

Kode Program

❖ Program Pada Master

/******

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

Project :
Version :
Date : 5/1/2012
Author : F4CG
Company : F4CG
Comments:

Chip type : ATmega16
Program type : Application
Clock frequency : 7.372800 MHz
Memory model : Small
External SRAM size : 0
Data Stack size : 256

*****/

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

```
// I2C Bus functions
#asm
.equ __i2c_port=0x15 ;PORTC
.equ __sda_bit=1
.equ __scl_bit=0
#endasm
#include <i2c.h>
```

```
// DS1307 Real Time Clock functions
#include <ds1307.h>
```

```
// Alphanumeric LCD Module functions
#asm
.equ __lcd_port=0x18 ;PORTB
#endasm
#include <lcd.h>
```

```
#define RXB8 1
#define TXB8 0
#define UPE 2
#define OVR 3
#define FE 4
#define UDRE 5
#define RXC 7
```

```
#define FRAMING_ERROR (1<<FE)
#define PARITY_ERROR (1<<UPE)
#define DATA_OVERRUN (1<<OVR)
#define DATA_REGISTER_EMPTY (1<<UDRE)
#define RX_COMPLETE (1<<RXC)
```

```
// USART Receiver buffer
#define RX_BUFFER_SIZE 8
char rx_buffer[RX_BUFFER_SIZE];
```

```
#if RX_BUFFER_SIZE<256
unsigned char rx_wr_index,rx_rd_index,rx_counter;
#else
unsigned int rx_wr_index,rx_rd_index,rx_counter;
#endif
```

```
// This flag is set on USART Receiver buffer overflow
bit rx_buffer_overflow;
```

```
// USART Receiver interrupt service routine
interrupt [USART_RXC] void usart_rx_isr(void)
{
```

```
char status,data;
status=UCSRA;
data=UDR;
```

```
if(data=='a')
{
    PORTD.7=0;
}
else
{
    if(data=='b')
    {
        PORTD.7=1;
    }
}
```

```
if(data=='c')
{
    PORTD.6=0;
}
else
{
    if(data=='d')
    {
        PORTD.6=1;
    }
}
```

```
if(data=='e')
{
    PORTD.5=0;
}
else
{
    if(data=='f')
    {
        PORTD.5=1;
    }
}
```

```
if ((status & (FRAMING_ERROR | PARITY_ERROR |
DATA_OVERRUN))==0)
{
    rx_buffer[rx_wr_index]=data;
    if (++rx_wr_index == RX_BUFFER_SIZE) rx_wr_index=0;
    if (++rx_counter == RX_BUFFER_SIZE)
    {
        rx_counter=0;
        rx_buffer_overflow=1;
    };
};
}
```

```
#ifndef _DEBUG_TERMINAL_IO_
// Get a character from the USART Receiver buffer
#define _ALTERNATE_GETCHAR_
#pragma used+
char getchar(void)
{
    char data;
    while (rx_counter==0);
    data=rx_buffer[rx_rd_index];
    if (++rx_rd_index == RX_BUFFER_SIZE) rx_rd_index=0;
    #asm("cli")
    --rx_counter;
    #asm("sei")
    return data;
}
#pragma used-
#endif
```

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

```
// Declare your global variables here
```

```
// inialisasi key & buff
char key,tampil[6],buff[17];
```

```

int ikey, genap;

// inialisasi while
int
counter_waktu=0, waktu_ok=0, pindah=0, opsi1=0, opsi2=0, tampil_ja
m=0, syarat_waktu, waktu_temp, kolam1, kolam2, kolam3;

// inialisasi waktu
int ji1, ji2, mi1, mi2, di1, di2;
unsigned char j, m, d, jam, menit, detik;

// inialisasi menu
int menu1, menu2,
kol1_jtot_x1, kol1_mtoto_x1, kol1_jtot_x2, kol1_mtoto_x2, kol1_jtot_x3
, kol1_mtoto_x3, kol2_jtot_x1, kol2_mtoto_x1, kol2_jtot_x2, kol2_mtoto
_x2, kol2_jtot_x3, kol2_mtoto_x3, kol3_jtot_x1, kol3_mtoto_x1, kol3_jtot
_x2, kol3_mtoto_x2, kol3_jtot_x3, kol3_mtoto_x3,
kol1_j1_x1, kol1_j2_x1, kol1_m1_x1, kol1_m2_x1,
kol1_j1_x2, kol1_j2_x2, kol1_m1_x2, kol1_m2_x2,
kol1_j1_x3, kol1_j2_x3, kol1_m1_x3, kol1_m2_x3,
kol2_j1_x1, kol2_j2_x1, kol2_m1_x1, kol2_m2_x1,
kol2_j1_x2, kol2_j2_x2, kol2_m1_x2, kol2_m2_x2,
kol2_j1_x3, kol2_j2_x3, kol2_m1_x3, kol2_m2_x3,
kol3_j1_x1, kol3_j2_x1, kol3_m1_x1, kol3_m2_x1,
kol3_j1_x2, kol3_j2_x2, kol3_m1_x2, kol3_m2_x2,
kol3_j1_x3, kol3_j2_x3, kol3_m1_x3, kol3_m2_x3;

//-----KEYPAD-----//

unsigned char keypad()
{
    PORTA = 0b11111100;

    if(PINA.5==0) return (*);
    if(PINA.6==0) return (2);
    if(PINA.7==0) return (3);

    PORTA = 0b11111010;

    if(PINA.5==0) return (1);
    if(PINA.6==0) return (5);
    if(PINA.7==0) return (6);

    PORTA = 0b11110110;

    if(PINA.5==0) return (4);
    if(PINA.6==0) return (8);
    if(PINA.7==0) return (9);

    PORTA = 0b11101110;

    if(PINA.5==0) return (7);
    if(PINA.6==0) return (0);
    if(PINA.7==0) return (#);

    delay_ms(30);

}
//-----akhir KEYPAD-----//

//-----scan_ikey-----//
unsigned int scan_ikey()
{
    //inisial keypad ke variabel key
    key=keypad();

    //jadikan key menjadi nilai integer
    if(key=='0')
    {
        ikey=0;
    }
    else

```

```

{
    if(key=='1')
    {
        ikey=1;
    }
    else
    {
        if(key=='2')
        {
            ikey=2;
        }
        else
        {
            if(key=='3')
            {
                ikey=3;
            }
            else
            {
                if(key=='4')
                {
                    ikey=4;
                }
                else
                {
                    if(key=='5')
                    {
                        ikey=5;
                    }
                    else
                    {
                        if(key=='6')
                        {
                            ikey=6;
                        }
                        else
                        {
                            if(key=='7')
                            {
                                ikey=7;
                            }
                            else
                            {
                                if(key=='8')
                                {
                                    ikey=8;
                                }
                                else
                                {
                                    if(key=='9')
                                    {
                                        ikey=9;
                                    }
                                    else
                                    {
                                        ikey=255 ;
                                    }
                                }
                            }
                        }
                    }
                }
            }
        }
    }
}

return(ikey);
}
//-----akhir ikey-----//

//-----SETTING WAKTU-----//

void setting_waktu()
{

```

```

        counter_waktu=0;
        syarat_waktu=0;
    }

while((waktu_ok==0) && (pindah==0))
{
    //setting jam1
    while(counter_waktu==0)
    {
        scan_ikey();
        lcd_gotoxy(0,0);
        lcd_putsf("Masukkan JAM:");
        lcd_gotoxy(0,1);
        lcd_putsf("(Format 24JAM)");

        if((ikey==1 || ikey==2 || ikey==3 || ikey==4 || ikey==5 ||
ikey==6 || ikey==7 || ikey==8 || ikey==9 || ikey==0)&& pindah==0)
        {
            ji1=ikey;

            itoa(ji1,tampil);
            lcd_gotoxy(13,0);
            lcd_puts(tampil);
            pindah=pindah+1;

        }

        if((ji1!=ikey) && (pindah==1))
        {
            pindah=0;
            counter_waktu=counter_waktu+1;
        }

    }

    //akhir counter_waktu=0

    //setting jam2
    while(counter_waktu==1)
    {
        scan_ikey();
        lcd_gotoxy(0,0);
        lcd_putsf("Masukkan JAM:");
        lcd_gotoxy(0,1);
        lcd_putsf("(Format 24JAM)");

        if((ikey==1 || ikey==2 || ikey==3 || ikey==4 || ikey==5 ||
ikey==6 || ikey==7 || ikey==8 || ikey==9 || ikey==0)&& pindah==0)
        {
            ji2=ikey;

            itoa(ji2,tampil);
            lcd_gotoxy(14,0);
            lcd_puts(tampil);
            pindah=pindah+1;

        }

        if((ji2!=ikey) && (pindah==1))
        {
            //jadikan ji1 dan ji2 menjadi satu kesatuan
            j=(ji1*10)+ji2;

            if(j<=23)
            {
                lcd_clear();
                pindah=0;
                counter_waktu=counter_waktu+1;
                syarat_waktu=1;
            }
            else
            {
                lcd_clear();
                pindah=0;
            }
        }
    }
}

```

```

        counter_waktu=0;
        syarat_waktu=0;
    }

    }

    //akhir counter_waktu=1

    //setting menit1
    while(counter_waktu==2 && syarat_waktu==1)
    {
        scan_ikey();
        lcd_gotoxy(0,0);
        lcd_putsf("Masukkan MNT:");
        lcd_gotoxy(0,1);
        lcd_putsf("(Format 24JAM)");

        if((ikey==1 || ikey==2 || ikey==3 || ikey==4 || ikey==5 ||
ikey==6 || ikey==7 || ikey==8 || ikey==9 || ikey==0)&& pindah==0)
        {
            mi1=ikey;

            itoa(mi1,tampil);
            lcd_gotoxy(13,0);
            lcd_puts(tampil);
            pindah=pindah+1;

        }

        if((mi1!=ikey) && (pindah==1))
        {
            syarat_waktu=0;
            pindah=0;
            counter_waktu=counter_waktu+1;
        }

    }

    //akhir counter_waktu=2

    //setting menit2
    while(counter_waktu==3)
    {
        scan_ikey();
        lcd_gotoxy(0,0);
        lcd_putsf("Masukkan MNT:");
        lcd_gotoxy(0,1);
        lcd_putsf("(Format 24JAM)");

        if((ikey==1 || ikey==2 || ikey==3 || ikey==4 || ikey==5 ||
ikey==6 || ikey==7 || ikey==8 || ikey==9 || ikey==0)&& pindah==0)
        {
            mi2=ikey;

            itoa(mi2,tampil);
            lcd_gotoxy(14,0);
            lcd_puts(tampil);
            pindah=pindah+1;

        }

        if((mi2!=ikey) && (pindah==1))
        {
            //jadikan ji1 dan ji2 menjadi satu kesatuan
            m=(mi1*10)+mi2;

            if(m<=59)
            {
                lcd_clear();
                pindah=0;
                counter_waktu=counter_waktu+1;
                syarat_waktu=1;
            }
            else
            {
                lcd_clear();
                pindah=0;
            }
        }
    }
}

```

```

        counter_waktu=2;
        syarat_waktu=1;
    }

}

//akhir counter_waktu=3

//setting_detik1
while(counter_waktu==4 && syarat_waktu==1)
{
    scan_ikey();
    lcd_gotoxy(0,0);
    lcd_putsf("Masukkan DTK:");
    lcd_gotoxy(0,1);
    lcd_putsf("Format 24JAM");

    if((ikey==1 || ikey==2 || ikey==3 || ikey==4 || ikey==5 ||
ikey==6 || ikey==7 || ikey==8 || ikey==9 || ikey==0)&& pindah==0)
    {
        di1=ikey;

        itoa(di1,tampil);
        lcd_gotoxy(13,0);
        lcd_puts(tampil);
        pindah=pindah+1;
    }

    if((di1!=ikey) && (pindah==1))
    {
        syarat_waktu=0;
        pindah=0;
        counter_waktu=counter_waktu+1;
    }

}

//akhir counter_waktu=4

while(counter_waktu==5)
{
    scan_ikey();
    lcd_gotoxy(0,0);
    lcd_putsf("Masukkan DTK:");
    lcd_gotoxy(0,1);
    lcd_putsf("Format 24JAM");

    if((ikey==1 || ikey==2 || ikey==3 || ikey==4 || ikey==5 ||
ikey==6 || ikey==7 || ikey==8 || ikey==9 || ikey==0)&& pindah==0)
    {
        di2=ikey;

        itoa(di2,tampil);
        lcd_gotoxy(14,0);
        lcd_puts(tampil);
        pindah=pindah+1;
    }

    if((di2!=ikey) && (pindah==1))
    {
        //jadikan ji1 dan ji2 menjadi satu kesatuan
        d=(di1*10)+di2;

        if(d<=59)
        {
            lcd_clear();
            pindah=0;
            counter_waktu=counter_waktu+1;
            syarat_waktu=1;
        }
        else
        {

```

```

        lcd_clear();
        pindah=0;
        counter_waktu=4;
        syarat_waktu=1;
    }

}

//akhir counter_waktu=5

while(counter_waktu==6 && syarat_waktu==1)
{
    lcd_gotoxy(0,0);
    lcd_putsf("Anda yakin?");

    lcd_gotoxy(13,0);
    lcd_putsf("1.Y");

    //tampil jam
    itoa(j,tampil);
    lcd_gotoxy(0,1);
    lcd_puts(tampil);

    lcd_gotoxy(2,1);
    lcd_putsf(":");

    //tampil menit
    itoa(m,tampil);
    lcd_gotoxy(3,1);
    lcd_puts(tampil);

    lcd_gotoxy(5,1);
    lcd_putsf(":");

    //tampil detik
    itoa(d,tampil);
    lcd_gotoxy(6,1);
    lcd_puts(tampil);

    lcd_gotoxy(13,1);
    lcd_putsf("2.N");

    scan_ikey();

    if((ikey==1) && (opsi1==0))
    {
        waktu_temp=ikey;
        opsi1=opsi1+1;
    }

    if((waktu_temp!=ikey) && (opsi1==1))
    {
        waktu_ok=1;
        opsi1=0;
        counter_waktu=0;
        tampil_jam=1;
        lcd_clear();
    }

    if((ikey==2) && (opsi2==0))
    {
        waktu_temp=ikey;
        opsi2=opsi2+1;
    }

    if((waktu_temp!=ikey) && (opsi2==1))
    {
        waktu_ok=0;
        opsi2=0;
        counter_waktu=0;
        lcd_clear();
    }

}

//akhir counter_waktu=6

