



PROTEK AKHIR - VE 180626

**PENERAPAN *HUMAN MACHINE INTERFACE* PADA
MESIN *FILTER OLI VTM-80* MENGGUNAKAN
RASPBERRY PI.**

Muhammad Amin Rais
NRP 10311600000040

Dosen Pembimbing
Ir. Joko Susila, M.T.
Ciptian Wieried Priananda, S.ST., M.T.

Pembimbing Lapangan
Ir. Yusup

**Departemen Teknik Elektro Otomasi
Fakultas Vokasi
Institut Teknologi Sepuluh Nopember
Surabaya 2019**



FINAL PROJECT - VE 180626

***APPLICATION OF HUMAN MACHINE INTERFACE IN
OIL FILTERING MACHINES VTM-80 USING
RASPBERRY PI.***

Muhammad Amin Rais
NRP 10311600000040

Supervisor
Ir. Joko Susila, M.T.
Ciptian Wieried Priananda, S.ST., M.T.

Field Guide
Ir. Yusup

***AUTOMATION ELECTRICAL ENGINEERING DEPARTEMENT
Faculty of Vocations
Institut Teknologi Sepuluh Nopember
Surabaya 2019***

PERNYATAAN KEASLIAN PROYEK AKHIR

Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan Proyek Akhir saya dengan judul "**Penerapan Human Machine Interface Pada Mesin Filter Oli VTM-80 Menggunakan 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, 15 Maret 2019



Muhammad Amin Rais
NRP 10311600000040

---Halaman ini sengaja dikosongkan---

**PENERAPAN HUMAN MACHINE INTERFACE PADA MESIN
FILTER OLI VTM-80 MENGGUNAKAN RASPBERRY PI**

PROYEK AKHIR

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

Menyetujui:

Pembimbing 1,

Pembimbing 2,

Ir. Joko Susila, M.T.

NTP. 196606061991021001

Optian Wericd Prananda, S.ST, M.T.

NPP. 1990201711060

Pembimbing Lapangan

Ir. Yusup

**SURABAYA
JULI, 2019**

---Halaman ini sengaja dikosongkan---

PENERAPAN *HUMAN MACHINE INTERFACE* PADA MESIN FILTER OLI VTM-80 MENGGUNAKAN *RASPBERRY PI*.

Nama Mahasiswa : Muhammad Amin Rais
NRP : 10311500010040
Dosen Pembimbing I : Ir. Joko Susila, M.T.
NIP : 196606061991021001
Dosen Pembimbing II : Ciptian Weried Priananda, S.ST.,M.T.
NIP : 1990201711060
Pembimbing Lapangan : Ir. Yusup

ABSTRAK

Di era industri sekarang ini, sebagian besar produksinya dilakukan oleh mesin. Agar mesin-mesin tersebut dapat dioperasikan oleh manusia, maka dibutuhkan penghubung yang mengartikan bahasa mesin tersebut pada tampilan yang dapat dipahami oleh manusia yang disebut HMI atau *Human Machine Interface*. HMI ini bertujuan untuk memudahkan operator dalam menggunakan mesin tersebut. Seperti pada mesin *filter* oli VTM-80.

Perancangan HMI dari mesin *filter* oli hidrolik dengan menggunakan sistem vakum ini menggunakan Raspberry Pi dan LCD layar sentuh sebagai komponen utamanya. HMI ini berisikan gambaran dari kondisi *plant* VTM-80 tersebut, kontrol untuk memulai dan menghentikan sistem serta nilai aktual pembacaan-pembacaan didalam *plant* VTM-80.

HMI yang telah dibuat dilakukan pengujian terhadap pembacaan kondisi ON/OFF komponen-kompnen *plant* pembacaan nilai aktual kadar air dan temperatur, dan dapat menampilkan perubahan-perubahan tersebut pada LCD. Pembacaan nilai aktual kadar air dan temperatur dilakukan melalui program kontrol otomatis dengan rata rata *error* sebesar 1,46% untuk pembacaan temperatur dan 5.0% untuk pembacaan temperatur.

Kata kunci : VTM-80, *Human Machine Interface*, Bahasa Pemograman Python

---Halaman ini sengaja dikosongkan---

APPLICATION OF HUMAN MACHINE INTERFACE IN OIL FILTER MACHINES VTM-80 USING RASPBERRY PI.

Student's Name : Muhammad Amin Rais
Registration Number : 10311500010040
Supervisor I : Ir. Joko Susila, M.T.
ID : 196606061991021001
Supervisor II : Ciptian Weried Priyananda, S.ST.,M.T.
ID : 1990201711060
Supervisor III : Ir. Yusup

ABSTRACT

In the current industrial era, most of the production is done by machines. In order for these machines to be operated by humans, a link is needed that defines the language of the machine on a display that can be understood by humans called HMI or Human Machine Interface. This HMI aims to make it easier for operators to use the machine. Like in the oil filtering machine VTM-80.

Design of the HMI from the engine Hydraulic oil filter using this vacuum system uses the Raspberry Pi and LCD touch screen as its main components. This HMI contains an overview of the condition of the VTM-80 plant, the dick to start and stop the system and the actual value of readings in the plant.

HMI that has been made is tested to read the condition of ON / OFF components of the plant reading the actual value of the water content and temperature, and can display these changes on the LCD. The reading of the actual water content and temperature is done through an automatic control program with an average error of 1.46% for temperature readings and 5.0% for temperature readings.

Keywords : VTM-80, Human Machine Interface, Python programming language

---Halaman ini sengaja dikosongkan---

KATA PENGANTAR

Puji syukur penulis panjatkan kehadiran Allah SWT yang selalu memberikan rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan Proyek Akhir.

Proyek Akhir ini merupakan sebagian syarat kelulusan di Departemen Teknik Elektro Otomasi, bidang studi Elektro Otomasi, Fakultas Vokasi, Institut Teknologi Sepuluh Nopember Surabaya.

Pada kesempatan ini penulis menyampaikan terima kasih atas segala doa, bantuan dan dukungannya yang telah diberikan selama proses pembuatan Proyek Akhir ini kepada :

1. Kedua orang tua yang senantiasa mendoakan dan memberikan dukungan dengan tulus tiada henti.
2. Bapak Ir. Joko Susila, M.T. selaku dosen pembimbing pertama.
3. Bapak Ciptian Wieried Priananda, S.ST.,M.T selaku dosen pembimbing kedua.
4. Semua pihak yang telah membantu baik secara langsung maupun tidak langsung dalam proses penyelesaian Proyek Akhir ini.

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

Surabaya, 30 Juli 2018

Penulis

---Halaman ini sengaja dikosongkan---

DAFTAR ISI

COVER.....	i
COVER DALAM.....	iii
PERNYATAAN KEASLIAN PROYEK AKHIR ...	Error! Bookmark not defined.
ABSTRAK	ix
ABSTRACT	xi
KATA PENGANTAR.....	xiii
DAFTAR ISI	xv
DAFTAR GAMBAR.....	xix
DAFTAR TABEL	xxiii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Permasalahan	2
1.3 Batasan Masalah	2
1.4 Tujuan	2
1.5 Sistematika Laporan.....	3
1.6 Relevansi.....	3
BAB II TEORI PENUNJANG.....	5
2.1 Oli Hidrolik	5
2.1.1 Kontaminasi Air dalam Oli	5
2.1.2 Metode untuk Menentukan Kadar Air Dalam Oli	6
2.1.3 Jumlah Kadar Air Dalam Berbagai Oli	6
2.2 Vacuum Technology Oil Purifrier (VTM-80 SERIES) [1] 8	
2.2.1 Heater (Pemanas).....	9
2.2.2 Vacuum Chamber (Tabung Vakum)	10
2.2.3 Chamber Filter Unit [1]	10
2.2.4 Vacuum Pump (Pompa Vakum).....	11
2.2.5 Motor listrik 3 Fasa [5]	12
2.2.6 Relay 24 V.....	12
2.3 Raspberry Pi	13
2.4 LCD Touchscreen	15
2.5 Bahasa Pemrograman Python	15
2.5.1 Modul Tkinter	15
2.5.2 Modul pyserial	15

2.6	<i>Arduino</i>	15
2.6.1	Perangkat Keras <i>Arduino Nano</i>	16
2.6.2	Perangkat Lunak <i>Arduino IDE</i>	17
2.7	Modul <i>Relay 5V</i>	18
2.8	<i>Analog to Digital Converter</i>	20
2.9	Sensor Kadar Air dalam Oli <i>SHT20</i>	20
2.10	Sensor Temperatur <i>DS18B20</i>	21
2.11	Module <i>ADS1115</i>	22
2.12	LED (<i>Light Emitting Diode</i>)	24
2.13	<i>Switch</i>	24
BAB III PERANCANGAN SISTEM		27
3.1	Cara Kerja Alat	27
3.2	Blok Fungsional Sistem	31
3.3	Perancangan <i>Hardware</i>	32
3.3.1	Rangkaian <i>Arduino Nano</i> pada <i>Raspberry Pi</i>	32
3.3.2	Rangkaian <i>Raspberry Pi</i> pada LCD.....	33
3.3.3	Rangkaian Komunikasi 2q <i>Arduino Nano</i>	33
3.3.4	Rangkaian Sensor Kadar Air dalam Oli <i>SHT 20</i>	34
3.3.5	Rangkaian Sensor Temperatur <i>Heater DS18B20</i>	35
3.3.6	Rangkaian <i>Relay</i> pada <i>Arduino Nano 1</i>	36
3.3.7	Rangkaian Module <i>Relay</i> pada <i>Arduino Nano 1</i>	37
3.3.8	Rangkaian Module <i>Relay</i> pada <i>Arduino Nano 2</i>	38
3.3.9	Konfigurasi <i>Hardware</i> Pengontrol <i>Heater</i>	39
3.3.10	Konfigurasi <i>Hardware</i> pada <i>Vacuum Chamber</i>	39
3.3.11	Konfigurasi <i>Hardware</i> pada <i>Filter Unit</i>	40
3.3.12	Konfigurasi <i>Hardware</i> pada <i>Vacuum Pump</i>	40
3.3.13	Konfigurasi <i>Hardware</i> pada <i>Oil Pump</i>	41
3.3.14	Konfigurasi <i>Hardware</i> pada <i>Evaquation Pump</i>	42
3.3.15	Konfigurasi <i>Hardware</i> pada <i>Directional Valve</i>	42
3.3.16	Perancangan <i>Miniature Plan</i>	43
3.3.17	Rangkaian <i>Miniature plan</i>	43
3.3.18	Perancangan Rangkaian Pengganti Sensor Kadar Air dengan Modul <i>ADS1115</i>	46
3.3.19	Perancangan Rangkaian Sensor Temperatur <i>Heater</i> <i>DS18B20</i>	47
3.3.20	Perancangan Rangkaian <i>Hardware</i> Menggunakan LED	47

3.3.21	Perancangan Rangkaian <i>Hardware</i> Menggunakan <i>Switch</i>	49
3.4	Perancangan <i>Software</i>	50
3.4.1	Perancangan Tampilan HMI	50
3.4.2	Algoritma HMI VTM-80	54
3.4.3	Algoritma Menu <i>Input</i> Pada HM	56
3.4.4	Algoritam Komunikasi Serial <i>Arduino</i> dan <i>Raspberry Pi</i>	59
BAB IV PENGUJIAN ANALISIS		61
4.1	Pengujian <i>Software</i>	61
4.1.1	Pengujian Tampilan Utama HMI	61
4.1.2	Pengujian Tampilan Menu <i>Input</i> HMI	63
4.1.3	Tampilan HMI Saat Mesin Bekerja Normal	63
4.1.4	Tampilan HMI Saat Mesin Bekerja Dengan Kondisi <i>Error</i>	71
4.2	Pengujian Nilai Kadar Air pada Sensor SHT-20	75
4.3	Pengujian Nilai Temperatur DS18B20	78
BAB V PENUTUP		81
4.4	Kesimpulan	81
4.5	Saran	81
DAFTAR PUSTAKA		83
LAMPIRAN A		A-1
LAMPIRAN B		B-1
RIWAYAT PENULIS		1

---Halaman ini sengaja dikosongkan---

DAFTAR GAMBAR

Gambar 2.1	Perbedaan Kontaminasi Air Dalam Oli	6
Gambar 2.2	Estimasi Sederhana Dari Kelembaban Relatif Oli.....	7
Gambar 2.3	VTM-80.....	8
Gambar 2.4	Proses Diagram Kerja VTM-80.....	9
Gambar 2.5	<i>Heater</i>	9
Gambar 2.6	<i>Vacuum Chamber</i>	10
Gambar 2.7	<i>Filter Unit</i>	11
Gambar 2.8	<i>Vacuum Pump</i>	11
Gambar 2.9	Motor listrik 3 fasa	12
Gambar 2.10	<i>Relay 24 V</i>	13
Gambar 2.11	<i>GPIO Raspberry Pi 3B</i>	14
Gambar 2.12	<i>Arduino Nano</i>	16
Gambar 2.13	Tampilan IDE <i>Arduino</i>	17
Gambar 2.14	<i>Module Relay</i>	19
Gambar 2.15	Rangkaian dalam <i>Relay</i>	19
Gambar 2.16	Sensor Kadar Air SHT20.....	21
Gambar 2.17	Sensor Temperatur DS18B20	22
Gambar 2.18	Modul ADS1115.....	23
Gambar 2.19	<i>Input dan Output (Datasheet ADS1115)</i>	23
Gambar 2.20	Struktur LED	24
Gambar 2.21	<i>Switch</i>	25
Gambar 3.1	VTM-80.....	27
Gambar 3.2	Komponen-Komponen VTM-80 (<i>Landscape</i>)0	28
Gambar 3.3	Blok Fungsional Sistem (<i>Landscape</i>).....	29
Gambar 3.4	Rangkaian <i>Arduino Nano</i> pada <i>Raspberry Pi</i>	32
Gambar 3.5	Konfigurasi <i>Arduino Nano</i> pada HMI	33
Gambar 3.6	Komunikasi Dua <i>Arduino Nano</i>	34
Gambar 3.7	Rangkaian Sensor Kadar Air dalam Oli SHT 20	35
Gambar 3.8	Rangkaian Sensor Temperatur <i>Heater</i> DS18B20	36
Gambar 3.9	Rangkaian <i>Relay</i> pada <i>Arduino Nano 1</i>	36
Gambar 3.10	Rangkaian <i>Module Relay</i> pada <i>Arduino Nano 1</i>	37
Gambar 3.11	Rangkaian <i>Module Relay</i> pada <i>Arduino Nano 2</i>	38
Gambar 3.12	Konfigurasi <i>Hardware</i> Pengontrol <i>Heater</i>	39
Gambar 3.13	Konfigurasi <i>Hardware</i> pada <i>Vacuum Chamber</i>	40

Gambar 3.14	Konfigurasi <i>Hardware</i> pada <i>Filter Unit</i>	40
Gambar 3.15	Konfigurasi <i>Hardware</i> pada <i>Vacuum Pump</i>	41
Gambar 3.16	Konfigurasi <i>Hardware</i> pada <i>Oil Pump</i>	41
Gambar 3.17	Konfigurasi <i>Hardware</i> pada <i>Evaquation Pump</i>	42
Gambar 3.18	Konfigurasi <i>Hardware</i> pada <i>Directional Valve</i>	43
Gambar 3.19	Rangkaian Skematik <i>Miniature Plan</i>	44
Gambar 3.20	Pengganti Rangkaian Sensor Kadar Air ke Arduino Nano.	47
Gambar 3.21	Rangkaian <i>Hardware</i> Menggunakan LED	48
Gambar 3.22	Rangkaian <i>Hardware</i> Menggunakan Switch	50
Gambar 3.23	Perancangan Tampilan Menu Utama HMI VTM-80.....	51
Gambar 3.24	Perancangan Tampilan Menu <i>Input</i>	51
Gambar 3.25	<i>Algoritma</i> Pemrograman HMI VTM-80.....	54
Gambar 3.26	<i>Algoritma</i> Menu <i>Input</i>	57
Gambar 3.27	<i>Algoritma</i> Komunikasi Serial Arudino Dengan Raspberry <i>PI</i>	58
Gambar 4.1	Tampilan Menu Utama HMI	62
Gambar 4.2	Tampilan Menu <i>Input</i> Kadar Air	63
Gambar 4.3	Tampilan HMI Saat <i>Oil Pump</i> ON	65
Gambar 4.4	Tampilan HMI Saat <i>Oil Pump</i> dan <i>Heater</i> Mati.....	65
Gambar 4.5	Tampilan HMI Saat Kondisi <i>Level 1</i>	66
Gambar 4.6	Tampilan HMI Saat Kondisi <i>Level 1</i> dan <i>Heater</i> Mati....	67
Gambar 4.7	Tampilan HMI Saat Kondisi <i>Level 2</i>	68
Gambar 4.8	Tampilan HMI Saat Kondisi <i>Level 2</i> dan <i>Heater</i> Mati....	68
Gambar 4.9	Tampilan HMI Saat Pengecekan	69
Gambar 4.10	Tampilan HMI Saat Kondisi Pengecekan dan <i>Heater</i> Mati	69
Gambar 4.11	Tampilan HMI Saat Pengosongan Oli di <i>Vacuum</i> <i>Chamber</i>	70
Gambar 4.12	Tampilan HMI Saat Sistem Selesai Bekerja.....	70
Gambar 4.13	Tampilan HMI Dengan <i>Contition Level 1 Error</i>	72
Gambar 4.14	Tampilan HMI Dengan <i>Contition : Vacuum Chamber</i> <i>Overlaod</i>	73
Gambar 4.15	Tampilan HMI Dengan <i>Contition : Filter Clogged</i>	75
Gambar 4.16	Pengujian Sensor SHT20.....	76
Gambar 4.17	Alat FCU1000	76

Gambar 4.18 Pengujian Sensor DS18B20 79

---Halaman ini sengaja dikosongkan---

DAFTAR TABEL

Tabel 2.1	Tingkat air <i>dissolved</i> , <i>emulsified</i> dan <i>free</i> dalam oli	7
Tabel 2.2	Spesifikasi Sensor SHT20	21
Tabel 2.3	Spesifikasi Sensor DS18B20	22
Tabel 3.1	Rangkaian <i>Arduino Nano</i> pada <i>Raspberry Pi</i>	32
Tabel 3.2	Pin Modul Sensor Kadar Air dalam Oli SHT 20 Dengan <i>Arduino Nano</i>	34
Tabel 3.3	Pin Rangkaian Temperatur <i>Heater</i> DS18B20 Dengan <i>Arduino Nano</i>	35
Tabel 3.4	Pin Rangkaian <i>Relay</i> pada <i>Arduino Nano</i> 1	37
Tabel 3.5	Pin Rangkaian Module <i>Relay</i> pada <i>Arduino Nano</i> 1	38
Tabel 3.6	Pin Rangkaian Module <i>Relay</i> pada <i>Arduino Nano</i> 2	39
Tabel 3.7	Komponen Miniature Plan	44
Tabel 3.8	Pin Pengganti Rangkaian Sensor Kadar Air Dengan <i>Arduino Nano</i>	46
Tabel 3.10	Pin Rangkaian <i>Hardware</i> Menggunakan <i>Switch</i> pada <i>Arduino Nano</i>	49
Tabel 4.1	Pengujian Kondisi Komponen Plant VTM-80 Saat Sistem Bekerja Dengan Normal.....	64
Tabel 4.2	Pengujian Kondisi Komponen Plant VTM-80 Saat Sistem Mengalami <i>Error</i> Pada <i>Switch Level</i> 1.....	71
Tabel 4.3	Pengujian Kondisi Komponen Plant VTM-80 Saat <i>Vacuum Chamber Overload</i>	73
Tabel 4.4	Pengujian Kondisi Komponen Plant VTM-80 Saat <i>Filter</i> Tersumbat.....	74
Tabel 4.5	Hasil Pengujian Nilai Kadar Air Sensor SHT-20 Dengan Alat Ukur FCU-1000.....	77
Tabel 4.6	Hasil Kalibrasi Sensor DS8B20	78

---Halaman ini sengaja dikosongkan---

BAB I

PENDAHULUAN

1.1 Latar Belakang

Dalam teknologi industri modern ini, sebagian besar produksinya dilakukan oleh mesin. Karena dengan bantuan mesin hasil produksi dapat ditingkatkan dan kebutuhan akan suatu produk lebih mudah untuk dipenuhi. Agar mesin tersebut dapat melakukan hal-hal yang kita inginkan, dibutuhkanlah suatu sistem yang dapat mengartikan perintah-perintah tersebut ke bahasa yang dapat dimengerti oleh mesin. Sistem ini dinamakan HMI atau *Human Machine Interface*. HMI ini bertujuan untuk memudahkan operator dalam menggunakan mesin yang akan dioperasikannya.

