



TUGAS AKHIR - TE 141599

SISTEM PERINGATAN DINI DETEKSI DAN PEMADAM KEBAKARAN BERBASIS RASPBERRY PI

Sumarto
NRP 2212100001

Dosen Pembimbing
Dr. Muhammad Rivai, ST., MT.
Fajar Budiman, ST., M.Sc.

DEPARTEMEN TEKNIK ELEKTRO
Fakultas Teknologi Elektro
Institut Teknologi Sepuluh Nopember
Surabaya 2017



ITS
Institut
Teknologi
Sepuluh Nopember

TUGAS AKHIR- TE141599

**SISTEM PERINGATAN DINI DETEKSI
DAN PEMADAM KEBAKARAN BERBASIS RASPBERRY PI**

Sumarto
NRP 2212100001

Dosen Pembimbing
Dr. Muhammad Rivai, ST., MT.
Fajar Budiman, ST., M.Sc.

DEPARTEMEN TEKNIK ELEKTRO
Fakultas Teknologi Elektro
Institut Teknologi Sepuluh Nopember
Surabaya 2017



FINAL PROJECT - TE 141599

**EARLY WARNING FIRE DETECTION SYSTEM
BASED ON RASPBERRY PI**

Sumarto
NRP 2212100001

Advisor
Dr. Muhammad Rivai, ST., MT.
Fajar Budiman, ST., M.Sc.

DEPARTMENT OF ELECTRICAL ENGINEERING
Faculty of Electrical Technology
Institut Teknologi Sepuluh Nopember
Surabaya 2017

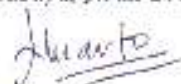
PERNYATAAN KEASLIAN TUGAS AKHIR

Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan Tugas Akhir saya dengan judul "SISTEM PERINGATAN DINI DETEKSI dan PEMADAM KEBAKARAN BERBASIS RASPBERRY PI" 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, 3 Juli 2017



Sumarto

Nrp. 2212 100 001


**SISTEM PERINGATAN DINI DETEKSI DAN PEMADAM
KEBAKARAN BERBASIS RASPBERRY PI**

TUGAS AKHIR


Diajukan untuk Memenuhi Sebagian Persyaratan
Untuk Memperoleh Gelar Sarjana Teknik
Pada
Bidang Studi Elektronika Industri
Departemen Teknik Elektro
Institut Teknologi Sepuluh Nopember

Menyetujui:

Dosen Pembimbing I


Dr. Muhammad Rivai, ST., MT.
NIP. 196904261994031003

Dosen Pembimbing II


Fajar Budiman, ST., M.Sc.
NIP. 198607072014041001



Halaman ini sengaja dikosongkan

SISTEM PERINGATAN DINI DETEKSI DAN PEMADAM KEBAKARAN BERBASIS RASPBERRY PI

Nama : Sumarto
Dosen Pembimbing I : Dr. Muhammad Rivai, ST.,MT.
Dosen Pembimbing II : Fajar Budiman, ST., M.Sc.

ABSTRAK

Kebakaran merupakan hal yang sangat sering terjadi terutama pada rumah yang ditinggal pergi oleh penghuninya. Ketika terjadi kebakaran, intensitas api saat diketahui cenderung sudah membesar sehingga sulit untuk ditanggulangi. Untuk itu pada penelitian ini telah dirancang dan dibuat suatu sistem peringatan dini pendeteksi kebakaran yang dapat menginformasikan kepada penghuni rumah. Sehingga penghuni dapat menanggulangi/mencegah terjadinya kebakaran. Pada sistem ini menitikberatkan pada metode pendeteksian kebakaran dengan menggunakan 4 buah sensor yaitu *flame sensor* 5 kanal, *photoelectric IR break beam sensor*, sensor gas *MQ-5* dan *MQ-7*. Hal ini bertujuan untuk memperoleh tingkat keakurasian yang tinggi sehingga dapat menghindari adanya kesalahan informasi. Keluaran sistem yang berupa kondisi tingkat bahaya akan mengaktifkan pemadaman otomatis, dan disampaikan ke penghuni berupa *alarm*, *SMS Gateway* dan notifikasi email. Informasi tersebut berisi gambar kondisi ruangan sehingga dapat dimonitor jarak jauh melalui internet. *Raspberry pi* digunakan untuk membaca inputan sensor dan sebagai *server*. Hasil percobaan menunjukkan ketika tegangan *flame sensor* 5 kanal lebih dari 4V maka sistem mengirim *SMS*, email ke penghuni serta mengaktifkan *alarm* dan *water solenoid valve* untuk menyiram air. Ketika konsentrasi gas *LPG* diatas 1000 *ppm* atau *CO* 200 *ppm* maka sistem mengirim *SMS* ke penghuni berisi konsentrasi gas melebihi ambang batas. Sistem membutuhkan waktu untuk pengiriman *SMS* rata-rata 6 detik.

Kata kunci: Kebakaran, Notifikasi Email, *Raspberry Pi*, *SMS Gateway*.

Halaman ini sengaja dikosongkan

EARLY WARNING FIRE DETECTION SYSTEM BASED ON RASPBERRY PI

Name : Sumarto
1st Supervisor : Dr. Muhammad Rivai, ST., MT.
2nd Supervisor : Fajar Budiman, ST., M.Sc.

ABSTRACT

Fire at home happens very frequent especially in the home that is being leaved by its residents. When fire occurs, people can only realize when the flame intensity has already huge. So it makes hard to be extinguisher. For that reason this research has been conducted to developed an early warning system to detect fire that will inform to home residents. So that the resident can prevent the home fire. this system focus on the fire detection that is developed using 4 sensors which are 5 channel flame sensors, photoelectric IR break beam sensor, MQ-5 and MQ-7 gas sensor. The goal is to obtain a high level of accuracy sensor to prevent misinformation. The output of this system for a danger level conditions will activate the automatic extinguisher, then the alarm, SMS Gateway, and a notification email will be sent to the resident. The information contains the room conditions so that the room will be monitored for long distance via internet. Raspberry pi is employed to read the sensor inputs and also as a server. The experiment results shows that when the voltages at the 5 channel flame sensors are over 4 volts then the system will send an SMS and an email to the resident and also activate the alarm and the water solenoid valve to drain for water. When the LPG gas concentration is over 1000 ppm or CO 200 ppm then the system will send an SMS to the resident containing the gas treshold message. The time needed to send the SMS on average is 6 seconds.

Keywords: *Email notification, Fire, Raspberry Pi, SMS Gateway*

Halaman ini sengaja dikosongkan

KATA PENGANTAR

Puji syukur penulis panjatkan kehadiran Allah SWT yang selalu memberikan rahmat dan hidayah-Nya sehingga penelitian ini dapat terselesaikan tepat waktu. Shalawat serta salam semoga selalu dilimpahkan kepada Rasulullah Muhammad SAW.

Penelitian ini disusun untuk memenuhi sebagian persyaratan guna menyelesaikan pendidikan Sarjana pada Bidang Studi Elektronika Industri, Jurusan Teknik Elektro, Fakultas Teknologi Elektro, Institut Teknologi Sepuluh Nopember Surabaya yang berjudul :

SISTEM PERINGATAN DINI DETEKSI DAN PEMADAM KEBAKARAN BERBASIS *RASPBERRY PI*

Pada kesempatan ini penulis menyampaikan terima kasih kepada :

1. Bapak Dr. Muhammad Rivai, ST., MT. dan Bapak Fajar Budiman., ST., M.Sc. selaku dosen pembimbing yang telah banyak memberikan saran serta bimbingan.
2. Bapak Rahmad Setiawan, ST., MT., Bapak Astria Nur Irfansyah, ST., M.Eng., Bapak Dr. Tri Arief Sardjono, ST., MT., dan Ibu Ir.Siti Halimah Baki., MT. selaku dosen penguji yang telah banyak memberikan saran serta evaluasi.
3. Bapak Dr. Ardyono Priyadi, S.T, M.Eng. selaku Ketua Jurusan Teknik Elektro ITS serta seluruh dosen Teknik Elektro ITS yang telah memberikan banyak ilmu selama proses perkuliahan.
4. Dimas Arif, Ichsan Adi Pratama, dan Hendra Irawan yang turut membantu dalam pengerjaan penelitian ini.
5. Ibu, dan Kakak yang telah banyak memberikan bantuan baik doa, materi, tenaga maupun waktu.

Penulis menyadari bahwa laporan penelitian ini masih banyak kekurangan, oleh karena itu saran dan masukan sangat diharapkan untuk perbaikan di masa yang akan datang. Semoga penelitian ini bermanfaat bagi pembaca dan masyarakat pada umumnya.

Surabaya, Juli 2017

Penulis

Halaman ini sengaja dikosongkan

DAFTAR ISI

HALAMAN JUDUL	
LEMBAR PERNYATAAN KEASLIAN	
LEMBAR PENGESAHAN	
ABSTRAK	i
ABSTRACT	iii
KATA PENGANTAR	v
DAFTAR ISI	vii
DAFTAR GAMBAR	ix
DAFTAR TABEL	xi

BAB I PENDAHULUAN

1.1 Latar Belakang	1
1.2 Permasalahan	2
1.3 Tujuan	2
1.4 Metodologi	2
1.5 Sistematika Penulisan	5
1.6 Relevansi	5

BAB II DASAR TEORI

2.1 Identifikasi Kebakaran	7
2.2 <i>Infrared Flame Sensor</i> 5 Kanal	8
2.3 <i>Photoelectric IR Break Beam Sensor</i>	9
2.4 <i>MQ-5</i>	10
2.5 <i>MQ-7</i>	11
2.6 <i>Analog to Digital Converter (ADC)</i>	13
2.7 <i>Serial Peripheral Interface (SPI)</i>	14
2.8 Raspberry pi 3 Model B	15
2.9 <i>SMS Gateway</i>	17
2.10 <i>Relay</i>	19
2.11 <i>Water Solenoid Valve</i>	19
2.12 <i>Webcam M-Tech Box 6-LED</i>	20
2.13 <i>Buzzer</i>	21
2.14 Program <i>Python IDLE 2</i>	21
2.15 <i>Wvdial</i> untuk <i>dial-up modem</i> di Raspberry Pi	22
2.16 <i>SMS Gateway</i> Menggunakan <i>Gammu</i>	22
2.17 <i>OpenCV</i>	23
2.18 <i>SSMTP (Software Simple Mail Transfer Protokol)</i>	24

BAB III PERANCANGAN SISTEM

3.1 Konfigurasi Sistem	25
3.2 Perancangan Teknis.....	26
3.3 Perancangan <i>Hardware</i>	27
3.3.1 <i>Infrared Flame Sensor</i> 5 kanal	27
3.3.2 <i>Photoelectric IR Break Beam Sensor</i>	28
3.3.3 <i>MQ-5</i>	28
3.3.4 <i>MQ-7</i>	30
3.3.5 Komunikasi <i>Serial Modem Huawei E173</i>	31
3.3.6 Komunikasi <i>Serial Webcam</i>	31
3.3.7 <i>Buzzer</i> dan <i>Water Solenoid</i>	31
3.4 Perancangan <i>Software</i>	32
3.4.1 Program <i>Python IDLE 2</i> Untuk <i>ADC</i>	33
3.4.2 Program <i>Python IDLE 2</i> Untuk <i>SMS</i>	33
3.4.3 Program <i>Python IDLE 2</i> Untuk <i>EMAIL</i>	33
3.4.4 Program <i>Python IDLE 2</i> Untuk <i>Webcam</i>	33
3.4.5 Program <i>Python IDLE 2</i> Untuk <i>GPIO</i>	34
3.4.6 Algoritma Pengambilan Keputusan.....	34

BAB IV PENGUJIAN dan ANALISA

4.1 Desain Sistem.....	35
4.2 Pengujian Pembacaan Nilai <i>ADC</i>	36
4.3 Pengujian <i>Flame Sensor</i> 5 kanal	37
4.4 Pengujian <i>Photoelectric IR Break Beam Sensor</i>	41
4.5 Pengujian <i>MQ-5</i>	43
4.6 Pengujian <i>MQ-7</i>	46
4.7 Pengujian & Penentuan <i>Threshold</i>	48
4.8 Pengujian Modem <i>Huawei E173</i>	49
4.9 Pengujian <i>Webcam</i>	50
4.10 Pengujian <i>Buzzer</i>	51
4.11 Pengujian <i>Water Solenoid valve</i>	52
4.12 Pengujian Integrasikan Seluruh Sistem.....	53

BAB V PENUTUP

5.1 Kesimpulan	55
5.2 Saran.....	55
DAFTAR PUSTAKA	57
BIODATA PENULIS	59
LAMPIRAN	61

DAFTAR GAMBAR

Gambar 2.1	Rangkaian Dasar Photodioda	8
Gambar 2.2	Spektrum Gelombang <i>Infra Red</i>	9
Gambar 2.3	Rangkaian Dasar <i>Photoelectric IR Break Beam Sensor</i>	9
Gambar 2.4	Skematik <i>Photoelectric IR Break Beam Sensor</i>	9
Gambar 2.5	Konfigurasi <i>MQ-5</i>	10
Gambar 2.6	Prinsip Kerja <i>MQ-5</i>	11
Gambar 2.7	Konfigurasi <i>MQ-7</i>	11
Gambar 2.8	Prinsip Kerja <i>MQ-7</i>	12
Gambar 2.9	Proses <i>ADC</i>	13
Gambar 2.10	<i>MCP3008</i>	13
Gambar 2.11	<i>Setup Hardware SPI</i>	14
Gambar 2.12	<i>Single Master Diagram Interface SPI</i>	14
Gambar 2.13	<i>Raspberry Pi 3 Model B</i>	15
Gambar 2.14	<i>Pin GPIO Raspberry Pi 3 model B</i>	17
Gambar 2.15	<i>SMS Gateway</i>	17
Gambar 2.16	Proses <i>SMS</i>	18
Gambar 2.17	<i>Relay</i>	19
Gambar 2.18	Skematik <i>Relay Driver</i>	19
Gambar 2.19	<i>Water Solenoid Valve</i>	20
Gambar 2.20	<i>Webcam M-Tech Box 6-LED</i>	20
Gambar 2.21	<i>Buzzer</i>	21
Gambar 2.22	Tampilan <i>menu Python Idle 2</i>	22
Gambar 2.23	Konfigurasi <i>Gammu</i>	23
Gambar 3.1	Konfigurasi Sistem	25
Gambar 3.2	Perancangan Teknis	26
Gambar 3.3	Karakteristik <i>Sensitifitas MQ-5</i>	29
Gambar 3.4	Karakteristik Temperatur dan Kelembabpan <i>MQ-5</i>	29
Gambar 3.5	Karakteristik <i>Sensitifitas MQ-7</i>	30
Gambar 3.6	Karakteristik Temperatur dan Kelembabpan <i>MQ-7</i>	31
Gambar 3.7	<i>Flowchart Perancangan Software</i>	32
Gambar 4.1	Desain Sistem Tampak Depan	35
Gambar 4.2	Desain Sistem Tampak Belakang	35
Gambar 4.3	Grafik hubungan jarak dengan tegangan kanal 1	38
Gambar 4.4	Grafik hubungan jarak dengan tegangan kanal 2	39
Gambar 4.5	Grafik hubungan jarak dengan tegangan kanal 3	40
Gambar 4.6	Grafik hubungan jarak dengan tegangan kanal 4	40
Gambar 4.7	Grafik hubungan jarak dengan tegangan kanal 5	41

Gambar 4.8	Pengujian <i>IR Break Beam Sensor</i>	42
Gambar 4.9	Grafik hubungan waktu dengan konsentrasi <i>LPG</i>	44
Gambar 4.10	Power Regresi <i>Rs/Ro</i> dengan Konsentrasi <i>MQ-5</i>	45
Gambar 4.11	Grafik Asap Material dengan Konsentrasi <i>CO</i>	46
Gambar 4.12	Power Regresi <i>Rs/Ro</i> dengan Konsentrasi <i>MQ-7</i>	48
Gambar 4.13	Pengujian modem <i>Huawei 173</i>	50
Gambar 4.14	Pengujian <i>webcam</i> dan email	51

DAFTAR TABEL

Tabel 2.1 Konfigurasi <i>MQ-5</i>	10
Tabel 2.2 Konfigurasi <i>MQ-7</i>	12
Tabel 4.1 Hasil Pembacaan Nilai <i>ADC MCP3008</i>	36
Tabel 4.2 Hasil Pembacaan Nilai <i>ADC MCP3008</i>	36
Tabel 4.3 Hasil Pengujian <i>Flame Sensor</i> kanal 1	38
Tabel 4.4 Hasil Pengujian <i>Flame Sensor</i> kanal 2	39
Tabel 4.5 Hasil Pengujian <i>Flame Sensor</i> kanal 3	39
Tabel 4.6 Hasil Pengujian <i>Flame Sensor</i> kanal 4	40
Tabel 4.7 Hasil Pengujian <i>Flame Sensor</i> kanal 5	41
Tabel 4.8 Hasil Pengujian <i>MQ-5</i>	44
Tabel 4.9 Perbandingan <i>Rs/Ro</i> dengan Konsentrasi <i>MQ-5</i>	45
Tabel 4.10 Hasil Pengujian <i>MQ-7</i>	46
Tabel 4.11 Perbandingan <i>Rs/Ro</i> dengan Konsentrasi <i>MQ-7</i>	47
Tabel 4.12 Penentuan <i>Threshold</i>	49
Tabel 4.13 Lama Waktu Pengiriman <i>SMS</i>	50
Tabel 4.14 Hasil Integrasi Sistem (<i>threshold</i> tegangan).....	53
Tabel 4.15 Hasil Integrasi Sistem (<i>threshold ppm</i>)	53

Halaman ini sengaja dikosongkan

BAB I

PENDAHULUAN

1.1 Latar Belakang

Rumah kosong yang telah ditinggal penghuninya jelas tidak bisa lepas dari kemungkinan kecelakaan kerja atau bahaya yang salah satunya adalah kebakaran. Aspek bahaya ini menjadikan penghuni rumah membuat dan menciptakan suatu sistem keselamatan kerja. Selain itu perlu dipahami pula bagaimana proses terjadinya kebakaran, bahan-bahan kimia apa saja yang mudah terbakar serta bagaimana cara penanggulangannya secara benar. Definisi api adalah suatu fenomena yang dapat diamati dengan adanya cahaya dan panas serta adanya proses perubahan zat menjadi zat baru melalui reaksi kimia oksidasi *eksotermal*. Api terbentuk karena adanya interaksi beberapa unsur/elemen yang pada kesetimbangan tertentu dapat menimbulkan api. Sedangkan kebakaran yaitu peristiwa bencana yang ditimbulkan oleh api, yang tidak dikehendaki oleh manusia dan bisa mengakibatkan kerugian nyawa dan harta.

