

IMPLEMENTASI SENSOR SUHU LM35 BERBANTUAN MIKROKONTROLER PADA PERANCANGAN SISTEM PENGKONDISIAN SUHU RUANGAN

Saefurrochman¹, Arief Goeritno², Rakhmad Yatim³, Dwi Jatmiko Nugroho⁴

¹SMK Negeri 2 Bogor

Jl. Pangeran Sogiri no. 404, Tanah Baru, Bogor Utara, Kota Bogor 16710 Telepon: 0251 8659385

²Jurusan Teknik Elektro, Fakultas Teknik, Universitas Ibn Khaldun Bogor

Jl. Sholeh Iskandar km.2, Kedung Badak, Tanah Sareal, Kota Bogor 16132 Telepon: 0251 8356884

³Pusat Teknologi Satelit, Lembaga Penerbangan dan Antariksa Nasional (LAPAN)

Jl. Cagak Satelit km.4, Rancabungur, Kabupaten Bogor 16310 Telepon: 0251 8623010

⁴PT Tirta Fresindo Jaya

Jl. Mayjend H.E. Edi Sukma km.16, Caringin, Kabupaten Bogor 16730

Email: the_reams@yahoo.com

Abstract

Implementation of LM35 temperature sensor aided the microcontroller on design room temperature conditioning system has been done, through the steps of: (1) integration LM35 temperature sensor to the microcontroller system, (2) programming of the Arduino Uno R3 microcontroller, and (3) verification of the test program. Integration into the microcontroller system, such as the provision of four main ports of microcontroller was connected to: i) 5 volt dc power supply, ii) LM35, iii) electromechanical relays, and iv) 2x16 LCD. Based on the sensor connection to the Arduino Uno R3 obtained orders addressing "int tempPin= 0;" for sensor-1 serves to measure the temperature in the room and "int tempPin = 1;" for sensor-2 is used to measure the rate of cooling the thermoelectric cooling system. Verification test was conducted to determine the achievement of the target value. The target temperature of 25 °C for the detection performed by the sensor-1 and 10 °C for the detection performed by the sensor-2. These conditions are described in the description, namely (a) if the value of the detected temperature is upper than 25 °C, then the blower fan will operate ("on"), (b) if the value of the detected temperature is less than 25 °C, then the blower fan will stop operating ("off"), and (c) if the temperature that detected by sensor-1 more than the value of 25 °C and temperature that detected by sensor-2 in thermoelectric cooling system is below the value of 10 °C, then the blower fan and thermoelectric cooling system will operate ("on").

Keywords: *Arduino Uno R3, conditioning room temperature, the LM35 temperature sensor.*

1. PENDAHULUAN

Sensor suhu secara elektrik paling sering digunakan, tetapi sulit dalam implementasinya, seperti implementasi termokopel yang memiliki tingkat keluaran (*output*) yang rendah dan butuh kompensasi di sisi dingin atau implementasi termistor yang cenderung nonlinear (National

Semiconductor, 2002). Selain hal itu, keluaran sensor-sensor tersebut tidak berbanding lurus terhadap setiap skala penunjukan nilai suhu. Keberadaan sensor monolitik, seperti LM3911, LM134, dan LM135 telah mampu mengatasi kesulitan-kesulitan tersebut, tetapi keluarannya masih terkait dengan skala suhu derajat Kelvin daripada derajat Celcius atau Fahrenheit yang

lebih populer. Dalam perkembangannya, telah dikenalkan pada tahun 1983, dua rangkaian terintegrasi (*integrated circuit, IC*), yaitu sensor LM34 yang presisi terhadap skala suhu Fahrenheit (National Semiconductor, 2002) dan sensor seri LM35 dengan keluaran berupa tegangan yang presisi dan berbanding lurus dengan derajat Celsius (National Semiconductor, 2002; National Semiconductor, 1995; National Semiconductor, 2000).

Sensor LM35 memiliki keuntungan lebih linear dibandingkan sensor suhu yang terkalibrasi dalam derajat Kelvin, sehingga tidak diperlukan pengurangan besar konstanta tegangan keluaran untuk perolehan skala dalam derajat Celsius. LM35 tidak memerlukan kalibrasi eksternal, karena telah memiliki akurasi yang sangat khas, yaitu $\pm 0,25\text{ }^{\circ}\text{C}$ pada suhu kamar dan $\pm 0,25\text{ }^{\circ}\text{C}$ pada skala kisaran penuh dari -55 sampai +150 $^{\circ}\text{C}$, sehingga pembuatan antarmuka (*interface*) untuk pembacaan atau rangkaian pengontrol sangat mudah. LM35 sangat murah, impedans keluaran rendah dan linear, hanya terjadi penambahan 0,1 kali untuk setiap kenaikan beban sebesar 0,1 mA. Sisi keunggulan lain dari LM35, yaitu hanya butuh pasokan arus listrik 60 mA dengan tegangan masukan 4-30 volt, dan pemanasan internal yang timbul kurang dari 0,1 $^{\circ}\text{C}$ dalam udara. LM35 yang tersedia dalam kemasan berupa paket transistor keatas udara TO-46 (National Semiconductor, 2000; Peinhopf, 2006, Intel Corporation, 1997).

