PROYEK AKHIR – VE180626

SISTEM MONITORING *FIRE ALARM* BERBASIS ARDUINO DENGAN *WIFI*

Fitria Rahmawati
NRP 10311600010029

Dosen Pembimbing
Ir. Joko Susila, MT.
Dr. Eng. Imam Wahyudi Farid, ST., MT.

Departemen Teknik Elektro Otomasi
Fakultas Vokasi
Institut Teknologi Sepuluh Nopember
Surabaya 2020

-----Halaman ini sengaja dikosongkan-----

FINAL PROJECT – **VE180626**

MONITORING OF FIRE ALARM ARDUINO-BASED WITH WIFI

Fitria Rahmawati
NRP 10311600010029

Supervisor
Ir. Joko Susila, MT.
Dr. Eng. Imam Wahyudi Farid, ST., MT.

AUTOMATION ELECTRICAL ENGINEERING DEPARTEMENT
Faculty of Vocations
Institut Teknologi Sepuluh Nopember
Surabaya 2020

-----Halaman ini sengaja dikosongkan-----

PERNYATAAN KEASLIAN
PROYEK AKHIR

Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan Proyek Akhir saya dengan judul "**Sistem Monitoring *Fire Alarm* Berbasis Arduino Dengan *Wifi***" adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diijinkan dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri.

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka.

Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Surabaya, Januari 2020

Fitria Rahmawati
NRP. 10311600010029

-----Halaman ini sengaja dikosongkan-----

SISTEM MONITORING *FIRE ALARM* BERBASIS ARDUINO MENGGUNAKAN *WIFI*

LEMBAR PENGESAHAN
PROYEK AKHIR

Diajukan Guna Memenuhi Sebagian Persyaratan
Untuk Memperoleh Gelar Ahli Madya
Pada
Departemen Teknik Elektro Otomasi
Fakultas Vokasi
Institut Teknologi Sepuluh Nopember

Menyetujui:

Pembimbing 1,	Pembimbing 2,
	31/01/2020
Ir. Joko Susila, MT.	Dr. Eng. Imam Wahyudi Farid, ST., MT.
NIP. 19660606 199102 1 001	NPP. 199020191186

SURABAYA
JANUARI, 2020

-----Halaman ini sengaja dikosongkan-----

SISTEM MONITORING *FIRE ALARM* BERBASIS ARDUINO DENGAN *WIFI*

Nama Mahasiswa	**: Fitria Rahmawati**
NRP	**: 10311600010029**
Dosen Pembimbing 1	**: Ir. Joko Susila, MT.**
NIP	**: 19660606 199102 1 001**
Dosen Pembimbing 2	**: Dr. Eng. Imam Wahyudi Farid, ST., MT.**
NPP	**: 199020191186**

ABSTRAK

Fire Alarm yang umum dikenal dengan sistem terintegrasi yang didesain dan dibangun untuk mendeteksi adanya gejala kebakaran secara otomatis, untuk kemudian memberi peringatan (*warning*) dalam sistem evakuasi dan ditindak lanjuti secara otomatis maupun manual dengan sistem instalasi pemadam kebakaran (*fire fighting System*). *fire alarm* memberikan akurasi yang memudahkan penggunaan untuk proteksi bangunan bertingkat, installer, dan petugas pemadam kebakaran. Dengan terdeteksinya awal kebakaran, maka tindakan untuk mematikan api dapat segera dilakukan. Sehingga dapat meminimalisir kerugian sejak awal. Maka dari itulah dibuat alat untuk memonitoring kenaikan temperatur dan pendeteksi asap pada suatu ruangan.

Pada alat ini menggunakan Arduino Uno sebagai pengendali. Alat ini juga menggunakan *router* dan *ethernet shield* sebagai *access point* guna menampilkan data pada Android. Pada alat ini juga dapat mengirimkan notifikasi ke perangkat Android.

Hasil yang diperoleh adalah pembacaan nilai temperatur, indikator *smoke detector* dan *heat detector* yang ditampilkan pada Android, serta dapat mengirim notifikasi apabila ketika temperatur pada ruangan berkisar mulai 30°C atau terdeteksinya *heat detector* jika temperatur berkisar mulai 36°C dan saat kepadatan asap (*smoke density*) sudah memenuhi ambang batas (*threshold*) *smoke detector*. Pada pembacaan sensor suhu DS18B20 presentase kesalahan deteksi yaitu 3% yang mengakibatkan terkadang suhu ruangan berbeda 1 angka dari sebenarnya.

Kata Kunci: Sistem Alarm Kebakaran, Arduino Uno, *Ethernet Shield, Android*

-----Halaman ini sengaja dikosongkan-----

ARDUINO-BASED MCFA MONITORING FIRE ALARM SYSTEM WITH WIFI

Student's Name	**: Fitria Rahmawati**
Registration Number	**: 10311600010029**
Supervisor 1	**: Ir. Joko Susila, MT.**
ID	**: 19660606 199102 1 001**
Supervisor 2	**: Dr. Eng. Imam Wahyudi Farid, ST., MT.**
ID	**: 199020191186**

ABSTRACT

Fire alarms are commonly known as integrated systems that are designed and built to detect the symptoms of fire automatically, to then give a warning (warning) in the evacuation system and followed up automatically or manually with a fire fighting installation system (fire fighting system). fire alarm provides accuracy that makes it easy to use for the protection of high rise buildings, installers, and firefighters. With the initial detection of a fire, actions to put out the fire can be done immediately. So as to minimize losses from the start. So from that made a tool to monitor temperature rises and smoke detectors in a room.

This tool uses Arduino Uno as a controller. This tool also uses a router and ethernet shield as an access point to display data on Android. This tool can also send notifications to Android devices.

The results obtained are temperature reading, smoke detector and heat detector indicators displayed on Android, and can send notifications when the temperature in the room ranges from 30 ° C or the detection of heat detector if the temperature ranges from 36 ° C and when the density of smoke (smoke) density) meets the smoke detector threshold. On the reading of the DS18B20 temperature sensor the percentage of detection error is 3% which causes sometimes the room temperature is 1 digit different from the actual.

Keywords: Fire Alarm System, Arduino Uno, Ethernet Shield, Android

-----Halaman ini sengaja dikosongkan-----

KATA PENGANTAR

Puji syukur penulis panjatkan kehadirat Allah SWT yang selalu memberikan rahmat dan hidayah-Nya sehingga Proyek Akhir ini dapat terselesaikan, shalawat serta salam semoga selalu dilimpahkan kepada Rasulullah Muhammad SAW yang telah menunjukkan jalan kebanaran.

Proyek Akhir ini disusun untuk memenuhi persyaratan guna menyelesaikan pendidikan Diploma di Departemen Teknik Elektro Otomasi, Fakultas Vokasi, Institut Teknologi Sepuluh Nopember Surabaya.

Dengan terselesaikannya Proyek Akhir ini tidak akan terwujud tanpa bantuan, dukungan, doa, serta dorongan semangat dari semua pihak yang terkait. Oleh karena itu, penulis mengucapkan terimakasih kepada:

1. Kedua orang tua yang senantiasa mendoakan dan memberikan dukungan dengan tulus tiada henti.
2. Bapak Ir. Joko Susila, MT. selaku dosen pembimbing pertama.
3. Bapak Dr. Eng. Imam Wahyudi Farid, ST., MT. selaku dosen pembimbing kedua.
4. Teman-teman Foxvire DE-11 yang selalu memberikan doa, semangat serta dukungan.
5. Semua pihak yang telah membantu baik secara langsung maupun tidak langsung dalam proses penyelesaian Proyek Akhir ini.

Penulis menyadari dan memohon maaf atas segala kekurangan pada Proyek Akhir ini. Akhir kata, semoga Proyek Akhir ini dapat bermanfaat dalam pengembangan ilmu di kemudian hari.

Surabaya, 1 Desember 2019

Fitria Rahmawati
10311600010029

-----Halaman ini sengaja dikosongkan-----

DAFTAR ISI

HALAMAN JUDUL i
PERNYATAAN KEASLIAN PROYEK AKHIR v
LEMBAR PENGESAHAN vii
ABSTRAK ix
KATA PENGANTAR xiii
DAFTAR ISI xv
DAFTAR GAMBAR xvii
DAFTAR TABEL xx
1 BAB I PENDAHULUAN 1
1.1 Latar Belakang 1
1.2 Permasalahan 1
1.3 Batasan Masalah 2
1.4 Tujuan 2
1.5 Sistematika Laporan 2
1.6 Relevansi 3
2 BAB II TEORI PENUNJANG 5
2.1 *Fire Alarm* 5
2.2 *Photoelectric Smoke Detector* 6
2.3 *Rate of Rise Heat Detector* 8
2.4 Arduino Uno 9
2.5 *Ethernet Shield* 10
2.6 Sensor Suhu DS18B20 11
2.7 *Router* TP-LINK TL-WR840N 13
2.8 Modul Relay 14
2.9 *Driver* Motor 14
3 BAB III PERANCANGAN *HARDWARE* DAN *SOFTWARE* 17
3.1 Perancangan Perangkat Keras (*Hardware*) 18
3.1.1 Rancangan Komunikasi Antar Sensor Menggunakan Arduino Uno 18
3.1.2 Perancangan Komunikasi Arduino Uno Menggunakan *Ethernet Shield* 19
3.1.3 Perancangan *Driver* Motor 19
3.2 Perancangan Perangkat Lunak (*Software*) 19
3.2.1 Perancangan Program Pembacaan Rangkaian Penurun Tegangan 20
3.2.2 Perancangan Program Pembacaan Sensor Suhu DS18B20 23

3.2.3 Perancangan Program Aplikasi Android Sebagai Interface Hasil Monitoring....25
4 BAB IV PENGUJIAN DAN ANALISA....35
4.1 Pengujian *Smoke Detector*....35
4.2 Pengujian *Heat Detector*....39
4.3 Pengujian Sensor Suhu DS18B20....44
4.4 Pengujian Sensor Suhu DS18B20 Dan *Heat Detector*....47
4.5 Pengujian Komunikasi *Ethernet Shield*....47
4.6 Pengujian Pengiriman Data Tanpa Penghalang....47
4.7 Pengujian Koneksi Tanpa Penghalang....48
4.8 Tampilan Pengujian Sistem Secara Keseluruhan....49
5 BAB V PENUTUP....65
5.1 Kesimpulan....65
5.2 Saran....65
DAFTAR PUSTAKA....67
LAMPIRAN A....69
LAMPIRAN B....75
RIWAYAT HIDUP PENULIS....83

DAFTAR GAMBAR

Gambar 2.1 *Fire Alarm* 5
Gambar 2.2 *Photoelectric Smoke Detector* 6
Gambar 2.3 *Light Scattering* 7
Gambar 2.4 *Light Obscuration* 7
Gambar 2.5 Prinsip Kerja *Photoelectric Smoke Detector* 8
Gambar 2.6 *Rate Of Rise* HS-WS19L 8
Gambar 2.7 Arduino Uno 9
Gambar 2.8 *Ethernet Shield* 10
Gambar 2.9 Sensor Suhu DS18B20 11
Gambar 2.10 *Router* TP-LINK TL-WR480N 13
Gambar 2.11 Modul Relay 14
Gambar 2.12 Pompa Air *High Pressure* DC 12 V Mini 15
Gambar 3.1 Blok Fungsional Sistem 17
Gambar 3.2 *Wiring* Diagram Antara Sensor Dengan Arduino ... 18
Gambar 3.3 Pembacaan Rangkaian Penurun Tegangan *Smoke Detector* 20
Gambar 3.4 Pembacaan Rangkaian Penurun Tegangan *Heat Detector* 21
Gambar 3.5 Pembacaan Rangkaian Penurun Tegangan *Smoke Detector* dan *Heat Detector* 22
Gambar 3.6 *Flowchart* Pembacaan Sensor Suhu DS18B20 24
Gambar 3.7 *Flowchart* Pembacaan Sensor ke Aplikasi……...25
Gambar 3.8 Tampilan *Tab Designer* 26
Gambar 3.9 Rancangan Tampilan Awal 26
Gambar 3.10 Rancangan Halaman Utama 27
Gambar 3.11 Realisasi Perancangan Tampilan Awal 28
Gambar 3.12 Realisasi Perancangan Halaman Utama 29
Gambar 3.13 Tampilan *Tab Block* 30
Gambar 3.14 *Block* Menampilkan Data Dari Arduino Uno Ke Android 30
Gambar 3.15 *Block* Mengirimkan Notifikasi dan Pemberian Keterangan 31
Gambar 3.16 Alamat IP Pada *Ethernet Shield* 33
Gambar 3.17 Akses *Website* TP-LINK 33
Gambar 4.1 *Smoke Detector* Dalam Keadaan Mendeteksi 35
Gambar 4.2 *Interface* Aplikasi *Smoke Detector* Ruang 1 37
Gambar 4.3 *Interface* Aplikasi *Smoke Detector* Ruang 2 38

Gambar 4.4 *Interface* Aplikasi *Smoke Detector* Ruang 1 dan Ruang 2 39
Gambar 4.5 Pengujian *Head Detector* 40
Gambar 4.6 *Interface* Aplikasi *Head Detector* Ruang 1 41
Gambar 4.7 *Interface* Aplikasi *Head Detector* Ruang 2 42
Gambar 4.8 *Interface* Aplikasi *Head Detector* Ruang 1 dan Ruang 2 43
Gambar 4.9 Pengujian Sensor Suhu DS18B20 44
Gambar 4.10 *Interface* Aplikasi Suhu Ruangan 46
Gambar 4.11 Pengujian Heat Detector Dengan Sensor Suhu DS18B20 47
Gambar 4.12 *Interface* aplikasi Ketika kondisi *Heat Detector* Dan *Smoke Detector* Ruang 1 Serta *Heat Detector* Dan *Smoke Detector* Ruang 2 Tidak Mendeteksi 49
Gambar 4.13 *Interface* Aplikasi *Smoke Detector* Ruang 1 Mendeteksi 50
Gambar 4.14 *Interface* Aplikasi *Smoke Detector* Ruang 2 Mendeteksi 51
Gambar 4.15 *Interface* Aplikasi Heat Detector Ruang 1 Mendeteksi 52
Gambar 4.16 *Interface* Aplikasi Heat Detector Ruang 2 Mendeteksi 53
Gambar 4.17 *Interface* Aplikasi *Smoke Detector* Dan *Heat Detector* Ruang 1 Mendeteksi 54
Gambar 4.18 *Interface* Aplikasi *Smoke Detector* Dan *Heat Detector* Ruang 2 Mendeteksi 55
Gambar 4.19 *Interface* Aplikasi *Smoke Detector* Ruang 1 Serta *Smoke Detector* Ruang 2 Mendeteksi 56
Gambar 4.20 *Interface* Aplikasi *Heat Detector* Ruang 1 Serta *Heat Detector* Ruang 2 Mendeteksi 57
Gambar 4.21 *Interface* Aplikasi *Heat Detector* Ruang 1 Serta *Smoke Detector* Dan *Heat Detector* Ruang 2 Mendeteksi 58
Gambar 4.22 *Interface* Aplikasi *Smoke Detector* Dan *Heat Detector* Ruang 1 Serta *Heat Detector* Ruang 2 Mendeteksi 59
Gambar 4.23 *Interface* Aplikasi *Smoke Detector* Ruang 1 Serta *Smoke Detector* Dan *Heat Detector*

