

LAMPIRAN B
PROGRAM PADA PENGENDALI MIKRO
ATMEGA16

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

```
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 : TA RICKY  
Version :  
Date   : 3/25/2011  
Author : RICKY  
Company :  
Comments:
```

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

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

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

```
// Alphanumeric LCD Module functions  
#asm  
    .equ __lcd_port=0x15 ;PORTC  
#endasm  
#include <lcd.h>
```

```
// Declare your global variables here  
unsigned char lcd_buffer[33];  
unsigned int  
countred=0,countblue=0,countgreen=0,red,green,blue,x=0,w=0,z=0,r,g,b,black,ke  
y;  
float fr,tr,fg,tg,fb,tb,cyan,magenta,yellow,k;
```

```
void merah()  
{  
PORTA.6=0; // s2  
PORTA.7=0; // s3  
for(x=0;x<=10;x++) //signal cutter
```

```

{
while(PINA.5==0){countred=0;}
while(PINA.5==1){countred=0;}
}

while(PINA.5==0)
{
countred++;
}

while(PINA.5==1)
{
countred++;
}

tr=(countred*0.00000090422453703703703703703703704);
fr=(1/tr);
red=(fr-10004)/18363*255;

countred=0;

}

void hijau()
{
PORTA.6=1; // s2
PORTA.7=1; // s3
for(w=0;w<=10;w++) //signal cutter
{
while(PINA.5==0){countgreen=0;}
while(PINA.5==1){countgreen=0;}
}

while(PINA.5==0)
{
countgreen++;
}

while(PINA.5==1)
{
countgreen++;
}

tg=(countgreen*0.00000090422453703703703703703703704);
fg=(1/tg);
green=(fg-42589)/25225*255;

```

```

        countgreen=0;

    }

void biru()
{
PORTA.6=0;    // s2
PORTA.7=1;    // s3
for(z=0;z<=10;z++)    //signal cutter
{
    while(PINA.5==0){countblue=0;}
    while(PINA.5==1){countblue=0;}
}
}

    while(PINA.5==0)
    {
        countblue++;
    }

    while(PINA.5==1)
    {
        countblue++;
    }

    tb=(countblue*0.00000090422453703703703703703703704);
    fb=(1/tb);
    blue=(fb-10074)/22997*255;

countblue=0;

}

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

// Input/Output Ports initialization
// Port A initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In
Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T
PORTA=0x13;
DDRA=0xDF;

```

```

// 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=0x06;
DDRD=0x0F;

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

// Timer/Counter 1 initialization
// Clock source: System Clock
// Clock value: Timer 1 Stopped
// Mode: Normal top=FFFFh
// OC1A output: Discon.
// OC1B output: Discon.
// Noise Canceler: Off
// Input Capture on Falling Edge
// Timer 1 Overflow Interrupt: Off
// Input Capture Interrupt: Off
// Compare A Match Interrupt: Off
// Compare B Match Interrupt: Off
TCCR1A=0x00;
TCCR1B=0x00;
TCNT1H=0x00;
TCNT1L=0x00;

```

```

ICR1H=0x00;
ICR1L=0x00;
OCR1AH=0x00;
OCR1AL=0x00;
OCR1BH=0x00;
OCR1BL=0x00;

// Timer/Counter 2 initialization
// Clock source: System Clock
// Clock value: Timer 2 Stopped
// Mode: Normal top=FFh
// OC2 output: Disconnected
ASSR=0x00;
TCCR2=0x00;
TCNT2=0x00;
OCR2=0x00;

// External Interrupt(s) initialization
// INT0: Off
// INT1: Off
// INT2: Off
MCUCR=0x00;
MCUCSR=0x00;

// Timer(s)/Counter(s) Interrupt(s) initialization
TIMSK=0x00;

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

// LCD module initialization
lcd_init(16);

while (1)
{
// Place your code here

merah();
delay_ms(500);
hijau();
delay_ms(500);
biru();
}

```

```

delay_ms(500);

//convert
r=255-red;
g=255-green;
b=255-blue;

if(r<g && r<b)
{
black=r;
}
else if(g<r && g<b)
{
black=g;
}
else
{
black=b;
}
key=255-black;
cyan=(r-black)/key*100/255;
magenta=(g-black)/key*100/255;
yellow=(b-black)/key*100/255;
k=black*100/255;

//LCD
lcd_gotoxy(0,0);
lcd_clear();
sprintf(lcd_buffer,"C:%f M:%f \nY:%f K:%f",cyan,magenta,yellow,k);
lcd_puts(lcd_buffer);
x=0;w=0;z=0;

};
}

```