```

```

//akhir waktu_ok=0

}
//-----AKHIR SETTING WAKTU-----
//-----

//-----SETTING MENU-----
//-----

void menu()
{
    rtc_set_time(j,m,d);

    while((waktu_ok==1) && (pindah==0))
    {

        rtc_get_time(&jam,&menit,&detik);
        sprintf(buff," %d:%d:%d ",jam,menit,detik);

        //-----menu utama(tampil_jam=1)-----
        while(tampil_jam==1)
        {

            //-----menampilkan jam-----//
            format_bener
            lcd_gotoxy(0,0);
            rtc_get_time(&jam,&menit,&detik);
            sprintf(buff," %d:%d:%d ",jam,menit,detik);
            lcd_puts(buff);

            scan_ikey(); //scankey

            delay_ms(100);
            lcd_clear();

            lcd_gotoxy(0,1);
            lcd_putsf("1.Jadwal ");
            lcd_gotoxy(10,1);
            lcd_putsf("2.RUN");

            //-----akhir menampilkan jam-----//

            //-----menu utama1-----//
            if((ikey==1) && (opsi1==0))
            {
                menu1=ikey;
                opsi1=opsi1+1;
            }

            if((menu1!=ikey) && (opsi1==1))
            {

                tampil_jam=2;
                opsi1=0;
                lcd_clear();

                //-----akhir opsi menu utama1-----//

            }

            //-----menu utama2-----
            if((ikey==2) && (opsi2==0))
            {
                menu2=ikey;
                opsi2=opsi2+1;
            }

            if((menu2!=ikey) && (opsi2==1))
            {

                opsi2=0;
                lcd_gotoxy(0,0);
                lcd_putsf("Anda belum set Jadwal [!])");

                delay_ms(3000);
                lcd_clear();
            }

            //-----akhir opsi menu utama2-----
            //-----

            }

            //-----akhir
            menu_utama(tampil_jam=1)-----
            //-----

            //------(tampil=2)-----
            //-----//
            while(tampil_jam==2)
            {
                lcd_gotoxy(0,0);
                lcd_putsf("Kolam1");
                lcd_gotoxy(0,1);
                lcd_putsf("Masukkan Jam1:");

                scan_ikey();

                rtc_get_time(&jam,&menit,&detik);
                sprintf(buff," %d:%d:%d ",jam,menit,detik);

                //-----setting kolam1 jam1 x1-----
                //-----//
                if((ikey==1 || ikey==2 || ikey==0)&& pindah==0)
                {
                    kol1_j1_x1=ikey;

                    itoa(kol1_j1_x1,tampil);
                    lcd_gotoxy(14,1);
                    lcd_puts(tampil);
                    pindah=pindah+1;
                }

                if((kol1_j1_x1!=ikey) && (pindah==1))
                {
                    pindah=0;
                    tampil_jam=tampil_jam+1;
                }

                //-----akhir setting 1kolam1 jam1-----
                //-----//

                }

                //-----akhir tampil=2-----
                //-----//

                //-----tampil=3-----
                //-----//

```

```

while(tampil_jam==3)
{
    lcd_gotoxy(0,0);
    lcd_putsf("Kolam1");
    lcd_gotoxy(0,1);
    lcd_putsf("Masukkan Jam1:");

    scan_ikey();

    rtc_get_time(&jam,&menit,&detik);
    sprintf(buff," %d:%d:%d ",jam,menit,detik);

    //-----setting 1kolam1 jam2-----
    -----//

    if((ikey==1 || ikey==2 || ikey==3 || ikey==4 || ikey==5
|| ikey==6 || ikey==7 || ikey==8 || ikey==9 || ikey==0)&&
pindah==0)
    {
        kol1_j2_x1=ikey;

        itoa(kol1_j2_x1,tampil);
        lcd_gotoxy(15,1);
        lcd_puts(tampil);
        pindah=pindah+1;
    }

    if((kol1_j2_x1!=ikey) && (pindah==1))
    {
        kol1_jtot_x1=(kol1_j1_x1*10)+kol1_j2_x1;

        if(kol1_jtot_x1<=23)
        {
            pindah=0;
            tampil_jam=tampil_jam+1;
            lcd_clear();
        }
        else
        {
            pindah=0;
            tampil_jam=2;
            lcd_clear();
        }
    }

    //-----1kolam1 jam2-----
    -----//

    }
    //-----akhir tampil=3-----
    -----//

    //-----kolam1 x1-----
    -----//

    //-----tampiljam=4-----
    -----//

    while(tampil_jam==4)
    {
        lcd_gotoxy(0,0);
        lcd_putsf("Kolam1");
        lcd_gotoxy(0,1);
        lcd_putsf("Masukkan Mnt1:");

        scan_ikey();

        rtc_get_time(&jam,&menit,&detik);
        sprintf(buff," %d:%d:%d ",jam,menit,detik);

        //-----setting kolam1 menit1 x1-----
        -----//

        if((ikey==1 || ikey==2 || ikey==3 || ikey==4 || ikey==5
|| ikey==6 || ikey==7 || ikey==8 || ikey==9 || ikey==0)&&
pindah==0)

```

```

{
    kol1_m1_x1=ikey;

    itoa(kol1_m1_x1,tampil);
    lcd_gotoxy(14,1);
    lcd_puts(tampil);
    pindah=pindah+1;
}

if((kol1_m1_x1!=ikey) && (pindah==1))
{
    pindah=0;
    tampil_jam=tampil_jam+1;
}

//-----akhir setting 1kolam1 m1-----
-----//

}
//-----akhir tampiljam=4-----
-----//

//-----tampiljam=5-----
-----//

while(tampil_jam==5)
{
    lcd_gotoxy(0,0);
    lcd_putsf("Kolam1");
    lcd_gotoxy(0,1);
    lcd_putsf("Masukkan Mnt1:");

    scan_ikey();

    rtc_get_time(&jam,&menit,&detik);
    sprintf(buff," %d:%d:%d ",jam,menit,detik);

    //-----setting 1kolam1 menit2-----
    -----//

    if((ikey==1 || ikey==2 || ikey==3 || ikey==4 || ikey==5
|| ikey==6 || ikey==7 || ikey==8 || ikey==9 || ikey==0)&&
pindah==0)
    {
        kol1_m2_x1=ikey;

        itoa(kol1_m2_x1,tampil);
        lcd_gotoxy(15,1);
        lcd_puts(tampil);
        pindah=pindah+1;
    }

    if((kol1_m2_x1!=ikey) && (pindah==1))
    {
        kol1_mtoto_x1=(kol1_m1_x1*10)+kol1_m2_x1;

        if(kol1_mtoto_x1<=59)
        {
            pindah=0;
            tampil_jam=tampil_jam+1;
            lcd_clear();
        }
        else
        {
            pindah=0;
            tampil_jam=4;
            lcd_clear();
        }
    }

    //-----1kolam1 menit2-----
    -----//

```

```

    }
    //-----akhir tampiljam=5-----
    -----//

    //-----tampil_jam6-----
    -----//
    while(tampil_jam==6)
    {
        lcd_gotoxy(0,0);
        lcd_putsf("Anda yakin?");

        lcd_gotoxy(13,0);
        lcd_putsf("1.Y");

        //tampil jam
        itoa(kol1_jtot_x1,tampil);
        lcd_gotoxy(0,1);
        lcd_puts(tampil);

        lcd_gotoxy(2,1);
        lcd_putsf(":");

        //tampil menit
        itoa(kol1_mt看_x1,tampil);
        lcd_gotoxy(3,1);
        lcd_puts(tampil);

        lcd_gotoxy(13,1);
        lcd_putsf("2.N");

        scan_ikey();

        rtc_get_time(&jam,&menit,&detik);
        sprintf(buff," %d:%d:%d ",jam,menit,detik);

        if((ikey==1) && (opsi1==0))
        {
            waktu_temp=ikey;
            opsi1=opsi1+1;
        }

        if((waktu_temp!=ikey) && (opsi1==1))
        {
            opsi1=0;
            waktu_temp=0;
            tampil_jam=7;
            lcd_clear();
        }

        if((ikey==2) && (opsi2==0))
        {
            waktu_temp=ikey;
            opsi2=opsi2+1;
        }

        if((waktu_temp!=ikey) && (opsi2==1))
        {
            opsi2=0;
            waktu_temp=0;
            tampil_jam=2;
            lcd_clear();
        }

    }

    //-----akhir tampil_jam6-----
    -----//

    //-----akhir kolom1 x1-----
    -----//

```

```

    //-----kolam1 x2-----
    -----//

    //------(tampil=7)-----
    -----//
    while(tampil_jam==7)
    {
        lcd_gotoxy(0,0);
        lcd_putsf("Kolam1");
        lcd_gotoxy(0,1);
        lcd_putsf("Masukkan Jam2:");

        scan_ikey();

        rtc_get_time(&jam,&menit,&detik);
        sprintf(buff," %d:%d:%d ",jam,menit,detik);

        //-----setting kolom1 jam1 x2-----
        -----//

        if((ikey==1 || ikey==2 || ikey==0)&& pindah==0)
        {
            kol1_j1_x2=ikey;

            itoa(kol1_j1_x2,tampil);
            lcd_gotoxy(14,1);
            lcd_puts(tampil);
            pindah=pindah+1;
        }

        if((kol1_j1_x2!=ikey) && (pindah==1))
        {
            pindah=0;
            tampil_jam=tampil_jam+1;
        }

        //-----akhir setting kolom1 jam1 x2----
        -----//

        }
        //-----akhir tampil=7-----
        -----//

        //-----tampil=8-----
        -----//
        while(tampil_jam==8)
        {
            lcd_gotoxy(0,0);
            lcd_putsf("Kolam1");
            lcd_gotoxy(0,1);
            lcd_putsf("Masukkan Jam2:");

            scan_ikey();

            rtc_get_time(&jam,&menit,&detik);
            sprintf(buff," %d:%d:%d ",jam,menit,detik);

            //-----setting kolom1 jam2 x2-----
            -----//

            if((ikey==1 || ikey==2 || ikey==3 || ikey==4 || ikey==5
            || ikey==6 || ikey==7 || ikey==8 || ikey==9 || ikey==0)&&
            pindah==0)
            {
                kol1_j2_x2=ikey;

                itoa(kol1_j2_x2,tampil);
                lcd_gotoxy(15,1);
                lcd_puts(tampil);
                pindah=pindah+1;
            }
        }
    }

```

```

if((kol1_j2_x2!=ikey) && (pindah==1))
{
    kol1_jtot_x2=(kol1_j1_x2*10)+kol1_j2_x2;

    if(kol1_jtot_x2<=23)
    {
        pindah=0;
        tampil_jam=tampil_jam+1;
        lcd_clear();
    }
    else
    {
        pindah=0;
        tampil_jam=7;
        lcd_clear();
    }
}

//-----akhir setting kolam1 jam2 x2-----
//-----//

}
//-----akhir tampil=8-----
//-----//

//-----tampiljam=9-----
//-----//
while(tampil_jam==9)
{
    lcd_gotoxy(0,0);
    lcd_putsf("Kolam1");
    lcd_gotoxy(0,1);
    lcd_putsf("Masukkan Mnt2:");

    scan_ikey();

    rtc_get_time(&jam,&menit,&detik);
    sprintf(buff," %d:%d:%d ",jam,menit,detik);

    //-----setting kolam1 menit1 x2-----
    //-----//

    if((ikey==1 || ikey==2 || ikey==3 || ikey==4 || ikey==5
|| ikey==6 || ikey==7 || ikey==8 || ikey==9 || ikey==0)&&
pindah==0)
    {
        kol1_m1_x2=ikey;

        itoa(kol1_m1_x2,tampil);
        lcd_gotoxy(14,1);
        lcd_puts(tampil);
        pindah=pindah+1;
    }

    if((kol1_m1_x2!=ikey) && (pindah==1))
    {
        pindah=0;
        tampil_jam=tampil_jam+1;
    }

    //-----akhir setting kolam1 menit1 x2-----
    //-----//

}
//-----akhir tampiljam=9-----
//-----//

//-----tampiljam=10-----
//-----//
while(tampil_jam==10)
{
    lcd_gotoxy(0,0);

```

```

    lcd_putsf("Kolam1");
    lcd_gotoxy(0,1);
    lcd_putsf("Masukkan Mnt2:");

    scan_ikey();

    rtc_get_time(&jam,&menit,&detik);
    sprintf(buff," %d:%d:%d ",jam,menit,detik);

    //-----setting kolam1 menit2 x2-----
    //-----//

    if((ikey==1 || ikey==2 || ikey==3 || ikey==4 || ikey==5
|| ikey==6 || ikey==7 || ikey==8 || ikey==9 || ikey==0)&&
pindah==0)
    {
        kol1_m2_x2=ikey;

        itoa(kol1_m2_x2,tampil);
        lcd_gotoxy(15,1);
        lcd_puts(tampil);
        pindah=pindah+1;
    }

    if((kol1_m2_x2!=ikey) && (pindah==1))
    {
        kol1_mtot_x2=(kol1_m1_x2*10)+kol1_m2_x2;

        if(kol1_mtot_x2<=59)
        {
            pindah=0;
            tampil_jam=tampil_jam+1;
            lcd_clear();
        }
        else
        {
            pindah=0;
            tampil_jam=7;
            lcd_clear();
        }
    }

    //-----akhir kolam1 menit2 x2-----
    //-----//

}
//-----akhir tampil_jam=10-----
//-----//

//-----tampil_jam11-----
//-----//
while(tampil_jam==11)
{
    lcd_gotoxy(0,0);
    lcd_putsf("Anda yakin?");

    lcd_gotoxy(13,0);
    lcd_putsf("1.Y");

    //tampil jam
    itoa(kol1_jtot_x2,tampil);
    lcd_gotoxy(0,1);
    lcd_puts(tampil);

    lcd_gotoxy(2,1);
    lcd_putsf(":");

    //tampil menit
    itoa(kol1_mtot_x2,tampil);
    lcd_gotoxy(3,1);
    lcd_puts(tampil);

    lcd_gotoxy(13,1);
    lcd_putsf("2.N");