Ruang 2 Mendeteksi 60
Gambar 4.24 *Interface* Aplikasi *Smoke Detector* Dan *Heat Detector* Ruang 1 Serta *Smoke Detector* Ruang 2 Mendeteksi 61
Gambar 4.25 *Interface* Aplikasi *Heat Derector* Ruang 1 Serta *Smoke Detector* Ruang 2 Mendeteksi 62
Gambar 4.26 *Interface* Aplikasi *Smoke Detector* Ruang 1 Serta *Heat Detector* Ruang 2 Mendeteksi..... 63
Gambar 4.27 *Interface* Aplikasi *Smoke Detector* Dan *Heat Detector* Ruang 1 Serta *Smoke Detector* Dan *Heat Detector* Ruang 2 Mendeteksi..... 64

DAFTAR TABEL

Tabel 3.1 Konfigurasi *Port* Mikrokontroler 18
Tabel 3.2 Kemungkinan Sensor Mendeteksi.....32
Tabel 4.1 Pengujian *Smoke Detector* Dengan *Smoke Tester*.....35
Tabel 4.2 Hasil Pengujian *Smoke Detector* Ruang 1 36
Tabel 4.3 Hasil Pengujian *Smoke Detector* Ruang 2 36
Tabel 4.4 Hasil Pengujian *Heat Detector* Ruang 1.....40
Tabel 4.5 Hasil Pengujian *Heat Detector* Ruang 2.....40
Tabel 4.6 Pegujian Sensor Suhu DS18B20.....45
Tabel 4.7 Sensor Suhu DS18B20 di Ukur Tiap 5 Detik 45
Tabel 4.8 Pengiriman Data.....47
Tabel 4.9 Hasil Pengujian Koneksi Ethernet Shield.....48
Tabel 4.10 Pengujian Tegangan dengan Sensor Suhu DS18B20. 48

BAB I
PENDAHULUAN

1.1 Latar Belakang

Secara teknis, "kecelakaan" tidak termasuk dalam kejadian yang disebabkan oleh kesalahan seseorang, akan tetapi peristiwa yang terjadi secara tidak sengaja. Kecelakaan yang sering timbul dan efeknya besar, namun juga terkadang tidak disadari adalah kebakaran. Keamanan terhadap kebakaran adalah hal yang harus diprioritaskan. Keamanan yang dimaksud adalah keamanan terhadap manusia, lingkungan, dan tempat kerja maupun peralatan kerja. Kebakaran bisa diindikasikan dengan adanya asap dan temperatur panas yang tinggi [1]. Oleh karena itu terdapat suatu sistem untuk memberikan peringatan serta penanggulangan dini kebakaran di dalam gedung yaitu dengan *fire alarm system* atau sistem alarm kebakaran. *Fire alarm system* adalah suatu sistem terintegrasi yang didesain dan dibangun untuk mendeteksi adanya gejala kebakaran, untuk kemudian memberi peringatan dalam sistem evakuasi dan ditindak lanjuti secara otomatis maupun manual dengan sistem instalasi pemadam kebakaran.

Dengan terdeteksinya awal kebakaran, maka tindakan untuk mematikan api dapat segera dilakukan. Sehingga dapat meminimalisir kerugian sejak awal. Detektor kebakaran adalah suatu alat yang berfungsi mendeteksi secara dini kebakaran, agar kebakaran yang terjadi tidak berkembang menjadi lebih besar. Dengan terdeteksinya cikal bakal kebakaran yang ditampilkan pada LCD yang terletak pada panel maka tindakan untuk mematikan api dapat segera dilakukan. Oleh karena itu diperlukan suatu alat yang dapat membantu untuk memonitoring sensor asap dan suhu tanpa perlu melihat LCD yang terletak pada panel supaya dapat dipantau dari mana saja.

1.2 Permasalahan

Sistem monitoring suhu dan asap saat ini hanya sebatas ditampilkan pada LCD, sehingga apabila tidak ada yang memantau jika terjadi kemungkinan adanya kebakaran, tindakan pencegahan tidak dapat segera dilakukan.

1.3 Batasan Masalah

Adapun batasan masalah yang dibahas dalam Proyek Akhir ini meliputi:

1. Alat ini hanya dapat memonitoring berupa sensor suhu dan asap
2. Komunikasi data menggunakan *ethernet shield*
3. Kontroler yang digunakan adalah arduino uno

1.4 Tujuan

Tujuan yang akan dicapai dari Proyek Akhir ini adalah:

1. Dapat memonitoring sensor suhu dan asap melalui interface Android.
2. Dapat memonitoring sensor suhu dan asap secara wireless.

1.5 Sistematika Laporan

Pembahasan Proyek Akhir ini akan dibagi menjadi lima bab dengan sistematika sebagai berikut:

Bab I Pendahuluan

Bab ini meliputi latar belakang, permasalahan, batasan masalah, tujuan, sistematika laporan, dan relevansi.

Bab II Teori Penunjang

Dalam bab ini menjelaskan tinjauan tentang *Fire Alarm*, Arduino Uno, *Ethernet Shield, Photoelectric Smoke Detector, Rate of Rise Heat Detector,* Sensor Suhu DS18B20, *Router TP-LINK TL-WR840N, Modul* Relay, *Driver* Motor.

Bab III Perancangan Alat

Dalam bab ini membahas perancangan sistem *hardware* maupun *software* pada sistem monitoring berdasarkan teori dasar pada Bab II.

Bab IV Pengujian dan Analisa

Dalam bab ini membahas tentang pengujian, serta pengukuran terhadap prinsip kerja dan proses dari suatu alat yang dibuat.

Bab V Penutup

Dalam bab ini berisi penutup yang menjelaskan tentang kesimpulan dari Proyek Akhir dan saran – saran untuk pengembangan alat ini lebih lanjut.

1.6 Relevansi

Dengan adanya sistem monitoring *Fire Alarm* yang menggunakan *wifi* diharapkan bisa mempermudah untuk *monitoring smoke detector* dan *heat detector* tanpa perlu melihat LCD yang terpasang pada panel.

-----Halaman ini sengaja dikosongkan-----

BAB II
TEORI PENUNJANG

Pada bab ini dibahas mengenai teori yang dapat menunjang dalam proses pembuatan Proyek Akhir.

2.1 *Fire Alarm*

Gambar 2.1 *Fire Alarm*

Sumber : https://www.tedsystems.com/wp-content/uploads/2019/03/fire-alarm-on-wall-1024x683.jpg

Definisi *fire alarm system* adalah sistem pendeteksi keberadaan api secara otomatis dengan melihat perubahan-perubahan yang terjadi di lingkungan sekitar yang berkaitan dengan kebakaran. Perubahan pada lingkungan sekitar dapat diasumsikan sebagai tanda pendeteksi bahaya kebakaran. Perubahan yang mungkin terjadi misalnya adalah munculnya asap, meningkatnya suhu ruangan, dan munculnya api ataupun gas. Maka dari itu, sebuah *fire alarm system* selalu dilengkapi dengan sensor yang peka terhadap keberadaan asap, panas, api, maupun gas. *Fire alarm system* ini dapat dibedakan menjadi dua, yaitu sistem konvensional dan *addressable*. Sistem konvensional umumnya digunakan pada bangunan yang tidak terlalu besar sedangkan sistem *addressable* dapat digunakan pada bangunan besar karena sistemnya menggunakan kode digital yang dapat mendeteksi langsung lokasi terjadinya kebakaran pada suatu bangunan. Sistem *addressable* hanya terhubung dalam satu panel alarm yang biasanya ditempatkan di ruangan kontrol. Pada sistem konvensional, tiap zona membutuhkan panel alarm sendiri. Oleh karenanya, *fire alarm system* konvensional ini terbatas apabila digunakan pada bangunan-bangunan besar.

Pada bangunan besar, sensor asap, panas, maupun api yang terdapat di seluruh bangunan terhubung kepada satu panel alarm utama. Penggunaan sensor pada *fire alarm system* disesuaikan dengan karakteristik ruangan. Sensor asap tidak dapat diletakkan di dapur, karena dapur merupakan penghasil asap. Sebaiknya pakailah sensor pendeteksi api atau gas. Begitu pula pada kondisi ruangan panas, detektor yang paling sesuai digunakan adalah sensor gas dan asap. Komponen yang terdapat pada *fire alarm system* adalah *manual call point* (Gambar 2.1) yang apabila kaca bagian tengah dipecahkan dapat mengaktifkan sirine tanda kebakaran. Komponen selanjutnya adalah *fire bell* yang berfungsi untuk mengeluarkan suara nyaring pada saat terjadi kebakaran. Komponen terakhir yang terdapat pada *fire alarm system* adalah *indicator lamp* yang memiliki dua buah fungsi, pertama yaitu sebagai tanda aktifnya sebuah *fire alarm system* dan yang kedua adalah sebagai penanda adanya kebakaran.

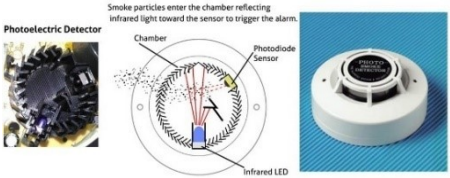
Dari definisi *fire alarm system* tadi, diketahui bahwa sistem ini memiliki cara kerja yang berbeda dengan sistem pemadam kebakaran. Kedua sistem ini berdiri secara mandiri. *Fire alarm system* bukan alat pemadam kebakaran. Tujuan dari *fire alarm system* ini sendiri adalah untuk menginformasikan kebakaran kepada seluruh penghuni bangunan melalui sistem alarm agar dapat segera mengevakuasi diri dan mengosongkan bangunan. *Fire alarm system* sangat diwajibkan untuk dipasang pada gedung-gedung penting dan yang bersifat publik seperti rumah sakit, bank, hotel, pusat perbelanjaan dan sebagainya. Definisi *Fire alarm system* perlu dipahami agar masyarakat cepat tanggap saat terjadi kebakaran [2].

2.2 *Photoelectric Smoke Detector*

Gambar 2.2 *Photoelectric Smoke Detector*

Sumber : https://attackfire.com/images/fire-detector.jpg

Pada Gambar 2.2 *Photoelectric Smoke Detector* atau Detektor Asap Fotolistrik adalah jenis *Smoke Detector* yang menggunakan cahaya untuk mendeteksi adanya gumpalan asap. Saat kepadatan asap (*smoke density*) sudah memenuhi ambang batas (*threshold*), rangkaian elektronik yang terdapat didalam *smoke detector* akan aktif. Karena berisi rangkaian elektronik *smoke detector* membutuhkan tegangan. Prinsip Kerja *Fire alarm Smoke Detector* tipe *Photoelectric (Optical) (*pada Gambar 2.3) *Smoke Detector* bekerja berdasarkan perubahan cahaya di dalam ruang detektor (*chamber*) disebabkan oleh adanya asap dengan kepadatan tertentu.



Gambar 2.3 Prinsip Kerja *Photoelectric Smoke Detector*

Sumber : https://www.bromindo.com/wp-content/uploads/2014/03/8.jpg

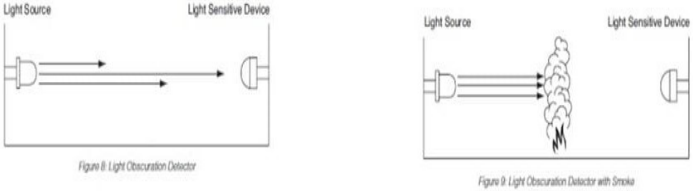
Berdasarkan prinsip kerjanya, kita kenal dua jenis *optical smoke*, yaitu:

- *Light Scattering*. Prinsip ini yang banyak dipakai oleh *smoke detector* saat ini. Terdiri atas *Light-emitting diode* (LED) sebagai sumber cahaya dan *photodiode* sebagai penerima cahaya. LED diarahkan ke area yang tidak terlihat oleh *photodiode* (Gambar 2.4). Jika ada asap yang masuk, maka cahaya akan dipantulkan ke *photodiode*, sehingga menyebabkan *detector* bereaksi.

Gambar 2.4 *Light Scattering*

Sumber : https://www.bromindo.com/wp-content/uploads/2014/03/6.jpg

- *Light Obscuration*. Prinsip ini mirip dengan cara kerja *beam* sensor pada alarm. Cahaya yang terhalang oleh asap menyebabkan *detector* mendeteksi (Gambar 2.5). Prinsip ini pula yang digunakan pada *smoke detector* jenis *infraRed beam*, sehingga bisa mencapai panjang hingga 100m [3].



