

### Listing Program

```

/*****
Chip type      : ATmega8
AVR Core Clock frequency: 8,000000 MHz
*****/

// lib atmega8
#include <mega8.h>
// lib standart i/o
#include <stdio.h>
// lib delay
#include <delay.h>
char lcdbuff[33];
// Alphanumeric LCD functions
#include <alcd.h>

#define s1 PINC.0
#define s2 PINC.1
#define s3 PINC.2

#define out OCR2

int parameter_waktu[]={ 1,5,10,15 };
int parameter_tegangan[]={ 130,150,180,215,230 };

eeprom int detik=0,jeda=5,pwm=0;
int timer=0;
bit start=0;

/*
rumus timer:
(16bit+1)-(1detik*(xtal/prescaller)

```

```

TCNT: (65535+1)+(1*(8mhz/1024))
TCNT: 57723
jadikan HEXADESIMAL
TCNT: E17B
*/

// Timer1 overflow interrupt service routine
interrupt [TIM1_OVF] void timer1_ovf_isr(void)
{
// Reinitialize Timer1 value
TCNT1H=0xE17B >> 8;
TCNT1L=0xE17B & 0xFF;
// Place your code here

if(start==1){
timer--;
if(timer==0)start=0;
}

}

// fungsi pengaturan

void setting(){
int menu=0;
lcd_clear();
delay_ms(200);
while(1){
lcd_clear();
// geser menu
if(s1==0)menu++;

```

```

// keluar dari menu
if(menu>2)break;

// pengaturan pertama
if(menu==0){
lcd_gotoxy(0,0);
lcd_putsf("Set Timer");
lcd_gotoxy(0,1);
sprintf(lcdbuff,"%i Minuts",parameter_waktu[detik]);
lcd_puts(lcdbuff);
// naik/turunkan angka
if(s2==0)detik++;
if(s3==0)detik--;
// batas angka
if(detik>3)detik=0;
if(detik<0)detik=3;
}

if(menu==1){
lcd_gotoxy(0,0);
lcd_putsf("Set Duration");
lcd_gotoxy(0,1);
sprintf(lcdbuff,"%i Ms",jeda*50);
lcd_puts(lcdbuff);

if(s2==0)jeda++;
if(s3==0)jeda--;

if(jeda>6)jeda=1;
if(jeda<1)jeda=6;
}

```

```

if(menu==2){
lcd_gotoxy(0,0);
lcd_putsf("Set Level");
lcd_gotoxy(0,1);
sprintf(lcdbuff,pwm+1);
lcd_puts(lcdbuff);

if(s2==0)pwm++;
if(s3==0)pwm--;

if(pwm>5)pwm=1;
if(pwm<1)pwm=5;
}

delay_ms(200);
}
lcd_clear();
delay_ms(200);
}
// program eksekusi
void yang_utama(){
// masuk menu
if(s1==0) {
start=0;
out=0;
setting();
}

// start
if(s3==0) {
if(start==0){
timer=parameter_waktu[detik]*60;

```

```

start=1;
}
}

// stop
if(s2==0) {
start=0;
out=0;
}
// display
lcd_clear();
lcd_gotoxy(0,0);
sprintf(lcdbuff,"Level:%i",pwm+1);
lcd_puts(lcdbuff);
lcd_gotoxy(0,1);
sprintf(lcdbuff,"Timer:%i:%i ",timer/60,timer%60);
lcd_puts(lcdbuff);

// jeda
if(start==1) {
out=parameter_tegangan[pwm];
delay_ms(jeda*50);
out=0;
delay_ms(jeda*50);
}
delay_ms(10);
}

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=In
// State7=T State6=T State5=T State4=T State3=0 State2=T State1=T State0=T
PORTB=0x00;
DDRB=0x08;

// 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: 7,813 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=0xE1;
TCNT1L=0x7B;
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=0x63;
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;
SFIO=0x00;

// ADC initialization
// ADC Clock frequency: 1000,000 kHz
// ADC Voltage Reference: AVCC pin
ADMUX=ADC_VREF_TYPE & 0xff;
ADCSRA=0x83;

// 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")
lcd_clear();
lcd_gotoxy(0,0);
lcd_putsf("Ihya Ulumuddin G");
lcd_gotoxy(0,1);
lcd_putsf("20143010033");
delay_ms(1000);
lcd_clear();
lcd_gotoxy(0,0);
lcd_putsf("Electro Teraphy");
delay_ms(1000);
while (1)
{
    // Place your code here
    yang_utama();
}
}
```