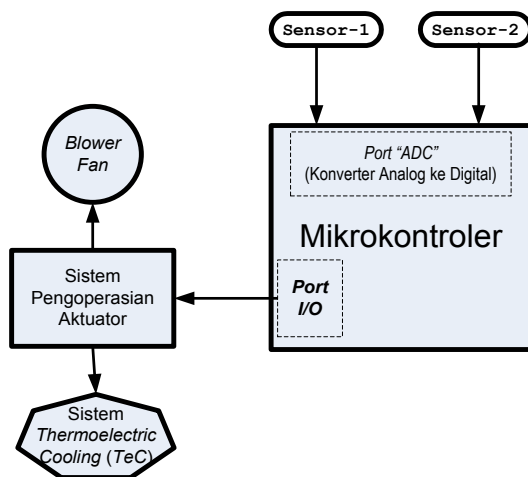
Mikrokontroler (pengontrol mikro, *microcontroller*) adalah sistem mikroprosesor lengkap yang terkandung di dalam sebuah *chip* (Putra, 2010), sehingga mikrokontroler dapat dibuat lebih murah dalam kemasan yang lebih kecil dengan jumlah kaki (*pin*) lebih sedikit. Sebuah *chip* mikrokontroler umumnya memiliki: (i) *central processing unit (CPU*, mulai dari prosesor 4-bit yang sederhana hingga prosesor kinerja tinggi 64-bit); (ii) antarmuka masukan/keluaran (*input/output*) jaringan, seperti *port serial (Universal Asynchronous Receiver/Transmitter, UART)*; (iii)

antarmuka komunikasi serial lain, seperti *inter integrated circuits (I²C)*, *serial peripheral interface and controller area network* untuk sambungan sistem; (iv) periferal, seperti *timer* dan *watchdog*; (v) *random access memory (RAM)* untuk penyimpanan data; (vi) *read only memory (ROM)*, *erasable programmable read only memory (EPROM)*, *electrical erasable programmable read only memory (EEPROM)*, atau *flash memory* untuk penyimpanan program komputer; (vii) pembangkit *clock* (biasanya berupa resonator rangkaian RC), dan (viii) pengubah analog ke digital (*analog to digital converter, ADC*). Sistem mikrokontroler lebih banyak untuk pekerjaan-pekerjaan sederhana tetapi penting, seperti pengontrolan motor, pengendalian saklar, pengaturan resistor variabel, atau perangkat elektronis lain (Putra, 2010). Keberadaan mikrokontroler untuk pekerjaan-pekerjaan tersebut, maka mikrokontroler telah mengandung syarat minimal guna sebagai sebuah sistem mikroprosesor, yakni memori untuk data dan program dan sistem antarmuka (*interface*) masukan/keluaran (*input/output, I/O unit*) sederhana, sehingga mikrokontroler tidak selalu memerlukan memori eksternal (Putra, 2010).

Saat mikrokontroler sebagai sistem *embedded*, maka mikrokontroler tersebut mempunyai fungsi pokok tidak melakukan komputasi, tetapi sebagai sesuatu yang dikontrol oleh komputer yang tertanam di dalamnya (Abandah, 2013). Untuk kondisi dimana mikrokontroler difungsi sebagai *embedded product*, maka langkah-langkah beroperasinya, yaitu: (langkah-1) Nilai yang terdapat pada *program counter register* dalam mikrokontroler mengambil data pada *ROM* dengan alamat sebagaimana yang tertera pada *program counter register*, selanjutnya isi *program counter register* ditambah dengan satu (*increment*) secara otomatis, sehingga data yang diambil pada *ROM* merupakan urutan instruksi program yang telah dibuat dan diisikan sebelumnya oleh pengguna; (langkah-2) Instruksi yang diambil tersebut diolah dan dioperasikan oleh

mikrokontroler, dimana proses pengerjaan bergantung pada jenis instruksi, dapat membaca, mengubah nilai-nilai pada register, RAM, isi port, atau melakukan pembacaan dan dilanjutkan dengan perubahan data; dan (langkah-3) Program counter register telah berubah nilainya (baik karena penambahan otomatis pada langkah (i), atau karena perubahan-perubahan pada langkah-2, selanjutnya mikrokontroler akan mengulang-ulang kembali siklus pada langkah-1 hingga pasokan daya ke mikrokontroler terputus atau diputus.

Berdasarkan uraian tersebut, maka dibuat rancangan sistem berbantuan sensor suhu LM35 dan mikrokontroler *Arduino Uno R3* untuk pengkondisian suhu ruangan. Diagram skematis rancangan sistem pengkondisian suhu berbantuan sensor suhu dan mikrokontroler, seperti ditunjukkan pada Gambar 1.



Gambar 1 Diagram skematis rancangan sistem pengkondisian suhu berbantuan sensor suhu dan mikrokontroler

Berdasarkan Gambar 1, implementasi sensor suhu LM35 berbantuan mikrokontroler untuk pengkondisian suhu ruangan melalui pengoperasian sistem aktuator untuk *blower fan* dan sistem *thermoelectric cooling (TeC)* berdasarkan keadaan yang dideteksi oleh sensor berupa tujuan penelitian, yaitu: (a) pengintegrasian sensor suhu LM35 ke sistem mikrokontroler

Arduino Uno R3, (b) pemrograman mikrokontroler *Arduino Uno R3*, dan (c) uji verifikasi terhadap program.