Gambar 2.5 *Light Obscuration*

Sumber : https://www.bromindo.com/wp-content/uploads/2014/03/7.jpg

2.3 *Rate of Rise Heat Detector*

Gambar 2.6 *Rate of Rise* HS-WS19L

Sumber : https://s2.bukalapak.com/img/239363611/w-1000/Addressable_Rate_of_Rise_Heat_Detector.jpg.webp

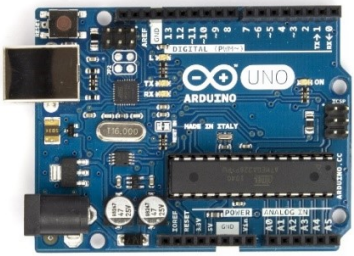
Heat Detector merupakan alat pendeteksi kebakaran melalui suhu panas. Detektor berfungsi untuk mendeteksi peningkatan suhu ruangan secara drastis pada suatu ruangan. Detektor ini dapat terintegrasi dengan *fire alarm control panel*. Detektor ini dapat dibilang lebih efektif dibandikan detektor kebakaran lainnya. Pasalnya, detektor ini dapat lebih cepat menangkap tanda-tanda kebakaran, hanya dengan suhu panas 55°C hingga 63°C.

Heat Detector adalah detektor panas yang dapat diintegrasikan dengan panel *controller (security alarm*). Detektor ini telah dilengkapi dengan fitur *auto-reset* apabila mengalami *trigger alarm*. *Heat Detector* ini dapat ditempatkan pada *ceiling* ruangan dengan berbagai pola interior. Detektor panas pun memiliki jenis lainnya bila dilihat dari perbedaan cara kerjanya. Yakni, jenis *Rate of Rise (ROR)* dan *Fixed Detector*.

Pada jenis ROR (Gambar 2.6), detektor ini mendeteksi perubahan suhu ruangan sebesar 12-15 derajat celsius per menitnya. Sistem pada ROR *heat detector* dapat bekerja dibawah suhu api pada umumnya. ROR detector memanfaatkan teknologi *thermocouple* dan *thermistor* yang responsif dengan panas. Alat pendeteksi panas ini umumnya digunakan pada ruang kantor, perhotelan, rumah sakit, ruang server, ruang arsip, gudang, ataupun kegiatan industri lainya.

Sementara pada *Fixed Detector*, jenis ini mendeteksi panas yang digunakan pada ruangan-ruangan dengan memiliki suhu relatif tinggi. Cara kerja heat detector pada jenis ini dapat bereaksi terhadap panas yang lebih dari 68 derajat celcius. Detector ini sangat cocok digunakan pada ruangan seperti ruang mesin, basement, dan lainnya[4].

2.4 Arduino Uno



Gambar 2.7 Arduino Uno

Sumber : https://3.bp.blogspot.com/-Q0B8D5rOQuk/V9PHzcMEd8I/AAAAAAAAACI/dSB2rp56Pcwe_W2AWLEFyLf1cVyaEunMQCLcB/s1600/Board%2Barduino.jpg

Arduino Uno adalah sebuah *board* yang menggunakan mikrokontroler ATmega328. Arduino Uno memiliki 14 pin digital (6 pin dapat digunakan sebagai output PWM), 6 input analog, sebuah 16 MHz

osilato kristal, sebuah koneksi USB, sebuah konektor sumber tegangan, sebuah *header* ICSP, dan sebuah tombol *reset*. Arduino Uno memuat segala hal yang dibutuhkan untuk mendukung sebuah mikrokontroler. Hanya dengan menghubungkan ke sebuah komputer melalui USB atau memberikan tegangan DC dari baterai atau adaptor AC ke DC sudah dapat membuatnya bekerja [5]. Arduino Uno menggunakan ATmega16U2 yang diprogram sebagai USB to serial *converter* untuk komunikasi serial ke komputer melalui port USB seperti pada Gambar 2.7.

Adapun spesifikasi teknis board Arduino uno adalah sebagai berikut :

- Mikrokontroler : ATmega328
- Tegangan Operasi : 5V
- Tegangan *Input (recommended)* : 7 - 12 V
- Tegangan *Input (limit)* : 6-20 V
- Pin digital I/O : 14 (6 diantaranya pin PWM)
- Pin Analog *input* : 6
- Arus DC per pin I/O : 40 mA
- Arus DC untuk pin 3.3 V : 150 mA
- *Flash Memory* : 32 KB dengan 0.5 KB digunakan untuk *bootloader*
- EEPROM : 1 KB
- Kecepatan Pewaktuan : 16 Mhz

2.5 *Ethernet Shield*

Gambar 2.8 *Ethernet Shield*

Sumber : https://encrypted-tbn0.gstatic.com/images?q=tbn%3AANd9GcRs9y0G1ab5qHlTbabOo12bspIaKooVlCamfZKH96BUX_n1WfPU

Ethernet Shield merupakan suatu perangkat yang dapat menambah kemampuan Arduino untuk terhubung ke jaringan komputer. *Ethernet Shield* berbasiskan cip *Ethernet Wiznet* W5100. Cip *Ethernet Wiznet* W5100 ini menyediakan jaringan internet (IP) baik TCP dan UDP. *Ethernet Shield* juga didukung oleh 4 soket koneksi yang simultan. Penggunaan perangkat ini mengacu pada *library Ethernet Shield* untuk penulisan programnya.

Pada *Ethernet Shield* terdapat sebuah slot *micro*-SD (*Secure Digital*), yang dapat digunakan untuk menyimpan *file* yang diakses melalui jaringan. *Board* Arduino uno dapat berkomunikasi dengan cip *Winzet* W5100 dan SD *card* menggunakan bus SPI (*SERIAL Peripheral interface).* Bus SPI menggunakan pin digital 11, 12, dan 13 pada Arduino uno. Pin digital 10 digunakan untuk memilih W5100 dan pin digital 4 digunakan untuk memilih SD *card*. Pin-pin yang sudah disebutkan sebelumnya tidak dapat digunakan untuk *input/output* umum ketika kita menggunakan *Ethernet Shield*. Karena W5100 dan SD card berbagi bus SPI, hanya salah satu yang dapat aktif pada satu waktu [6]. Pada Gambar 2.8 merupakan bentuk fisik dari *Ethernet Shield*.

2.6 Sensor Suhu DS18B20

Gambar 2.9 Sensor Suhu DS18B20

Sumber : https://encrypted-tbn0.gstatic.com/images?q=tbn%3AANd9GcTr2xQTLCDoce-w0lFuyAwH2U1884dG777F9rqXYGLD8y_JNe8v

Sensor Suhu DS18B20 pada Gambar 2.9 adalah Jenis sensor yang berfungsi untuk mendeteksi suhu ruangan yang merupakan jenis seri sensor terbaru dari keluaran produsen Maxim.

Sensor ini dapat mendeteksi suhu dari -55°C sampai 125°C dengan tingkat keakurasian (+/-0.5°C) dan dengan resolusi 9 – 12-bit. Sensor ini merupakan salah satu jenis sensor suhu yang unik.

Apabila terdapat banyak sensor yang disusun secara paralel data dari keluaran setiap sensor tersebut dapat dibaca hanya dengan menggunakan 1 kabel data atau (*oneWire*) saja.

Sensor ini memiliki 3 kaki

- kaki 1 = GND (0V)
- kaki 2 = VCC (3-5.5 Vdc)
- kaki 3 (s) = Data (*to digital Pin Controller*)

Sensor suhu ini bisa di baca datanya oleh arduino jika di hubungkan dengan cara yang benar dengan memasang resistor *Pull up* resistor 4k7 atau resistor 10k.

Kegunaan dari resistor yaitu sebagai '*pull up*' dari jalur data, dan dapat membantu memastikan proses transfer data tetap berjalan stabil dan baik.

Fitur dari sensor suhu ini

1. *Interface* menggunakan 1-*wire* sebagai komunikasi data
2. Terdapat pengenal unik 64 bit pada setiap sensor
3. Dapat mengukur suhu dari *range* -55°C sampai +125 °C
4. Keakurasian sensor yaitu +/-0.5 °C pada suhu -10 °C sampai +85 °C
5. Resolusi sensor yaitu 9 – 12 bit.
6. Dapat mengkonversi data suhu 12-bit (*digital word*) hanya membutuhkan waktu 750 ms
7. Mempunyai konfigurasi alarm yang dapat di*Setting*

Pengaplikasianya yaitu pada sistem industri, termometer, atau sistem apapun yang memerlukan pembacaan suhu [7].

2.7 *Router* TP-LINK TL-WR840N

Gambar 2.10 *Router* TP-LINK TL-WR840N
Sumber : https://sourceofproduct.com/wp-content/uploads/2019/11/TP-Link-TL-WR840N-300Mbps-Wireless-Router-SOP-2.jpg

Router adalah suatu *hardware* jaringan komputer yang berfungsi untuk mengirimkan paket data melalui jaringan atau internet dari satu perangkat komputer ke perangkat lainnya, dimana proses tersebut disebut dengan *routing*. *Router* sangat banyak digunakan dalam jaringan berbasis teknologi protokol tcp/ip, dan *router* jenis itu disebut juga dengan ip router. Selain *ip router*, ada lagi *appletalk router*, dan masih ada beberapa jenis *router* lainnya. Selain itu dapat juga digunakan untuk menghubungkan banyak jaringan kecil ke sebuah jaringan yang lebih besar, yang disebut dengan *internetwork*, atau untuk membagi sebuah jaringan besar ke dalam beberapa *subnetwork* untuk meningkatkan kinerja dan juga mempermudah manajemennya. *Router* juga kadang digunakan untuk mengoneksikan dua buah jaringan yang menggunakan media yang berbeda (seperti halnya *router wireless* yang pada umumnya selain dapat menghubungkan komputer dengan menggunakan radio, juga mendukung penghubungan komputer dengan kabel utp), atau berbeda arsitektur jaringan. TL-WR840N adalah gabungan kabel / jaringan nirkabel yang dirancang khusus untuk kebutuhan jaringan usaha kecil dan usaha rumahan. TL-WR840N menciptakan kinerja nirkabel yang luar biasa dan canggih, sehingga ideal untuk *streaming* video HD, VoIP dan *game* online. Dan juga, Tombol *setup* (WPS) *Wi-Fi Protected* pada bagian luar ramping dan modis memastikan enkripsi WPA2, mencegah jaringan dari intrusi luar [8]. Bentuk fisik TP-LINK TL-WR840N bisa dilihat pada Gambar 2.10.

2.8 Modul Relay

Gambar 2.11 Modul Relay
Sumber : https://encrypted-tbn0.gstatic.com/images?q=tbn%3AANd9GcSI8NwvgQtLyp_ePisL92Fnv-lYFsNo_CVnfDvQIbOgopIF8Kzy

Relay adalah saklar (*switch*) yang dioperasikan secara listrik dan merupakan komponen *electromechanical* (elektromekanikal) yang terdiri dari 2 bagian utama yakni elektromagnet (*coil*) dan mekanikal (seperangkat kontak saklar/*switch*). Relay menggunakan prinsip elektromagnetik untuk menggerakkan kontak saklar sehingga dengan arus listrik yang kecil (*low power*) dapat menghantarkan listrik yang bertegangan lebih tinggi. Modul relay ini dapat digunakan sebagai switch untuk menjalankan berbagai peralatan elektronik. Misalnya lampu listrik, motor listrik, dan berbagai peralatan elektronik lainnya. Kendali ON / OFF switch (relay), sepenuhnya ditentukan oleh nilai *output* sensor, yang setelah diproses mikrokontroler akan menghasilkan perintah kepada relay untuk melakukan fungsi on / off [9]. Bentuk fisik modul relay bisa dilihat pada Gambar 2.11.

2.9 *Driver* Motor

Driver motor yang digunakan adalah Pompa Air *High Pressure* DC 12V Mini. Pompa ini digunakan sebagai indikator dari sensor *smoke detector* dan *heat detector*.Bentuk fisik Pompa Air *High Pressure* DC 12V Mini dapat dilihat pada Gambar 2.12.

Spesifikasi Pompa Air High Pressure DC 12V Mini:

- Dimensi 170 x 100 x 67 mm
- Max pressure 100 Psi
- Daya / Arus : 60-65 Watt (disarankan pake adaptor minimal 12 Volt 5A)

- Ukuran selang in/out nya 3/8 inch atau 5/16 inch
- Voltage: DC 12V (mak 15 Volt)
- Max Flow: 4.0L/Min - Disarankan pake 5/16
-

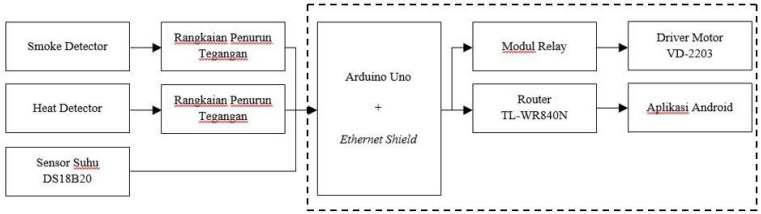
Gambar 2.12 Pompa Air High Pressure DC 12V Mini

Sumber : https://encrypted-tbn0.gstatic.com/images?q=tbn%3AANd9GcTu-HGRa_NxYnFRIl5rXzoL2raxU_w3VLeKAIQfgq8-FK13JTgt

-----Halaman ini sengaja dikosongkan-----

BAB III
PERANCANGAN *HARDWARE* DAN *SOFTWARE*

Pada bab ini dibahas mengenai blok fungsional sistem, perancangan perangkat keras (*hardware*) dan perangkat lunak (*software*). Hal tersebut guna mewujudkan Proyek Akhir ini.



Gambar 3.1 Blok Fungsional Sistem

Dari Gambar 3.1 dapat dijelaskan bahwa sistem alarm kebakaran dipicu dari pendeteksi *smoke detector, heat detector*, dan sensor suhu DS18B20. Ketiga *device* ini akan mengirim data ke Arduino Uno. Ada 3 keadaan dari tiap sensor tersebut, yaitu kondisi menyala, mati, dan mendeteksi. Sedangkan sensor suhu DS18B20 akan mendeteksi suhu panas yang ada di dalam ruangan dan akan mengirim data ke Arduino Uno.

Arduino Uno akan membaca kondisi apa yang terjadi oleh *smoke detector* dan *heat detector* yang telah melewati rangkaian penurun tegangan. Rangkaian penurun tegangan digunakan untuk menurunkan tegangan dari *smoke detector* dan *heat detector* yang awalnya bertegangan 24V menjadi 5V agar data dari *smoke detector* dan *heat detector* terbaca pada Arduino Uno. Arduino Uno juga akan membaca berapa suhu yang ada di dalam ruangan melalui sensor suhu DS18B20.