Berdasarkan pada inovasi alat “*Vacuum Technology Oil Purifier (VTM-80 SERIES)*” yang dimana berguna untuk memfilter oli dari partikel air, dan debu [1]. Menindak lanjuti pentingnya alat ini di sini saya menerapkan HMI (*Human Machine Interface*) agar alat ini dapat dioperasikan dengan mudah. Dalam dunia industri penggunaan oli avtur sangat sering digunakan, kegunaan oli ini sangat penting karena tidak hanya sebagai tenaga mesin akan tetapi juga sebagai pelumas pada mesin tersebut. Oli yang digunakan pada industri adalah oli hidrolik, yang dimana oli ini sendiri berguna untuk tenaga mesin dan pelumas. Prinsip kerja alat ini dengan cara memproses oli dengan menyedot. Oli akan melewati *high magnetic Filter* untuk menarik partikel besi. Kemudian oli akan dipanaskan pada suhu tertentu dan divakumkan dengan *Vacuum Chamber* sehingga titik didih air tercapai tanpa merusak oli tersebut. Uap air yang terbentuk akan dikeluarkan dan oli yang telah terpisah akan di *Filter* dengan *Filter unit* untuk memisahkan dengan partikel debu dan logam yang lain.

Proses alat ini sendiri masih manual dikarenakan tidak ada suatu kondisi yang dibaca utamanya kondisi saat pemisahan air dengan oli. Saat pemanasan suhu harus diatur sekaligus dengan kondisi tekanan tertentu sehingga titik didih air tercapai dulu, tidak hanya saat 100 °C pada tekanan 76 mmHg (1 Bar) [2]. Dan titik didih sebuah cairan juga sangat dipengaruhi oleh besarnya tekanan lingkungan sekitar. Semakin tinggi tekanan lingkungan sekitar maka akan semakin tinggi pula titik didihnya, begitu pula sebaliknya. Sedangkan jika berada pada tekanan yang sama namun cairannya berbeda akan terlihat pada perbedaan suhu saat

mencapai titik didih [2]. Dan operasinya masih manual dikarenakan kondisi titik didih dilihat melalui uap yang telah tercipta dari proses vakum. Dengan menggunakan layar sentuh, HMI dapat lebih mudah untuk digunakan, karena kita dapat mengatur hal-hal apa saja yang bisa dioperasikan pada saat-saat tertentu dengan hanya menampilkan perintah-perintah yang berhubungan pada saat itu juga.

1.2 Permasalahan

Permasalahan yang dibahas dalam Proyek Akhir ini adalah sistem peoperasian pada mesin *filter* Oli VTM-80 yang masih manual dan memerlukan *operator* khusus untuk menjalankan mesin tersebut karena pengoprasian sistem kontrolnya cukup rumit dan menggunakan beberapa modul modul kotrol yang berbeda beda. Dengan menggunakan HMI, seluruh modul control tersebut dapat disatukan menjadi satu tampilan sehingga lebih mudah di operasikan.

Selain itu penggunaan *Raspberry Pi* dan *LCD touchscreen* sebagai HMI terbilang lebih murah dibandingkan dengan menggunakan PLC dan HMI *touchscreen*-nya.

1.3 Batasan Masalah

Batasan masalah yang dibahas dalam Proyek Akhir ini adalah sebuah mesin *filter* oli VTM-80 SERIES dimana pengoperasiannya masih masih cukup rumit sehingga membutuhkan operator khusus dalam mengoprasikan mesin:

1. VTM-80 memfilter oli dari logam, air, dan debu, akan tetapi variable yang dibaca dalam sistem ini adalah kadar air dalam oli tersebut.
2. Alat yang digunakan untuk HMI adalah *Raspberry Pi* dengan system operasi *Raspbian*
3. HMI yang dibuat hanya sebagai pemermudah komunikasi antara Operator dengan Sistem kontrol
4. Data yang ditampilkan pada HMI berasal dari sistem kontrol, bukan dari sensor langsung.
5. Sistem kontrol menggunakan 2 buah *Arduino Nano* yang dibahas pada laporan lain.

1.4 Tujuan

Tujuan yang akan dicapai dari Proyek Akhir ini adalah sebagai berikut:

1. Memudahkan pengorasian mesin VTM-80 dengan HMI
2. Memonitor variabel temperatur dan variabel jumlah campuran air pada HMI.
3. Merancang sistem HMI menggunakan *Raspberry Pi*

1.5 Sistematika Laporan

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

Bab I Pendahuluan

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

Bab II Teori Penunjang

Bab ini menjelaskan tentang tinjauan pustaka dan teori - teori penunjang yang dijadikan prinsip dasar dan mendukung dalam perencanaan dan pembuatan alat.

Bab III Perancangan Sistem

Bab ini membahas perencanaan dan pembuatan alat yang meliputi perencanaan dan pembuatan perangkat keras (*Hardware*) dan perangkat lunak (*Software*).

Bab IV Pengujian dan Analisis

Bab ini membahas hasil dan analisis perangkat keras (*Hardware*) dan perangkat lunak (*Software*).

Bab V Penutup

Bab ini berisi kesimpulan dan saran dari hasil pembahasan yang telah diperoleh.

1.6 Relevansi

Tugas yang dikerjakan hanya berupa *monitoring* pada sistem mesin *Filter* oli VTM-80 yang berupa parameter parameter yang membuat mesin ini dapat bekerja dengan baik dan tidak termasuk pada sistem kendali otomatis,. Untuk campuran oli yang di *monitoring* hanya berupa campuran air.

---Halaman ini sengaja dikosongkan---

BAB II

TEORI PENUNJANG

2.1 Oli Hidrolik

Salah satu penggunaan pelumas paling utama adalah oli mesin yang dipakai pada mesin pembakaran dalam. Sistem pelumasan ini memiliki beberapa fungsi dan tujuan Mengurangi gesekan serta mencegah keausan dan panas, dengan cara oli dapat membentuk suatu lapisan tipis (*oil film*) untuk mencegah kontak langsung permukaan logam dengan logam. Sebagai bahan pembersih, yaitu dengan mengeluarkan kotoran pada bagianbagian mesin. Mencegah karat pada bagian-bagian mesin. Mencegah terjadinya kebocoran gas hasil pembakaran [2].

2.1.1 Kontaminasi Air dalam Oli

Air dalam sistem pelumasan dapat berasal dari lingkungan, seperti hujan atau kelembaban di udara. *Gasket* yang rusak pada penutup *reservoir* , *reservoir* udara berkinerja buruk, atau *wiper* yang rusak pada silinder hidrolik juga merupakan sumber yang mungkin. Kondensasi udara di reservoir oli karena perbedaan suhu antara siang dan malam akan mengubah kelembaban di udara menjadi tetesan air, bercampur dengan oli. Sistem pendingin berbasis air yang rusak dalam peralatan uap adalah sumber tetesan air juga.

Air yang terkontaminasi dalam oli terdapat tiga jenis yaitu : *dissolved*, *emulsified* and *free*. Di bawah tingkat kejenuhan, molekul-molekul air tersebar bersama molekul-molekul oli, menghasilkan air dalam minyak yang tidak terlihat. Ini dikenal sebagai air terlarut, kondisi air paling tidak berbahaya untuk sistem pelumas. Ketika jumlah air terlarut melebihi titik jenuh, oli tidak lagi mampu menyerap lebih banyak molekul air, menghasilkan air *emulsified*. Ini ditandai dengan penampilan minyak yang kabur atau berawan. Peningkatan kadar air lebih lanjut dalam minyak akan menghasilkan tingkat terpisah antara minyak dan pembentukan air. Keadaan ini dikenal sebagai air *free*. Karena kepadatannya yang lebih tinggi, air membentuk lapisan bawah, mengendap di dasar bak, dengan oli mengambang di atas. Namun, air yang diemulsi juga akan hadir dalam fase menengah, terus bersirkulasi dalam sistem pelumas. Gambar 2.1 menunjukkan perbedaan yang terlihat antara air *dissolved*, *emulsified* and *free* dalam sampel Oli [3].



Gambar 2.1 Perbedaan Kontaminasi Air Dalam Oli[3]

Efek air dalam minyak ada dua, menghancurkan baik sifat fisik, kimia dan karakteristik oli. dan dapat menyebabkan kerusakan komponen mesin.

2.1.2 Metode untuk Menentukan Kadar Air Dalam Oli

Oli dalam minyak dapat diukur dengan 3 cara; di tempat dengan pengambilan sampel, di laboratorium, dan *On-line* dalam waktu nyata. untuk menentukan kadar air minyak dan di luar cakupan makalah ini untuk membahas ini sebagai pilihan metode mana yang akan digunakan adalah dengan *On-line* (Pengukuran *Real Time*) memakai *moisture sensor* (sensor kelembapan). Dengan meningkatnya masuknya air secara kritis, pemantauan air dalam oli yang berkelanjutan mungkin diperlukan. Sensor kelembaban dapat dipasang pada mesin yang membutuhkan pemantauan selama mesin berjalan [3].

2.1.3 Jumlah Kadar Air Dalam Berbagai Oli

Mengatur *level* alarm untuk air dalam minyak sangat penting untuk keandalan alat berat. Pembentukan *level*, dikombinasikan dengan pengujian pada interval yang tepat, akan memungkinkan pengguna akhir untuk bertindak cepat jika kenaikan air tiba-tiba terdeteksi.

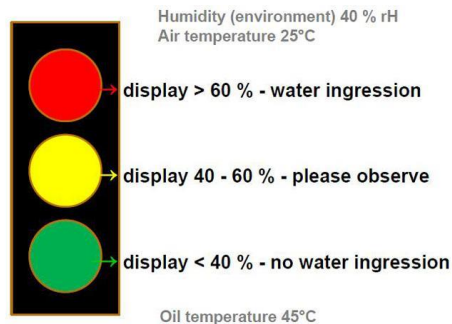
Guna menunjukkan tingkat air *dissolved*, *emulsified* and *free* dalam minyak yang akan diharapkan untuk berbagai jenis oli. Penting untuk mengetahui jenis oli pangkalan oli, paket aditif, suhu dan tekanan operasi. Tabel 2.1 menunjukkan Tingkat air *dissolved*, *emulsified* and *free* untuk berbagai jenis oli^x.

Tabel 2.1 Tingkat air *dissolved*, *emulsified* dan *free* dalam oli

Oli	Dissolved (ppm)	Emulsified (ppm)	Free (ppm)
<i>New hydraulic fluid</i>	0-200	200-1000	>1000
<i>Aged hydraulic fluid</i>	0-600	600-5000	>5000
<i>New R&O Oil</i>	0-150	150-500	>500
<i>Aged R&O Oil</i>	0-500	500-1000	>1000
<i>New crankcase oil</i>	0-2000	2000-5000	>50000

Konsekuensi negatif yang disebabkan oleh air bebas di dalam cairan hidrolik dan pelumas, beberapa produsen dan organisasi standar telah menerbitkan panduan dan membatasi nilai-nilai berkenaan dengan kadar air dslm oli yang diijinkan. Tabel 2.1 menunjukkan beberapa contoh.

Nilai batas hanya diberikan untuk kadar air. Seharusnya, harus dibuktikan bahwa nilai batas tertentu belum terlampaui. Hal ini menyebabkan korelasi antara nilai-nilai sensor kelembaban dalam persen rH (*Relative Humidity*) dan kadar air (ppm) yang ditentukan dan disimpan sebagai "kalibrasi." Sejauh ini, standar untuk tingkat kelembaban relatif maksimum hanya ditunjukkan oleh beberapa produsen *filter* dan sensor kelembaban.



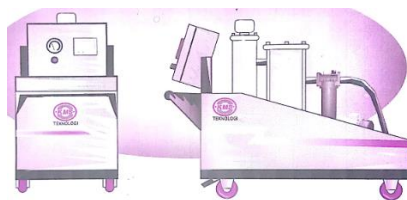
Gambar 2.2 Estimasi Sederhana Dari Kelembaban Relatif Oli[4]

Menyesuaikan spesifikasi, pedoman, dan standar yang ada dengan menunjukkan tingkat maksimum nilai kelembapan/aktivitas air dapat bermakna karena alasan teknis dan ekonomis. Ini secara substansial akan

menyederhanakan penggunaan sensor kelembaban karena akan menjadi tidak perlu untuk menentukan kurva saturasi, dan korelasi matematis dapat diabaikan [4].

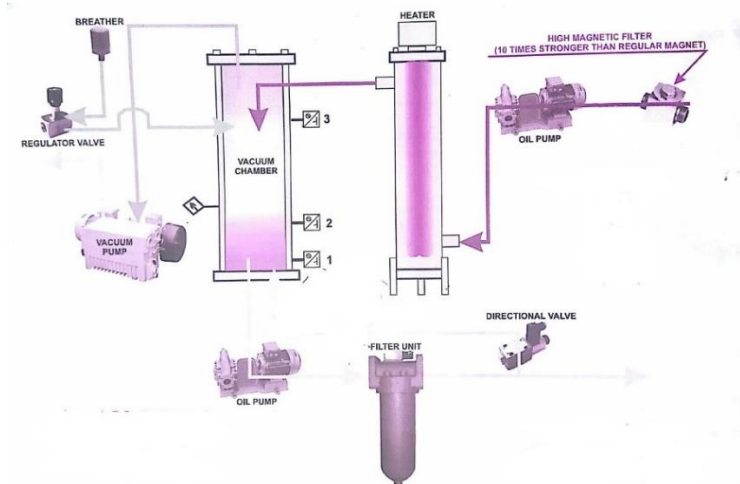
2.2 Vacuum Technology Oil Purifier (VTM-80 SERIES) [1]

Vacuum Technology Oil Purifier (VTM-80) merupakan sebuah alat yang digunakan untuk memfilter oli yang kotor dengan cara pemanasan pada tabung vakum. Dimana oli akan dialirkan pada tabung vakum dan dipanaskan sehingga oli dan air akan terpisah. VTM-80 sendiri masih berupa *prototype*. *prototype* VTM-80 dimana disajikan dalam Gambar 2.3.



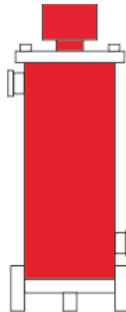
Gambar 2.3 VTM-80[1]

Mekanisme kerja VTM-80 Saat proses dijalankan, motor akan menyedot oli, oli akan melewati *filter* magnetic untuk menyaring campuran logam besi yang ada pada oli tersebut. Setelah itu oli akan memasuki ruang pemanas untuk memanaskan oli. Lalu, oli memasuki ruang vakum untuk memisahkan kandungan air di dalam oli. Dengan bantuan pompa vakum, air dapat menguap lebih cepat karena titik uap air lebih kecil saat tekanan dibawah tekanan atmosfer. Didalam tabung terdapat 3 buah *level Switch* yang berfungsi sebagai sensor untuk mendeteksi volume oli yang terdapat pada tabung vakum, yang selanjutnya informasi tersebut di gunakan untuk mengkontrol *directional valve*. Pada *level 1*, *directional valve* akan membuka ke arah tabung vakum dan pompa oli akan memompa oli kembali ke tabung vakum dengan melewati *Filter* oli. Pada *level 2*, *directional valve* akan membuka ke arah *outputx* sehingga jumlah oli yang terdapat pada tabung vakum akan berkurang dan menuju ke *level 1*. *Level 3*, berfungsi sebagai pengaman apabila *Switch level 2* tidak bekerja. Saat *level 3* terjadi, alarm akan berbunyi dan mesin akan menghentikan pengoperasian. Proses diagram kerja VTM-80 disajikan dalam Gambar 2.4.



Gambar 2.4 Proses Diagram Kerja VTM-80[1]

2.2.1 Heater (Pemanas)



Gambar 2.5 Heater[1]

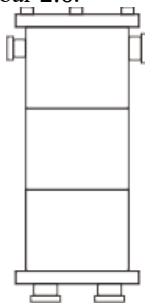
Heater (pemanas) yang merupakan sebuah tangki berbentuk silinder yang berisi fluida dan kemudian dipanaskan oleh piranti *heater* untuk kemudian mengalirkan temperatur panas kepada *section* yang melintasi di tengah – tengah tangki silinder. Fluida yang telah panas

setelah melewati *heater* mampu mengalir menuju tangki pendingin untuk seterusnya bergerak mengalir secara kontinyu. *heater* difungsikan hanya untuk proyeksi dari reaktor nuklir. Karena pada dasarnya, reaktor bekerja akan menghasilkan panas atau proyeksi alat yang bisa bekerja untuk menghasilkan panas. *heater* disajikan pada Gambar 2.5

2.2.2 *Vacuum Chamber* (Tabung Vakum)

Penerapan konstruksi penyalur gas tekanan tinggi maupun untuk keperluan *suplay* fluida bertekanan dari daerah satu ke tempat yang lain banyak menggunakan model konstruksi tabung tekanan tinggi atau tabung vakum (*vacuum chamber*). Cara kerjanya *vacuum pump* mengeluarkan molekul-molekul gas dari dalam tabung vakum. Prinsip dari pompa ini adalah dengan jalan mengekspansi volume ruang pompa sehingga terjadi penurunan tekanan vakum parsial. Jika dilakukan secara berkali-kali, maka vakum terbentuk di ruangan tersebut. *Vacuum chamber* ini memiliki volume 80 liter.

Vacuum chamber yang digunakan didalamnya memiliki ruang untuk udara/uap dan ruang untuk cairan oli itu sendiri. Pada ruang untuk oli didalamnya memiliki tiga *level switch* untuk volume oli tersebut. Dibagian bawah terdapat saluran untuk mengarahkan oli keluar. *Vacuum chamber* disajikan pada Gambar 2.6.



Gambar 2.6 *Vacuum Chamber*[1]

2.2.3 *Chamber Filter Unit* [1]

Filter unit adalah *filter* yang didesain untuk menyaring kotoran yang mengkontaminasi oli mesin, agar oli tetap dalam keadaan bersih untuk melumasi komponen mesin. Oli mesin dapat terkontaminasi dari

berbagai hal, seperti : Serbuk besi/gram besi, *sludge*/deposit oli dan kotoran debu dari udara. *Filter unit* disajikan dalam Gambar 2.7.



