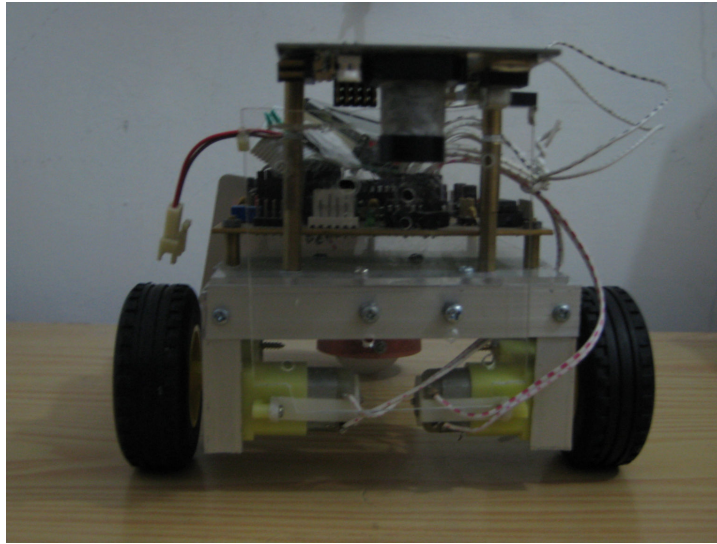
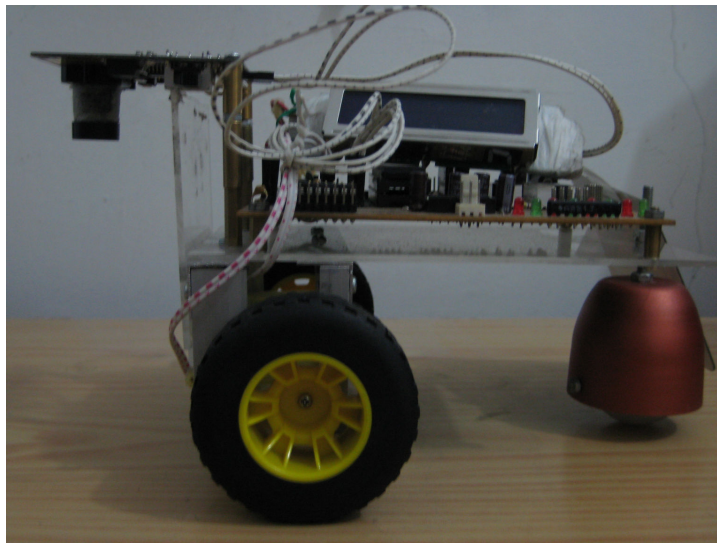


