

## Listing Program

```
/*  
  
Chip type      : ATmega8  
Program type   : Application  
AVR Core Clock frequency: 16,000000 MHz  
rumus timer:  
(61bit+1)-( 1 detik*(xtal/prescaller)  
TCNT1: (65535+1)+(1*(16mhz/1024))  
TCNT1: 49911  
jadikan HEXADESIMAL  
TCNT1: C2F7  
*****/  
  
#include <mega8.h> // library mikrokontroller atmega8  
#include <stdio.h> // standart i/o  
#include <delay.h> // delay  
// Alphanumeric LCD functions  
#include <alcd.h> // lcd  
  
// i/o konfig  
#define sw1 PINC.0  
#define sw2 PINC.1  
#define sw3 PINC.2  
#define relay PORTB.0  
#define servo OCR2  
#define mpx 3
```

```

char buff[33];

// map waktu
int timer[]={1,2,3,4,5};

// parameter index waktu dan pulsa servo
int eeprom time=0;
int eeprom pulsa=40;

// pewaktu dan tanda start/stop
int menit=0,detik=0,go=0;

// timer interrupt
// Timer1 overflow interrupt service routine
interrupt [TIM1_OVF] void timer1_ovf_isr(void)
{
    // Reinitialize Timer1 value
    TCNT1H=0xC2F7 >> 8;
    TCNT1L=0xC2F7 & 0xff;
    // Place your code here

    // program menit detik
    if(go==1)
    { detik--; if(detik<0)
        { menit--; detik=59; }
    }}

#define ADC_VREF_TYPE 0x40
// Read the AD conversion result
unsigned int read_adc(unsigned char adc_input)
{
    ADMUX=adc_input | (ADC_VREF_TYPE & 0xff);

    // Delay needed for the stabilization of the ADC input voltage
    delay_us(10);

```

```

// Start the AD conversion
ADCSRA |= 0x40;

// Wait for the AD conversion to complete
while ((ADCSRA & 0x10) == 0);
ADCSRA |= 0x10;
return ADCW; }

// Declare your global variables here

// membaca sensor mpx
float bacatekanan()
{
    int i;
    float buffer, baca1, baca2, volt_tampung, volt=0, pressure, sample, ratarata;
    float offset=0.2;
    for(i=0; i<20; i++)
    { sample=sample+read_adc(mpx); delay_ms(10); }
    ratarata = sample / 20;
    volt = (float)(ratarata * 5) / 1023;
    volt=volt-offset;
    pressure=volt*((float)500/((float)4.7-offset));
    //pressure=volt;
    // membatasi minimal 0
    if(pressure<0)pressure=0;
    // memberi nilai balik
    return pressure; }

// fungsi set untuk seting waktu
void set()
{
    int setmenu=0;
    // matikan waktu dan relay

```

```

go=0; relay=0; lcd_clear(); delay_ms(200); while(1)
{ if(sw3==0) setmenu++;
  if(setmenu==0)
  { lcd_clear(); lcd_gotoxy(0,0); sprintf(buff,"Waktu: %d",timer[time]);
    lcd_puts(buff);
    if(sw1==0)time++;
    if(sw2==0)time--;
    if(time<0)time=4;
    if(time>4)time=0; }

    if(setmenu>0)break; delay_ms(150);
  }
lcd_clear();
delay_ms(200); }
void programku()
{
  float tekanan=bacatekanan();
  int valve = 20-(pulsa-40);
  if(sw1==0&&go==0)
  { detik=0; menit=timer[time]; go=1; relay=1; lcd_clear(); lcd_gotoxy(0,0);
    lcd_putsf("Start");
    delay_ms(500);
  }

  // atur katup valve
  if(sw1==0&&go==1)pulsa--;
  if(sw2==0&&go==1)pulsa++;
  if(pulsa<40)pulsa=40;
  if(pulsa>60)pulsa=60;
  if(sw3==0) set();

```

```

if(menit==0&&detik==0&&go==1)
    { go=0; relay=0; lcd_clear();
      lcd_gotoxy(0,0);
      lcd_putsf("Stop");
      delay_ms(500);
    }

/**
lcd_clear();
lcd_gotoxy(0,0);
sprintf(buff,"Press:%.1f Kpa", tekanan);
lcd_puts(buff);
**/

lcd_gotoxy(0,1);
sprintf(buff,"Valve:%d",valve);
lcd_puts(buff);
lcd_gotoxy(10,1);
sprintf(buff,"%02d:%02d",menit,detik);
lcd_puts(buff);

servo=pulsa;

delay_ms(100); }

void main(void)
{
// Declare your local variables here

// Input/Output Ports initialization
// Port B initialization
// Func7=In Func6=In Func5=In Func4=In Func3=Out Func2=In Func1=In Func0=Out

