

# LAMPIRAN

## **Pengoperasian Alat**

1. Tempatkan alat di ruangan yang tertutup
2. Lepaskan penutup debu.
3. Hubungkan alat dengan catu daya.
4. Hidupkan alat dengan menekan tombol ON/OFF ke posisi ON.
5. Atur posisi pasien dengan jarak aman penyinaran 45-60 cm
6. Matikan alat dengan menekan tombol ON/OFF ke posisi OFF.
7. Lepaskan hubungan catu daya.
8. Bersihkan alat dan kembalikan ke tempat penyimpanan. Pastikan alat dalam kondisi baik dan siap untuk difungsikan kembali.
9. Pasang penutup debu.
10. Catat beban kerja alat → dalam jumlah pemakaian.

# Listing Program

/\*\*\*\*\*\*  
/\*\*\*\*\*

This program was produced by the

CodeWizardAVR V2.05.3 Standard

Automatic Program Generator

© Copyright 1998-2011 Pavel Haiduc, HP InfoTech s.r.l.

<http://www.hpinfotech.com>

Project :

Version :

Date : 28/02/2017

Author : Microsoft

Company : Microsoft

Comments:

Chip type : ATmega16

Program type : Application

AVR Core Clock frequency: 12,000000 MHz

Memory model : Small

External RAM size : 0

*Listing* Program

Data Stack size : 256

\*\*\*\*\*/

```
#include <mega16.h>
```

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
#include <delay.h>
```

```
#define trigger PORTA.1
```

```
#define echo PINA.0
```

```
unsigned int jarak;
```

```
char buf[33];
```

```
unsigned char temp[6];
```

```
int detik=0, menit=0;
```

```
// Alphanumeric LCD functions
```

```
#include <alcd.h>
```

```
// Timer1 overflow interrupt service routine

interrupt [TIM1_OVF] void timer1_ovf_isr(void)

{

// Reinitialize Timer1 value

TCNT1H=0xD23A >> 8;

TCNT1L=0xD23A & 0xff;

detik++; // Place your code here

}

void jam_digital()

{

if (detik>=60)

{

    lcd_clear();

    detik=0;

    menit++;

}

}
```

```
    if (menit>=30)
    {
        lcd_clear();
        menit=0;
    }
}
```

```
void tampil_lcd()
{
    lcd_clear();
    lcd_gotoxy(0,0);
    lcd_putsf("Timer=");

    lcd_gotoxy(0,1);
    lcd_putsf("Jarak=");
```

```
    itoa(detik,temp); //menampilkan DETIK di LCD
```

```
    lcd_gotoxy(10,0);
```

```
lcd_puts(temp);
```

```
lcd_gotoxy(9,0); //menampilkan :
```

```
lcd_putsf(":");
```

```
itoa(menit,temp); //menampilkan MENIT di LCD
```

```
lcd_gotoxy(7,0);
```

```
lcd_puts(temp);
```

```
sprintf(buf,"%d cm",jarak);
```

```
lcd_gotoxy (6,1);
```

```
lcd_puts (buf);
```

```
}
```

```
void ukur_jarak()
```

```
{
```

```
unsigned int i;
```

```
jarak=0;

delay_us(100);

trigger=1; //tout, H=5 us

delay_us(15);

trigger=0;

delay_us(100);

while(!echo);

for (i=0;i<=200;i++)

{

if (echo) {jarak++;}

delay_us(58);

}

}

// Declare your global variables here

void main(void)

{
```



// Timer/Counter 0 initialization

// Clock source: System Clock

// Clock value: Timer 0 Stopped

// Mode: Normal top=0xFF

// OC0 output: Disconnected

TCCR0=0x00;

TCNT0=0x00;

OCR0=0x00;

// Timer/Counter 1 initialization

// Clock source: System Clock

// Clock value: 11,719 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=0xD2;
```

```
TCNT1L=0x3A;
```

```
ICR1H=0x00;
```

```
ICR1L=0x00;
```

```
OCR1AH=0x00;
```

```
OCR1AL=0x00;
```

```
OCR1BH=0x00;
```

```
OCR1BL=0x00;
```

```
// Timer/Counter 2 initialization
```

```
// Clock source: System Clock
```

```
// Clock value: Timer2 Stopped
```

```
// Mode: Normal top=0xFF
```

```
// OC2 output: Disconnected
```

```
ASSR=0x00;
```

```
TCCR2=0x00;
```

```
TCNT2=0x00;
```

```
OCR2=0x00;

// External Interrupt(s) initialization

// INT0: Off

// INT1: Off

// INT2: Off

MCUCR=0x00;

MCUCSR=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 disabled
```

```
ADCSRA=0x00;
```

```
// 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 - 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: 8

lcd_init(16);

#asm("sei")

DDRA.0=0;

DDRA.1=1;

DDRC.1=1;

lcd_clear();

lcd_putsf("TERAPI INFRAMERAH");

lcd_gotoxy(0,1);

lcd_putsf("Jarak=");

while (1)

{

ukur_jarak();

sprintf(buf,"%d cm",jarak);
```

```
lcd_gotoxy (6,1);

lcd_puts (buf);

if(jarak>=30 && jarak <=100 && menit < 10 )
{
    PORTC.1=1;

    tampil_lcd();

    jam_digital();

}

else
{
    PORTC.1=0;

    delay_ms(2000);

    detik = 0;

    menit = 0;

}

if (jarak>=30 && jarak <=100 && menit == 10 )
{
    PORTC.1=0;
```

```
delay_ms(30000);
```

```
detik=0;
```

```
menit=0;
```

```
}
```

```
delay_ms(500);
```

```
}
```

```
}
```