	0	1	$clk_{I/O}$/1024 (From prescaler)
1	1	0	External clock source on T0 pin. Clock on falling edge.
1	1	1	External clock source on T0 pin. Clock on rising edge.

Sesuai dengan tabel diatas maka sumber *clock* dapat dibagi sehingga timer/counter dapat disesuaikan dengan banyak data yang dihitung.

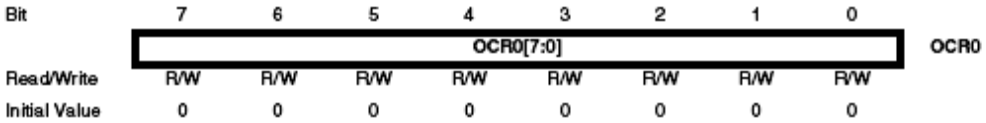
Register Timer/Counter TCNT0



Gambar 8 Register timer TCNT0

Register timer/counter memberikan akses secara langsung, keduanya digunakan untuk membaca dan menulis operasi, untuk penghitung unit 8-bit timer/counter. Menulis ke blok-blok register TCNT0 (*removes*) disesuaikan dengan clock timer berikutnya. Memodifikasi counter (TCNT0) ketika perhitungan berjalan, memperkenalkan resiko kehilangan perbandingan antara TCNC0 dengan register OCR0.

Register Timer/Counter OCR0



Gambar 9. Register timer OCR0

Register output pembanding berisi sebuah haraga 8 bit yang mana secara terus-menerus dibandingkan dengan harga counter (TCNT0). Sebuah penyesuaian dapat digunakan untuk membangkitkan output *interrupt* pembanding, atau untuk membangkitkan sebuah output bentuk gelombang pada pin OC0.

Register Timer/Counter *Interrupt Mask*

Bit 1-OCIE0: output timer counter menyesuaikan dengan kesesuaian interrupt yang aktif.

Ketika bit OCIE0 ditulis satu, dan 1-bit pada register status dalam kondisi set (satu), membandingkan timer/counter pada *interrupt* yang sesuai diaktifkan. Mencocokkan *interrupt* yang dijalankan kesesuaian pembanding pada timer/counter0 terjadi, ketika bit OCF0 diset pada *register* penanda timer/counter-TIFR.

Bit 0 – TOIE0: Timer/Counter 0 Overflow Interrupt Enable

Ketika bit TOIE0 ditulis satu, dan 1-bit pada *register* status dalam kondisi set (satu), timer/counter melebihi *interrupt* diaktifkan. Mencocokkan *interrupt* dijalankan jika kelebihan pada timer/counter0 terjadi, ketika bit TOV0 diset pada *register* penanda timer/counter-TIFR

Register Timer/Counter Register - TIFR

Bit	7	6	5	4	3	2	1	0	
	OCF2	TOV2	ICF1	OCF1A	OCF1B	TOV1	OCF0	TOV0	TIFR
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	

Gambar 10 Register timer TIFR

Bit 1 – OCF0: Output Compare Flag 0

OCF0 dalam kondisi set (satu) kesesuaian pembanding terjadi antara timer/counter dan data pada OCRO – Register 0 keluaran pembanding. OCF0 diclear oleh *hardware* ketika eksekusi pencocokan penanganan *vector interrupt*. Dengan alternatif mengclearkan OCF0 dengan menuliskan logika satu pada flag. Ketika I-bit pada SREG, OCIE0 (Timer/Counter0 penyesuaian pembanding *interrupt enable*), dan OCF0 diset (satu), timer/counter pembanding kesesuaian *interrupt* dijalankan.

Bit 0 – TOV0: Timer/Counter Overflow Flag

Bit TOV0 diset (satu) ketika kelebihan terjadi pada timer/counter0. TOV0 diclearkan dengan *hardware* ketika penjalanan pencocokan penanganan *vector interrupt*. Dengan alternatif, TOV0 diclearkan dengan jalan memberikan logika satu pada flag. Ketika I-bit pada SREG, TOIE0 (Timer/Counter0 *overflow interrupt enable*), dan TOV0 diset (satu), timer/counter *overflow* interrupt dijalankan. Pada tahap mode PWM yang tepat, bit ini di set ketika timer/counter merubah bagian perhitungan pada $00.