Kebakaran merupakan hal yang sangat sering terjadi terutama pada rumah yang ditinggal pergi oleh penghuninya, hal ini akan diperparah karena ketika terjadi kebakaran pada ruang kosong intensitas api saat diketahuinya terjadinya kebakaran tersebut cenderung sudah membesar sehingga sulit untuk ditanggulangi. Pada tahun 2010 terjadi 77 kasus kebakaran, 2011 terjadi 203 kasus kebakaran, 2012 terjadi 166 kasus kebakaran di Jakarta dari semua kasus kebakaran tersebut 80% terjadi sepanjang Ramadhan dan lebaran yang kebanyakan ditinggal pergi oleh pemiliknya dan kerugian yang harus ditanggung tercatat lebih dari Rp 200 milyar. Berdasarkan data Dinas Damkar-PB DKI Jakarta, jumlah kebakaran sejak Januari 2013 telah mencapai 739 kasus. Kejadian itu menimbulkan 36 korban jiwa meninggal dunia dan 54 korban luka-luka. Kerugian akibat kebakaran tersebut diperkirakan mencapai Rp 124 miliar [1].

Untuk itu pada penelitian ini dirancanglah suatu sistem peringatan dini pendeteksi kebakaran yang dapat menginformasikan kepada penghuni ketika terjadi indikasi kebakaran. Sehingga memungkinkan penghuni untuk menanggulangi/mencegah terjadinya kebakaran yang membesar. Pada sistem ini menitikberatkan pada metode pendeteksian kebakaran dengan menggunakan 4 buah sensor yaitu *flame sensor* 5

kanal, *photoelectric IR break beam sensor*, sensor gas *MQ-5* dan *MQ-7* untuk memperoleh tingkat keakuratan yang tinggi. Sehingga dapat menghindari adanya kesalahan penyampaian informasi kepada penghuni. Keluaran sistem yang berupa kondisi tingkat bahaya akan mengaktifkan pemadaman otomatis, dan disampaikan ke penghuni berupa *alarm*, *SMS Gateway* dan notifikasi email. Informasi tersebut berisi gambar kondisi ruangan sehingga dapat dimonitor jarak jauh melalui internet. Raspberry Pi digunakan untuk membaca sinyal sensor dan sebagai server. Dengan demikian dengan adanya sistem ini diharapkan dapat mendeteksi dini kebakaran dan penanggulangan kebakaran yang lebih cepat.

1.2 Permasalahan

Permasalahan dalam penelitian yang akan diangkat adalah :

1. Bagaimana sistem atau alat dituntut untuk dapat mengidentifikasi kebakaran yang presisi dan akurat?
2. Bagaimana cara sistem mengontrol pemadam berdasarkan identifikasi kebakaran?
3. Bagaimana sistem dapat mengirimkan *SMS* kepada penghuni?
4. Bagaimana cara sistem dapat mengambil gambar kondisi rumah dan mengirimkannya ke email penghuni?

1.3 Tujuan

Tujuan dari penulisan penelitian ini adalah:

1. Sistem menggunakan *flame sensor* 5 kanal, *photoelectric IR break beam sensor*, sensor gas *MQ-5* dan *MQ-7* untuk mengidentifikasi sumber api, asap, bahan bakar (*LPG*), dan gas karbon monoksida agar presisi dan akurat.
2. Sistem menggunakan *relay* untuk mengontrol *Solenoid Valve* mengalirkan air sebagai pemadam kebakaran.
3. Sistem yang mampu mengirimkan *sms* dengan menggunakan modem *Huawei E173*.
4. Sistem menggunakan *webcam USB* untuk mengambil gambar dan Raspberry pi mengirimkannya ke email penghuni.

1.4 Metodologi

Pengerjaan studi penelitian ini terbagi atas delapan tahapan dengan uraian sebagai berikut :

1. Study Literatur

Tahapan ini adalah serangkaian kegiatan pencarian dan pengkajian referensi yang berhubungan dengan topik penelitian. Referensi studi literatur didapatkan melalui buku, jurnal ilmiah, dan *browsing* melalui internet yang berhubungan dengan judul penelitian ini. Studi literatur dilakukan secara beriringan dengan penelitian yang dilakukan.

2. Perancangan Sistem

Pada penelitian ini menitikberatkan pada kemampuan alat untuk membaca kondisi rumah yang ditentukan dengan menggunakan beberapa parameter sensor. Karena sensor memiliki kecepatan respon yang berbeda-beda maka diperlukan metode yang dapat mengakomodasi semua data dari tiap-tiap sensor sebelum ditentukan sebuah informasi yang sedang terjadi. Perancangan lokasi tiap sensor adalah terdiri dari penggunaan 4 buah sensor yaitu *flame sensor* 5 kanal, *photoelectric IR break beam sensor*, sensor gas *MQ-5* dan *MQ-7* yang diletakkan dalam satu *box* pada *center* setiap dinding tembok dengan jarak ± 30 cm dari lantai dengan asumsi sebagian besar sumber api terletak/terjadi pada lantai dengan intensitas yang masih cukup kecil. Setelah data berhasil dikumpulkan di Raspberry pi maka selanjutnya akan diproses untuk menentukan data yang masuk apakah sesuai dengan *set point* tiap sensor dan memenuhi kondisi tertentu untuk tiap-tiap sensor dengan parameter perubahan nilai input tiap sensor. Ketika memperoleh data dari tiap sensor maka selanjutnya akan dilakukan *delay* untuk menunggu data masukan sensor yang lain karena tiap sensor mempunyai kecepatan respon yang berbeda. Kemudian setelah dikondisikan hasil pembacaan keseluruhan sensor maka keluaran berupa informasi kondisi ruangan akan dikirimkan ke modem untuk diteruskan dengan *sms* ke penghuni, pengambilan gambar dengan *Webcam USB* dan dikirim ke email penghuni serta mengalirkan air melalui *Water solenoid valve* dan mengaktifkan bunyi *alarm*.

3. Perancangan Desain
Pada desain penelitian ini dibutuhkan suatu *box* untuk meletakkan *flame sensor* 5 kanal, *photoelectric IR break beam sensor*, sensor gas *MQ-5* dan *MQ-7*. Pada *box* tersebut berisi Raspberry pi, Modem *GSM Huawei E173*, *webcam USB* serta sensor dan *pheriperalnya*.
4. Pembuatan Perangkat Keras dan Perangkat Lunak
Sistem yang terdapat dalam penelitian ini terdiri dari perangkat keras dan perangkat lunak. perangkat keras dan perangkat lunak dibuat setelah melalui proses perancangan sesuai dengan sistem yang telah dirancang. Pada tahap pembuatan perangkat keras ini terdiri dari pembuatan sebuah rangkaian pengontrol menggunakan *Raspberry pi*, Modem *GSM Huawei E173*, *webcam USB*, *Solenoid Valve*, *alarm*, *flame sensor* 5 kanal, *photoelectric IR break beam sensor*, *MQ-5* dan *MQ-7*. Sedangkan perangkat lunak meliputi perancangan sistem dan program *Phyton* untuk menentukan keputusan informasi ruangan dari nilai yang dibaca sensor.
5. Pengujian dan Integrasi Sistem
Pada tahap ini dilakukan pengujian komunikasi antara tiap-tiap sensor dengan raspberry pi secara parsial serta pengujian seluruh sensor ketika digabungkan, komunikasi modem *GSM Huawei E173*, *webcam USB*, *Solenoid Valve* dan *alarm*. Setelah itu semua sistem di integrasikan menjadi satu sistem yang saling terkoneksi.
6. Pengumpulan dan Analisa Data
Dari hasil pengujian didapatkan beberapa data yang dapat digunakan untuk menganalisa kondisi yang tercover oleh sistem. Pengumpulan data dan analisa ini juga dapat menunjukkan kemampuan dan tingkat keberhasilan dari seluruh sistem.
7. Evaluasi Hasil Pengujian Sistem
Dari hasil pengujian, pengumpulan data dan analisa data dapat dilakukan sebuah evaluasi keseluruhan sistem sehingga dapat dilakukan perbaikan maupun pengembangan untuk peningkatan kemampuan dari sistem itu sendiri.
8. Pembuatan Laporan
Pembuatan laporan dimaksudkan untuk dokumentasi dari keseluruhan pembuatan sistem pada penelitian ini mulai dari

perancangan sampai pengujian sistem. Pada laporan juga disertakan data-data yang dapat menunjukkan kemampuan dari sistem yang dibuat.

1.5 Sistematika Penulisan

Penulisan penelitian ini terbagi atas lima bagian dengan rincian sebagai berikut:

1. Bab 1 merupakan pendahuluan yang membahas tentang latar belakang, permasalahan, tujuan, metode penelitian, manfaat, serta sistematika penulisan penelitian.
2. Bab 2 merupakan dasar teori yang menjelaskan identifikasi api dan gejala-gejala kebakaran, *flame sensor* 5 kanal, *photoelectric IR break beam sensor*, *MQ-5*, *MQ-7*, komunikasi serial (*SPI MCP3008*), Raspberrpy pi 3 model B, *Solenoid Valve*, *Modem GSM Huawei E173*, *webcam USB*, *alarm* dan bahasa pemrograman *python*.
3. Bab 3 merupakan perancangan alat Sistem Peringatan Dini Deteksi dan Pemadam Kebakaran Berbasis *Raspberry pi*.. Dalam bab ini dijelaskan pula komunikasi *serial* antara sensor dan *Raspberry pi* maupun komunikasi *serial* antara *Raspberry pi* dengan modem *GSM Huawei E173*, dan *webcam USB*.
4. Bab 4 berisi hasil pengujian dan analisis dari Sistem Peringatan Dini Deteksi dan Pemadam Kebakaran Berbasis *Raspberry pi*.
5. Bab 5 berisi kesimpulan dan saran dari studi yang telah dikerjakan.

1.6 Relevansi

Manfaat yang diharapkan dari pengerjaan penelitian ini adalah :

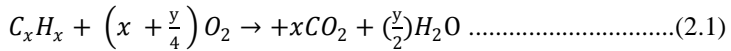
1. Memberikan kemudahan bagi pengguna untuk mengidentifikasi dini kebakaran dan penanggulangan yang lebih cepat.
2. Memberi kemudahan bagi pengguna *smartphone* atau *PC* yang terkoneksi internet untuk memonitor ruangan.
3. Dapat dijadikan dasar pengembangan tentang aplikasi sistem berbasis internet, misalnya untuk digunakan pada *smart car*, *Telerobotik*, *Telemedicine*, cuaca *online* dan sebagainya.

Halaman ini sengaja dikosongkan

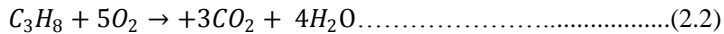
BAB II DASAR TEORI

2.1 Identifikasi Kebakaran

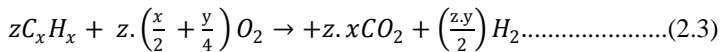
Proses terjadinya api membutuhkan tiga unsur pada waktu yang bersamaan, yaitu tersedianya bahan bakar, oksigen dan sumber penyalaaan. Ketiga unsur ini harus dalam perbandingan yang sesuai dan keadaan yang tepat agar api dapat menyala, biasanya api membutuhkan 15-21% oksigen untuk menyala dan terbakar. Pada umumnya persamaan kimia untuk pembakaran Hidrokarbon dengan oksigen adalah:



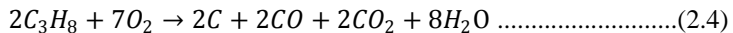
Contoh, persamaan kimia pembakaran Propana:



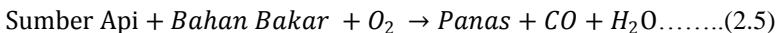
Secara umum, persamaan kimia untuk pembakaran Hidrokarbon yang tidak sempurna (kekurangan oksigen) adalah sebagai berikut:



Contoh, persamaan kimia pembakaran Propana yang tidak sempurna:



Jadi secara umum terjadinya proses suatu kebakaran adalah sebagai berikut:



Dari berbagai polutan udara dalam ruangan, gas *CO* dan gas *CO2* merupakan unsur polutan terbesar dan paling berbahaya. Paparan udara dengan gas *CO* dapat mengakibatkan keracunan sistem saraf pusat dan jantung. Keracunan ini terjadi jika paparan gas *CO* melampaui batas dari yang bisa di toleransi tubuh, yaitu lebih dari 250 ppm [2].

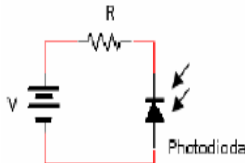
2.2 Infrared Flame Sensor 5 kanal

Infrared Flame Sensor 5 kanal adalah sensor untuk mendeteksi api dengan memanfaatkan Photodiode yang peka terhadap gelombang *IR* dengan rentang 700-1100nm yang biasanya berasal dari sumber api. Sangat bisa diaplikasikan dengan *Raspberry pi* atau *minsys* lainnya yang ingin merancang robot pemadam api (*fire fighter robot*). Photodiode mempunyai resistansi yang rendah pada kondisi *forward* bias, namun photodiode ini dapat dimanfaatkan dalam keadaan *reverse* bias, dimana resistansi dari photodiode akan turun seiring dengan intensitas cahaya yang masuk. Photodiode sambungan *PN* yang secara khusus dirancang untuk mendeteksi cahaya. Energi cahaya lewat melalui lensa yang mengekspos sambungan. Photodiode dirancang beroperasi pada mode bias mundur. Pada alat ini arus bocor bias mundur meningkat dengan peningkatan level cahaya. Harga arus umumnya adalah dalam rentang mikroampere. Photodiode mempunyai waktu respon yang sangat cepat terhadap cahaya. Rangkaian dasar photodiode ditunjukkan pada gambar 2.6. Photodiode dihubungkan seri dengan sebuah *R* dan dicatu dengan sumber tegangan *DC*. Arus balik akan bertambah bila sebuah cahaya jatuh pada pertemuan *PN* photodiode dan arus balik (*I*) akan menjadi sangat kecil bila pertemuan *PN* photodiode tidak terdapat cahaya yang jatuh padanya. Arus yang mengalir pada kondisi gelap disebut *dark current*. Sedangkan resistansinya ditentukan dengan hukum *Ohm* sebagai berikut [3]:

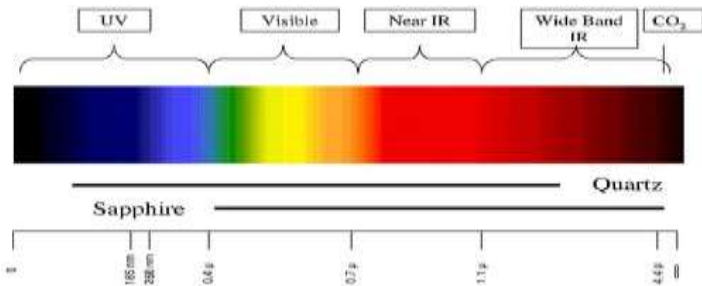
$$R_r = \frac{V_{CC} - V_{photo}}{I_r} \dots\dots\dots(2.6)$$

Keterangan:

- R_r* = Hambatan pada *R* (Ω)
- V_{CC}* = Sumber tegangan *DC* (*V*)
- V_{photo}* = Tegangan pada photodiode (*V*)
- I_r* = Kuat arus *reverse* (*A*)



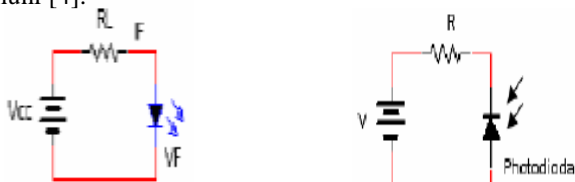
Gambar 2.1 Rangkaian Dasar Photodiode



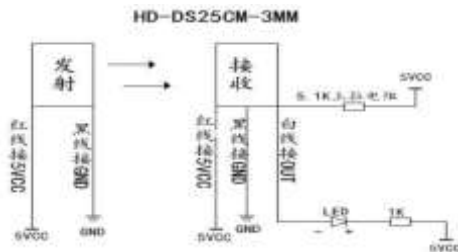
Gambar 2.2 Spektrum gelombang *Infra Red*

2.3 *Photoelectric IR Break Beam Sensor*

Infra merah (IR) break-beam sensor adalah sensor sederhana untuk mendeteksi gerakan. Sensor bekerja dengan memiliki sisi *emitor* yang mengirimkan seberkas cahaya *IR* yang tak terlihat oleh manusia, lalu cahaya *IR* melintasi ke Receiver yang sensitif terhadap cahaya *IR*. Ketika ada yang lewat di antara keduanya, dan tidak *transparan* terhadap *IR*, maka akan tertutup dan receiver akan memberi informasi kepada penghuni [4].

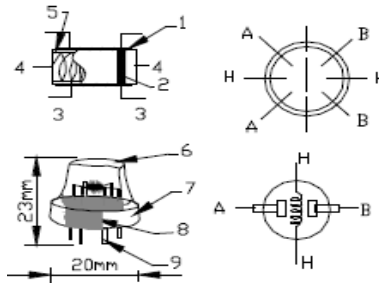


Gambar 2.3 Rangkaian dasar *Photoelectric IR Break Beam Sensor*



Gambar 2.4 Skematik *Photoelectric IR Break Beam Sensor*

2.4 MQ-5

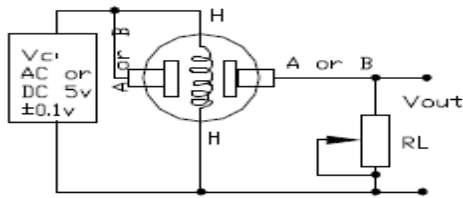


Gambar 2.5 Konfigurasi Rangkaian MQ-5

Tabel 2.1 Konfigurasi MQ-5

No. Pin	Nama	Material
1	Lapisan Pendeteksi	SnO ₂
2	Elektroda	Au
3	Kawat Elektroda	Pt
4	Koil Pemanas	Campuran Ni-Cr
5	Tabung Keramik	Al ₂ O ₃
6	Jaring Anti Ledakan	Stainless steel gauze
7	Cincin Penjepit	Plat Tembaga
8	Dasar Resin	Bakelite
9	Pin konektor	Tembaga

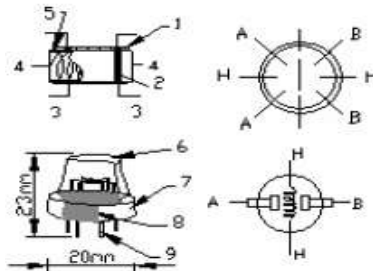
Sensor MQ-5 adalah sensor *universal* yang mampu mendeteksi berbagai jenis gas seperti *hidrogen (H₂)*, *karbon monoksida (CO)*, *metana (CH₄)*, *etanol (CH₃CH₂OH)*, *propana (C₃H₈)*, *butana (C₄H₁₀)*, dan gas *hidrokarbon* lainnya. Sensor ini memang tidak sesensitif sensor gas lainnya, namun kelebihanannya adalah sifatnya yang *universal* yang mampu mendeteksi tipe gas yang lebih luas. Secara umum struktur dari sensor MQ-5 seperti pada gambar 2.6. Sensor ini terdiri dari lapisan semikonduktor logam oksida yang terbentuk di atas sebuah *substrat alumina* pada sebuah *sensing chip* bersama dengan sebuah pemanas yang terintegrasi. Sensor ini bekerja berdasarkan perubahan resistansinya terhadap gas-gas tertentu termasuk asap.



