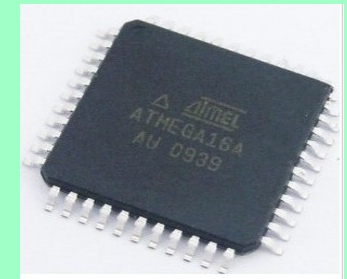


**MIKROKONTROLER ATMEGA
BERBASIS
CODEVISION AVR
(Pengenalan Timer)**

Mikrokontroler ATmega16A

- High-performance, Low-power Atmel AVR **8-bit Microcontroller**
- Advanced RISC Architecture. Up to 16MIPS Throughput at 16MHz
- High Endurance Non-volatile Memory segments
 - ▣ 16KBytes of In-System Self-programmable Flash program memory
 - ▣ 512Bytes EEPROM
 - ▣ 1KByte Internal SRAM
- Peripheral Features
 - ▣ **Timers: Two 8-bit Timer/Counter and One 16-bit Timer/Counter**
 - ▣ 8-channel, 10-bit ADC
 - ▣ Serial Communication: Programmable Serial USART, Master/Slave SPI Serial Interface, Byte-Oriented Two Wire Interface.
- I/O and Packages: 32 I/O Lines. 40-pin PDIP, 44-lead TQFP, and 44-pad QFN/MLF
- Operating Voltages: 2.7 - 5.5V
- Speed Grades: 0 - 16MHz