Gambar 2.7 *Filter Unit*

(Sumber : <https://www.morebeer.com/products/water-Filter-canister>)

2.2.4 Vacuum Pump (Pompa Vakum)



Gambar 2.8 *Vacuum Pump*

(Sumber: <https://beckerpumps.com/Vacuum-pumps-2/>)

Vacuum pump adalah sebuah alat untuk mengeluarkan molekul-molekul gas dari dalam sebuah ruangan tertutup untuk mencapai tekanan vakum. Prinsip dari pompa ini adalah dengan jalan mengekspansi volume ruang oleh pompa sehingga terjadi penurunan tekanan vakum parsial. Sistem *sealing* mencegah gas masuk ke dalam ruang tersebut. Selanjutnya pompa melakukan gerakan buang, dan kembali mengekspansi ruang

tersebut. Jika dilakukan secara siklis dan berkali-kali, maka vakum akan terbentuk di ruangan tersebut. *vacuum pump* disajikan pada Gambar 2.8.

2.2.5 Motor listrik 3 Fasa [5]

Motor listrik 3 fasa adalah motor yang bekerja dengan memanfaatkan perbedaan fasa pada sumber untuk menimbulkan gaya putar pada bagian rotornya. Perbedaan fasa pada motor 3 phase didapat langsung dari sumber. Hal tersebut yang menjadi pembeda antara motor 1 fasa dengan motor 3 fasa. Secara umum, motor 3 fasa memiliki dua bagian pokok, yakni stator dan rotor. Bagian tersebut dipisahkan oleh celah udara yang sempit atau yang biasa disebut dengan air gap. Jarak antara stator dan rotor yang terpisah oleh air gap sekitar 0,4 milimeter sampai 4 milimeter.

Motor listrik ini digunakan untuk mengalirkan oli ketika menerima kontak dari kontaktor. Pada VTM-80 motor listrik berupa *oil pump* untuk mendorong aliran oli masuk dan *evaquation pump* untuk mendorong aliran oli keluar. Motor Listrik 3 Fasa disajikan pada Gambar 2.9.



Gambar 2.9 Motor listrik 3 fasa

(Sumber : <http://rohmatullah.student.telkomuniversity.ac.id/contoh-penggunaan-listrik-1-fasa-dan-3-fasa/>)

2.2.6 Relay 24 V

Relay adalah komponen elektronika yang berupa saklar atau *Switch* elektrik yang dioperasikan menggunakan listrik. *Relay* juga biasa disebut sebagai komponen electromechanical atau elektromekanikal yang terdiri dari dua bagian utama yaitu coil atau elektromagnet dan kontak saklar. Komponen *relay* menggunakan prinsip elektromagnetik sebagai penggerak kontak saklar, sehingga dengan menggunakan arus listrik yang

kecil atau *low power*, dapat menghantarkan arus listrik yang memiliki tegangan lebih tinggi [7]. Pada *Relay* ini bertegangan 24 V. *Relay* 24 V disajikan pada Gambar 2.10.



Gambar 2.10 *Relay* 24 V[7]

2.3 *Raspberry Pi*

Raspberry Pi merupakan komputer dalam satu *singleboard*. *Chip* mengintegrasikan sebuah prosesor (CPU), *graphics processing unit* (GPU), dan memori pada suatu unit tunggal. Pada pengerjaan Proyek Akhir ini *Raspberry Pi* yang digunakan berupa model 3B [15]. Bagian – bagian pada *Raspberry Pi* ini adalah sebagai berikut :

1. **Prosesor**
Prosesor berupa *chip* 32 bit, 1.2 GHz *System on a Chip* (SOC) dengan merk *Broadcom BCM2837*. Menggunakan empat inti *ARM Cortex-A53* dengan kecepatan 1,2 GHz. *Raspberry Pi* model 3 mempunyai RAM sebesar 1 GB
2. **Slot *Secure Digital Card* (SD Card)**
Raspberry Pi menggunakan *SD Card* sebagai media penyimpanan seluruh data (*hard drive*).
3. **Port USB**
Raspberry Pi Model 3 ini mempunyai 4 port USB sedangkan *Raspberry Pi* Model A hanya mempunyai sebuah port USB.
4. **Port Ethernet, Wifi, dan Bluetooth**
Raspberry Pi model 3 mempunyai port *ethernet* dengan standar RJ45 dan antenna berkecepatan 2.4 GHz 802.11n untuk menangkap sinyal *wifi*. *Raspberry* model 3 juga terdapat *bluetooth* untuk transfer data dan koneksi dengan sebuah *hardware* yang menggunakan koneksi tersebut.
5. **Konektor HDMI (*High-Definition Multimedia Interface*)**

Port HDMI digunakan sebagai penyedia keluaran video dan audio digital. Sinyal HDMI mampu dikonversi menjadi DVI (*Digital Visual Interface*) sehingga dapat digunakan untuk berbagai monitor.

6. Port GPIO

Selain itu terdapat *port General Purpose Input/Output* (GPIO) digunakan untuk berhubungan dengan suatu *hardware eksternal*. *Raspberry Pi* mempunyai 26 pin GPIO. Gambar 2.11. menunjukkan konfigurasi pin pada *port* GPIO.

Pin#	NAME		NAME	Pin#
01	3.3v DC Power		DC Power 5v	02
03	GPIO02 (SDA1 , I ² C)		DC Power 5v	04
05	GPIO03 (SCL1 , I ² C)		Ground	06
07	GPIO04 (GPIO_GCLK)		(TXD0) GPIO14	08
09	Ground		(RXD0) GPIO15	10
11	GPIO17 (GPIO_GEN0)		(GPIO_GEN1) GPIO18	12
13	GPIO27 (GPIO_GEN2)		Ground	14
15	GPIO22 (GPIO_GEN3)		(GPIO_GEN4) GPIO23	16
17	3.3v DC Power		(GPIO_GEN5) GPIO24	18
19	GPIO10 (SPI_MOSI)		Ground	20
21	GPIO09 (SPI_MISO)		(GPIO_GEN6) GPIO25	22
23	GPIO11 (SPI_CLK)		(SPI_CE0_N) GPIO08	24
25	Ground		(SPI_CE1_N) GPIO07	26
27	ID_SD (I ² C ID EEPROM)		(I ² C ID EEPROM) ID_SC	28
29	GPIO05		Ground	30
31	GPIO06		GPIO12	32
33	GPIO13		Ground	34
35	GPIO19		GPIO16	36
37	GPIO26		GPIO20	38
39	Ground		GPIO21	40

Rev. 2
29/02/2016

www.element14.com/RaspberryPi

Gambar 2.11 GPIO *Raspberry Pi* 3B
(Sumber : <http://www.element14.com/RaspberryPi>)

2.4 LCD Touchscreen

Liquid crystal display (LCD) merupakan suatu jenis media tampilan yang menggunakan kristal cair sebagai penampil utama. Jenis LCD yang digunakan merupakan LCD *touchscreen* 7” yang dikeluarkan oleh *Raspberry Foundation*. LCD ini digunakan untuk tampilan layar pada *Raspberry Pi*. LCD yang digunakan dilengkapi *touchscreen* yang nantinya akan digunakan untuk mengakses tombol-tombol yang berada pada tampilan HMI [16].

2.5 Bahasa Pemrograman Python

Python ditemukan oleh Guido van Rossum. Bahasa *python* merupakan bahasa pemrograman yang dikategorikan *highlevel language*. Berbeda dengan *lowlevel language*, *highlevel language* tidak dapat langsung dijalankan oleh mesin, perlu diproses terlebih dahulu agar dapat dijalankan oleh mesin. Pada sisi *interface*, digunakan modul *tkinter* yang merupakan *interface* standar yang biasa digunakan pada pemrograman *python*.

Program atau *script python* dapat langsung dieksekusi, tidak perlu proses *compiling* ke kode mesin. Dalam *python*, pemrograman tidak perlu menegaskan sebuah variabel berupa *number*, *list*, atau *string*. Python merupakan *open source software*. Bahasa pemrograman *python* masih dikembangkan dan diperbaiki [16].

2.5.1 Modul Tkinter

Tkinter merupakan modul *python* untuk pembuatan tampilan atau biasa disebut GUI (*Graphical User Interface*). Modul mempunyai *widget* atau komponen yang dapat diatur dengan kode – kode tertentu untuk menghasilkan tampilan yang dapat dikendalikan dari layar monitor LCD [16].

2.5.2 Modul pyserial

Pyserial merupakan modul *python* untuk mengakses komunikasi *serial* melalui *serial port* [16].

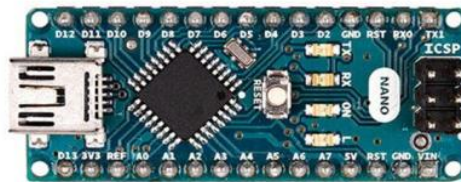
2.6 Arduino

Arduino adalah pengendali mikro *single-board* yang bersifat *open-source*, diturunkan dari *wiring platform*, dirancang untuk memudahkan penggunaan elektronika dalam berbagai bidang. *Hardware* dalam

Arduino memiliki prosesor *atmel AVR* dan Menggunakan *software* serta bahasa sendiri [9].

2.6.1 Perangkat Keras *Arduino Nano*

Perangkat keras dalam *Arduino* memiliki beberapa jenis, yang mempunyai kelebihan dan kekurangan dalam setiap papannya. Penggunaan jenis *Arduino* disesuaikan dengan kebutuhan, hal ini yang akan mempengaruhi dari jenis prosesor yang digunakan. Jika semakin kompleks perancangan dan program yang dibuat, maka harus sesuai pula jenis *controller* yang digunakan. Yang membedakan antara *Arduino* yang satu dengan yang lainnya adalah penambahan fungsi dalam setiap papan sirkuitnya dan jenis mikrokontroler yang digunakan. Dalam Proyek Akhir, jenis *Arduino* yang digunakan adalah *Arduino Nano*, dengan ditunjukkan Gambar 2.12. *Hardware Arduino Nano* memiliki spesifikasi sebagai berikut



Gambar 2.12 *Arduino Nano*[9]

1. 14 pin IO Digital (pin 0–13)
Sejumlah pin digital dengan nomor 0–13 yang dapat dijadikan *input* atau *output* yang diatur dengan cara membuat program IDE.
2. 6 pin *Input* Analog (pin 0–5)
Sejumlah pin analog bernomor 0–5 yang dapat digunakan untuk membaca nilai *input* yang memiliki nilai analog dan mengubahnya ke dalam angka antara 0 dan 1023.
3. 6 pin *Output* Analog (pin 3, 5, 6, 9, 10 dan 11)
Sejumlah pin yang sebenarnya merupakan pin digital tetapi sejumlah pin tersebut dapat diprogram kembali menjadi pin *output* analog dengan cara membuat programnya pada IDE.

Papan *Arduino Nano* dapat mengambil daya dari *USB port* pada komputer dengan menggunakan *USB charger* atau dapat pula mengambil daya dengan menggunakan suatu *AC adapter* dengan tegangan 9 Volt. Jika tidak terdapat *power supply* yang melalui *AC adapter*, maka papan *Arduino* akan mengambil daya dari *USB port*. Tetapi apabila diberikan daya melalui *AC adapter* secara bersamaan dengan *USB port* maka papan *Arduino* akan mengambil daya melalui *AC adapter* secara otomatis.

2.6.2 Perangkat Lunak *Arduino IDE*



Gambar 2.13 Tampilan IDE *Arduino*[9]

Software Arduino yang digunakan adalah *driver* dan IDE, walaupun masih ada beberapa *software* lain yang sangat berguna selama pengembangan *Arduino*. IDE atau *Integrated Development Environment* suatu program khusus untuk suatu komputer agar dapat membuat suatu rancangan atau sketsa program untuk papan *Arduino* [9]. IDE *Arduino*

merupakan *software* yang sangat canggih ditulis dengan menggunakan *Java*. IDE *Arduino* terdiri dari:

1. Editor program, sebuah *window* yang memungkinkan pengguna menulis dan mengubah program dalam bahasa *processing*.
2. *Compiler*, sebuah modul yang mengubah kode program (bahasa *processing*) menjadi kode biner. Bagaimanapun sebuah mikrokontroler tidak akan bisa memahami bahasa *processing*. Yang bisa dipahami oleh mikrokontroler adalah kode biner. Itulah sebabnya *compiler* diperlukan dalam hal ini.
3. *Uploader*, sebuah modul yang memuat kode biner dari komputer ke dalam *memory* didalam papan *Arduino*. Sebuah kode program *Arduino* umumnya disebut dengan istilah *sketch*. Kata “*sketch*” digunakan secara bergantian dengan “kode program” dimana keduanya memiliki arti yang sama. Berikut adalah Gambar 2.13 tampilan IDE *Arduino* dengan sebuah *sketch*.

2.7 Modul Relay 5V

Relay adalah komponen elektronika yang berupa saklar atau *Switch* elektrik yang dioperasikan menggunakan listrik. *Relay* juga biasa disebut sebagai komponen electromechanical atau elektromekanikal yang terdiri dari dua bagian utama yaitu coil atau elektromagnet dan kontak saklar. Komponen *relay* menggunakan prinsip elektromagnetik sebagai penggerak kontak saklar, sehingga dengan menggunakan arus listrik yang kecil atau *low power*, dapat menghantarkan arus listrik yang memiliki tegangan lebih tinggi. *relay* memiliki fungsi sebagai saklar elektrik. Namun jika diaplikasikan ke dalam rangkaian elektronika, *relay* memiliki beberapa fungsi yang cukup unik. Berikut adalah beberapa fungsi komponen *relay* saat diaplikasikan ke dalam sebuah rangkaian elektronika.

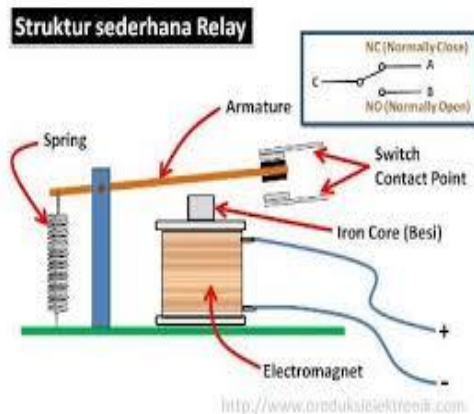
1. Mengendalikan sirkuit tegangan tinggi dengan menggunakan bantuan signal tegangan rendah.
2. Menjalankan fungsi logika alias logic function.
3. Memberikan fungsi penundaan waktu alias *time delay function*.
4. Melindungi motor atau komponen lainnya dari kelebihan tegangan atau korsleting.



Gambar 2.14 Module Relay

(Sumber : <http://www.electronicapro.com.pk/product/one-1-channel-relay-module/>)

Cara kerja atau prinsip kerja *relay* bahwa dalam sebuah *relay* terdapat 4 buah bagian penting yakni *electromagnet* (Coil), *armature*, *switch contact point* (saklar), dan *spring*.



Gambar 2.15 Rangkaian dalam *Relay*[10]

Dari Gambar 2.15 rangkaian dalam *relay* dapat diketahui bahwa sebuah besi (*Iron Core*) yang dililit oleh kumparan coil, berfungsi untuk mengendalikan besi tersebut. Apabila kumparan coil dialiri arus listrik, maka akan muncul gaya elektromagnetik yang dapat menarik *armature* sehingga dapat berpindah dari posisi sebelumnya tertutup (NC) menjadi posisi baru yakni terbuka (NO). Dalam posisi (NO) saklar dapat menghantarkan arus listrik. Pada saat tidak dialiri arus listrik, *armature*

akan kembali ke posisi awal (NC). Sedangkan coil yang digunakan oleh *relay* untuk menarik contact poin ke posisi *close* hanya membutuhkan arus listrik yang relatif cukup kecil [10].

2.8 Analog to Digital Converter

Analog to Digital Converter (ADC) adalah pengubah *input* analog menjadi kode – kode digital. ADC banyak digunakan sebagai pengatur proses industri, komunikasi digital dan rangkaian pengukuran/ pengujian. Umumnya ADC digunakan sebagai perantara antara sensor yang kebanyakan analog dengan sistem komputer seperti sensor suhu, cahaya, tekanan/ berat, aliran dan sebagainya kemudian diukur dengan menggunakan sistem digital.

Resolusi ADC menentukan “ketelitian nilai hasil konversi ADC”. Sebagai contoh: ADC 8 bit akan memiliki *output* 8 bit data digital, ini berarti sinyal *input* dapat dinyatakan dalam 255 ($2^n - 1$) nilai diskrit. ADC 12 bit memiliki 12 bit *output* data digital, ini berarti sinyal *input* dapat dinyatakan dalam 4096 nilai diskrit. Dari contoh diatas ADC 12 bit akan memberikan ketelitian nilai hasil konversi yang jauh lebih baik daripada ADC 8 bit.

Prinsip kerja ADC adalah mengkonversi sinyal analog ke dalam bentuk besaran yang merupakan rasio perbandingan sinyal *input* dan tegangan referensi. Sebagai contoh, bila tegangan referensi 5 Volt, tegangan *input* 3 Volt, rasio *input* terhadap referensi adalah 60%. Jadi, jika menggunakan ADC 8 bit dengan skala maksimum 255, akan didapatkan sinyal digital sebesar $60\% \times 255 = 153$ (bentuk decimal) atau 10011001 (bentuk biner) [11].

2.9 Sensor Kadar Air dalam Oli SHT20

Sensor Kadar Air adalah sensor yang mengubah kadar air menjadi sinyal listrik analog. Ada berbagai jenis sensor air seperti *transducer* sensor air, sensor air *output* digital, tegangan/ arus, dan banyak lainnya. Namun, dalam semua jenis sensor air, konversi tekanan menjadi sinyal listrik dicapai oleh deformasi fisik diafragma molekul air kemudian menghasilkan perubahan resistansi listrik sebanding dengan molekul air.



Gambar 2.16 Sensor Kadar Air SHT20 [12]

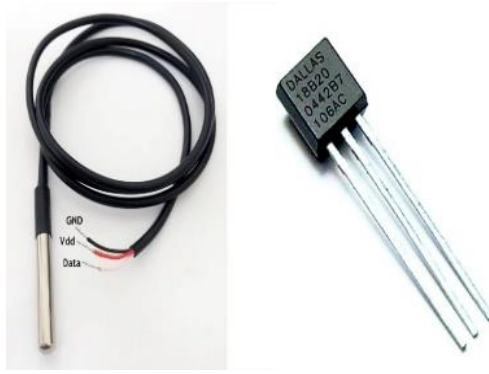
SHT 20 merupakan sensor suhu dan kelembaban relatif. Sensor ini digunakan sebagai pengindra suhu dan kelembaban dalam aplikasi pengendali suhu dan kelembaban maupun aplikasi pemantau suhu dan kelembaban relatif [12]. Adapun spesifikasi SHT20 ditunjukkan pada Tabel 2.2 sebagai berikut.

Tabel 2.2 Spesifikasi Sensor SHT20

No.	Spesifikasi	Karakteristik
1.	Power Supply	5v
2.	RH jarak operasi	0-100% RH
3.	T Jarak operasi	-40 -125oC (-40 – 257oF)
4.	rH <i>respon time</i>	8 sec (tau63%)
5.	<i>Output</i>	RH = 12-bit T = 14-bit
6.	Komunikasi	<i>Digital 2-wire interface, I2C protocol</i>

2.10 Sensor Temperatur DS18B20

Sensor suhu atau *temperature sensor* adalah suatu komponen yang dapat mengubah besaran panas menjadi besaran listrik sehingga dapat mendeteksi gejala perubahan suhu pada obyek tertentu. Sensor suhu melakukan pengukuran terhadap jumlah energi panas/dingin yang dihasilkan oleh suatu obyek sehingga memungkinkan untuk mengetahui atau mendeteksi gejala perubahan-perubahan suhu tersebut dalam bentuk *output digital*. Sensor Suhu juga merupakan dari keluarga *transducer*.