Gambar 2.6 Prinsip kerja *MQ-5*

Sensor ini mendeteksi konsentrasi gas *LPG* dari udara dan *ouput* membaca sebagai tegangan analog. Sensor dapat mengukur konsentrasi gas *LPG* dari 200 sampai 10000 *ppm*. Sensor ini dapat beroperasi pada suhu dari -20 sampai 50 °C dan mengkonsumsi kurang dari 150 mA pada 5V. Heater pada sensor ini berfungsi sebagai pemicu sensor untuk dapat mendeteksi target gas yang diharapkan setelah diberi tegangan 5V. Sehingga dua elemen logam (A dan B) akan bekerja. Dan diantara dua logam tersebut, terdapat ruang yang jaraknya telah ditentukan. Apabila ada sensor mendeteksi gas maka kerapatan ruang yang terdapat antara logam A dan B akan membesar / mengecil. Saat tahanan semakin kecil maka arus akan mengalir dari A ke B sehingga output tegangan sensor akan membesar [5].

2.5 *MQ-7*

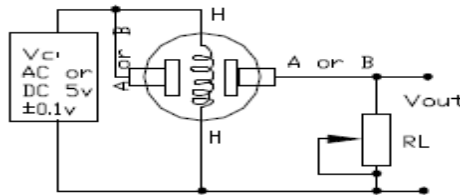


Gambar 2.7 Konfigurasi Rangkaian *MQ-7*

Tabel 2.2 Konfigurasi *MQ-7*

No. Pin	Nama	Material
1	Lapisan Pendeteksi	SnO ₂
2	Elektroda	Au
3	Kawat Elektroda	Pt
4	Koil Pemanas	Campuran Ni-Cr
5	Tabung Keramik	Al ₂ O ₃
6	Jaring Anti Ledakan	Stainless steel gauze
7	Cincin Penjepit	Plat Tembaga
8	Dasar Resin	Bakelite
9	Pin konektor	Tembaga

MQ-7 merupakan sensor gas yang digunakan dalam peralatan untuk mendeteksi gas karbon monoksida (*CO*) dalam kehidupan sehari-hari, industri, atau mobil. Fitur dari sensor gas *MQ-7* ini adalah mempunyai sensitivitas yang tinggi terhadap karbon monoksida (*CO*), stabil, dan berumur panjang. Sensor ini menggunakan catu daya heater : 5V AC/DC dan menggunakan catu daya rangkaian : 5VDC, jarak pengukuran : 20 – 2000 ppm untuk mampu mengukur gas karbon monoksida. Secara umum struktur dari sensor *MQ-5* seperti pada gambar 2.8.

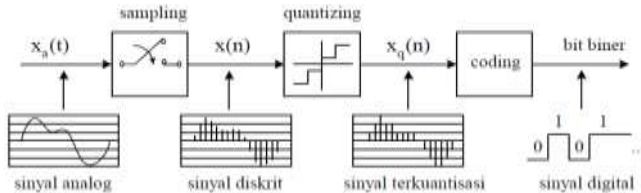


Gambar 2.8 Prinsip kerja *MQ-7*

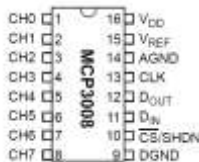
Heater pada sensor ini berfungsi sebagai pemicu sensor untuk dapat mendeteksi target gas yang diharapkan setelah diberi tegangan 5V. Sehingga dua elemen logam (A dan B) akan bekerja. Dan diantara dua logam tersebut, terdapat ruang yang jaraknya telah ditentukan. Apabila ada sensor mendeteksi gas maka kerapatan ruang yang terdapat antara logam A dan B akan membesar / mengecil. Saat tahanan semakin kecil maka arus akan mengalir dari A ke B sehingga output tegangan sensor akan membesar [6].

2.6 Analog to Digital Converter (ADC)

ADC banyak digunakan sebagai perantara antara sensor yang kebanyakan analog dengan sistem *Mikrokontroler*. Proses konversi tersebut dapat digambarkan sebagai proses tiga (3) langkah seperti diilustrasikan pada Gambar 2.9. ADC di penelitian ini menggunakan *MCP3008*, ADC ini bisa digunakan dengan *Raspberry pi* dengan *analog input* dari sensor. ADC ini terdiri dari 8 saluran dengan 10 bit data. *MCP3008* terhubung ke *Raspberry Pi* menggunakan koneksi *SPI*. Kita bisa menggunakan *Hardware bus SPI*, atau empat *pin GPIO* dan *Software SPI* untuk komunikasi serial dengan *MCP3008*. *Software SPI* sedikit lebih *fleksibel* karena bisa bekerja dengan pin pada *Raspberry Pi*, sedangkan *hardware SPI* sedikit lebih cepat tapi kurang *fleksibel* karena hanya bekerja dengan pin tertentu [7].



Gambar 2.9 Proses ADC



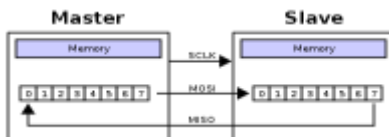
Gambar 2.10 MCP3008

Koneksi *MCP3008* ke *Raspberry pi* dengan *Hardware SPI*:

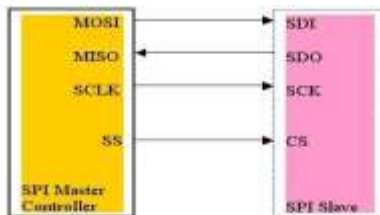
1. *MCP3008 VDD* ke *Raspberry Pi 3.3V*
2. *MCP3008 VREF* ke *Raspberry Pi 3.3V*
3. *MCP3008 AGND* ke *Raspberry Pi GND*
4. *MCP3008 DGND* ke *Raspberry Pi GND*
5. *MCP3008 CLK* ke *Raspberry Pi SCLK*
6. *MCP3008 DOUT* ke *Raspberry Pi MISO*
7. *MCP3008 DIN* ke *Raspberry Pi MOSI*
8. *MCP3008 CS/SHDN* ke *Raspberry Pi CE0*

2.7 Serial Peripheral Interface (SPI)

Serial Peripheral Interface (SPI) adalah salah satu *protocol* komunikasi serial *shynchronous* yang di-*develop* oleh Motorola. Dalam koneksi SPI, *device* yang terhubung satu sama lain akan bersifat *Full Duplex*, yaitu ada *device* yang bertindak sebagai *Master* dan *Slave*. *Master device* adalah perangkat yang memulai sambungan dengan cara menginialisasi SPI *address* dari *slave device*. Lalu *master* dan *slave* dapat mengirim atau menerima data. Hal ini sudah disebutkan sebelumnya bahwa komunikasi *full duplex* yang artinya *master* dan *slave* dapat menerima ataupun mengirim data. *Slave device* dapat menerima atau mengirim data dalam waktu yang bersamaan, itulah yang disebut *Full Duplex*. SPI beroperasi berdasarkan *shift register* baik *master device* maupun *slave device*, keduanya akan mempunyai 8 bit *shift register*. Namun tergantung dari berbagai macam arsitektur mikrokontroler, ada yang bisa memiliki 10 bit ataupun 12 bit *shift register*. Untuk memulai komunikasi, *bus master* melakukan konfigurasi *clock*, dengan catatan frekuensi atau kecepatan transfer data antara SPI *master device* dan *slave device* harus sama, biasanya bisa mencapai beberapa MHz. Master akan memilih perangkat *slave* dengan mengeluarkan logika 0, lalu master akan menunggu proses yang telah dijadwalkan di master itu sendiri seperti urutan intrupsi *timer*, konversi analog ke digital (ADC), dll. Lalu setelah periode itu selesai master akan mengeluarkan *clock* yang pertanda akan dimulainya proses komunikasi Serial [8].

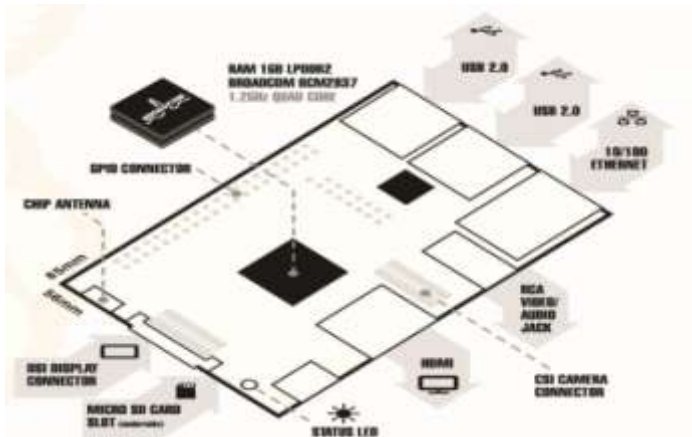


Gambar 2.11 Setup Hardware SPI



Gambar 2.12 Single master diagram interface SPI

2.8 Raspberry Pi 3 Model B



Gambar 2.13 Raspberry Pi 3 Model B

Raspberry Pi 3 model B adalah suatu perangkat mini *computer* berukuran sebesar kartu kredit. *Raspberry Pi 3* model B memiliki sistem *Quad Core Broadcom BCM2837 64-bit ARMv8* dengan kecepatan *processor 1.2 GHz*, serta *BCM43143* yang mensupport *WIFI*. Sistem operasinya ditanam pada sebuah *SD Flash Card*, yang menjadikannya sangat mudah untuk diganti dan ditukar. Termasuk *built-in hard disk* atau *solid-state drive*, tetapi menggunakan kartu *SD* untuk *booting* dan penyimpanan jangka panjang, Serta telah diuji sebagai multimedia player dengan kemampuan streaming, sebagai perangkat *game machine*, *internet browsing* dan sebagai *mainboard* pengembangan *hardware*. Hal tersebut memungkinkan perangkat ini digunakan sebagai perangkat pendidikan bagi orang-orang dari segala usia dan tingkat keterampilan [9].

Berikut *spesifikasi* dari *Raspberry Pi 3* model B:

1. *Processor*:

- *1.2GHz 64-bit quad-core ARMv8 CPU*
- *802.11n Wireless LAN*
- *Bluetooth 4.1*
- *Bluetooth Low Energy (BLE)*
- *VideoCore IV 3D graphics core*

- 2 *Memory:*
 - *1GB RAM*
- 3 *Sistem Operasi:*
 - *Booting* di kartu Micro SD, Sistem Operasi Linux atau Windows 10 IOT.
- 4 *Power Supply:*
 - *Micro USB 5.1 V, 2.5 A.*
- 5 *Audio Output*
 - *4 USB Ports*
 - *Jack 3.5mm audio output*
 - *HDMI Ports*
- 6 *Video Output*
 - *Composite Video RCA (PAL and NTSC)*
 - *HDMI Ports*
- 7 *GPIO*
 - *40-pin 2.54 mm (100 mil) expansion header: 2x20 strip*
 - *Terdiri dari 27 pin GPIO, +3.3 V, +5 V and GND.*
 - *Ethernet: 10/100 BaseT Ethernet socket*
- 8 *Camera : 15-pin MIPI Camera Serial Interface (CSI-2)*
- 9 *Display : Display Serial Interface (DSI)*
- 10 *Memory Card Slot: Push/pull Micro SDIO*

GPIO (general purpose input output)

GPIO (general purpose input output) Raspberry Pi 3 model B adalah pin generic pada chip yang dapat dikontrol (diprogram) melalui perangkat lunak baik di konfigurasi sebagai pin input maupun pin *output*. Raspberry Pi 3 model B *GPIO* memiliki 26 pin dengan ukuran 2,54 mm. konektor *GPIO* memiliki fitur-fitur diantaranya:

- 1 Pin antarmuka *I2C* yang memungkinkan untuk menghubungkan modul hardware dengan hanya dua pin control.
- 2 *SPI* antarmuka, memiliki konsep mirip dengan *I2C* tetapi dengan standar yang berbeda.
- 3 Serial *Rx* dan *Tx*, pin untuk berkomunikasi dengan perangkat serial.
- 4 Pin *PWM (Pulse Width Modulation)* untuk *control* daya.
- 5 Pin *PPM (Pulse Position Modulation)* untuk mengendalikan motor *servo*.

Raspberry Pi 3 GPIO Header

Pin	NAME	NAME	Pin
01	3.3v DC Power	DC Power 5v	02
03	GPIO2 (SDA1, I2C)	DC Power 5v	04
05	GPIO3 (SCL1, I2C)	Ground	06
07	GPIO4 (GPIO_CLOCK)	(TRIG) GPIO14	08
09	Ground	(RXD0) GPIO15	10
11	GPIO17 (GPIO_GEN0)	(SPM1_GEN1) GPIO18	12
13	GPIO27 (GPIO_GEN0)	Ground	14
15	GPIO22 (GPIO_GEN0)	(GPIO_GEN4) GPIO23	16
17	3.3v DC Power	(GPIO_GEN0) GPIO24	18
19	GPIO10 (SPI_MOSI)	Ground	20
21	GPIO9 (SPI_MISO)	(GPIO_GEN0) GPIO25	22
23	GPIO11 (SPI_CLK)	(SPI_CS_N) GPIO26	24
25	Ground	(SPI_CS_N) GPIO27	26
27	ID_SD (I2C ID EEPROM)	(I2C ID EEPROM) ID_SC	28
29	GPIO15	Ground	30
31	GPIO16	GPIO12	32
33	GPIO13	Ground	34
35	GPIO19	GPIO16	36
37	GPIO28	GPIO20	38
39	Ground	GPIO21	40

www.element14.com/RaspberryPi

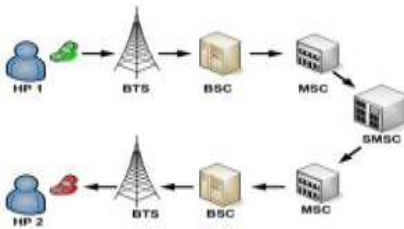
Gambar 2.14 Pin GPIO Raspberry Pi 3 model B

2.9 SMS Gateway

SMS Gateway merupakan aplikasi yang digunakan untuk mengirim maupun menerima SMS. Dalam Tugas Akhir ini digunakan *Modem USB GSM Huawei E173* yang dikomunikasikan serial dengan *Raspberry Pi 3 model B* yang diaplikasikan sebagai penyampaian informasi adanya indikasi kebakaran berintensitas kecil kepada user.



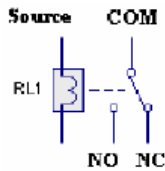
Gambar 2.15 SMS Gateway



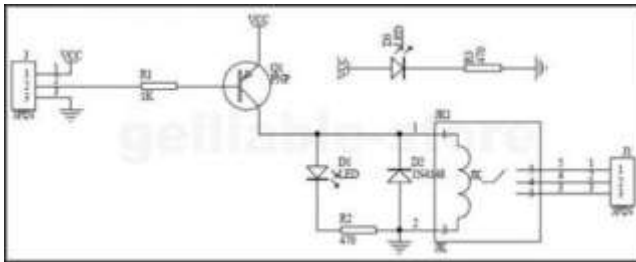
Gambar 2.16 Proses SMS

Ketika pengguna mengirim SMS, maka pesan dikirim ke MSC (*Mobile Switching Center*) melalui jaringan seluler yang tersedia yang meliputi tower *BTS* (*Base Transceiver Station*) yang sedang meng-handle komunikasi pengguna, lalu ke *BSC* (*Base Station Controller*), kemudian sampai ke *MSC* (*Mobile Switching Center*). *MSC* kemudian mem-forward lagi SMS ke *SMSC* (*Short Message Service Center*) untuk disimpan. *SMSC* (*Short Message Service Center*) kemudian mengecek (lewat *HLR* – *Home Location Register*) untuk mengetahui apakah *handphone* tujuan sedang aktif dan dimanakah *handphone* tujuan tersebut. Jika *handphone* sedang tidak aktif maka pesan tetap disimpan di *SMSC* itu sendiri, menunggu *MSC* memberitahukan bahwa *handphone* sudah aktif kembali untuk kemudian SMS dikirim dengan batas maksimum waktu tunggu yaitu *validity period* dari pesan SMS itu sendiri. Jika *handphone* tujuan aktif maka pesan disampaikan *MSC* lewat jaringan yang sedang meng-handle penerima (*BSC* dan *BTS*). Sebenarnya, di dalam kebanyakan *handphone* dan *GSM/CDMA modem* terdapat suatu komponen *wireless modem/engine* yang dapat diperintah antara lain untuk mengirim suatu pesan SMS dengan protokol tertentu. Standar perintah tersebut dikenal sebagai *AT-Command*, sedangkan protokolnya disebut sebagai *PDU* (*Protokol Data Unit*). Melalui *AT-Command* dan *PDU* inilah kita dapat membuat komputer/mikrokontroler mengirim/menerima SMS secara otomatis berdasarkan program yang kita buat [10].

2.10 Relay



Gambar 2.17 simbol Relay

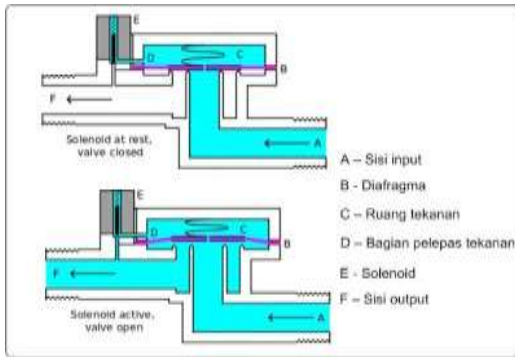


Gambar 2.18 Skematik Relay Driver

Relay merupakan komponen elektronika yang dapat mengimplementasikan logika *switching (On/Off)*. Relay adalah saklar elektronik yang digerakan oleh arus listrik sehingga dapat dikendalikan dari rangkaian elektronik lainnya [11]. Dalam penelitian ini Relay dikendalikan oleh Raspberry Pi 3 model B melalui *Pin GPIO* untuk mengaktifkan atau mematikan *Water Solenoid Valve*.