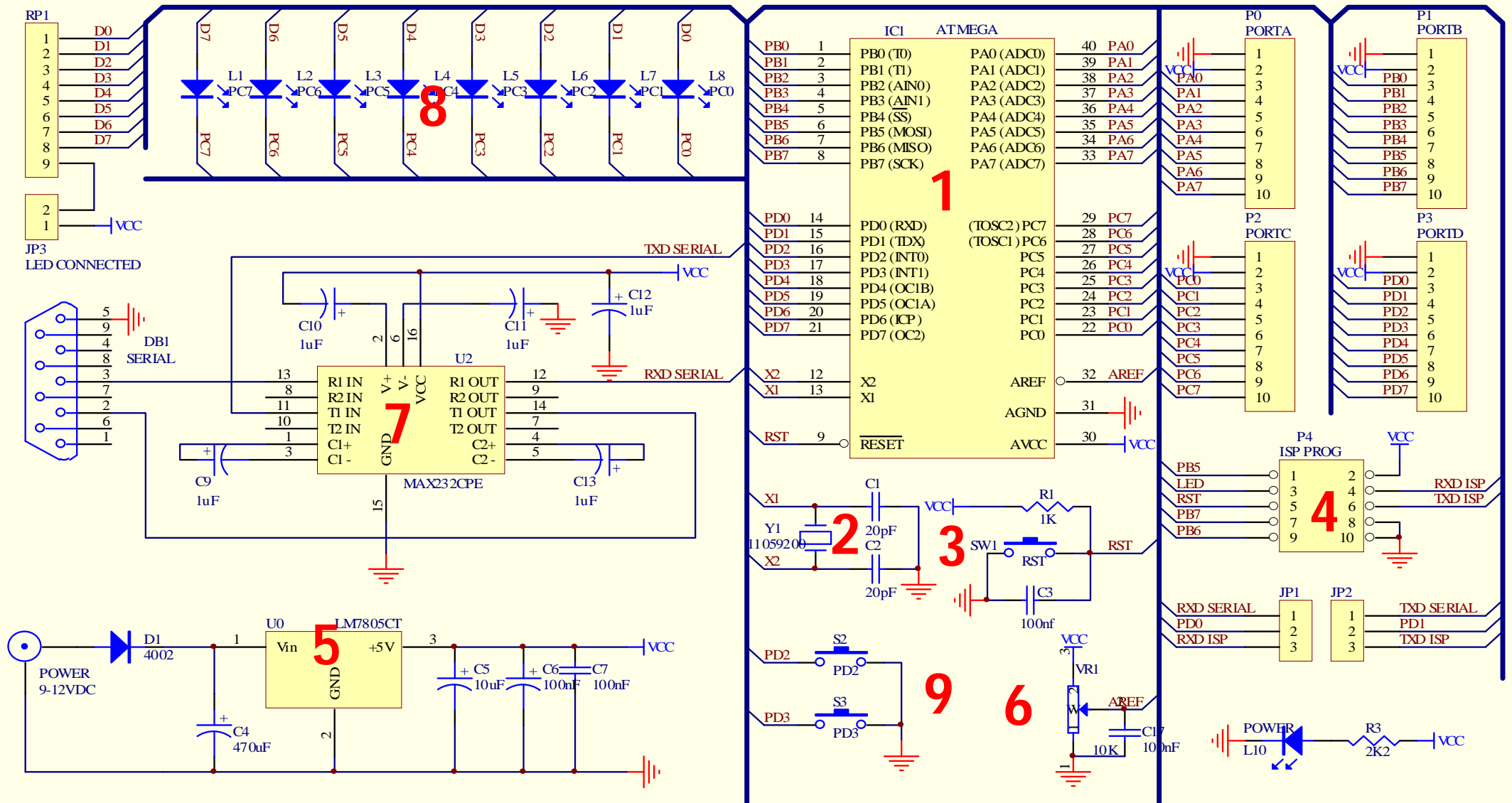
Fitur TIMERS ATmega16A

- Timers: 2x 8-bit Timer/Counter and 1x 16-bit Timer/Counter
 - Timer-0/Counter-0 (T0)
 - 8-Bit
 - Timer-1/Counter-1 (T1)
 - 16-Bit
 - Timer-2/Counter-2 (T2)
 - 8-Bit
- Perbedaan antara Timer dan Counter
 - **Timer**: Sumber detak dari **internal**.
 - **Counter**: Sumber detak dari **eksternal/pin**.

Prescaler TIMERS ATmega16A

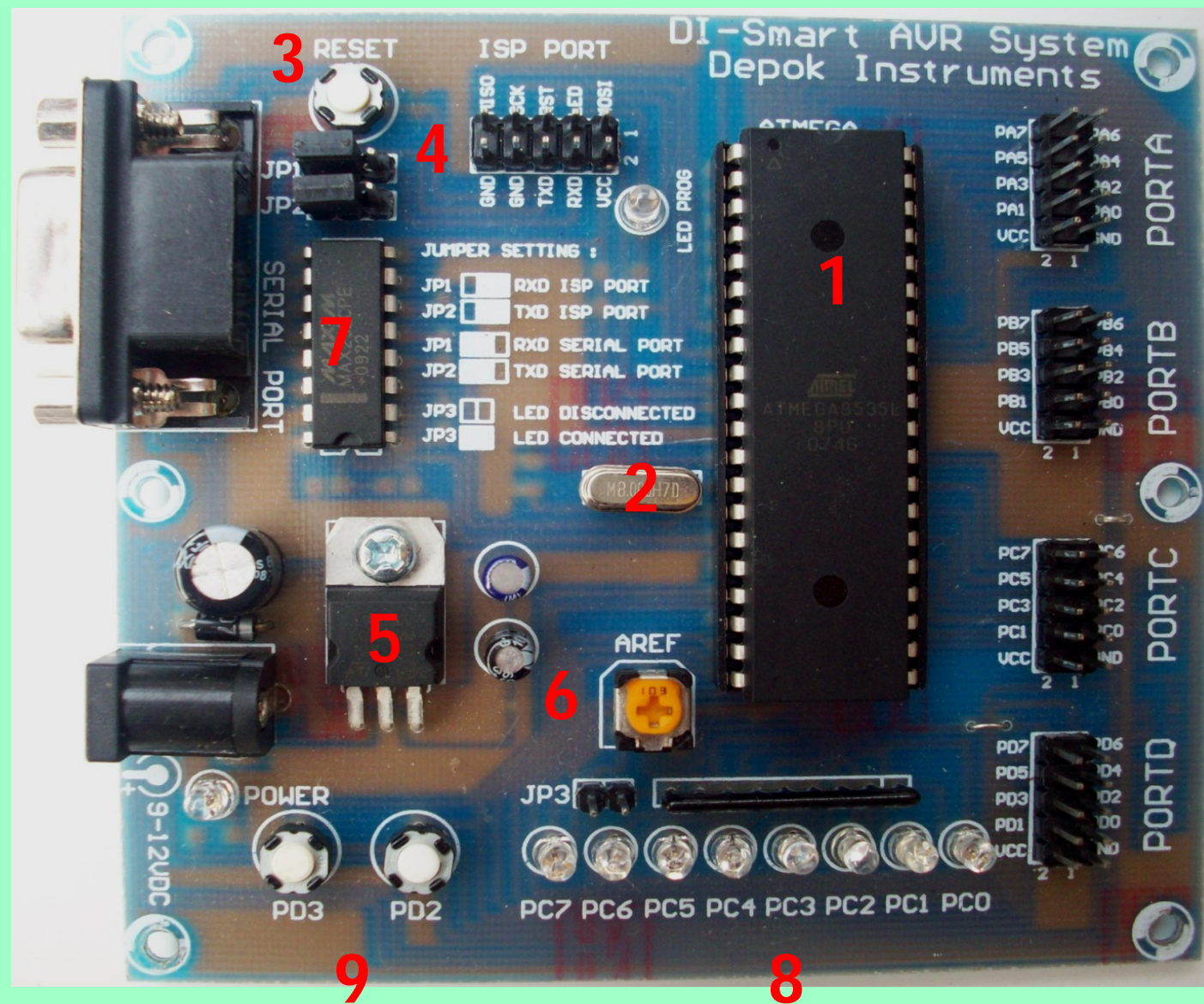
- Nilai **prescaler** adalah perkalian jumlah detak/input yang diperlukan untuk **1x pencacahan** nilai **Timer/Counter**.
- Semakin besar nilai prescaler, maka semakin lama terpenuhinya nilai Timer/Counter.
- **Prescaler Factor:**
 - 1
 - 8
 - 64
 - 256
 - 1024

