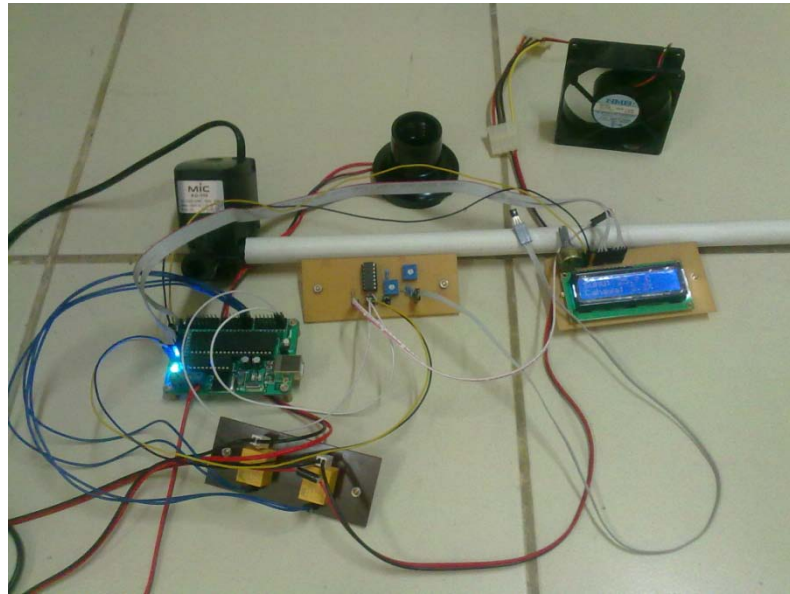


LAMPIRAN

A

LAMPIRAN A

FOTO ALAT

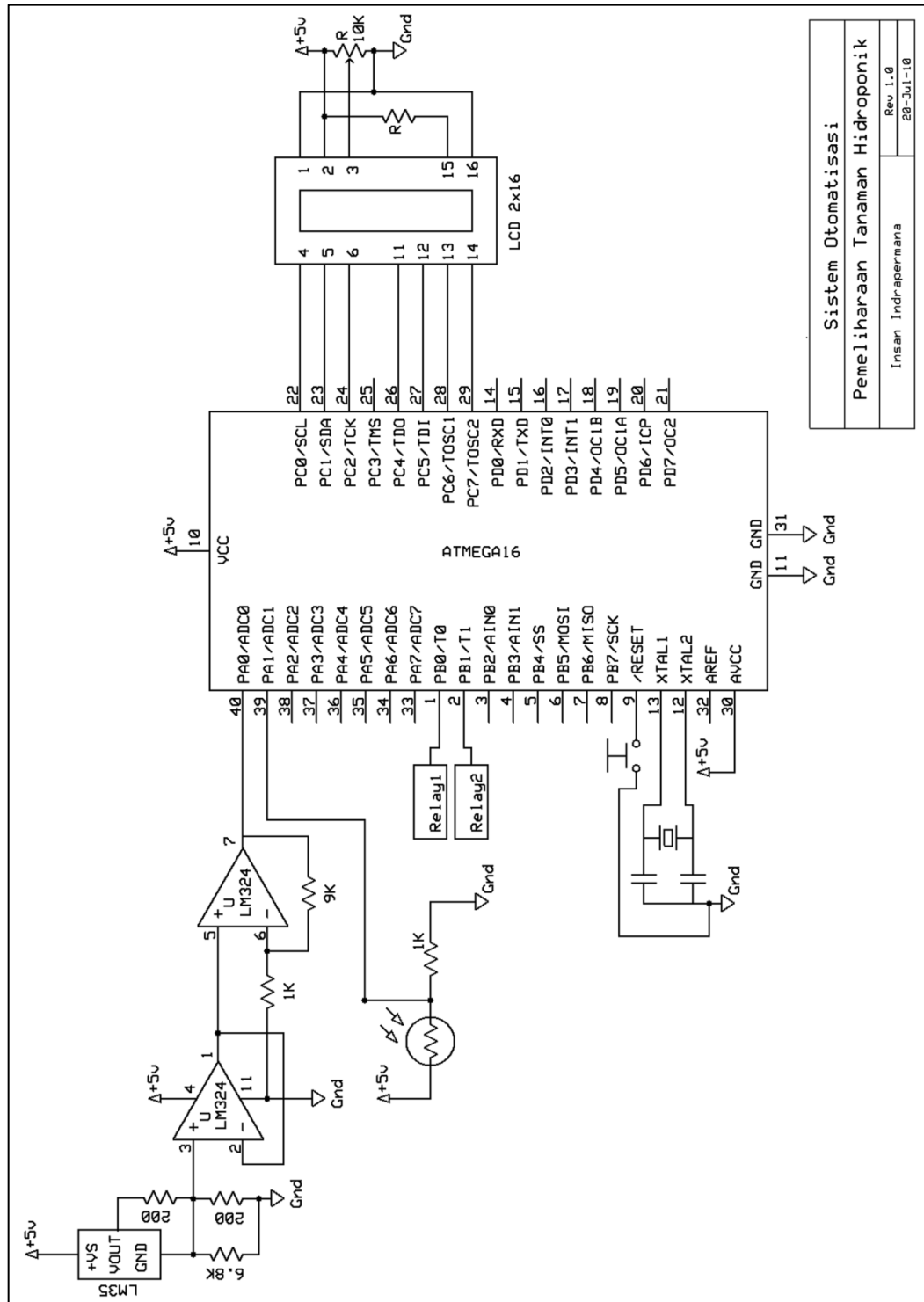


LAMPIRAN

B

LAMPIRAN B

SKEMA RANGKAIAN



LAMPIRAN

C

LAMPIRAN C

PROGRAM

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

```
This program was produced by the  
CodeWizardAVR V2.03.9 Standard  
Automatic Program Generator  
© Copyright 1998-2008 Pavel Haiduc, HP InfoTech s.r.l.  
http://www.hpinfotech.com
```

```
Project : Sistem Otomatisasi Pemeliharaan Tanaman  
Hidroponik  
Version : 1.00  
Date : 07-Jul-2010  
Author : Insan Indrapermana - 0727006  
Comments: Proyek Tugas Akhir
```

```
Chip type : ATmega16  
Program type : Application  
AVR Core Clock frequency: 8.000000 MHz  
Memory model : Small  
External RAM size : 0  
Data Stack size : 256
```

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

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

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

```

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

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

#define ADC_VREF_TYPE 0x40

// 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;
}

// Declare your global variables here

void main(void)
{

```

```

// Declare your local variables here
char suhu_buffer[16];