```

```

scan_ikey();

rtc_get_time(&jam,&menit,&detik);
sprintf(buff," %d:%d:%d ",jam,menit,detik);

if((ikey==1) && (opsi1==0))
{
    waktu_temp=ikey;
    opsi1=opsi1+1;
}

if((waktu_temp!=ikey) && (opsi1==1))
{
    opsi1=0;
    waktu_temp=0;
    tampil_jam=12;
    lcd_clear();
}

if((ikey==2) && (opsi2==0))
{
    waktu_temp=ikey;
    opsi2=opsi2+1;
}

if((waktu_temp!=ikey) && (opsi2==1))
{
    opsi2=0;
    waktu_temp=0;
    tampil_jam=7;
    lcd_clear();
}

}

//-----akhir tampil_jam11-----
//-----

//-----akhir kolom1
x2-----

//-----kolam1
x3-----

//-----(tampil=12)-----
while(tampil_jam==12)
{
    lcd_gotoxy(0,0);
    lcd_putsf("Kolam1");
    lcd_gotoxy(0,1);
    lcd_putsf("Masukkan Jam3:");

    scan_ikey();

    rtc_get_time(&jam,&menit,&detik);
    sprintf(buff," %d:%d:%d ",jam,menit,detik);

    //-----setting kolom1 jam1 x3-----
    -----

    if((ikey==1 || ikey==2 || ikey==0)&& pindah==0)
    {
        kol1_j1_x3=ikey;

        itoa(kol1_j1_x3,tampil);
        lcd_gotoxy(14,1);
        lcd_puts(tampil);
        pindah=pindah+1;
    }

    if((kol1_j1_x3!=ikey) && (pindah==1))
    {
        pindah=0;
        tampil_jam=tampil_jam+1;
    }

    //-----akhir setting kolom1 jam1 x3----
    -----

    }
    //-----akhir tampil=13-----
    -----

    //-----tampiljam=14-----
    -----

    if((ikey==1 || ikey==2 || ikey==3 || ikey==4 || ikey==5
    || ikey==6 || ikey==7 || ikey==8 || ikey==9 || ikey==0)&&
    pindah==0)
    {
        kol1_j2_x3=ikey;

        itoa(kol1_j2_x3,tampil);
        lcd_gotoxy(15,1);
        lcd_puts(tampil);
        pindah=pindah+1;
    }

    if((kol1_j2_x3!=ikey) && (pindah==1))
    {
        kol1_jtot_x3=(kol1_j1_x3*10)+kol1_j2_x3;

        if(kol1_jtot_x3<=23)
        {
            pindah=0;
            tampil_jam=tampil_jam+1;
            lcd_clear();
        }
        else
        {
            pindah=0;
            tampil_jam=12;
            lcd_clear();
        }
    }

    //-----akhir setting kolom1 jam2 x3----
    -----

    }
    //-----akhir tampil=13-----
    -----

    //-----tampiljam=14-----
    -----

```

```

while(tampil_jam==14)
{
    lcd_gotoxy(0,0);
    lcd_putsf("Kolam1");
    lcd_gotoxy(0,1);
    lcd_putsf("Masukkan Mnt3:");

    scan_ikey();

    rtc_get_time(&jam,&menit,&detik);
    sprintf(buff," %d:%d:%d ",jam,menit,detik);

    //-----setting kolam1 menit1 x3-----
    -----//

    if((ikey==1 || ikey==2 || ikey==3 || ikey==4 || ikey==5
|| ikey==6 || ikey==7 || ikey==8 || ikey==9 || ikey==0)&&
pindah==0)
    {
        kol1_m1_x3=ikey;

        itoa(kol1_m1_x3,tampil);
        lcd_gotoxy(14,1);
        lcd_puts(tampil);
        pindah=pindah+1;
    }

    if((kol1_m1_x3!=ikey) && (pindah==1))
    {
        pindah=0;
        tampil_jam=tampil_jam+1;
    }

    //-----akhir setting kolam1 menit1 x3-
    -----//

    }
    //-----akhir tampiljam=14-----
    -----//

    //-----tampiljam=15-----
    -----//

    while(tampil_jam==15)
    {
        lcd_gotoxy(0,0);
        lcd_putsf("Kolam1");
        lcd_gotoxy(0,1);
        lcd_putsf("Masukkan Mnt3:");

        scan_ikey();

        rtc_get_time(&jam,&menit,&detik);
        sprintf(buff," %d:%d:%d ",jam,menit,detik);

        //-----setting kolam1 menit2 x3-----
        -----//

        if((ikey==1 || ikey==2 || ikey==3 || ikey==4 || ikey==5
|| ikey==6 || ikey==7 || ikey==8 || ikey==9 || ikey==0)&&
pindah==0)
        {
            kol1_m2_x3=ikey;

            itoa(kol1_m2_x3,tampil);
            lcd_gotoxy(15,1);
            lcd_puts(tampil);
            pindah=pindah+1;
        }

        if((kol1_m2_x3!=ikey) && (pindah==1))
        {
            kol1_mtoto_x3=(kol1_m1_x3*10)+kol1_m2_x3;

            if(kol1_mtoto_x3<=59)
            {
                pindah=0;
                tampil_jam=tampil_jam+1;
                lcd_clear();
            }
            else
            {
                pindah=0;
                tampil_jam=14;
                lcd_clear();
            }
        }

        //-----akhir kolam1 menit2 x3-----
        -----//

        }
        //-----akhir tampil_jam=15-----
        -----//

        //-----tampil_jam16-----
        -----//

        while(tampil_jam==16)
        {
            lcd_gotoxy(0,0);
            lcd_putsf("Anda yakin?");

            lcd_gotoxy(13,0);
            lcd_putsf("1.Y");

            //tampil jam
            itoa(kol1_jtot_x3,tampil);
            lcd_gotoxy(0,1);
            lcd_puts(tampil);

            lcd_gotoxy(2,1);
            lcd_putsf(":.");

            //tampil menit
            itoa(kol1_mtoto_x3,tampil);
            lcd_gotoxy(3,1);
            lcd_puts(tampil);

            lcd_gotoxy(13,1);
            lcd_putsf("2.N");

            scan_ikey();

            rtc_get_time(&jam,&menit,&detik);
            sprintf(buff," %d:%d:%d ",jam,menit,detik);

            if((ikey==1) && (opsi1==0))
            {
                waktu_temp=ikey;
                opsi1=opsi1+1;
            }

            if((waktu_temp!=ikey) && (opsi1==1))
            {
                opsi1=0;
                waktu_temp=0;
                tampil_jam=17;
                lcd_clear();
            }

            if((ikey==2) && (opsi2==0))
            {
                waktu_temp=ikey;
                opsi2=opsi2+1;
            }
        }
    }
}

```

```

        if((waktu_temp!=ikey) && (opsi2==1))
        {

                opsi2=0;

                waktu_temp=0;

                tampil_jam=12;
                lcd_clear();

        }

}

//-----akhir tampil_jam=16-----
//-----//

//-----akhir
kolam1 x3-----//

//-----kolam2 x1--
//-----//

//-----tampil 17-----
//-----//
while(tampil_jam==17)
{
        lcd_gotoxy(0,0);
        lcd_putsf("kolam2");
        lcd_gotoxy(0,1);
        lcd_putsf("Masukkan Jam1:");

        scan_ikey();

        rtc_get_time(&jam,&menit,&detik);
        sprintf(buff," %d:%d:%d ",jam,menit,detik);

        //-----setting kolom2 jam1 x1-----
        //-----//

        if((ikey==1 || ikey==2 || ikey==0)&& pindah==0)
        {
                kol2_j1_x1=ikey;

                itoa(kol2_j1_x1,tampil);
                lcd_gotoxy(14,1);
                lcd_puts(tampil);
                pindah=pindah+1;

        }

        if((kol2_j1_x1!=ikey) && (pindah==1))
        {
                pindah=0;
                tampil_jam=tampil_jam+1;

        }

        //-----akhir tampil 17-----
        //-----//

        //-----akhir tampil 17-----
        //-----//

        //-----tampil=18-----
        //-----//
        while(tampil_jam==18)
        {

                lcd_gotoxy(0,0);
                lcd_putsf("kolam2");
                lcd_gotoxy(0,1);
                lcd_putsf("Masukkan Jam1:");

                scan_ikey();

                rtc_get_time(&jam,&menit,&detik);
                sprintf(buff," %d:%d:%d ",jam,menit,detik);

                //-----setting kolom2 menit1 x1-----
                //-----//

                if((ikey==1 || ikey==2 || ikey==3 || ikey==4 || ikey==5
                || ikey==6 || ikey==7 || ikey==8 || ikey==9 || ikey==0)&&
                pindah==0)
                {

                        kol2_j2_x1=ikey;

                        itoa(kol2_j2_x1,tampil);
                        lcd_gotoxy(15,1);
                        lcd_puts(tampil);
                        pindah=pindah+1;

                }

                if((kol2_j2_x1!=ikey) && (pindah==1))
                {
                        kol2_jtot_x1=(kol2_j1_x1*10)+kol2_j2_x1;

                        if(kol2_jtot_x1<=23)
                        {

                                pindah=0;
                                tampil_jam=tampil_jam+1;
                                lcd_clear();

                        }
                        else
                        {
                                pindah=0;
                                tampil_jam=17;
                                lcd_clear();

                        }

                }

                //-----1kolam2 jam2-----
                //-----//

                //-----akhir tampil=18-----
                //-----//

                //-----kolam2 x1-----
                //-----//

                //-----tampiljam=19-----
                //-----//
                while(tampil_jam==19)
                {

                        lcd_gotoxy(0,0);
                        lcd_putsf("kolam2");
                        lcd_gotoxy(0,1);
                        lcd_putsf("Masukkan Mnt1:");

                        scan_ikey();

                        rtc_get_time(&jam,&menit,&detik);
                        sprintf(buff," %d:%d:%d ",jam,menit,detik);

                        //-----setting kolom2 menit1 x1-----
                        //-----//

                        if((ikey==1 || ikey==2 || ikey==3 || ikey==4 || ikey==5
                        || ikey==6 || ikey==7 || ikey==8 || ikey==9 || ikey==0)&&
                        pindah==0)
                        {

                                kol2_m1_x1=ikey;

```

```

        itoa(kol2_m1_x1,tampil);
        lcd_gotoxy(14,1);
        lcd_puts(tampil);
        pindah=pindah+1;
    }

    if((kol2_m1_x1!=ikey) && (pindah==1))
    {
        pindah=0;
        tampil_jam=tampil_jam+1;
    }

    //-----akhir setting 1kolam2 m1-----
    -----//

    }
    //-----akhir tampiljam=19-----
    -----//

    //-----tampiljam=20-----
    -----//
    while(tampil_jam==20)
    {
        lcd_gotoxy(0,0);
        lcd_putsf("kolam2");
        lcd_gotoxy(0,1);
        lcd_putsf("Masukkan Mnt1:");

        scan_ikey();

        rtc_get_time(&jam,&menit,&detik);
        sprintf(buff," %d:%d:%d ",jam,menit,detik);

        //-----setting 1kolam2 menit2-----
        -----//

        if((ikey==1 || ikey==2 || ikey==3 || ikey==4 || ikey==5
|| ikey==6 || ikey==7 || ikey==8 || ikey==9 || ikey==0)&&
pindah==0)
        {
            kol2_m2_x1=ikey;

            itoa(kol2_m2_x1,tampil);
            lcd_gotoxy(15,1);
            lcd_puts(tampil);
            pindah=pindah+1;
        }

        if((kol2_m2_x1!=ikey) && (pindah==1))
        {
            kol2_mtot_x1=(kol2_m1_x1*10)+kol2_m2_x1;

            if(kol2_mtot_x1<=59)
            {

                pindah=0;
                tampil_jam=tampil_jam+1;
                lcd_clear();
            }
            else
            {
                pindah=0;
                tampil_jam=19;
                lcd_clear();
            }
        }
    }

    //-----1kolam2 menit2-----
    -----//

```

```

    }
    //-----akhir tampiljam=20-----
    -----//

    //-----tampil_jam21-----
    -----//
    while(tampil_jam==21)
    {
        lcd_gotoxy(0,0);
        lcd_putsf("Anda yakin?");

        lcd_gotoxy(13,0);
        lcd_putsf("1.Y");

        //tampil jam
        itoa(kol2_jtot_x1,tampil);
        lcd_gotoxy(0,1);
        lcd_puts(tampil);

        lcd_gotoxy(2,1);
        lcd_putsf(":".");

        //tampil menit
        itoa(kol2_mtot_x1,tampil);
        lcd_gotoxy(3,1);
        lcd_puts(tampil);

        lcd_gotoxy(13,1);
        lcd_putsf("2.N");

        scan_ikey();

        rtc_get_time(&jam,&menit,&detik);
        sprintf(buff," %d:%d:%d ",jam,menit,detik);

        if((ikey==1) && (opsi1==0))
        {
            waktu_temp=ikey;
            opsi1=opsi1+1;
        }

        if((waktu_temp!=ikey) && (opsi1==1))
        {

            opsi1=0;
            waktu_temp=0;
            tampil_jam=22;
            lcd_clear();
        }

        if((ikey==2) && (opsi2==0))
        {
            waktu_temp=ikey;
            opsi2=opsi2+1;
        }

        if((waktu_temp!=ikey) && (opsi2==1))
        {

            opsi2=0;
            waktu_temp=0;
            tampil_jam=17;
            lcd_clear();
        }
    }

    //-----akhir tampil_jam 21-----
    -----//

    //-----akhir kolom2 x1-----
    -----//

```

```

//-----kolam2 x2-----
//

//-----tampil=22-----
//
while(tampil_jam==22)
{
    lcd_gotoxy(0,0);
    lcd_putsf("kolam2");
    lcd_gotoxy(0,1);
    lcd_putsf("Masukkan Jam2:");

    scan_ikey();

    rtc_get_time(&jam,&menit,&detik);
    sprintf(buff," %d:%d:%d ",jam,menit,detik);

    //-----setting kolom2 jam1 x2-----
    //

    if((ikey==1 || ikey==2 || ikey==0)&& pindah==0)
    {
        kol2_j1_x2=ikey;

        itoa(kol2_j1_x2,tampil);
        lcd_gotoxy(14,1);
        lcd_puts(tampil);
        pindah=pindah+1;
    }

    if((kol2_j1_x2!=ikey) && (pindah==1))
    {
        pindah=0;
        tampil_jam=tampil_jam+1;
    }

    //-----akhir setting kolom2 jam1 x2-----
    //

}
//-----akhir tampil=22-----
//

//-----tampil=23-----
//
while(tampil_jam==23)
{
    lcd_gotoxy(0,0);
    lcd_putsf("kolam2");
    lcd_gotoxy(0,1);
    lcd_putsf("Masukkan Jam2:");

    scan_ikey();

    rtc_get_time(&jam,&menit,&detik);
    sprintf(buff," %d:%d:%d ",jam,menit,detik);

    //-----setting kolom2 jam2 x2-----
    //

    if((ikey==1 || ikey==2 || ikey==3 || ikey==4 || ikey==5
|| ikey==6 || ikey==7 || ikey==8 || ikey==9 || ikey==0)&&
pindah==0)
    {
        kol2_j2_x2=ikey;

        itoa(kol2_j2_x2,tampil);
        lcd_gotoxy(15,1);
        lcd_puts(tampil);
        pindah=pindah+1;
    }

    if((kol2_j2_x2!=ikey) && (pindah==1))
    {

