

Lampiran I

```

/*****
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

Project : Sistem Monitoring Pemberi Pakan Ikan Otomatis dengan Kontrol SMS
Version : 1
Date    : 24-Aug-2016
Author  : Lucky Bahar Saputro
Company : Universitas Muhammadiyah Ponorogo
Comments:

Chip type      : ATmega16
Program type   : Application
AVR Core Clock frequency: 11.059200 MHz
Memory model   : Small
External RAM size : 0
Data Stack size : 256
*****/

#include <mega16.h>
#include <delay.h>

// Alphanumeric LCD Module functions
#include <alcd.h>

// Standard Input/Output functions
#include <stdio.h>

// Declare your global variables here

void main(void)
{
// Declare your local variables here
char index=0;
int y=0,i;
// Input/Output Ports initialization
// Port A initialization
// Func7=Out Func6=Out Func5=Out Func4=Out Func3=In Func2=In Func1=In Func0=In
// State7=0 State6=0 State5=0 State4=0 State3=P State2=P State1=P State0=P
PORTA=0x0F;
DDRA=0xF0;
// 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=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=0xFF
// OC0 output: Disconnected
TCCR0=0x00;
TCNT0=0x00;
OCR0=0x00;

// Timer/Counter 1 initialization
// Clock source: System Clock
// Clock value: Timer1 Stopped
// Mode: Normal top=0xFFFF
// OC1A output: Discon.
// 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=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: 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=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;

// Alphanumeric LCD initialization
// Connections 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);

while (1)
{
    // Place your code here

    PORTA.0=0;
    Delay_ms(300);
    PORTA.0=1;
    Delay_ms(500);
    PORTA.1=0;
    Delay_ms(300);
    PORTA.1=1;
}
}

```

Lampiran 2

```

/*****
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

Project : Sistem Monitoring Pemberi Pakan Ikan Otomatis dengan Kontrol SMS
Version : 1
Date    : 24-Aug-2016
Author  : Lucky Bahar Saputro
Company : Universitas Muhammadiyah Ponorogo
Comments:

Chip type      : ATmega16
Program type   : Application
AVR Core Clock frequency: 11.059200 MHz
Memory model   : Small
External RAM size : 0
Data Stack size : 256
*****/

#include <mega16.h>
#include <delay.h>

// Alphanumeric LCD Module functions
#include <alcd.h>

// Standard Input/Output functions
#include <stdio.h>

// Declare your global variables here

void main(void)
{
// Declare your local variables here
char index=0;
int y=0,i;
// Input/Output Ports initialization
// Port A initialization
// Func7=Out Func6=Out Func5=Out Func4=Out Func3=In Func2=In Func1=In Func0=In
// State7=0 State6=0 State5=0 State4=0 State3=P State2=P State1=P State0=P
PORTA=0x0F;
DDRA=0xF0;
// 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=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=0xFF
// OC0 output: Disconnected
TCCR0=0x00;
TCNT0=0x00;
OCR0=0x00;

// Timer/Counter 1 initialization
// Clock source: System Clock
// Clock value: Timer1 Stopped
// Mode: Normal top=0xFFFF
// OC1A output: Discon.
// 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=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: 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=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;

// Alphanumeric LCD initialization
// Connections 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);

while (1)
{
    // Place your code here

    lcd_gotoxy(0,0);
    lcd_putsf(" SKRIPSI LUCKY ");
    lcd_gotoxy(0,1);
    lcd_putsf(" PAKAN IKAN ");
    delay_ms(500);
}
}

```

Lampiran 3

```

/*****
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

Project : Sistem Monitoring Pemberi Pakan Ikan Otomatis dengan Kontrol SMS
Version : 1
Date : 24-Aug-2016
Author : Lucky Bahar Saputro
Company : Universitas Muhammadiyah Ponorogo
Comments:

Chip type : ATmega16
Program type : Application
AVR Core Clock frequency: 11.059200 MHz
Memory model : Small
External RAM size : 0
Data Stack size : 256
*****/

#include <mega16.h>
#include <delay.h>

// Alphanumeric LCD Module functions
#include <alcd.h>

// Standard Input/Output functions
#include <stdio.h>
#define baud rate 9600
#define TXC 6
#define RXC 7
int isi=0,beri=0,servo,buka=0,tutup=0;
//=====
//buat variable
unsigned char dataRX=0;
unsigned char buff[33];
unsigned char no_hp[15]; //jumlah karakter no hp maksimal 15
unsigned char kode[20]; //jumlah karakter kode maksimal 20
unsigned char on_1[11]={'i','s','i',' ','p','a','k','a','n'};
unsigned char on_2[11]={'b','e','r','i',' ','p','a','k','a','n'};
unsigned char cek[13]={'c','e','k',' ','t','a','m','p','u','n','g','a','n'};
unsigned char balas_sms;

//=====

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

//=====

void kirim_no_hp(void)
{
    unsigned char a;
    unsigned char b;
    putchar('+');
    a=0;
    do
    {
        b=no_hp[a];
        putchar(b);
        a++;
    }
    while(b!==''); //+62
}

//=====

void pesan (void)
{
    if(balas_sms==0)
    {
        printf("Format salah");
    }
    if(balas_sms==1)
    {
        printf("proses isi pakan sukses,");
        enter();
        printf(" status : tampungan penuh");
        enter();
    }
}