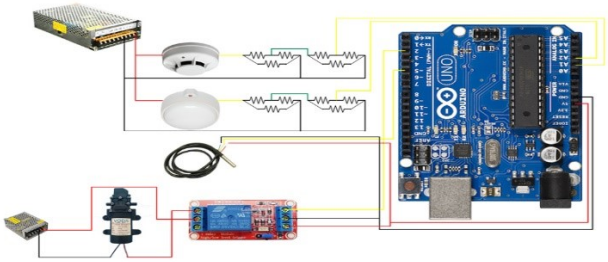
Selanjutnya Arduino Uno akan memberikan perintah kepada *ethernet shield* sesuai dengan kondisi yang telah terbaca di tiap sensor. Perintah tersebut akan dilaksanakan oleh *modul* relay untuk mengontrol *driver* motor agar menyala ketika *smoke detector* dan *heat detector* dalam keadaan mendeteksi. Kondisi dari *smoke detector*, *heat detector*, dan suhu yang dibaca oleh sensor suhu DS18B20 akan ditampilkan ke dalam aplikasi.

3.1 Perancangan Perangkat Keras (*Hardware*)

Perancangan Hardware Pada Proyek Akhir ini terdapat beberapa perancangan elektronik yang akan digunakan agar alat dapat berjalan dengan baik. Berikut merupakan beberapa perancangan untuk alat *monitoring fire alarm* yaitu.

1. Perancangan komunikasi antar sensor menggunakan Arduino uno
2. Perancangan komunikasi Arduino uno menggunakan *ethernet Shield*
3. Perancangan *driver* motor

3.1.1 Rancangan Komunikasi Antar Sensor Menggunakan Arduino Uno



Gambar 3.2 *Wiring* Diagram Antara Sensor Dengan Arduino Uno

Pada rancangan ini yaitu menyambungkan antara sensor dengan Arduino uno. Adapun *wiring* diagram antara sensor dengan Arduino uno dapat dilihat pada Gambar 3.2. Pada Tabel 3.1 menunjukkan konfigurasi *port* mikrokontroler.

Tabel 3.1 Konfigurasi *Port* Mikrokontroler

No.	Pin Arduino	Keterangan
1.	5V	VCC Sensor Suhu DS18B20, + *Modul* Relay
2.	GND	GND Sensor Suhu DS18B20, - *Modul* Relay

No.	Pin Arduino	Keterangan
3.	A1	*Smoke Detector*
4.	A2	*Heat Detector*
5.	D2	*In Modul* Relay
6.	D5	Sensor Suhu DS18B20

3.1.2 Perancangan Komunikasi Arduino Uno Menggunakan *Ethernet Shield*

Pada rancangan kali ini yaitu dilakukan pada modul *ethernet shield* yang telah dipasang pada bagian atas *board* Arduino Uno, kemudian *ethernet shield* dihubungkan ke router menggunakan kabel *Lan*.

3.1.3 Perancangan *Driver* Motor

Driver motor dihidupkan oleh *supply* 12V. Motor yang digunakan adalah *water pump* atau pompa air dengan tegangan kerja 12V. V+ pada *power supply* disambungkan pada *com* yang ada di modul relay, sedangkan V- pada *power supply* disambung pada (-) motor. (+) pada motor disambungkan ke NO pada modul relay. Pada modul relay terdapat pin (+) yang disambung ke pin 5V Arduino Uno, pin (-) disambung ke GND Arduino Uno, dan *in* disambung pada pin D2 Arduino Uno yang berfungsi untuk *input* yang akan diatur oleh Arduino Uno.

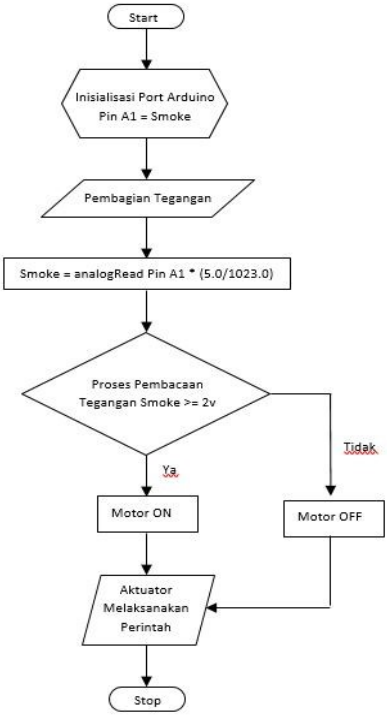
3.2 Perancangan Perangkat Lunak (*Software*)

Untuk perancangan perangkat lunak (*software*) yang digunakan sebagai pemograman yaitu dengan menggunakan *Arduino* IDE yang dihubungkan ke mikrokontroler berupa Arduino uno. Untuk pembahasan perangkat lunak meliputi:

1. Perancangan Program Pembacaan Rangkaian Penurun Tegangan
2. Perancangan Program Pembacaan Sensor Suhu DS18B20
3. Perancangan Pembuatan Aplikasi Android Sebagai *Interface* Hasil Monitoring
4. Perancangan Komunikasi *Wifi*

3.2.1 Perancangan Program Pembacaan Rangkaian Penurun Tegangan

Rangkaian penurun tegangan yang digunakan pada proyek akhir ini adalah rangkaian pembagi tegangan yang *output*nya akan dibaca oleh Arduino uno. Rangkaian ini menggunakan resistor yang disusun seri paralel untuk menghasilkan output tegangan yang berbeda.



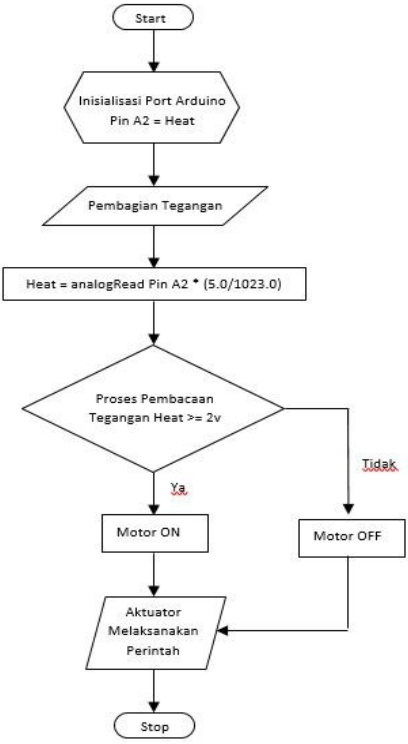
Gambar 3.3 Pembacaan Rangkaian Penurun Tegangan *Smoke Detector*

Penjelasan *flowchart* berdasarkan Gambar 3.3 adalah sebagai berikut:

1. *Start* adalah ketika program dimulai
2. Inisialisasi pin A1 digunakan untuk pembacaan tegangan *output smoke detector*
3. Sensor tegangan mendapat *input* dari *smoke detector*. Kemudian dilakukan pembagian tegangan oleh rangkaian

resistor dan menghasilkan *output* tegangan sesuai dengan keadaan dari *smoke detector*

4. Data dibaca oleh Arduino Uno dengan *range* 0-5V.
5. Dengan batas tegangan yang sudah ditentukan, program akan melaksanakan perintah sesuai dengan *output* dari rangkaian pembagi tegangan.
6. Jika tegangan lebih dari 2V maka motor menyala, dan jika tegangan kurang dari 2V maka motor tidak menyala
7. Aktuator akan bekerja dengan perintah yang diberikan.

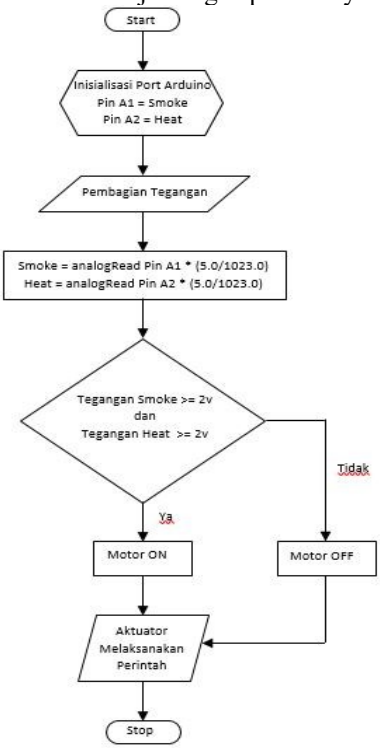


Gambar 3.4 Pembacaan Rangkaian Penurun Tegangan *Heat Detector*

Penjelasan *flowchart* berdasarkan Gambar 3.4 adalah sebagai berikut:

1. *Start* adalah ketika program dimulai

2. Inisialisasi pin A2 digunakan untuk pembacaan tegangan *output heat detector*
3. Sensor tegangan mendapat *input* dari *heat detector*. Kemudian dilakukan pembagian tegangan oleh rangkaian resistor dan menghasilkan *output* tegangan sesuai dengan keadaan dari *heat detector*
4. Data dibaca oleh Arduino Uno dengan *range* 0-5V.
5. Dengan batas tegangan yang sudah ditentukan, program akan melaksanakan perintah sesuai dengan *output* dari rangkaian pembagi tegangan.
6. Jika tegangan lebih dari 2V maka motor menyala, dan jika tegangan kurang dari 2V maka motor tidak menyala
7. Aktuator akan bekerja dengan perintah yang diberikan.



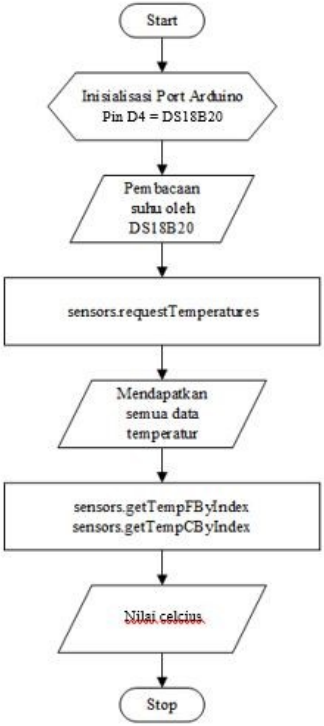
Gambar 3.5 Pembacaan Rangkaian Penurun Tegangan *Smoke Detector* dan *Heat Detector*

Penjelasan *flowchart* berdasarkan Gambar 3.5 adalah sebagai berikut:

1. *Start* adalah ketika program dimulai
2. Inisialisasi pin A1 digunakan untuk pembacaan tegangan *output smoke detector,* pin A2 digunakan untuk pembacaan tegangan *output heat detector*
3. Sensor tegangan mendapat *input* dari *smoke detector* dan *heat detector*. Kemudian dilakukan pembagian tegangan oleh rangkaian resistor dan menghasilkan *output* tegangan sesuai dengan keadaan dari *smoke detector* dan *heat detector*
4. Data dibaca oleh Arduino Uno dengan *range* 0-5V.
5. Dengan batas tegangan yang sudah ditentukan, program akan melaksanakan perintah sesuai dengan *output* dari rangkaian pembagi tegangan.
6. Jika tegangan lebih dari 2V maka motor menyala, dan jika tegangan kurang dari 2V maka motor tidak menyala
7. Aktuator akan bekerja dengan perintah yang diberikan.

3.2.2 Perancangan Program Pembacaan Sensor Suhu DS18B20

Pembacaan suhu dalam proyek akhir ini menggunakan DS18B20. Pembacaan suhu menggunakan satuan *Celcius*. Pembacaan suhu digunakan untuk mengetahui kondisi suhu dalam ruangan.

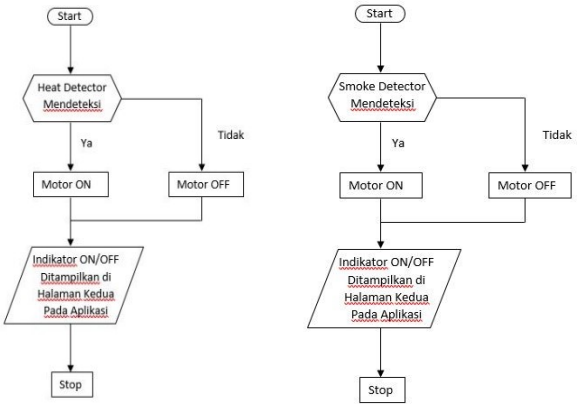


Gambar 3.6 *Flowchart* Pembacaan Sensor Suhu DS18B20

Penjelasan *flowchart* berdasarkan Gambar 3.6 adalah sebagai berikut:

1. *Start* adalah ketika program dimulai
2. Inisialisasi pin D5 digunakan untuk *input* dari DS18B20
3. Sensor DS18B20 membaca suhu ruangan sebagai *input* dan mengirim data ke Arduino uno sebagai *output*
4. Arduino uno membaca data *output* dari DS18B20
5. Setelah data *output* DS18B20 didapatkan, akan dikonversi ke satuan *Celcius*
6. Pembacaan suhu *Celcius* sebagai *output*

3.2.3 Perancangan Program Aplikasi Android Sebagai Interface Hasil Monitoring



Gambar 3.7 *Flowchart* Pembacaan Sensor ke aplikasi

Penjelasan *flowchart* pada Gambar 3.7 menjelaskan alur pembacaan sensor ke aplikasi adalah sebagai berikut :

1. *Start* adalah ketika program dimulai
2. Dengan batas tegangan yang sudah ditentukan, program, *smoke detector* dan atau *heat* detector akan melaksanakan perintah sesuai dengan *output* dari Arduino Uno
3. Jika *smoke detector* dan atau *heat* detector mendeteksi maka motor menyala, dan jika *smoke detector* dan atau *heat* detector tidak mendeteksi maka motor tidak menyala
4. Dari Arduino Uno mengirim data ke *ethernet shield* agar dapat ditampilkan di halaman kedua pada aplikasi

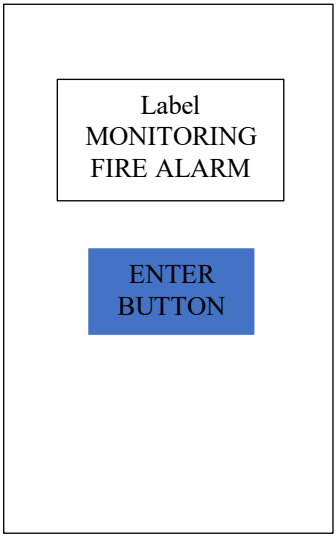
Pada perancangan interface pada android ini yaitu menggunakan MIT App Inventor 2. Dalam pembuatan interface android ini terdapat 3 langkah utama yaitu :

1. Membuka *http://appinventor.mit.edu/* pada *browser*, terlebih dahulu *login* menggunakan akun Gmail.
2. Membuat desain aplikasi Android dengan menggunakan komponen – komponen yang ada dalam App Inventor pada *tab designer*.
3. Membuat program pada *tab block editor*, dimana pada *tab* ini yaitu mengatur komponen – komponen yang sudah dipilih dari *tab designer*.

