

/\*\*\*\*\*

Project : SIMULATOR MONITORING DAN PEMBERSIH GAS CO PADA RUANGAN  
BERBASIS MIKROKONTROLER ATmega8535

Date : 8/3/2016

Author : DELIYANA HARUN

Company : UNIVERSITAS MUHAMMADIYAH YOGYAKARTA

Chip type : ATmega8535  
Program type : Application  
AVR Core Clock frequency: 1.000000 MHz  
Memory model : Small  
External RAM size : 0  
Data Stack size : 128

\*\*\*\*\*/

Penulisan Header Program, Baris yang berisi deklarasi fungsi atau variabel. File ini digunakan semacam perputakaan bagi pernyataan yang ada di tubuh program

```
#include <mega8535.h>
#include <delay.h>
#include <stdlib.h>
#include <stdio.h>
#include <alcd.h>
#include <math.h>
```

Pendeklarasian Tipe Data yang digunakan dalam program

```
char mikrodetik=0,detik,i=0;
int data1, data2, X , Y , Rs , D , E , Ro;
float data, tegangan, Ppm, rata;
unsigned char temp[6], temp2[6];
```

Deklarasi ADC

```
#define ADC_VREF_TYPE 0x00
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;
}
```

#### Deklarasi Timer0 Untuk PWM

```
// Timer 0 overflow interrupt service routine
interrupt [TIM0_OVF] void timer0_ovf_isr(void)
{
// Reinitialize Timer 0 value
TCNT0=0x9E;
// Place your code here
if(mikrodetik==10)
{
detik++;
if(detik>1)
{
OCR1A=data1;
OCR1B=data2;
detik=0;
}
mikrodetik=0;
} }
}
```

#### Deklarasi Fungsi/Program Utama (Rumus pembacaan ADC,PWM dan Sensor)

```
void baca_ppm()
{
data1= 51;
data2= 144;
TIFR=0;
for(i=0;i<100;i++)
{
data=read_adc(0);
tegangan=(data*4.20/1024);
rata+=tegangan;
}
rata=rata/100;
//lcd_gotoxy(0,1);
ftoa(rata,2,temp2);
//lcd_puts(temp2);
delay_ms(100);
X= (4.22 * 10);
Rs= (X / rata);
Ro = 34;

D = Rs/Ro;
E = 184.51/D;
Ppm = pow(E,0.88);

lcd_gotoxy(0,1);
lcd_putsf("UDARA =");
lcd_gotoxy(0,0);
lcd_putsf("PPM  =");
lcd_gotoxy(9 ,0);
ftoa(Ppm,1,temp);
lcd_puts(temp);
}
void driver_set()
{
if(Ppm>=30)
{
PORTB.0=1;
PORTB.1=1;
PORTB.2=1;
PORTB.3=1;
}else {PORTB.0 = 0;PORTB.1=0; PORTB.2=0; PORTB.3=0;}}
```

## Program Pengatur tampilan pada LCD

```
void atur_tampilan()
{
if (Ppm>=30)
{
lcd_gotoxy(9,1);
lcd_puts("kotor");
}else
{lcd_gotoxy(9,1);
lcd_puts("bersih");
delay_ms (500);
} }
}
```

## Konfigurasi input/output pada PORT

```
void main(void)
{
PORTA=0x00;
DDRA=0x00;

PORTB=0x00;
DDRB=0xFF;

PORTC=0x00;
DDRC=0x00;

PORTD=0x00;
DDRD=0x30;

TCCR0=0x05;
TCNT0=0x9E;
OCR0=0x00;

TCCR1A=0xB3;
TCCR1B=0x09;
TCNT1H=0x00;
TCNT1L=0x00;
ICR1H=0x00;
ICR1L=0x00;
OCR1AH=0x00;
OCR1AL=0x00;
OCR1BH=0x00;
OCR1BL=0x00;

TIMSK=0x01;

UCSRB=0x00;

ACSR=0x80;
SFIOR=0x00;

ADMUX=ADC_VREF_TYPE & 0xff;
ADCSRA=0xA7;
SFIOR&=0x0F;

SPCR=0x00;

TWCR=0x00;
```

## Konfigurasi LCD

```
// Alphanumeric LCD initialization
// Connections specified in the
// Project|Configure|C Compiler|Libraries|Alphanumeric LCD
menu:
// RS - PORTC Bit 0
// RD - PORTC Bit 1
// EN - PORTC Bit 2
// D4 - PORTC Bit 4
// D5 - PORTC Bit 5
// D6 - PORTC Bit 6
// D7 - PORTC Bit 7
// Characters/line: 16
lcd_init(16);

// Global enable interrupts
#asm("sei")
lcd_clear();
lcd_gotoxy(1,0);
lcd_putsf("MONITORING CO");
lcd_gotoxy(3,1);
lcd_putsf("TEM 2013");
delay_ms(1000);
lcd_clear();
lcd_gotoxy(1,0);
lcd_putsf("DELIYANA HARUN");
lcd_gotoxy(2,1);
lcd_putsf("20133010047");
delay_ms(1000);
lcd_clear ();
```

## Pengulangan Program

```
while (1)
{
    baca_ppm();
    driver_set();
    atur_tampilan();

}
}
```