

### Lampiran 1

```
// LCD module initialization
lcd_init(16);
lcd_gotoxy(0,0);
lcd_putsf(" DETEKTOR BANJIR ");
lcd_gotoxy(0,1);
lcd_putsf("  MADE IN  ");
delay_ms(2000);
lcd_gotoxy(0,0);
lcd_putsf("  ATIKNO  ");
lcd_gotoxy(0,1);
lcd_putsf("  11520225  ");
delay_ms(2000);
lcd_gotoxy(0,0);
lcd_putsf("          ");
lcd_gotoxy(0,1);
lcd_putsf("LOADING...  ");
delay_ms(2000);
lcd_gotoxy(0,0);
lcd_putsf("          ");
lcd_gotoxy(0,1);
lcd_putsf("LOADING..... ");
delay_ms(3000);
lcd_gotoxy(0,0);
lcd_putsf("          ");
lcd_gotoxy(0,1);
lcd_putsf("LOADING..... ");
delay_ms(6000);
lcd_gotoxy(0,0);
lcd_putsf("          ");
lcd_gotoxy(0,1);
lcd_putsf("          ");
delay_ms(100);
lcd_gotoxy(0,0);
lcd_putsf(" Koneksi Modem ");
lcd_gotoxy(0,1);
lcd_putsf(" Loading..... ");
delay_ms(500);
lcd_gotoxy(0,0);
lcd_putsf("Detector Banjir");
lcd_gotoxy(0,1);
lcd_putsf("          ");
trigger = 0;
delay_ms(1);
```

## *Lampiran 2*

```
// Place your code here

lcd_gotoxy(0,0);
lcd_putsf("Detector Banjir");
jarak = 0;
count=0;
trigger = 0;
delay_us(100);
trigger = 1;
delay_us(15);
trigger = 0;
delay_us(10);
while (echo == 0);
while (echo == 1)
{
    count++;
}
jarak = (unsigned int)(((float)count)/65);
sprintf(buf,"jarak:%dcm ",jarak);
lcd_gotoxy(0,1);
lcd_puts(buf);
delay_ms(500);
```

## *Lampiran 3*

```
//=====
void enter(void)
{
    putchar(13);
```

```

}
void pesan()
{
if(data==2){printf("SIAGA BANJIR!!!!");}
if(data==3){printf("BAHAYA BANJIR!!!!");}
}
void SMS()
{
printf("AT+CMGS=");
putchar(34);
printf("+6287751753550");
putchar(34);
putchar(13);
pesan();
putchar(26);
delay_ms(500);
}

```

#### *Lampiran 4*

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

This program was produced by the

CodeWizardAVR V1.25.5 Professional

Automatic Program Generator

© Copyright 1998-2007 Pavel Haiduc, HP InfoTech s.r.l.

<http://www.hpinfotech.com>

Project : Detektor Banjir

Version :

Date : 08/08/2015

Author : F4CG

Company : F4CG

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>
```

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

```
#asm
```

```
.equ __lcd_port=0x15 ;PORTC
```

```
#endasm
```

```
#include <lcd.h>
```

```
// Standard Input/Output functions
```

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

```
#define trigger PORTA.0
```

```
#define echo PINA.1
```

```
#define baudrate 9600
```

```
unsigned int jarak;
```

```
unsigned int count=0;
```

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

```
char data;
```

```
char keadaan,kirim_sms;
```

```
//=====
```

```
void enter(void)
{
  putchar(13);
}

void pesan()
{
  if(data==2){printf("SIAGA BANJIR!!!!");}
  if(data==3){printf("BAHAYA BANJIR!!!!");}
  if(data==4){printf("SENSOR RUSAK!!!!");}
}

void SMS()
{
  printf("AT+CMGS=");
  putchar(34);
  printf("+6287751753550");
  putchar(34);
  putchar(13);
  pesan();
  putchar(26);
  delay_ms(500);
}

void SMS_1()
{
  printf("AT+CMGS=");
  putchar(34);
  printf("+6283845846503");
  putchar(34);
```

```

    putchar(13);
    pesan();
    putchar(26);
    delay_ms(500);
}
void SMS_A()
{
    printf("AT+CMGS=");
    putchar(34);
    printf("+6283845846503");
    putchar(34);
    putchar(13);
    pesan();
    putchar(26);
    delay_ms(500);
}
// Declare your global variables here
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=Out
    // State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=0
    PORTA=0x00;
    DDRA=0x01;
    // Port B initialization

```