```

```

        kol2_jtot_x2=(kol2_j1_x2*10)+kol2_j2_x2;

        if(kol2_jtot_x2<=23)
        {

            pindah=0;
            tampil_jam=tampil_jam+1;
            lcd_clear();
        }
        else
        {
            pindah=0;
            tampil_jam=22;
            lcd_clear();
        }
    }

    //-----akhir setting kolom2 jam2 x2-----
    //

}
//-----akhir tampil=23-----
//

//-----tampiljam=24-----
//
while(tampil_jam==24)
{
    lcd_gotoxy(0,0);
    lcd_putsf("kolam2");
    lcd_gotoxy(0,1);
    lcd_putsf("Masukkan Mnt2:");

    scan_ikey();

    rtc_get_time(&jam,&menit,&detik);
    sprintf(buff," %d:%d:%d ",jam,menit,detik);

    //-----setting kolom2 menit1 x2-----
    //

    if((ikey==1 || ikey==2 || ikey==3 || ikey==4 || ikey==5
|| ikey==6 || ikey==7 || ikey==8 || ikey==9 || ikey==0)&&
pindah==0)
    {
        kol2_m1_x2=ikey;

        itoa(kol2_m1_x2,tampil);
        lcd_gotoxy(14,1);
        lcd_puts(tampil);
        pindah=pindah+1;
    }

    if((kol2_m1_x2!=ikey) && (pindah==1))
    {
        pindah=0;
        tampil_jam=tampil_jam+1;
    }

    //-----akhir setting kolom2 menit1 x2-----
    //

}
//-----akhir tampiljam=24-----
//

//-----tampiljam=25-----
//
while(tampil_jam==25)
{
    lcd_gotoxy(0,0);
    lcd_putsf("kolam2");
    lcd_gotoxy(0,1);

```

```

        lcd_putsf("Masukkan Mnt2:");

        scan_ikey();

        rtc_get_time(&jam,&menit,&detik);
        sprintf(buff," %d:%d:%d ",jam,menit,detik);

        //-----setting kolam2 menit2 x2-----
        //-----

        if((ikey==1 || ikey==2 || ikey==3 || ikey==4 || ikey==5
|| ikey==6 || ikey==7 || ikey==8 || ikey==9 || ikey==0)&&
pindah==0)
        {
            kol2_m2_x2=ikey;

            itoa(kol2_m2_x2,tampil);
            lcd_gotoxy(15,1);
            lcd_puts(tampil);
            pindah=pindah+1;
        }

        if((kol2_m2_x2!=ikey) && (pindah==1))
        {
            kol2_mt2_x2=(kol2_m1_x2*10)+kol2_m2_x2;

            if(kol2_mt2_x2<=59)
            {
                pindah=0;
                tampil_jam=tampil_jam+1;
                lcd_clear();
            }
            else
            {
                pindah=0;
                tampil_jam=24;
                lcd_clear();
            }
        }

        }

        //-----akhir kolam2 menit2 x2-----
        //-----

        }

        //-----akhir tampil_jam=25-----
        //-----

        //-----tampil_jam 26-----
        //-----
        while(tampil_jam==26)
        {
            lcd_gotoxy(0,0);
            lcd_putsf("Anda yakin?");

            lcd_gotoxy(13,0);
            lcd_putsf("1.Y");

            //tampil jam
            itoa(kol2_jtot_x2,tampil);
            lcd_gotoxy(0,1);
            lcd_puts(tampil);

            lcd_gotoxy(2,1);
            lcd_putsf(":");

            //tampil menit
            itoa(kol2_mt2_x2,tampil);
            lcd_gotoxy(3,1);
            lcd_puts(tampil);

            lcd_gotoxy(13,1);
            lcd_putsf("2.N");

            scan_ikey();

```

```

        rtc_get_time(&jam,&menit,&detik);
        sprintf(buff," %d:%d:%d ",jam,menit,detik);

        if((ikey==1) && (opsi1==0))
        {
            waktu_temp=ikey;
            opsi1=opsi1+1;
        }

        if((waktu_temp!=ikey) && (opsi1==1))
        {
            opsi1=0;
            waktu_temp=0;
            tampil_jam=27;
            lcd_clear();
        }

        if((ikey==2) && (opsi2==0))
        {
            waktu_temp=ikey;
            opsi2=opsi2+1;
        }

        if((waktu_temp!=ikey) && (opsi2==1))
        {
            opsi2=0;
            waktu_temp=0;
            tampil_jam=22;
            lcd_clear();
        }

        }

        //-----akhir tampil_jam 26-----
        //-----

        //-----akhir kolam2
x2-----//

        //-----kolam2
x3-----//

        //------(tampil=27)-----
        //-----
        while(tampil_jam==27)
        {
            lcd_gotoxy(0,0);
            lcd_putsf("kolam2");
            lcd_gotoxy(0,1);
            lcd_putsf("Masukkan Jam3:");

            scan_ikey();

            rtc_get_time(&jam,&menit,&detik);
            sprintf(buff," %d:%d:%d ",jam,menit,detik);

            //-----setting kolam2 jam1 x3-----
            //-----

            if((ikey==1 || ikey==2 || ikey==0)&& pindah==0)
            {
                kol2_j1_x3=ikey;

                itoa(kol2_j1_x3,tampil);
                lcd_gotoxy(14,1);

```

```

        lcd_puts(tampil);
        pindah=pindah+1;
    }

    if((kol2_j1_x3!=ikey) && (pindah==1))
    {
        pindah=0;
        tampil_jam=tampil_jam+1;
    }

    //-----akhir setting kolom2 jam1 x3-----
    -----//

    }
    //-----akhir tampil=27-----
    -----//

    //-----tampil=28-----
    -----//
    while(tampil_jam==28)
    {
        lcd_gotoxy(0,0);
        lcd_putsf("kolam2");
        lcd_gotoxy(0,1);
        lcd_putsf("Masukkan Jam3:");

        scan_ikey();

        rtc_get_time(&jam,&menit,&detik);
        sprintf(buff," %d:%d:%d ",jam,menit,detik);

        //-----setting kolom2 jam2 x3-----
        -----//

        if((ikey==1 || ikey==2 || ikey==3 || ikey==4 || ikey==5
|| ikey==6 || ikey==7 || ikey==8 || ikey==9 || ikey==0)&&
pindah==0)
        {
            kol2_j2_x3=ikey;

            itoa(kol2_j2_x3,tampil);
            lcd_gotoxy(15,1);
            lcd_puts(tampil);
            pindah=pindah+1;
        }

        if((kol2_j2_x3!=ikey) && (pindah==1))
        {
            kol2_jtot_x3=(kol2_j1_x3*10)+kol2_j2_x3;

            if(kol2_jtot_x3<=23)
            {
                pindah=0;
                tampil_jam=tampil_jam+1;
                lcd_clear();
            }
            else
            {
                pindah=0;
                tampil_jam=27;
                lcd_clear();
            }
        }

        //-----akhir setting kolom2 jam2 x3-----
        -----//

        }
        //-----akhir tampil=28-----
        -----//

        //-----tampiljam=29-----
        -----//
        while(tampil_jam==29)
        {

```

```

        lcd_gotoxy(0,0);
        lcd_putsf("kolam2");
        lcd_gotoxy(0,1);
        lcd_putsf("Masukkan Mnt3:");

        scan_ikey();

        rtc_get_time(&jam,&menit,&detik);
        sprintf(buff," %d:%d:%d ",jam,menit,detik);

        //-----setting kolom2 menit1 x3-----
        -----//

        if((ikey==1 || ikey==2 || ikey==3 || ikey==4 || ikey==5
|| ikey==6 || ikey==7 || ikey==8 || ikey==9 || ikey==0)&&
pindah==0)
        {
            kol2_m1_x3=ikey;

            itoa(kol2_m1_x3,tampil);
            lcd_gotoxy(14,1);
            lcd_puts(tampil);
            pindah=pindah+1;
        }

        if((kol2_m1_x3!=ikey) && (pindah==1))
        {
            pindah=0;
            tampil_jam=tampil_jam+1;
        }

        //-----akhir setting kolom2 menit1 x3-
        -----//

        }
        //-----akhir tampiljam=29-----
        -----//

        //-----tampiljam=30-----
        -----//
        while(tampil_jam==30)
        {
            lcd_gotoxy(0,0);
            lcd_putsf("kolam2");
            lcd_gotoxy(0,1);
            lcd_putsf("Masukkan Mnt3:");

            scan_ikey();

            rtc_get_time(&jam,&menit,&detik);
            sprintf(buff," %d:%d:%d ",jam,menit,detik);

            //-----setting kolom2 menit2 x3-----
            -----//

            if((ikey==1 || ikey==2 || ikey==3 || ikey==4 || ikey==5
|| ikey==6 || ikey==7 || ikey==8 || ikey==9 || ikey==0)&&
pindah==0)
            {
                kol2_m2_x3=ikey;

                itoa(kol2_m2_x3,tampil);
                lcd_gotoxy(15,1);
                lcd_puts(tampil);
                pindah=pindah+1;
            }

            if((kol2_m2_x3!=ikey) && (pindah==1))
            {
                kol2_mtota_x3=(kol2_m1_x3*10)+kol2_m2_x3;

                if(kol2_mtota_x3<=59)
                {

```

```

        pindah=0;
        tampil_jam=tampil_jam+1;
        lcd_clear();
    }
    else
    {
        pindah=0;
        tampil_jam=29;
        lcd_clear();
    }
}

//-----akhir kolom2 menit2 x3-----
//-----

}
//-----akhir tampil_jam=30-----
//-----

//-----tampil_jam31-----
//-----
while(tampil_jam==31)
{
    lcd_gotoxy(0,0);
    lcd_putsf("Anda yakin?");

    lcd_gotoxy(13,0);
    lcd_putsf("1.Y");

    //tampil jam
    itoa(kol2_jtot_x3,tampil);
    lcd_gotoxy(0,1);
    lcd_puts(tampil);

    lcd_gotoxy(2,1);
    lcd_putsf(":");

    //tampil menit
    itoa(kol2_mt看t_x3,tampil);
    lcd_gotoxy(3,1);
    lcd_puts(tampil);

    lcd_gotoxy(13,1);
    lcd_putsf("2.N");

    scan_ikey();

    rtc_get_time(&jam,&menit,&detik);
    sprintf(buff," %d:%d:%d ",jam,menit,detik);

    if((ikey==1) && (opsi1==0))
    {
        waktu_temp=ikey;
        opsi1=opsi1+1;
    }

    if((waktu_temp!=ikey) && (opsi1==1))
    {
        opsi1=0;
        waktu_temp=0;
        tampil_jam=32;
        lcd_clear();
    }

    if((ikey==2) && (opsi2==0))
    {
        waktu_temp=ikey;
        opsi2=opsi2+1;
    }

    if((waktu_temp!=ikey) && (opsi2==1))
    {

```

```

        opsi2=0;
        waktu_temp=0;
        tampil_jam=27;
        lcd_clear();
    }
}

//-----akhir tampil_jam=31-----
//-----

//-----akhir
kolam2 x3-----
//-----

//-----kolam3 x1--
//-----

//-----tampil 32-----
//-----
while(tampil_jam==32)
{
    lcd_gotoxy(0,0);
    lcd_putsf("kolam3");
    lcd_gotoxy(0,1);
    lcd_putsf("Masukkan Jam1:");

    scan_ikey();

    rtc_get_time(&jam,&menit,&detik);
    sprintf(buff," %d:%d:%d ",jam,menit,detik);

    //-----setting kolom3 jam1 x1-----
    //-----

    if((ikey==1 || ikey==2 || ikey==0)&& pindah==0)
    {
        kol3_j1_x1=ikey;

        itoa(kol3_j1_x1,tampil);
        lcd_gotoxy(14,1);
        lcd_puts(tampil);
        pindah=pindah+1;
    }

    if((kol3_j1_x1!=ikey) && (pindah==1))
    {
        pindah=0;
        tampil_jam=tampil_jam+1;
    }

    //-----akhir tampil 32-----
    //-----

}
//-----akhir tampil 33-----
//-----

//-----tampil=18-----
//-----
while(tampil_jam==33)
{
    lcd_gotoxy(0,0);
    lcd_putsf("kolam3");
    lcd_gotoxy(0,1);
    lcd_putsf("Masukkan Jam1:");

    scan_ikey();

    rtc_get_time(&jam,&menit,&detik);
    sprintf(buff," %d:%d:%d ",jam,menit,detik);

```

```

//-----setting 1kolam3 jam2-----
//-----//

if((ikey==1 || ikey==2 || ikey==3 || ikey==4 || ikey==5
|| ikey==6 || ikey==7 || ikey==8 || ikey==9 || ikey==0)&&
pindah==0)
{
    kol3_j2_x1=ikey;

    itoa(kol3_j2_x1,tampil);
    lcd_gotoxy(15,1);
    lcd_puts(tampil);
    pindah=pindah+1;
}

if((kol3_j2_x1!=ikey) && (pindah==1))
{
    kol3_jtot_x1=(kol3_j1_x1*10)+kol3_j2_x1;

    if(kol3_jtot_x1<=23)
    {
        pindah=0;
        tampil_jam=tampil_jam+1;
        lcd_clear();
    }
    else
    {
        pindah=0;
        tampil_jam=32;
        lcd_clear();
    }
}

//-----1kolam3 jam2-----
----//

}
//-----akhir tampil=33-----
//-----//

//-----kolam3 x1-----
//-----//

//-----tampiljam=34-----
//-----//
while(tampil_jam==34)
{
    lcd_gotoxy(0,0);
    lcd_putsf("kolam3");
    lcd_gotoxy(0,1);
    lcd_putsf("Masukkan Mnt1:");

    scan_ikey();

    rtc_get_time(&jam,&menit,&detik);
    sprintf(buff," %d:%d:%d ",jam,menit,detik);

    //-----setting kolam3 menit1 x1-----
    //-----//

    if((ikey==1 || ikey==2 || ikey==3 || ikey==4 || ikey==5
|| ikey==6 || ikey==7 || ikey==8 || ikey==9 || ikey==0)&&
pindah==0)
    {
        kol3_m1_x1=ikey;

        itoa(kol3_m1_x1,tampil);
        lcd_gotoxy(14,1);
        lcd_puts(tampil);
        pindah=pindah+1;
    }

    if((kol3_m1_x1!=ikey) && (pindah==1))

```

```

{
    pindah=0;
    tampil_jam=tampil_jam+1;
}

//-----akhir setting 1kolam3 m1-----
//-----//

}
//-----akhir tampiljam=34-----
//-----//

//-----tampiljam=35-----
//-----//
while(tampil_jam==35)
{
    lcd_gotoxy(0,0);
    lcd_putsf("kolam3");
    lcd_gotoxy(0,1);
    lcd_putsf("Masukkan Mnt1:");

    scan_ikey();

    rtc_get_time(&jam,&menit,&detik);
    sprintf(buff," %d:%d:%d ",jam,menit,detik);

    //-----setting 1kolam3 menit2-----
    //-----//

    if((ikey==1 || ikey==2 || ikey==3 || ikey==4 || ikey==5
|| ikey==6 || ikey==7 || ikey==8 || ikey==9 || ikey==0)&&
pindah==0)
    {
        kol3_m2_x1=ikey;

        itoa(kol3_m2_x1,tampil);
        lcd_gotoxy(15,1);
        lcd_puts(tampil);
        pindah=pindah+1;
    }

    if((kol3_m2_x1!=ikey) && (pindah==1))
    {
        kol3_mtot_x1=(kol3_m1_x1*10)+kol3_m2_x1;

        if(kol3_mtot_x1<=59)
        {
            pindah=0;
            tampil_jam=tampil_jam+1;
            lcd_clear();
        }
        else
        {
            pindah=0;
            tampil_jam=34;
            lcd_clear();
        }
    }

    //-----1kolam3 menit2-----
    -----//

}
//-----akhir tampiljam=35-----
//-----//

//-----tampil_jam36-----
//-----//
while(tampil_jam==36)
{

