

## Lampiran

### Program Penerima Data (*Receiver*)

```
/*  
This program was produced by the  
CodeWizardAVR V2.03.4 Standard  
Automatic Program Generator  
© Copyright 1998-2008 Pavel Haiduc, HP InfoTech s.r.l.  
http://www.hpinfotech.com
```

```
Project :  
Version :  
Date    : 5/2/2017  
Author  :  
Company :  
Comments:
```

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

```
*/
```

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

```
// Alphanumeric LCD Module functions  
#asm  
    .equ __lcd_port=0x15 ;PORTC  
#endasm
```

```
#include <lcd.h>
int i,j=0,status_a,status_b,status_c,status_d;
int data_a=0;
int data_b=0;
int data_c=0;
int data_d=0;
char buff[33];

// Declare your global variables here
void bunyi_satu(void)
{
PORTD.6=1;
delay_ms(200);
PORTD.6=0;
delay_ms(100);
}

void bunyi_terus(void)
{
PORTD.6=1;
delay_ms(200);
}

void stop_bunyi(void)
{
PORTD.6=0;
delay_ms(100);
}

void status(void)
{
if(PINA.2==1){j=1;data_a=0;}
```

```
if(PINA.3==1){j=1;data_b=0;}
if(PINA.4==1){j=1;data_c=0;}
if(PINA.5==1){j=1;data_d=0;}

if(j==1)
{
for(i=0;i<500;i++)
{
if(PINA.2==1){data_a++;delay_ms(200);bunyi_satu();i=0;}
if(PINA.3==1){data_b++;delay_ms(200);bunyi_satu();i=0;}
if(PINA.4==1){data_c++;delay_ms(200);bunyi_satu();i=0;}
if(PINA.5==1){data_d++;delay_ms(200);bunyi_satu();i=0;}
delay_ms(10);
}
status_a=data_a;
status_b=data_b;
status_c=data_c;
status_d=data_d;

j=0;
}
}

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=P State4=P State3=P
State2=P State1=T State0=T
PORTA=0x3C;
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=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;

// 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(" SKRIPSI  TOTOK ");
lcd_gotoxy(0,1);
lcd_putsf("  TAHUN  2017  ");
delay_ms(2000);
lcd_clear();
```

```
lcd_gotoxy(0,0);
lcd_putsf("R1 R2 R3 R4 ");
while (1)
{
    // Place your code here
status();

switch(status_a)
{
case 1: lcd_gotoxy(0,1); lcd_putsf("0%");stop_bunyi();break;
case 2: lcd_gotoxy(0,1); lcd_putsf("5%");bunyi_terus();break;
case 3: lcd_gotoxy(0,1); lcd_putsf("10%");bunyi_satu();break;
case 4: lcd_gotoxy(0,1); lcd_putsf("20%");bunyi_satu();break;
case 5: lcd_gotoxy(0,1); lcd_putsf("30% ");break;
case 6: lcd_gotoxy(0,1); lcd_putsf("40% ");break;
case 7: lcd_gotoxy(0,1); lcd_putsf("50% ");break;
case 8: lcd_gotoxy(0,1); lcd_putsf("60% ");break;
case 9: lcd_gotoxy(0,1); lcd_putsf("70% ");break;
case 10: lcd_gotoxy(0,1); lcd_putsf("80% ");break;
case 11: lcd_gotoxy(0,1); lcd_putsf("90% ");break;
case 12: lcd_gotoxy(0,1); lcd_putsf("100%");break;
}

switch(status_c)
{
case 1: lcd_gotoxy(4,1); lcd_putsf("0%");stop_bunyi();break;
```

```
case 2: lcd_gotoxy(4,1); lcd_putsf("5%
");bunyi_terus();break;
case 3: lcd_gotoxy(4,1); lcd_putsf("10%
");bunyi_satu();break;
case 4: lcd_gotoxy(4,1); lcd_putsf("20%
");bunyi_satu();break;
case 5: lcd_gotoxy(4,1); lcd_putsf("30% ");break;
case 6: lcd_gotoxy(4,1); lcd_putsf("40% ");break;
case 7: lcd_gotoxy(4,1); lcd_putsf("50% ");break;
case 8: lcd_gotoxy(4,1); lcd_putsf("60% ");break;
case 9: lcd_gotoxy(4,1); lcd_putsf("70% ");break;
case 10: lcd_gotoxy(4,1); lcd_putsf("80% ");break;
case 11: lcd_gotoxy(4,1); lcd_putsf("90% ");break;
case 12: lcd_gotoxy(4,1); lcd_putsf("100%");break;
}

switch(status_b)
{
case 1: lcd_gotoxy(8,1); lcd_putsf("0%
");stop_bunyi();break;
case 2: lcd_gotoxy(8,1); lcd_putsf("5%
");bunyi_terus();break;
case 3: lcd_gotoxy(8,1); lcd_putsf("10%
");bunyi_satu();break;
case 4: lcd_gotoxy(8,1); lcd_putsf("20%
");bunyi_satu();break;
case 5: lcd_gotoxy(8,1); lcd_putsf("30% ");break;
case 6: lcd_gotoxy(8,1); lcd_putsf("40% ");break;
case 7: lcd_gotoxy(8,1); lcd_putsf("50% ");break;
case 8: lcd_gotoxy(8,1); lcd_putsf("60% ");break;
case 9: lcd_gotoxy(8,1); lcd_putsf("70% ");break;
case 10: lcd_gotoxy(8,1); lcd_putsf("80% ");break;
```



```
case 11: lcd_gotoxy(8,1); lcd_putsf("90% ");break;
case 12: lcd_gotoxy(8,1); lcd_putsf("100%");break;
    }

switch(status_d)
    {
case      1:      lcd_gotoxy(12,1);      lcd_putsf("0%
");stop_bunyi();break;
case      2:      lcd_gotoxy(12,1);      lcd_putsf("5%
");bunyi_terus();break;
case      3:      lcd_gotoxy(12,1);      lcd_putsf("10%
");bunyi_satu();break;
case      4:      lcd_gotoxy(12,1);      lcd_putsf("20%
");bunyi_satu();break;
case 5: lcd_gotoxy(12,1); lcd_putsf("30% ");break;
case 6: lcd_gotoxy(12,1); lcd_putsf("40% ");break;
case 7: lcd_gotoxy(12,1); lcd_putsf("50% ");break;
case 8: lcd_gotoxy(12,1); lcd_putsf("60% ");break;
case 9: lcd_gotoxy(12,1); lcd_putsf("70% ");break;
case 10: lcd_gotoxy(12,1); lcd_putsf("80% ");break;
case 11: lcd_gotoxy(12,1); lcd_putsf("90% ");break;
case 12: lcd_gotoxy(12,1); lcd_putsf("100%");break;
    }
};
}
```