2. BAHAN DAN METODE

Bahan

Untuk keperluan pelaksanaan metode penelitian, diperlukan bahan penelitian berupa: (i) sensor suhu LM35, (ii) board untuk *Arduino Uno R3*, (iii) sejumlah resistor, kapasitor, dan relai elektromekanik, (iv) IC regulator LM358, (v) LCD 2x16, (vi) catu daya (*power supply*) 5 Vdc dan 12 Vdc, dan (viii) program aplikasi, meliputi: *Integrated Development Enviroment (IDE) Arduino* dan *Proteus*. *IDE Arduino*, adalah *software* pemrograman bebas lisensi (gratis) untuk *Arduino Uno R3* yang dirancang oleh Masimo Banzi, David Cuartielles, Tom Igoe, Gianluca Martino, dan David Melis dan dapat diunduh (*downloaded*) di <http://www.arduino.cc>.

Proteus dikembangkan oleh *Labcenter Electronics*, adalah sebuah perangkat lunak yang *user friendly* dengan kemudahan dalam pembuatan diagram skematis rangkaian elektronika, pengembangan PCB, dan pensimulasian mikroprosesor (*Labcenter Electronics, 2007; NETZSCH, no year*).

Metode

Metode penelitian yang dilakukan, adalah tahapan untuk perolehan setiap tujuan penelitian.

1) Pengintegrasian sensor suhu LM35 ke sistem mikrokontroler *Arduino Uno R3*

Tahapam pengintegrasian, adalah: (a) penetapan diagram pengawatan untuk sistem mikrokontroler yang dipilih, yaitu *Arduino Uno R3* yang dapat langsung diunduh dari laman

http://arduino.cc/en/uploads/Main/Arduino_Uno_Rev3-schematic.pdf dan (b) pembuatan rangkaian sistem berbasis mikrokontroler *Arduino Uno R3* berbantuan program aplikasi *Proteus* (*Labcenter Electronics, 2007; E-MU, 2007; E-mu Systems, Inc., no year; Microcomputer System Design, no year; NETZSCH, no year*).

2) Pemrograman mikrokontroler Arduino Uno R3

Tahapan untuk pemrograman, adalah: (a) pembuatan algoritma dalam bentuk diagram alir dan (b) penulisan sintaks pemrograman berbasis bahasa C (Burgess, 1999).

3) Uji verifikasi terhadap program

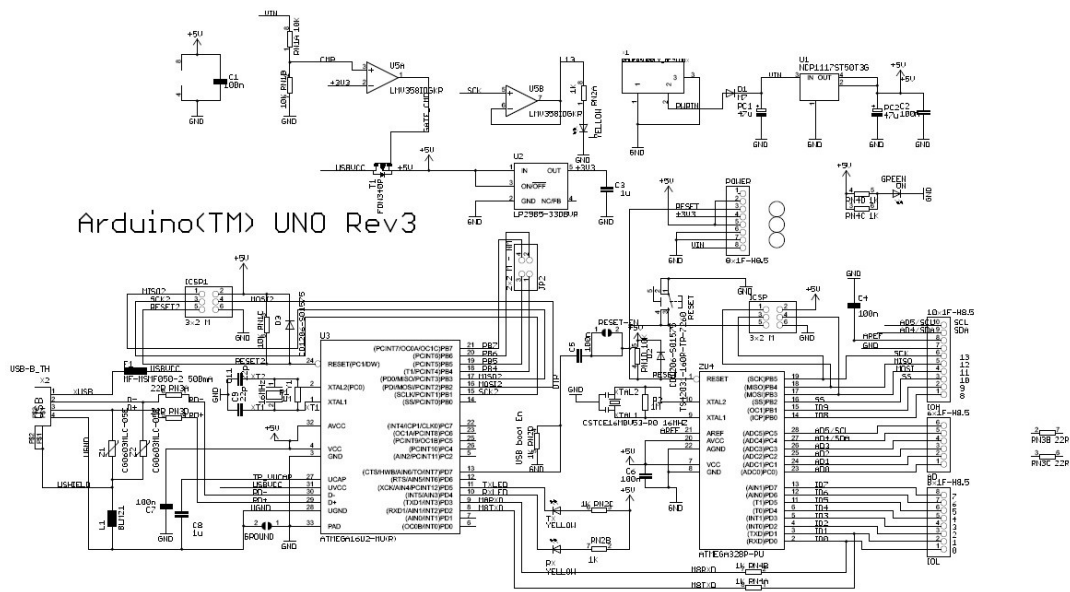
Tahapan uji verifikasi program berbasis bahasa C (Burgess, 1999) yang telah dibuat diujikan ke dalam program aplikasi Proteus (Labcenter Electronics, 2007; E-MU, 2007; E-mu Systems, Inc., no year; Microcomputer System Design, no year; NETZSCH, no year)., adalah: (a) pemantauan dan pengukuran suhu melalui simulasi pemberian kondisi berbeda terhadap sensor dan (b) penjelasan mekanisme pengoperasian aktuator untuk blower fan dan sistem thermoelectric cooling berdasarkan keadaan yang dideteksi oleh sensor.

Hasil dan Bahasan

Berdasarkan tujuan penellitian, maka diuraikan hasil dan bahasan yang meliputi integrasi sensor suhu LM35 ke sistem mikrokontroler, pemrograman untuk *Arduino Uno R3*, dan uji verifikasi program.