```

```

}
if(balas_sms==2)
{
printf("proses penebaran pakan sukses,");
enter();
printf(" status : tampungan kosong");
enter();
}
if(balas_sms==3)
{
if((PINA.0==1)&&(PINA.1==1))
{
printf("tampungan pakan penuh");
enter();
}
if((PINA.0==0)&&(PINA.1==0))
{
printf("tampungan pakan kosong");
enter();
}
}
if(balas_sms==4)
{
printf("tampungan pakan masih kosong");
}
if(balas_sms==5)
{
printf("tampungan pakan sudah penuh");
}
}
//=====
//=====

void kirim_sms(void)
{
printf("AT+CMGS="); //intruksi kirim sms
putchar(34); //kirim karakter "
//kirim_no_hp();
printf("+628213350473"); //no tujuan
putchar(34);
enter(); //kirim enter
pesan(); //pesan yang di kirim
putchar(26); //kirim pesan
}
//=====

void terima_sms(void)
{
unsigned char a;
unsigned char b;
lcd_gotoxy(0,1);
lcd_putsf(" SMS MASUK ");
//while(getchar()!=','){}; //tunggu terima karakter
delay_ms(100); //tunggu 100 ms
//=====
//baca sms
printf("AT+CMGR=1"); //baca sms masuk
enter();
a=0; //isi a dengan 0
while(getchar()!=','){}; //tunggu karakter ,
while(getchar()!='+'){}; //tunggu karakter +
//=====
//simpan no hp ke var no_hp
do
{
b=getchar(); //simpan no hp ke variable
b
no_hp[a]=b; //pindahkan variable b ke
variable no_hp[a]
a++; //naikan isi a
}
while(b!=''); //tunggu karakter "
//=====
//simpan kode
while(getchar()!='\n'){}; //tunggu baris baru di
mulai
a=0; // isi a dengan 0
b=getchar(); //simpan kode ke variable
b
while(b!='\n') //tunggu baris baru di
mulai

```

```

    {
        kode[a]=b;
        b=getchar();
        a++;
        if(a==19)
        {
            b=10;
        }
    }
    delay_ms(500); //tunggu 500ms
    //=====
    printf("AT+CMGD=1"); //panggil hapus_sms
    putchar(13);
    while(getchar()!='K'){}; //tunggu terima karakter K
    while(getchar()!='0x0a'){};
    delay_ms(500); //tunggu sms di hapus
    lcd_gotoxy(0,1);
    lcd_putsf(" SMS TERBACA ");
}
void inisialisasi(void)
{
    //=====
    //seting modem agar tidak ada eco
    printf("ATE0");
    enter();
    while(getchar()!='K'){}; //tunggu terima karakter
    K/OK
    while(getchar()!='0x0a'){};
    //=====
    //seting modem format text
    printf("AT+CMGF=1");
    enter();
    while(getchar()!='K'){}; //tunggu terima karakter
    K/OK
    while(getchar()!='0x0a'){};
    //=====
    //cek modem ok
    printf("AT");
    enter();
    while(getchar()!='K'){}; //tunggu terima karakter K
    while(getchar()!='0x0a'){};
    //=====
    //hapus pesan no 1
    printf("AT+CMGD=1");
    enter();
    while(getchar()!='K'){}; //tunggu terima karakter K
    while(getchar()!='0x0a'){};
    //=====
    //seting ada sms modem langsung kirim ke mikro
    printf("AT+CNMI=1,1,0,0,0");
    enter();
    while(getchar()!='K'){}; //tunggu terima karakter K
    while(getchar()!='0x0a'){};
}

char isi_pakan(void)
{
    //isi pakan
    unsigned char e;
    e=0;
    lcd_clear();
    while((kode[0]==on_1[0])&&(kode[1]==on_1[1])&&(kode[2]==on_1[2])&&(kode[3]==on_1[3])
    &&(kode[4]==on_1[4])&&(kode[5]==on_1[5])&&(kode[6]==on_1[6])&&(kode[7]==on_1[7])
    &&(kode[8]==on_1[8]))
    {
        lcd_gotoxy(0,0);
        lcd_putsf(" ISI PAKAN ");

        if((PINA.0==1)&&(PINA.1==1))
        {
            isi=1;
        }

        if((PINA.0==0)&&(PINA.1==0))
        {
            isi=2;
            tutup=1;
        }

        e=0xff;
        e=0xff;
        break;
    }
}