## PROGRAM PEMANCAR (TRANSMITTER)

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

This program was produced by the  
CodeWizardAVR V2.03.4 Standard  
Automatic Program Generator  
© Copyright 1998-2008 Pavel Haiduc, HP InfoTech s.r.l.  
<http://www.hpinfotech.com>

Project :

Version :

Date : 5/2/2017

Author :

Company :

Comments:

Chip type : ATmega8  
Program type : Application  
Clock frequency : 11.059200 MHz  
Memory model : Small  
External RAM size : 0  
Data Stack size : 256

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

```
#include <mega8.h>  
#include <delay.h>  
#include <stdio.h>  
#include <stdbool.h>  
#include <stdlib.h>  
#include <io.h>
```

```
char text[16];  
char buf[33];
```

```
#define HX711_SCK PORTB.5
#define HX711_DT PINB.4
#define HIGH 1
#define LOW 0

// Alphanumeric LCD Module functions
#asm
    .equ __lcd_port=0x12 ;PORTD
#endasm
#include <lcd.h>

// Declare your global variables here
long HX711_Buffer=0;
long Weight_Maopi=0, Weight_Shiwu=0;
float Weight=0,berat=0;

unsigned long HX711_Read(void)
{
    unsigned long cout;
    unsigned char i;
    bool flag =0;
    HX711_DT=HIGH;
    delay_us(1);
    HX711_SCK=LOW;
    delay_us(1);

    cout=0;
    while(HX711_DT);
    for(i=0;i<24;i++)
    {
        HX711_SCK=HIGH;
        delay_us(1);
```

```
cout=cout<<1;
HX711_SCK=LOW;
delay_us(1);
if(HX711_DT) cout++;
}
HX711_SCK=HIGH;
cout ^=0x800000;
delay_us(1);
HX711_SCK=LOW;
delay_us(1);
return (cout);
}

void Get_Maopi()
{
HX711_Buffer=HX711_Read();
Weight_Maopi=HX711_Buffer/100;
}

unsigned int Get_Weight()
{
HX711_Buffer=HX711_Read();
HX711_Buffer=HX711_Buffer/100;

Weight_Shivu=HX711_Buffer;
Weight_Shivu=Weight_Shivu-Weight_Maopi;
Weight_Shivu=(unsigned
int)((float)Weight_Shivu/7.35+0.05);
return Weight_Shivu;

}
```

```

void main(void)
{
// Declare your local variables here
int loop;
int
level_1=0,level_2=0,level_3=0,level_4=0,level_5=0,level
_6=0,level_7=0,level_8=0,level_9=0,level_10=0;
int j;
int darurat_1=0,darurat_2=0;
// Input/Output Ports initialization
// Port B initialization
// Func7=In Func6=In Func5=Out Func4=In Func3=In
Func2=In Func1=Out Func0=In
// State7=T State6=T State5=0 State4=T State3=T
State2=T State1=0 State0=T
PORTB=0x00;
DDRB=0x22;

// Port C initialization
// Func6=In Func5=In Func4=In Func3=In Func2=In
Func1=In Func0=In
// 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
```

```
TCCR0=0x00;
```

```
TCNT0=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
MCUCR=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;
SFIOR=0x00;

// LCD module initialization
lcd_init(16);
lcd_gotoxy(0,0);
lcd_putsf(" SKRIPSI  TOTOK ");
delay_ms(2000);
lcd_clear();
while (1)
{
```

```
// Place your code here
for(loop=0;loop<500;loop++)
{
    Weight=Get_Weight();
    lcd_gotoxy(0,0);
    sprintf(buf,"%0.0001f",Weight);