```

```

        lcd_gotoxy(0,0);
        lcd_putsf("Anda yakin?");

        lcd_gotoxy(13,0);
        lcd_putsf("1.Y");

        //tampil jam
        itoa(kol3_jtot_x1,tampil);
        lcd_gotoxy(0,1);
        lcd_puts(tampil);

        lcd_gotoxy(2,1);
        lcd_putsf(":");

        //tampil menit
        itoa(kol3_mtot_x1,tampil);
        lcd_gotoxy(3,1);
        lcd_puts(tampil);

        lcd_gotoxy(13,1);
        lcd_putsf("2.N");

        scan_ikey();

        rtc_get_time(&jam,&menit,&detik);
        sprintf(buff," %d:%d:%d ",jam,menit,detik);

        if((ikey==1) && (opsi1==0))
        {
            waktu_temp=ikey;
            opsi1=opsi1+1;
        }

        if((waktu_temp!=ikey) && (opsi1==1))
        {
            opsi1=0;
            waktu_temp=0;
            tampil_jam=37;
            lcd_clear();
        }

        if((ikey==2) && (opsi2==0))
        {
            waktu_temp=ikey;
            opsi2=opsi2+1;
        }

        if((waktu_temp!=ikey) && (opsi2==1))
        {
            opsi2=0;
            waktu_temp=0;
            tampil_jam=32;
            lcd_clear();
        }

        }

        //-----akhir tampil_jam 36-----
        -----//

        //-----akhir kolom3 x1-----
        -----//

        //-----kolam3 x2-----
        -----//

        //-----tampil=37-----
        -----//
        while(tampil_jam==37)

{
    lcd_gotoxy(0,0);
    lcd_putsf("kolam3");
    lcd_gotoxy(0,1);
    lcd_putsf("Masukkan Jam2:");

    scan_ikey();

    rtc_get_time(&jam,&menit,&detik);
    sprintf(buff," %d:%d:%d ",jam,menit,detik);

    //-----setting kolom3 jam1 x2-----
    -----//

    if((ikey==1 || ikey==2 || ikey==0)&& pindah==0)
    {
        kol3_j1_x2=ikey;

        itoa(kol3_j1_x2,tampil);
        lcd_gotoxy(14,1);
        lcd_puts(tampil);
        pindah=pindah+1;
    }

    if((kol3_j1_x2!=ikey) && (pindah==1))
    {
        pindah=0;
        tampil_jam=tampil_jam+1;
    }

    //-----akhir setting kolom3 jam1 x2----
    -----//

    }
    //-----akhir tampil=37-----
    -----//

    //-----tampil=38-----
    -----//
    while(tampil_jam==38)
    {
        lcd_gotoxy(0,0);
        lcd_putsf("kolam3");
        lcd_gotoxy(0,1);
        lcd_putsf("Masukkan Jam2:");

        scan_ikey();

        rtc_get_time(&jam,&menit,&detik);
        sprintf(buff," %d:%d:%d ",jam,menit,detik);

        //-----setting kolom3 jam2 x2-----
        -----//

        if((ikey==1 || ikey==2 || ikey==3 || ikey==4 || ikey==5
        || ikey==6 || ikey==7 || ikey==8 || ikey==9 || ikey==0)&&
        pindah==0)
        {
            kol3_j2_x2=ikey;

            itoa(kol3_j2_x2,tampil);
            lcd_gotoxy(15,1);
            lcd_puts(tampil);
            pindah=pindah+1;
        }

        if((kol3_j2_x2!=ikey) && (pindah==1))
        {
            kol3_jtot_x2=(kol3_j1_x2*10)+kol3_j2_x2;

            if(kol3_jtot_x2<=23)
            {
                pindah=0;
                tampil_jam=tampil_jam+1;
                lcd_clear();
            }
        }
    }
}

```

```

        else
        {
            pindah=0;
            tampil_jam=37;
            lcd_clear();
        }

    }

    //-----akhir setting kolom3 jam2 x2-----
    -----//

    }
    //-----akhir tampil=38-----
    -----//

    //-----tampiljam=39-----
    -----//
    while(tampil_jam==39)
    {
        lcd_gotoxy(0,0);
        lcd_putsf("kolam3");
        lcd_gotoxy(0,1);
        lcd_putsf("Masukkan Mnt2:");

        scan_ikey();

        rtc_get_time(&jam,&menit,&detik);
        sprintf(buff," %d:%d:%d ",jam,menit,detik);

        //-----setting kolom3 menit1 x2-----
        -----//

        if((ikey==1 || ikey==2 || ikey==3 || ikey==4 || ikey==5
        || ikey==6 || ikey==7 || ikey==8 || ikey==9 || ikey==0)&&
        pindah==0)
        {
            kol3_m1_x2=ikey;

            itoa(kol3_m1_x2,tampil);
            lcd_gotoxy(14,1);
            lcd_puts(tampil);
            pindah=pindah+1;
        }

        if((kol3_m1_x2!=ikey) && (pindah==1))
        {
            pindah=0;
            tampil_jam=tampil_jam+1;
        }

        //-----akhir setting kolom3 menit1 x2-
        -----//

    }
    //-----akhir tampiljam=39-----
    -----//

    //-----tampiljam=40-----
    -----//
    while(tampil_jam==40)
    {
        lcd_gotoxy(0,0);
        lcd_putsf("kolam3");
        lcd_gotoxy(0,1);
        lcd_putsf("Masukkan Mnt2:");

        scan_ikey();

        rtc_get_time(&jam,&menit,&detik);
        sprintf(buff," %d:%d:%d ",jam,menit,detik);

        //-----setting kolom3 menit2 x2-----
        -----//

```

```

        if((ikey==1 || ikey==2 || ikey==3 || ikey==4 || ikey==5
        || ikey==6 || ikey==7 || ikey==8 || ikey==9 || ikey==0)&&
        pindah==0)
        {
            kol3_m2_x2=ikey;

            itoa(kol3_m2_x2,tampil);
            lcd_gotoxy(15,1);
            lcd_puts(tampil);
            pindah=pindah+1;
        }

        if((kol3_m2_x2!=ikey) && (pindah==1))
        {
            kol3_mtott_x2=(kol3_m1_x2*10)+kol3_m2_x2;

            if(kol3_mtott_x2<=59)
            {
                pindah=0;
                tampil_jam=tampil_jam+1;
                lcd_clear();
            }
            else
            {
                pindah=0;
                tampil_jam=39;
                lcd_clear();
            }
        }

        //-----akhir kolom3 menit2 x2-----
        -----//

    }
    //-----akhir tampil_jam=40-----
    -----//

    //-----tampil_jam 41-----
    -----//
    while(tampil_jam==41)
    {
        lcd_gotoxy(0,0);
        lcd_putsf("Anda yakin?");

        lcd_gotoxy(13,0);
        lcd_putsf("1.Y");

        //tampil jam
        itoa(kol3_jtott_x2,tampil);
        lcd_gotoxy(0,1);
        lcd_puts(tampil);

        lcd_gotoxy(2,1);
        lcd_putsf(":.");

        //tampil menit
        itoa(kol3_mtott_x2,tampil);
        lcd_gotoxy(3,1);
        lcd_puts(tampil);

        lcd_gotoxy(13,1);
        lcd_putsf("2.N");

        scan_ikey();

        rtc_get_time(&jam,&menit,&detik);
        sprintf(buff," %d:%d:%d ",jam,menit,detik);

        if((ikey==1) && (opsi1==0))
        {
            waktu_temp=ikey;
            opsi1=opsi1+1;
        }
    }

```

```

if((waktu_temp!=ikey) && (opsi1==1))
{
    opsi1=0;
    waktu_temp=0;
    tampil_jam=42;
    lcd_clear();
}

if((ikey==2) && (opsi2==0))
{
    waktu_temp=ikey;
    opsi2=opsi2+1;
}

if((waktu_temp!=ikey) && (opsi2==1))
{
    opsi2=0;
    waktu_temp=0;
    tampil_jam=37;
    lcd_clear();
}

}

//-----akhir tampil_jam 26-----
//-----

//-----akhir kolom3
x2-----

//-----kolam3
x3-----

//-----
//-----
//-----
while(tampil_jam==42)
{
    lcd_gotoxy(0,0);
    lcd_putsf("kolam3");
    lcd_gotoxy(0,1);
    lcd_putsf("Masukkan Jam3:");

    scan_ikey();

    rtc_get_time(&jam,&menit,&detik);
    sprintf(buff," %d:%d:%d ",jam,menit,detik);

    //-----setting kolom3 jam1 x3-----
    -----//

    if((ikey==1 || ikey==2 || ikey==0)&& pindah==0)
    {
        kol3_j1_x3=ikey;

        itoa(kol3_j1_x3,tampil);
        lcd_gotoxy(14,1);
        lcd_puts(tampil);
        pindah=pindah+1;
    }

    if((kol3_j1_x3!=ikey) && (pindah==1))
    {
        pindah=0;
        tampil_jam=tampil_jam+1;
    }

}

//-----akhir setting kolom3 jam1 x3----
//-----

//-----akhir tampil=42-----
//-----

//-----tampiljam=44-----
//-----

while(tampil_jam==44)
{
    lcd_gotoxy(0,0);
    lcd_putsf("kolam3");
    lcd_gotoxy(0,1);
    lcd_putsf("Masukkan Mnt3:");

    scan_ikey();

    rtc_get_time(&jam,&menit,&detik);
    sprintf(buff," %d:%d:%d ",jam,menit,detik);

    if((ikey==1 || ikey==2 || ikey==3 || ikey==4 || ikey==5
|| ikey==6 || ikey==7 || ikey==8 || ikey==9 || ikey==0)&&
pindah==0)
    {
        kol3_j2_x3=ikey;

        itoa(kol3_j2_x3,tampil);
        lcd_gotoxy(15,1);
        lcd_puts(tampil);
        pindah=pindah+1;
    }

    if((kol3_j2_x3!=ikey) && (pindah==1))
    {
        kol3_jtot_x3=(kol3_j1_x3*10)+kol3_j2_x3;

        if(kol3_jtot_x3<=23)
        {
            pindah=0;
            tampil_jam=tampil_jam+1;
            lcd_clear();
        }
        else
        {
            pindah=0;
            tampil_jam=42;
            lcd_clear();
        }
    }

    //-----akhir setting kolom3 jam2 x3----
    -----//

}

//-----akhir tampil=43-----
//-----

//-----tampiljam=44-----
//-----

while(tampil_jam==44)
{
    lcd_gotoxy(0,0);
    lcd_putsf("kolam3");
    lcd_gotoxy(0,1);
    lcd_putsf("Masukkan Mnt3:");

    scan_ikey();

    rtc_get_time(&jam,&menit,&detik);
    sprintf(buff," %d:%d:%d ",jam,menit,detik);

```

```

//-----setting kolom3 menit1 x3-----
//
if((ikey==1 || ikey==2 || ikey==3 || ikey==4 || ikey==5
|| ikey==6 || ikey==7 || ikey==8 || ikey==9 || ikey==0)&&
pindah==0)
{
    kol3_m1_x3=ikey;

    itoa(kol3_m1_x3,tampil);
    lcd_gotoxy(14,1);
    lcd_puts(tampil);
    pindah=pindah+1;
}

if((kol3_m1_x3!=ikey) && (pindah==1))
{
    pindah=0;
    tampil_jam=tampil_jam+1;
}

//-----akhir setting kolom3 menit1 x3-
//

}
//-----akhir tampiljam=44-----
//

//-----tampiljam=45-----
//
while(tampil_jam==45)
{
    lcd_gotoxy(0,0);
    lcd_putsf("kolam3");
    lcd_gotoxy(0,1);
    lcd_putsf("Masukkan Mnt3:");

    scan_ikey();

    rtc_get_time(&jam,&menit,&detik);
    sprintf(buff," %d:%d:%d ",jam,menit,detik);

    //-----setting kolom3 menit2 x3-----
    //

    if((ikey==1 || ikey==2 || ikey==3 || ikey==4 || ikey==5
|| ikey==6 || ikey==7 || ikey==8 || ikey==9 || ikey==0)&&
pindah==0)
    {
        kol3_m2_x3=ikey;

        itoa(kol3_m2_x3,tampil);
        lcd_gotoxy(15,1);
        lcd_puts(tampil);
        pindah=pindah+1;
    }

    if((kol3_m2_x3!=ikey) && (pindah==1))
    {
        kol3_mtот_x3=(kol3_m1_x3*10)+kol3_m2_x3;

        if(kol3_mtот_x3<=59)
        {
            pindah=0;
            tampil_jam=tampil_jam+1;
            lcd_clear();
        }
        else
        {
            pindah=0;
            tampil_jam=44;
        }
    }
}

```

```

        lcd_clear();
    }
}

//-----akhir kolom3 menit2 x3-----
//

}
//-----akhir tampil_jam=45-----
//

//-----tampil_jam46-----
//
while(tampil_jam==46)
{
    lcd_gotoxy(0,0);
    lcd_putsf("Anda yakin?");

    lcd_gotoxy(13,0);
    lcd_putsf("1.Y");

    //tampil jam
    itoa(kol3_jtot_x3,tampil);
    lcd_gotoxy(0,1);
    lcd_puts(tampil);

    lcd_gotoxy(2,1);
    lcd_putsf(":".);

    //tampil menit
    itoa(kol3_mtот_x3,tampil);
    lcd_gotoxy(3,1);
    lcd_puts(tampil);

    lcd_gotoxy(13,1);
    lcd_putsf("2.N");

    scan_ikey();

    rtc_get_time(&jam,&menit,&detik);
    sprintf(buff," %d:%d:%d ",jam,menit,detik);

    if((ikey==1) && (opsi1==0))
    {
        waktu_temp=ikey;
        opsi1=opsi1+1;
    }

    if((waktu_temp!=ikey) && (opsi1==1))
    {
        opsi1=0;
        waktu_temp=0;
        tampil_jam=47;
        lcd_clear();
    }

    if((ikey==2) && (opsi2==0))
    {
        waktu_temp=ikey;
        opsi2=opsi2+1;
    }

    if((waktu_temp!=ikey) && (opsi2==1))
    {
        opsi2=0;
        waktu_temp=47;

        tampil_jam=1;
        lcd_clear();
    }
}