char cahaya_buffer[16];
unsigned int suhu;
unsigned int cahaya;

// 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=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
PORTB=0xFF;
DDRB=0xFF;

// Port C 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
PORTC=0xFF;
DDRC=0xFF;

```

```

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

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

// ADC initialization
// ADC Clock frequency: 125.000 kHz
// ADC Voltage Reference: AVCC pin
// ADC Auto Trigger Source: None
ADMUX=ADC_VREF_TYPE & 0xff;
ADCSRA=0x86;

// LCD module initialization
lcd_init(16);

while (1)
{

```

```

// Place your code here

suhu = read_adc(0); //membaca sensor suhu
cahaya = read_adc(1); //membaca sensor cahaya
delay_ms(50);

//Konversi suhu
sprintf(suhu_buffer, "Suhu: %0.1f C", (float)suhu*0.0488);
lcd_gotoxy(0,0);
lcd_puts(suhu_buffer);

//Konversi Cahaya
sprintf(cahaya_buffer, "Cahaya: %0.1f", (float)cahaya*0.0977);
lcd_gotoxy(0,1);
lcd_puts(cahaya_buffer);
lcd_putsf("% ");
delay_ms(50);

//Proses membandingkan data dan pengontrolan
if((suhu<=513)&&(cahaya>=308)) PORTB = 0b11111111;
if((suhu<=513)&&(cahaya<=103)) PORTB = 0b11111101;
if((suhu>=574)&&(cahaya>=308)) PORTB = 0b11111110;
if((suhu>=574)&&(cahaya<=103)) PORTB = 0b11111100;
delay_ms(1000);

};
}

```