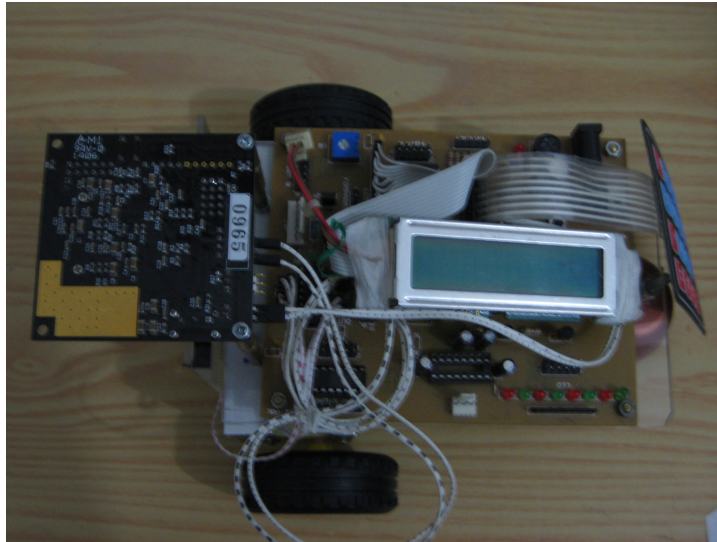
LAMPIRAN A
FOTO ROBOT MOBIL



Gambar A.1. Gambar Robot Mobil Normal Dilihat dari Depan



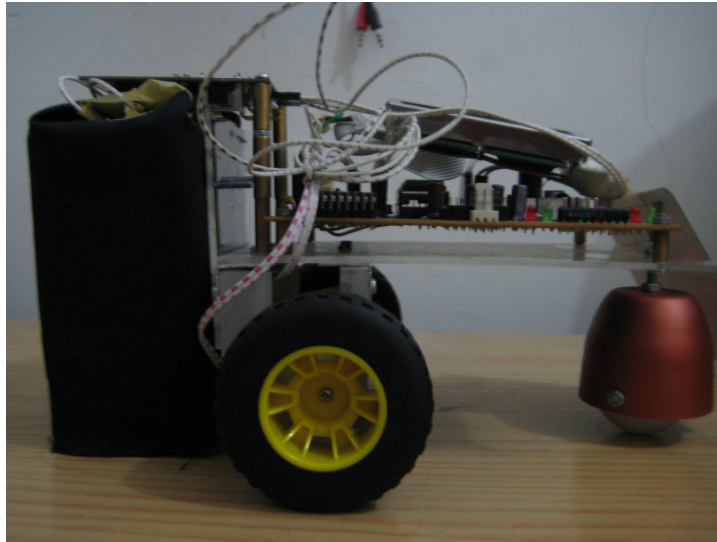
Gambar A.2. Gambar Robot Mobil Normal Dilihat dari Samping



Gambar A.3. Gambar Robot Mobil Normal Dilihat dari Atas



Gambar A.4. Gambar Robot Mobil *Box* Dilihat dari Depan



Gambar A.5. Gambar Robot Mobil *Box* Dilihat dari Samping



Gambar A.6. Gambar Robot Mobil *Box* Dilihat dari Atas

LAMPIRAN B
PROGRAM PADA PENGONTROL MIKRO
ATMEGA16

PROGRAM UTAMA

/******

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 : 5/7/2008
Author : F4CG
Company : F4CG
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>

#asm
.equ __lcd_port=0x18 ;PORTB
#endasm
#include <lcd.h>
#include <delay.h>
#include <stdio.h>
#include <scankeypad.h>

unsigned char tdata[31];
unsigned char tdata2[31];
unsigned char tdata3[31];
unsigned char lcd_buff[33];
unsigned char pos_pos,i,iw,ix,iz,iw1,iw2,iw3;
unsigned char pos[9];

#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)

#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

bit rx_buffer_overflow;

interrupt [USART_RXC] void usart_rx_isr(void)
{
char status,data;
status=UCSRA;
data=UDR;
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;
};
};

if (data == 13)
{
i=0;
for (iw=0;iw<31;iw++)tdata2[iw]=tdata[iw];
for (iw=0;iw<31;iw++)tdata[iw]=0;
}
tdata[i]=data;
i++;
}

#ifndef _DEBUG_TERMINAL_IO_
#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

```

```

interrupt [TIM1_COMPA] void timer1_compa_isr(void)
{
    for (ix=0;ix<9;ix++) pos[ix]=0;
    for (ix=0;ix<31;ix++)tdata3[ix]=tdata2[ix];
    pos_pos=0;
    for (ix=0;ix<31;ix++)
    {
        if(tdata3[ix]==32)pos_pos++;
        if(tdata3[ix+1]!=32)pos[pos_pos]=(pos[pos_pos]*10)+ (tdata3[ix+1]-48);
    }
}

void main(void)
{
    TCCR1A=0x00;
    TCCR1B=0x0B;
    OCR1AH=0x08;
    OCR1AL=0x70;