```

```

    }

    //-----akhir tampil_jam=46-----
    -----//

    //-----akhir
    kolom3 x3-----//

    //-----RUN-----
    -----//

    //-----tampil_jam 47-----
    -----//
    while(tampil_jam==47)
    {
        lcd_gotoxy(0,0);
        rtc_get_time(&jam,&menit,&detik);
        sprintf(buff," %d:%d:%d ",jam,menit,detik);
        lcd_puts(buff);

        scan_ikey(); //scankey

        delay_ms(100);
        lcd_clear();

        lcd_gotoxy(0,1);
        lcd_putsf("1.Jadwal ");
        lcd_gotoxy(10,1);
        lcd_putsf("2.RUN");

        //-----menu utama1-----//
        if((ikey==1) && (opsi1==0))
        {
            menu1=ikey;
            opsi1=opsi1+1;

        }

        if((menu1!=ikey) && (opsi1==1))
        {
            tampil_jam=48;
            opsi1=0;
            lcd_clear();

        }

        //-----akhir opsi menu utama1-----//

        //-----menu utama2-----//
        if((ikey==2) && (opsi2==0))
        {
            menu2=ikey;
            opsi2=opsi2+1;

        }

        if((menu2!=ikey) && (opsi2==1))
        {
            lcd_clear();
            tampil_jam=49;
            opsi2=0;

        }

        //-----akhir opsi menu utama2-----//

    }

    //-----akhir tampil jam 47-----
    -----//

    //-----tampil_jam 48-----
    -----//
    while(tampil_jam==48)
    {
        lcd_gotoxy(0,0);
        lcd_putsf("Set ulang Jadwal");
    }

```

```

        lcd_gotoxy(0,1);
        lcd_putsf("1.Ya");
        lcd_gotoxy(8,1);
        lcd_putsf("2.Tidak");

        scan_ikey();

        //-----menu utama1-----//
        if((ikey==1) && (opsi1==0))
        {
            menu1=ikey;
            opsi1=opsi1+1;

        }

        if((menu1!=ikey) && (opsi1==1))
        {
            tampil_jam=2;
            opsi1=0;
            lcd_clear();

        }

        //-----akhir opsi menu utama1-----//

        //-----menu utama2-----
        -----//
        if((ikey==2) && (opsi2==0))
        {
            menu2=ikey;
            opsi2=opsi2+1;

        }

        if((menu2!=ikey) && (opsi2==1))
        {
            tampil_jam=47;
            opsi2=0;

        }

        //-----akhir opsi menu utama2-----
        -----//

        }

        //-----akhir tampil jam 48-----
        -----//

        //-----tampil jam 49-----
        -----//
        while(tampil_jam==49)
        {
            lcd_gotoxy(0,0);
            rtc_get_time(&jam,&menit,&detik);
            sprintf(buff," %d:%d:%d ",jam,menit,detik);
            lcd_puts(buff);

            //-----kirim jadwal-----
            -----//

            if(((jam==kol1_jtot_x1 &&
            menit==kol1_mtot_x1)||jam==kol1_jtot_x2
            menit==kol1_mtot_x2)||jam==kol1_jtot_x3
            menit==kol1_mtot_x3)) && detik==0)
            {

            }

        }

        while(1)
        {

            if(kolam1==1)
            {

```

```

}
//-----AKHIR SETTING MENU-----
//

```

```

// 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=0b11111111;
DDRA=0b00011111;

// 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=0b11100000;
DDRD=0b11100100;

// 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=0x98;
UCSRC=0x86;
UBRRH=0x00;
UBRRL=0x2F;

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

// I2C Bus initialization
i2c_init();

// DS1307 Real Time Clock initialization
// Square wave output on pin SQW/OUT: Off
// SQW/OUT pin state: 0
rtc_init(0,0,0);

// LCD module initialization
lcd_init(16);

// Global enable interrupts
#asm("sei")

while (1)
{
    // Place your code here

    // setting_waktu
    setting_waktu();

    // setting_menu
    menu();

};
}

```

❖ Program Pada Slave

SLAVE1

/******

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

Project :
Version :
Date : 4/23/2012
Author : F4CG
Company : F4CG
Comments:

Chip type : ATmega16
Program type : Application
Clock frequency : 12.000000 MHz
Memory model : Small
External SRAM size : 0
Data Stack size : 256

#include <mega16.h>


```

// Declare your global variables here
int kirim1,kirim2;

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=0xFF;

// 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=0b11000000;
DDRD=0b11000100;

// 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=0x98;
UCSRC=0x86;
UBRRH=0x00;
UBRRL=0x4D;

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

// Global enable interrupts
#asm("sei")

while (1)
{
// Place your code here

if(PINC.0==1)
{
while(1)
{

if(kirim1==1)
{
break;
}
else
{

PORTD.2=1;

putchar('a');
delay_ms(1);

PORTD.2=0;

kirim1=kirim1+1;
kirim2=0;
}

}

}

if(PINC.0==0)
{
while(1)
{

if(kirim2==1)

```

```

    {
        break;
    }
    else
    {

PORTD.2=1;

        putchar('b');
        delay_ms(1);

PORTD.2=0;

        kirim2=kirim2+1;
        kirim1=0;
    }

}

};
}

```

SLAVE2

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

Project :
Version :
Date : 4/23/2012
Author : F4CG
Company : F4CG
Comments:

Chip type : ATmega16
Program type : Application
Clock frequency : 12.000000 MHz
Memory model : Small
External SRAM size : 0
Data Stack size : 256

*****/

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

```
#define RXB8 1
#define TXB8 0
#define UPE 2
#define OVR 3
#define FE 4
#define UDRE 5
#define RXC 7
```

```
#define FRAMING_ERROR (1<<FE)
#define PARITY_ERROR (1<<UPE)
#define DATA_OVERRUN (1<<OVR)
#define DATA_REGISTER_EMPTY (1<<UDRE)
#define RX_COMPLETE (1<<RXC)
```

```
// USART Receiver buffer
#define RX_BUFFER_SIZE 8
char rx_buffer[RX_BUFFER_SIZE];

#if RX_BUFFER_SIZE<256
unsigned char rx_wr_index,rx_rd_index,rx_counter;
#else
unsigned int rx_wr_index,rx_rd_index,rx_counter;
#endif
```

```
// This flag is set on USART Receiver buffer overflow
```

```
bit rx_buffer_overflow;
```

```
// USART Receiver interrupt service routine
interrupt [USART_RXC] void usart_rx_isr(void)
{
    int i,motor;
    char status,data;
    PORTD.2=0;
```

```
status=UCSRA;
data=UDR;
```

```
if(data=='2')
{
```

```
    while(1)
    {
        if(motor==1)
        {
            break;
        }
        else
        {
```

```
//////////KIRI//////////
for(i=0;i<3;i++)
{
```

```
    PORTB=0x01;
    delay_ms(25);
    PORTB=0x02;
    delay_ms(25);
    PORTB=0x04;
    delay_ms(25);
    PORTB=0x08;
    delay_ms(25);
}
```

```
//////////KIRI//////////
```

```
//////////KANAN//////////
for(i=0;i<3;i++)
{
```

```
    PORTB=0x08;
    delay_ms(25);
    PORTB=0x04;
    delay_ms(25);
    PORTB=0x02;
    delay_ms(25);
    PORTB=0x01;
    delay_ms(25);
}
```

```
//////////KANAN//////////
```

```
//////////KIRI//////////
for(i=0;i<3;i++)
{
```

```
    PORTB=0x01;
    delay_ms(25);
    PORTB=0x02;
    delay_ms(25);
    PORTB=0x04;
    delay_ms(25);
    PORTB=0x08;
    delay_ms(25);
}
```

```
//////////KIRI//////////
```

```
//////////KANAN//////////
for(i=0;i<3;i++)
{
```

```
    PORTB=0x08;
    delay_ms(25);
    PORTB=0x04;
    delay_ms(25);
    PORTB=0x02;
```

```

        delay_ms(25);
        PORTB=0x01;
        delay_ms(25);
    }
    //////////KANAN////////

PORTB=0x00;
motor=1;

    }
}

if ((status & (FRAMING_ERROR | PARITY_ERROR |
DATA_OVERRUN))==0)
{
    rx_buffer[rx_wr_index]=data;
    if (++rx_wr_index == RX_BUFFER_SIZE) rx_wr_index=0;
    if (++rx_counter == RX_BUFFER_SIZE)
    {
        rx_counter=0;
        rx_buffer_overflow=1;
    };
};
}

#ifndef _DEBUG_TERMINAL_IO_
// Get a character from the USART Receiver buffer
#define _ALTERNATE_GETCHAR_
#pragma used+
char getchar(void)
{
    char data;
    while (rx_counter==0);
    data=rx_buffer[rx_rd_index];
    if (++rx_rd_index == RX_BUFFER_SIZE) rx_rd_index=0;
    #asm("cli")
    --rx_counter;
    #asm("sei")
    return data;
}
#pragma used-
#endif

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

// Declare your global variables here
int kirim1,kirim2;

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=0xFF;

    // 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=0b11000000;
    DDRD=0b11000100;

    // 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=0x98;
    UCSRC=0x86;
    UBRRH=0x00;
    UBRRL=0x4D;

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

```

```
// Global enable interrupts
#asm("sei")
```

```
while (1)
{
    // Place your code here
```

```
    if(PINC.0==1)
    {
        while(1)
        {
```

```
            if(kirim1==1)
            {
                break;
            }
            else
            {
```

```
                PORTD.2=1;
```

```
                putchar('c');
                delay_ms(1);
```

```
                PORTD.2=0;
```

```
                kirim1=kirim1+1;
                kirim2=0;
            }
        }
    }
```

```
}
```

```
}
```

```
if(PINC.0==0)
{
    while(1)
    {
```

```
        if(kirim2==1)
        {
            break;
        }
        else
        {
```

```
            PORTD.2=1;
```

```
            putchar('d');
            delay_ms(1);
```

```
            PORTD.2=0;
```

```
            kirim2=kirim2+1;
            kirim1=0;
        }
    }
```

```
}
```

```
}
```

```
};
```

```
}
```

SLAVE3

```
/******
```

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

Project :
Version :
Date : 4/23/2012
Author : F4CG
Company : F4CG
Comments:

Chip type : ATmega16
Program type : Application
Clock frequency : 12.000000 MHz
Memory model : Small
External SRAM size : 0
Data Stack size : 256

```
*****/
```

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

```
#define RXB8 1
#define TXB8 0
#define UPE 2
#define OVR 3
#define FE 4
#define UDRE 5
#define RXC 7
```

```
#define FRAMING_ERROR (1<<FE)
#define PARITY_ERROR (1<<UPE)
#define DATA_OVERRUN (1<<OVR)
#define DATA_REGISTER_EMPTY (1<<UDRE)
#define RX_COMPLETE (1<<RXC)
```

```
// USART Receiver buffer
#define RX_BUFFER_SIZE 8
char rx_buffer[RX_BUFFER_SIZE];
```

```
#if RX_BUFFER_SIZE<256
unsigned char rx_wr_index,rx_rd_index,rx_counter;
#else
unsigned int rx_wr_index,rx_rd_index,rx_counter;
#endif
```

```
// This flag is set on USART Receiver buffer overflow
bit rx_buffer_overflow;
```

```
// USART Receiver interrupt service routine
interrupt [USART_RXC] void usart_rx_isr(void)
{
    int i,motor;
    char status,data;
    PORTD.2=0;
```

```
    status=UCSRA;
    data=UDR;
```

```
    if(data=='3')
    {
```

```
        while(1)
        {
            if(motor==1)
            {
                break;
            }
            else
            {
```

```
                //buka pertama
```

```
                ///////////KIRI//////////
                for(i=0;i<3;i++)
                {
```

```

PORTB=0x01;
delay_ms(25);
PORTB=0x02;
delay_ms(25);
PORTB=0x04;
delay_ms(25);
PORTB=0x08;
delay_ms(25);
}
//////////

//////////KANAN//////////
for(i=0;i<3;i++)
{

    PORTB=0x08;
    delay_ms(25);
    PORTB=0x04;
    delay_ms(25);
    PORTB=0x02;
    delay_ms(25);
    PORTB=0x01;
    delay_ms(25);
}
//////////KANAN//////////

//buka kedua

//////////KIRI//////////
for(i=0;i<3;i++)
{
    PORTB=0x01;
    delay_ms(25);
    PORTB=0x02;
    delay_ms(25);
    PORTB=0x04;
    delay_ms(25);
    PORTB=0x08;
    delay_ms(25);
}
//////////

//////////KANAN//////////
for(i=0;i<3;i++)
{

    PORTB=0x08;
    delay_ms(25);
    PORTB=0x04;
    delay_ms(25);
    PORTB=0x02;
    delay_ms(25);
    PORTB=0x01;
    delay_ms(25);
}
//////////KANAN//////////

//buka ketiga

//////////KIRI//////////
for(i=0;i<3;i++)
{
    PORTB=0x01;
    delay_ms(25);
    PORTB=0x02;
    delay_ms(25);
    PORTB=0x04;
    delay_ms(25);
    PORTB=0x08;
    delay_ms(25);
}
//////////

//////////KANAN//////////
for(i=0;i<3;i++)
{
    PORTB=0x01;
    delay_ms(25);
    PORTB=0x02;
    delay_ms(25);
    PORTB=0x04;
    delay_ms(25);
    PORTB=0x08;
    delay_ms(25);
}
//////////

//////////KANAN//////////
for(i=0;i<3;i++)
{
    PORTB=0x01;
    delay_ms(25);
    PORTB=0x02;
    delay_ms(25);
    PORTB=0x04;
    delay_ms(25);
    PORTB=0x08;
    delay_ms(25);
}
//////////

PORTB=0x00;
motor=1;

}

}

if ((status & (FRAMING_ERROR | PARITY_ERROR |
DATA_OVERRUN))==0)
{
    rx_buffer[rx_wr_index]=data;
    if (++rx_wr_index == RX_BUFFER_SIZE) rx_wr_index=0;
    if (++rx_counter == RX_BUFFER_SIZE)
    {
        rx_counter=0;
        rx_buffer_overflow=1;
    };
}

#endif _DEBUG_TERMINAL_IO_
// Get a character from the USART Receiver buffer
#define _ALTERNATE_GETCHAR_
#pragma used+
char getchar(void)
{
    char data;
    while (rx_counter==0);
    data=rx_buffer[rx_rd_index];
    if (++rx_rd_index == RX_BUFFER_SIZE) rx_rd_index=0;
    #asm("cli")
    --rx_counter;
    #asm("sei")
    return data;
}
#pragma used-
#endif

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

// Declare your global variables here
int kirim1,kirim2;

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=0xFF;