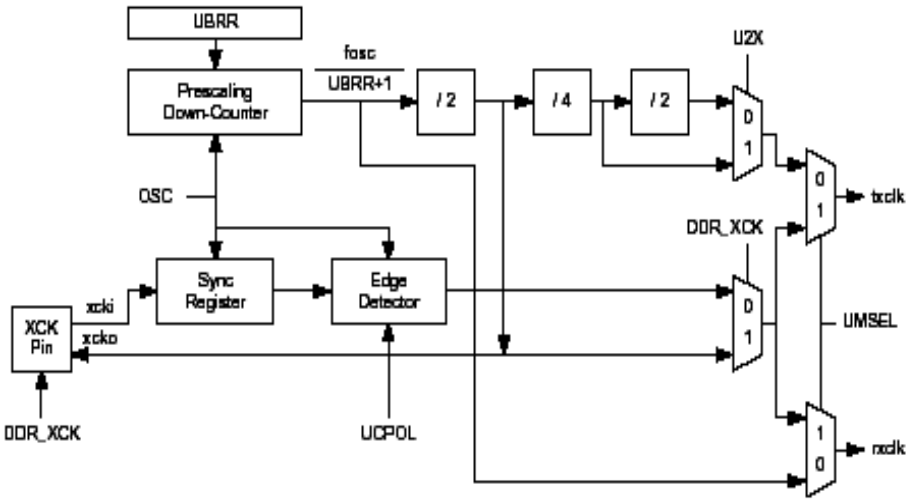
Serial pada ATMega16

Universal synchronous dan *asynchronous* pemancar dan penerima serial adalah suatu alat komunikasi serial sangat fleksibel. Jenis yang utama adalah :

a) Operasi *full duplex* (register penerima dan pengirim serial dapat berdiri sendiri)
b) Operasi *Asychronous* atau *synchronous*
c) *Master* atau *slave* mendapat clock dengan operasi *synchronous*
d) Pembangkit *baud rate* dengan resolusi tinggi
e) Dukung *frames serial* dengan 5, 6, 7, 8 atau 9 Data bit dan 1 atau 2 Stop bit
f) Tahap *odd* atau *even parity* dan *parity check* didukung oleh *hardware*
g) Pendeteksian data *overrun*
h) Pendeteksi *framing error*
i) Pemfilteran gangguan (noise) meliputi pendeteksian bit *false start* dan pendeteksian *low pass filter* digital
j) Tiga *interrupt* terdiri dari TX complete, TX data *register empty* dan RX complete.
k) Mode komunikasi multi-processor
l) Mode komunikasi *double speed asynchronous*

Generator Clock

Logic generator *clock* menghasilkan dasar *clock* untuk pengirim dan penerima. USART mendukung empat mode operasi clock: Normal *Asynchronous*, *Double Speed Asynchronous* mode *Master Synchronous* dan *Slave Synchronous*. Bit UMSEL pada USART control dan status register C (UCSRC) memilih antara operasi *Asychronous* dan *Synchronous*. *Double speed* (hanya pada *mode Asynchronou*) dikontrol oleh U2X yang mana terdapat pada register UCSRA. Ketika mengunakan mode operasi *synchronous* (UMSEL = 1) dan data direction register untuk pin XCk (DDR_XCK) mengendalikan apakah sumber clock tersebut adalah *internal (master mode*) atau *eksternal (slave mode*) pin-pin XCK hanya akan aktif ketika menggunakan mode *Synchronous*.