DI-Smart AVR.16 System (Sistem Plus μ C. ATmega16A)



DI-Smart AVR.16 System

1. ATmega16A
2. Crystal
3. Reset
4. ISP
5. Power
6. Vref ADC
7. RS232
8. LED Array
9. Push Button



Pemrograman **Interupsi Timer-1** pada CVAVR

- Inisialisasi "Wizard"
 - Pilih "Clock Source"
 - Pilih "Clock Value"
 - Pilih "Mode"
 - Input "Value"

Pemrograman Interupsi Timer-1 16-Bit Timer (Setup "Wizard")

USART Analog Comparator ADC SPI
I2C 1 Wire TWI (I2C)
Alphanumeric LCD
Bit-Banged Project Information
Chip Ports External IRQ Timers

1

Chip: ATmega16

Clock: 11.059200 MHz

Check Reset Source

Program Type:
Application

USART Analog Comparator ADC SPI
I2C 1 Wire TWI (I2C)
Alphanumeric LCD
Bit-Banged Project Information
Chip Ports External IRQ Timers

2

Timer0 Timer1 Timer2 Watchdog

Clock Source: System Clock

Clock Value: 11059.200 kHz

Mode: Normal top=0xFFFF

Out. A: Discon. Out. B: Discon.

Input Capt. : Noise Cancel

Interrupt on: Timer1 Overflow

Value: 3cb0 h Inp. Capture: 0 h

Comp. A: 0 h B: 0 h

USART Analog Comparator ADC SPI
I2C 1 Wire TWI (I2C)
Bit-Banged Project Information
Chip Ports External IRQ Timers

3

Alphanumeric LCD

Enable Alphanumeric LCD Support

Characters/Line: 16

Connections

LCD Module AVR

RS	PORTB	Bit: 0
RD	PORTB	Bit: 1
EN	PORTB	Bit: 2
D4	PORTB	Bit: 4
D5	PORTB	Bit: 5
D6	PORTB	Bit: 6
D7	PORTB	Bit: 7