Integrasi Sensor LM35 ke Sistem Mikrokontroler

Integrasi berupa penetapan diagram pengawatan sistem mikrokontroler untuk penyediaan jalur masukan dan keluaran. Diagram pengawatan sistem mikrokontroler yang dipilih untuk perancangan, seperti ditunjukkan pada Gambar 2.



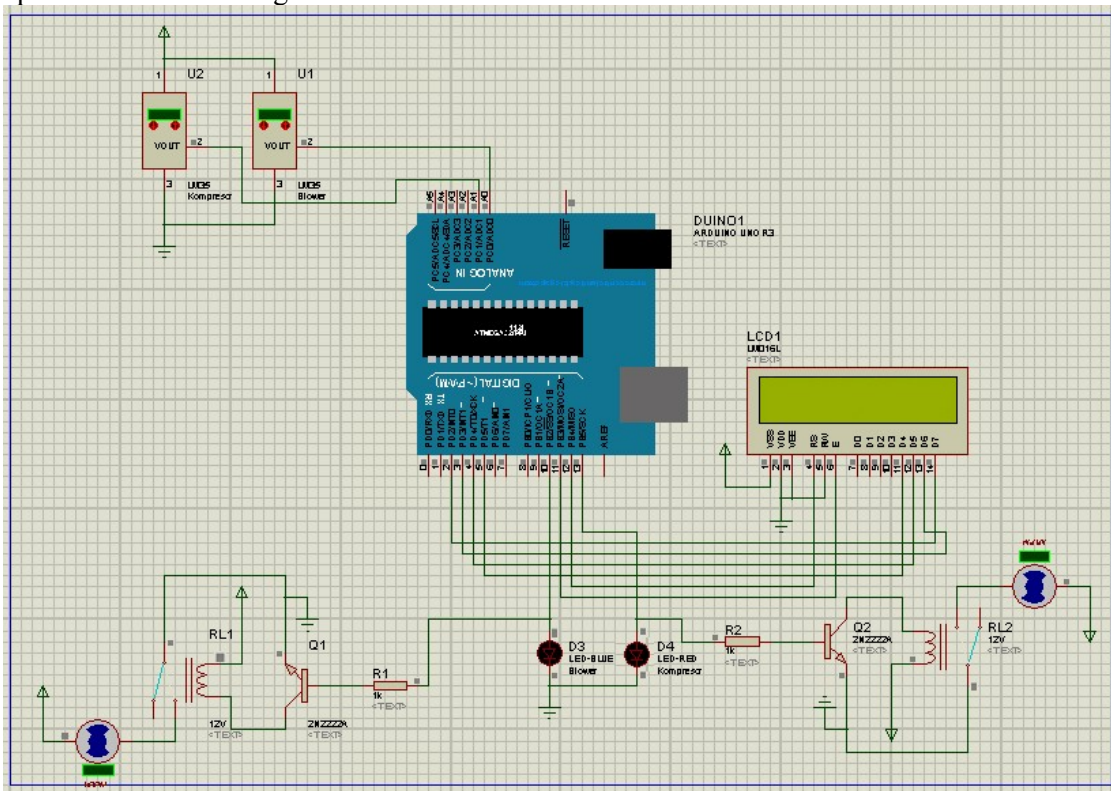
Gambar 2 Diagram pengawatan sistem mikrokontroler yang dipilih untuk perancangan

Berdasarkan rangkaian pada Gambar 2, maka dibuat *board* sistem mikrokontroler *Arduino Uno 3R* untuk pengkondisian suhu.

Board sistem mikrokontroler *Arduino Uno 3R* dibutuhkan untuk pembuatan aplikasi guna simulasi pengkondisian suhu ruangan. Untuk

pengujian terhadap pengawatan dan tata letak komponen pada sistem pengkondisian suhu ruangan, dilakukan penanaman program pada aplikasi *Proteus*. Pengawatan antara sensor

suhu, penampil (*display*), rangkaian pengoperasian aktuator, dan sistem Arduino Uno R3, seperti ditunjukkan pada Gambar 3.



Gambar 3 Pengawatan antara sensor suhu, tampilan, rangkaian pengoperasian aktuator, dan sistem Arduino Uno R3

Berdasarkan Gambar 3 ditunjukkan, bahwa selain sensor LM35 yang terhubung ke sistem mikrokontroler, terdapat juga penampil (berupa *LCD 2x16*), dan rangkaian pengoperasian aktuator untuk *blower fan* dan sistem *Thermoelectric Cooling (TeC)*. Pengoperasian sensor saat untuk pengukuran suhu, digunakan sumber tegangan 5 Vdc. Sensor mempunyai satu *pin* pengeluran berupa tegangan yang dikonversikan ke dalam bentuk digital oleh konverter analog ke digital (*analog to digital converter, ADC*), pengambilan, dan pengiriman data. Pengambilan data untuk masing-masing pengukuran dilakukan dengan pemberian perintah pengalamatan oleh mikrokontroler. Data nilai suhu keluaran sensor LM35 berupa nilai tegangan analog, agar sensor dapat

beroperasi dengan teratur. Pemanfaatan *pin* pada sensor LM35 hanya 3 *pin* yang tersedia, dimana masing-masing digunakan untuk keluaran (*pin-2*), *Vdc* (*pin-1*), dan *ground* (*pin-3*).