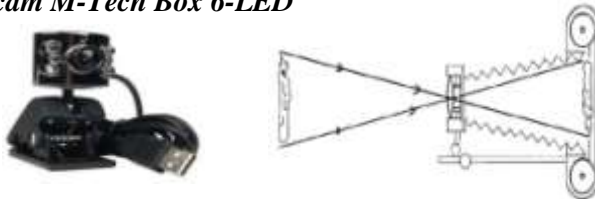
2.11 Water Solenoid Valve

Solenoid valve akan bekerja bila kumparan/coil mendapatkan tegangan arus listrik yang sesuai dengan tegangan kerja (kebanyakan tegangan kerja *solenoid valve* adalah 100/200VAC dan 12/24VDC). Dan sebuah pin akan tertarik karena gaya magnet yang dihasilkan dari kumparan selenoida tersebut. Dan saat pin tersebut ditarik naik maka fluida akan mengalir dari ruang C menuju ke bagian D dengan cepat. Sehingga tekanan di ruang C turun dan tekanan fluida yang masuk mengangkat diafragma. Sehingga katup utama terbuka dan fluida mengalir langsung dari A ke F [12].



Gambar 2.19 Water Solenoid Valve

2.12 Webcam M-Tech Box 6-LED



Gambar 2.20 Webcam M-Tech Box 6-LED

Sebuah *web camera* yang sederhana terdiri dari sebuah lensa standar, dipasang di sebuah papan sirkuit untuk menangkap sinyal gambar; *casing (cover)*, termasuk *casing* depan dan *casing* samping untuk menutupi lensa standar dan memiliki sebuah lubang lensa di *casing* depan yang berguna untuk memasukkan gambar; kabel *support*, yang dibuat dari bahan yang fleksibel, salah satu ujungnya dihubungkan dengan papan sirkuit dan ujung satu lagi memiliki *connector*, kabel ini dikontrol untuk menyesuaikan ketinggian, arah dan sudut pandang *web camera*. Sebuah *web camera* biasanya dilengkapi dengan *software*, *software* ini mengambil gambar-gambar dari kamera digital secara terus menerus ataupun dalam interval waktu tertentu dan menyiarkannya melalui koneksi *internet*. Ada beberapa metode penyiaran, metode yang paling umum adalah *hardware* mengubah gambar ke dalam bentuk file *JPG* dan menguploadnya ke *web server* menggunakan *File Transfer Protocol (FTP)* [13].

2.13 Buzzer



Gambar 2.21 Buzzer

Dalam penelitian ini jika ada indikasi Kebakaran dalam intensitas kecil digunakan *Buzzer* sebagai alarm untuk memberi informasi kepada orang yang berada disekitar rumah. *Buzzer* adalah komponen elektronika yang berfungsi untuk mengubah getaran listrik menjadi getaran suara. jadi *buzzer* terdiri dari kumparan yang terpasang pada diafragma dan kemudian kumparan tersebut dialiri arus sehingga menjadi elektromagnet, kumparan tadi akan tertarik ke dalam atau keluar, tergantung dari arah arus dan polaritas magnetnya, karena kumparan dipasang pada diafragma maka setiap gerakan kumparan akan menggerakkan diafragma secara bolak-balik sehingga membuat udara bergetar yang akan menghasilkan suara.

2.14 Program *Python IDLE 2*

Dalam Tugas Akhir ini digunakan bahasa pemrograman *Python IDLE 2*. Beberapa kelebihan bahasa *python* antara lain: dengan sintaks yang simpel, sederhana dan dokumentasi yang lengkap membuat *Python* menjadi bahasa pemrograman yang mudah untuk dipelajari, dengan pendekatan pemrograman berorientasi objek, *Python* adalah bahasa yang powerful. Belum lagi dengan fitur yang melimpah, *library* yang luas, komunitas yang besar dan masih banyak hal lainnya. Bahasa yang *efektif* adalah bahasa yang dapat disampaikan dan ditangkap dengan cepat. Dengan kelebihan tersebut programmer akan lebih fokus pada pengembangan aplikasi. Menggunakan *IDLE 2* karena librarinya lebih lengkap dibanding dengan *IDLE 3* [13].



Gambar 2.22 Program *Python IDLE 2*

2.15 Memasang *wvdial* untuk melakukan *dial-up modem* di *Raspberry Pi*

Langkah pertama dalam melakukan konfigurasi *modem GSM* di *Raspi* adalah menginstall *wvdial*. *Wvdial* adalah program pembantu yang terdapat di berbagai distribusi *Linux* yang fungsinya untuk membantu kita melakukan panggilan data modem ke *operator* sehingga kita bisa terhubung ke *internet*. Berikut konfigurasi dari *wvdial*:

```
[Dialer telkomsel]
Init1 = ATZ
Init2 = ATQ0 V1 E1 S0=0 &C1 &D2 +FCLASS=0
Init3 = AT+CGDCONT=1,"IP","internet"
Stupid Mode = 1
Modem Type = Analog Modem
ISDN = 0
Phone = *99#
Modem = /dev/gsmmodem
Username = {}
Password = {}
Baud = 460800
```

2.16 SMS Gateway Menggunakan *Gammu*

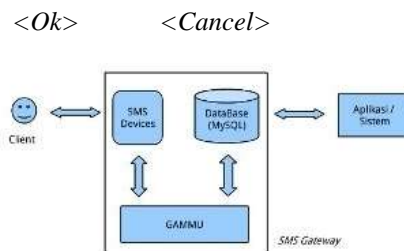
SMS Gateway adalah sebuah gerbang yang menghubungkan antara komputer dengan client melalui *SMS*, jadi secara garis besar, *SMS Gateway* dapat digambarkan seperti gambar di bawah. adalah sebuah aplikasi *cross-platform* yang digunakan untuk menjembatani/

mengomunikasikan antara database *SMS Gateway* dengan *sms devices*. Aplikasi *Gammu* berupa *daemon* yang berjalan secara *background*. Setiap saat, *gammu* memonitor *sms devices* dan *database sms gateway*. Saat ada *sms* masuk ke *sms devices*, maka *gammu* langsung memindahkannya ke dalam *inbox* dalam *database sms gateway*. Sebaliknya saat Aplikasi Pengirim *SMS* memasukkan *sms* ke dalam *outbox* dalam *database sms gateway*, maka *gammu* mengirimkannya melalui *sms devices*, dan memindahkan *sms* ke *sentitem* dalam *database*. Berikut konfigurasi dari *Gammu*:

```

P Port          (/dev/ttyUSB0)
C Connection    (at19200)
M Model        ()
D Synchronize time (yes)
F Log file     ()
O Log format   (nothing)
L Use locking  ()
G Gammu localisation ()
H Help
S Save

```



Gambar 2.23 Konfigurasi *Gammu*

2.17 *OpenCV*

OpenCV (*Open Computer Vision*) yang merupakan sebuah *API* (*Application Programming Interface*) *Library* yang sudah sangat familiar pada Pengolahan Citra *Computer Vision*. *OpenCV* sendiri direlease dalam *lisensi BSD* dan bebas digunakan untuk keperluan akademik maupun komersial. Produk ini mendukung *interface C/C++*,

python dan *java* serta bisa berjalan diberbagai *platform* seperti *Windows*, *Linux*, *Mac OS*, *iOS* dan *Android*. *Computer Vision* itu sendiri adalah salah satu cabang dari Bidang Ilmu Pengolahan Citra atau dikenal sebagai *Image Processing* yang memungkinkan komputer dapat melihat seperti manusia. Dengan *vision* tersebut komputer dapat mengambil keputusan, melakukan aksi, dan mengenali terhadap suatu objek. Beberapa pengembangan dan *project-project* dari produk ini adalah *Face Recognition*, *Face Detection*, *Face/Object Tracking*, *Road Tracking*, dll [14].

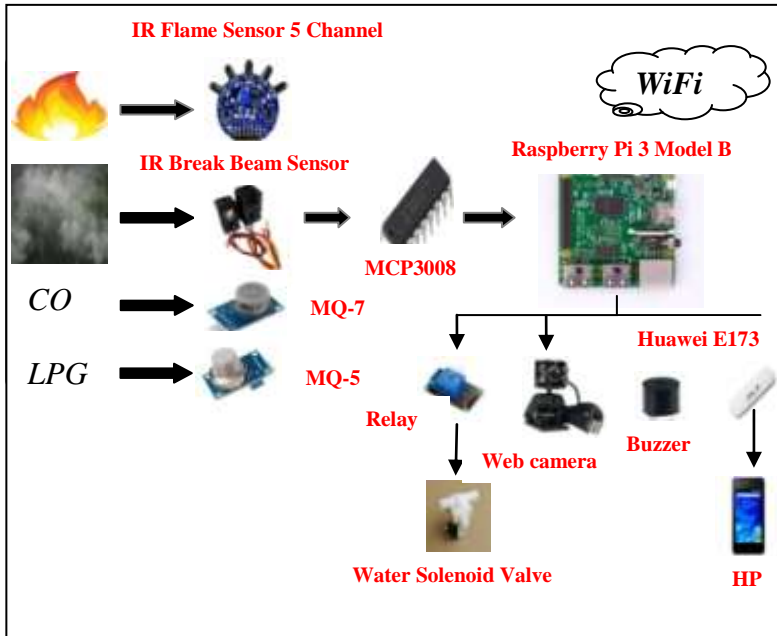
2.18 SSMTP (Software Simple Mail Transfer Protokol)

Simple Mail Transfer Protocol (SMTP) adalah salah satu *protokol* yang umum digunakan untuk pengiriman surat elektronik di *Internet*. Protokol ini digunakan untuk mengirimkan data dari komputer pengirim surat elektronik ke *server* surat elektronik penerima. *Protokol* ini timbul karena desain sistem surat elektronik yang mengharuskan adanya *server* surat elektronik yang menampung sementara, sampai surat elektronik diambil oleh penerima yang berhak.

BAB 3 PERANCANGAN SISTEM

3.1 Konfigurasi Sistem

Berikut adalah gambar dari Sistem Peringatan Dini Deteksi dan Pemadam Kebakaran Berbasis Raspberry Pi secara garis besar:



Gambar 3.1 Konfigurasi Sistem

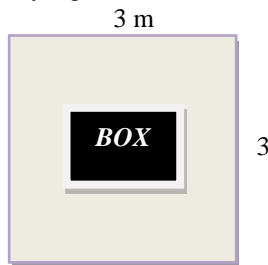
Konfigurasi sistem ini meliputi beberapa bagian antara lain:

1. Perencanaan teknis peletakan sensor supaya bisa mengcover seluruh ruangan laboratorium.
2. Perencanaan *Hardware*, terdiri dari *wiring flame* sensor 5 kanal, *Photoelectric IR Break Beam Sensor*, *MQ-5*, dan *MQ-7* ke *MCP3008*, komunikasi *serial MCP3008*, *Modem Huawei E173*, dan *Webcame USB* ke *Raspberry Pi*, *wiring relay* ke *Water solenoid Valve* serta *Buzzer* ke *Raspberry Pi*.

3. Perencanaan *Software*, terdiri dari pemrograman python untuk pembacaan sensor, Pengiriman *SMS* ke penghuni, pengambilan gambar ruangan dan mengirimkan ke email penghuni, mengontrol *relay* dan *buzzer* berdasarkan kondisi ruangan yang terbakar.

3.2 Perancangan Teknis

Dalam perancangan teknis ini sistem diletakkan dalam suatu *box* yang berisi *flame sensor* 5 kanal, *Photoelectric IR Break Beam sensor*, *MQ-7*, *MQ-5*, *Raspberry pi* 3 model B, *modem Huawei E173*, dan *Buzzer*. Berdasarkan pertimbangan kemampuan dalam pembacaan nilai parameter tiap jenis sensor yang dipakai maka dilakukan peletakkan sistem pada suatu ruangan yang berukuran 3 x 3 m sebagai berikut:



Gambar 3.2 Perancangan Teknis

Untuk pemasangan sistem ini adalah pada ruangan yang memiliki ketinggian maksimum 2.5 m hal ini didasarkan kemampuan sensor api 5 kanal dalam menjangkau sumber api dalam ruangan. Pemasangan *Box* diletakkan diatas yaitu menempel di plafon dengan pertimbangan *flame sensor* 5 kanal bisa menjangkau sumber api, dan *MQ-5*, *MQ-7*, *Photoelectric IR Break Beam sensor* dengan pertimbangan sifat gas *LPG*, *CO* dan asap merambat ke atas. Untuk modem *Huawei E173* karena *USB* maka peletakkannya di *USB Raspberry pi*. Peletakkan *webcam* diluar kotak dengan pertimbangan supaya *webcam* bisa memonitor seluruh ruangan. *Buzzer* diletakan di *box* dengan pertimbangan berat *buzzer* yang ringan, pengkabelan yang mudah dan suara diletakkan dimanapun tetap terdengar oleh orang sekitar. Pemasangan *water solenoid valve* diletakkan terpisah dari *box* dengan pertimbangan bagaimana *water solenoid valve* bisa mengalirkan air

ketika ada indikasi kebakaran dalam intensitas yang kecil, kita asumsikan disini diatas ruangan sudah memiliki tandon air.

3.3 Perancangan *Hardware*.

Sistem *Hardware* pada penelitian ini merupakan suatu sistem yang diterapkan pada proses akuisisi data dari sensor ke Raspberry pi, pemasangan Modem *USB Huawei E173*, *webcam*, *Buzzer*, serta *Relay* untuk mengendalikan *Water Solenoid Valve*. Berikut beberapa sub sistem dalam perancangan Tugas akhir ini yaitu:

1. Sistem akuisisi data pada tiap-tiap sensor yang kemudian akan diproses melalui *ADC MCP3008*.
2. Komunikasi serial Modem *Huawei E173* dan *webcam*.
3. Sistem output yang mengendalikan *Buzzer* dan *Water Solenoid Valve*.

3.3.1 *MCP3008*

ADC di penelitian ini menggunakan *MCP3008*, *ADC* ini bisa digunakan dengan Raspberry pi dengan *analog input* dari sensor. *ADC* ini terdiri dari 8 saluran dengan 10 bit data. *MCP3008* terhubung ke Raspberry Pi menggunakan koneksi *SPI*. kita bisa menggunakan *Hardware bus SPI*, atau empat *pin GPIO* dan *Software SPI* untuk komunikasi serial dengan *MCP3008*. Berikut pengkabelan *MCP3008* ke *Raspberry pi*:

VDD ke 3.3V	CLK ke CLK/Pin 23
VREF ke 3.3V	Dout ke MISO/Pin 21
AGND ke GND	Din ke MOSI/Pin19
DGND ke GND	CS/SHDN ke CE0/Pin 20

3.3.2 *Infrared Flame sensor 5 kanal*

Infrared Flame Sensor 5 kanal adalah sensor untuk mendeteksi api dengan memanfaatkan 5 buah *IR receiver* yang peka terhadap gelombang *IR* dengan rentang 700-1100nm yang berasal dari sumber api. Sangat bisa diaplikasikan dengan *Raspberry pi* atau *minsys* lainnya yang ingin merancang robot pendeteksi / pemadam api (*fire fighter robot*). Pada penelitian ini digunakan komunikasi *SPI (Serial Peripheral Interface)* antara sensor dengan *Raspberry pi* menggunakan *MCP3008*. Berikut adalah pengkabelan sensor api 5 *channel*:

Flame Sensor 5 Channel ke MCP3008:
Vcc ke 5V A3 ke CH 5
GND ke GND A4 ke CH 6
A1 ke CH 3 A5 ke CH 7
A2 ke CH 4

3.3.3 Photoelectric IR Break Beam Sensor

Dalam penelitian ini digunakan *Infra merah (IR) break-beam sensor* untuk mendeteksi adanya asap. Sensor bekerja dengan memiliki sisi emitor yang mengirimkan seberkas cahaya *IR* yang tak terlihat oleh manusia, lalu cahaya *IR* menuju ke *Receiver* yang sensitif terhadap cahaya *IR*. Ketika ada penghalang di antara keduanya, dan tidak transparan terhadap *IR*, maka 'beam tertutup' dan *Receiver* akan memberi tahu kita. Dibandingkan sensor *PIR*, break beams lebih cepat dan kontrol yang lebih baik jika kita ingin mendeteksi gerakan.

Wiring sensor ini sangat mudah. Pertama, kita harus menghidupkan pemancar. Hubungkan kabel hitam ke *ground* dan kabel merah langsung ke 3.3V atau 5V power. Ini akan menggunakan 9 mA dari 3.3V (daya rendah) dan 20 mA dari 5V. Selanjutnya menghubungkan *receiver*. Sambungkan kabel hitam ke *ground*, kabel merah ke 3.3V atau 5V (tingkat logika mana yang kita suka) dan kemudian kabel kuning ke input digital kita. Penerima adalah *open collector* yang berarti kita memang perlu *resistor pull up*. Kebanyakan *mikrokontroler* memiliki kemampuan untuk menghidupkan *resistor pull up* yang terpasang. Jika tidak, hubungkan resistor 1K antara kabel kuning penerima dan kabel merah. Berikut adalah pengkabelan *Infra merah (IR) break-beam sensor*:

Transmitter :

Kabel Merah Ke 5V
Kabel Hitam ke GND

Receiver:

Kabel Merah ke 5V
Kabel Hitam ke GND
Kabel Kuning ke GPIO 16

3.3.4 MQ-5

Sensor *MQ-5* adalah sensor *universal* yang mampu mendeteksi berbagai jenis gas seperti *hidrogen (H₂)*, *karbon monoksida (CO)*, *metana (CH₄)*, *etanol (CH₃CH₂OH)*, *propana (C₃H₈)*, *butana (C₄H₁₀)*, dan gas *hidrokarbon* lainnya. Sensor ini memang tidak sesensitif sensor gas lainnya, namun kelebihanannya adalah sifatnya yang

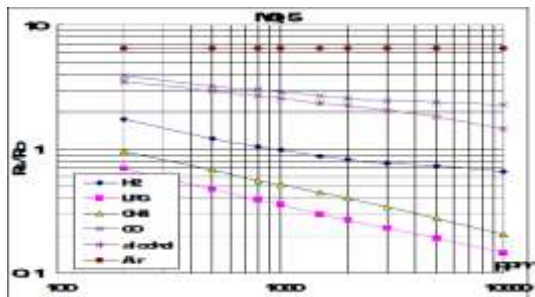
universal yang mampu mendeteksi tipe gas yang lebih luas. Dalam Tugas Akhir ini untuk mendeteksi gas *LPG* maka digunakan sensor *MQ-5* yang telah dikemas menjadi sebuah modul. Pada tugas akhir ini digunakan komunikasi *SPI (Serial Peripheral Interface)* antara sensor dengan *Raspberry pi* menggunakan *MCP3008*. Berikut adalah pengkabelan sensor *MQ-5*:

MQ-5 ke MCP3008:

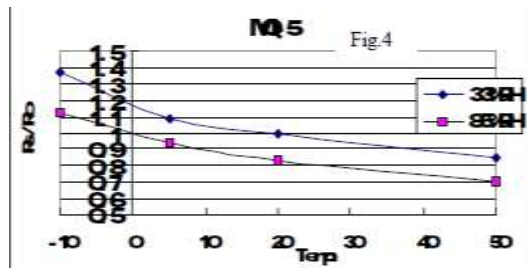
Vcc ke 5V

GND ke GND

A0 ke CH 2



Gambar 3.3 Karakteristik sensitifitas *MQ-5*



Gambar 3.4 Karakteristik temperatur dan kelembapan *MQ-5*

Keterangan:

R_s : Resistansi sensor pada konsentrasi tertentu

R_o : Resistansi sensor pada 1000 ppm H_2

Sensor dapat mengukur konsentrasi gas *LPG* dari 200 sampai 10.000 ppm. Sensor ini dapat beroperasi pada suhu dari -10 sampai 50 °C dan mengonsumsi kurang dari 150 mA pada 5V.