Untuk lebih jelasnya tentang MIT App Inventor 2 bisa dilihat pada Gambar 3.8.

Gambar 3.8 Tampilan *Tab Designer*

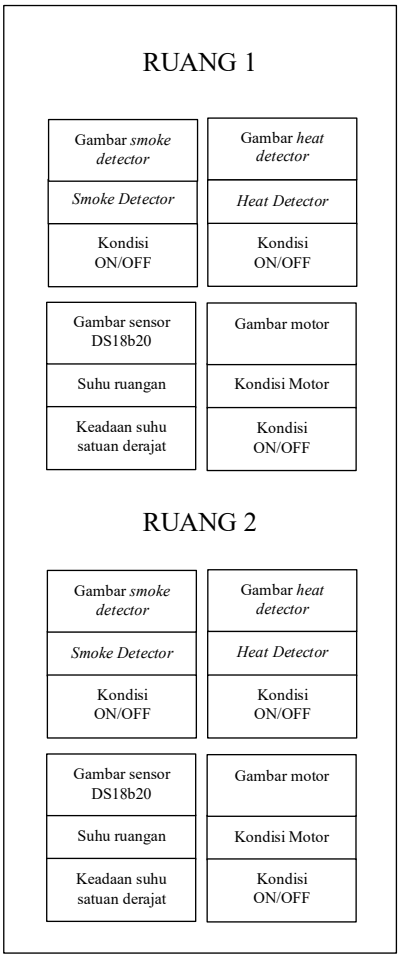
Pada Gambar 3.8 merupakan tampilan *Tab Designer* pada MIT APP Inventor 2, yang merupakan tampilan awal dalam pembuatan aplikasi pada Android nantinya. Pada tab ini yaitu digunakan untuk membuat desain pada aplikasi, yang nantinya pada tampilan awal seperti pada Gambar 3.9 akan berisi tampilan untuk masuk ke halaman utama.



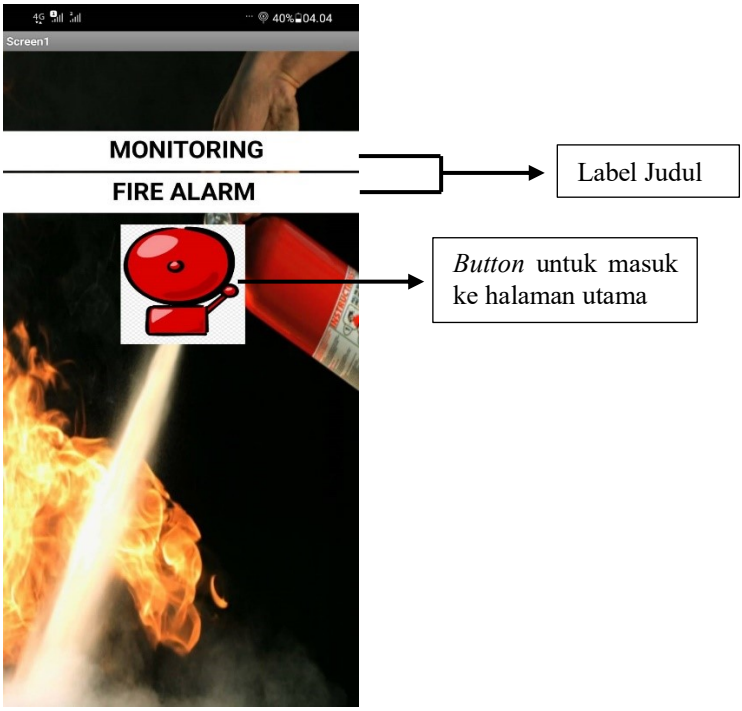
Gambar 3.9 Rancangan Tampilan Awal

Setelah melewati tampilan awal maka akan muncul halaman utama seperti pada Gambar 3.10. Halaman utama pada aplikasi ini merupakan halaman yang nantinya digunakan untuk melihat kondisi pada

smoke detector dan *heat detector,* suhu ruangan dalam satuan derajat *celcius*, dan kondisi motor dalam keadaan menyala/mati.

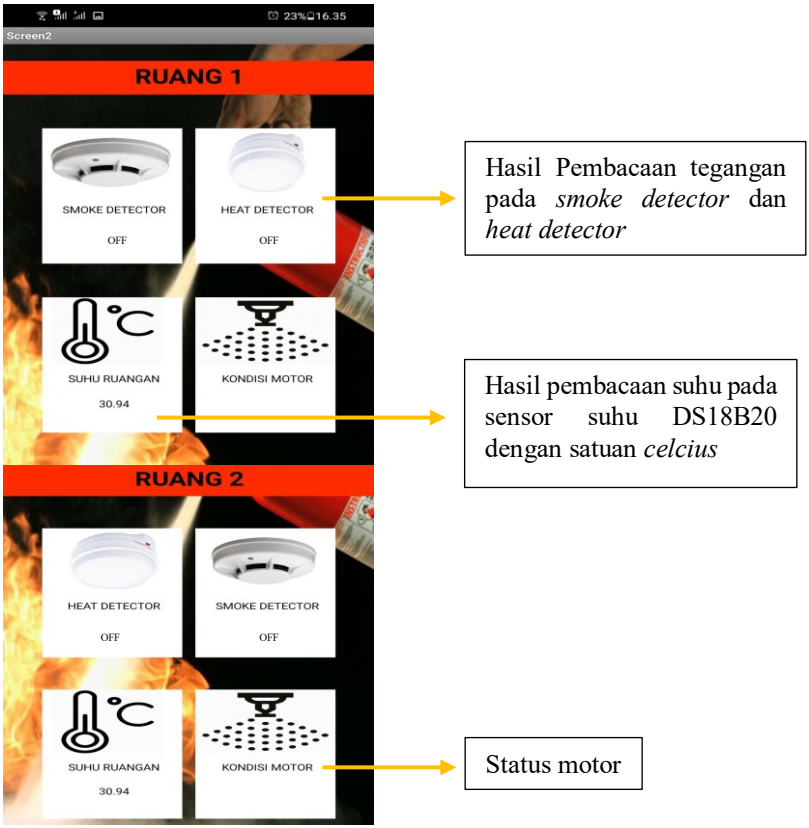


Gambar 3.10 Rancangan Halaman Utama



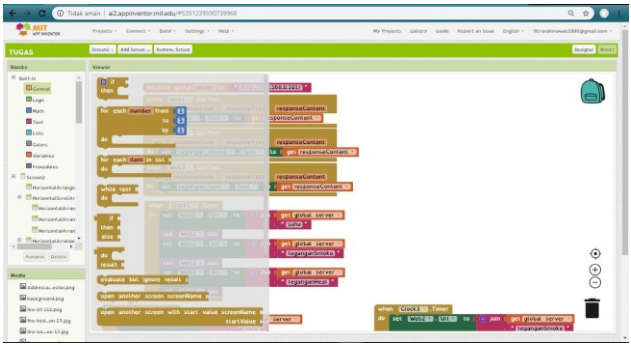
Gambar 3.11 Realisasi Perancangan Tampilan Awal

Pada Gambar 3.11 merupakan realisasi perancangan tampilan awal pada *android* yang sudah dibuat dari MIT App Inventor. Pada tampilan ini menggunakan komponen seperti label dan button yang sudah tersedia di dalam MIT App Inventor. Terdapat label *judul* yaitu " MONITORING FIRE ALARM", ada juga sebuah tombol yang berfungsi untuk masuk ke halaman utama atau halaman *monitoring*. Untuk lebih jelasnya tentang realisasi perancangan halaman utama atau halaman monitoring bisa dilihat pada Gambar 3.12.



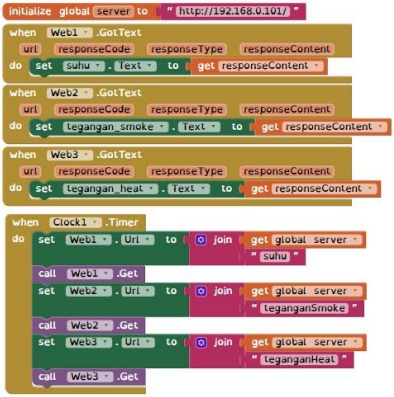
Gambar 3.12 Realisasi Perancangan Halaman Utama

Pada Gambar 3.12 merupakan realisasi perancangan halaman utama atau halaman monitoring. Pada halaman ini terdapat hasil pembacaan tegangan *smoke detector* dan *heat detector,* suhu ruangan dalam satuan derajat *celcius*, dan kondisi motor dalam keadaan menyala/mati. Tampilan tersebut berfungsi untuk menunjukkan kondisi tegangan dan suhu serta keadaan motor. Jika tegangan *smoke detector* dan *heat detector* sekitar 0.0 V - 1,98 V maka kondisi motor dalam keadaan mati (tidak akan muncul tulisan ON), dan sebaliknya, jika tegangan *smoke detector* dan *heat detector* sekitar 2.01 V – 5.00 V maka kondisi motor dalam keadaan nyala (akan muncul tuisan ON).



Gambar 3.13 Tampilan Tab *Block*

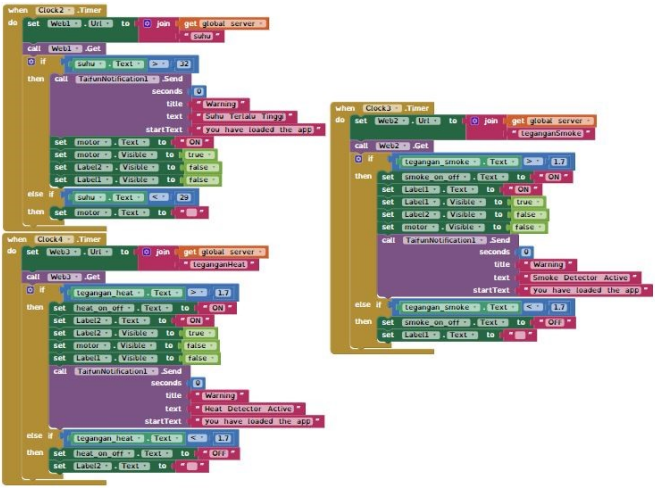
Pada Gambar 3.13 merupakan tampilan pada Tab *Block*. Pada tab ini berfungsi untuk membuat blok program setelah membuat tampilan *interface* sebelumnya seperti pada Gambar 3.11 dan Gambar 3.12. Pada tab ini berisi blok program dimana cara memakainya dengan cara *drag* dan *drop* ke bagian *viewer*.



Gambar 3.14 *Block* Menampilkan Data Dari Arduino Uno ke Android

Pada Gambar 3.14 merupakan *block* untuk menampilkan data hasil *monitoring* dari Arduino uno ke android. Terdapat *block initialize global server to* http://192.168.0.101/ yang merupakan *block* untuk membaca alamat IP milik *ethernet shield* yaitu 192.168.0.101 yang sudah

diprogram sebelumnya. Tak hanya menampilkan nilai hasil monitoring saja, namun pada aplikasi ini juga dirancang untuk menampilkan notifikasi jika *smoke detector* dan *heat detector* terdeteksi. Pada Gambar 3.15 merupakan block untuk mengirimkan notifikasi pada android.



Gambar 3.15 *Block* Mengirimkan Notifikasi dan Pemberian Keterangan

Pada Gambar 3.15 adalah perancangan *block* untuk mengirimkan notifikasi serta pemberian keterangan. Dapat dilihat pada Gambar 3.15 ketika nilai tegangan lebih dari 2 maka motor akan menyala dan muncul tulisan "ON" dan aplikasi ini akan mengirimkan notifikasi yaitu "*Warning Smoke Detector Active*". Akan tetapi ketika nilai tegangan kurang dari 2 maka tidak akan muncul tulisan apa-apa dan tidak mengirimkan notifikasi.

Terdapat beberapa kemungkinan ketika sensor mendeteksi, antara lain :

Tabel 3.2 Kemungkinan Sensor Mendeteksi

No.	*Smoke Detector* Ruang 1	*Smoke Detector* Ruang 2	*Heat Detector* Ruang 1	*Heat Detector* Ruang 2	Motor Ruang 1	Motor Ruang 2
1.	OFF	OFF	OFF	OFF	OFF	OFF
2.	OFF	OFF	OFF	ON	OFF	ON
3.	OFF	OFF	ON	ON	OFF	ON
4.	OFF	ON	ON	ON	ON	ON
5.	OFF	ON	OFF	OFF	ON	OFF
6.	OFF	ON	ON	OFF	ON	ON
7.	OFF	OFF	ON	OFF	OFF	ON
8.	OFF	ON	OFF	ON	ON	ON
9.	ON	OFF	OFF	OFF	ON	OFF
10.	ON	OFF	OFF	ON	ON	ON
11.	ON	OFF	ON	ON	ON	ON
12.	ON	ON	ON	ON	ON	ON
13.	ON	ON	OFF	OFF	ON	OFF
14.	ON	ON	ON	OFF	ON	ON
15.	ON	OFF	ON	OFF	ON	ON
16.	ON	ON	OFF	ON	ON	ON

3.2.4. Perancangan Komunikasi Wifi

Agar keseluruhan fungsi dalam alat ini dapat berjalan dengan baik mulai dari *hardware* hingga *software*, maka diperlukannya komunikasi wifi dalam perancangannya. Komunikasi wifi disini berfungsi untuk menjalankan sistem *monitoring* dapat dijalankan secara *wireless*. Maka dari itu dibutuhkan sebuah *access point*. *Access point* yang digunakan dalam proyek akhir ini adalah TP-LINK TL-WR840N. *Access point* inilah yang nantinya akan tersambung langsung dengan *ethernet shield* dengan slot *Lan* pada *access point* yang menggunakan kabel RJ45. Dengan begitu proses *monitoring* dengan komunikasi wifi dapat dilakukan. Berikut adalah perancangan komunikasi wifi dalam proyek akhir ini :

1. Menentukan alamat IP pada *Ethernet shield*.

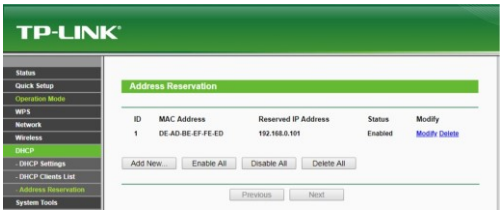
2. Mengakses *website* TP-LINK guna mendaftarkan alamat IP milik *Ethernet shield*.
3. Menyesuaikan alamat IP pada *access point Local Area* di *connection properties*.

```
byte mac[] = {0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED};
IPAddress ip(192,168,0,101);
EthernetServer server(80);
int motor = 2; // motor connect to Arduino digital 2

void setup(void)
{
  pinMode(motor,OUTPUT);
  // Start serial communication for debugging purposes
  Serial.begin(9600);
  // Start up the library
  sensors.begin();
    Ethernet.begin(mac, ip);
  server.begin();
    Serial.print("server is at ");
  Serial.println(Ethernet.localIP());
}
```

Gambar 3.16 Alamat IP pada *Ethernet Shield*

Pada Gambar 3.16 merupakan alamat IP milik *Ethernet shield* yang ditentukan melalui *software* Arduino IDE. IP inilah yang nantinya akan didaftarkan pada *website* TP-LINK supaya bisa dijadikan *access point*.



