

**LAMPIRAN A**  
**FOTO ROBOT DALAM AIR**



**LAMPIRAN B**  
**PROGRAM PADA PENGONTROL**  
**ATMEGA16**

## PROGRAM UTAMA

## PROGRAM UTAMA

/\*\*\*\*\*

This program was produced by the  
CodeWizardAVR V1.25.3 Standard  
Automatic Program Generator  
© Copyright 1998-2007 Pavel Haiduc, HP InfoTech s.r.l.  
<http://www.hpinfotech.com>

Project :  
Version :  
Date : 3/27/2007  
Author : Laboratorium  
Company : Fisika dan Instrumentasi  
Comments:

Chip type : ATmega16  
Program type : Application  
Clock frequency : 11.059200 MHz  
Memory model : Small  
External SRAM size : 0  
Data Stack size : 256

\*\*\*\*\*/

```
#include <mega16.h>
#include <delay.h>
#include <stdio.h>
#include <scankeypadB.h>
```

```
unsigned char text[32];
unsigned int temp,pilihan;
```

```
// I2C Bus functions
#asm
.equ __i2c_port=0x1B ;PORTA
.equ __sda_bit=0
.equ __scl_bit=1
#endasm
#include <i2c.h>
```

```
// Alphanumeric LCD Module functions
```

```

#asm
.equ __lcd_port=0x15 ;PORTC
#endasm
#include <lcd.h>

// Declare your global variables here

temp=((float)data1*256+data2)/10;

}

// void tenggelam(void)
// {
//     kompas();
//     if (kompas>90)PORTD.2=1;
// }

void main(void)
{
// Declare your local variables here

// Input/Output Ports initialization
// Port A 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
PORTA=0x00;
DDRA=0x00;

// Port B 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
PORTB=0x00;
DDRB=0x00;

// Port C 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
PORTC=0x00;
DDRC=0x00;

// Port D initialization
// Func7=Out Func6=Out Func5=Out Func4=Out Func3=Out Func2=Out Func1=Out Func0=Out
// State7=0 State6=0 State5=0 State4=0 State3=0 State2=0 State1=0 State0=0
PORTD=0x00;
DDRD=0xFF;

```

```

// Timer/Counter 0 initialization
// Clock source: System Clock
// Clock value: Timer 0 Stopped
// Mode: Normal top=FFh
// OC0 output: Disconnected
TCCR0=0x00;
TCNT0=0x00;
OCR0=0x00;

// Timer/Counter 1 initialization
// Clock source: System Clock
// Clock value: Timer 1 Stopped
// Mode: Normal top=FFFh
// OC1A output: Discon.
// OC1B output: Discon.
// Noise Canceler: Off
// Input Capture on Falling Edge
// Timer 1 Overflow Interrupt: Off
// Input Capture Interrupt: Off
// Compare A Match Interrupt: Off
// Compare B Match Interrupt: Off
TCCR1A=0x00;
TCCR1B=0x00;
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: Timer 2 Stopped
// Mode: Normal top=FFh
// 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=0x00;

// Analog Comparator initialization
// Analog Comparator: Off
// Analog Comparator Input Capture by Timer/Counter 1: Off
ACSR=0x80;
SFIOA=0x00;

// I2C Bus initialization
i2c_init();

// LCD module initialization
lcd_init(16);
lcd_putsf("loading1...");
delay_ms(1000);
lcd_clear();
while (1)
    {

// kompas();
//
//
//
//
//
// sprintf(text,"kompas:%d",temp);
// lcd_puts(text);
// delay_ms(100);
// lcd_clear();

if(scan_keypadB()!=0)
pilihan=scan_keypadB();

    kompas();
    sprintf(text,"kompas:%d\npilihan=",temp);
    lcd_puts(text);
lcd_putchar(pilihan);
n=keytonum();
switch (pilihan)

```

```

{
  case '1':
  // PORTD.0=1;
  // delay_ms(1000);

  if(temp>4 && temp<180)
  PORTD=0x05;
  else if(temp>=180 && temp<356)PORTD=0x03;
  else PORTD=0x07;
  sprintf(text,"kompas:%d",temp);
  lcd_puts(text);
  delay_ms(5000);
  lcd_clear();
  PORTD=0x00;
  break;

  case '2':
  // PORTD.0=1;
  // delay_ms(1000);

  if(temp>94 && temp<270)
  PORTD=0x05;
  else if((temp>=270 && temp<359.9) || (temp>0 && temp<86)) PORTD=0x03;
  else PORTD=0x07;
  sprintf(text,"kompas:%d",temp);
  lcd_puts(text);
  delay_ms(5000);
  lcd_clear();
  PORTD=0x00;
  break;

  case '3':
  // PORTD.0=1;
  // delay_ms(1000);

  if(temp>184 && temp<359.9)
  PORTD=0x05;
  else if(temp>=0 && temp<176)PORTD=0x03;
  else PORTD=0x07;
  sprintf(text,"kompas:%d",temp);
  lcd_puts(text);
  delay_ms(5000);

```



```

        lcd_clear();
        PORTD=0x00;
        break;

    case '4':
//    PORTD.0=1;
//    delay_ms(1000);

        if((temp>=274 && temp<359.9)||(temp>0 && temp<90))
            PORTD=0x05;
        else if(temp>=90 && temp<266)PORTD=0x03;
        else PORTD=0x07;
        sprintf(text,"kompas:%d",temp);
        lcd_puts(text);
        delay_ms(5000);
        lcd_clear();
        PORTD=0x00;
        break;

    case '5':
        PORTD=0x03;
        delay_ms(5000);
        break;

    case '6':
        PORTD=0x00;
        delay_ms(5000);
        break;

    case '9': PORTD=0x01;break;

};

    delay_ms(100);
    lcd_clear();

//    case 4:
//    PORTD.0=1;
//    delay_ms(2000);
//    sprintf(text,"kompas:%d",temp);
//    lcd_puts(text);

```

```

//   Kompas();
//   if(temp>195 && temp<64) PORTD=0x05;
//   else if(temp>64 && temp<190)PORTD=0x03;
//   PORTD=0x07;
//   delay_ms(5000);
//   lcd_clear();
//   PORTD=0x00;
//   break;
// case 5:
//   PORTD.0=1;
//   delay_ms(2000);
//   sprintf(text,"kompas:%d",temp);
//   lcd_puts(text);
//   Kompas();
//   if(temp>3 && temp<128)
//   PORTD=0x05;
//   else if(temp>128 && temp<252)PORTD=0x03;
//   else PORTD=0x07;
//   delay_ms(500);
//   lcd_clear();
//   PORTD=0x00;
//   break;
// case 6:
//   PORTD.0=1;
//   delay_ms(2000);
//   sprintf(text,"kompas:%d",temp);
//   lcd_puts(text);
//   Kompas();
//   if(temp>67 && temp<192) PORTD=0x05;
//   else if(temp>192 && temp<61)PORTD=0x03;
//   PORTD=0x07;
//   delay_ms(500);
//   lcd_clear();
//   PORTD=0x00;
//   break;
// case 7:
//   PORTD.0=1;
//   delay_ms(2000);
//   sprintf(text,"kompas:%d",temp);
//   lcd_puts(text);
//   Kompas();
//   if(temp>131 && temp<254) PORTD=0x05;
//   else if(temp>254 && temp<125)PORTD=0x03;
//   PORTD=0x07;
//   delay_ms(500);

```

**LAMPIRAN C**  
**DATASHEET**

**Sensor Kompas (CMPS03) ..... C-1**