Gambar 2.17 Sensor Temperatur DS18B2 [12]

Sensor DS18B20 merupakan sensor digital yang memiliki 12-bit ADC internal. Sangat presisi, sebab jika tegangan referensi sebesar 5 Volt, maka akibat perubahan suhu, ia dapat merasakan perubahan terkecil sebesar $5/(2^{12}-1) = 0.0012$ Vol Pada rentang suhu -10 sampai +85 derajat celcius, sensor ini memiliki akurasi ± 0.5 derajat. Sensor ini bekerja menggunakan protokol komunikasi *one-wire* [13]. Sensor temperatur seperti Gambar 2.17 dan spesifikasi alat disajikan pada Tabel 2.3

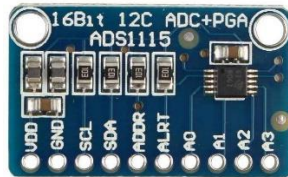
Tabel 2.3 Spesifikasi Sensor DS18B20

No.	Spesifikasi	Karakteristik
1.	Power Supply	3.0V to 5.5V
2.	Suhu operasi	-55°C to +125°C (-67F to +257F)
3.	T Jarak operasi	10°C to +85°C ($\pm 0.5^\circ\text{C}$)
4.	<i>Output</i>	8-bit
5.	Komunikasi	<i>Digital 1-wire interface</i>

2.11 Module ADS1115

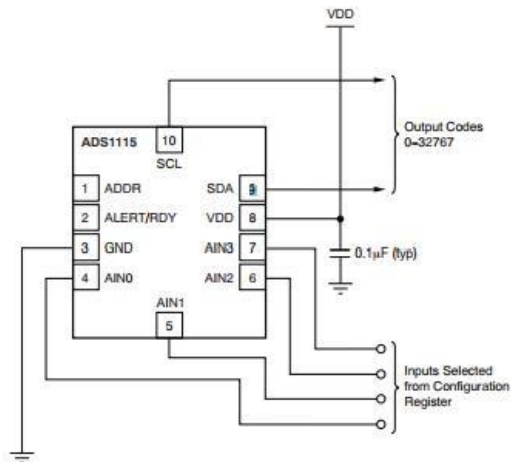
Modul ADS1115 merupakan jenis ADC yang memiliki resolusi 16 bit, ini berarti ADC ini memiliki tingkat ketelitian nilai hasil konversi yang tinggi dibandingkan dengan ADC yang memiliki sedikit resolusi. Dalam ADC ini juga terdapat 4 *channel* yang dapat mengkonversi nilai untuk 4 sensor sekaligus dengan *differensial* bipolar maupun tunggal. Fitur ADC ini yaitu sebuah referensi onboard dan oscillator. Data yang diterima akan ditransfer atau dikirim melalui komunikasi serial I2C.

Serial tersebut terdiri dari SDA dan SCL. Berikut Gambar 2.18 yang menunjukkan modul ADS1115.



Gambar 2.18 Modul ADS1115 [11]

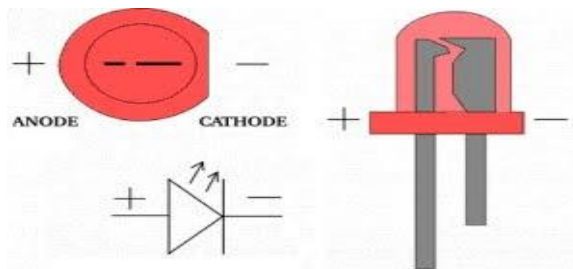
Meskipun ADS1115 memiliki 2 *input* yang berbeda, alat ini tetap mampu mengukur 4 sinyal *single-ended*. Pada Gambar 2.21 menunjukkan sebuah skema koneksi *single-ended*. ADS1115 dikonfigurasi untuk pengukuran *single-ended* dengan mengkonfigurasi MUX (*Multiplexer*) untuk mengukur setiap saluran dengan memperhatikan *ground*. Data tersebut kemudian dibaca dari satu masukan berdasarkan seleksi pada register konfigurasi. Sinyal *single-ended* dapat menyuplai dari batas 0 Volt. Tegangan negatif tidak bisa diaplikasikan pada sirkuit ini karena ADS1115 hanya bisa menerima tegangan positif [11].



Gambar 2.19 Input dan Output (Datasheet ADS1115) [11]

2.12 LED (*Light Emitting Diode*)

Light Emitting Diode (LED) pada Gambar 2.20 adalah komponen elektronika yang dapat memancarkan cahaya monokromatik ketika diberikan tegangan maju. LED merupakan keluarga Dioda yang terbuat dari bahan semikonduktor. Warna-warna cahaya yang dipancarkan oleh LED tergantung pada jenis bahan semikonduktor yang dipergunakannya. LED juga dapat memancarkan sinar inframerah yang tidak tampak oleh mata seperti yang sering kita jumpai pada *remote control* TV ataupun *remote control* perangkat elektronik lainnya. Bentuk LED mirip dengan sebuah bohlam (bola lampu) yang kecil dan dapat dipasangkan dengan mudah ke dalam berbagai perangkat elektronika. Berbeda dengan lampu pijar, LED tidak memerlukan pembakaran *filamen* sehingga tidak menimbulkan panas dalam menghasilkan cahaya [14].



Gambar 2.20 Struktur LED [14]

2.13 Switch

Switch merupakan jenis saklar yang dilengkapi dengan katup yang berfungsi menggantikan tombol. Prinsip kerja *switch* sama seperti saklar Push ON yaitu hanya akan menghubungkan pada saat katupnya ditekan pada batas penekanan tertentu yang telah ditentukan dan akan memutuskan saat katup tidak ditekan. *Switch* termasuk dalam kategori sensor mekanis yaitu sensor yang akan memberikan perubahan elektrik saat terjadi perubahan mekanik pada sensor tersebut. Penerapan dari *switch* adalah sebagai sensor posisi suatu benda (objek) yang bergerak.

Switch umumnya digunakan untuk memutuskan dan menghubungkan rangkaian menggunakan objek atau benda lain; Menghidupkan daya yang besar, dengan sarana yang kecil; Sebagai sensor posisi atau kondisi suatu objek.

Prinsip kerja limit *switch* diaktifkan dengan penekanan pada tombolnya pada batas/daerah yang telah ditentukan sebelumnya sehingga terjadi pemutusan atau penghubungan rangkaian dari rangkaian tersebut. *Switch* memiliki 2 kontak yaitu NO (*Normally Open*) dan kontak NC (*Normally Close*) dimana salah satu kontak akan aktif jika tombolnya tertekan.



Gambar 2.21 *Switch*[14]

---Halaman ini sengaja dikosongkan---

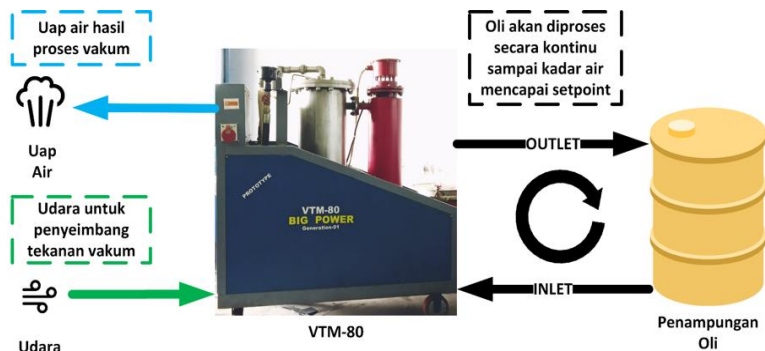
BAB III

PERANCANGAN SISTEM

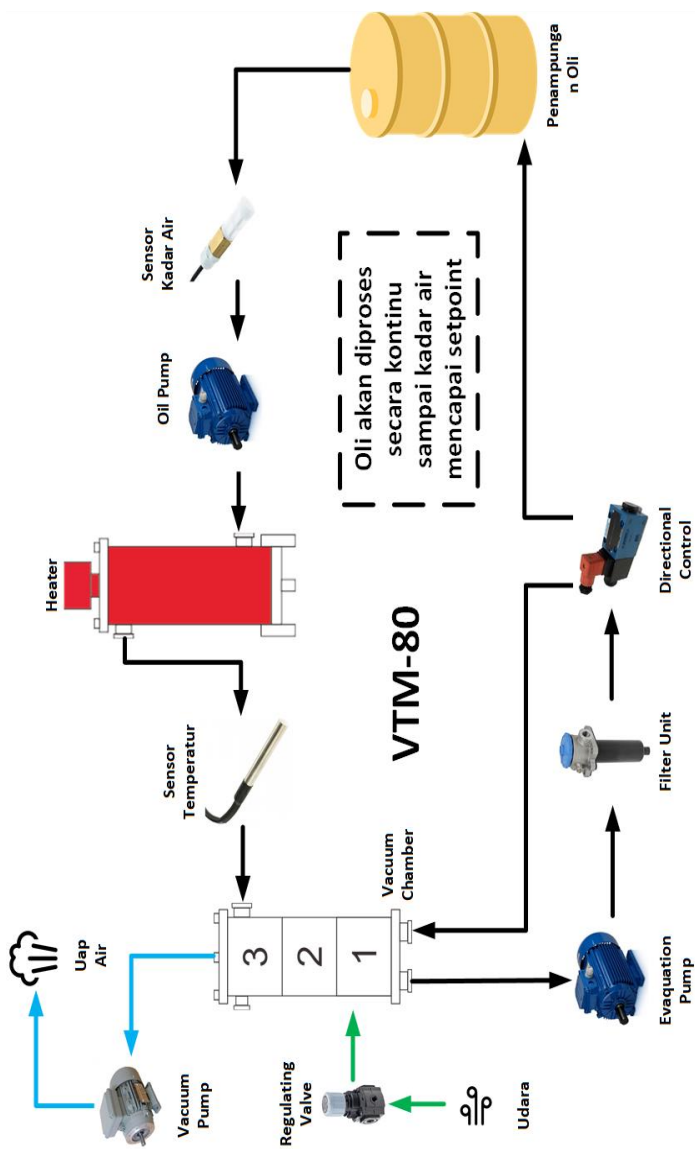
Pada bab ini akan membahas mengenai tahapan yang dilakukan terhadap perancangan dan pembuatan *software* dan *hardware* untuk Proyek Akhir penerapan *human machine interface* pada mesin Filter oli VTM-80 menggunakan *Raspberry Pi*.

3.1 Cara Kerja Alat

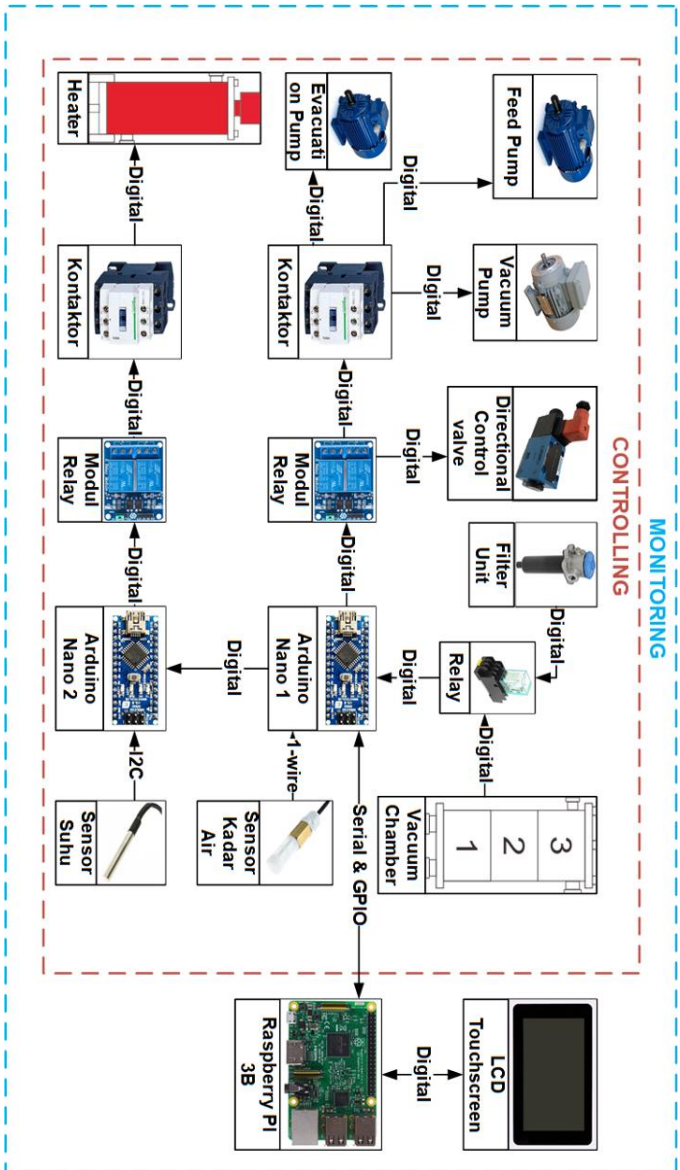
Sebelum memasuki memasuki perancangan sistem akan dijelaskancara kerja mesin filter oli VTM-80. Mesin filter oli VTM-80 sendiri merupakan sebuah alat yang digunakan untuk memfilter oli yang memiliki kandungan air yang berlebih. Oli dengan kandungan air yang berlebih tersebut akan diproses secara kontinu, Dimana oli akan dipanaskan dan dialirkan pada tabung vakum. Saat berada di tabung vakum setelah dipanaskan air yang berada pada oli akan mencapai titik didih dan menguap (tanpa merusak oli). Uap air tersebut akan dikeluarkan, sedangkan oli akan dikeluarkan menuju ke penampungan oli untuk diproses kembali. Kondisi vakum diusahakan stabil, sehingga diperlukan udara secukupnya. VTM-80 akan memproses oli pada penampung oli secara kontinu hingga mencapai *setpoint* kadar air yang ditentukan. VTM-80 disajikan pada Gambar 3.1



Gambar 3.1 VTM-80



Gambar 3.2 Komponen-Komponen VTM-80



Gambar 3.3 Blok Fungsional Sistem (Landscape)

VTM-80 didalamnya terdapat beberapa komponen dengan cara kerja sebagai berikut. Sebelumnya disini kami menambahkan sensor kadar air dan sensor temperatur . pertama oli pada penampungan oli diarahkan menuju *inlet* dipompa menggunakan *oil pump*, *oil pump* dikendalikan menggunakan kontaktor. Oli yang telah masuk akan dicek kadar airnya menggunakan sensor kadar air. Kadar air akan dibaca kemudian oli dipanaskan dengan *heater* dikendalikan dengan kontaktor berkerja secara ON/OFF maka akan diatur suhu pada *heater* pada batas tertentu. Suhu tersebut akan *dimonitoring* dengan sensor temperatur. Setelah dipanaskan oli dialirkan dan divakum pada *vacuum chamber*. *vacuum chamber* memiliki ruang untuk menampung oli dengan kapasitas 3 level berupa *level switch*. Uap air yang dihasilkan dari proses vakum akan dikeluarkan oleh *vacuum pump*. *Vacuum pump* dikendalikan menggunakan kontaktor. Oli yang telah divakum pada *vacuum chamber* akan dikeluarkan dengan *Evaquation Pump* menuju *outlet*. *Evaquation pump* dikendalikan menggunakan kontaktor. *directional valve* berguna untuk mengembalikan aliran oli pada *vacuum chamber* jika pada ketinggian volume terendah. *filter* berguna memfilter oli dari partikel dan debu. Oli yang telah keluar pada *outlet* akan masuk kembali pada *inlet* untuk diukur kadar air, saat kadar air dalam oli belum normal maka proses akan dilanjutkan. Proses tersebut akan berjalan secara kontinu hingga nilai kadar air sesuai *setpoint*. Cara kerja komponen VTM-80 disajikan pada Gambar 3.2

Fungsi dari masing-masing komponen juga dapat dilihat dibawah ini:

1. Sensor kadar air SHT20
Berfungsi untuk membaca persentase kandungan air pada partikel oli yang akan diproses oleh VTM-80
2. *Oil pump*
Berfungsi untuk memompa oli dari penampungan oli (*inlet*) yang akan diproses oleh VTM-80
3. *Heater*
Berfungsi untuk memanaskan oli dan menjaga suhu oli sesuai dengan nilai yang ditentukan.
4. Sensor temperatur
Berupa sensor DS18B20 yang berguna mendeteksi perubahan suhu pada *heater*.
5. Tabung vakum (*vacuum chamber*)

Berfungsi untuk memvakum oli setelah dipanaskan pada *Heater* sehingga titik didih air bisa tercapai dan air pada oli akan menguap. Tabung vakum memiliki 3 *level* kapasitas didalamnya, dimana masing-masing *level* mempengaruhi aksi kontrol pada kerja VTM-80

6. *Vacuum pump*

Berfungsi untuk menyedot uap air pada tabung vakum dan menciptakan kondisi vakum pada tabung vakum.

7. *Evaquation pump*

Berfungsi untuk memompa oli dari tabung vakum

8. *Directional valve*

Berguna untuk mengontrol arah aliran oli dari tabung vakum keluar ke penampungan oli (*outlet*) atau kembali lagi ke tabung vakum sesuai dengan kondisi *level Switch* yang bekerja.

9. *Filter unit*

Berguna untuk menyaring oli dari partikel dan debu. Terdapat *switch* yang akan bekerja jika kapasitas penyaringan *filter unit* tersebut sudah mencapai nilai maksimum.

3.2 Blok Fungsional Sistem

Blok fungsional sistem merupakan blok diagram yang memberikan gambaran bagaimana nantinya sistem yang akan diterapkan bekerja pada Proyek Akhir ini. Blok diagram ini berisi arah komunikasi dari sistem dan jenis *protocol* atau *interface* yang digunakan untuk berkomunikasi. Secara keseluruhan blok fungsional sistem dapat dilihat pada Gambar 3.3.

Berikut penjelasan mengenai blok diagram pada Gambar 3.3, blok diagram ini dibagi menjadi 2 bagian yaitu bagian *contolling* yang menggunakan 2 buah *Arduino Nano* yang masing masing fungsinya membaca nilai temperatur atau jumlah kandungan air dari sensor. Nilai pembacaan dari sensor tersebut digunakan untuk mengontrol modul *relay* yang masing-masing *relay* nya akan mengontrol masing-masing komponen yang terdapat pada VTM-80. Selain itu kondisi dari *Switch* yang terpasang pada tabung vakum dan *filter unit* juga akan mempengaruhi pengontrolan modul *relay* tersebut.

Data data hasil pembacaan sensor-sensor, dan kondisi komponen yang dikontrol (ON/OFF) dikirim ke *Raspberry Pi* dan data data tersebut disajikan pada HMI dan ditampilkan pada LCD *touchscreen*. Pada perancangan Proyek Akhir ini akan meliputi bagian *monitoring*

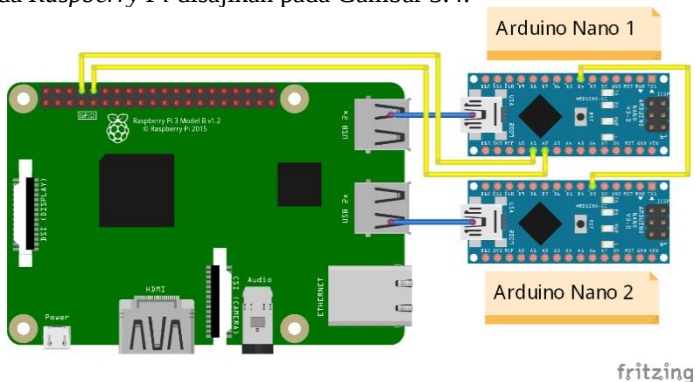
yang *memonitoring* sistem kontrol dan disajikan pada LCD melalui *Raspberry Pi*.