Gambar 11 Blok diagram clock generator logic

Keterangan sinyal :

txclk : clock pengirim (*internal clock*)
rxclk : clock dasar penerima *(internal clock*)
xcki : input dari pin XCK (sinyal *internal*). Digunakan untuk operasi *slave synchronous*.
xcko : clock output ke pin XCK (sinyal *internal*). Digunakan untuk operasi *master synchronous*
fosc : frekuensi pin XTAL (system clock)

Generator Internal Clock – Pembangkit Baud rate

Generasi *internal clock* digunakan untuk mode – mode operasi master *asynchronous* dan *synchronous*. Register USART *baud rate* (UBRR) dan *down-counter* dikoneksikan kepada fungsinya sebagai *programmable prescaler* atau pembangkit *baud rate*. *Down-counter*, dijalankan pada *system clock* (*fosc*), dibebani dengan nilai UBRR setiap counter telah dihitung mundur ke nol atau ketika register UBRRL ditulisi. Clock dibangkitkan setiap counter mencapai nol. Clock ini adalah pembangkit *baud rate clock* output (fosc/(UBBR+1)). Pemancar membagi baud rete generator clock output dengan 2, 8,

atau 16 cara tergantung pada mode. Pembangkit output *baud rate* digunakan secara langsung oleh penerima clock dan unit-unit pelindung data. Unit-unit *recovery* menggunakan suatu mesin status yang menggunakan 2, 8, atau 16 cara yang tergantung pada cara menyimpan status dari UMSEL, bit-bit U2X dan DDR_XCK.

Tabel di bawah menunjukan penyamaan perhitungan *baud rate* dan nilai UBRR tiap mode operasi mengunakan sumber pembangkit clock *internal*.

Tabel 7 Persamaan untuk menyeting perhitungan register baud rate

Operating Mode	Equation for Calculating Baud Rate(1)	Equation for Calculating UBRR Value
Asynchronous Normal Mode (U2X = 0)	$BAUD = \frac{f_{osc}}{16(UBRR+1)}$	$UBRR = \frac{f_{osc}}{16BAUD} - 1$
Asynchronous Double Speed Mode (U2X = 1)	$BAUD = \frac{f_{osc}}{8(UBRR+1)}$	$UBRR = \frac{f_{osc}}{8BAUD} - 1$
Synchronous Master Mode	$BAUD = \frac{f_{osc}}{2(UBRR+1)}$	$UBRR = \frac{f_{osc}}{2BAUD} - 1$

note: *baud rate* menunjukan pengiriman rate bit tiap detik (bps)

BAUD :*baud rate* (pada bit-bit per detik,bps) fosc frekuensi sistem *clock* osilator
UBRR: terdiri dari UBRRH dan UBBRL,(0-4095)

Eksternal Clock

Eksternal clock digunakan untuk operasi mode *slave synchronous*. *Eksternal clock* masuk dari pin XCK dicontohkan oleh suatu daftar sinkronisasi register untuk memperkecil kesempatan meta-stabilitas. Keluaran dari sinkronisasi register kemudian harus menerobos detector tepi sebelum digunakan oleh pengirim dan penerima.

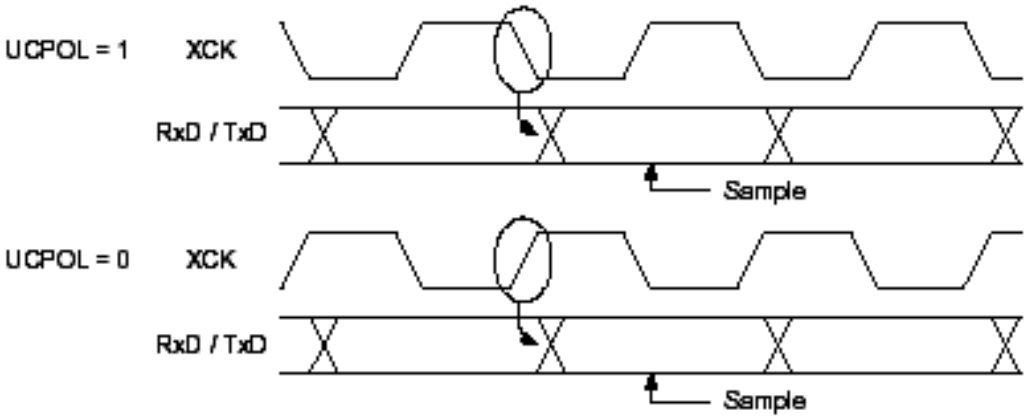
Proses ini mengenalkan dua *period delay clock* CPU dan oleh karena itu maksimal frekuensi clock XCK *eksternal* dibatasi oleh persamaan sebagai berikut

Fxck < fosc/4

Keterangan: *fosc* tergantung pada stabilitas sistem sumber *clock*.