Gambar 3.17 Akses *Website* TP-LINK

Pada Gambar 3.17 merupakan alamat IP milik *Ethernet shield* yang ditentukan melalui *software* Arduino IDE. IP inilah yang nantinya akan didaftarkan pada *website* TP-LINK supaya bisa dijadikan *access point.*

-----Halaman ini sengaja dikosongkan-----

BAB IV
PENGUJIAN DAN ANALISA

4.1 Pengujian *Smoke Detector*

Pengujian dilakukan untuk mengetahui respon dari detektor asap, dengan cara menyemprotkan sumber asap ke dalam ruang sensor. Sumber asap berasal dari asap buatan atau *smoke tester*. Penyemprotan dilakukan dengan perlahan, sedikit demi sedikit, sebab kalau terlalu banyak akan menimbulkan akumulasi asap di dalam ruang detektor, yang mengakibatkan *detector* terus merespon adanya asap. Apabila pada badan detektor tampak ada Lampu merah yang menyala, maka kondisi detektor sedang mendeteksi adanya asap seperti yang ditampilkan pada Gambar 4.1.

Gambar 4.1 *Smoke Detector* dalam Keadaan Mendeteksi

Tabel 4.1 Pengujian *Smoke Detector* Menggunakan *Smoke Tester*

No.	Jarak *Smoke Tester* (cm)	*Smoke Detector*
1.	20	Mendeteksi
2.	40	Mendeteksi
3.	60	Mendeteksi
4.	80	Mendeteksi
5.	100	Mendeteksi
6.	120	Mendeteksi
7.	140	Mendeteksi

No.	Jarak *Smoke Tester* (cm)	*Smoke Detector*
8.	160	Mendeteksi
9.	180	Mendeteksi
10.	200	Tidak mendeteksi

Tabel 4.2 Hasil Pengujian *Smoke Detector* Ruang 1

No.	Kondisi	Tegangan
1.	Mati	0.00 V
2.	Menyala	0.00 V
3.	Mendeteksi	2.71 V

Seperti Tabel 4.2 menunjukkan hasil dari pengujian *smoke detector* ruang 1 bahwa ketika kondisi mendeteksi maka tegangan sebesar lebih dari 2 volt. Begitu juga dengan Tabel 4.3 menunjukkan hasil dari pengujian *smoke detector* ruang 2 bahwa ketika kondisi mendeteksi maka tegangan sebesar 2 volt.

Tabel 4.3 Hasil Pengujian *Smoke detector* Ruang 2

No.	Kondisi	Tegangan
1.	Mati	0.00 V
2.	Menyala	0.00 V
3.	Mendeteksi	2.00 V

Gambar 4.2 *Interface* aplikasi *Smoke Detector* Ruang 1

Pada Gambar 4.2 menunjukkan ketika *smoke detector* mendeteksi maka kondisi motor di ruang 1 menyala.

Gambar 4.3 *Interface* aplikasi *Smoke Detector* Ruang 2

Pada Gambar 4.3 menunjukkan ketika *smoke detector* mendeteksi maka kondisi motor di ruang 2 menyala.

Gambar 4.4 *Interface* aplikasi *Smoke Detector* Ruang 1 dan Ruang 2

Pada Gambar 4.4 menunjukkan ketika *smoke detector* di ruang 1 dan 2 mendeteksi maka kondisi motor di ruang 1 dan 2 menyala.

4.2 Pengujian *Heat Detector*

Untuk mengetahui kinerja detektor api dilakukan dengan cara mendekatkan sumber panas dari *hair dryer* pada posisi detektor, seperti yang ditampilkan pada Gambar 4.5.

Gambar 4.5 Pengujian *Heat Detector*

Tabel 4.4 Hasil Pengujian *Heat Detector* Ruang 1

No.	Kondisi	Tegangan
1.	Mati	0.00 V
2.	Menyala	1.15 V
3.	Mendeteksi	4.57 V

Seperti Tabel 4.4 menunjukkan hasil dari pengujian *heat detector* ruang 1 bahwa ketika kondisi mendeteksi maka tegangan sebesar lebih dari 2 volt. Begitu juga dengan Tabel 4.5 menunjukkan hasil dari pengujian *heat detector* ruang 2 bahwa ketika kondisi mendeteksi maka tegangan sebesar 2 volt.

Tabel 4.5 Hasil Pengujian *Heat Detector* Ruang 2

No.	Kondisi	Tegangan
1.	Mati	0.00 V
2.	Menyala	0.00 V
3.	Mendeteksi	2.00 V

Gambar 4.6 *Interface* aplikasi *Heat Detector* Ruang 1

Pada Gambar 4.6 menunjukkan ketika *heat detector* mendeteksi maka kondisi motor di ruang 1 menyala.

Gambar 4.7 *Interface* aplikasi *Heat Detector* Ruang 2

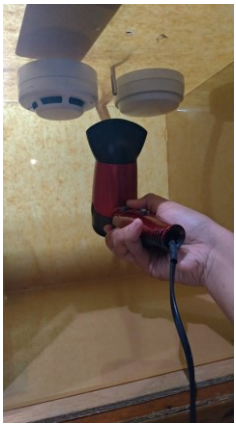
Pada Gambar 4.7 menunjukkan ketika *heat detector* mendeteksi maka kondisi motor di ruang 2 menyala.

Gambar 4.8 *Interface* aplikasi *Heat Detector* Ruang 1 dan Ruang 2

Pada Gambar 4.8 menunjukkan ketika *heat detector* ruang 1 dan ruang 2 mendeteksi maka kondisi motor di ruang 1 dan ruang 2 menyala.

4.3 Pengujian Sensor Suhu DS18B20

Pengujian sensor suhu DS18B20 pada Gambar 4.9 dilakukan dengan memberikan udara panas dari hairdryer untuk meningkatkan suhu. Dari perlakuan tersebut diperoleh keluaran suhu pada mikrokontroler Arduino dan dibandingkan nilainya dengan nilai suhu pada *Richmeter* versi 102.



Gambar 4.9 Pengujian Sensor Suhu DS18B20

Pengambilan data dilakukan dengan menggunakan Power supply 5 V dengan serta dilakukan perbandingan antara suhu yang terbaca oleh sensor dengan alat ukur *Richmeter* Versi 102. Selanjutnya dihitung pula nilai % error dengan rumus pada persamaan berikut:

$$\%\ Error = \left|\frac{(nilai\ alat\ ukur - nilai\ sensor)}{nilai\ alat\ ukur}\right| \text{ x } 100\%$$

sebagai contoh:

$$\%\ Error = \left|\frac{(30-29)}{30}\right| \text{ x } 100\% = 3\%$$

Dari hasil perhitungan, dapat dianalisa bahwa nilai suhu pada alat ukur adalah 30°C, namun suhu yang terbaca oleh sensor adalah 29°C, sehingga didapatkan persentase error sebesar 3 %. Selanjutnya, dengan cara dan rumus yang sama, dihitung persentase error untuk suhu sampai dengan 100°C. Pada Tabel 4.6 berikut merupakan data perbandingan suhu yang terbaca oleh sensor dengan nilai suhu pada alat ukur *Richmeter*.

Tabel 4-6 Hasil Pengujian Sensor Suhu DS18B20

No.	Suhu yang terbaca pada *Richmeter* (°C)	Suhu yang terbaca oleh sensor (°C)	*% Error*
1.	28	28	0%
2.	30	29	3%
3.	35	33	5,7%
4.	40	36	10%
5.	45	41	8,8%
6.	50	48	4%
7.	55	51	7,2%
8.	60	56	6,6%
9.	65	60	7,6%
10.	70	67	4,2%
11.	75	71	5,3%
12.	80	76	5%
13.	85	82	3,5%
14.	90	86	4,4%
15.	95	91	4,2%
16.	100	93	7%

Pengujian sensor suhu DS18B20 dilakukan dengan memberikan udara panas dari *hairdryer* untuk meningkatkan suhu. Dari perlakuan tersebut diperoleh hasil keluaran suhu pada mikrokontroler Arduino Uno tiap 5 detik pada Tabel 4.7.

Tabel 4.7 Sensor Suhu DS18B20 di Ukur Tiap 5 Detik

No.	Waktu	Suhu (°C)
1.	15.22.09	31.00
2.	15.22.14	31.69
3.	15.22.19	33.25
4.	15.22.24	34.31
5.	15.22.29	36.13
6.	15.22.34	37.21
7.	15.22.39	38.45
8.	15.22.44	39.56
9.	15.22.49	41.05
10.	15.22.54	42.33

Gambar 4.10 *Interface* aplikasi Suhu Ruangan

Pada Gambar 4.10 menunjukkan ketika sensor suhu membaca suhu pada ruang 1 dan ruang 2.

4.4 Pengujian Sensor Suhu DS18B20 Dan *Heat Detector*

Pengujian sensor suhu DS18B20 dan *heat detector* dilakukan dengan cara memicu *heat detector* menggunakan *hairdryer*. Dapat dilihat pada Gambar 4.11.

Gambar 4.11 Pengujian *Heat Detector dengan DS18B20*

Tabel 4.8 Pengujian Tegangan *Heat detector* dan Sensor Suhu DS18B20

No.	Tegangan (V)	Suhu (°C)
1.	1.17	32.31
2.	1.18	33.00
3.	5.00	36.63
4.	5.00	37.94
5.	5.00	38.54

4.5 Pengujian Komunikasi *Ethernet Shield*

Pengujian ini dilakukan pada modul *ethernet shield* yang sudah dipasang pada bagian atas Arduino uno serta *ethernet shield* sudah terhubungan dengan *router* atau *access point* yaitu TP-LINK TL-WR840N. Caranya yaitu dilakukan tes ping terhadap IP Address ethernet shield.

4.6 Pengujian Pengiriman Data Tanpa Penghalang

Pengujian ini dilakukan pada ruangan tanpa penghalang dengan jarak 10 – 50 meter dengan cara mengirimkan nilai tegangan dan

temperatur melalui serial monitor arduino. Pengujian ini dilakukan di area lingkar perpus ITS Setelah dilakukan pengujian dalam pengiriman data didapatkan Tabel 4.9 yang merupakan rangkuman dari hasil pengujian pada ruangan tanpa penghalang.

Tabel 4.9 Pengiriman Data

No.	Jarak (Meter)	Jumlah Percobaan	Keberhasil (%)
1.	10	5	100
2.	20	5	100
3.	30	5	80
4.	40	5	80
5.	50	5	40

Semakin jauh jaraknya maka akan mengakibatkan loss data, maka dari itu jangkauan pengiriman data yang disarankan yaitu 10 – 45 meter, dikarenakan pada jarak 50 meter mengalami loss data yang cukup banyak sehingga persentase keberhasilan dalam pengiriman data sangat kecil. Keberhasilan data merupakan nilai yang diterima oleh serial monitor pada master, pengiriman data dilakukan setiap 3 detik. Pada Tabel 4.8 terdapat keberhasilan yang dimana adalah persentase keberhasilan data keseluruhan, keberhasilan didapatkan dari rumus berikut :

$$\left|\frac{jumlah\ data\ yang\ masuk}{jumlah\ seluruh\ data}\right| \text{ x } 100\%$$

4.7 Pengujian Koneksi Tanpa Penghalang

Pengujian ini dilakukan dengan jarak 10 meter – 80 meter tanpa adanya penghalang, pengambilan data ini dilakukan di lingkaran perpustakaan ITS. Setelah dilakukan pengujian koneksi didapatkan Tabel 4.9 yang merupakan rangkuman dari hasil pengambilan data.

hasil pengujian *ethernet shield* dengan melakukan ping menggunakan *command prompt* bahwa *ethernet shield* siap digunakan serta untuk melihat loss data bisa dilihat pada 0% loss di Tabel 4.10, yang merupakan indikator persentasi loss data.

Tabel 4.10 Hasil Pengujian Koneksi *Ethernet Shield* Tanpa Penghalang

No.	Jarak (Meter)	Koneksi Loss (%)
1.	10	0
2.	20	0
3.	30	0
4.	40	0

No.	Jarak (Meter)	Koneksi Loss (%)
5.	50	0
6.	60	0
7.	70	0
8.	75	50
9.	80	50

4.8 Tampilan Pengujian Sistem Secara Keseluruhan

Berikut tampilan pada aplikasi dengan beberapa kemungkinan sensor mendeteksi, antara lain :

Gambar 4.12 *Interface* aplikasi Ketika kondisi *Heat Detector* Dan *Smoke Detector* Ruang 1 Serta *Heat Detector* Dan *Smoke Detector* Ruang 2 Tidak Mendeteksi

Pada Gambar 4.12 menunjukkan ketika kondisi *heat detector* dan *smoke detector* ruang 1 serta *heat detector* dan *smoke detector* ruang 2 tidak mendeteksi maka motor pada kedua ruangan tidak menyala.