3.3.5 MQ-7

MQ-7 merupakan sensor gas yang digunakan dalam peralatan untuk mendeteksi gas karbon monoksida (CO) dalam kehidupan sehari-hari, industri, atau mobil. Fitur dari sensor gas MQ-7 ini adalah mempunyai sensitivitas yang tinggi terhadap karbon monoksida (CO), stabil, dan berumur panjang. Sensor ini menggunakan catu daya heater : 5V AC/DC dan menggunakan catu daya rangkaian : 5 VDC, jarak pengukuran : 20 – 2000 ppm untuk mampu mengukur gas karbon monoksida. Pada tugas akhir ini digunakan komunikasi SPI(Serial Peripheral Interface) antara sensor dengan Raspberry pi menggunakan MCP3008. Berikut gambar skematik pengkabelan sensor MQ-7:

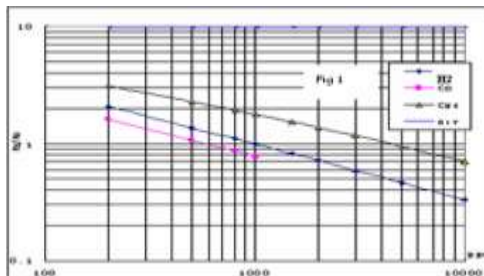
MQ-5 ke MCP3008:

Vcc ke 5V

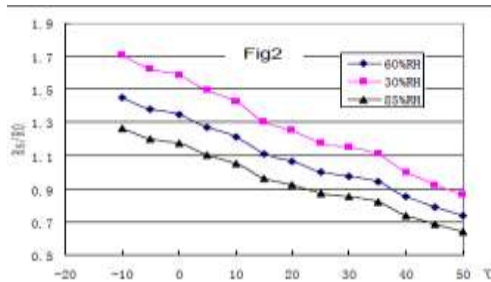
GND ke GND

A0 ke CH 1

Sensor ini mendeteksi konsentrasi gas CO hasil dari pembakaran di udara dan output membaca sebagai tegangan analog. Sensor dapat mengukur konsentrasi gas CO dari 20 sampai 2000 ppm. Sensor ini dapat beroperasi pada suhu dari -20 sampai 50 °C dan mengkonsumsi kurang dari 150 mA pada 5V.



Gambar 3.5 Karakteristik sensitifitas MQ-7



Gambar 3.6 Karakteristik temperatur dan kelembapan *MQ-7*

3.3.6 Komunikasi Serial Modem Huawei E173 dan Webcam

Dalam tugas akhir ini digunakan modem Huawei E173 untuk mengirim sms ke user ketika ada indikasi kebakaran dalam intensitas kecil, modem ini mempunyai USB jadi penggunaanya tinggal dicolokkan ke USB Raspberry pi. Untuk mengambil gambar kondisi ruangan maka digunakan webcam USB yang pemasangannya juga tinggal dicolokkan ke USB Raspberry pi. Hasil dari gambar Webcam ini dikirim ke email User ketika ada indikasi kebakaran.

3.3.7 Buzzer dan Water Solenoid Valve

Untuk memberi informasi kepada orang disekitar ruangan ketika ada indikasi kebakaran maka digunakan Buzzer sebagai alarm. Sebagai pemadam kebakaran maka digunakan water solenoid valve untuk mengalirkan air diasumsikan ruangan mempunyai tandon air. Untuk mengaktifkan atau mematikan water solenoid ini maka digunakan relay yang dikontrol oleh Rspberry pi. Berikut adalah pengkabelan Buzzer dan Water Solenoid valve.

Buzzer ke Pin GPIO Raspberry Pi 3 Model B:

Kaki (+) ke GPIO 20

Kaki (-) ke GND

Solenoid Valve ke Modul Relay:

Kaki 1 ke NC

Kaki 2 ke GND (milik 12V)

Relay ke Pin GPIO Raspberry Pi 3 Model B:

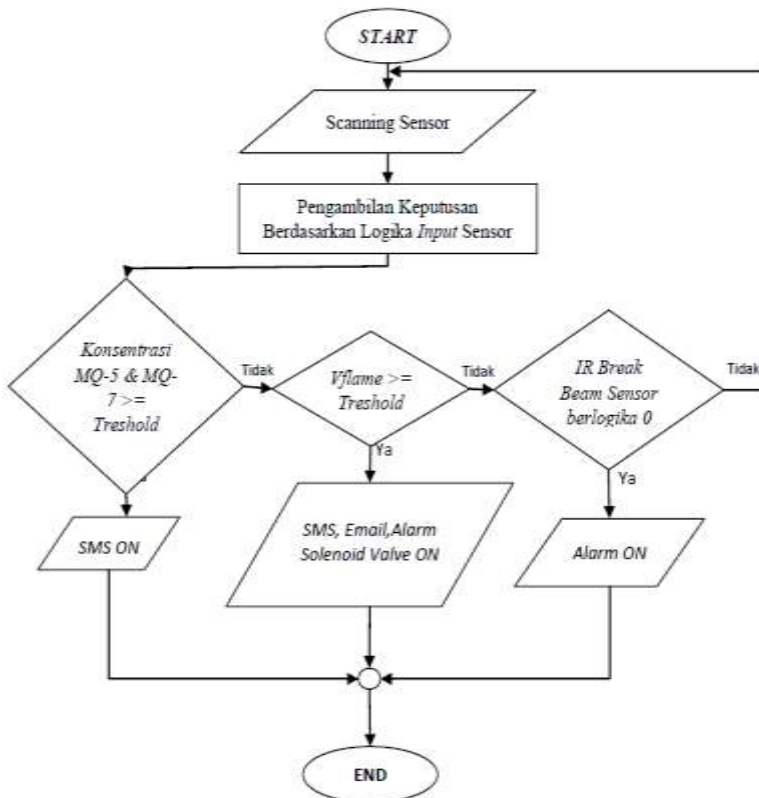
Vcc ke 5V

GND ke GND

IN ke GPIO 21
12V ke 12V

3.4 Perancangan *Software*

Pada bagian ini adalah menjelaskan tentang perancangan dan pembuatan perangkat lunak pembacaan serta pengambilan keputusan kondisi ruangan. Algoritma pengiriman *SMS*, pengiriman gambar kondisi ruangan ke *email User*, *alarm* serta pemadam pengaliran air. Berikut *flowchart* program dalam sistem ini secara garis besar:



Gambar 3.7 *Flowchart* Perancangan *Software*

3.4.1 Program Python IDLE 2 Untuk ADC Sensor

Program ini untuk pengambilan data dari *flame sensor 5 channel*, *IR MQ-5* dan *MQ-7* melalui *ADC MCP3008*, *ADC* ini menggunakan resolusi 10 *bit*(1024). Untuk program *python* ada di lampiran.

3.4.2 Program Python IDLE 2 Untuk SMS

Program ini untuk memberikan informasi kepada *user* ketika ada indikasi kebakaran yang masih dalam intensitas kecil. Berikut *script program python* untuk mengirim *sms* ke *user*:

```
#!/usr/bin/python
import time
import os
import serial
# Initalisasi sms
ser=serial.Serial('/dev/ttyUSB0',19200,timeout=1)
ser.write('ATZ\r')
time.sleep(0.5)
ser.write('AT+CMGF=1\r')
time.sleep(0.5)

while True:
    if sensor1_ppm>250:
        ser.write('AT+CMGS="085649262967"\r')
        time.sleep(1)
        ser.write('AWAS ADA API DIRUMAH ANDA')
        ser.write(chr(26))
        ser.write(chr(26))
```

3.4.3 Program Python IDLE 2 Untuk EMAIL

Program ini berupa pemberian informasi kepada *user* berupa *notifikasi email* yang berisi gambar kondisi ruangan yang diambil dari *webcam*. Untuk program *python* ada di lampiran.

3.4.4 Program Python IDLE 2 Untuk Webcam

Program ini berisi perintah ke *webcam* untuk mengambil gambar ketika ada indikasi kebakaran dalam intensitas kecil. Untuk *script program python* ada di lampiran.

3.4.5 Program Python IDLE 2 Untuk GPIO

Program ini untuk mengkondisikan pin *GPIO* dalam keadaan *HIGH* atau *LOW*. Dalam penelitian ini *GPIO* digunakan sebagai *input Photelectric IR Break Beam sensor* dan untuk mengontrol *Water Solenoid Valve* serta *alarm*. Berikut *script* program *python* untuk mengontrol *GPIO*:

```
import RPi.GPIO as GPIO
from time import sleep
GPIO.setmode(GPIO.BCM)
GPIO.setup(25, GPIO.IN)
GPIO.setup(24, GPIO.OUT)
try:
    while True:
        if GPIO.input(25) == 1 :
            print "Port 25 is 1/HIGH/True - LED ON"
            GPIO.output(24, 1)
        else:
            print "Port 25 is 0/LOW/False - LED OFF"
            GPIO.output(24, 0)
            sleep(0.1)
finally:
    GPIO.cleanup()
```

3.4.6 Algoritma Pengambilan Keputusan

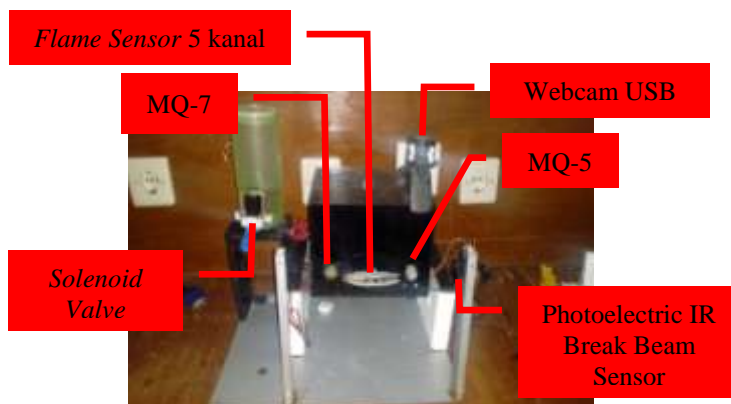
Untuk pengambilan keputusan didasarkan pada *set threshold* pembacaan masing-masing sensor yang didapatkan dari hasil percobaan dengan ketentuan sebagai berikut:

- Jika tegangan *flame sensor* 5 kanal lebih dari atau sama dengan 4V dan *IR Break Beam sensor* berlogika 0 maka sistem akan mengirim *SMS*, *Email* ke *USER*, membunyikan alarm serta mengaktifkan *solenoid valve* untuk mengalirkan air.
- Jika sensor gas *LPG* dan *CO* mendeteksi konsentrasi lebih dari atau sama dengan *threshold*, yaitu konsentrasi *LPG* > 1000 PPM dan *CO* > 200 PPM maka sistem akan mengirim *SMS* ke penghuni.

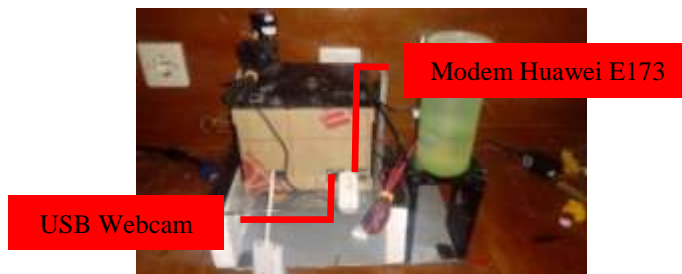
BAB 4 PENGUJIAN DAN ANALISA

Pada bab ini membahas tentang desain dan pengujian sistem. *Hardware* yang dibuat sesuai dengan bab perancangan sistem. Untuk menentukan *threshold* dari masing-masing sensor sesuai dengan algoritma pengambilan keputusan. Dengan dilakukan pengujian maka dapat diketahui kelemahan dari *Hardware* tersebut sehingga dapat segera diperbaiki bila diperlukan. Berikut bentuk desain dan tahap pengujian sistem yang dilakukan:

4.1 Bentuk Desain Sistem



Gambar 4.1 Desain sistem tampak depan



Gambar 4.2 Desain sistem tampak belakang

4.2 Pengujian pembacaan nilai ADC

1) Tujuan

Untuk mengetahui kemampuan pembacaan ADC multichannel MCP3008.

2) Parameter

Pengujian ini dilakukan dengan menggunakan potensiometer yang power suplynya dari 5V Raspberry pi untuk mengetahui respon channel ADC MCP3008.

3) Setting Pengujian

- 1 ADC yang digunakan adalah MCP3008 yang mempunyai resolusi 10 bit (1024).
- 2 Digunakan 2 buah potensiometer yang dihubungkan ke CH0 dan CH1, dengan CH0 diberi tegangan 3V dan CH1 diberi tegangan linier 0-3V dengan perubahan 500mV.

4) Hasil Pengujian

Tabel 4.1 Hasil Pembacaan nilai ADC

NO	Vin 1 (mV)	ADC 1	ADC Perhitungan	V terbaca (mV)	% Error
1	0	3	0	0	0%
2	500	105	103	500	2%
3	1000	206	204	1000	1%
4	1500	305	307	1500	0.60%
5	2000	409	410	2000	0.20%
6	2500	510	511	2500	0.20%
7	3000	613	613	3000	0%

Tabel 4.2 Hasil pembacaan nilai ADC

NO	Vin 1 (mV)	ADC 1	ADC Perhitungan	V terbaca (mV)	% Error
1	3000	613	613	3000	0%
2		613	613	3000	0%
3		613	613	3000	0%
4		613	613	3000	0%
5		612	613	3000	0.1%
6		613	613	3000	0%
7		613	613	3000	0%

5) Analisa

Dari hasil pengujian diatas dapat ditarik kesimpulan bahwa tingkat keakurasian ADC yang digunakan sangat bagus

dengan % eror tidak lebih dari 2%. Selain itu dengan pengujian lebih dari 1 *channel* diketahui tidak adanya *interferensi* sehingga memungkinkan untuk menggunakan seluruh *channel ADC* secara bersamaan. Rumus untuk menghitung *ADC* adalah sebagai berikut:

$$ADC = \frac{V_{in}}{V_{ref}} \times 1023 \dots\dots\dots(4.1)$$

$$V_{terbaca} = \frac{ADC}{1023} \times V_{ref} \dots\dots\dots(4.2)$$

Dengan rumus diatas nilai *ADC* dapat dicari sehingga dapat dibandingkan antara nilai *ADC* yang terbaca dengan *ADC* hasil perhitungan sesuai teori diatas.

4.3 Pengujian *Flame Sensor* 5 kanal

1) Tujuan

Pengujian ini bertujuan untuk menguji sensitifitas atau kepekaan sensor terhadap perubahan jarak dengan api. Sensor memiliki keluaran berupa tegangan *analog* yang berubah-ubah mengikuti perbedaan jarak sensor dengan api, dengan diadakannya pengujian ini diharapkan penulis dapat mengetahui perubahan nilai tegangan keluaran sensor dengan mengubah jarak sensor dengan api dan mencatat hasil setiap pengujiannya.

2) Parameter

Pengujian ini dilakukan dengan menggunakan lilin yang dinyalakan sebagai sumber api dengan pengujian berdasarkan jarak antara lilin dan *flame sensor* 5 kanal mulai dari 10 cm sampai dengan 100 cm dengan kenaikan sebesar 10 cm.

3) Setting Pengujian

Pengujian menggunakan *flame sensor* 5 kanal yang dihubungkan ke *MCP3008* sebagai *ADC*.

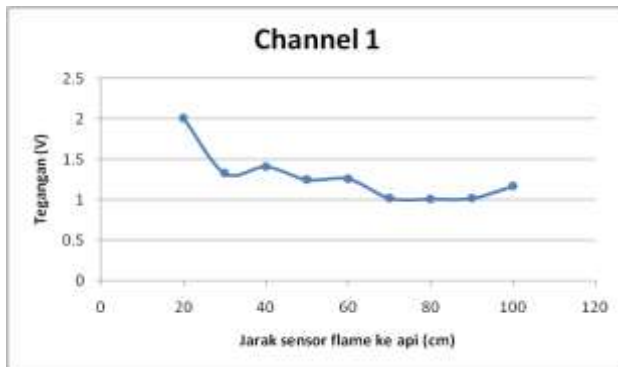
4) Hasil Pengujian

Proses pengujian menggunakan data nilai dari keluaran sensor yang tegangan *analog* yang berubah berdasarkan perbedaan jarak sensor dengan api, semakin jauh api dari

sensor maka keluaran tegangannya semakin kecil sebaliknya ketika sensor berada dekat dengan api maka keluaran tegangannya pun semakin besar. Data nilai dari sensor dapat dilihat pada Tabel 4.3.

