DOKUMENTASI PROSES PEMBUATAN ALAT

Pembuatan rotor alat *centrifuge*

Proses perakitan box alat *centrifuge*
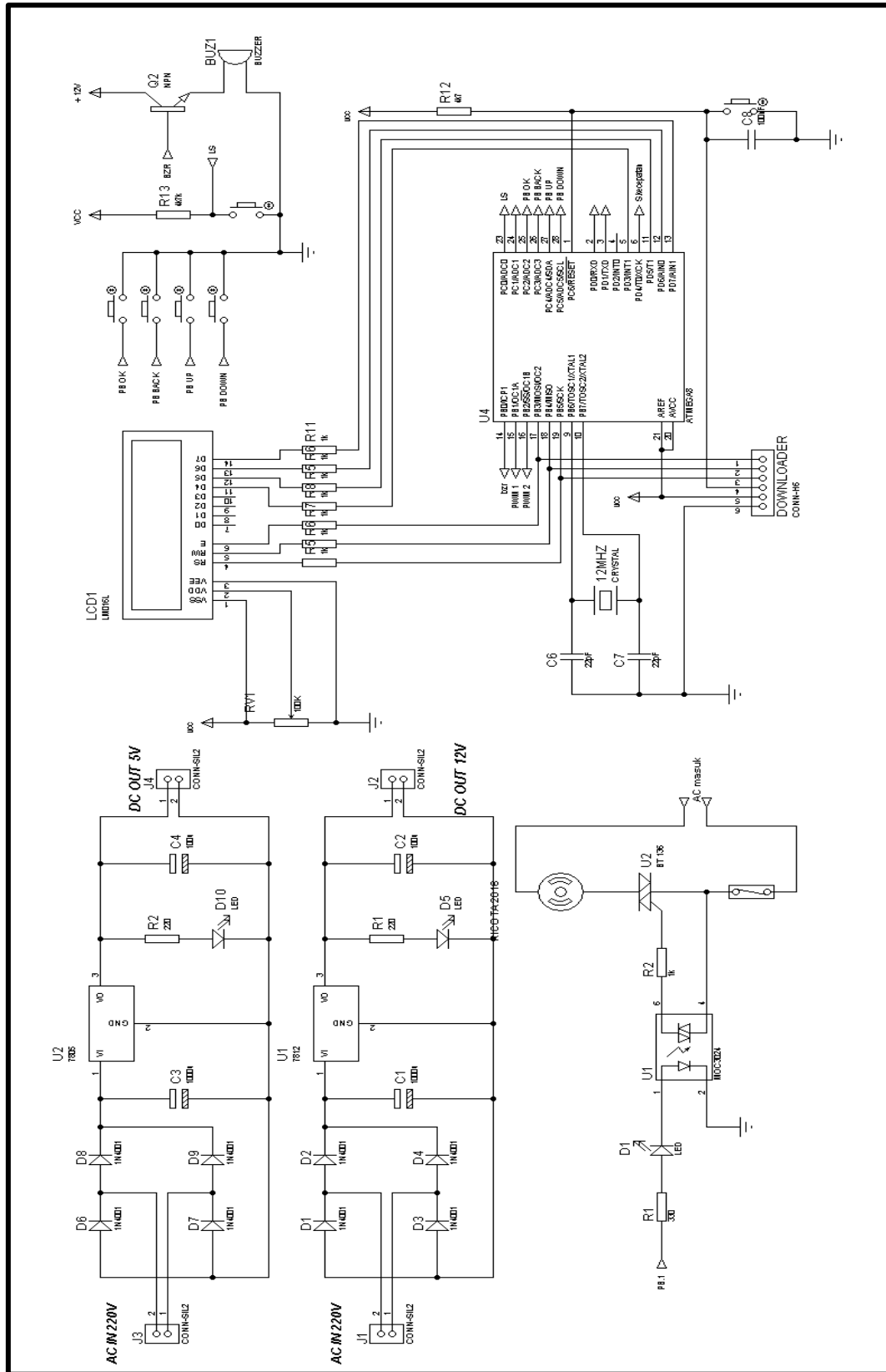

Perakitan MINSIS At mega 8

Proses pengkabelan alat *centrifuge*

RANGKAIAN KESELURUHAN

LISTING PROGRAM "CENTRIFUGE DENGAN ROTOR SUDUT TETAP BERBASIS MICROCONTROL AT MEGE8"

```c
/******************************************************
This program was produced by the
CodeWizardAVR V2.05.0 Professional
Automatic Program Generator
© Copyright 1998-2010 Pavel Haiduc, HP InfoTech s.r.l.
http://www.hpinfotech.com
Chip type               : ATmega8
Program type            : Application
AVR Core Clock frequency: 12,000000 MHz
Memory model            : Small
External RAM size        : 0
Data Stack size         : 256
*******************************************************/
#include <mega8.h>
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include <delay.h>
// Alphanumeric LCD Module functions
#include <alcd.h>
// ------------------
#define up PINC.2
#define down PINC.3
#define ok PINC.1
#define cancel PINC.0
#define lock PINC.5
#define pwm  OCR1A
//--------------------
unsigned int mikrodetik, detik,hitung=0, input=10;
float pulsa,frekuensi=0;
int menit;
unsigned char temp[2],temp2[2],temp3[4],chitung[10];
// Timer2 overflow interrupt service routine
interrupt [TIM2_OVF] void timer2_ovf_isr(void)
{
// Reinitialize Timer2 value
TCNT2=0x8A;
// Place your code here
mikrodetik++;
if(mikrodetik==99)
{
//frekuensi=frekuensi-100;
pulsa=frekuensi*30;
frekuensi=0;
TCNT0=0;
if (detik==0)
{menit--;detik=59;}else{detik--;}
mikrodetik=0;}
}
// Declare your global variables here
// tampilan pewaktu
    void tampilkan_timer()
```

```
    {
    lcd_gotoxy(0,0);
    lcd_putsf("TIMER:");
    if(menit<10)
    {
    lcd_gotoxy(6,0);
    lcd_putsf("0");
    lcd_gotoxy(7,0);
    itoa(menit,temp);
    lcd_puts(temp);
    }
        else
        {
        lcd_gotoxy(6,0);
        itoa(menit,temp);
        lcd_puts(temp);
        }
    if(detik<10)
    {
    lcd_gotoxy(9,0);
    lcd_putsf("0");
    lcd_gotoxy(10,0);
    itoa(detik,temp2);
    lcd_puts(temp2);
    }
        else
        {
        lcd_gotoxy(9,0);
        itoa(detik,temp);
        lcd_puts(temp);
        lcd_gotoxy(8,0);
        lcd_putsf(":");
        }
    }
void datapwm()
{
   if (input==10)
  {pwm=42;}
   if (input==11)
  {pwm=50;}
   if (input==12)
  {pwm=59;}
   if (input==13)
  {pwm=74;}
   if (input==14)
  {pwm=85;}
   if (input==15)
  {pwm=95;}
   if (input==16)
  {pwm=114;}
   if (input==17)
  {pwm=140;}
   if (input==18)
  {pwm=155;}
   if (input==19)
  {pwm=170;}
```

```c
   if (input==20)
  {pwm=185;}
   if (input==21)
  {pwm=202;}
   if (input==22)
  {pwm=220;}
   if (input==23)
  {pwm=250;}
   if (input==24)
  {pwm=280;}
   if (input==25)
  {pwm=308;}
   if (input==26)
  {pwm=328;}
   if (input==27)
  {pwm=340;}
   if (input==28)
  {pwm=367;}
   if (input==29)
  {pwm=380;}
   if (input==30)
  {pwm=440;}
}
// tampil rpm
void tampil_rpm()
{
    lcd_clear();
    lcd_gotoxy(8,0);
    lcd_putsf(":");
    lcd_gotoxy(0,1);
    lcd_putsf("RPM:");
    lcd_gotoxy(4,1);
    itoa(pulsa,temp3);
    lcd_puts(temp3);
