DAFTAR PUSTAKA

Budiharto Widodo, ”Perancangan Sistem dan Aplikasi Mikrokontroler”, ElexMedia Komputindo, Jakarta, 2005

Wardana Lingga, “Belajar Sendiri Mikrokontroler AVR“, penerbit Andi, Yogyakarta, 2006

Sembiring., Rehulina. “Berkenalan dengan android studio”. http://panduanandroidstudio.blogspot.com/. Diakses pada tanggal 7Juni 2015.

Alakadarnya, puguh. “Light Emitting Dioda”, https://rasapas.wordpress.com/2011/03/04/8/. Diakses pada tanggal 5 Agustus 2015

“Membuat Alarm yang Diaktifkan oleh Suara/getaran “, http://madnsodik.blogspot.com/. Diakses pada tanggal 14 Mei 2015

http://repository.usu.ac.id/bitstream/123456789/47743/4/Chapter%20II.pdf. Diakses pada tanggal 7 Agustus 2015

LAMPIRAN

(Program Mikrokontroller)

```
#include <mega8.h>
#include <delay.h>

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

#define ADC_VREF_TYPE 0x00

// Read the AD conversion result
unsigned int read_adc(unsigned char adc_input)
{
ADMUX=adc_input | (ADC_VREF_TYPE & 0xff);
// Delay needed for the stabilization of the ADC input voltage
delay_us(10);
// Start the AD conversion
ADCSRA|=0x40;
// Wait for the AD conversion to complete
while ((ADCSRA & 0x10)==0);
ADCSRA|=0x10;
return ADCW;
}

#define buz_on PORTC.4=1
```

```
#define buz_off PORTC.4=0
#define rel_on PORTC.3=1
#define rel_off PORTC.3=0
#define h_on PORTB.1=0;
#define h_off PORTB.1=1;
#define k_on PORTB.2=0;
#define k_off PORTB.2=1;
#define m_on PORTB.3=0;
#define m_off PORTB.3=1;
#define emr PINC.1

unsigned char t1,t2,data_rx,text,isi,s,c,req;
int temp,i,x;
float vin,v1;

unsigned char UartGetChar()
{
unsigned long tout=0;
unsigned char dat;
        for(;;)
        {
                if(UCSRA & (1<<RXC))
                        {
                                dat= UDR;
                                break;
                        }
```

```
                tout++;
                if(tout>10000)
                        {
                                dat='-';
                                break;

                        }
   }
return dat;
}

void getar()
{    v1=0;
     for (i=0;i<=5;i++){
     temp=read_adc(5);
     vin= ((float)temp*5)/1023;
     v1=v1+vin;
     }
     v1=(float)v1/5;//printf("%0.1f \r",v1);
}

void android()
{isi=0;
for (x=0;x<=50;x++)
{//printf("A:");
data_rx=UartGetChar();
```

```
if (data_rx=='*')
{text=UartGetChar();isi=1;break;}
getar();
}
}

void delay()
{
for (t2=0;t2<=100;t2++)
{delay_ms(1);}
}

void jalan()
{     if (req==0){
      for (t1=0;t1<=30;t1++)
      {buz_off;
      delay();
      if (t1<20){h_on;k_off;m_off;}
      if (t1>=20 && t1<28){h_off;k_on;m_off;}
      if (t1>=28 && t1<=30){h_off;k_off;m_on;}
      if (t1>25){printf("*R");
      android();
      delay();
      if (text=='J' && isi==1){t1=0;}
      delay_ms(1);
      }
```

```
      if (PINC.0==1){rel_off;text='S';c=1;break;}

      }}

}


void alarm()
{
rel_off;m_on;
getar();
if (v1>=0.5){buz_on;}
}
void main(void)
{
PORTB=0x00;DDRB=0xff;
PORTC=0x00;DDRC=0x18;
PORTD=0x00;DDRD=0x80;


// 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=0x18;
UCSRC=0x86;
UBRRH=0x00;
UBRRL=0x33;

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

// ADC initialization
// ADC Clock frequency: 1000.000 kHz
// ADC Voltage Reference: AREF pin
ADMUX=ADC_VREF_TYPE & 0xff;
ADCSRA=0x83;
DDRC.4=1;PORTC.4=0;DDRC.0=0;DDRC.1=0;PORTC.1=1;
PORTB=0xff;PORTC.0=1;k_on;s=1;req=0;
while (1)
      {
      android();
      if (c==1 && PINC.0==0){text='J';isi=1;}
      if (isi==1 && text=='J'){h_on;m_off;k_off;
      while (text=='J' && PINC.0==1){android();}c=0;rel_on;isi=0;jalan();
      if(PINC.0==0){s=0;isi=0;rel_off;}}    //jalan
```

```
      if (text=='P'){m_on;k_off;h_off;alarm();s=0;}  //parkir

      if (text=='S'){k_on;m_off;h_off;buz_off;rel_off;s=1;}  //behenti tanpa alarm

      if (isi==0 && PINC.0==0 && s==0){buz_on;}

      //emergency only

      if (req==0 && emr==0){req=1;h_on;m_off;k_off; while
(emr==0){delay_ms(100);}}

      while (req==1){req=1;h_on;m_off;k_off;

      if (req==1 && emr==0){req=0;while
(emr==0){delay_ms(100);h_off;m_off;k_on;}}

      if (PINC.0==0){rel_on;} else {rel_off;}

      }

      }

}
```