Gambar 4.13 *Interface* aplikasi Ketika kondisi *Smoke Detector* Ruang 1 Mendeteksi

Pada Gambar 4.13 menunjukkan ketika kondisi *smoke detector* ruang 1 mendeteksi maka motor pada ruang 1 menyala.

Gambar 4.14 *Interface* aplikasi Ketika kondisi *Smoke Detector* Ruang 2 Mendeteksi

Pada Gambar 4.14 menunjukkan ketika kondisi *smoke detector* ruang 2 mendeteksi maka motor pada ruang 2 menyala.

Gambar 4.15 *Interface* aplikasi Ketika kondisi *Heat Detector* Ruang 1 Mendeteksi

Pada Gambar 4.15 menunjukkan ketika kondisi *heat detector* ruang 1 mendeteksi maka motor pada ruang 1 menyala.

Gambar 4.16 *Interface* aplikasi Ketika kondisi *Heat Detector* Ruang 2 Mendeteksi

Pada Gambar 4.16 menunjukkan ketika kondisi *heat detector* ruang 2 mendeteksi maka motor pada ruang 2 menyala.

Gambar 4.17 *Interface* aplikasi Ketika kondisi *Smoke Detector* Dan *Heat Detector* Ruang 1 Mendeteksi

Pada Gambar 4.17 menunjukkan ketika kondisi *smoke detector* dan *heat detector* ruang 1 mendeteksi maka motor pada ruang 1 menyala.

Gambar 4.18 *Interface* aplikasi Ketika kondisi *Smoke Detector* Dan *Heat Detector* Ruang 2 Mendeteksi

Pada Gambar 4.18 menunjukkan ketika kondisi *smoke detector* dan *heat detector* ruang 2 mendeteksi maka motor pada ruang 2 menyala.

Gambar 4.19 *Interface* aplikasi Ketika kondisi *Smoke Detector* Ruang 1 Dan *Smoke Detector* Ruang 2 Mendeteksi

Pada Gambar 4.19 menunjukkan ketika kondisi *smoke detector* ruang 1 dan *smoke detector* ruang 2 mendeteksi maka motor pada ruang 1 dan ruang 2 menyala.

Gambar 4.20 *Interface* aplikasi Ketika kondisi *Heat Detector* Ruang 1 Dan *Heat Detector* Ruang 2 Mendeteksi

Pada Gambar 4.20 menunjukkan ketika kondisi *heat detector* ruang 1 dan *heat detector* ruang 2 mendeteksi maka motor pada ruang 1 dan ruang 2 menyala.

Gambar 4.21 *Interface* aplikasi Ketika kondisi *Heat Detector* Ruang 1 Serta *Smoke Detector Dan Heat Detector* Ruang 2 Mendeteksi

Pada Gambar 4.21 menunjukkan ketika kondisi *heat detector* ruang 1 serta *smoke detector* dan *heat detector* ruang 2 mendeteksi maka motor pada ruang 1 dan ruang 2 menyala.

Gambar 4.22 *Interface* aplikasi Ketika kondisi *Smoke Detector* Dan *Heat Detector* Ruang 1 Serta *Heat Detector* Ruang 2 Mendeteksi

Pada Gambar 4.22 menunjukkan ketika kondisi *smoke detector* dan *heat detector* ruang 1 serta *heat detector* ruang 2 mendeteksi maka motor pada ruang 1 dan ruang 2 menyala.

Gambar 4.23 *Interface* aplikasi Ketika kondisi *Smoke Detector* Ruang 1 Serta *Smoke Detector* Dan *Heat Detector* Ruang 2 Mendeteksi

Pada Gambar 4.23 menunjukkan ketika kondisi *smoke detector* ruang 1 serta *smoke detector* dan *heat detector* ruang 2 mendeteksi maka motor pada ruang 1 dan ruang 2 menyala.

Gambar 4.24 *Interface* aplikasi Ketika kondisi *Smoke Detector* Dan *Heat Detector* Ruang 1 Serta *Smoke Detector* Ruang 2 Mendeteksi

Pada Gambar 4.24 menunjukkan ketika kondisi *smoke detector* dan *heat detector* ruang 1 serta *smoke detector* ruang 2 mendeteksi maka motor pada ruang 1 dan ruang 2 menyala.

Gambar 4.25 *Interface* aplikasi Ketika kondisi *Heat Detector* Ruang 1 Serta *Smoke Detector* Ruang 2 Mendeteksi

Pada Gambar 4.25 menunjukkan ketika kondisi *heat detector* ruang 1 serta *smoke detector* ruang 2 mendeteksi maka motor pada ruang 1 dan ruang 2 menyala.

Gambar 4.26 *Interface* aplikasi Ketika kondisi *Smoke Detector* Ruang 1 Serta *Heat Detector* Ruang 2 Mendeteksi

Pada Gambar 4.26 menunjukkan ketika kondisi *smoke detector* ruang 1 serta *heat detector* ruang 2 mendeteksi maka motor pada ruang 1 dan ruang 2 menyala.

Gambar 4.27 *Interface* aplikasi Ketika kondisi *Smoke Detector* Dan *Heat detector* Ruang 1 Serta *Smoke Detector* Dan *Heat Detector* Ruang 2 Mendeteksi

Pada Gambar 4.27 menunjukkan ketika kondisi *smoke detector* dan *heat detector* ruang 1 serta *smoke detector* dan *heat detector* ruang 2 mendeteksi maka motor pada ruang 1 dan ruang 2 menyala.

BAB V
PENUTUP

5.1 Kesimpulan

Dari tahapan secara keseluruhan yang sudah dilaksanakan pada penyusunan proyek akhir ini mulai dari studi literatur, perancangan dan pembuatan sampai dengan pengujian alat, maka dapat diperoleh kesimpulan bahwa:

1. Sistem monitoring hanya dapat diakses di *local area*
2. Membutuhkan waktu kurang lebih selama 5 detik untuk menampilkan nilai pada interface android
3. Membutuhkan waktu 2 detik untuk memperbarui tegangan dan suhu
4. Pembacaan pada android terkadang masih error dikarenakan beberapa faktor seperti belum terhubung dengan *access point* dan aplikasi
5. Untuk pembacaan notifikasi pada aplikasi, hanya dapat menampilkan salah satu sensor yang mendeteksi terlebih dahulu pada kondisi ketika terdapat sensor yang menyala hampir bersamaan

5.2 Saran

Untuk lebih menyempurnakan alat ini yaitu diharapkan bisa *monitoring* melalui jaringan internet, sehingga pelaksanaan *monitoring* bisa dilakukan dimana saja. Serta tampilan dari android bisa lebih menarik lagi. Selain itu diperlukan sebuah inovasi untuk menambahkan fitur pada aplikasi android.

-----Halaman ini sengaja dikosongkan-----

DAFTAR PUSTAKA

[1] Om.makpus. **Definisi dan Pengertian Kecelakaan Kerja Serta Latar Belakang Faktor Terjadinya Kecelakaan Kerja.** *http://www.definisi-pengertian.com/2015/07/definisi-pengertian-faktor-kecelakaan-kerja.html.* 10 Oktober 2019

[2] Muntoha, Wahyu Asyari. **Definisi Fire Alarm System.** https://patigeni.com/definisi-fire-alarm-system/#. 10 Oktober 2019

[3] Admin. **Mengenal Lebih Dekat Smart Smoke Detector.** https://www.konekthing.com/blog/2019/08/27/mengenal-lebih-dekat-smart-smoke-detector. 10 Oktober 2019

[4] Admin. **Apa itu Heat Detector?.** https:www.bromindo.com/-portfolio/heat-detector/. 22 Januari 2020

[5] Electricityofdream. **Pengertian Kegunaan dan Fungsi Arduino.** http://electricityofdream.blogspot.com/2016/09/kegunaan-dan-fungsi-arduino.html. 10 Oktober 2019

[6] Admin. **Pengertian Ethernet Shield Dan Cara Kerjanya.** https://www.immersa-lab.com/pengertian-ethernet-shield-dan-cara-kerjanya.htm. 10 Oktober 2019

[7] Faudin, Agus. **Tutorial mengakses sensor suhu DS18b20.** https://www.nyebarilmu.com/tutorial-mengakses-sensor-suhu-ds18b20/. 10 Oktober 2019

[8] DSComputer. **TP-Link TL-WR840N Wireless N Router.** http://www.dscomputer.co.id/detail/tp-link-tl-wr840n-wireless-n-router-826.html. 12 Oktober 2019

[9] Syefudin, M. **Tutorial Menggunakan Modul Relay pada Arduino.** http://indomaker.com/index.php/2019/01/03/tutorial-menggunakan-modul-relay-pada-arduino/. 12 Oktober 2019

-----Halaman ini sengaja dikosongkan-----

LAMPIRAN A

PROGRAM

```
#include <OneWire.h>
#include <DallasTemperature.h>
#include <Wire.h>

// Data wire is conntec to the Arduino digital pin 5
#define ONE_WIRE_BUS 5
#include <SPI.h>
#include <Ethernet.h>
// Setup a oneWire instance to communicate with any OneWire devices
OneWire oneWire(ONE_WIRE_BUS);

// Pass our oneWire reference to Dallas Temperature sensor
DallasTemperature sensors(&oneWire);
byte mac[] = {0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED};
IPAddress ip(192,168,0,101);
EthernetServer server(80);
int motor = 2; // motor connect to Arduino digital 2

void setup(void)
{
```

```
  pinMode(motor,OUTPUT);
  // Start serial communication for debugging purposes
  Serial.begin(9600);
  // Start up the library
  sensors.begin();
    Ethernet.begin(mac, ip);
  server.begin();
    Serial.print("server is at ");
  Serial.println(Ethernet.localIP());
}

void loop(void){
  // Call sensors.requestTemperatures() to issue a global temperature and
Requests to all devices on the bus
  sensors.requestTemperatures();
  int SensorValueSMOKE = analogRead(A1);
  int SensorValueHEAT = analogRead(A2);
  float voltageHEAT = SensorValueHEAT *(5.0 / 1023.0);
  float voltageSMOKE = SensorValueSMOKE *(5.0 / 1023.0);
  Serial.print("Voltage Heat = ");
  Serial.println(voltageHEAT);
  Serial.print("Voltage Smoke = ");
  Serial.println(voltageSMOKE);
```

```
  Serial.print("Celsius temperature: ");
  // Why "byIndex"? You can have more than one IC on the same bus. 0 refers to the first IC on the wire
  Serial.print(sensors.getTempCByIndex(0));
  Serial.print(" - Fahrenheit temperature: ");
  Serial.println(sensors.getTempFByIndex(0));
  delay(1000);
  EthernetClient client = server.available();
   if (client) {
   Serial.println("new client");
   // an http request ends with a blank line
   boolean currentLineIsBlank = true;
   String x = "";
   while (client.connected()) {
    if (client.available()) {
     char c = client.read();
     x += c;
     //Serial.write(c);
     // dibawah ini merupakan script webserver yang berisi variabel untuk request ke android.
     if(x.indexOf("suhu") >= 0 && c == '\n' && currentLineIsBlank) {

      client.println("HTTP/1.1 200 OK");
```

```
          client.println("Content-Type: text/html");
          client.println("Connection: close");
          client.println();
          client.print(sensors.getTempCByIndex(0));
          break;
        }
       if(x.indexOf("teganganHeat") >= 0 && c == '\n' &&
currentLineIsBlank) {

          client.println("HTTP/1.1 200 OK");
          client.println("Content-Type: text/html");
          client.println("Connection: close");
          client.println();
          client.print(voltageHEAT);
          break;
        }
         if(x.indexOf("teganganSmoke") >= 0 && c == '\n' &&
currentLineIsBlank) {

          client.println("HTTP/1.1 200 OK");
          client.println("Content-Type: text/html");
          client.println("Connection: close");
          client.println();
```

```
      client.print(voltageSMOKE);
      break;
    }
    if (c == '\n') {
      // you're starting a new line
      currentLineIsBlank = true;
    } else if (c != '\r') {
      // you've gotten a character on the current line
      currentLineIsBlank = false;
    }
  }
}
// give the web browser time to receive the data
delay(5);
//close the connection:
client.stop();
Serial.println("client disconnected");
}

//--------------------AKTUATOR----------------------//
//NORMAL
if (
```

```
((voltageSMOKE <= 2))
and
((voltageHEAT <= 2))
)
{
  delay(500);
  digitalWrite(motor,HIGH);
}

//ADA API
else if (
(voltageSMOKE>=2)
or
(voltageHEAT>=2.00)
)
{
  delay(50);
  digitalWrite(motor,LOW);
}
}
```

LAMPIRAN B

DATASHEET

1. *Datasheet Smoke Detector*

Technical Specifications	Smoke Detectors
Normal Rated Voltage	1.5 - 5 VDC
Operating Voltage	1.5 VDC
Alarm Current	20mA
Weight	30g

2. *Datasheet Heat Detector*

HOOSEKI

FIRE ALARM SYSTEM

Rate of Rise Heat Detector

ISO 9001 certificated CE

DESCRIPTION

The rate-of-rise heat detector, HS-WS19L, is designed to sense the rapid or abnormal rise in temperature of air. If ambient temperature rises rapidly, the air in chamber expands and presses the diaphram to close the contact point to give advanced warning. When a slower temperature change occurs, the expanded air escapes through a bleeding hole to keep the pressure balanced so that the fault alarm never occurs.

ADVANTAGES

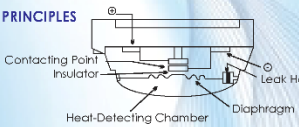
- Rate-of-rise heat detector detects intense fires by quickly responding to rapid temperature increases.
- Mechanical operating principles ensure the long-term durability and the least malfunction.
- Self-restoring ensures those elements repeated use.
- High quality components are fitted for greater reliability.
- Red indication lamp lights when detector is activated.
- High-temperature-enduring plastics and fire-proof material are used to build up the main body of the detector.
- The base is in common with other detectors made in this company for the facilitation of installation.
- Coonections are non-polarized to avoid wiring mistakes.