    //ftoa(Weight,1,text);
    lcd_puts(buf);
    Weight=(Weight-11532)/2.597;
    //Weight=Weight-1.3;
    berat=Weight;
    lcd_gotoxy(0,1);
    lcd_putsf("Infus: ");
    lcd_gotoxy(7,1);
    sprintf(buf,"%0.0001f gr ",Weight);
    //ftoa(Weight,1,text);
    lcd_puts(buf);
    delay_ms(10);

    if((Weight<=5)&&(Weight>=0))
    {

        level_1=0;level_2=0;level_3=0;level_4=0;level_5=0;level
        _6=0;level_7=0;level_8=0;level_9=0;level_10=0;
        darurat_1=0;
    }

    if(Weight>=25){darurat_2=0;}

    if(Weight>=500)
    {
        for(j=0;j<12;j++)
```



```
    {  
if(level_10==0)  
    {  
        PORTB.1=1;  
        delay_ms(500);  
        PORTB.1=0;  
        delay_ms(500);  
    }  
    }  
    level_10=1;  
    darurat_2=0;  
    }
```

```
if((Weight<=450)&&(Weight>=440))
```

```
    {  
for(j=0;j<11;j++)  
    {  
if(level_9==0)  
    {  
        PORTB.1=1;  
        delay_ms(500);  
        PORTB.1=0;  
        delay_ms(500);  
    }  
    }  
    level_9=1;  
    darurat_2=0;  
    }
```

```
if((Weight<=400)&&(Weight>=390))  
    {
```

```
for(j=0;j<10;j++)
{
if(level_8==0)
{
PORTB.1=1;
delay_ms(500);
PORTB.1=0;
delay_ms(500);
}
}
level_8=1;
darurat_2=0;
}
```

```
if((Weight<=350)&&(Weight>=340))
```

```
{
for(j=0;j<9;j++)
{
if(level_7==0)
{
PORTB.1=1;
delay_ms(500);
PORTB.1=0;
delay_ms(500);
}
}
level_7=1;
darurat_2=0;
}
```



```
if((Weight<=300)&&(Weight>=290))
{
for(j=0;j<8;j++)
{
if(level_6==0)
{
PORTB.1=1;
delay_ms(500);
PORTB.1=0;
delay_ms(500);
}
}
level_6=1;
darurat_2=0;
}

if((Weight<=250)&&(Weight>=240))
{
for(j=0;j<7;j++)
{
if(level_5==0)
{
PORTB.1=1;
delay_ms(500);
PORTB.1=0;
delay_ms(500);
}
}
level_5=1;
darurat_2=0;
}
```



```
if ((Weight<=200) && (Weight>=190))
{
for (j=0; j<6; j++)
{
if (level_4==0)
{
PORTB.1=1;
delay_ms (500);
PORTB.1=0;
delay_ms (500);
}
}
level_4=1;
darurat_2=0;
}

if ((Weight<=150) && (Weight>=140))
{
for (j=0; j<5; j++)
{
if (level_3==0)
{
PORTB.1=1;
delay_ms (500);
PORTB.1=0;
delay_ms (500);
}
}
level_3=1;
darurat_2=0;
}
```



```
if ((Weight<=100) && (Weight>=90))
{
for (j=0; j<4; j++)
{
if (level_2==0)
{
PORTB.1=1;
delay_ms (500);
PORTB.1=0;
delay_ms (500);
}
}
level_2=1;
darurat_2=0;
}

if ((Weight<=50) && (Weight>=40))
{
for (j=0; j<3; j++)
{
if (level_1==0)
{
PORTB.1=1;
delay_ms (500);
PORTB.1=0;
delay_ms (500);
}
}
level_1=1;
darurat_2=0;
}
```



```
if ((Weight<=25) && (Weight>=15))
{
for (j=0; j<2; j++)
{
if (darurat_1==0)
{
PORTB.1=1;
delay_ms (500);
PORTB.1=0;
delay_ms (500);
}
}
darurat_1=1;
darurat_2=0;
}

if ((Weight<=5) && (Weight>=0))
{
for (j=0; j<1; j++)
{
if (darurat_2==0)
{
PORTB.1=1;
delay_ms (500);
PORTB.1=0;
delay_ms (500);
}
}
darurat_2=1;
}
}
```



```
if((Weight<=0)&&(Weight>=-100))
{
for(j=0;j<1;j++)
{
if(darurat_2==0)
{
PORTB.1=1;
delay_ms(500);
PORTB.1=0;
delay_ms(500);
}
}
darurat_2=1;
};
}
```