    TIMSK=0x10;
    UCSRA=0x00;
    UCSRB=0x98;
    UCSRC=0x86;
    UBRRH=0x00;
    UBRL=0x47;
    ACSR=0x80;
    SFIOR=0x00;
    DDRD.2=1;
    DDRD.3=1;
    DDRD.4=1;
    DDRD.5=1;
    DDRD.6=1;
    DDRD.7=1;
    lcd_init(16);
    #asm("sei")

    printf("L0 1r");
    delay_ms(200);
    printf("L0 0r");
    delay_ms(200);

    while (1)
    {
        for (iz=0;iz<10;iz++)
        {

            lcd_putsf("A>SET WARNA\n");
            lcd_putsf("B>RUN ");
            delay_ms(1500);
            if(scan_keypad()=='A')
            {
                lcd_clear();

                for (iw1=0;iw1<10;iw1++)

```



```

{
lcd_putsf("A>MERAH\n");
lcd_putsf("B>BIRU ");
delay_ms(150);
if(scan_keypad()=='A')
{
lcd_clear();
printf("RS\r");
delay_ms(200);
printf("L0 1\r");
delay_ms(200);
printf("L0 0\r");
delay_ms(200);
printf("TC 71 121 22 72 40 90\r");
delay_ms(200);
lcd_putsf("TRACK \nWARNA MERAH");
delay_ms(500);
lcd_clear();
}

if(scan_keypad()=='B')
{
lcd_clear();
printf("RS\r");
delay_ms(200);
printf("L0 1\r");
delay_ms(200);
printf("L0 0\r");
delay_ms(200);
printf("TC 24 74 71 121 180 230\r");
delay_ms(200);
lcd_putsf("TRACK \nWARNA BIRU");
delay_ms(500);
lcd_clear();
}

delay_ms(150);
lcd_clear();
};

for (iw2=0;iw2<10;iw2++)
{
lcd_putsf("C>KUNING\n");
if(scan_keypad()=='C')
{
lcd_clear();
printf("RS\r");
delay_ms(200);
printf("L0 1\r");
delay_ms(200);
printf("L0 0\r");
delay_ms(200);
printf("TC 104 154 127 177 24 74\r");
delay_ms(200);
lcd_putsf("TRACK \nWARNA KUNING");
delay_ms(500);
}
}

```

```

        lcd_clear();
    };
    delay_ms(150);
    lcd_clear();
}

delay_ms(1000);
lcd_clear();
};

if(scan_keypad()=='B')
{
    lcd_clear();
    while (scan_keypad()!='#')
    {
        sprintf(lcd_buff,"X%d Y%d P%d",pos[1],pos[2],pos[6]);

        if ((pos[1]<60)&&(pos[1]>40))
        {
            PORTD.2=1;
            PORTD.3=0;
            PORTD.4=1;
            PORTD.5=1;
            PORTD.6=1;
            PORTD.7=0;
            delay_ms(400);
            PORTD.2=0;
            PORTD.3=0;
            PORTD.4=0;
            PORTD.5=0;
            PORTD.6=0;
            PORTD.7=0;
        };

        if ((pos[1]<90)&&(pos[1]>60)) //kiri
        {
            PORTD.2=0;
            PORTD.3=0;
            PORTD.4=1;
            PORTD.5=1;
            PORTD.6=1;
            PORTD.7=0;
            delay_ms(400);
            PORTD.2=0;
            PORTD.3=0;
            PORTD.4=0;
            PORTD.5=0;
            PORTD.6=0;
            PORTD.7=0;
        };

        if ((pos[1]<40)&&(pos[1]>1)) //kanan
        {
            PORTD.2=1;
            PORTD.3=0;
            PORTD.4=1;

```

```

        PORTD.5=1;
        PORTD.6=0;
        PORTD.7=0;
        delay_ms(400);
        PORTD.2=0;
        PORTD.3=0;
        PORTD.4=0;
        PORTD.5=0;
        PORTD.6=0;
        PORTD.7=0;
    };

    lcd_puts(lcd_buff);
    delay_ms(100);
    lcd_clear();
}

delay_ms(1000);
lcd_clear();
};

for (iw3=0;iw3<10;iw3++)
{
    lcd_clear();
    lcd_putsf("C>RESET");

    if(scan_keypad()=='C')
    {
        lcd_putsf("RESET");
        printf("RS\r");
        delay_ms(200);
        printf("L0 1\r");
        delay_ms(200);
        printf("L0 0\r");
        delay_ms(200);
        delay_ms(1000);
        lcd_clear();
    };

    delay_ms(150);
    lcd_clear();
}

delay_ms(150);
};
}
}

```