File Program Edit Help

4

Generate program, save and exit

USART Analog Compara I2C 1 Wire TWI (I2C)

Pemrograman **Interupsi Timer-1** (Inisialisasi Timer-1 Hasil "Wizard")

```
D:\PROGRAM C\AVR\Buffer.c
Notes Buffer.c
79 // Timer/Counter 1 initialization
80 // Clock source: System Clock
81 // Clock value: 11059.200 kHz
82 // Mode: Normal top=0xFFFF
83 // OC1A output: Discon.
84 // OC1B output: Discon.
85 // Noise Canceler: Off
86 // Input Capture on Falling Edge
87 // Timer1 Overflow Interrupt: On
88 // Input Capture Interrupt: Off
89 // Compare A Match Interrupt: Off
90 // Compare B Match Interrupt: Off
91 TCCR1A=0x00;
92 TCCR1B=0x01;
93 TCNT1H=0x3C;
94 TCNT1L=0xB0;
95 ICR1H=0x00;
96 ICR1L=0x00;
97 OCR1AH=0x00;
98 OCR1AL=0x00;
99 OCR1BH=0x00;
100 OCR1BL=0x00;
```

Inisialisasi Timer-1
sesuai dengan yang
diinputkan saat
Wizard.

Pemrograman Interupsi Timer-1 (Timer Registers)

- `TCCR1A=0x00; // Timer Control Register 1A`
- `TCCR1B=0x01; // Timer Control Register 1B`
- `TCNT1H=0xFC; // Timer/Counter 1H`
- `TCNT1L=0x18; // Timer/Counter 1L`
- `ICR1H=0x00; // Input Capture Register 1H`
- `ICR1L=0x00; // Input Capture Register 1L`
- `OCR1AH=0x00; // Output Compare Register 1AH`
- `OCR1AL=0x00; // Output Compare Register 1AL`
- `OCR1BH=0x00; // Output Compare Register 1BH`
- `OCR1BL=0x00; // Output Compare Register 1BL`
- `TOV1=0; // Timer/Counter Overflow 1`
- `TIFR=0B00000000; // Timer/Counter Overflow Flag Register`

Pemrograman Interupsi Timer-1 (Waktu Tunda)

```
□ #include <mega16.h>
□ #include <alcd.h>
□ #include <stdio.h>
□ unsigned int idata=0, iterasi=0;
□ unsigned char cstr[16];
□ interrupt [TIM1_OVF] void timer1_ovf_isr(void)
□ {
□ TCNT1H=0x3CB0 >> 8;
□ TCNT1L=0x3CB0 & 0xff;
□ iterasi++;
□ if(iterasi==100)
□   {
□   iterasi=0;
□   sprintf(cstr,"%d",idata++);
```

Pemrograman Interupsi Timer-1 (Waktu Tunda)

```
    □ lcd_clear();  
    □ lcd_puts(cstr);  
    □ }  
□ }  
□ void main(void)  
□ {  
□ TCCR1A=0x00;  
□ TCCR1B=0x01;  
□ TCNT1H=0x3C;  
□ TCNT1L=0xB0;  
□ ICR1H=0x00;  
□ ICR1L=0x00;  
□ OCR1AH=0x00;  
□ OCR1AL=0x00;
```

Pemrograman Interupsi Timer-1 (Waktu Tunda)

```
□ OCR1BH=0x00;  
□ OCR1BL=0x00;  
□ TIMSK=0x04;  
□ // RS - PORTB Bit 0  
□ // RD - PORTB Bit 1  
□ // EN - PORTB Bit 2  
□ // D4 - PORTB Bit 4  
□ // D5 - PORTB Bit 5  
□ // D6 - PORTB Bit 6  
□ // D7 - PORTB Bit 7  
□ // Characters/line: 16  
□ lcd_init(16);  
□ // Global enable interrupts  
□ #asm("sei")
```

Pemrograman Interupsi Timer-1 (Waktu Tunda)

```
□ while (1)
  □ {
  □ // Place your code here
  □ // No code!!!
  □ }
□ }
```

E-BOOK DINS

- <http://depokinstruments.com/category/3-e-book/>