Port mikrokontroler yang digunakan untuk sensor LM35, yaitu *port-C* (*PC*). *PC0* dan *PC1* digunakan untuk pembacaan data dari sensor suhu LM35 berupa pengukuran suhu (*temperature*). Lima *port* utama pada *board Arduino Uno R3* yang digunakan, adalah untuk: (i) catu daya 5 Vdc untuk *Arduino Uno R3*, (ii) sensor suhu LM35, (iii) *LCD 2x16*, dan (iv) keluaran. Keluaran *Arduino Uno R3* terhubung ke rangkaian penggerak relai elektromekanik. Titik-titik kontak bantu (*auxiliary contacts*) pada relai elektromekanik digunakan untuk pemutus

atau penghubung guna pengoperasian aktuator berupa *blower fan* dan sistem *thermoelectric cooling*. Untuk kondisi dimana suhu sudah mencapai target tertentu, maka keluaran dari mikrokontroler Arduino Uno R3 adalah yang sudah dipilih atau disetel dan memberikan nilai tegangan sebesar 5 Vdc ke koil relai elektromekanik. Penggunaan relai elektromekanik didasarkan kepada kebutuhan nilai arus yang diinginkan untuk pengoperasian alat yang dikontrol (aktuator).

Pemrograman pada Mikrokontroler Arduino Uno R3 Untuk Pengoperasian Sistem Aktuator

Pembacaan suhu dimulai dengan pengiriman sinyal analog ke *Arduino Uno R3*. Program akan mengkonversi nilai yang dibaca oleh sensor suhu LM35 yang masih berupa sinyal analog dan akan dikonversikan ke sinyal digital dengan bantuan ADC (analog To Digital Converter) yang terdapat pada *Arduino Uno R3*. Nilai suhu dapat dihasilkan dengan pengkonversian terlebih dahulu terhadap data hasil pembacaan sensor LM35 sebelum ditampilkan di *LCD 2x16*. Pengkonversian data hasil pembacaan sensor LM35 sebelum ditampilkan, seperti ditunjukkan pada Gambar 4.

```
tempC = analogread*(tempin);
tempC = (5.0*tempC*100.0)/1024.0;
```

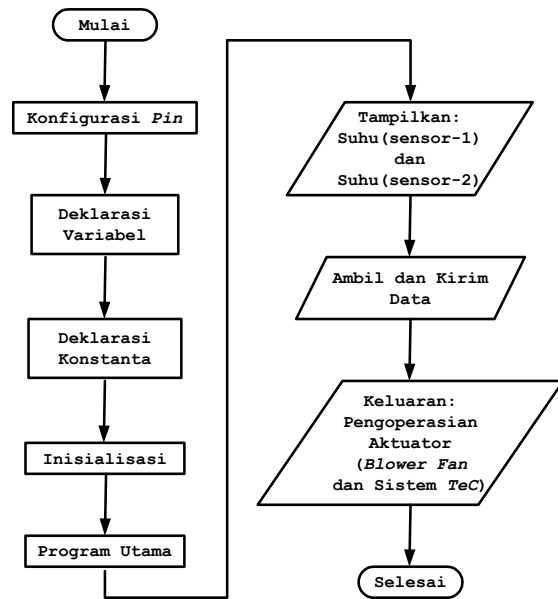
Keterangan: analogread, adalah data hasil pembacaan sensor LM35

Gambar 4 Pengkonversian data hasil pembacaan sensor LM35 sebelum ditampilkan

Algoritma pemrograman mikrokontroler

Pemrograman terhadap *Arduino Uno R3* untuk pengoperasian sistem pengkondisian suhu digunakan *IDE Arduino (Integrated Development Enviroment)*. Fungsi utama *IDE Arduino*, sebagai peng-*comfile* kode program menjadi hexa-desimal (bahasa mesin). Diagram alir pemrograman mikrokontroler *Arduino Uno R3* secara

keseluruhan, seperti ditunjukkan pada Gambar 5.



Gambar 5 Diagram alir pemrograman mikrokontroler *Arduino Uno R3* secara keseluruhan

Berdasarkan Gambar 5 ditunjukkan, bahwa diagram alir untuk pemrograman *Arduini Uno R3* terdiri atas sejumlah tahapan, yaitu: (i) konfigurasi pin, (ii) deklarasi variabel (peubah), (iii) deklarasi konstanta (tetapan), (iv) inisialisasi, (v) program utama, (vi) tampilan: sensor-1 (*line-1*) dan sensor-2 (*line-2*), (vii) ambil dan kirim data, dan (viii) hasil keluaran: aktivasi *blower fan* dan sistem *thermoelectric cooling*.

i) Konfigurasi pin

Konfigurasi pin merupakan penentuan port/pin yang digunakan baik sebagai masukan atau keluaran. Port/pin tersebut dijadikan sebagai parameter dalam setiap pengalamatan program untuk penentuan pin pada *Arduini Uno R3* baik untuk sensor LM35, *LCD 2x16*, dan keluaran berupa *blower fan*, dan sistem *thermoelectric cooling*.

ii) Deklarasi variabel

Deklarasi variabel dilakukan untuk pendeklarasian jenis dari data yang harus dikerjakan.