Tabel 4.3 Hasil Pengujian kanal 1

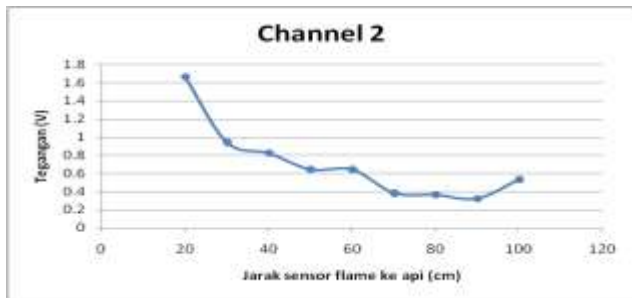
Jarak sensor dengan api (cm)	Tegangan CH 1 (Volt)
20	2.01
30	1.33
40	1.41
50	1.25
60	1.26
70	1.02
80	1.01
90	1.02
100	1.17



Gambar 4.3 Grafik hubungan jarak dengan tegangan kanal 1

Tabel 4.4 Hasil Pengujian kanal 2

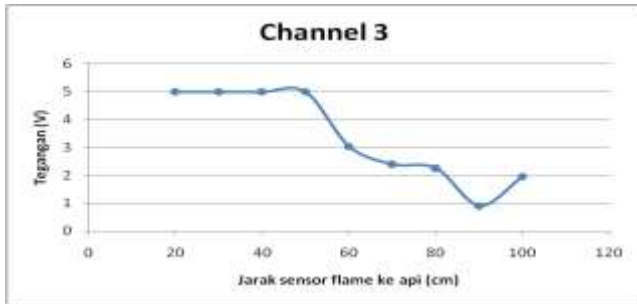
Jarak sensor dengan api (cm)	Tegangan CH 2 (Volt)
20	1.66
30	0.94
40	0.82
50	0.64
60	0.64
70	0.38
80	0.36
90	0.32
100	0.53



Gambar 4.4 Grafik hubungan jarak dengan tegangan kanal 2

Tabel 4.5 Hasil Pengujian kanal 3

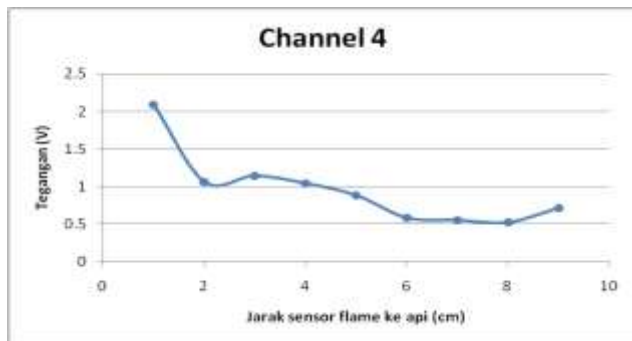
Jarak sensor dengan api (cm)	Tegangan CH 3(Volt)
20	5
30	5
40	5
50	5
60	3.04
70	2.4
80	2.26
90	0.91
100	1.96



Gambar 4.5 Grafik hubungan jarak dengan tegangan kanal 3

Tabel 4.6 Hasil Pengujian kanal 4

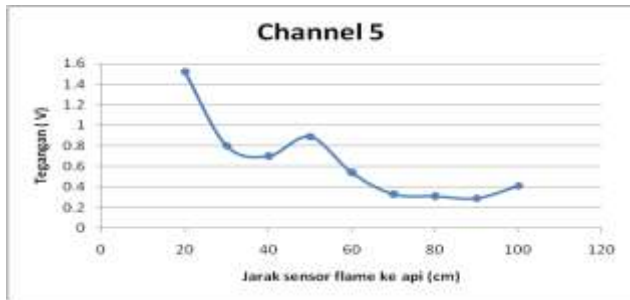
Jarak sensor dengan api (cm)	Tegangan CH 4 (Volt)
20	2.09
30	1.06
V	1.14
50	1.04
60	0.88
70	0.58
80	0.55
90	0.52
100	0.71



Gambar 4.6 Grafik hubungan jarak dan tegangan kanal 4

Tabel 4.7 Hasil Pengujian kanal 5

Jarak sensor dengan api (cm)	Tegangan CH 5 (Volt)
20	1.52
30	0.8
40	0.7
50	0.89
60	0.54
70	0.33
80	0.31
90	0.29
100	0.41



Gambar 4.7 Grafik hubungan jarak dan tegangan kanal 5

5) Analisa

Dengan melihat hasil seperti pada Gambar dan Tabel di atas dapat disimpulkan bahwa *flame sensor* 5 kanal memiliki *sensitifitas* yang baik dilihat dari perubahan nilai keluaran tegangan yang dihasilkan berbeda-beda sesuai dengan jarak sensor terhadap api. Pada jarak 50 cm *flame sensor* 5 kanal mengeluarkan tegangan 5V.

4.4 Pengujian *Photoelectric IR Break Beam Sensor*

1) Tujuan

Untuk mengetahui respon dari *Photoelectric IR Break Beam Sensor* terhadap asap hasil pembakaran dan

selanjutnya akan dijadikan sebagai acuan dalam mengetahui efektifitas dari sensor.

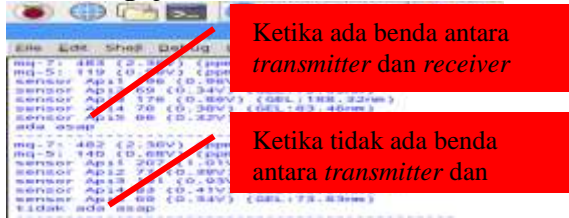
2) Parameter

Pengujian ini dilakukan dengan pembakaran kertas yang menciptakan asap tebal dan diarahkan diantar *transmitter* dan *receiver IR Break Beam Sensor*.

3) Setting Pengujian

Pengujian menggunakan *IR Break Beam Sensor* yang sudah terpasang pada sistem dengan jarak 10 cm antara *transmitter* dan *receiver* dan dilakukan pembakaran kertas *receiver* yang menyebabkan intensitas cahaya yang diterima *receiver* berkurang.

4) Hasil Pengujian



Gambar 4.8 Pengujian *IR Break Beam Sensor*

5) Analisa

Dengan melihat hasil pengujian dapat disimpulkan bahwa pengujian *IR Break Beam Sensor* telah berhasil dan dapat disimpulkan bahwa sensor memiliki sensitifitas yang baik dilihat dari informasi ada asap ketika ada benda penghalang antara *transmitter* dan *receiver*, dengan demikian *IR Break Beam Sensor* tersebut telah bekerja sebagai mana mestinya. Berikut *script program python* untuk sensor asap:

```
#!/usr/bin/python
import os
import time
import RPi.GPIO as GPIO
GPIO.setmode(GPIO BCM)
GPIO.setwarnings (False)
```

```

GPIO_IR=16
GPIO_setup(GPIO_IR,GPIO.IN)
#inisialisasi sensor asap
sensorstate=0
laststate=0
while True:
    sensorstate=GPIO.input(GPIO_IR)
    if sensorstate==0:
        GPIO.output(20,1)
    else:
        GPIO.output (20,0)
    if sensorstate==1 and laststate==0
        print ("tidak ada asap")
        time.sleep (1)
    else:
        print ("ada asap")
        time.sleep (1)

```

4.5 Pengujian MQ-5

1) Tujuan

Untuk mengetahui respon dari MQ-5 terhadap gas LPG dan selanjutnya akan dijadikan sebagai acuan dalam mengetahui keefektifitas dari sensor MQ-5.

2) Parameter

Pengujian ini dilakukan dengan menggunakan gas LPG yang disemprotkan untuk mengetahui respon konsentrasi gas LPG dari sensor MQ-5.

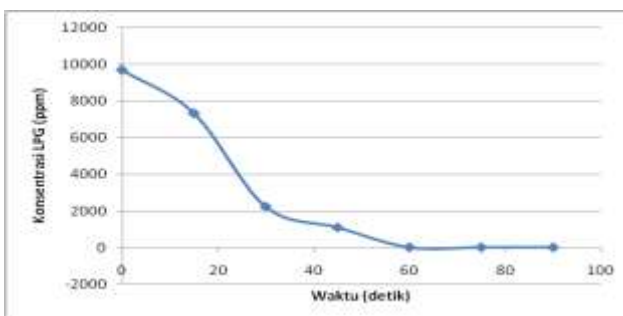
3) Setting Pengujian

Pengujian menggunakan MQ-5 yang sudah terpasang pada pusat ruangan dan dilakukan penyemprotan gas LPG.

4) Hasil Pengujian

Tabel 4.8 Pengujian MQ-5

NO	Waktu (detik)	Konsentrasi LPG (ppm)	Tegangan (Volt)
1	0	9698	4.9
2	15	7332	4.8
3	30	2222	4.69
4	45	1089	2.7
5	60	1.39	0.16
6	75	1.12	0.14
7	90	0	0



Gambar 4.9 Grafik hubungan waktu dengan konsentrasi LPG

5) Analisa

Dengan melihat hasil seperti pada Tabel 4.8 dapat disimpulkan bahwa pengujian MQ-5 telah berhasil dan dapat disimpulkan bahwa sensor memiliki sensitifitas yang baik dilihat dari perubahan nilai keluaran tegangan yang dihasilkan berbeda-beda sesuai dengan adanya gas LPG atau tidak, dengan demikian MQ-5 tersebut telah bekerja sebagai mana mestinya. Berikut konversi R_s/R_o ke ppm:

$$V_{cc} = 5 \text{ Volt}$$

$$R_o = 6.5 \text{ Kohm}$$

$$R_L = 20 \text{ Kohm (Data Sheet)}$$

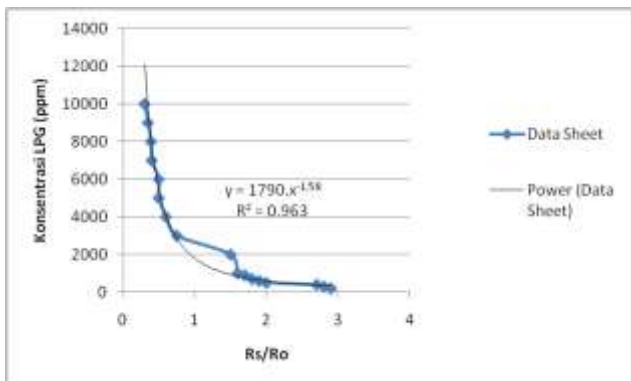
$$V_{RL} = \text{Tegangan yang dihasilkan sensor MQ-5}$$

$$R_s = \frac{(V_{cc} - V_{RL})}{V_{RL}} \times R_L \dots\dots\dots(4.4)$$

$R = R_s/R_o$
 $p = -1.58$
 $z = \text{pow}(R,p)$
 $\text{ppm} = 1790 \times z$

Tabel 4.9 Perbandingan R_s/R_o dengan konsentrasi MQ-5

Rs/Ro	ppm
2.9	200
2.8	300
2.7	400
2	500
1.9	600
1.8	700
1.7	900
1.6	1000
1.5	2000
0.75	3000
0.6	4000
0.5	5000
0.5	6000
0.4	7000
0.39	8000
0.35	9000
0.3	10000



Gambar 4.10 Power Regresi R_s/R_o dengan KonsentrasiMQ-5

4.6 Pengujian MQ-7

1) Tujuan

Untuk mengetahui respon dari MQ-7 terhadap beberapa gas hasil pembakaran beberapa material dan selanjutnya akan dijadikan sebagai acuan dalam mengetahui keefektifitas dari sensor MQ-7.

2) Parameter

Pengujian ini dilakukan dengan menggunakan plate tempat pembakaran yang dinyalakan dengan menggunakan beberapa material pembakaran yakni kertas, plastik, bensin serta spirtus untuk mengetahui respon nilai CO dari sensor MQ-7.

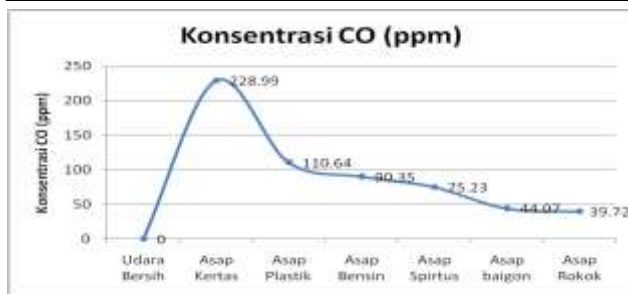
3) Setting Pengujian

Pengujian menggunakan MQ-7 yang sudah terpasang pada pusat ruangan dan dilakukan pembakaran tepat dibawah sistem.

4) Hasil Pengujian

Tabel 4.10 Pengujian MQ-7

No	Material	Konsentrasi CO (ppm)	Tegangan (Volt)
1	Udara Bersih	0	0
2	Asap Kertas	228.99	4.61
3	Asap Plastik	110.64	4.34
4	Asap Bensin	90.35	4.24
5	Asap Spirtus	75.23	4.14
6	Asap baigon	44.07	3.79
7	Asap Rokok	39.72	3.71



Gambar 4.11 Grafik hubungan material dan konsentrasi MQ-7

5) Analisa

Dengan melihat hasil seperti pada Tabel 4.8 dapat disimpulkan bahwa pengujian MQ-7 telah berhasil dan dapat di simpulkan bahwa sensor memiliki sensitifitas yang baik dilihat dari perubahan nilai keluaran tegangan yang dihasilkan berbeda-beda sesuai dengan adanya gas hasil pembakaran atau tidak, dengan demikian MQ-7 tersebut telah bekerja sebagai mana mestinya. Berikut konversi Rs/Ro ke ppm:

$$V_{cc} = 5 \text{ Volt}$$

$$R_o = 3.4 \text{ Kohm}$$

$$R_L = 20 \text{ Kohm (Data Sheet)}$$

$$V_{RL} = \text{Tegangan yang dihasilkan sensor MQ-7}$$

$$R_s = \frac{(V_{cc} - V_{RL})}{V_{RL}} \times R_L \dots\dots\dots(4.3)$$

$$R = R_s/R_o$$

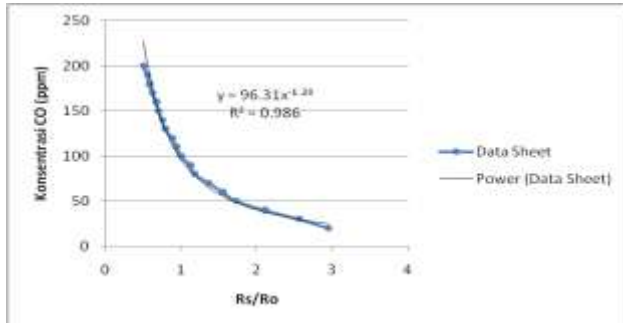
$$p = -1.23$$

$$z = \text{pow}(R,p)$$

$$\text{ppm} = 96.31 \times z$$

Tabel 4.11 Perbandingan Rs/Ro dengan Konsentrasi MQ-7

Rs/Ro	ppm
2.95	20
2.56	30
2.11	40
1.73	50
1.55	60
1.37	70
1.18	80
1.12	90
1	100
0.94	110
0.88	120
0.79	130
0.75	140
0.7	150
0.67	160
0.62	170
0.58	180
0.55	190
0.5	200



Gambar 4.12 *Power Regresi Rs/Ro dengan konsentrasi MQ-7*

4.7 Pengujian dan Penentuan *Threshold* Pembacaan Sensor

1) Tujuan

Untuk mengetahui kemampuan pembacaan kondisi oleh sensor serta menentukan *threshold* dalam pengambilan keputusan pada masing-masing sensor.

2) Parameter

Pengujian dilakukan dengan melakukan instalasi sistem sesuai rencana awal dengan menggunakan api lilin, pembakaran kertas, penyemprotan *LPG*, dan asap supaya dapat dikenali oleh sistem.

3) Setting Pengujian

- 1 Pengujian dilakukan setelah sistem sudah terpasang pada ruangan sesuai dengan rencana awal untuk mengetahui kemampuan pembacaan sensor.
- 2 Penentuan *threshold* didasarkan pada pembacaan ADC dari tiap-tiap sensor.
- 3 Data akan dilihat pada *Hyperterminal python 2* dengan perubahan data sebanyak 5 detik.

4) Hasil Pengujian

Tabel 4.12 Penentuan *Threshold*

No	Sensor	Tegangan (volt)	Konsentrasi (ppm)
1	Sensor Api 5 Channel	≥ 4	
2	MQ-7	≥ 4.5	≥ 200
3	MQ-5	≥ 2.5	≥ 1000
4	IR Break Beam Sensor	5	

5) Analisa

Dari hasil pengujian maka dapat dibandingkan data ketika kondisi normal dan kondisi ketika terjadi pembakaran, maka didapatkan beberapa point penting dalam penentuan *threshold* setiap sensor yang ditunjukkan tabel 4.12.

4.8 Pengujian *Modem Huawei E173* untuk *SMS Gateway*

1) Tujuan

Untuk mengetahui koneksi serta respon modem *Huawei E173* terhadap *AT command* serta mengetahui waktu yang diperlukan untuk mengirim dan menerima *SMS* melalui *Raspberry pi*.

2) Parameter

Pengujian ini dilakukan dengan menggunakan *Hyperterminal* untuk mengirim perintah ke *modem Huawei E173*.

3) Setting Pengujian

- 1 Pengujian menggunakan modem *Huawei E173* yang pemasangannya langsung dicolokkan ke *USB Raspberry pi*.
- 2 Pengujian dilakukan sebatas untuk mengetahui respon modem terhadap perintah melalui *Hyperterminal*.
- 3 Sebelum pengujian dilakukan setting *port* dan *baudrate*.
- 4 Untuk mengetahui respon pengiriman maupun penerimaan maka digunakan *delay* 5 detik.

4) Hasil Pengujian



Gambar 4.13 Pengujian modem *Huawei E173*

Tabel 4.13 Lama waktu pengiriman *sms*

No	Waktu Kirim WIB	Waktu Terima WIB	Lama Pengiriman (detik)
1	12:24:36	12:24:41	6
2	12:24:42	12:24:48	6
3	12:24:50	12:24:56	6
4	12:24:57	12:25:03	6
5	12:25:05	12:25:11	6

5) Analisa

Dengan melihat hasil pengujian dapat disimpulkan bahwa pengujian modem *Huawei E173* telah berhasil. Waktu pengiriman dan penerimaan SMS paling lama 6 detik.

4.9 Pengujian *Webcam*

1) Tujuan

Untuk mengetahui respon *webcam* dalam menanggapi perintah untuk mengambil gambar.

2) Parameter

Pengujian ini dilakukan dengan menggunakan *Hyperterminal* untuk mengirim perintah ke *webcam*.

3) Setting Pengujian

1. Menginstall *openCV* ke Raspberry pi.

- 2 Memberi perintah melalui program *Python 2* untuk mengambil gambar dan mengirimkan ke *email* penghuni.

4) Hasil Pengujian



Gambar 4.14 Pengujian *webcam* dan email

5) Analisa

Dengan melihat hasil pengujian dapat disimpulkan bahwa pengujian *webcam* telah berhasil dan dapat di simpulkan bahwa *webcam* memiliki kualitas yang cukup baik dilihat dari gambar yang dihasilkan, dengan demikian *webcam* tersebut telah bekerja sebagai mana mestinya.

4.10 Pengujian *Buzzer*

1) Tujuan

Untuk mengetahui respon *Buzzer* dalam menanggapi kondisi *HIGH* ataupun *LOW*.

2) Parameter

Pengujian ini dilakukan dengan menggunakan Program *python 2* dengan memberi kondisi *HIGH* dan *LOW*.

3) Setting Pengujian

Pengujian dilakukan dengan menyambungkan *Buzzer* ke *GPIO 20* Raspberry pi, dengan memberi kondisi *HIGH* dan *LOW*.