LAMPIRAN

(Program Android)

```
#Region  Project Attributes
	#ApplicationLabel: APLIKASI
	#VersionCode: 1
	#VersionName:
	'SupportedOrientations possible values: unspecified, landscape or portrait.
	#SupportedOrientations: unspecified
	#CanInstallToExternalStorage: False
#End Region

#Region  Activity Attributes
	#FullScreen: False
	#IncludeTitle: True
#End Region
'Activity module
Sub Process_Globals

	Dim admin As BluetoothAdmin
	Dim serial1 As Serial
	Dim foundDevices As List
	Type NameAndMac (Name As String, Mac As String)
	Dim connectedDevice As NameAndMac
		Dim AStream As AsyncStreams
		Dim cnt1 As Int
```

```
        Dim cnt2 As Int

End Sub

Sub Globals
Private timer1 As Timer
    Private lblStatus As Label
    'Private txtTarget As EditText
    'Private LBLOK As Label
    'Private LBLREJECT As Label
    Private labeldarimikro As Label
End Sub

Sub Activity_Create(FirstTime As Boolean)

    If FirstTime Then

        'Do not forget to load the layout file created with the visual designer.
For example:
    Activity.LoadLayout("SATU")
    Activity.Title="ALARM MALING"
    Activity.AddMenuItem("Connect", "mnuConnect")



    Activity.AddMenuItem("Disconnect", "mnuDisconnect")
    'Activity.AddMenuItem("Instruction", "mnuInstruction")
```

```
                admin.Initialize("admin")
                serial1.Initialize("serial1")
timer1.Initialize("timer1",5000)
timer1.Enabled=False
        End If

lblStatus.Text = "Not Connected"
End Sub

Sub Button1_Click

End Sub

Sub VR_Result (Success As Boolean, Texts As List)
Dim hasilTextSuara As String
Dim textTampil As String
Dim nyala As String = "1"
Dim mati As String = "2"

        If Success = True Then
        textTampil= Texts.Get(0)
        ToastMessageShow(textTampil,False)
        hasilTextSuara = Texts.Get(0)
        If hasilTextSuara.EqualsIgnoreCase("nyala") =True Then
        textTampil = "Anda Mengucapkan "  &   Texts.Get(0) & " dan akan
menyalakan lampu"
```

```
        AStream.Write(nyala.GetBytes("UTF8"))
        Else If hasilTextSuara.EqualsIgnoreCase("mati") =True  Then
        textTampil = "Anda Mengucapkan "   &  Texts.Get(0) & " dan akan
mematikan lampu"

        AStream.Write(mati.GetBytes("UTF8"))
        Else
        textTampil = "Anda Mengucapkan "  &  Texts.Get(0) & " dan perintahnya
salah gan"

        End If


        End If
End Sub

Sub Activity_Pause (UserClosed As Boolean)
If UserClosed = True Then
                serial1.Disconnect
        End If
End Sub

Sub mnuConnect_Click
foundDevices.Initialize
        If admin.StartDiscovery      = False Then
                ToastMessageShow("Error starting discovery process.", True)
        Else
```

```
            ProgressDialogShow("Searching for devices...")
        End If
End Sub

Sub mnuDisconnect_Click
serial1.Disconnect
End Sub
Sub Admin_DiscoveryFinished
        ProgressDialogHide
        If foundDevices.Size = 0 Then
            ToastMessageShow("No device found.", True)
