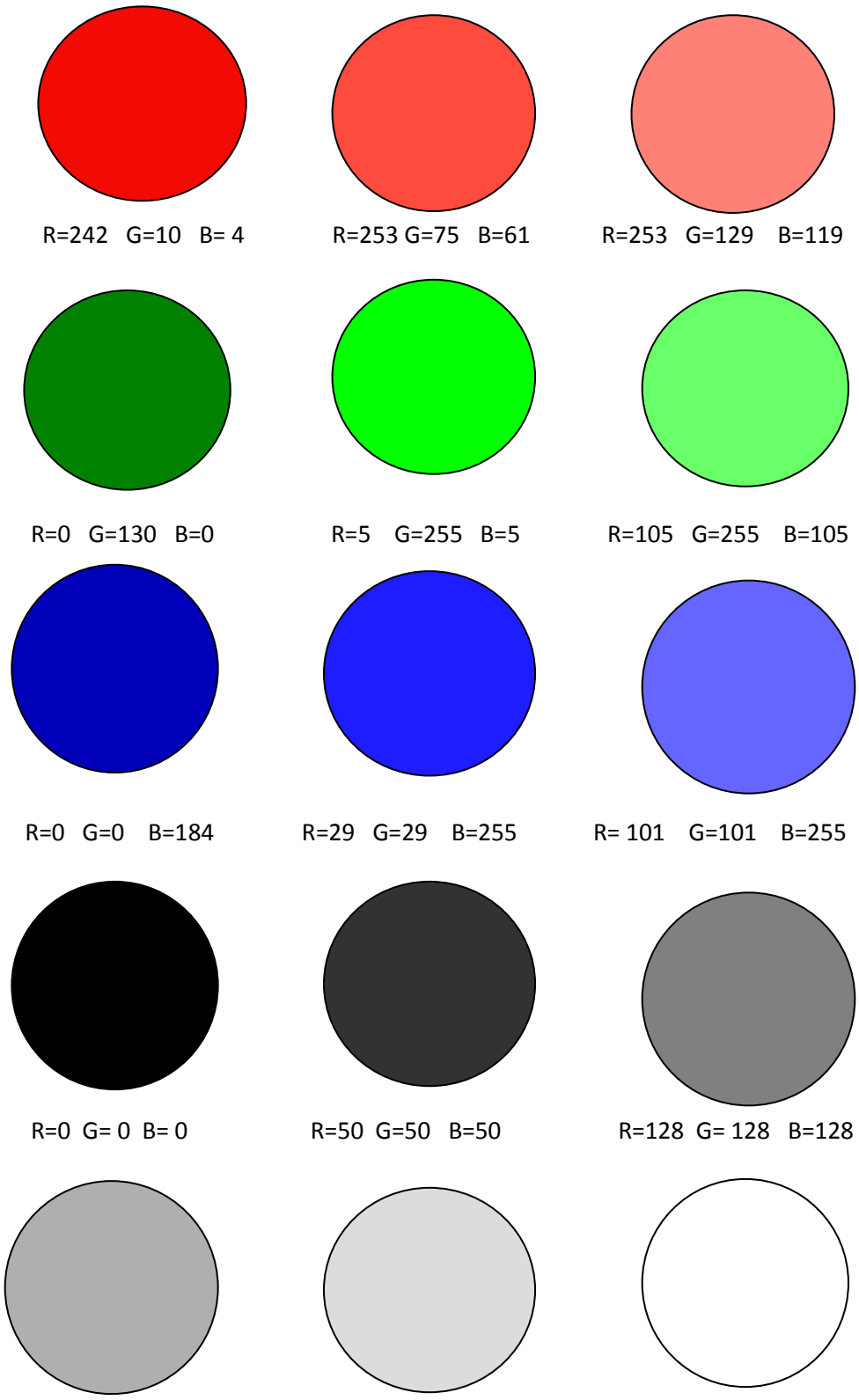
LAMPIRAN

Warna-warna yang digunakan dalam pengujian sensor



R=175 G=175 B=175 R=220 G=220 B=220 R=225 G=225 B=225

PROGRAM ROBOT DENGAN BAHASA C++

```
/*****************************************************
This program was produced by the
CodeWizardAVR V1.25.9
Project :
Version :
Date    : 3/1/2010
Author  : inovation
Company : inovation electronic
Comments:


Chip type           : ATmega8535
Program type        : Application
Clock frequency     : 12.000000 MHz
Memory model        : Small
External SRAM size  : 0
Data Stack size     : 128
*****************************************************
#include <mega8535.h>
#include <stdio.h>
#include <stdlib.h>
#include <delay.h>
// Alphanumeric LCD Module functions
#asm
   .equ __lcd_port=0x15 ;PORTC
#endasm
#include <lcd.h>
#define S2 PORTB.2
```

```
#define S3 PORTB.3
#define S0 PORTB.4
#define S1 PORTB.5
#define OE PORTB.6
unsigned char status;
unsigned int freq;
unsigned char old_color;
unsigned char strbuf[16];
unsigned char color;
unsigned int dummy;
// Timer 1 overflow interrupt service routine
interrupt [TIM1_OVF] void timer1_ovf_isr(void)
{
// Place your code here
lcd_gotoxy(0,0);
lcd_putsf("OVER FLOW");
lcd_gotoxy(0,1);
lcd_putsf("CHANGE SCALE!!");
delay_ms(500);
}
interrupt [TIM0_OVF] void timer0_ovf_isr(void)
{
// Reinitialize Timer 0 value
freq=TCNT1;
TCNT0=0x00;
status=1;
// Place your code here
}
```

```
void get_freq(void)
{

    lcd_gotoxy(0,1);
    lcd_putsf("FREQ :       kHz");
    dummy=freq/10;
    if(dummy>=62 & dummy<=64)
    {

    color=1;
    lcd_gotoxy(0,0);
    lcd_putsf("COLOR: BLACK   ");
    }
    else
    if(dummy>=86 & dummy<=88)
    {
    color=2;
    lcd_gotoxy(0,0);
    lcd_putsf("COLOR: GREEN   ");
    }
    else
    if(dummy>=92 & dummy<=94)
    {
    color=3;
    lcd_gotoxy(0,0);
    lcd_putsf("COLOR: BLUE    ");
    }
    else
    if(dummy>=107 & dummy<=109)
```

```
        {
        color=4;
        lcd_gotoxy(0,0);
        lcd_putsf("COLOR: RED     ");
        }
        else
        if(dummy>=202 & dummy<=209)
        {
        color=5;
        lcd_gotoxy(0,0);
        lcd_putsf("COLOR: WHITE   ");
        }
        else
        {
        color=6;
        lcd_gotoxy(0,0);