4) Hasil Pengujian

Hasil percobaan menunjukkan jika pin *GPIO 20* diberi *HIGH* maka *Buzzer* berbunyi sedangkan jika *LOW* *Buzzer* diam.

5) Analisa

Dengan melihat hasil pengujian dapat disimpulkan bahwa pengujian *Buzzer* telah berhasil dan dapat di simpulkan bahwa *Buzzer* memiliki kualitas yang cukup baik dilihat dari suara yang dihasilkan, dengan demikian *Buzzer* tersebut telah bekerja sebagai mana mestinya.

4.11 Pengujian *Water Solenoid Valve*

1) Tujuan

Untuk mengetahui respon *Water Solenoid Valve* dalam menanggapi kondisi *HIGH* ataupun *LOW*.

2) Parameter

Pengujian ini dilakukan dengan menggunakan Program *python 2* dengan memberi kondisi *HIGH* dan *LOW*.

3) Setting Pengujian

Pengujian dilakukan dengan menyambungkan *Water Solenoid Valve* ke *relay* yang dihubungkan ke *GPIO 21* *Raspberry pi*, dengan memberi kondisi *HIGH* dan *LOW*.

4) Hasil Pengujian

Hasil pengujian menunjukkan jika pin *GPIO* diberi *HIGH* maka *relay* akan *ON* sehingga *Water solenoid* akan mengeluarkan air, jika *LOW* maka *relay* *OFF* dan *Water solenoid* akan tertutup.

5) Analisa

Dengan melihat hasil pengujian dapat disimpulkan bahwa pengujian *Water Solenoid Valve* telah berhasil dan dapat di simpulkan bahwa *Water Solenoid Valve* memiliki kualitas yang cukup baik dilihat dari aliran air yang dihasilkan, dengan demikian *Water Solenoid Valve* tersebut telah bekerja sebagai mana mestinya.

4.12 Pengujian Integrasi dari seluruh Sistem

1) Tujuan

Untuk mengetahui respon dan hasil integrasi keseluruhan sistem.

2) Parameter

Pengujian dilakukan dengan menggunakan keseluruhan sistem yang sudah terintegrasi.

3) Setting Pengujian

- Pengujian dilakukan di lab B402 elektronika industri S1 lantai 4.
- Pengujian dilakukan dengan pembakaran *diplate* yang telah dibuat sebelumnya.

4) Hasil Pengujian

Tabel 4.14 Hasil integrasi keseluruhan sistem (Tegangan)

	Sensor Api	IR Break Beam Sensor	MQ-5	MQ-7	SMS	Email	Alarm	Solenoid Valve
Tegangan (Volt)	≥ 4	≥ 3	≥ 4	≥ 4	ON	ON	ON	ON
	< 4	≥ 3	≥ 4	≥ 4	ON	OFF	OFF	OFF
	< 4	< 3	≥ 4	≥ 4	ON	OFF	OFF	OFF
	< 4	< 3	< 4	≥ 4	ON	OFF	OFF	OFF

Tabel 4.15 Hasil Integrasi Keseluruhan Sistem (ppm)

Konsentrasi MQ-7 (ppm)	Konsentrasi MQ-5 (ppm)	IR Flame sensor 5 kanal (nm)	SMS	Email	Solenoid Valve	Buzzer
≥ 200	≥ 1000	≥ 700 nm	ON	ON	ON	ON
< 200	< 1000	≥ 700 nm	ON	ON	ON	ON
≥ 200	< 1000	< 700 nm	ON	OFF	OFF	OFF
< 200	≥ 1000	< 700 nm	ON	OFF	OFF	OFF

5) Analisa

Dengan melihat hasil pengujian dapat disimpulkan bahwa pada saat tegangan dari *flame sensor* 5 kanal lebih dari 4V maka sistem mengirim *SMS*, *Email* ke penghuni serta mengaktifkan *alarm* dan *water solenoid valve*. Ketika konsentrasi gas *LPG* diatas 1000 ppm dan gas *CO* 200 ppm maka sistem mengirim *SMS* ke penghuni berisi konsentrasi gas melebihi ambang batas. waktu pengiriman sampai *SMS* diterima kurang lebih 6 detik.

Halaman ini sengaja dikosongkan

BAB 5

PENUTUP

5.1 Kesimpulan

Setelah dilakukan kegiatan perancangan dan pengujian sistem peringatan dini deteksi dan pemadam kebakaran berbasis Raspberry Pi, didapatkan beberapa kesimpulan sebagai berikut:

1. Ketika tegangan *flame sensor* 5 kanal sama dengan atau lebih dari 4V maka sistem mengirim *SMS*, *email* ke penghuni serta mengaktifkan *alarm* dan *water solenoid valve*.
2. Sistem mengirim *SMS* ke penghuni jika konsentrasi gas *LPG* diatas 1000 *ppm* dan konsentrasi gas *CO* diatas 200 *ppm*
3. Gas sensor MQ-5 Membutuhkan waktu sekitar 90 detik agar konsentrasi gas *LPG* kembali ke kondisi normal.
4. Sistem membutuhkan waktu untuk pengiriman *SMS* kurang lebih 6 detik.

5.2 Saran

Dari hasil penelitian maka dapat disarankan untuk pengembangan lebih lanjut tentang penelitian ini agar kekurangan dapat diminimalisir. Beberapa saran tersebut antara lain:

1. Pada box dapat dilakukan penambahan keypad dan *LCD* untuk memasukkan dan mengganti nomor *handphone* tujuan *SMS* dan alamat *email* penghuni sehingga lebih *fleksibel*.
2. Untuk pengambilan keputusan dapat diterapkan algoritma kontrol *fuzzy logic* sehingga keluaran keputusan dapat lebih *fleksibel* lagi.

Halaman ini sengaja dikosongkan

DAFTAR PUSTAKA

- [1] Dinas Pemadam Kebakaran dan Penanggulangan Bencana (Damkar dan PB) DKI Jakarta, 2013.
- [2] Smith, Kirk R., et al. "*Indoor Air Pollution in Developing Countries and Acute Lower Respiratory Infections in Children.*" *Thorax* 55.6(2000) :518-532.
- [3] "5 Channel IR Flame Sensor datasheet" NewWay Electronics Co. Ltd, China.
- [4] IR Break Beam Sensor website [online], Available: <https://learn.adafruit.com/ir-breakbeam-sensors>. Created Ladyada 2014.
- [5] "MQ-5 datasheet" Hanwei Electronics Co., Ltd, China.
- [6] "MQ-7 datasheet" Hanwei Electronics Co., Ltd, China.
- [7] Analog to Digital Converter by AGL Telkom University.
- [8] Choudhury, S., Singh G.K., Mehra R.M. (2014). *Design and Verification Serial Peripheral Interface (SPI) Protocol for Low Power Applications.* International Journal of Innovative Research in Science, Engineering and Technology: 16750-16758. ISSN: 2319-8753.
- [9] Raspberry pi website. [Online]. Available: <https://www.raspberrypi.org/>
- [10] Afrina, mira dan Ali Ibrahim, (2017) "*Pengembangan Sistem Informasi SMS Gateway dalam Meningkatkan Layanan Komunikasi Sekitar Akademika Fakultas Unsri.*" ISSN: 2355-4614.
- [11] Daniel Alexander Octavianus Turang, "*Pengembangan Sistem Relay Pengendalian dan penghematan Pemakaian Lampu Berbasis Mobile.*" (SemNas IF 2015). Teknik Informatika Sokolah Tinggi Teknologi Bandung.
- [12] Hasan, mahmud., dan Wahyudi, Arif (2014) "*Rancang Bangun Rangkain Pengendali untuk Valve yang digunakan Sebagai Saluran Masuk Gas N2 dan O2 pada Alat Kalibrasi Sensor Oksigen.*" Jurusan Teknik Mesin, Fakultas teknologi Industri, Institut Teknologi Sepuluh Nopember (ITS).
- [13] Belajar Python website. [online]. Available: <http://www.belajarpython.com/>. Diakses tanggal 20 Juni 2017.
- [14] OpenCv Documentation [online]. Available: <http://opencv.org/>. Diakses tanggal 20 Juni 2017.

Halaman ini sengaja di kosongkan

BIODATA PENULIS



Sumarto lahir di Tuban, 2 Februari 1991. Merupakan anak ke-4 dari 4 bersaudara pasangan Senen (Almarhum) dan Suwarni. Penulis memulai pendidikannya di SDN 1 Sumurjalak, pada tahun 1997 dan lulus pada tahun 2003. Penulis melanjutkan jenjang pendidikannya di SMPN 1 Plumpang dan lulus pada tahun 2006. Penulis sempat vakum 3 tahun tidak bisa melanjutkan pendidikan dikarenakan masalah biaya, kemudian di 3 tahun tersebut penulis bekerja ikut orang Cina sebagai perawat rumah selama 2 tahun dan bekerja di home industri sepatu Sruni Gedangan Sidoarjo selama 9 bulan. Pada tahun 2009 penulis melanjutkan pendidikannya di SMAN 1 Rengel dan lulus pada tahun 2012. Selama masa sekolah Alhamdulillah penulis selalu menjadi juara umum di SMAN 1 Rengel. Penulis kemudian mendapatkan beasiswa BIDIK MISI sehingga bisa melanjutkan kuliah di jurusan Teknik Elektro ITS, Surabaya. Penulis kemudian memilih prodi elektronika industri sebagai bidang studi yang dipelajari. Selama masa kuliah penulis aktif sebagai tentor di beberapa LBB hal ini penulis mencari uang untuk biaya hidup dan biaya keperluan kuliah karena penulis tidak mendapat uang dari orang tua.

Halaman ini sengaja dikosongkan

LAMPIRAN 1



Gambar Sistem tampak depan



Gambar Sistem tampak Belakang



Gambar Rangkaian dalam BOX

LAMPIRAN 2

Data Sheet MQ-5

HANWEI ELECTRONICS CO., LTD

MQ-5

<http://www.hw-sensor.com>

TECHNICAL DATA

MQ-5 GAS SENSOR

FEATURES

- * High sensitivity to LPG, natural gas, town gas.
- * Small sensitivity to alcohol, smoke.
- * Fast response. * Stable and long life. * Simple drive circuit.

APPLICATION

They are used in gas leakage detecting equipments in family and industry, are suitable for detecting of LPG, natural gas, town gas, avoid the noise of alcohol and cooking flames and cigarette smoke.

SPECIFICATIONS

A. Standard work condition

Symbol	Parameter name	Technical condition	Remarks
V_c	Circuit voltage	5V±0.1	AC OR DC
V_H	Heating voltage	5V±0.1	AC OR DC
R_L	Load resistance	20K Ω	
R_H	Heater resistance	31 ± 10%	Room Temp
P_H	Heating consumption	less than 800mw	

B. Environment condition

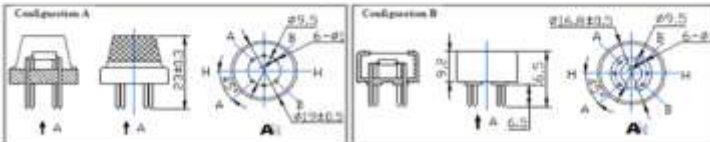
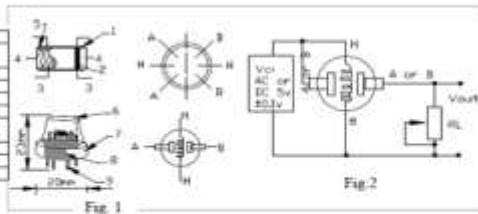
Symbol	Parameter name	Technical condition	Remarks
T_{ao}	Using Temp	-10 $^{\circ}$ C ~ 50 $^{\circ}$ C	
T_{st}	Storage Temp	-20 $^{\circ}$ C ~ 70 $^{\circ}$ C	
R_H	Relative humidity	less than 95%RH	
O_2	Oxygen concentration	21%(standard condition)Oxygen concentration can affect sensitivity	minimum value is over 21%

C. Sensitivity characteristic

Symbol	Parameter name	Technical parameter	Remarks
R_a	Sensing Resistance	10K Ω ~ 60K Ω (5000ppm methane)	Detecting concentration scope: 200-10000ppm LPG, LNG Natural gas, iso-butane, propane Town gas
$\frac{R_a}{R_a(5000ppm/1000ppm CH_4)}$	Concentration slope rate	≤ 0.6	
Standard detecting condition	Temp: 20 $^{\circ}$ C ± 2 $^{\circ}$ C Humidity: 65%±5%	V_c : 5V±0.1 V_H : 5V±0.1	
Preheat time	Over 24 hours		

D. Structure and configuration, basic measuring circuit

No.	Form	Materials
1	Gas sensing layer	SnO_2
2	Electrode	Au
3	Electrode lead	Pt
4	Sealing coat	Ni-Cr alloy
5	Yttrium ceramic	Al_2O_3
6	Anti-explosion ceramic	Stainless steel gauze (SU71) (100-mesh)
7	Clamp ring	Copper plating Zn
8	Resin base	Bakelite
9	Tube Pin	Copper plating Ni



Structure and configuration of MQ-5 gas sensor is shown as Fig. 1 (Configuration A or B), sensor composed by

TEL: 86-371-67188070 67169000 FAX: 86-371-67169000

E-mail: sales@hw-sensor.com

micro Al_2O_3 ceramic tube, Tin Dioxide (SnO_2) sensitive layer, measuring electrode and heater are fixed into a crust made by plastic and stainless steel net. The heater provides necessary work conditions for work of sensitive components. The enveloped MQ-5 have 6 pin, 4 of them are used to fetch signals, and other 2 are used for providing heating current.

Electric parameter measurement circuit is shown as Fig.2

E. Sensitivity characteristic curve

Fig.2 sensitivity characteristics of the MQ-5

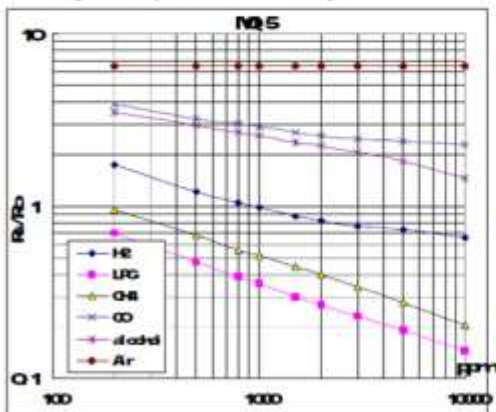


Fig.3 shows the typical sensitivity characteristics of the MQ-5 for several gases, in their: Temp: 20°C,

Humidity: 65%.

O₂ concentration 21%

RL=20k Ω

R₀: sensor resistance at 1000ppm of H₂ in the clean air.

R_r: sensor resistance at various concentrations of gases.

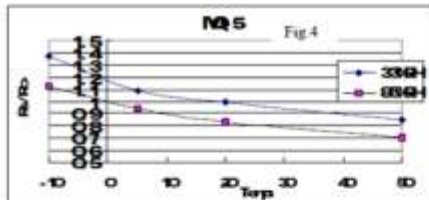


Fig.4 shows the typical dependence of the MQ-5 on temperature and humidity. R₀: sensor resistance at 1000ppm of H₂ in air at 33%RH and 20 degree. R_r: sensor resistance at different temperatures and humidities.

SENSITIVITY ADJUSTMENT

Resistance value of MQ-5 is difference to various kinds and various concentration gases. So, When using this components, sensitivity adjustment is very necessary. we recommend that you calibrate the detector for 1000ppm H₂ or LPG concentration in air and use value of Load resistance (R_L) about 20 K Ω (10K Ω to 47K Ω).

When accurately measuring, the proper alarm point for the gas detector should be determined after considering the temperature and humidity influence.

TECHNICAL DATA**MQ-7 GAS SENSOR****FEATURES**

- * High sensitivity to carbon monoxide
- * Stable and long life

APPLICATION

They are used in gas detecting equipment for carbon monoxide(CO) in family and industry or car.

SPECIFICATIONS**A. Standard work condition**

Symbol	Parameter name	Technical condition	Remark
Vc	circuit voltage	5V ± 0.1	Ac or Dc
VH (H)	Heating voltage (high)	5V ± 0.1	Ac or Dc
VH (L)	Heating voltage (low)	1.4V ± 0.1	Ac or Dc
R _L	Load resistance	Can adjust	
RH	Heating resistance	33 Ω ± 5%	Room temperature
TH (H)	Heating time (high)	60 ± 1 seconds	
TH (L)	Heating time (low)	90 ± 1 seconds	
PH	Heating consumption	About 350mW	

b. Environment conditions

Symbol	Parameters	Technical conditions	Remark
T _{ao}	Using temperature	-20℃-50℃	
T _{as}	Storage temperature	-20℃-50℃	Advice using scope
RH	Relative humidity	Less than 95%RH	
O ₂	Oxygen concentration	21%(stand condition) the oxygen concentration can affect the sensitivity characteristic	Minimum value is over 2%

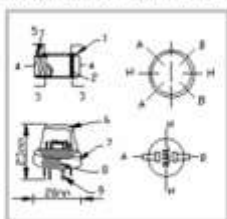
c. Sensitivity characteristic

symbol	Parameters	Technical parameters	Remark
R _s	Surface resistance Of sensitive body	2-20k	In 100ppm Carbon Monoxide
α (300/100ppm)	Concentration slope rate	Less than 0.5	R _s (300ppm)/R _s (100ppm)
Standard working condition	Temperature -20℃ ± 2℃	relative humidity 65% ± 5%	RL:10K Ω ± 5%
	Vc:5V ± 0.1V	VH:5V ± 0.1V	VH:1.4V ± 0.1V
Preheat time	No less than 48 hours	Detecting range: 20ppm-2000ppm carbon monoxide	

D. Structure and configuration, basic measuring circuit

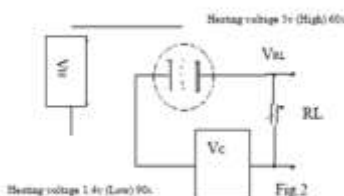
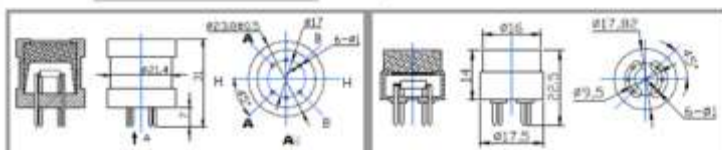
Structure and configuration of MQ-7 gas sensor is shown as Fig. 1 (Configuration A or B), sensor composed by micro Al₂O₃ ceramic tube, Tin Dioxide (SnO₂) sensitive layer, measuring electrode and heater are fixed into a crust made by plastic and stainless steel net. The heater provides necessary work conditions for work of sensitive components. The enveloped MQ-7 have

6 pin, 4 of them are used to fetch signals, and other 2 are used for providing heating current.