```

```

// 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=0b11000000;
DDRD=0b11000100;

// 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=0x98;
UCSRC=0x86;
UBRRH=0x00;
UBRRL=0x4D;

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

// Global enable interrupts
#asm("sei")

while (1)
{
    // Place your code here

    if(PINC.0==1)
    {
        while(1)
        {
            if(kirim1==1)
            {
                break;
            }
            else
            {
                PORTD.2=1;

                putchar('e');
                delay_ms(1);

                PORTD.2=0;

                kirim1=kirim1+1;
                kirim2=0;
            }
        }
    }

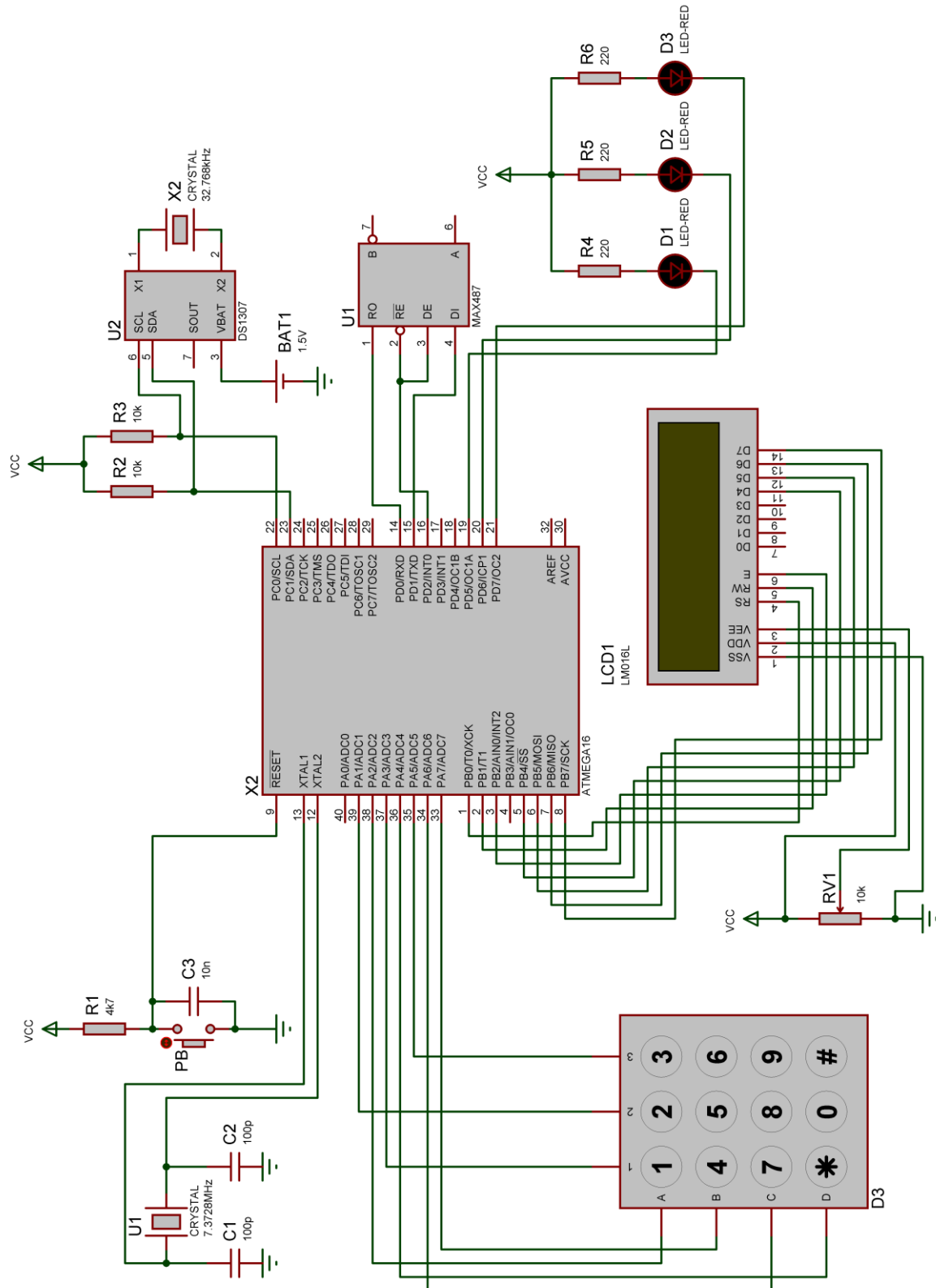
    if(PINC.0==0)
    {
        while(1)
        {
            if(kirim2==1)
            {
                break;
            }
            else
            {
                PORTD.2=1;

                putchar('f');
                delay_ms(1);

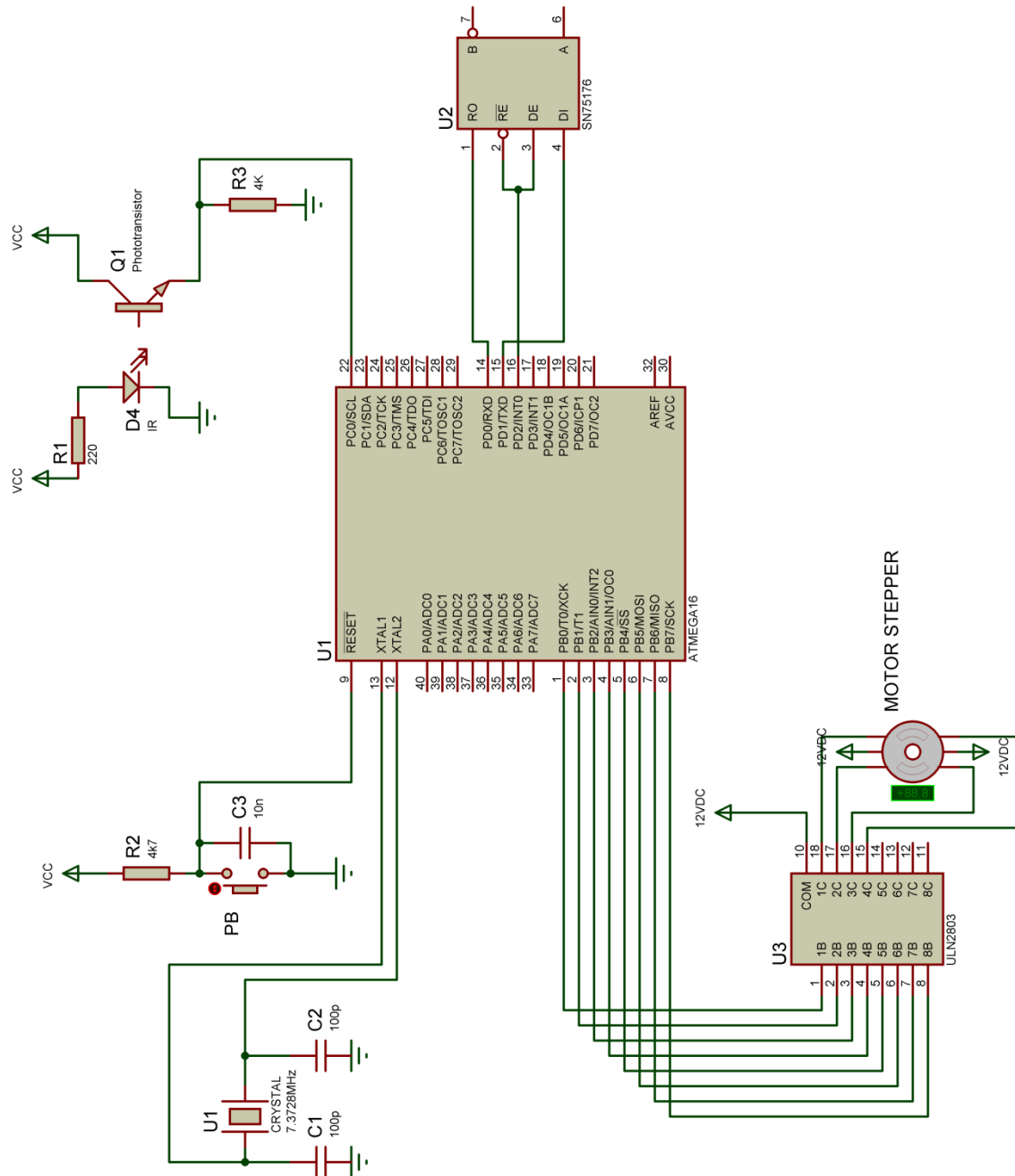
                PORTD.2=0;

                kirim2=kirim2+1;
                kirim1=0;
            }
        }
    }
};
}

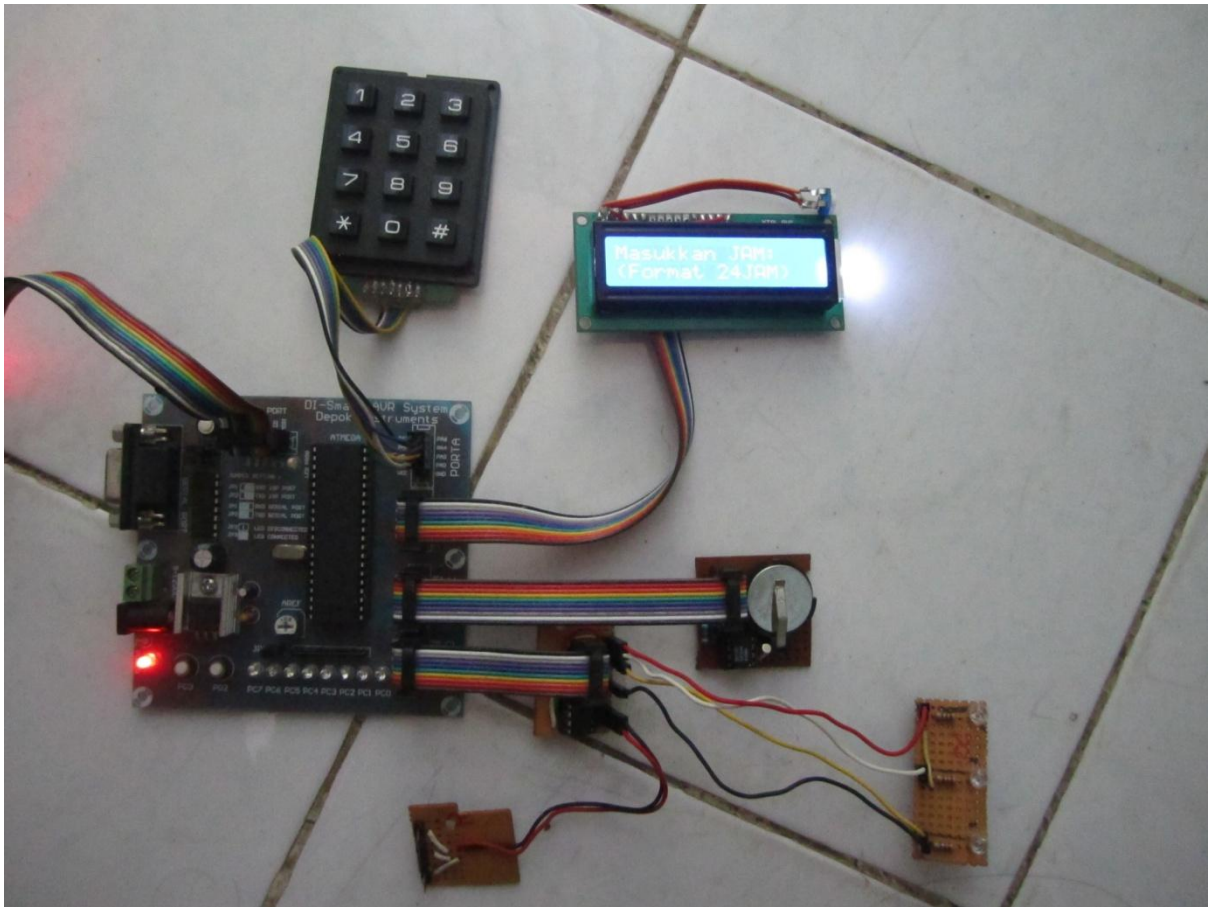
```

❖ **Skematik Master**

❖ Skematik Slave



❖ Foto Master



❖ Foto Slave



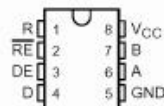
DATASHEET

SN75176A DIFFERENTIAL BUS TRANSCEIVER

SLLS100A — JUNE 1984 — REVISED MAY 1995

- Bidirectional Transceiver
- Meets or Exceeds the Requirements of ANSI Standards EIA/TIA-422-B and ITU Recommendation V.11
- Designed for Multipoint Transmission on Long Bus Lines in Noisy Environments
- 3-State Driver and Receiver Outputs
- Individual Driver and Receiver Enables
- Wide Positive and Negative Input/Output Bus Voltage Ranges
- Driver Output Capability . . . ± 60 mA Max
- Thermal-Shutdown Protection
- Driver Positive- and Negative-Current Limiting
- Receiver Input Impedance . . . $12\text{ k}\Omega$ Min
- Receiver Input Sensitivity . . . ± 200 mV
- Receiver Input Hysteresis . . . 50 mV Typ
- Operates From Single 5-V Supply
- Low Power Requirements

D OR P PACKAGE
(TOP VIEW)



description

The SN75176A differential bus transceiver is a monolithic integrated circuit designed for bidirectional data communication on multipoint bus-transmission lines. It is designed for balanced transmission lines and meets ANSI Standard EIA/TIA-422-B and ITU Recommendation V.11.

The SN75176A combines a 3-state differential line driver and a differential input line receiver, both of which operate from a single 5-V power supply. The driver and receiver have active-high and active-low enables, respectively, that can be externally connected together to function as a direction control. The driver differential outputs and the receiver differential inputs are connected internally to form differential input/output (I/O) bus ports that are designed to offer minimum loading to the bus whenever the driver is disabled or $V_{CC} = 0$. These ports feature wide positive and negative common-mode voltage ranges making the device suitable for party-line applications.

The driver is designed to handle loads up to 60 mA of sink or source current. The driver features positive- and negative-current limiting and thermal shutdown for protection from line fault conditions. Thermal shutdown is designed to occur at a junction temperature of approximately 150°C . The receiver features a minimum input impedance of $12\text{ k}\Omega$, an input sensitivity of ± 200 mV, and a typical input hysteresis of 50 mV.

The SN75176A can be used in transmission-line applications employing the SN75172 and SN75174 quadruple differential line drivers and SN75173 and SN75175 quadruple differential line receivers.

The SN75176A is characterized for operation from 0°C to 70°C .



Please be aware that an important notice concerning availability, standard warranty, and use in critical applications of Texas Instruments semiconductor products and disclaimers thereto appears at the end of this data sheet.

PRODUCTION DATA information is current as of publication date. Products conform to specifications per the terms of Texas Instruments standard warranty. Production processing does not necessarily include testing of all parameters.