}
// menu awal
void menuawal()
    {
        lcd_gotoxy(0,0);
        lcd_putsf("    WELCOME");
        lcd_gotoxy(0,1);
        lcd_putsf("   Tugas Ahkir ");
        delay_ms(1000);
        lcd_clear();
        lcd_gotoxy(0,0);
        lcd_putsf("   Centrifuge");
        lcd_gotoxy(0,1);
        lcd_putsf(" Rico Pradana N");
        delay_ms(1000);
        lcd_clear();
        for (hitung=0;hitung<=5;hitung++)
            {
                lcd_clear();
                itoa (hitung,chitung);
                lcd_clear();
                lcd_gotoxy(0,0);
```

```c
                    lcd_putsf("Mohon Tunggu");
                    lcd_gotoxy(7,1);
                    lcd_puts(chitung);
                    delay_ms(500);
                }
                    lcd_clear();
        }
// tampil seting rpm
void tampil( int input)
    {
    int koera;
    koera=input/10;
    lcd_putchar(48+koera);
    input%=10;
    lcd_putchar(48+input);
    }
// tampil seting waktu
void tampil1( int menit)
    {
    int koera1;
    koera1=menit/10;
    lcd_putchar(48+koera1);
    menit%=10;
    lcd_putchar(48+menit);
    }
// menu utama
void menu()
{
            lcd_clear();
            //input=0;
menu01:
        delay_ms(200);
        lcd_gotoxy(0,0);
        lcd_putsf("Seting Kecepatan");
        lcd_gotoxy(2,1);
        lcd_putsf("00 RPM");
        lcd_gotoxy(0,1);
        tampil(input);
        if (up==0)
        {
        input++;
        if (input>=31)
            {input=10;}
            delay_ms(100);
        }
        if (down==0)
        {
        input--;
            if (input<=10)
            {input=30;}
            delay_ms(100);
        }
        if (ok==0)
        {
        goto menu02;
        }
```

```c
        goto menu01;
menu02:
        delay_ms(200);    // bouncing sw
        lcd_clear();
        lcd_gotoxy(0,0);
        lcd_putsf("  Seting Waktu");
        lcd_gotoxy(0,1);
        tampil1(menit);
        lcd_gotoxy(3,1);
        lcd_putsf("Menit");
        if (up==0)
        {
        menit++;
            if (menit>=61)
            {menit=0;}
            delay_ms(100);
        }
        if (down==0)
        {
        menit--;
            if (menit==0)
            {menit=60;}
            delay_ms(100);
        }
        if (cancel==0)
        {
        goto menu01;
        }
        if (ok==0&&menit>0&&lock==0)
        {
            goto menu03;
        }
        goto menu02;
menu03:
lcd_clear();
}
void main(void)
{
// Declare your local variables here
// Input/Output Ports initialization
// Port B initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=Out
Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T State1=0
State0=T
PORTB=0x00;
DDRB=0x02;
// Port C initialization
// Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In
// State6=T State5=T State4=T State3=P State2=P State1=P State0=P
PORTC=0x2F;
DDRC=0x00;
// Port D initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In
Func0=In
```

```c
// 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: T0 pin Falling Edge
TCCR0=0x06;
TCNT0=0x00;
// Timer/Counter 1 initialization
// Clock source: System Clock
// Clock value: 46,875 kHz