iii) Deklarasi konstanta

Deklarasi konstanta merupakan pemberian nilai konstanta pada program berdasarkan *datasheet* sensor yang merupakan. Dalam deklarasi konstanta langsung disebut nilai, tidak digunakan tanda titik dua (:) seperti pada deklarasi variabel tetapi digunakan tanda sama dengan (=).

iv) Inisialisasi

Inisialisasi, adalah pemberian inisial terhadap program yang dibuat untuk mengetahui status setiap perintah pada program. Keberadaan inisialisasi diharapkan dapat mempersingkat perintah pada program selanjutnya.

v) Program utama

Program utama merupakan sumber dari pengontrolan program, karena semua perintah pada program diurutkan dari tampilan awal, pengambilan data, penampilan data pada *LCD* dan reaksi atau keluaran dari program yang dibuat.

vi) Tampilkan suhu

Sintaks program

Sintaks program berbasis bahasa C untuk mikrokontroler *Arduino Uno R3* pada proses rancangan sistem pengkondisian suhu ruangan, yaitu:

```

/*****
Implementasi Sensor Suhu LM35 Berbantuan Mikrokontroler pada
Perancangan Sistem Pengkondisian Suhu Ruangan
Author      : Rahman Adya
Date       : 16 November 2014
Email      : the_reams@yahoo.com
*****/

```

```

#include "LiquidCrystal.h";

// Inisialisasi LCD dan menentukan pin yang dipakai
// Pin RS ke Pin 12
// Pin E ke Pin 11
// Pin D7 Ke Pin 2
// Pin D6 Ke Pin 3
// Pin D5 Ke Pin 4
// Pin D4 Ke Pin 5
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
// deklarasi variabel
float tempC; // float sensor 1
float tempC2; // float sensor 2
int tempPin = 0; // Masukan Analog dari sensor 1
int tempPin1 = 1; // Masukan Analog dari sensor 2

void setup() {
  pinMode(10,OUTPUT); // Output Ke Blower
  pinMode(13,OUTPUT); // Output Ke Kompresor
  // Serial.begin(9600);

```

Tampilkan suhu dilakukan untuk mengetahui setiap perubahan yang terjadi pada suatu ruangan atau tempat. Suhu ditampilkan dengan ketentuan untuk suhu ditampilkan pada baris kedua (posisi bawah).

vii) Ambil dan kirim data

Data suhu dan kelembaban yang ditampilkan terlebih dahulu harus dilakukan pengambilan data dari sensor yang digunakan dengan perintah atau ketentuan yang sesuai dengan *datasheet* sensor yang digunakan. Setelah diperoleh datanya, maka data tersebut dikirim untuk dan selanjutnya ditampilkan pada *LCD*.

viii) Keluaran

Keluaran akibat keberadaan sintaks program yang merupakan reaksi terhadap masukan dari sumber masukan atau sensor. Program untuk keluaran diperuntukan pengoperasian dua aktuator (berupa *blower fan* dan sistem *TeC*) akibat setiap perubahan yang terdeteksi oleh sensor.

```

    // Set jumlah kolom dan baris LCD
    lcd.begin(16, 2);
    // Tulis Temperatur di LCD
    lcd.print("Tmp:BLF=");
    Serial.begin(9600);
}

void loop() {
    // Set cursor ke kolom 0 dan baris 1
    // Catatan: Baris dan kolom diawali dengan 0
    lcd.setCursor(9, 0);
    // baca data dari sensor
    tempC = analogRead(tempPin); // Bagian Sensor Suhu Ruangan
    // konversi analog ke suhu
    tempC = (5.0 * tempC * 100.0)/1024.0; // Rumus pembacaan sensor LM35
ke Atmega 328
    // tampilkan ke LCD
    lcd.print(tempC);
    lcd.print (" C");
    Serial.print(tempC); // Kirim data ke serial
    Serial.print(" C Suhu Sensor 1 // ");
    // Serial.println(tempC);
    if(tempC>=30) //jika temperatur >=30 derajat
        digitalWrite(13,HIGH); // output digital high 1
    else digitalWrite(13,LOW); // output digital low 0
    delay(2000); // berhenti 2 detik untuk menunggu perubahan temperatur
        lcd.setCursor(0, 1);
        lcd.print("Tmp:STeC=");
        lcd.setCursor(9, 6);
    // baca data dari sensor
    tempC2 = analogRead(tempPin1); // Bagian Sensor Suhu Kompresor
    // konversi analog ke suhu
    tempC2 = (5.0 * tempC2 * 100.0)/1024.0; // Rumus pembacaan sensor
LM35 ke Atmega 328
    // tampilkan ke LCD
    lcd.print(tempC2); // float dari sensor 2
    lcd.print (" C");
    Serial.print(tempC2); // Kirim data Ke serial
    Serial.print(" C Suhu Sensor 2");
    Serial.println();
    // Serial.println(tempC);
    if(tempC2>=10) //jika temperatur >=10 derajat
        digitalWrite(10,HIGH); // output digital high 1
    else digitalWrite(10,LOW); // output digital low
    delay(2000); // berhenti 2 detik untuk menunggu perubahan temperatur
}

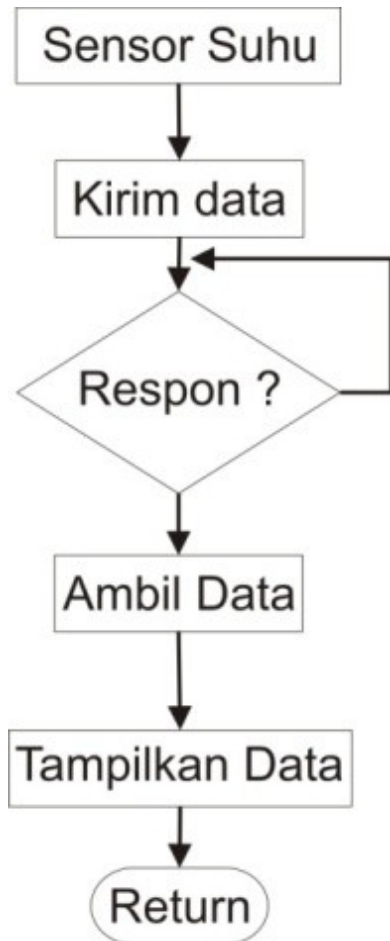
```