Part	Materials
1 Gas sensing layer	SnO ₂
2 Electrode	Au
3 Electrode line	Pt
4 Heating coil	50-Cr alloy
5 Tubular ceramic	Al ₂ O ₃
6 Air-implosion network	Stainless steel mesh (FC3216 100-mesh)
7 Circuit pad	Copper plating 20
8 Resin base	Epoxy resin
9 Tube Pin	Copper plating 20

Fig.1



Standard circuit:

As shown in Fig 2, standard measuring circuit of MQ-7 sensitive components consists of 2 parts, one is heating circuit having time control function (the high voltage and the low voltage work circularly). The second is the signal output circuit, it can accurately respond changes of surface resistance of the sensor.

Electric parameter measurement circuit is shown as Fig.2

E. Sensitivity characteristic curve

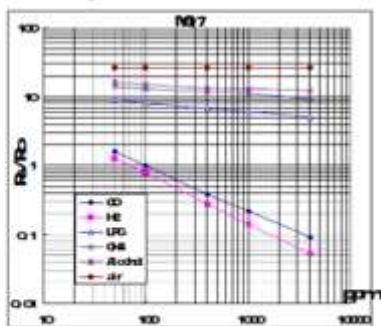


Fig.3 sensitivity characteristics of the MQ-7

Fig.3 shows the typical sensitivity characteristics of the MQ-7 for several gases.

in their Temp: 20°C.

Humidity: 65%.

O₂ concentration 21%

RL=10kΩ

R₀: sensor resistance at 100ppm

CO in the clean air.

R_x: sensor resistance at various concentrations of gases.

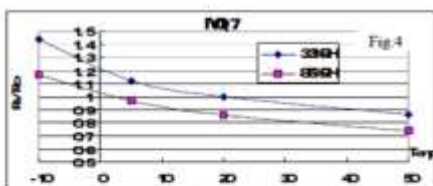


Fig.4 shows the typical dependence of the MQ-7 on temperature and humidity.

R_0 : sensor resistance at 100ppm CO in air at 33%RH and 20degree.

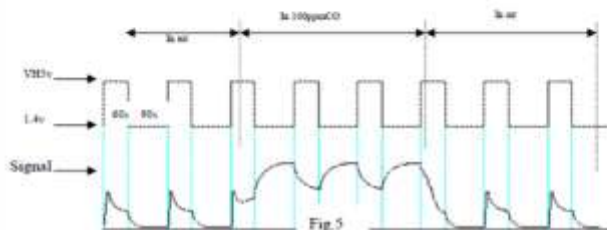
R_s : sensor resistance at 100ppm CO at different temperatures and humidities.

OPERATION PRINCIPLE

The surface resistance of the sensor R_s is obtained through effected voltage signal output of the load resistance R_L which series-wound. The relationship between them is described:

$$R_s/R_L = (V_c - V_{RL}) / V_{RL}$$

Fig. 5 shows alterable situation of RL signal output measured by using Fig. 2 circuit output



signal when the sensor is shifted from clean air to carbon monoxide (CO), output signal measurement is made within one or two complete heating period (2.5 minute from high voltage to low voltage).

Sensitive layer of MQ-7 gas sensitive components is made of SnO_2 with stability. So, it has excellent long term stability. Its service life can reach 5 years under using condition.

SENSITIVITY ADJUSTMENT

Resistance value of MQ-7 is difference to various kinds and various concentration gases. So, When using this components, sensitivity adjustment is very necessary. we recommend that you calibrate the detector for 200ppm CO in air and use value of Load resistance that (R_L) about $10\text{K}\Omega$ ($5\text{K}\Omega$ to $47\text{K}\Omega$).

When accurately measuring, the proper alarm point for the gas detector should be determined after considering the temperature and humidity influence. The sensitivity adjusting program:

- Connect the sensor to the application circuit.
- Turn on the power, keep preheating through electricity over 48 hours.
- Adjust the load resistance R_L until you get a signal value which is respond to a certain carbon monoxide concentration at the end point of 90 seconds.
- Adjust the another load resistance R_L until you get a signal value which is respond to a CO concentration at the end point of 60 seconds.

[Supplying special IC solutions, More detailed technical information, please contact us](#)

LAMPIRAN 3

List Program Python di Raspberry Pi 3 Model B

```
#!/usr/bin/python
```

```
import spidev
from spidev import SpiDev
import time
import os
import serial
import numpy as np
import cv2
from datetime import datetime
import smtplib
from email.mime.multipart import MIMEMultipart
from email.mime.base import MIMEBase
from email.mime.text import MIMEText
from email import encoders
import math
import RPi.GPIO as GPIO
GPIO.setmode(GPIO.BCM)
GPIO.setwarnings(False)
GPIO.setup(20,GPIO.OUT) # Buzzer
GPIO.setup(21, GPIO.OUT) # Relay
GPIO_IR=16 # Sensor asap
GPIO.setup(GPIO_IR,GPIO.IN)
```

```
# inialisasi sensor asap
```

```
sensorstate=0
laststate=0
```

```
# Inialisasi sms
```

```
ser=serial.Serial('/dev/ttyUSB0',19200,timeout=1)
ser.write('ATZ\r')
time.sleep(0.5)
ser.write('AT+CMGF=1\r')
time.sleep(0.5)
```

```
# Inialisasi email
```

```
gmail_user = "sumarto354@yahoo.com" #Sender email address
```

```

gmail_pwd = "raspberry354" #Sender email password
to = "sumarto354@gmail.com" #Receiver email address
subject = "Peringatan Dini & Deteksi Kebakaran"
text = "There is some activity in your home. See the attached picture."
previous_state = False
current_state = False

```

```

Ro=3.4
RL=20
Ro2=6.5
RL2=20

```

Open SPI bus

```

spi = spidev.SpiDev()
spi.open(0,0)

```

Function to read SPI data from MCP3008 chip

Channel must be an integer 0-7

```

def ReadChannel(channel):
    adc = spi.xfer2([1,(8+channel)<<4,0])
    data = ((adc[1]&3) << 8) + adc[2]
    return data

```

Function to convert data to voltage level,

rounded to specified number of decimal places.

```

def ConvertVolts(data,places):
    volts = (data * 5) / float(1023)
    volts = round(volts,places)
    return volts

```

Function to convert data to Rs,

rounded to specified number of decimal places.

```

def ConvertRs (volts,places):
    Rs=float(RL*(5-MQ7_volts)/float(MQ7_volts)) if MQ7_volts !=0 else 0
    Rs=round(Rs,places)
    return Rs

```

Function to convert data to ppm,

rounded to specified number of decimal places.

```

def Convertppm(Rs,places):
    R=MQ7_Rs/Ro

```

```

p=(-1.239)
z=pow(R,p) if R !=0 else 0
ppm=96.31*z
ppm=round (ppm,places)
return ppm
# Function to convert data to Rs,
# rounded to specified number of decimal places.
def ConvertRs2 (volts,places):
Rs2=float(RL2*(5-MQ5_volts)/float(MQ5_volts)) if MQ5_volts !=0 else 0
Rs2=round(Rs2,places)
return Rs2
# Function to convert data to ppm,
# rounded to specified number of decimal places.
def Convertppm2(Rs2,places):
R2=MQ5_Rs/Ro2
p2=(-1.58)
z2=pow(R2,p2) if R2 !=0 else 0
ppm2=1790*z2
ppm2=round (ppm2,places)
return ppm2

# Define sensor channels
MQ7_channel = 0
MQ5_channel = 1
Api1_channel = 2
Api2_channel = 3
Api3_channel = 4
Api4_channel = 5
Api5_channel = 6

delay=5
while True:

# Read mq-7
MQ7_level = ReadChannel(MQ7_channel)
MQ7_volts = ConvertVolts(MQ7_level,2)
MQ7_Rs=ConvertRs(MQ7_level,2)
MQ7_ppm=Convertppm(MQ7_level,2)
if MQ7_ppm>200:

```

```

ser.write('AT+CMGS="085649262967"\r')
time.sleep(1)
ser.write('AWAS KONSENTRASI CO MELEBIHI AMBANG BATAS')
ser.write(chr(26))

```

Read mq-5

```

MQ5_level = ReadChannel(MQ5_channel)
MQ5_volts = ConvertVolts(MQ5_level,2)
MQ5_Rs=ConvertRs2(MQ5_level,2)
MQ5_ppm=Convertppm2(MQ5_level,2)
if MQ5_ppm>1000:
    ser.write('AT+CMGS="085649262967"\r')
    time.sleep(1)
    ser.write('AWAS KONSENTRASI CO MELEBIHI AMBANG BATAS')
    ser.write(chr(26))

```

sensor api 1

```

Api1_level = ReadChannel(Api1_channel)
Api1_volts = ConvertVolts(Api1_level,2)
if Api1_volts>4:
    ser.write('AT+CMGS="085649262967"\r')
    time.sleep(1)
    ser.write('***Awat ada api di rumah anda***')
    ser.write(chr(26))

```

```

cap = cv2.VideoCapture(0)
ret, frame = cap.read()
cap = cv2.VideoCapture(0)
print ("Saving Photo")
picname = datetime.now().strftime("%y-%m-%d-%H-%M")
picname = picname+'.jpg'
cv2.imwrite(picname, frame)
print ("Sending email")

```

```

attach = picname
msg = MIMEMultipart()

```

```

msg['From'] = gmail_user
msg['To'] = to
msg['Subject'] = subject

```



```

msg.attach(MIMEText(text))

part = MIMEBase('application', 'octet-stream')
part.set_payload(open(attach, 'rb').read())
encoders.encode_base64(part)
part.add_header('Content-Disposition', 'attachment;
filename="%s"' % os.path.basename(attach))
msg.attach(part)

mailServer = smtplib.SMTP("smtp.mail.yahoo.com", 587)
mailServer.ehlo()
mailServer.starttls()
mailServer.ehlo()
mailServer.login(gmail_user, gmail_pwd)
mailServer.sendmail(gmail_user, to, msg.as_string())
# Should be mailServer.quit(), but that crashes...
mailServer.close()
print ("Email Sent")
os.remove(picname)
# Nyalakan relay 21
GPIO.output(21, GPIO.HIGH)
# Nyalakan buzzer 20
GPIO.output(20, GPIO.HIGH)

else:
# Matikan relay 21
GPIO.output(21, GPIO.LOW)
# Matikan Buzzer 20
GPIO.output(21, GPIO.LOW)
# sensor api 2
Api2_level = ReadChannel(Api2_channel)
Api2_volts = ConvertVolts(Api2_level,2)
if Api2_volts>4:
    ser.write('AT+CMGS="085649262967"\r')
    time.sleep(1)
    ser.write('***Awat ada api di rumah anda***')
    ser.write(chr(26))

```

```

cap = cv2.VideoCapture(0)
ret, frame = cap.read()
cap = cv2.VideoCapture(0)
print ("Saving Photo")
picname = datetime.now().strftime("%y-%m-%d-%H-%M")
picname = picname+'.jpg'
cv2.imwrite(picname, frame)
print ("Sending email")

attach = picname
msg = MIMEMultipart()

msg['From'] = gmail_user
msg['To'] = to
msg['Subject'] = subject

msg.attach(MIMEText(text))

part = MIMEBase('application', 'octet-stream')
part.set_payload(open(attach, 'rb').read())
encoders.encode_base64(part)
part.add_header('Content-Disposition', 'attachment;
filename="%s"' % os.path.basename(attach))
msg.attach(part)

mailServer = smtplib.SMTP("smtp.mail.yahoo.com", 587)
mailServer.ehlo()
mailServer.starttls()
mailServer.ehlo()
mailServer.login(gmail_user, gmail_pwd)
mailServer.sendmail(gmail_user, to, msg.as_string())
# Should be mailServer.quit(), but that crashes...
mailServer.close()
print ("Email Sent")
os.remove(picname)
# Nyalakan relay 21
GPIO.output(21, GPIO.HIGH)
# Nyalakan buzzer 20
GPIO.output(20, GPIO.HIGH)

```

```

else:
    # Matikan relay 21
    GPIO.output(21, GPIO.LOW)
    # Matikan Buzzer 20
    GPIO.output(21, GPIO.LOW)
# sensor api 3
Api3_level = ReadChannel(Api3_channel)
Api3_volts = ConvertVolts(Api3_level,2)
if Api3_volts>4:
    ser.write('AT+CMGS="085649262967"\r')
    time.sleep(1)
    ser.write('***Awat ada api di rumah anda***')
    ser.write(chr(26))

    cap = cv2.VideoCapture(0)
    ret, frame = cap.read()
    cap = cv2.VideoCapture(0)
    print ("Saving Photo")
    picname = datetime.now().strftime("%y-%m-%d-%H-%M")
    picname = picname+'.jpg'
    cv2.imwrite(picname, frame)
    print ("Sending email")

    attach = picname
    msg = MIMEMultipart()

    msg['From'] = gmail_user
    msg['To'] = to
    msg['Subject'] = subject

    msg.attach(MIMEText(text))

    part = MIMEBase('application', 'octet-stream')
    part.set_payload(open(attach, 'rb').read())
    encoders.encode_base64(part)
    part.add_header('Content-Disposition', 'attachment;
filename="%s"' % os.path.basename(attach))
    msg.attach(part)

```

```

mailServer = smtplib.SMTP("smtp.mail.yahoo.com", 587)
mailServer.ehlo()
mailServer.starttls()
mailServer.ehlo()
mailServer.login(gmail_user, gmail_pwd)
mailServer.sendmail(gmail_user, to, msg.as_string())
# Should be mailServer.quit(), but that crashes...
mailServer.close()
print ("Email Sent")
os.remove(picname)
# Nyalakan relay 21
GPIO.output(21, GPIO.HIGH)
# Nyalakan buzzer 20
GPIO.output(20, GPIO.HIGH)

```

else:

```

# Matikan relay 21
GPIO.output(21, GPIO.LOW)
# Matikan Buzzer 20
GPIO.output(21, GPIO.LOW)
# sensor api 4
Api4_level = ReadChannel(Api4_channel)
Api4_volts = ConvertVolts(Api4_level,2)
if Api4_volts>4:
    ser.write('AT+CMGS="085649262967"\r')
    time.sleep(1)
    ser.write('***Awat ada api di rumah anda***')
    ser.write(chr(26))

cap = cv2.VideoCapture(0)
ret, frame = cap.read()
cap = cv2.VideoCapture(0)
print ("Saving Photo")
picname = datetime.now().strftime("%y-%m-%d-%H-%M")
picname = picname+'.jpg'
cv2.imwrite(picname, frame)
print ("Sending email")

```

```

attach = picname
msg = MIMEMultipart()

msg['From'] = gmail_user
msg['To'] = to
msg['Subject'] = subject

msg.attach(MIMEText(text))

part = MIMEBase('application', 'octet-stream')
part.set_payload(open(attach, 'rb').read())
encoders.encode_base64(part)
part.add_header('Content-Disposition', 'attachment;
filename="%s" % os.path.basename(attach))
msg.attach(part)

mailServer = smtplib.SMTP("smtp.mail.yahoo.com", 587)
mailServer.ehlo()
mailServer.starttls()
mailServer.ehlo()
mailServer.login(gmail_user, gmail_pwd)
mailServer.sendmail(gmail_user, to, msg.as_string())
# Should be mailServer.quit(), but that crashes...
mailServer.close()
print ("Email Sent")
os.remove(picname)
# Nyalakan relay 21
GPIO.output(21, GPIO.HIGH)
# Nyalakan buzzer 20
GPIO.output(20, GPIO.HIGH)

else:
# Matikan relay 21
GPIO.output(21, GPIO.LOW)
# Matikan Buzzer 20
GPIO.output(21, GPIO.LOW)
# sensor api 5
Api5_level = ReadChannel(Api5_channel)
Api5_volts = ConvertVolts(Api5_level,2)

```

```

if Api5_volts>4:
    ser.write('AT+CMGS="085649262967"\r')
    time.sleep(1)
    ser.write('***Awat ada api di rumah anda***')
    ser.write(chr(26))

    cap = cv2.VideoCapture(0)
    ret, frame = cap.read()
    cap = cv2.VideoCapture(0)
    print ("Saving Photo")
    picname = datetime.now().strftime("%y-%m-%d-%H-%M")
    picname = picname+'.jpg'
    cv2.imwrite(picname, frame)
    print ("Sending email")

    attach = picname
    msg = MIMEMultipart()

    msg['From'] = gmail_user
    msg['To'] = to
    msg['Subject'] = subject

    msg.attach(MIMEText(text))

    part = MIMEBase('application', 'octet-stream')
    part.set_payload(open(attach, 'rb').read())
    encoders.encode_base64(part)
    part.add_header('Content-Disposition', 'attachment;
filename="%s"' % os.path.basename(attach))
    msg.attach(part)

    mailServer = smtplib.SMTP("smtp.mail.yahoo.com", 587)
    mailServer.ehlo()
    mailServer.starttls()
    mailServer.ehlo()
    mailServer.login(gmail_user, gmail_pwd)
    mailServer.sendmail(gmail_user, to, msg.as_string())
    # Should be mailServer.quit(), but that crashes...
    mailServer.close()

```

```

print ("Email Sent")
os.remove(picname)
# Nyalakan relay 21
GPIO.output(21, GPIO.HIGH)
# Nyalakan buzzer 20
GPIO.output(20, GPIO.HIGH)

```

else:

```

# Matikan relay 21
GPIO.output(21, GPIO.LOW)
# Matikan Buzzer 20
GPIO.output(21, GPIO.LOW)

```

```

sensorstate =GPIO.input(GPIO_IR)
if sensorstate==0:
    GPIO.output(20,1)

```

else:

```

    GPIO.output(20,0)

```

```

if sensorstate==1 and laststate==0:
    print ("tidak ada asap")
    time.sleep(1)

```

else:

```

    print ("ada asap")
    time.sleep(1)

```

Print out results

```

print ("-----")
print("mq-7: {} (Volt:){V} (konsentrasi:){} ppm)".format(MQ7_level,MQ7_volts, MQ7_ppm))
print("mq-5: {} (Volt:){V} (konsentrasi:){} ppm)".format(MQ5_level,MQ5_volts, MQ5_ppm))
print("sensor Api1 {} (Volt:){V} )".format(Api1_level,Api1_volts))
print("sensor Api2 {} (Volt:){V} )".format(Api2_level,Api2_volts))
print("sensor Api3 {} (Volt:){V} )".format(Api3_level,Api3_volts))
print("sensor Api4 {} (Volt:){V} )".format(Api4_level,Api4_volts))
print("sensor Api5 {} (Volt:){V} )".format(Api5_level,Api5_volts))

```

Wait before repeating loop

```

time.sleep(delay)

```