        lcd_putsf("COLOR: NONE   ");
        }



        sprintf(strbuf,"%u.0",dummy);
//      sprintf(strbuf,"%d",freq);
        lcd_gotoxy(7,1);
        lcd_puts(strbuf);
}
void move_arm(void)
{
//mas instruksi gerak lengan taruh di sini.....
lcd_gotoxy(0,0);
lcd_putsf("MOVING ARM      ");
lcd_gotoxy(0,1);
```

```
lcd_putsf("WAIT FOR MINUTE ");
delay_ms(500);
}
// Declare your global variables here
void main(void)
{
// Declare your local variables here
// Input/Output Ports initialization
// Port A initialization
// Func7=Out Func6=Out Func5=Out Func4=In Func3=Out Func2=Out Func1=Out Func0=Out
// State7=1 State6=1 State5=1 State4=P State3=1 State2=1 State1=1 State0=1
PORTA=0xFF;
DDRA=0xEF;
// 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=0x03;
DDRB=0xFC;
// Port C initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T
PORTC=0x00;
DDRC=0x00;
// Port D initialization
// Func7=Out Func6=Out Func5=Out Func4=Out Func3=Out Func2=Out Func1=Out Func0=Out
// State7=1 State6=1 State5=1 State4=1 State3=1 State2=1 State1=1 State0=1
PORTD=0xFF;
DDRD=0xFF;

// Timer/Counter 0 initialization
// Clock source: System Clock
```

```
// Clock value: 46.875 kHz
// Mode: Normal top=FFh
// OC0 output: Disconnected
TCCR0=0x00;
TCNT0=0x00;     //timer intterupt every 1mS
OCR0=0x00;
// Timer/Counter 1 initialization
// Clock source: T1 pin Falling Edge
// Mode: Normal top=FFFFh
// OC1A output: Discon.
// OC1B output: Discon.
// Noise Canceler: Off
// Input Capture on Falling Edge
// Timer 1 Overflow Interrupt: On
// Input Capture Interrupt: Off
// Compare A Match Interrupt: Off
// Compare B Match Interrupt: Off
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=0x04;  //interupt mask register
// 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);
// Global enable interrupts
#asm("sei")
intro();
lcd_gotoxy(0,1);
//lcd_putsf("FREK        kHz");
init_TSL230();
 while (1)
     {
     // Place your code here
     status=0;
     get_freq();
        if (old_color==color)
        {
        status=0xff;
```

```
        }
        else
        {
            old_color=color;
        }
    delay_ms(500);

    if (status==0xff & color<6 & color>0 )
    {
    move_arm();
    old_color=6;
    }
    }
}
```