        Else
            Dim l As List
            l.Initialize
            For i = 0 To foundDevices.Size - 1
                Dim nm As NameAndMac
                nm = foundDevices.Get(i)
                l.Add(nm.Name)
            Next
            Dim res As Int
            res = InputList(l, "Choose device to connect", -1)
            If res <> DialogResponse.CANCEL Then
                connectedDevice = foundDevices.Get(res)
                ProgressDialogShow("Trying to connect to: " &
connectedDevice.Name & " (" & connectedDevice.Mac & ")")
                serial1.Connect(connectedDevice.Mac)
            End If
```

```
        End If
End Sub


Sub Admin_DeviceFound (Name As String, MacAddress As String)
        Log(Name & ":" & MacAddress)
        Dim nm As NameAndMac
        nm.Name = Name
        nm.Mac = MacAddress
        foundDevices.Add(nm)
        ProgressDialogShow("Searching for devices (~ device
found)...".Replace("~", foundDevices.Size))
End Sub


Sub Serial1_Connected (Success As Boolean)
        ProgressDialogHide
        Log("connected: " & Success)
        If Success = False Then
                Log(LastException.Message)
                ToastMessageShow("Error connecting: " & LastException.Message,
True)
                        lblStatus.Text = "Not Connected"
                        timer1.Enabled=False
        Else
        ToastMessageShow("berhasil konnek",True)
                'StartActivity(ChatActivity


                        lblStatus.Text = "Connected"
```

```
                timer1.Enabled=True
        If AStream.IsInitialized = False Then
            'AStream.InitializePrefix(Main.serial1.InputStream, True,
Main.serial1.OutputStream, "AStream")
            AStream.Initialize(serial1.InputStream, serial1.OutputStream,
"AStream")
        End If
        End If
End Sub
Sub timer1_Tick
If lblStatus.Text = "Connected" Then
        Dim Buffer() As Byte

Buffer="*J".GetBytes("UTF8")
        AStream.Write(Buffer)
End If
End Sub

Sub AStream_Terminated
        AStream_Error
            lblStatus.Text = "Not Connected"
            timer1.Enabled=False
End Sub
Sub AStream_Error
        ToastMessageShow("Connection is broken.", True)
            lblStatus.Text = "Not Connected"
            timer1.Enabled=False
```

```
End Sub
Sub BTNSTART_Click
Dim tmpkirim As String ="*J"
Dim buffer() As Byte
buffer=tmpkirim.GetBytes("UTF8")
        AStream.Write(buffer)

End Sub
Sub AStream_NewData (Buffer() As Byte)
    Dim msg As String

    msg = BytesToString(Buffer, 0, Buffer.Length, "UTF8")
   labeldarimikro.Text =msg
   If  labeldarimikro.Text="*R" Then
        Dim Buffer() As Byte

Buffer=labeldarimikro.text.GetBytes("UTF8")
        AStream.Write(Buffer)
   labeldarimikro.Text=""
   End If
End Sub
Sub BTNSTOP_Click
        Dim tmpkirim As String ="*P"
        Dim buffer() As Byte
```

```
buffer=tmpkirim.GetBytes("UTF8")
        AStream.Write(buffer)

End Sub
```