3.3 Perancangan *Hardware*

Pada perancangan *hardware* dilakukan dengan merancang rangkaian elektronika yang berupa *wiring* komponen-komponen, skematik alat dan konfigurasi *hardware*

3.3.1 Rangkaian *Arduino Nano* pada *Raspberry Pi*

Arduino Nano sebagai kontroling akan disambungkan dengan *Raspberry Pi* yang akan menerima dan mengirim data. Komunikasi *Arduino Nano* dan *Raspberry* dikirim melalui serial. Sekaligus *Arduino Nano* mengambil supply dari *GPIO Raspberry Pi*. Serta menggunakan *GPIO* untuk tombol mulai dan stop. Adapun Rangkaian *Arduino Nano* pada *Raspberry Pi* disajikan pada Gambar 3.4.



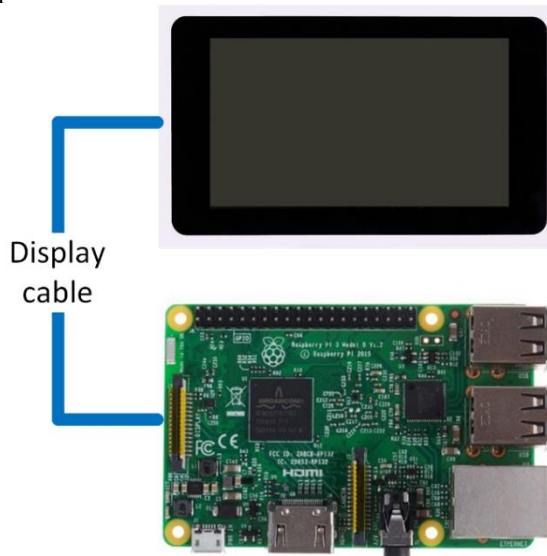
Gambar 3.4 Rangkaian *Arduino Nano* pada *Raspberry Pi*

Tabel 3.1 Rangkaian *Arduino Nano* pada *Raspberry Pi*

No.	<i>Arduino Nano 1</i>	<i>Arduino Nano 2</i>	<i>Raspberry Pi</i>
1.	USB	USB	USB
2.	A0	A1	GPIO 11
3.	A1	-	GPIO 12
4.	Vin	Vin	5V
5.	GND	GND	GND

3.3.2 Rangkaian *Raspberry Pi* pada LCD

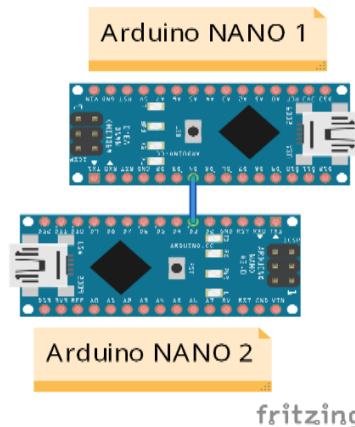
Pada konfigurasi *Arduino Nano* pada HMI sebagai *monitoring*, HMI menggunakan LCD Touchscreen yang akan menerima dan mengirim data dari *Raspberry Pi*. Komunikasi *Raspberry Pi* dan LCD dikirim melalui serial. Sekaligus *Arduino Nano* mengambil supply dari GPIO *Raspberry Pi*. Adapun rangkaian *Raspberry Pi* pada LCD disajikan pada Gambar 3.5.



Gambar 3.5 Konfigurasi *Arduino Nano* pada HMI

3.3.3 Rangkaian Komunikasi 2q *Arduino Nano*

Pada sistem kerja VTM-80 terdapat dua variabel yang akan dikontrol yaitu kadar air dalam oli dan suhu pada *heater* sehingga keduanya harus bisa dijalankan secara bersamaan dengan komponen yang dikontrol secara masing-masing. Komunikasi ini akan memberikan sinyal digital dari *Arduino 1* pada *Arduino 2*. Pin *Arduino Nano 1* D4 akan disambungkan dengan pin *Arduino Nano 2* D3. Berikut adalah gambar rangkaian komunikasi dua *Arduino Nano* yang disajikan pada Gambar 3.6.



Gambar 3.6 Komunikasi Dua *Arduino Nano*

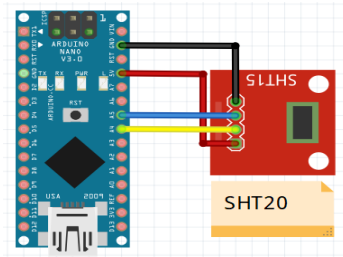
3.3.4 Rangkaian Sensor Kadar Air dalam Oli SHT 20

Pada rangkaian sensor kadar air dalam oli SHT 20 berfungsi mendeteksi perubahan jumlah kadungan air didalam oli pada sistem VTM-80. yang nantinya SHT 20 sendiri di hubungkan ke sumber 5 V pada *Arduino Nano* dan selanjutnya di hubungkan ke GND pada *Arduino Nano* dan pin SDA ke pin analog A4 guna untuk pengiriman sinyal menghasilkan perubahan resistansi listrik sebanding dengan molekul air dan pin SCL ke pin analog A5. Berikut adalah rangkaian sensor kadar air dalam Oli SHT 20 yang disajikan tabel 3.2

Tabel 3.2 Pin Modul Sensor Kadar Air dalam Oli SHT 20
Dengan *Arduino Nano*

No.	Pin SHT 20	Pin <i>Arduino Nano</i>
1.	VCC	5V
2.	GND	GND
3.	SDA	A4
4.	SCL	A5

Berikut adalah *wiring* rangkaian sensor SHT 20 ke *Arduino Nano* yang disajikan pada Gambar 3.7



Gambar 3.7 Rangkaian Sensor Kadar Air dalam Oli SHT 20

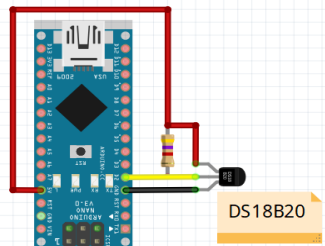
3.3.5 Rangkaian Sensor Temperatur *Heater* DS18B20

Pada sensor temperatur *heater* DS18B20 berfungsi yang berguna mendeteksi perubahan suhu pada *heater* yang nantinya DS18B20 sendiri di hubungkan ke resistor dengan resistansi $4,7K\Omega$ dan selanjutnya di hubungkan ke GND pada mikrokontroler *Arduino Nano*, dan pin satunya di hubungkan ke sumber 5V pada mikrokontroler *Arduino Nano* dan di antara LDR dan resistor resistansi $4,7K\Omega$ dihubungkan ke pin digital guna untuk pengiriman sinyal pengukuran terhadap jumlah energi panas/dingin yang dihasilkan oleh suatu obyek sehingga memungkinkan untuk mengetahui atau mendeteksi gejala perubahan-perubahan suhu tersebut dalam bentuk *output* digital. Berikut adalah rangkaian temperatur *heater* DS18B20 yang disajikan pada Tabel 3.3.

Tabel 3.3 Pin Rangkaian Temperatur *Heater* DS18B20 Dengan *Arduino Nano*

No.	Pin SHT 20	Pin Arduino Nano
1.	VCC	5V
2.	GND	GND
3.	Digital	D2

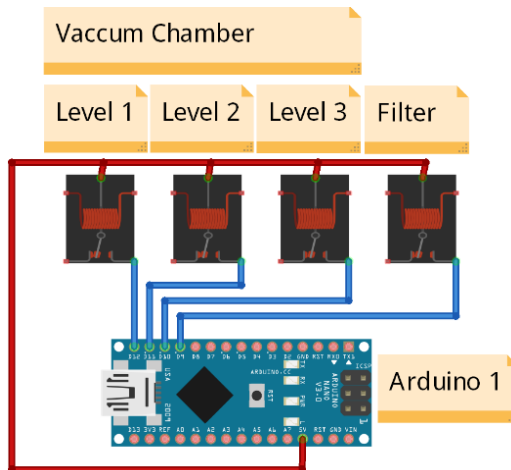
Pada perancangan alat Proyek Akhir ini, sensor temperatur . DS18B20 mendeteksi temperatur oli pada. Hasil pembacaan akan diarahkan pada *Arduino* untuk mengkontrol *module relay* yang mengarah pada kontaktor sehingga rangkaian temperatur *heater* DS18B20 seperti pada Gambar 3.8 :



Gambar 3.8 Rangkaian Sensor Temperatur Heater DS18B20

3.3.6 Rangkaian *Relay* pada *Arduino Nano 1*

Konfigurasi *Arduino Nano 1* pada sistem ini sebagai kontroling akan memberikan *input* pada *relay* 24V yang mengarah ke *Arduino 1*. Komponen tersebut adalah *vacuum chamber* (*level 1, level 2, level 3*) dan *clogging* pada filter yang bekerja secara ON/OFF. Rangkaian *relay* pada *Arduino Nano 1* disajikan pada Gambar 3.9



Gambar 3.9 Rangkaian *Relay* pada *Arduino Nano 1*

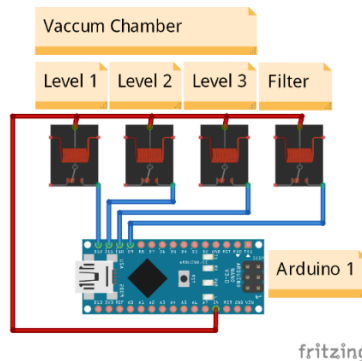
Adapaun kaki pin *relay* yang terintegrasi dengan *Arduino Nano* dan komponen VTM-80 sesuai yang disajikan pada Tabel 3.4

Tabel 3.4 Pin Rangkaian *Relay* pada *Arduino Nano 1*

No.	Pin Arduino Nano	Pin Relay	Komponen VTM-80
1.	D9	N.0.	Level 1 – Vacuum Chumber
2.	D10	N.0.	Level 2 – Vacuum Chumber
3.	D11	N.0.	Level 3 – Vacuum Chumber
4.	D12	N.0.	Filter
5.	5V	VCC Tiap Relay	-
6.	GND	GND Tiap Relay	-

3.3.7 Rangkaian Module *Relay* pada *Arduino Nano 1*

Konfigurasi *Arduino Nano 1* pada sistem ini sebagai kontroling akan memberikan *output* pada *module relay* yang mengarah pada *oil pump*, *evaquation pump*, *vacuum pump*, *directional valve*. Rangkaian *module relay* pada *Arduino Nano 1* disajikan pada Gambar 3.10



Gambar 3.10 Rangkaian *Module Relay* pada *Arduino Nano 1*

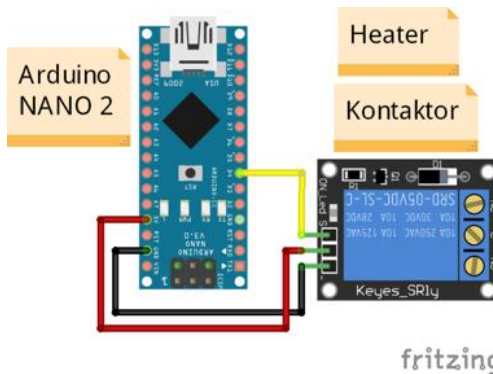
Adapun kaki pin *relay* yang terintegrasi dengan *Arduino Nano* dan komponen VTM-80 sesuai yang disajikan pada Tabel 3.5

Tabel 3.5 Pin Rangkaian *Module Relay* pada *Arduino Nano 1*

No.	Pin Arduino Nano	Pin Relay	Komponen VTM-80
1.	D5	input	<i>Directional Valve</i>
2.	D6	input	Kontaktor – <i>Vacuum Pump</i>
3.	D7	input	Kontaktor – <i>Evaquation Pump</i>
4.	D8	input	Kontaktor – <i>Oil Pump</i>
5.	5V	VCC Tiap Relay	-
6.	GND	GND Tiap Relay	-

3.3.8 Rangkaian *Module Relay* pada *Arduino Nano 2*

Konfigurasi *Arduino Nano 2* pada sistem ini sebagai kontroling akan memberikan *output* pada *module relay* yang mengarah pada *heater*. Rangkaian *module relay* pada *Arduino Nano 2* disajikan pada Gambar 3.11. Adapaun kaki pin *relay* yang terintegrasi dengan *Arduino Nano* dan komponen VTM-80 sesuai yang disajikan pada Tabel 3.6.



Gambar 3.11 Rangkaian *Module Relay* pada *Arduino Nano 2*

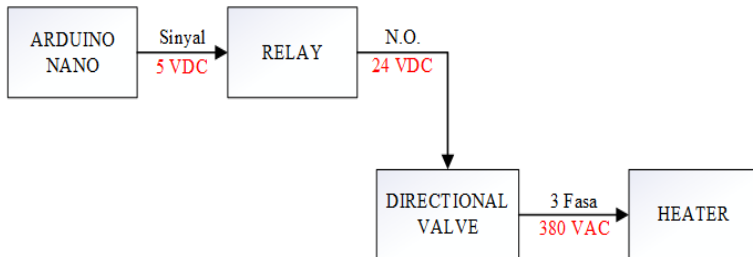
Tabel 3.6 Pin Rangkaian Module *Relay* pada *Arduino Nano 2*

No.	Pin Arduino Nano	Pin Relay	Komponen VTM-80
1.	D4	input	Driver – Heater
2.	5V	VCC	-
3.	GND	GND	-

3.3.9 Konfigurasi Hardware Pengontrol Heater

Pada konfigurasi *hardware* pengontrol *heater* yang berfungsi untuk memanaskan oli pada suhu tertentu sistem kerjanya ON/OFF dikendalikan dengan kontaktor. *heater* dikontrol oleh *Arduino Nano* yang dimana akan mengirimkan sinyal pada *input relay*. Pin N.O. *relay* akan disambungkan pada coil kontaktor untuk mengkontak *heater*.

Perancangan ini menggunakan *relay* agar bisa menerima sinyal digital dari *Arduino Nano* setelah itu sinyal tersebut tersalurkan ke kontaktor untuk mensaklar *heater*. Adapaun konfigurasi *hardware* pengontrol *heater* disajikan pada Gambar 3.12.

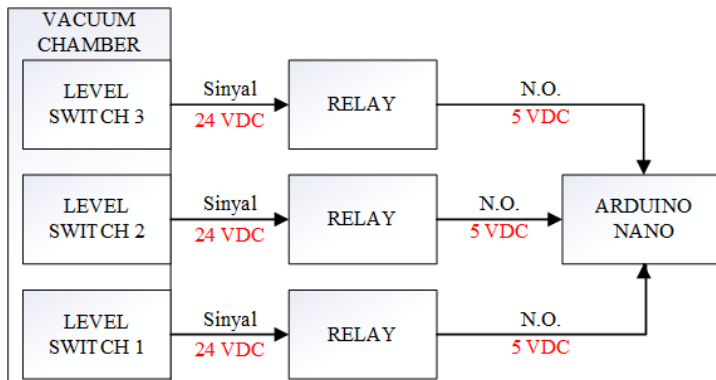


Gambar 3.12 Konfigurasi Hardware Pengontrol Heater

3.3.10 Konfigurasi Hardware pada Vacuum Chamber

Pada konfigurasi *hardware* pada *vacuum chamber* Berfungsi untuk memvakum oli setelah dipanaskan pada *heater* sehingga titik didih air bisa tercapai dan air pada oli akan menguap. *vacuum chamber* memiliki kapasitas tiga level didalamnya berupa *level switch*. *Level switch* akan mengirimkan sinyal pada *relay*. *Relay* pin N.O. akan disambungkan pada pin digital *Arduino Nano*.

Perancangan ini menggunakan *relay* agar bisa mengirim sinyal digital dari *Arduino Nano*. Adapaun konfigurasi *hardware* pada *vacuum chamber* disajikan pada Gambar 3.13.

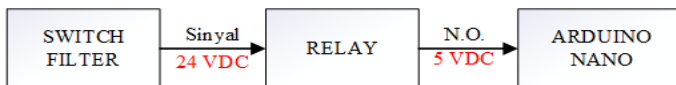


Gambar 3.13 Konfigurasi Hardware pada Vacuum Chamber

3.3.11 Konfigurasi Hardware pada Filter Unit

pada konfigurasi hardware pada filter unit berguna untuk memfilte oli dari partikel dan debu. filter unit memiliki level switch didalamnya dan akan aktif saat terjadi kemacetan sehingga filter unit perlu diganti. level switch akan mengirimkan sinyal pada relay. relay pin N.O. akan disambungkan pada pin digital Arduino Nano.

perancangan ini menggunakan relay agar bisa mengirim sinyal digital dari Arduino Nano. adapapun konfigurasi hardware pada vacuum chamber disajikan pada Gambar 3.14.

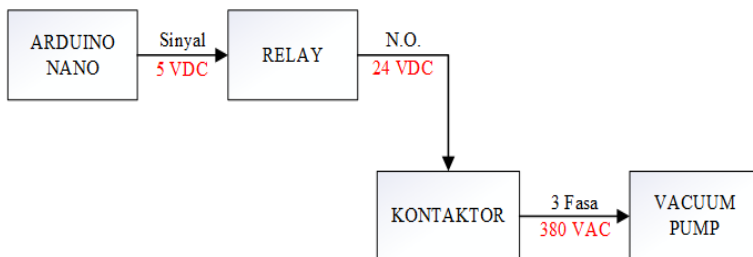


Gambar 3.14 Konfigurasi Hardware pada Filter Unit

3.3.12 Konfigurasi Hardware pada Vacuum Pump

Pada konfigurasi hardware pada vacuum pump yang berfungsi untuk mengeluarkan uap air pada vacuum chamber sistem kerjanya ON/OFF dikendalikan dengan kontaktor. Vacuum pump dikontrol oleh Arduino Nano yang dimana akan mengirimkan sinyal pada input relay. Pin N.O. relay akan disambungkan pada coil kontaktor untuk mengkontak vacuum pump.

Perancangan ini menggunakan *relay* agar bisa menerima sinyal digital dari *Arduino Nano* setelah itu sinyal tersebut diteruskan ke kontaktor untuk mensaklar *vacuum pump*. Adapun konfigurasi *hardware* pada *vacuum pump* disajikan pada Gambar 3.15.

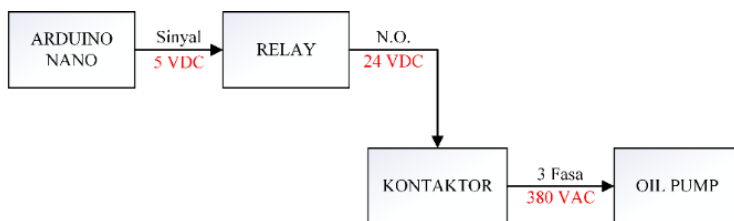


Gambar 3.15 Konfigurasi *Hardware* pada *Vacuum Pump*

3.3.13 Konfigurasi *Hardware* pada *Oil Pump*

Pada konfigurasi *hardware* pada *oil pump* yang berfungsi untuk memompa oli pada penampungan oli untuk diarahkan ke *Inlet*. sistem kerjanya ON/OFF dikendalikan dengan kontaktor. *oil pump* dikontrol oleh *Arduino Nano* yang dimana akan mengirimkan sinyal pada *input relay*. Pin N.O. *relay* akan disambungkan pada *coil* kontaktor untuk mengkontak *oil pump*.

Perancangan ini menggunakan *relay* agar bisa menerima sinyal digital dari *Arduino Nano* setelah itu sinyal tersebut diteruskan ke kontaktor untuk mensaklar *oil pump*. Adapun konfigurasi *hardware* pada *oil pump* disajikan pada Gambar 3.16.

