

LAMPIRAN

A. 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    : 6/8/2017
Author  : syaifuahmad
Company :
Comments:

Chip type           : ATmega16
Program type        : Application
AVR Core Clock frequency: 16.000000 MHz
Memory model        : Small
External RAM size   : 0
Data Stack size     : 256
*****/

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

```

#include <stdlib.h>

#include <delay.h>

Unsigned char
pulmenit,satmenit,puldetik,satdetik,data1,data2,detik;

int menit;

unsigned char temp1[2],temp2[2],temp3[2],temp4[2];

bit a=0;

// Timer1 overflow interrupt service routine
interrupt [TIM1_OVF] void timer1_ovf_isr(void)
{
// Reinitialize Timer1 value
TCNT1H=0xBDC >> 8;
TCNT1L=0xBDC & 0xff;

if(a==1)
{
if(detik==0)
{menit--;detik=59;}else{detik--;}
}

if(menit==0&&detik==0){a=0;}
}

void tombol()
{
if(a==0)
{
if(PIND.5==0){menit++;delay_ms(250);if(menit>5){menit=0;}}
if(PIND.6==0){menit--;delay_ms(250);if(menit<0){menit=0;}}
}
}

```

```

    if (PIND.7==0&&menit>0) {a=1;}

if (a==1) {TIMSK=0x04;TCCR1B=0x04;PORTA.7=1;}else{TCCR1B=0x00;TI
MSK=0x00;PORTA.7=0;}
}
}

void perubah_modulo()
{
    pulmenit=data1%100;
    pulmenit=pulmenit/10;

    satmenit=data1%100;
    satmenit=satmenit%10;

    puldetik=data2%100;
    puldetik=puldetik/10;

    satdetik=data2%100;
    satdetik=satdetik%10;
}

void tampil_lcd()
{
    lcd_gotoxy(2,0);
    lcd_putsf("BELLY DANCER");
    lcd_gotoxy(0,1);
    lcd_putsf("TIMER ;");
}

```

```
    lcd_gotoxy(10,1);  
    lcd_putsf(":");  
    lcd_gotoxy(8,1);  
    itoa(pulmenit,temp1);  
    lcd_puts(temp1);  
    lcd_gotoxy(9,1);  
    itoa(satmenit,temp2);  
    lcd_puts(temp2);  
    lcd_gotoxy(11,1);  
    itoa(puldetik,temp3);  
    lcd_puts(temp3);  
    lcd_gotoxy(12,1);  
    itoa(satdetik,temp4);  
    lcd_puts(temp4);  
}
```

```
void main(void)
```

```
{
```

```
// Declare your local variables here
```

```
// Input/Output Ports initialization
```

```
// Port A initialization
```

```
// Func7=In Func6=Out Func5=In Func4=In Func3=In Func2=In
```

```
Func1=In Func0=In
```

```
// State7=T State6=0 State5=T State4=T State3=T State2=T
```

```
State1=T State0=T
```

```
PORTA=0x00;
```

```
DDRA=0xFF;
```

```
// Port D initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In
Func1=In Func0=In
// State7=P State6=P State5=P State4=T State3=T State2=T
State1=T State0=T
PORTD=0xE0;
DDRD=0x00;

// Timer/Counter 1 initialization
// Clock source: System Clock
// Clock value: 62.500 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=0x04;
TCNT1H=0x0B;
TCNT1L=0xDC;
ICR1H=0x00;
ICR1L=0x00;
OCR1AH=0x00;
```

```
OCR1AL=0x00;

OCR1BH=0x00;

OCR1BL=0x00;

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

// Alphanumeric LCD initialization
// Connections are 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")

while (1)
{
    tombol();
    data1=menit;
```

```
data2=detik;  
perubah_modulo();  
tampil_lcd();  
}  
}
```