Uji Verifikasi

Diperlukan uji verifikasi terhadap program yang telah dibuat untuk pengoperasian aktuator. Program yang telah dibuat dilakukan simulasi berbantuan program aplikasi *Proteus*. Rangkaian terlebih dahulu dirangkai dengan program aplikasi *Proteus*, kemudian program yang sudah dibuat dengan bahasa *C* di-*compile* menjadi

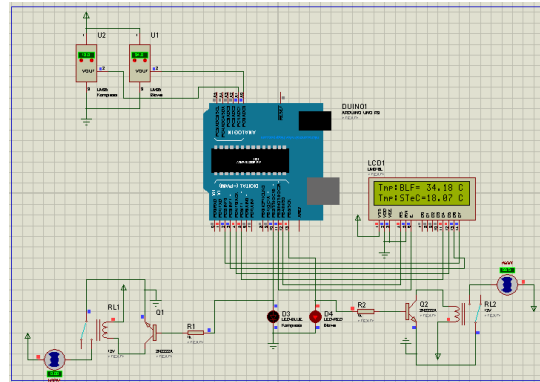
bentuk heksa-desimal (bahasa mesin) dan di-*download*-kan ke rangkaian tersebut. Suhu yang terdeteksi pada simulasi dapat diatur sesuai dengan yang diinginkan untuk pembuktian apakah program yang telah dibuat sudah sesuai dengan yang diharapkan atau tidak. Diagram alir proses pengkondisian suhu berbasis mikrokontroler

pada uji verifikasi, seperti ditunjukkan pada Gambar 6.

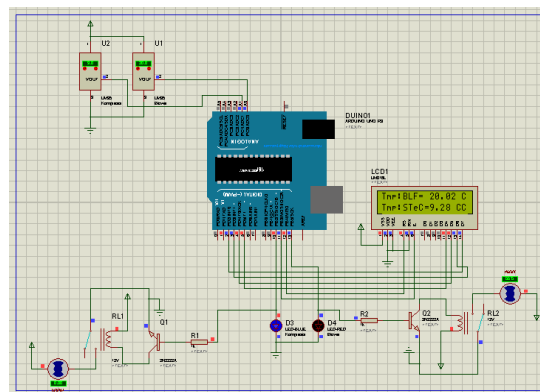


Gambar 6 Diagram alir proses pengkondisian suhu berbasis mikrokontroler pada uji verifikasi

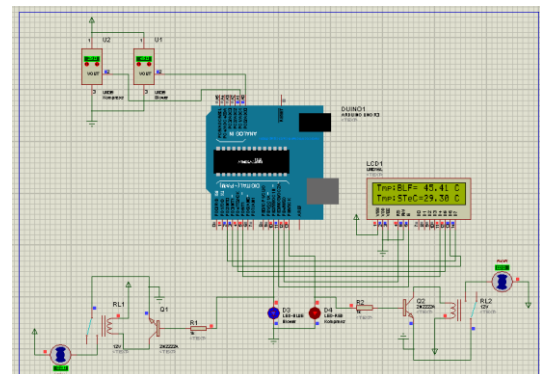
Data suhu yang terdeteksi dikirim melalui alamat yang sudah ditetapkan. Untuk kondisi dimana data suhu terjadi perubahan, maka data tersebut akan diambil dan ditampilkan, tetapi apabila data tidak terdapat perubahan atau tidak terjadi respon, maka data akan tetap pada tampilan data sebelumnya. Pemrograman terhadap sistem pengkondisian suhu berbasis mikrokontroler. Tampilan hasil uji verifikasi pemrograman terhadap mikrokontroler untuk pengoperasian aktuator, seperti ditunjukkan pada Gambar 7.



terhubung ke *blower fan*



terhubung ke sistem *thermoelectric cooling (TeC)*



terhubung ke *blower fan* dan sistem *TeC*

Berdasarkan Gambar 7 ditunjukkan, bahwa target nilai suhu pada sensor-1 dengan nilai 25 °C, dimana (i) jika nilai suhu yang dideteksi lebih besar dari 25 °C, maka *blower fan* akan beroperasi (“on”), (ii) jika nilai suhu yang dideteksi lebih kecil dari 25 °C, maka *blower fan* akan berhenti beroperasi (“off”), dan (iii) jika suhu pada sensor-1 lebih dari

nilai 25 °C dan suhu sensor-2 di dalam sistem *TeC* sudah dibawah nilai 10 °C, maka *blower fan* dan sistem *TeC* akan beroperasi (“on”).