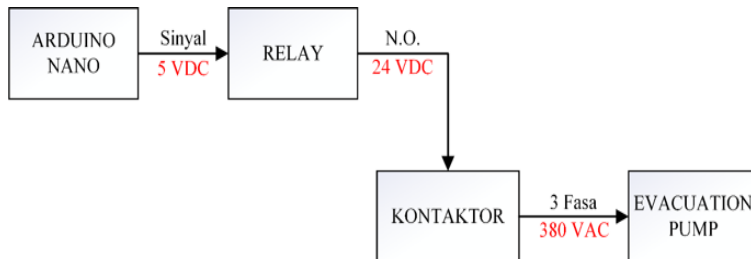


Gambar 3.16 Konfigurasi *Hardware* pada *Oil Pump*

3.3.14 Konfigurasi Hardware pada *Evaquation Pump*

Konfigurasi *hardware* pada *evaquation pump* yang berfungsi untuk memompa oli pada tangki oli untuk diarahkan ke *inlet*.sistem kerjanya ON/OFF dikendalikan dengan kontaktor. *Evaquation pump* dikontrol oleh *Arduino Nano* yang dimana akan mengirimkan sinyal pada *input relay*. Pin N.O. *relay* akan disambungkan pada *coil* kontaktor untuk mengkontak *evaquation pump*.

Perancangan ini menggunakan *relay* agar bisa menerima sinyal digital dari *Arduino Nano* setelah itu sinyal tersebut diteruskan ke kontaktor untuk mensaklar *evaquation pump*. Adapaun konfigurasi *hardware* pada *evaquation pump* disajikan pada Gambar 3.17.

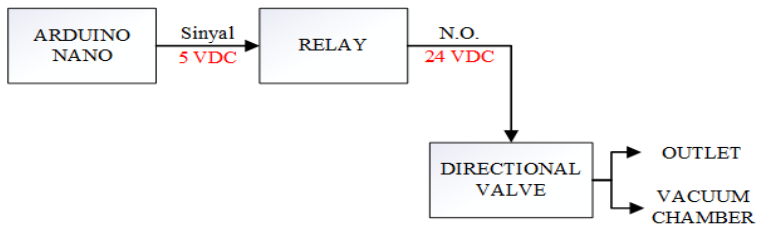


Gambar 3.17 Konfigurasi Hardware pada *Evaquation Pump*

3.3.15 Konfigurasi Hardware pada *Directional Valve*

Pada konfigurasi *hardware* pada *directional valve* yang berguna untuk membelokkan aliran oli kembali pada *vacuum chamber* saat *level* didalamnya pada posisi terendah Dan tetap mengarahkan aliran oli ke *outlet*. saat tidak di saklar sistem kerjanya 2 katup *directional valve* dikontrol oleh *Arduino Nano* yang dimana akan mengirimkan sinyal pada *input relay*. Pin N.O. *relay* akan mengkontak *directional valve*.

Perancangan ini menggunakan *relay* agar bisa menerima sinyal digital dari *Arduino Nano* setelah itu sinyal tersebut diteruskan untuk mensaklar *directional valve*. Adapaun konfigurasi *hardware* pada *directional valve* disajikan pada Gambar 3.18.



Gambar 3.18 Konfigurasi *Hardware* pada *Directional Valve*

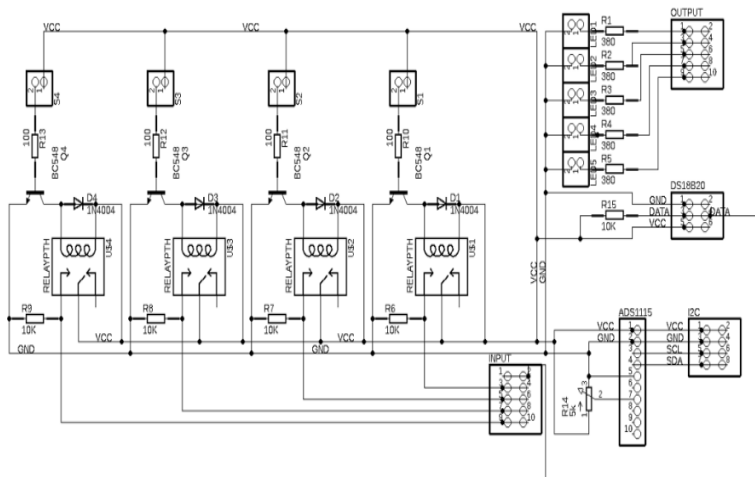
3.3.16 Perancangan *Miniature Plant*

Perancangan *miniature plant* ini dibuat sedemikian rupa agar bisa digunakan untuk menggantikan perancangan perangkat keras (*hardware*) sehingga bisa memberikan gambaran pada *plant* sebenarnya. Perancangan *miniature plant* dibagi menjadi beberapa sub bab yang akan dijelaskan per sub bab nya, antara lain:

1. Rangkaian *Miniature plant*
2. Perancangan Pengganti Rangkaian Sensor Kadar Air
3. Perancangan Rangkaian Sensor Temperatur *Heater DS18B20*
4. Perancangan Rangkaian *Hardware* menggunakan LED
5. Perancangan Rangkaian *Hardware* menggunakan *Push Button*

3.3.17 Rangkaian *Miniature plant*

Rangkaian *miniature plant* berfungsi untuk menggantikan perancangan perangkat keras (*hardware*) yang ada. Rangkaian *miniature plant* ini memiliki masukan daya 5 Volt. Terdapat potensiometer, sensor DS18B20, *switch* dan LED dimana masing-masing fungsi akan dijelaskan pada Tabel 3.7. Pada Gambar 3.19 akan dijelaskan rangkain skematik *miniature plant*.



Gambar 3.19 Rangkaian Skematik *Miniature Plan*

Tabel 3.7 Komponen *Miniature Plant*

No.	Komponen <i>Miniature Plan</i>	Komponen VTM-80	Cara Kerja
1.	Potensiometer dan Modul ADS1115	Sensor Kadar Air SHT20	Sensor Kadar Air SHT20 komunikasi menggunakan I2C sehingga peranannya akan digantikan dengan perubahan sinyal tegangan yang kemudian akan dikonversikan menjadi 12C menggunakan Potensiometer dan Modul ADS1115.
2.	Sensor DS18B20	Sensor Temperatur <i>Heater</i> DS18B20	Sensor Temperatur <i>Heater</i> tetap menggunakan Sensor DS18B20 yang selanjutnya akan dipanaskan.
3.	LED <i>Heater</i>	<i>Heater</i>	Sistem <i>Heater</i> kerjanya ON/OFF dikendalikan

Tabel 3.7 Komponen Miniature Plant

No.	Komponen Miniature Plan	Komponen VTM-80	Cara Kerja
			dengan kontaktor indikator menggunakan LED.
4.	LED Oil Pump	Oil Pump	Sistem <i>oil pump</i> kerjanya ON/OFF dikendalikan dengan kontaktor indikator menggunakan LED.
5.	LED Evaquation Pump	Evaquation Pump	Sistem <i>evaquation pump</i> kerjanya ON/OFF dikendalikan dengan kontaktor indikator menggunakan LED.
6.	LED Vacuum Pump	Vacuum Pump	Sistem <i>vacuum pump</i> kerjanya ON/OFF dikendalikan dengan kontaktor indikator menggunakan LED.
7.	LED Directional Valve	Directional Valve	Sistem <i>directional valve</i> kerjanya ON/OFF dikendalikan dengan kontaktor indikator menggunakan LED.
8.	Switch Level 1	Level Switch 1 – Vacuum Chumber	<i>Vacuum chamber</i> memiliki kapasitas tiga <i>level</i> didalamnya berupa <i>level Switch</i> . <i>Level switch</i> 1 akan mengirimkan sinyal pada <i>relay</i> . <i>Level switch</i> tersebut akan digantikan dengan <i>switch</i> .
9.	Switch Level 2	Level Switch 2 – Vacuum Chumber	<i>Level Switch</i> 2 akan mengirimkan sinyal pada <i>relay</i> . <i>Level switch</i> tersebut akan digantikan dengan <i>Switch</i> .
10.	Switch Level 3	Level Switch 3 –	<i>Level Switch</i> 3 akan mengirimkan sinyal pada

Tabel 3.7 Komponen Miniature Plant

No.	Komponen Miniature Plan	Komponen VTM-80	Cara Kerja
		<i>Vacuum Chumber</i>	<i>relay. Level switch</i> tersebut akan digantikan dengan <i>switch</i> .
11	<i>Switch Filter</i>	<i>Filter</i>	<i>Filter</i> akan mengirimkan sinyal pada <i>relay</i> . <i>Filter</i> tersebut akan digantikan dengan <i>switch</i>
12.	5V	24V	Supply pada alat 24V akan digantikan dengan Supply 5V.
13.	GND	GND	-

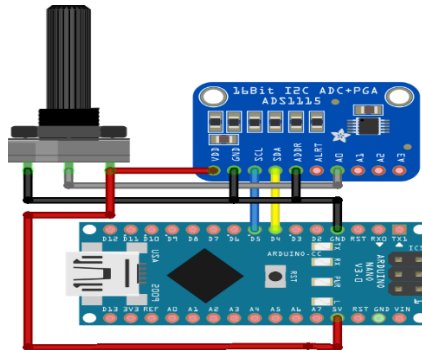
3.3.18 Perancangan Rangkaian Pengganti Sensor Kadar Air dengan Modul ADS1115

Perancangan ini menggantikan fungsi sensor kadar air dalam SHT20 yang memiliki komunikasi SDA-SCL sehingga digantikan dengan variabel tegangan yang dikonversikan menjadi SDA-SCL sehingga bisa menggantikan peran sensor SHT20 menggunakan potensiometer yang diberikan tegangan kemudian diarahkan pada pin analog module ADS1115. Module ADS1115 variabel tegangan 5 Volt yang dikonversikan menjadi sda-scl yang mengarah pada *Arduino Nano* 1. Variable tegangan akan diakumulasikan menjadi variabel kadar air dalam oli. Berikut adalah pengganti rangkaian sensor kadar air yang disajikan Tabel 3.8

Tabel 3.8 Pin Pengganti Rangkaian Sensor Kadar Air Dengan *Arduino Nano*

No.	Potensiometer	Pin Modul ADS1115	Pin Arduino Nano
1.	5V	VCC	5V
2.	GND	GND	GND
3.	Pin 3	A0	-
4.	-	SDA	A4
5	-	SCL	A5
6	-	ADDR	GND

Berikut adalah gambar 3.20 pengganti rangkaian sensor kadar air ke *Arduino Nano* yang diperlukan. Untuk lebih jelasnya terdapat pada Gambar 3.19 yang merupakan skematik dari rangkaian *Arduino Nano*.



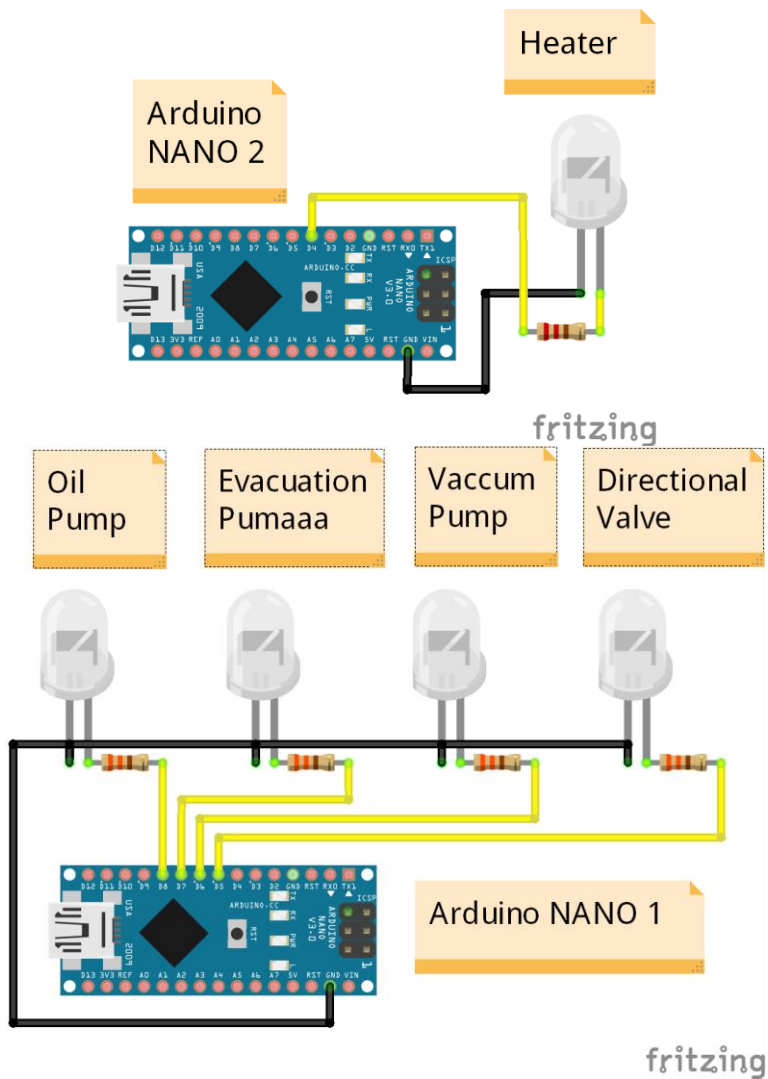
Gambar 3.20 Pengganti Rangkaian Sensor Kadar Air ke *Arduino Nano*

3.3.19 Perancangan Rangkaian Sensor Temperatur *Heater DS18B20*

Rangkaian sensor temperatur *heater DS18B20* berfungsi yang berguna mendeteksi perubahan suhu pada *heater* sama dengan perancangan *hardware*. Untuk pengiriman sinyal pengukuran terhadap jumlah energi panas/dingin yang dihasilkan oleh suatu obyek sehingga memungkinkan untuk mengetahui atau mendeteksi gejala perubahan-perubahan suhu tersebut dalam bentuk *output* digital.

3.3.20 Perancangan Rangkaian *Hardware* Menggunakan LED

Pada perancangan *hardware* ini digunakan untuk menggantikan konfigurasi *hardware* pada beberapa komponen yang sudah dijelaskan sebelumnya. LED tersebut akan dihubungkan dengan *Arduino Nano*. Berikut adalah tabel pin rangkaian *hardware* menggunakan LED yang disajikan pada Tabel 3.9. Pada perancangan ini menggunakan *relay* agar bisa mengirim sinyal digital dari *Arduino Nano*. Rangkaian *hardware* menggunakan LED seperti pada Gambar 3.21 :



Gambar 3.21 Rangkaian *Hardware* Menggunakan LED

Tabel 3.3 Pin Rangkaian *Hardware* menggunakan LED pada Arduino Nano

No.	Pin Arduino Nano	Pin Relay	Miniature Plan
1.	D4	input	LED Heater
2.	D5	input	LED Oil Pump
3.	D6	input	LED Evaporation Pump
4.	D7	input	LED Vacuum Pump
5.	D8	input	LED Directional Valve
6.	5V	VCC Tiap Relay	-
7.	GND	GND Tiap Relay	-

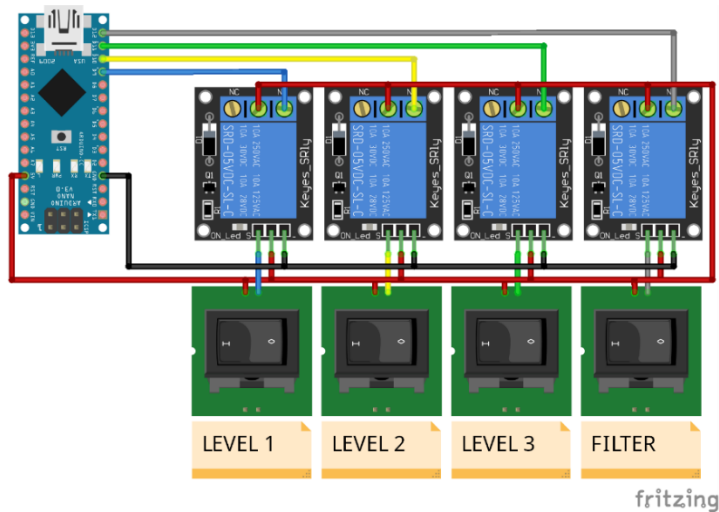
3.3.21 Perancangan Rangkaian *Hardware* Menggunakan Switch

Pada perancangan *hardware* ini digunakan untuk menggantikan konfigurasi *hardware* pada beberapa komponen yang sudah dijelaskan sebelumnya. Switch tersebut akan dihubungkan dengan *Arduino Nano*. Berikut adalah tabel pin rangkaian *hardware* menggunakan switch yang disajikan pada Tabel 3.10.

Tabel 3.9 Pin Rangkaian *Hardware* Menggunakan Switch pada Arduino Nano

No.	Pin Arduino Nano	Pin Relay	Miniature Plan
1.	D9	N.0.	Switch Level 1
2.	D10	N.0.	Switch Level 2
3.	D11	N.0.	Switch Level 3
4.	D12	N.0.	Filter
5.	5V	VCC Tiap Relay	-
6.	GND	GND Tiap Relay	-

Pada perancangan alat Proyek Akhir ini, *hardware* menggunakan LED menggantikan fungsi komponen VTM-80. Hasil indikator dari *Arduino* untuk mengontrol *module relay* yang mengarah pada *switch* sehingga rangkaian *hardware* menggunakan *switch* seperti pada Gambar 3.22



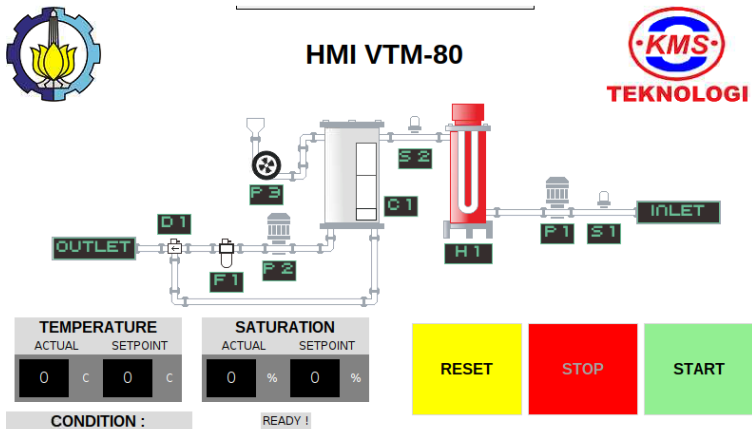
Gambar 3.22 Rangkaian *Hardware* Menggunakan *Switch*

3.4 Perancangan *Software*

Software yang digunakan untuk alat Proyek Akhir ini adalah pemrograman *python* untuk membuat program, dan aplikasi *corel draw* untuk membuat gambar yang akan ditampilkan pada HMI.

3.4.1 Perancangan Tampilan HMI

Tampilan pada HMI VTM-80 akan terdiri dari 2 bagian yaitu tampilan utama dan tampilan menu *input*. Tampilan utama terdiri dari gambar *plant* VTM-80, nilai temperatur dan kadar air yang dibaca oleh sensor, nilai temperatur dan kadar air yang telah di masukkan, status mesin, tombol *start* dan tombol *stop* seperti Gambar 3.23



Gambar 3.23 Perancangan Tampilan Menu Utama HMI VTM-80

V2 169.254.91.184 (raspberrypi) - VNC Viewer

SETPOINT

TEMPERATURE		0	C
SATURATION		50	%
1	2	3	
4	5	6	
7	8	9	
Clear	0	Enter	

Gambar 3.24 Perancangan Tampilan Menu *Input*.