SPECIFICATION

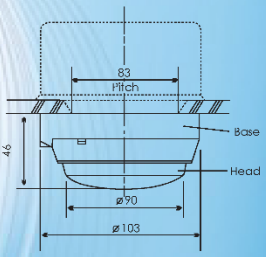
Model	HS-WS19L
Type	2 wire
Voltage Range	12~30V DC
Alarm Current	24V DC 30mA
Contact Rating	N/A
Polarization	Non-polarized
Alarm Temp.	Comply to EN54, CNS

Ambient Temp.	-10°C~+55°C
Ambient Humidity	0~95%
Material	Fire-proff Plastics
Dimensions	102 (Dia.) x 42 (H)mm
Weight	130g
Color	White

OPERATION PRINCIPLES

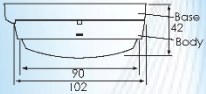


MOUNTING STATUS

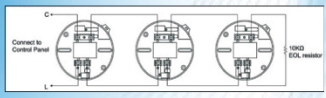


DIMENSIONS

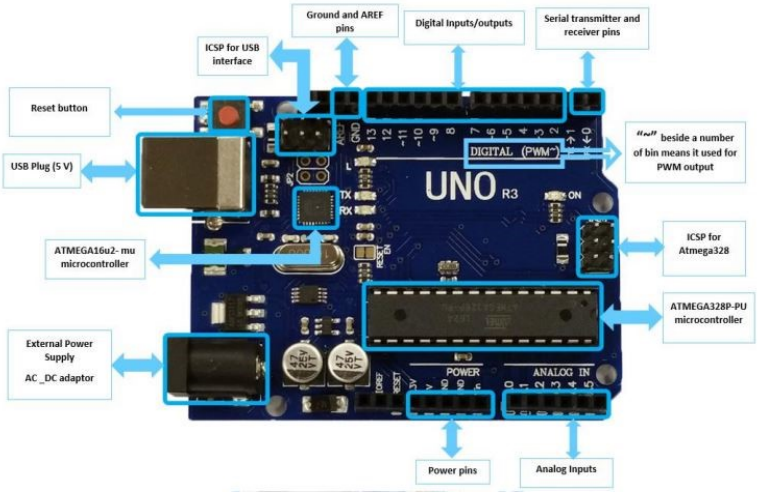
Unit: mm



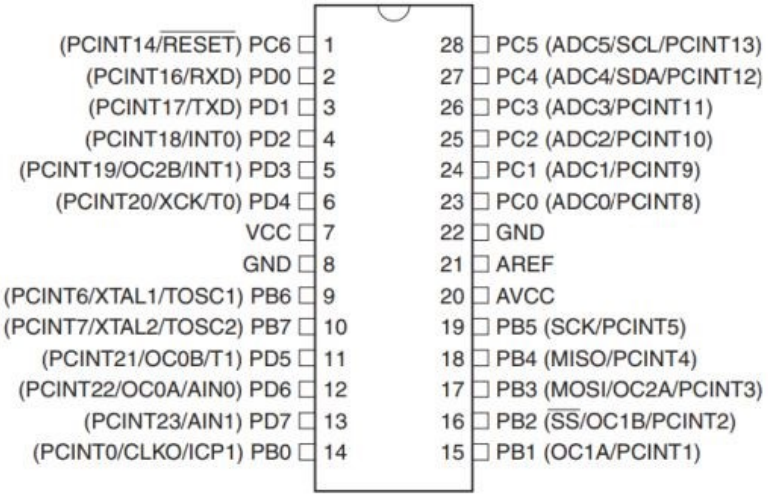
WIRING AND CONNECTION



3. *Datasheet* Arduino Uno



Pin configuration



4. *Datasheet* DS18B20

Click here for production status of specific part numbers.

DS18B20

Programmable Resolution 1-Wire Digital Thermometer

General Description

The DS18B20 digital thermometer provides 9-bit to 12-bit Celsius temperature measurements and has an alarm function with nonvolatile user-programmable upper and lower trigger points. The DS18B20 communicates over a 1-Wire bus that by definition requires only one data line (and ground) for communication with a central microprocessor. In addition, the DS18B20 can derive power directly from the data line ("parasite power"), eliminating the need for an external power supply.

Each DS18B20 has a unique 64-bit serial code, which allows multiple DS18B20s to function on the same 1-Wire bus. Thus, it is simple to use one microprocessor to control many DS18B20s distributed over a large area. Applications that can benefit from this feature include HVAC environmental controls, temperature monitoring systems inside buildings, equipment, or machinery, and process monitoring and control systems.

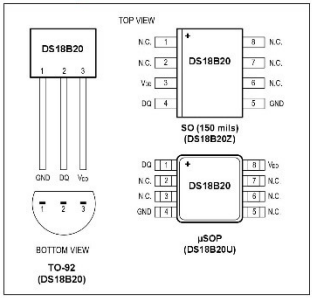
Applications

- Thermostatic Controls
- Industrial Systems
- Consumer Products
- Thermometers
- Thermally Sensitive Systems

Benefits and Features

- Unique 1-Wire® Interface Requires Only One Port Pin for Communication
- Reduce Component Count with Integrated Temperature Sensor and EEPROM
  - Measures Temperatures from -55°C to +125°C (-67°F to +257°F)
  - ±0.5°C Accuracy from -10°C to +85°C
  - Programmable Resolution from 9 Bits to 12 Bits
  - No External Components Required
- Parasitic Power Mode Requires Only 2 Pins for Operation (DQ and GND)
- Simplifies Distributed Temperature-Sensing Applications with Multidrop Capability
  - Each Device Has a Unique 64-Bit Serial Code Stored in On-Board ROM
- Flexible User-Definable Nonvolatile (NV) Alarm Settings with Alarm Search Command Identifies Devices with Temperatures Outside Programmed Limits
- Available in 8-Pin SO (150 mils), 8-Pin µSOP, and 3-Pin TO-92 Packages

Pin Configurations



Ordering Information ***appears at end of data sheet.***

1-Wire is a registered trademark of Maxim Integrated Products, Inc.

19-7487; Rev 5; 9/18

maxim integrated

Absolute Maximum Ratings

Voltage Range on Any Pin Relative to Ground....-0.5V to +6.0V
Operating Temperature Range..... -55°C to +125°C
Storage Temperature Range..... -55°C to +125°C
Solder Temperature..... Refer to the IPC/JEDEC J-STD-020 Specification.

These are stress ratings only and functional operation of the device at these or any other conditions above those indicated in the operation sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods of time may affect reliability.

DC Electrical Characteristics

(-55°C to +125°C; V_{DD} = 3.0V to 5.5V)

PARAMETER	SYMBOL	CONDITIONS		MIN	TYP	MAX	UNITS
Supply Voltage	V_{DD}	Local power (Note 1)		+3.0		+5.5	V
Pullup Supply Voltage	V_{PU}	Parasite power	(Notes 1, 2)	+3.0		+5.5	V
		Local power		+3.0		V_{DD}	
Thermometer Error	t_{ERR}	-10°C to +85°C	(Note 3)			±0.5	°C
		-30°C to +100°C				±1	
		-55°C to +125°C				±2	
Input Logic-Low	V_{IL}	(Notes 1, 4, 5)		-0.3		+0.8	V
Input Logic-High	V_{IH}	Local power	(Notes 1,6)	+2.2		The lower of 5.5 or V_{DD} + 0.3	V
		Parasite power		+3.0			
Sink Current	I_L	$V_{I/O}$ = 0.4V		4.0			mA
Standby Current	I_{DDS}	(Notes 7, 8)			750	1000	nA
Active Current	I_{DD}	V_{DD} = 5V (Note 9)			1	1.5	mA
DQ Input Current	I_{DQ}	(Note 10)			5		µA
Drift		(Note 11)			±0.2		°C

Note 1: All voltages are referenced to ground.
Note 2: The Pullup Supply Voltage specification assumes that the pullup device is ideal, and therefore the high level of the pullup is equal to V_{PU}. In order to meet the V_{IH} spec of the DS18B20, the actual supply rail for the strong pullup transistor must include margin for the voltage drop across the transistor when it is turned on; thus: $V_{PU_ACTUAL} = V_{PU_IDEAL} + V_{TRANSISTOR}$.
Note 3: See typical performance curve in Figure 1. Thermometer Error limits are 3-sigma values.
Note 4: Logic-low voltages are specified at a sink current of 4mA.
Note 5: To guarantee a presence pulse under low voltage parasite power conditions, V_{ILMAX} may have to be reduced to as low as 0.5V.
Note 6: Logic-high voltages are specified at a source current of 1mA.
Note 7: Standby current specified up to +70°C. Standby current typically is 3µA at +125°C.
Note 8: To minimize I_{DDS}, DQ should be within the following ranges: GND ≤ DQ ≤ GND + 0.3V or V_{DD} – 0.3V ≤ DQ ≤ V_{DD}.
Note 9: Active current refers to supply current during active temperature conversions or EEPROM writes.
Note 10: DQ line is high ("high-Z" state).
Note 11: Drift data is based on a 1000-hour stress test at +125°C with V_{DD} = 5.5V.

AC Electrical Characteristics–NV Memory

(-55°C to +125°C; V_{DD} = 3.0V to 5.5V)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
NV Write Cycle Time	t_{WR}			2	10	ms
EEPROM Writes	N_{EEWR}	-55°C to +55°C	50k			writes
EEPROM Data Retention	t_{EEDR}	-55°C to +55°C	10			years

AC Electrical Characteristics

(-55°C to +125°C; V_{DD} = 3.0V to 5.5V)

PARAMETER	SYMBOL	CONDITIONS		MIN	TYP	MAX	UNITS
Temperature Conversion Time	t_{CONV}	9-bit resolution	(Note 12)			93.75	ms
		10-bit resolution				187.5	
		11-bit resolution				375	
		12-bit resolution				750	
Time to Strong Pullup On	t_{SPON}	Start convert T command issued				10	µs
Time Slot	t_{SLOT}	(Note 12)		60		120	µs
Recovery Time	t_{REC}	(Note 12)		1			µs
Write 0 Low Time	t_{LOW0}	(Note 12)		60		120	µs
Write 1 Low Time	t_{LOW1}	(Note 12)		1		15	µs
Read Data Valid	t_{RDV}	(Note 12)				15	µs
Reset Time High	t_{RSTH}	(Note 12)		480			µs
Reset Time Low	t_{RSTL}	(Notes 12, 13)		480			µs
Presence-Detect High	t_{PDHIGH}	(Note 12)		15		60	µs
Presence-Detect Low	t_{PDLOW}	(Note 12)		60		240	µs
Capacitance	$C_{IN/OUT}$					25	pF

Note 12: See the timing diagrams in Figure 2.
Note 13: Under parasite power, if t_{RSTL} > 960µs, a power-on reset can occur.

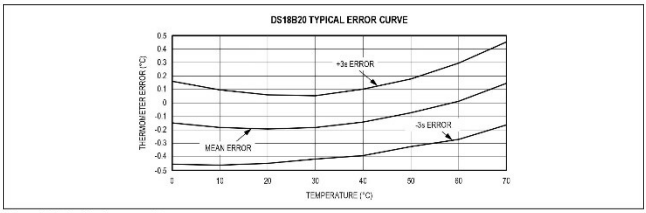


Figure 1. Typical Performance Curve

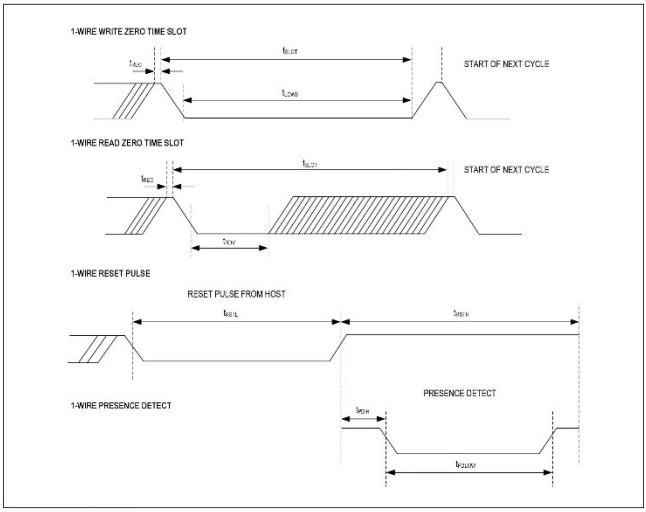


Figure 2. Timing Diagrams

Pin Description

PIN			NAME	FUNCTION
SO	µSOP	TO-92		
1, 2, 6, 7, 8	2, 3, 5, 6, 7	—	N.C.	No Connection
3	8	3	V_{DD}	Optional V_{DD}. V_{DD} must be grounded for operation in parasite power mode.
4	1	2	DQ	Data Input/Output. Open-drain 1-Wire interface pin. Also provides power to the device when used in parasite power mode (see the *Powering the DS18B20* section.)
5	4	1	GND	Ground

-----Halaman ini sengaja dikosongkan-----

RIWAYAT HIDUP PENULIS

Nama	: Fitria Rahmawati
TTL	: 26 Januari 1998
Jenis Kelamin	: Perempuan
Agama	: Islam
Alamat Asal	: Rejosari Baru Aspol Pel. Tg. Perak M/23 Surabaya
Telp/ HP	: 082264015431
E-mail	: *fitriarahmawati1998@gmail.com*

RIWAYAT PENDIDIKAN

- 2004 – 2010 : SDIT Al-Fathimiyyah Surabaya
- 2010 – 2013 : MTs Negeri 3 Surabaya
- 2013 – 2016 : SMA Ta'miriyah Surabaya
- 2016 – Sekarang : Departemen Teknik Elektro Otomasi Institut Teknologi Sepuluh Nopember Surabaya

PENGALAMAN KERJA

- Kerja Praktek PT. Pembangkit Jawa-Bali Gresik
- Magang Industri PT. Graha Pena Surabaya

PENGALAMAN ORGANISASI

- Remaja Bhayangkara Club Aspol KP3
- Departemen Minat Bakat HIMAD3TEKTRO 2017-2018
- Departemen Minat Bakat HIMAD3TEKTRO 2018-2019