```

```

}
return(e);
}
//=====
char beri_pakan(void)
{
//beri pakan
unsigned char e;
e=0;

while((kode[0]==on_2[0])&&(kode[1]==on_2[1])&&(kode[2]==on_2[2])&&(kode[3]==on_2[3])
&&(kode[4]==on_2[4])&&(kode[5]==on_2[5])&&(kode[6]==on_2[6])&&(kode[7]==on_2[7])
&&(kode[8]==on_2[8])&&(kode[9]==on_2[9]))
{
lcd_gotoxy(0,0);
lcd_putsf("  BERI PAKAN  ");

if((PINA.0==1)&&(PINA.1==1))
{
beri=1;
buka=1;
}

if((PINA.0==0)&&(PINA.1==0))
{
beri=2;
}

e=0xff;
e=0xff;
break;
}
return(e);
}
//=====
char cek_tampungan(void)
{
//cek tampungan
unsigned char e;
e=0;

while((kode[0]==cek[0])&&(kode[1]==cek[1])&&(kode[2]==cek[2])&&(kode[3]==cek[3])&&
(kode[4]==cek[4])&&(kode[5]==cek[5])&&(kode[6]==cek[6])&&(kode[7]==cek[7])&&(kode[8]
==cek[8])&&(kode[9]==cek[9])&&(kode[10]==cek[10])&&(kode[11]==cek[11])&&(kode[12]
==cek[12]))
{
lcd_gotoxy(0,0);
lcd_putsf("  CEK TAMPUNGAN  ");
balas_sms=3;
e=0xff;
e=0xff;
break;
}
return(e);
}

// Declare your global variables here

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

// 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;

// Alphanumeric LCD initialization
// Connections 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);
lcd_gotoxy(0,0);
lcd_putsf(" SKRIPSI__LUCKY ");
lcd_gotoxy(0,1);
lcd_putsf("___PAKAN IKAN___");
delay_ms(2000);
lcd_clear();
lcd_gotoxy(0,0);
lcd_putsf("CEK KONEKSI MDM ");
lcd_gotoxy(0,1);
lcd_putsf("LOADING.      ");
delay_ms(2000);
lcd_gotoxy(0,1);
lcd_putsf("LOADING..      ");
delay_ms(2000);
lcd_gotoxy(0,1);
lcd_putsf("LOADING...     ");
delay_ms(2000);
lcd_gotoxy(0,1);
lcd_putsf("LOADING....    ");
delay_ms(2000);
lcd_gotoxy(0,1);
lcd_putsf("LOADING.....   ");
delay_ms(2000);
lcd_gotoxy(0,1);
lcd_putsf("LOADING.....   ");
delay_ms(2000);
lcd_gotoxy(0,1);
lcd_putsf("LOADING.....   ");
delay_ms(2000);
lcd_gotoxy(0,1);
lcd_putsf("LOADING.....   ");
delay_ms(2000);
lcd_gotoxy(0,1);
lcd_putsf("LOADING.....   ");
delay_ms(2000);
inialisasi();
lcd_gotoxy(0,1);
lcd_putsf(" MODEM READY  ");
delay_ms(2000);

if((PINA.0==0)&&(PINA.1==0))
{
balas_sms=4;
kirim_sms();
delay_ms(10000);
}

if((PINA.0==1)&&(PINA.1==1))
{
balas_sms=5;
kirim_sms();
delay_ms(10000);
}

if((UCSRA &(1<<RXC))){dataRX=UDR;}
sprintf(buff, "UDR:%i",dataRX);
lcd_gotoxy(0,1);
lcd_puts(buff);
dataRX=0;
lcd_clear();

while (1)
{
// Place your code here

while(y==0)
{
lcd_gotoxy(0,0);
lcd_putsf("      WAITING  ");
lcd_gotoxy(0,1);
lcd_putsf("SMS [.....] ");
while(dataRX==13)
{
lcd_gotoxy(0,1);
lcd_putsf("SMS [RECEIVED] ");
delay_ms(1000);
}
}
}

```

```

//=====
a=isi_pakan();
while(a==0xff)
{
while(isi==2)
{
if(tutup==1)
{

for(servo=0;servo<200;servo++) //tutup servo
{
PORTA.6=1;
delay_us(700);
PORTA.6=0;
delay_us(20);
}
tutup=0;

lcd_gotoxy(0,1);
lcd_putsf(" MOTOR A ON "); //motor A jalan isi pakan
PORTA.4=1;
delay_ms(30000);
PORTA.4=0;
delay_ms(1000);
if((PINA.0==1)&&(PINA.1==1)){a=0x0f;isi=0;}
balas_sms=1;
y=0;
}
}
}

//=====
a=beri_pakan();
while(a==0xff)
{
while(beri==1)
{

lcd_gotoxy(0,1);
lcd_putsf(" MOTOR B ON "); //motor B jalan beri pakan
PORTA.5=1;
delay_ms(2000);
if(buka==1)
{
delay_ms(2000);
for(servo=0;servo<200;servo++) //buka servo
{
PORTA.6=1;
delay_us(900);
PORTA.6=0;
delay_ms(30);
}
buka=0;
}
}
}

//=====
a=cek_tampungan();
if(a==0xff)
{
lcd_gotoxy(0,1);
lcd_putsf("PROSES KIRIM SMS");
if((PINA.0==0)&&(PINA.1==0))
{
balas_sms=4;
}

if((PINA.0==1)&&(PINA.1==1))
{
balas_sms=5;
}
y=0;
}
}

//=====
}
}

```

Lampiran 4



Gambar alat tampak samping



Gambar alat tampak depan & belakang