Pada gambar plant VTM-80 pada HMI terdapat kode untuk setiap bagian komponen plant VTM -80 untuk memudahkan identifikasi komponen, kode kode tersebut adalah:

- P1 (*Oil pump*) : Pompa listrik yang digunakan untuk mengalirkan oli dari tangki penampungan ke *vacuum chamber* melalui *heater*.
- P2 (*Evaquation pump*) : Pompa listrik yang digunakan untuk mengalirkan oli dari *vacuum chamber* kembali ke tangki penampungan atau kembali lagi ke *vacuum chamber* untuk melewati *filter unit* secara berulang-ulang.
- P3 (*Vacuum pump*) : Pompa vakum yang digunakan untuk mengeluarkan uap air pada oli panas didalam *vacuum chamber*, serta menurunkan tekanan udara didalam *vacuum chamber* agar air dapat dengan mudah menguap
- S1 (*Sensor kadar air*) : Sensor yang digunakan untuk membaca persentase dari kadar air yang terdapat pada oli.
- S2 (*Sensor temperatur*) : Sensor yang digunakan untuk membaca temperatur oli yang telah dipanaskan oleh *heater*.
- H1 (*Heater*) : Pemanas listrik yang digunakan untuk memanaskan oli agar air didalam oli dapat menguap.
- C1 : *Vacuum chamber* yang digunakan sebagai tempat pemisah antara air dan oli. Didalam *vacuum chamber* terdapat 3 *level* ketinggian oli yang dibaca oleh *level switch*.
- F1 : *Filter unit* yang berfungsi untuk menyaring partikel partikel yang tercampur pada oli.
- D1 : *Directional valve* digunakan untuk mengatur arah aliran oli yang dipompa oleh *evaquation pump* untuk kembali ke *vacuum chamber* atau menuju ke tangki penampungan.

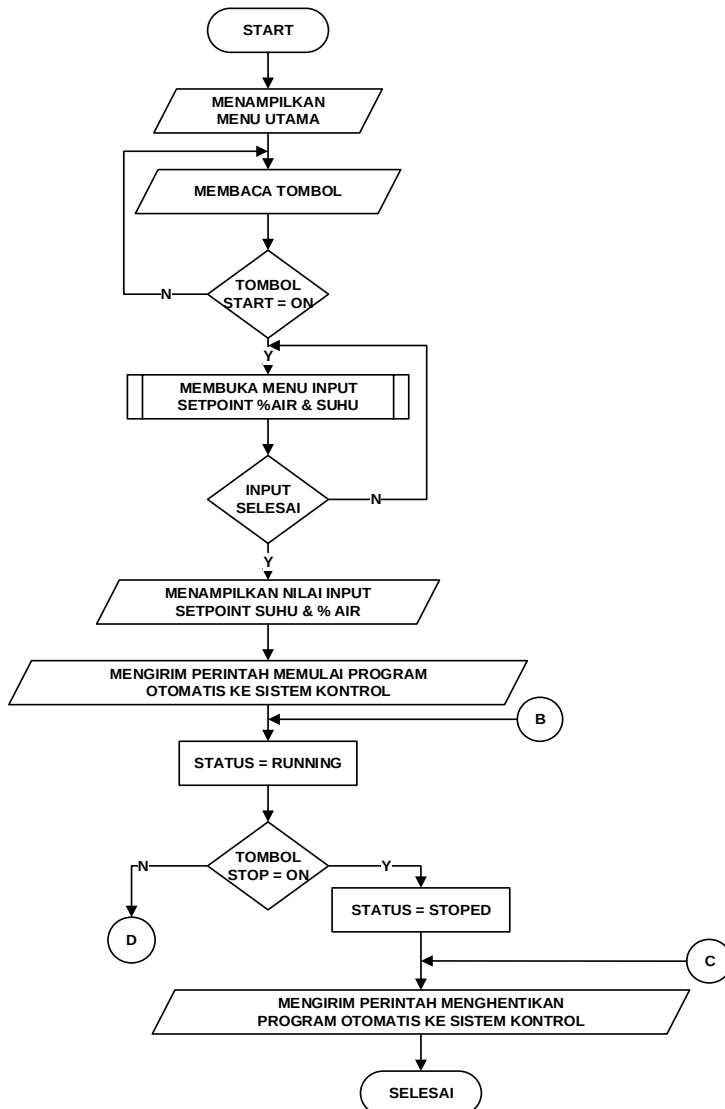
HMI ini juga menampilkan kondisi kerja plant VTM-80 yang meliputi:

- READY ! : Kondisi ini menyatakan bahwa plant VTM-80 dapat langsung dijalankan.
- RUNNING ! : Kondisi ini merupakan kondisi normal mesin saat sedang bekerja. Kondisi ini menyatakan bahawa nilai kadar air aktual yang terbaca belum mencapai nilai *setpoint* yang ditentukan

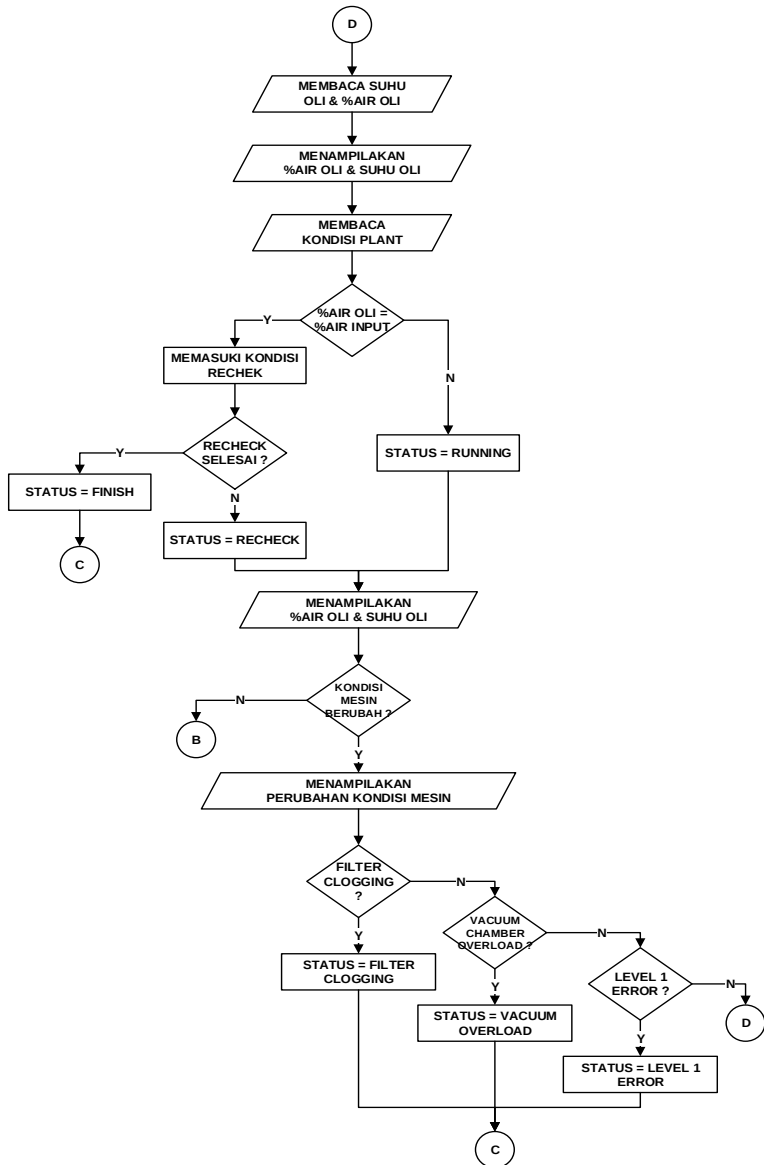
- **CHECKING !** : Kondisi ini berlangsung pada saat nilai kadar air sudah lebih kecil atau sama dengan nilai *setpoint* untuk memastikan bahwa dalam 5 menit, nilai kadar air aktual memang sudah mencapai *setpoint* yang ditentukan. Apabila dalam 5 menit, nilai aktual kadar air menjadi lebih besar dari nilai *setpoint*. Mesin kembali ke kondisi 'RUNNING !'
- **EMPTYED VACUUM CHAMBER** : Kondisi ini merupakan kondisi saat oli didalam *vacuum chamber* dikosongkan karena setelah melewati pengecekan nilai kadar air selama 5 menit pada kondisi *checking* terpenuhi atau terjadi *error* saat mesin bekerja sehingga oli harus dikeluarkan dari *vacuum chamber*.
- **FINISH !** : Kondisi ini merupakan pemberitahuan kepada operator bahwa proses penyaringan sudah selesai dan kadar air mencapai *setpoint* yang di tentukan sehingga plant sudah berhenti bekerja
- **FILTER CLOGGING !** : Kondisi ini merupakan peringatan kepada operator bahwa *filter* tersumbat sehingga plant berhenti bekerja dan *filter* harus diganti sebelum memulai proses kerja kembali.
- **VACUUM CHAMBER OVERLOAD !** : Kondisi ini merupakan peringatan kepada operator bahwa oli pada *vacuum chamber* mencapai *level 3* dan harus ditangani sebelum memulai proses kerja kembali.
- **LEVEL 1 ERROR !** :Kondisi ini merupakan peringatan kepada operator bahwa dalam 5 menit setelah memulai proses kerja *switch level 1* pada *vacuum chamber* belum juga menyala sehingga memerlukan pengecekan lebih lanjut oleh operator.
- **STOPED !** : Kondisi ini merupakan kondisi saat tombol stop pada HMI ditekan sehingga proses kerja berhenti sebelum tombol start kembali ditekan.

Tampilan menu *input* akan muncul saat tombol start di tekan. Tampilan ini berisi tombol 0-9, enter, delete yang digunakan untuk memasukkan nilai *input* dan tombol temperatur dan kadar air digunakan untuk pemilihan *input* sesuai pada Gambar 3.24 . Setelah tombol enter di tekan. Data akan di simpan didalam program dan mesin akan mulai bekerja.

3.4.2 Algoritma HMI VTM-80



Gambar 3.25 Algoritma Pemrograman HMI VTM-80



Gambar 3.25 Lanjutan

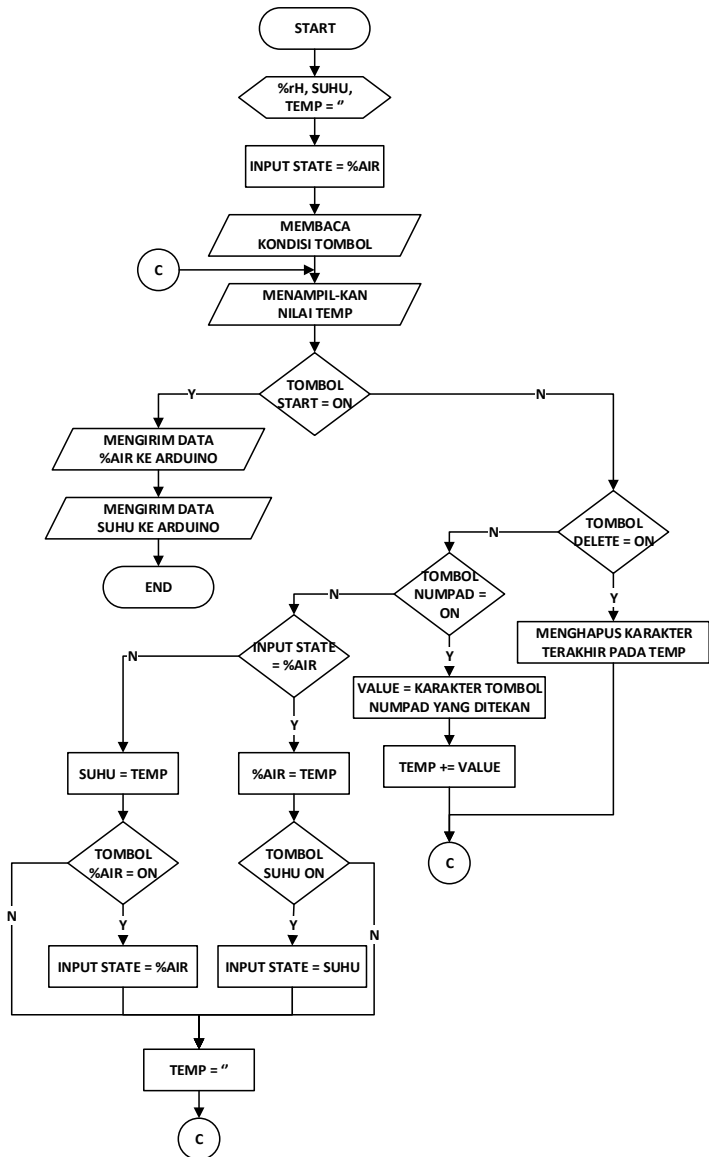
Pada Gambar 3.25 merupakan gambar *algoritma* HMI untuk VTM-80. Pengoprasian HMI ini dimulai saat program dijalankan. Pada saat itu HMI menampilkan gambar representasi dari *plant* VTM-80, tombol *START* dan tombol *STOP*. Saat tombol *START* ditekan, program akan membuka tampilan untuk memasukkan *setpoint* nilai kadar air dan temperatur . Setelah selesai memasukkan nilai *setpoint*, *Raspberry Pi* akan membaca nilai temperatur dan kadar air pada oli yang dibaca oleh sensor melalui *Arduino* serta perubahan kondisi kondisi mesin dari *plant* VTM-80 yang akan ditampilkan pada HMI.

Setelah tombol *START* ditekan, kolom status pada HMI akan menampilkan tulisan “*running*” yang menandakan bahwa VTM-80 sedang bekerja. Apa bila tombol *STOP* di tekan pada kondisi ini, proses akan langsung selesai dan *plant* VTM-80 akan berhenti bekerja sampai tombol *START* ditekan kembali.

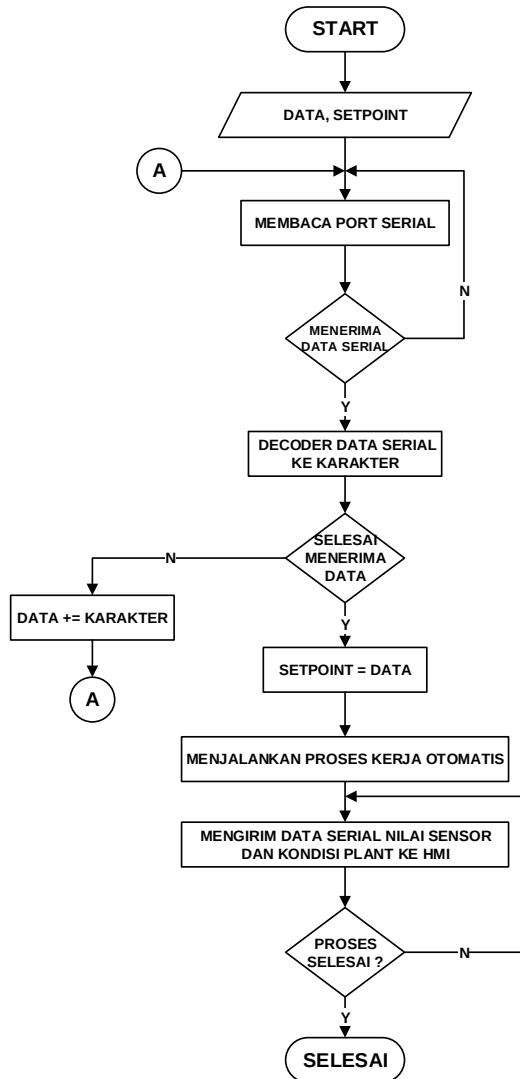
Untuk mengetahui kapan *filter* harus diganti dan oli pada *vacuum chamber* telah melewati batas normal (*overload*). VTM-80 menggunakan *switch* untuk mengetahui kondisi kondisi tersebut. Apabila kondisi salah satu *switch* tersebut berubah akan terbaca oleh *Raspberry* dan ditampilkan pada HMI. Pada kondisi ini kolom status akan menampilkan pesan ‘*Vacuum Overload*’ apabila oli pada *vacuum chamber* telah melewati batas normal dan ‘*Filter Clogging*’ apabila *filter* harus diganti, sedangkan apabila dalam waktu lima menit level *switch* 1 tidak terbaca, kolom status akan menampilkan ‘*Level 1 Error*’. Pada saat kondisi ini terjadi, *plant* VTM 80 harus berhenti bekerja karena harus diperiksa secara manual.

3.4.3 Algoritma Menu *Input* Pada HM

Menu *Input* ini digunakan untuk memasukkan *setpoint* nilai temperatur oli maksimum dan kadar air pada oli yang diinginkan. Menu ini terdiri dari tombol *numpad* yang merupakan tombol angka 0-9. Tombol suhu dan tombol *saturation* dan tombol *temperature* yang digunakan untuk memilih nilai antara nilai temperatur dan kadar air yang akan diubah. Tombol *clear* yang digunakan untuk menghapus angka jika terjadi kesalahan saat memasukkan nilai. Dan tombol *enter/start* yang digunakan untuk memasukkan nilai *setpoint* dari temperatur dan kadar air yang diinginkan kedalam HMI dan mengirim kedua data tersebut ke *Arduino Nano*. Di menu *input* data kadar air dan suhu akan disimpan dalam bentuk *string* untuk memudahkan saat masukkan nilai.



Gambar 3.26 Algoritma Menu Input



Gambar 3.27 *Algoritma Komunikasi Serial Arudino Dengan Raspberry Pi*

Gambar 3.26 merupakan algoritma pemrograman menu *input*. Saat menu *input* dijalankan, program akan membaca kondisi tombol yang ditekan. *input* awal berupa kadar air, *input* akan berubah ke temperatur jika tombol *temperature* di tekan dan kembali lagi ke kadar air jika tombol *saturation* di tekan. Saat tombol *numpad* ditekan, nilai akan disimpan dan ditambahkan dengan nilai sebelumnya. Karena tipe data berupa string, penjumlahan berupa penambahan karakter setelah karakter sebelumnya. Sehingga saat di tekan tombol 5 lalu tombol 0, data yang disimpan berupa 50. Menu *input* akan tertutup saat tombol *start* di tekan. Dan data temperatur dan kadar air akan dikirimkan ke *Arduino Nano*

3.4.4 Algoritam Komunikasi Serial *Arduino* dan *Raspberry Pi*

Gambar 3.27 merupakan algoritma komunikasi serial antara *Raspberry Pi* dan *Arduino Nano*. Komunikasi serial ini digunakan digunakan untuk pengiriman data dari *Raspberry* ke *Arduino* atau sebaliknya. Pada pembacaan data serial, data yang dikirimkan akan berupa kode ASCII yang merupakan hasil pengkodean dari data serial yang dikirimkan.

Setelah menerima data setpoint dari *Raspberry* sistem kontrol menjalankan *plant* VTM-80. Pada saat ini, *Raspberry* secara kontinu membaca nilai kadar air dan temperatur dari *plant* serta kondisi ON/OFF komponen *plant* VTM-80 dan menampilkan hasilnya pada tampilan HMI.

---Halaman ini sengaja dikosongkan---

BAB IV

PENGUJIAN ANALISIS

Pengujian sistem yang dilakukan merupakan pengujian terhadap perangkat keras dan perangkat lunak dari sistem secara keseluruhan yang telah selesai dibuat untuk mengetahui komponen-komponen sistem apakah berjalan dengan baik dan sesuai yang diharapkan.

4.1 Pengujian Software

Pengujian ini merupakan pengujian program HMI yang akan digunakan pada VTM-80. Pengujian ini berupa pengujian Tampilan utama, tampilan menu *input*, dan tampilan saat mesin berjalan sampai mesin mati, apakah sudah sesuai dengan pada perancangan atau tidak.