```
// Func7=In Func6=In Func5=In Func4=In Func3=Out Func2=Out Func1=Out
Func0=Out

// State7=T State6=T State5=T State4=T State3=0 State2=0 State1=0 State0=0

PORTB=0x00;

DDRB=0x0F;

// 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=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

// 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=FFFFh
```

```
// 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;
// USART initialization
// Communication Parameters: 8 Data, 1 Stop, No Parity
// USART Receiver: On
// USART Transmitter: On
// USART Mode: Asynchronous
// USART Baud Rate: 9600
UCSRA=0x00;
UCSRB=0x18;
UCSRC=0x86;
UBRRH=0x00;
UBRRL=0x47;
// Analog Comparator initialization
// Analog Comparator: Off
// Analog Comparator Input Capture by Timer/Counter 1: Off
ACSR=0x80;
SFIOR=0x00;
// LCD module initialization
lcd_init(16);
lcd_gotoxy(0,0);
lcd_putsf(" DETEKTOR BANJIR ");
lcd_gotoxy(0,1);
```

```
lcd_putsf("  MADE IN  ");
delay_ms(2000);
lcd_gotoxy(0,0);
lcd_putsf("  ATIKNO  ");
lcd_gotoxy(0,1);
lcd_putsf("  11520225  ");
delay_ms(2000);
lcd_gotoxy(0,0);
lcd_putsf("          ");
lcd_gotoxy(0,1);
lcd_putsf("LOADING...  ");
delay_ms(2000);
lcd_gotoxy(0,0);
lcd_putsf("          ");
lcd_gotoxy(0,1);
lcd_putsf("LOADING.....  ");
delay_ms(3000);
lcd_gotoxy(0,0);
lcd_putsf("          ");
lcd_gotoxy(0,1);
lcd_putsf("LOADING.....");
delay_ms(6000);
lcd_gotoxy(0,0);
lcd_putsf("          ");
lcd_gotoxy(0,1);
lcd_putsf("          ");
delay_ms(100);
lcd_gotoxy(0,0);
```

```
lcd_putsf(" Koneksi Modem ");
lcd_gotoxy(0,1);
lcd_putsf(" Loading..... ");
delay_ms(500);
lcd_gotoxy(0,0);
lcd_putsf("Detector Banjir");
lcd_gotoxy(0,1);
lcd_putsf("          ");
trigger = 0;
delay_ms(1);
keadaan=0;      //keadaan normal
kirim_sms=0;    //belum pernah kirim sms
while (1)
{
    // Place your code here
    lcd_gotoxy(0,0);
    lcd_putsf("Detector Banjir");
    jarak = 0;
    count=0;
    trigger = 0;
    delay_us(100);
    trigger = 1;
    delay_us(15);
    trigger = 0;
    delay_us(10);
    while (echo == 0)
    {
        count++;
```

```

if (count>1000)
{
    //sensor rusak
    data=4;
    SMS_A();
    lcd_gotoxy(0,0);
    lcd_putsf("SENSOR RUSAK!!!!");
    while (1);
}
}
while (echo == 1)
{
    count++;
}
jarak = (unsigned int)(((float)count)/65);
sprintf(buf,"jarak:%dcm ",jarak);
lcd_gotoxy(0,1);
lcd_puts(buf);
delay_ms(500);
while((jarak<=30)&&(jarak>=21))
{
    lcd_gotoxy(0,0);
    lcd_putsf("Debit Air Normal");
    delay_ms(500);
    jarak=100;
    keadaan=0;
    kirim_sms=0;
}

```

```

while((jarak<=20)&&(jarak>=11))
{
    lcd_gotoxy(0,0);
    lcd_putsf("!!Siaga Banjir!!");
    delay_ms(500);
    jarak=100;
    if (keadaan!=1) kirim_sms=0;
    keadaan=1;    //siaga banjir
}
while((jarak<=10)&&(jarak>= 0))
{
    lcd_gotoxy(0,0);
    lcd_putsf("!Bahaya Banjir!");
    PORTB.0=1;
    delay_ms(50);
    PORTB.0=0;
    delay_ms(500);
    jarak=100;
    if (keadaan!=2) kirim_sms=0;
    keadaan=2;    //bahaya banjir
}
if ((keadaan==2) && (kirim_sms=0))
{
    data=3;
    SMS();
    SMS_1();
    kirim_sms=1;
}

```

```
if ((keadaan==1) && (kirim_sms=0))
{
    data=2;
    SMS();
    SMS_1();
    kirim_sms=1;
}
}
}
```