**TEXAS
INSTRUMENTS**

POST OFFICE BOX 655303 • DALLAS, TEXAS 75265

Copyright © 1995, Texas Instruments Incorporated

1

SN75176A **DIFFERENTIAL BUS TRANSCEIVER**

SLS100A – JUNE 1984 – REVISED MAY 1985

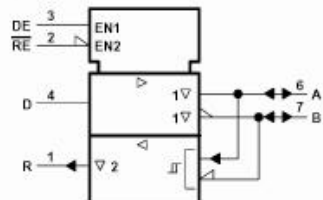
Function Tables

DRIVER			
INPUT D	ENABLE DE	OUTPUTS	
		A	B
H	H	H	L
L	H	L	H
X	L	Z	Z

RECEIVER		
DIFFERENTIAL INPUTS A – B	ENABLE RE	OUTPUT R
$V_{ID} \geq 0.2 \text{ V}$	L	H
$-0.2 \text{ V} < V_{ID} < 0.2 \text{ V}$	L	?
$V_{ID} \leq -0.2 \text{ V}$	L	L
X	H	Z
Open	L	?

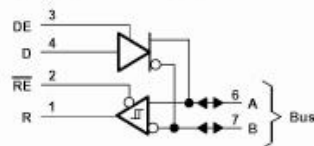
H = high level, L = low level, ? = indeterminate,
X = irrelevant, Z = high impedance (off)

logic symbol†



† This symbol is in accordance with ANSI/IEEE Std 91-1984 and IEC Publication 617-12.

logic diagram (positive logic)



**TEXAS
INSTRUMENTS**

POST OFFICE BOX 655303 • DALLAS, TEXAS 75265

Features

- High-performance, Low-power AVR[®] 8-bit Microcontroller
- Advanced RISC Architecture
 - 131 Powerful Instructions – Most Single-clock Cycle Execution
 - 32 x 8 General Purpose Working Registers
 - Fully Static Operation
 - Up to 16 MIPS Throughput at 16 MHz
 - On-chip 2-cycle Multiplier
- Nonvolatile Program and Data Memories
 - 16K Bytes of In-System Self-Programmable Flash
 - Endurance: 10,000 Write/Erase Cycles
 - Optional Boot Code Section with Independent Lock Bits
 - In-System Programming by On-chip Boot Program
 - True Read-While-Write Operation
 - 512 Bytes EEPROM
 - Endurance: 100,000 Write/Erase Cycles
 - 1K Byte Internal SRAM
 - Programming Lock for Software Security
- JTAG (IEEE std. 1149.1 Compliant) Interface
 - Boundary-scan Capabilities According to the JTAG Standard
 - Extensive On-chip Debug Support
 - Programming of Flash, EEPROM, Fuses, and Lock Bits through the JTAG Interface
- Peripheral Features
 - Two 8-bit Timer/Counters with Separate Prescalers and Compare Modes
 - One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
 - Real Time Counter with Separate Oscillator
 - Four PWM Channels
 - 8-channel, 10-bit ADC
 - 8 Single-ended Channels
 - 7 Differential Channels in TQFP Package Only
 - 2 Differential Channels with Programmable Gain at 1x, 10x, or 200x
 - Byte-oriented Two-wire Serial Interface
 - Programmable Serial USART
 - Master/Slave SPI Serial Interface
 - Programmable Watchdog Timer with Separate On-chip Oscillator
 - On-chip Analog Comparator
- Special Microcontroller Features
 - Power-on Reset and Programmable Brown-out Detection
 - Internal Calibrated RC Oscillator
 - External and Internal Interrupt Sources
 - Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby and Extended Standby
- I/O and Packages
 - 32 Programmable I/O Lines
 - 40-pin PDIP, 44-lead TQFP, and 44-pad MLF
- Operating Voltages
 - 2.7 - 5.5V for ATmega16L
 - 4.5 - 5.5V for ATmega16
- Speed Grades
 - 0 - 8 MHz for ATmega16L
 - 0 - 16 MHz for ATmega16



**8-bit AVR[®]
Microcontroller
with 16K Bytes
In-System
Programmable
Flash**

**ATmega16
ATmega16L**

Preliminary

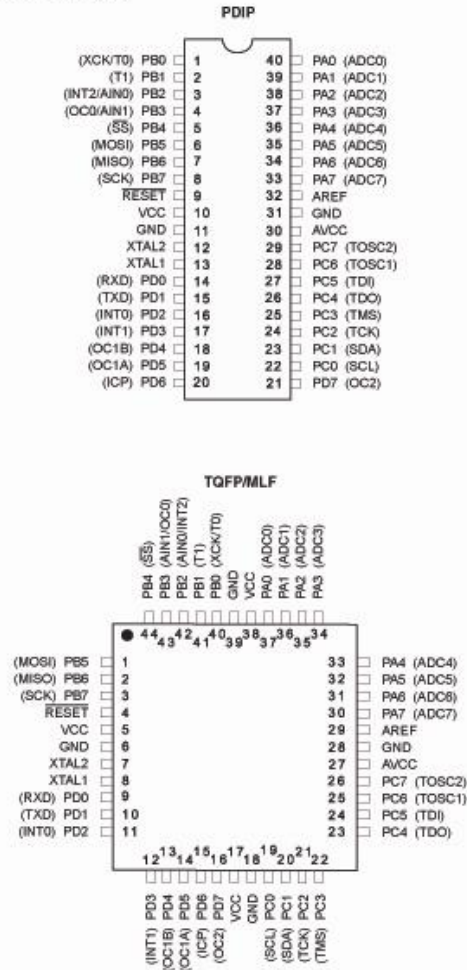
Rev. 2468E-AVR-10/02





Pin Configurations

Figure 1. Pinouts ATmega16



Disclaimer

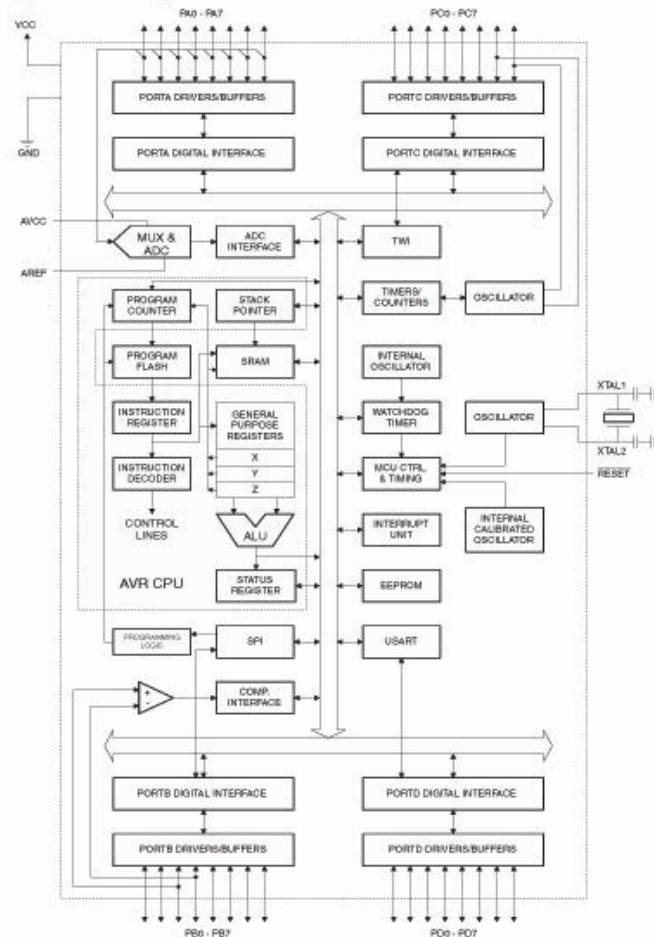
Typical values contained in this data sheet are based on simulations and characterization of other AVR microcontrollers manufactured on the same process technology. Min and Max values will be available after the device is characterized.

Overview

The ATmega16 is a low-power CMOS 8-bit microcontroller based on the AVR enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega16 achieves throughputs approaching 1 MIPS per MHz allowing the system designer to optimize power consumption versus processing speed.

Block Diagram

Figure 2. Block Diagram





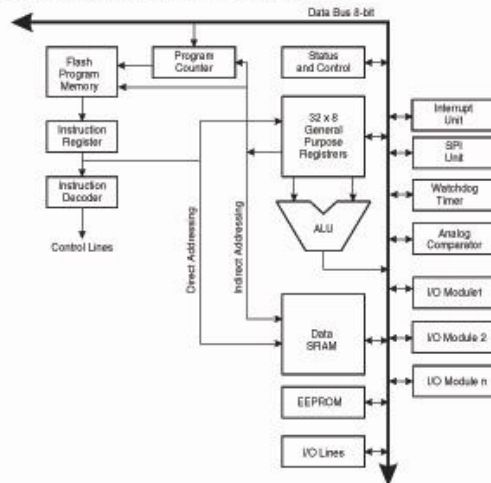
AVR CPU Core

Introduction

This section discusses the AVR core architecture in general. The main function of the CPU core is to ensure correct program execution. The CPU must therefore be able to access memories, perform calculations, control peripherals, and handle interrupts.

Architectural Overview

Figure 3. Block Diagram of the AVR MCU Architecture



In order to maximize performance and parallelism, the AVR uses a Harvard architecture – with separate memories and buses for program and data. Instructions in the program memory are executed with a single level pipelining. While one instruction is being executed, the next instruction is pre-fetched from the program memory. This concept enables instructions to be executed in every clock cycle. The program memory is In-System Reprogrammable Flash memory.

The fast-access Register file contains 32 x 8-bit general purpose working registers with a single clock cycle access time. This allows single-cycle Arithmetic Logic Unit (ALU) operation. In a typical ALU operation, two operands are output from the Register file, the operation is executed, and the result is stored back in the Register file – in one clock cycle.

Six of the 32 registers can be used as three 16-bit indirect address register pointers for Data Space addressing – enabling efficient address calculations. One of these address pointers can also be used as an address pointer for look up tables in Flash Program memory. These added function registers are the 16-bit X-, Y-, and Z-register, described later in this section.

The ALU supports arithmetic and logic operations between registers or between a constant and a register. Single register operations can also be executed in the ALU. After

TOSHIBA Bipolar Digital Integrated Circuit Silicon Monolithic

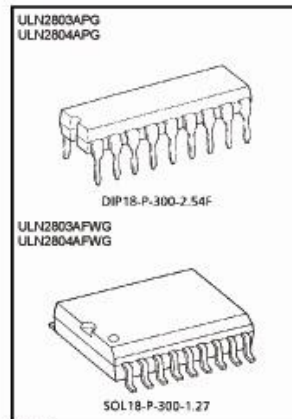
**ULN2803APG,ULN2803AFWG,ULN2804APG,ULN2804AFWG
(Manufactured by Toshiba Malaysia)**

8ch Darlington Sink Driver

The ULN2803APG / AFWG Series are high-voltage, high-current darlington drivers comprised of eight NPN darlington pairs.
All units feature integral clamp diodes for switching inductive loads.
Applications include relay, hammer, lamp and display (LED) drivers.
The suffix (G) appended to the part number represents a Lead (Pb)-Free product.

Features

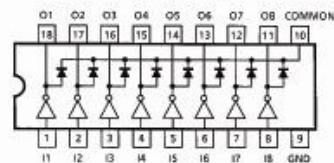
- Output current (single output)
500 mA (Max.)
- High sustaining voltage output
50 V (Min.)
- Output clamp diodes
- Inputs compatible with various types of logic.
- Package Type-APG : DIP-18pin
- Package Type-AFWG : SOL-18pin



Weight
DIP18-P-300-2.54F: 1.478 g (Typ.)
SOL18-P-300-1.27 : 0.48 g (Typ.)

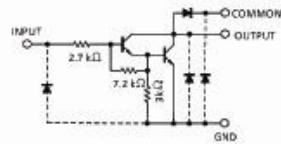
Pin Connection (top view)

Type	Input Base Resistor	Designation
ULN2803APG / AFWG	2.7 k Ω	TTL, 5 V CMOS
ULN2804APG / AFWG	10.5 k Ω	6~15 V PMOS, CMOS

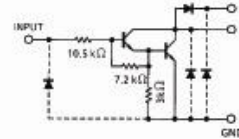


Schematics (each driver)

ULN2803APG / AFWG



ULN2804APG / AFWG



Note: The input and output parasitic diodes cannot be used as clamp diodes.

Absolute Maximum Ratings (Ta = 25°C)

Characteristic	Symbol	Rating	Unit
Output sustaining voltage	$V_{CE(SUS)}$	-0.5~50	V
Output current	I_{OUT}	500	mA / ch
Input voltage	V_{IN}	-0.5~30	V
Clamp diode reverse voltage	V_R	50	V
Clamp diode forward current	I_F	500	mA
Power dissipation	APG	P_D	1.47
	AFWG		0.92 / 1.31 (Note)
Operating temperature	T_{opr}	-40~85	°C
Storage temperature	T_{stg}	-55~150	°C

Note: On Glass Epoxy PCB (75 × 114 × 1.6 mm Cu 20%)

FEATURES

- Real-time clock (RTC) counts seconds, minutes, hours, date of the month, month, day of the week, and year with leap-year compensation valid up to 2100
- 56-byte, battery-backed, nonvolatile (NV) RAM for data storage
- Two-wire serial interface
- Programmable squarewave output signal
- Automatic power-fail detect and switch circuitry
- Consumes less than 500nA in battery backup mode with oscillator running
- Optional industrial temperature range: -40°C to +85°C
- Available in 8-pin DIP or SOIC
- Underwriters Laboratory (UL) recognized

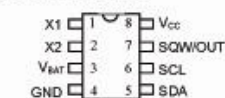
ORDERING INFORMATION

DS1307	8-Pin DIP (300-mil)
DS1307Z	8-Pin SOIC (150-mil)
DS1307N	8-Pin DIP (Industrial)
DS1307ZN	8-Pin SOIC (Industrial)

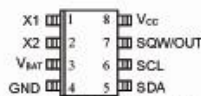
DESCRIPTION

The DS1307 Serial Real-Time Clock is a low-power, full binary-coded decimal (BCD) clock/calendar plus 56 bytes of NV SRAM. Address and data are transferred serially via a 2-wire, bi-directional bus. The clock/calendar provides seconds, minutes, hours, day, date, month, and year information. The end of the month date is automatically adjusted for months with fewer than 31 days, including corrections for leap year. The clock operates in either the 24-hour or 12-hour format with AM/PM indicator. The DS1307 has a built-in power sense circuit that detects power failures and automatically switches to the battery supply.

PIN ASSIGNMENT



DS1307 8-Pin DIP (300-mil)



DS1307 8-Pin SOIC (150-mil)

PIN DESCRIPTION

V _{CC}	- Primary Power Supply
X1, X2	- 32.768kHz Crystal Connection
V _{BAT}	- +3V Battery Input
GND	- Ground
SDA	- Serial Data
SCL	- Serial Clock
SQW/OUT	- Square Wave/Output Driver