Kesimpulan dan Rekomendasi

Kesimpulan

Berdasarkan hasil dan bahasan, maka ditarik kesimpulan sesuai tujuan penelitian.

- 1) Pengintegrasian sensor ke sistem mikrokontroler ATmega32 ditunjukkan, bahwa: (i) *pin* pada sensor suhu LM35 hanya 3, dimana masing-masing digunakan untuk untuk keluaran (*pin-2*), *Vdc* (*pin-1*), dan *ground* (*pin-3*).; (ii) *port* pada mikrokontroler yang digunakan untuk sensor suhu LM35, yaitu *port C* (*PC*). *PC0* dan *PC1* digunakan untuk pembacaan data dari sensor suhu LM35 berupa pengukuran suhu (*temperature*). Lima *port* utama pada *board Arduino Uno R3* yang digunakan, adalah untuk: (i) catu daya 5 *Vdc* untuk *Arduino Uno R3*, (ii) sensor suhu LM35, (iii) LCD 2x16 , dan (iv) keluaran. Keluaran *Arduino Uno R3* terhubung ke rangkaian pemberi daya ke koil relai elektromekanik.
- 2) Pemrograman mikrokontroler Arduino Uno R3 untuk sistem pengoperasian aktuator, dilakukan penanaman program berbahasa *C* dalam delapan tahapan, yaitu: (i) konfigurasi *pin*, (ii) deklarasi variabel (peubah), (iii) deklarasi konstanta (tetapan), (iv) inisialisasi, (v) program utama, (vi) tampilan: suhu (*line-1*) dan kelembaban (*line-2*), (vii) ambil dan kirim data, dan (viii) keluaran: aktivasi aktuator untuk pengoperasian *blower fan* dan sistem *TeC*. Setiap tahapan disertai dengan sintaks masing-masing sesuai kebutuhan.
- 3) Uji verifikasi terhadap hasil program, bahwa suhu target pada sensor-1 dengan nilai 25 °C, dimana (i) jika suhu lebih besar dari 25 °C, maka *fan blower* akan beroperasi (“on”), (ii) jika suhu lebih kecil dari 25 °C, maka *blower fan* akan berhenti beroperasi (“off”), dan (iii) jika suhu pada sensor-1 lebih dari nilai 25 °C dan suhu sensor 2 di dalam kompresor sudah

dibawah nilai 10 °C, maka *blower fan* dan sistem *TeC* akan beroperasi (“on”).

Rekomendasi

Untuk melengkapi kesimpulan yang telah disampaikan, maka direkomendasikan penelitian lanjutan berupa implementasi sensor LM35 berbantuan mikrokontroler pada rancang bangun sistem pengkodisian suhu ruangan dan pelaksanaan uji validasi dengan kondisi suhu realita yang terjadi atau dapat juga dilakukan pembuatan kondisi buatan. Hal itu didasarkan, bahwa hasil uji verifikasi terhadap rancangan sistem pengkondisian suhu berbasis program aplikasi *Proteus*, telah ditunjukkan kinerja yang memenuhi harapan (ekspektasi).

Daftar Pustaka

- Abandah, G., (*no year*), “Introducing Embedded Systems and the Microcontrollers”, pp. 3-23.
- Burgess, M., (1999), “C Programming Tutorial”, Faculty of Engineering, Oslo College, pp. 1-9.
- E-MU, (2007), “Proteus VX Operation Manual”, E-MU System, Pp. 9-24.
- E-mu Systems, Inc., (*no year*), “Proteus Operation Manual”, E-mu Systems Inc., pp. 27-40.
- Intel Corporation, (1997), “AP577 Application Note: An Introduction to Plastic Pin Grid Array (PPGA) Packaging”, Order Number: 243103-004, pp. 5-6.
- Labcenter Electronics, (2007),”Copyright Notice”, Labcenter Electronics, pp. 5-6.
- Microcomputer System Design, (*no year*), “Introduction to Proteus VSM (Part I)”, Microcomputer System Design, pp. 1-16.
- National Semiconductor, (1995), “LM35/LM35A/LM35C/LM35CA/LM35D Precision Centigrade Temperature Sensors”, National Semiconductor Corporation, pp. 1-12.
- National Semiconductor, (2000), “LM35 Precision Centigrade Temperature Sensors”, National Semiconductor Corporation, pp. 1-13.

- National Semiconductor, (2002), "LM34/LM35 Precision Monolithic Temperature Sensors Introduction", National Semiconductor Application Note 460 October 1986, pp. 1-13.
- NETZSCH, (*no year*), "Software Installation Manual NETZSCH Software Proteus® 6.1.0", Gerätebau GmbH, pp. 10-33.
- Peinhopf, W., (2006), "Application Note AN-TO220 Full PAK", Infineon Technologies AG, pp. 1-6.
- Putra, A.E., Nugraha, D., (2010), "Tutorial Pemrograman Mikrokontroler AVR dengan WinAVR GCC (ATMega 16/32/8535), FREE e-book v1.0, pp.3-5.