// Mode: Normal top=0xFFFF
// OC1A output: Set
// OC1B output: Discon.
// Noise Canceler: Off
// Input Capture on Falling Edge
// Timer1 Overflow Interrupt: Off
// Input Capture Interrupt: Off
// Compare A Match Interrupt: Off
// Compare B Match Interrupt: Off
TCCR1A=0xA3; //A3
TCCR1B=0x05;
TCNT1H=0x00;
TCNT1L=0x00;
ICR1H=0x00;
ICR1L=0x00;
OCR1AH=0x00;
OCR1AL=0x00;
OCR1BH=0x00;
OCR1BL=0x00;
// Timer/Counter 2 initialization
// Clock source: System Clock
// Clock value: 11,719 kHz
// Mode: Normal top=0xFF
// OC2 output: Disconnected
ASSR=0x00;
TCCR2=0x00;
TCNT2=0x8A;
OCR2=0x00;
// External Interrupt(s) initialization
// INT0: Off
// INT1: Off
MCUCR=0x00;
// Timer(s)/Counter(s) Interrupt(s) initialization
TIMSK=0x40;
// 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;
```

```c
// SPI initialization
// SPI disabled
SPCR=0x00;
// TWI initialization
// TWI disabled
TWCR=0x00;
// Alphanumeric LCD initialization
// Connections specified in the
// Project|Configure|C Compiler|Libraries|Alphanumeric LCD menu:
// RS - PORTB Bit 5
// RD - PORTB Bit 4
// EN - PORTB Bit 3
// D4 - PORTD Bit 7
// D5 - PORTD Bit 6
// D6 - PORTD Bit 5
// D7 - PORTD Bit 3
// Characters/line: 16
lcd_init(16);
// Global enable interrupts
#asm("sei")

while (1)
        {
        // Place your code here
        menuawal();
        menu();
        kerja01:
          TCCR2=0x07;
          datapwm();
          //pwm=input*10;
          frekuensi=TCNT0;
          tampilkan_timer();
          tampil_rpm();
          if (menit==0&&detik==0)
              {
                  TCCR2=0x00;
                  pwm=0;
                  detik=13;
                  delay_ms(500);
                  goto jeda;
              }
            if (lock==1)
              {
               pwm=0;
               menit=0;
               goto kerja03;
              }
          goto kerja01;
        jeda:
          TCCR2=0x07;
          tampilkan_timer();
          lcd_clear();
          lcd_gotoxy(1,1);
          lcd_putsf("ROTOR BERPUTAR");
              if (detik==0)
              {
```

```
                    TCCR2=0x00;
                    pwm=0;
                    menit=5;
                    detik=0;
                    delay_ms(500);
                    goto kerja02;
                }
        goto jeda;
        kerja02:
          lcd_clear();
          TCCR2=0x07;
          PORTB.2=1;
          tampilkan_timer();
          lcd_gotoxy(0,1);
          lcd_putsf("SELESAI BZZER ON");
          if (menit==0&&detik==0)
              {
                  TCCR2=0x00;
                  PORTB.2=0;
                  goto kerja03;
              }
          if (lock==1)
              {
               PORTB.2=0;
               goto kerja03;
              }
          goto kerja02;
        kerja03:
        lcd_clear();
         menit=0;
         detik=0;
        delay_ms(100);
        }
}
```

LAMPIRAN 1

DOKUMENTASI PEMBUATAN ALAT

LAMPIRAN 2
GAMBAR RANGKAIAN KESELURUHAN

LAMPIRAN 3
LISTING **PROGRAM KESELURUHAN**

LAMPIRAN 4

DATASHEET MICROCONTROL AVR AT MEGA8

LAMPIRAN 5
DATASHEET LCD 16X2

LAMPIRAN 6
DATASHEET MOC3021

LAMPIRAN 7
DATASHEET **BT136**