Operasi *Synchronous Clock*

Ketika mode sinkron digunakan (UMSEL=1), pin XCK akan digunakan sama seperti clock input (*slave*) atau clock output (*master*). Dengan ketergantungan antara tepi *clock* dan data sampling atau perubahan data menjadi sama. Prinsip dasarnya adalah data input (on RxD) dicontohkan pada *clock* XCK berlawanan dari tepi data output (TxD) sehingga mengalami perubahan.



Gambar 12 Operasi synchronous Clock

UCPOL bit UCRSC memilih tepi yang mana clock XCK digunakan untuk data sampling dan yang mana digunakan untuk perubahan data. Seperti yang ditunjukan pada gambar di atas, ketika UCPOL nol data akan diubah pada tepi kenaikan XCK dan dicontohkan pada tepi XCK saat jatuh. Jika UCPOL dalam kondisi set, data akan mengalami perubahan pada saat tepi XCK jatuh dan data akan dicontohkan pada saat tepi XCK naik.

Inisialisasi USART

USART harus diinisialisasi sebelum komunikasi manapun dapat berlangsung. Proses inisialisasi normalnya terdiri dari pengesetan *baud rate*, penyetingan *frame* format dan pengaktifan pengirim atau penerima tergantung pada pemakaian. Untuk interrupt menjalankan operasi USART, global *interrupt flag* (penanda) sebaiknya dibersihkan (dan *interrupt global disable*) ketika inisialisasi dilakukan.

Sebelum melakukan inisialisasi ulang dengan mengubah *baud rate* atau *frame* format, untuk meyakinkan bahwa tidak ada transmisi berkelanjutan sepanjang periode *register* yang diubah. Flag TXC dapat digunakan untuk mengecek bahwa pemancar telah melengkapi semua pengiriman, dan flag RXC dapat digunakan untuk mengecek bahwa tidak ada data yang tidak terbaca pada buffer penerima. Tercatat bahwa flag TXC harus dibersihkan sebelum tiap transmisi (sebelum UDR ditulisi) jika itu semua digunakan untuk tujuan tersebut.

REFERENSI

www.atmel.com.*Datasheet AVR ATMega16*

Biografi Penulis

Mokh. Sholihul Hadi, lahir di Jombang 25 Mei 1982. Menyelesaikan pendidikan Sarjana di Jurusan Teknik Elektro Universitas Brawijaya Malang tahun 2004. Terhitung sejak tahun yang sama mengabdikan diri menjadi PNS sebagai dosen di Jurusan Teknik Elektro Universitas Negeri Malang. Beberapa penghargaan yang pernah diperoleh antara lain: Nominator Peneliti Muda terbaik tingkat Nasional bidang Ilmu Pengetahuan Teknik dan Rekayasa oleh LIPI tahun 2006, Medali Emas Pekan Ilmiah Mahasiswa Nasional tahun 2004, Medali Emas sebagai Peneliti Remaja terbaik tingkat Nasional bidang Ilmu Pengetahuan Teknik dan Rekayasa oleh LIPI tahun 2004, Medali Perak Lomba Karya Ilmiah Olahraga Mahasiswa Indonesia Direktorat Jenderal Olahraga Republik Indonesia tahun 2004, Mahasiswa Teladan Universitas Brawijaya Malang tahun 2004 dll. Bidang penelitian yang sedang ditekuni antara lain: Robotika, Elektronika Medis, *Artificial Intelligent*, dan *Nano Technology*.