```

```
// State7=T State6=T State5=T State4=T State3=0 State2=T State1=T State0=0
PORTB=0x00;
DDRB=0x09;
// Port C initialization
// Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In
// State6=T State5=T State4=T State3=T State2=P State1=P State0=P
PORTC=0x07;
DDRC=0x00;
// Port D initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T
PORTD=0x00;
DDRD=0x00;
// Timer/Counter 0 initialization
// Clock source: System Clock
// Clock value: Timer 0 Stopped
TCCR0=0x00;
TCNT0=0x00;
// Timer/Counter 1 initialization
// Clock source: System Clock
// Clock value: 15,625 kHz
// Mode: Normal top=0xFFFF
// OC1A output: Discon.
// OC1B output: Discon.
// Noise Canceler: Off
// Input Capture on Falling Edge
// Timer1 Overflow Interrupt: On
// Input Capture Interrupt: Off
// Compare A Match Interrupt: Off
```

```
// Compare B Match Interrupt: Off
TCCR1A=0x00;
TCCR1B=0x05;
TCNT1H=0xC2;
TCNT1L=0xF7;
ICR1H=0x00;
ICR1L=0x00;
OCR1AH=0x00;
OCR1AL=0x00;
OCR1BH=0x00;
OCR1BL=0x00;

// Timer/Counter 2 initialization
// Clock source: System Clock
// Clock value: 62,500 kHz
// Mode: Phase correct PWM top=0xFF
// OC2 output: Non-Inverted PWM
ASSR=0x00;
TCCR2=0x66;
TCNT2=0x00;
OCR2=0x00;

// External Interrupt(s) initialization
// INT0: Off
// INT1: Off
MCUCR=0x00;

// Timer(s)/Counter(s) Interrupt(s) initialization
TIMSK=0x04;

// USART initialization
// USART disabled
UCSRB=0x00;
```

```
// Analog Comparator initialization
// Analog Comparator: Off
// Analog Comparator Input Capture by Timer/Counter 1: Off
ACSR=0x80;
SFIOR=0x00;

// ADC initialization
// ADC Clock frequency: 1000,000 kHz
// ADC Voltage Reference: AVCC pin
ADMUX=ADC_VREF_TYPE & 0xff;
ADCSRA=0x84;

// SPI initialization
// SPI disabled
SPCR=0x00;

// TWI initialization
// TWI disabled
TWCR=0x00;

// Alphanumeric LCD initialization
// Connections are specified in the
// Project|Configure|C Compiler|Libraries|Alphanumeric LCD menu:
// RS - PORTD Bit 0
// RD - PORTD Bit 7
// EN - PORTD Bit 1
// D4 - PORTD Bit 2
// D5 - PORTD Bit 3
// D6 - PORTD Bit 4
// D7 - PORTD Bit 5
// Characters/line: 16
lcd_init(16);
```



```
// Global enable interrupts

#asm("sei")

servo=pulsa;

lcd_clear();

lcd_gotoxy(0,0);

lcd_putsf("Variable");

lcd_gotoxy(0,1);

lcd_putsf("Nebulizer");

//go==0;

delay_ms(1000);

while (1)

{

// Place your code here

programku();  }}
```





Kebisingan Ruangan 34 dB



Pengukuran Sisi Kanan Alat 56 dB



Pengukuran Sisi Atas Alat



Pengukuran Sisi Bawah Alat 56 dB



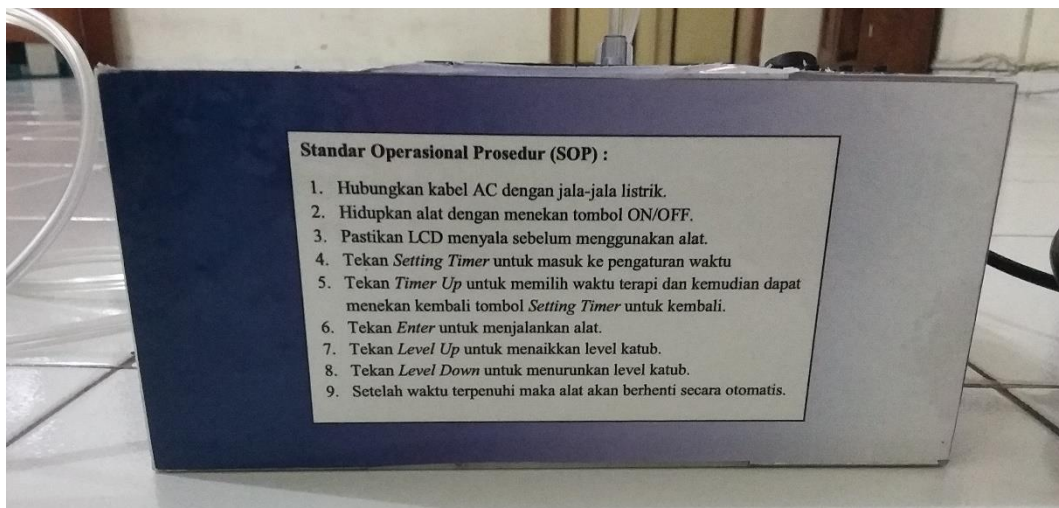
Pengukuran Sisi Kiri Alat 60 dB



Pengukuran Diatas Alat 55 dB



Nebulizer Kompresor



SOP Nebulizer Kompresor