4.1.1 Pengujian Tampilan Utama HMI

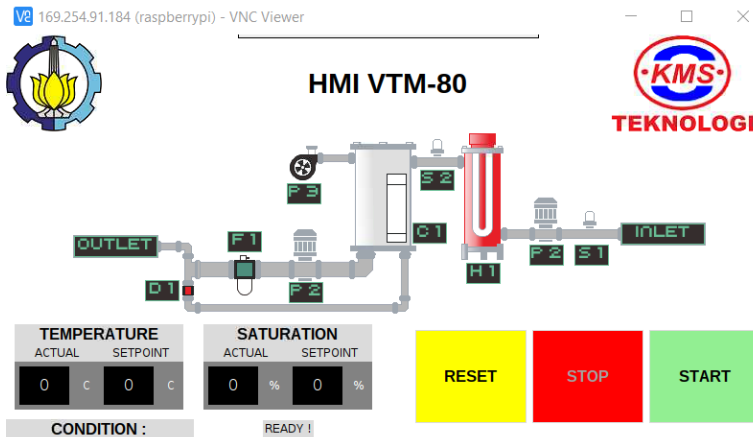
Gambar 4.1 merupakan tampilan utama saat program baru dijalankan. Tampilan utama ini meliputi gambaran dari komponen – komponen mesin VTM-80, nilai *setpoint*, nilai aktual temperatur dan persentase kadar air yang terbaca pada saat mesin VTM-80, *text box condition* yang menampilkan proses kerja mesin dan peringatan - peringatan *error*, serta tombol *start* dan *stop*, untuk memulai dan menghentikan proses kerja mesin.

Saat mesin belum bekerja, nilai *setpoint* dan aktual yang ditampilkan bernilai 0, karena nilai temperatur dan kadar air aktual belum dibaca oleh mesin VTM-80 dan nilai *setpoint* belum diberikan, dan *text box condition* akan menampilkan tulisan '*Ready!*' saat mesin siap untuk dijalankan

Pada bagian gambar dari komponen komponen mesin terdapat kode nama untuk setiap komponen yang digunakan, kode kode nama itu adalah:

- P1 (*Oil pump*) : Pompa listrik yang digunakan untuk mengalirkan oli dari tangki penampungan ke *vacuum chamber* melalui *heater*.
- P2 (*Evaquation pump*) : Pompa listrik yang digunakan untuk mengalirkan oli dari *vacuum chamber* kembali ke tangki penampungan atau kembali lagi ke *vacuum chamber* untuk melewati *filter unit* secara berulang-ulang.

- P3 (*vacuum pump*) : Pompa vakum yang digunakan untuk mengeluarkan uap air pada oli panas di dalam *vacuum chamber*, serta menurunkan tekanan udara didalam *vacuum chamber* agar air dapat dengan mudah menguap
- S1 (sensor kadar air) : Sensor yang digunakan untuk membaca persentase dari kadar air yang terdapat pada oli.
- S2 (sensor temperatur) : Sensor yang digunakan untuk membaca temperatur oli yang telah dipanaskan oleh *heater*.
- H1 (*heater*) : Pemanas listrik yang digunakan untuk memanaskan oli agar air didalam oli dapat menguap.
- C1 : *Vacuum chamber* yang digunakan sebagai tempat pemisah antara air dan oli. Didalam *vacuum chamber* terdapat 3 level ketinggian oli yang dibaca oleh *level switch*.
- F1 : *Filter unit* yang berfungsi untuk menyaring partikel partikel yang tercampur pada oli.
- D1 : *Directional valve* digunakan untuk mengatur arah aliran oli yang dipompa oleh *evaquation pump* untuk kembali ke *vacuum chamber* atau menuju ke tangki penampungan.



Gambar 4.1 Tampilan Menu Utama HMI

4.1.2 Pengujian Tampilan Menu *Input* HMI

Setelah tombol START pada menu utama ditekan, HMI akan menampilkan menu *input* seperti pada Gambar 4.2 dibawah ini. Menu ini digunakan untuk mengatur *setpoint* untuk temperatur oli maksimal yang akan dipanaskan oleh *heater* dan persentase nilai kadar air minimal untuk mengakhiri proses kerja mesin. Pada menu ini terdapat tombol angka, *enter*, *delete*, *temperature*, dan *saturation* serta tampilan nilai *setpoint* yang telah akan di masukkan. Pada pengujian tampilan menu ini semua tombol bekeja sesuai dengan fungsinya dan saat tombol Enter ditekan, tampilan menu *input* akan tertutup dan HMI akan mengirimkan nilai *setpoint* temperatur dan kadar air ke program kontrol otomatis untuk memulai proses kerja mesin VTM-80.

V8 169.254.91.184 (raspberrypi) - VNC Viewer

SETPOINT

TEMPERATURE		0	C
SATURATION		0	%
1	2	3	
4	5	6	
7	8	9	
Clear	0	Enter	

Gambar 4.2 Tampilan Menu *Input* Kadar Air

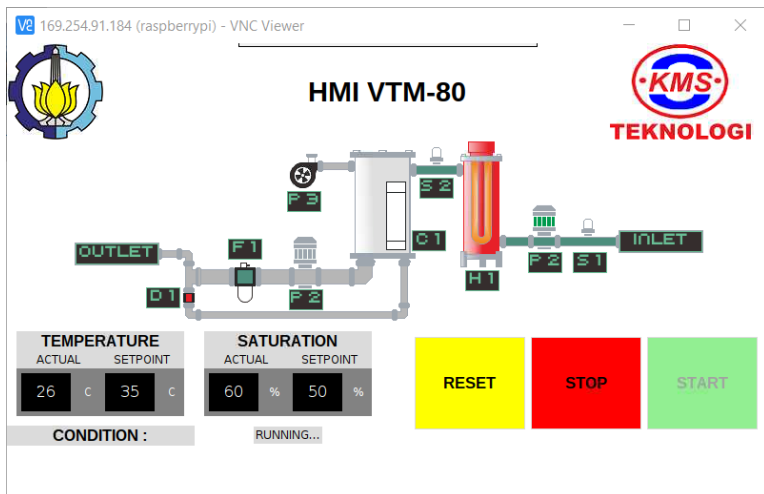
4.1.3 Tampilan HMI Saat Mesin Bekerja Normal

Pengujian ini bertujuan untuk mengetahui apakah perubahan kondisi komponen mesin VTM-80 serta nilai aktual kadar air dan temperatur dapat terbaca dengan baik pada HMI yang telah dibuat. Pengujian ini dilakukan dengan menggunakan miniatur plant yang telah dibuat dan diuji berdasarkan setiap kondisi kerja mesin.

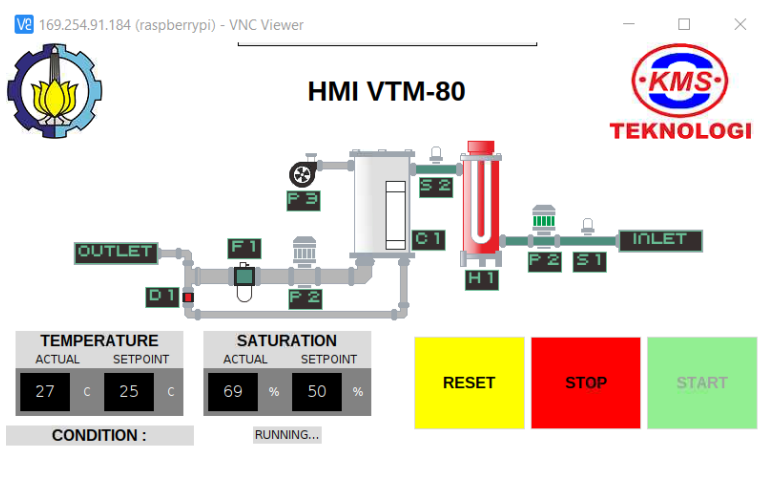
Pengujian dibagi menjadi 2 yaitu pengujian saat mesin berjalan normal dan pengujian saat terjadi *error* pada sistem plant VTM-80. Pengujian saat mesin berjalan normal yang meliputi kondisi *RUNNING !*, *CHECKING !*, *EMPTIED VACUUM CHAMBER !*, *FINNISH !* dengan kondisi *heater* ON dan OFF yang data hasil pengujiaannya di tampilkan pada Tabel 4.1 berikut ini.

Tabel 4.1 Pengujian Kondisi Komponen Plant VTM-80 Saat proses Bekerja Dengan Normal

Waktu	Kadar Air (%RH)		VC	P1	P2	H	P3c	DV
	Set Point	Aktual						
10.06	50	80	OFF	ON	OFF	OFF	OFF	ON
10.07	50	78	<i>Level 1</i>	ON	ON	OFF	ON	ON
10.09	50	75	<i>Level 1</i>	ON	ON	OFF	ON	ON
10.11	50	73	<i>Level 2</i>	OFF	ON	OFF	ON	OFF
10.14	50	63	<i>Level 2</i>	OFF	ON	OFF	ON	OFF
10.15	50	63	<i>Level 1</i>	ON	ON	OFF	ON	ON
10.18	50	64	<i>Level 2</i>	OFF	ON	OFF	ON	OFF
10.19	50	65	<i>Level 1</i>	ON	ON	ON	ON	ON
10.22	50	60	<i>Level 1</i>	ON	ON	ON	ON	ON
10.23	50	58	<i>Level 2</i>	OFF	ON	ON	ON	OFF
10.25	50	55	<i>Level 2</i>	OFF	ON	ON	ON	OFF
10.28	50	53	<i>Level 1</i>	ON	ON	ON	ON	ON
10.33	50	48	<i>Level 1</i>	OFF	ON	OFF	ON	ON
10.36	50	53	OFF	OFF	OFF	OFF	OFF	OFF



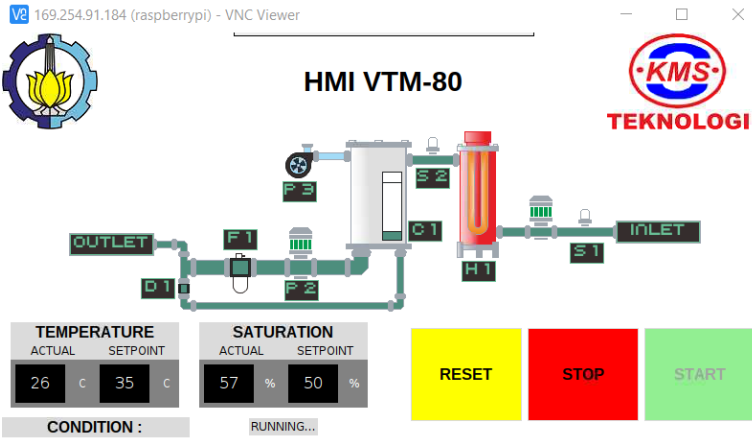
Gambar 4.3 Tampilan HMI Saat Oil Pump ON



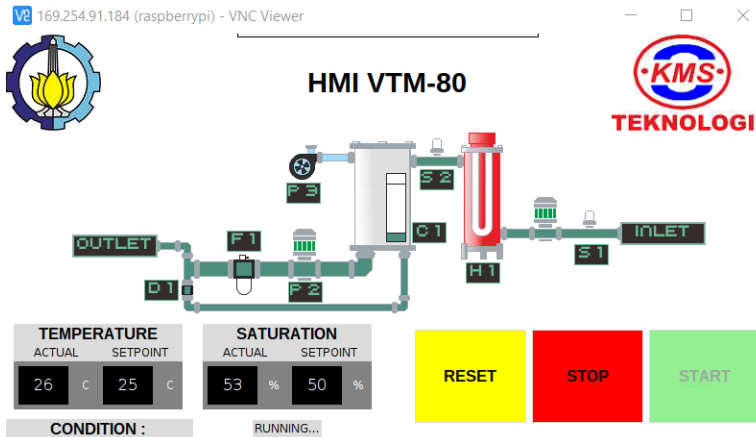
Gambar 4.4 Tampilan HMI Saat Oil Pump dan Heater Mati

Setelah tombol start dan *setpoint* dimasukkan, plant vtm-80 akan bekerja memisahkan oli dari campuran air. *Oil pump* (P1) memompa oli dari tangki oli memasuki *vacuum chamber* (C1) dan melewati *heater* (H1) untuk dipanaskan, apabila temperatur oli aktual yang terbaca pada plant VTM-80 lebih kecil dari nilai *setpoint* maka *heater* aktif dan tampilan hmi akan seperti pada Gambar 4.3, sedangkan apabila temperatur aktual oli sama dengan atau lebih besar dari *setpoint*, *heater* (H1) tidak aktif tampilan hmi akan seperti pada Gambar 4.4.

Oil pump (P1) akan terus memompa oli sehingga oli di dalam *vacuum chamber* mengaktifkan *switch level* 1. Saat *switch level* 1 menyala, *vacuum pump* (P3) aktif dan menyerap uap air yang terdapat pada oli. *Evaquation pump* (P2) aktif memompa oli dari *vacuum chamber* untuk melewati *filter unit* dan *directional valve* (D1) aktif untuk mengubah aliran oli kembali menuju *vacuum chamber* (C1) seperti pada Gambar 4.5 saat *heater* (H1) aktif dan Gambar 4.6 saat *heater* (H1) mati.



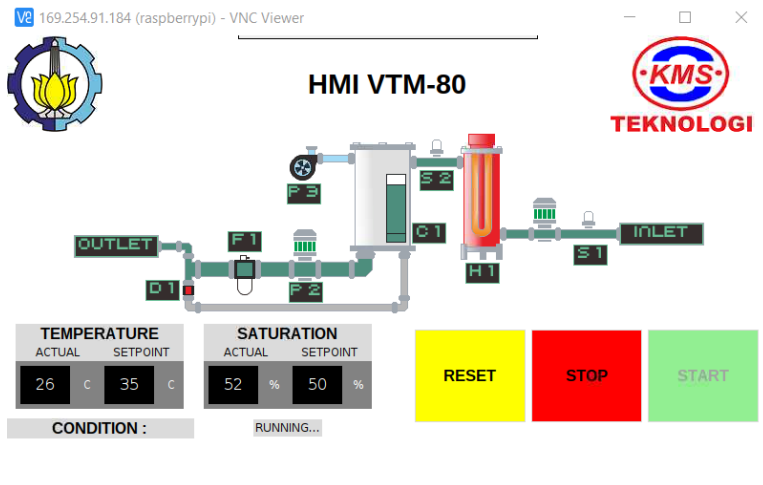
Gambar 4.5 Tampilan HMI Saat Kondisi *Level* 1



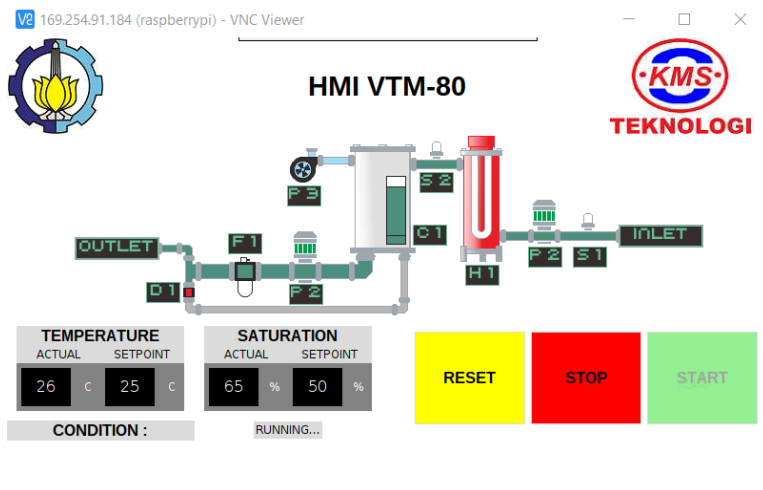
Gambar 4.6 Tampilan HMI Saat Kondisi *Level 1* dan *Heater* Mati

Oil pump (P1) dan *evaquation pump* (P2) tetap terus memompa oli ke dalam *vacuum chamber* (C1) sehingga ketinggian oli akan terus naik sehingga mengaktifkan *switch level 2*. Saat *switch level 2* menyala *oil pump* (P1) dan *directional valve* (D1) mati sehingga oli tidak lagi di alirkan ke *vacuum chamber* (C1) dan *evaquation pump* (P2) memompa oli kembali ke tangki oli seperti pada gambar 4.7 saat *heater* aktif dan Gambar 4.8 saat *heater* mati. Pada saat ini ketinggian air pada *vacuum chamber* akan terus menurun sehingga oli akan mematikan *switch level 1* dan 2. Setelah *switch level 1* mati. *Oil pump* dan *directional valve* akan aktif dan memompa oli ke *vacuum chamber* sampai oli pada *vacuum chamber* kembali mengaktifkan *switch level 2*. Proses ini berlangsung secara terus menerus sampai kadar air aktual pada oli mencapai nilai *setpoint*.

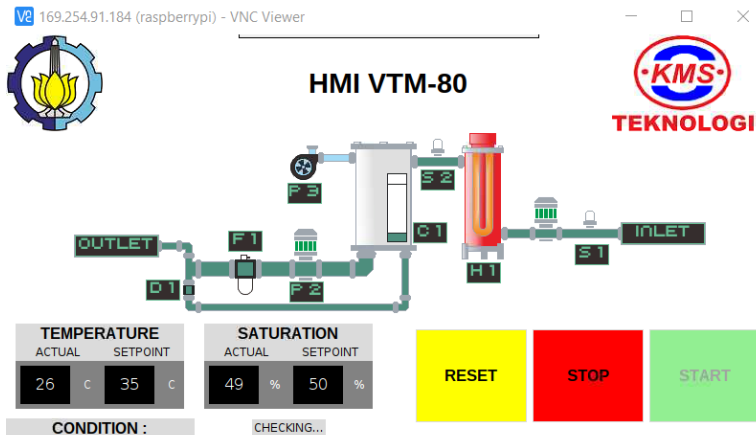
Apabila kadar air aktual pada oli telah mencapai *setpoint*, system VTM-80 akan melakukan pengecekan nilai kadar air aktual selama 5 menit untuk memastikan bahwa nilai kadar air aktual benar benar mencapai nilai *setpoint*. Apabila selama proses pengecekan nilai kadar air aktual diatas *setpoint* maka pengecekan gagal dan sistem kembali memisahkan air dari dalam oli. Proses ini ditampilkan pada hmi seperti pada Gambar 4.9 saat *heater* aktif dan Gambar 4.10 saat *heater* mati.



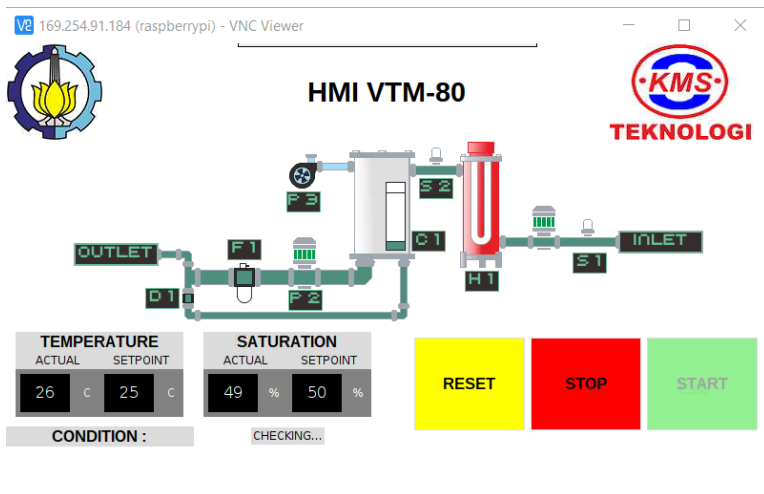
Gambar 4.7 Tampilan HMI Saat Kondisi *Level 2*



Gambar 4.8 Tampilan HMI Saat Kondisi *Level 2* dan *Heater Mati*



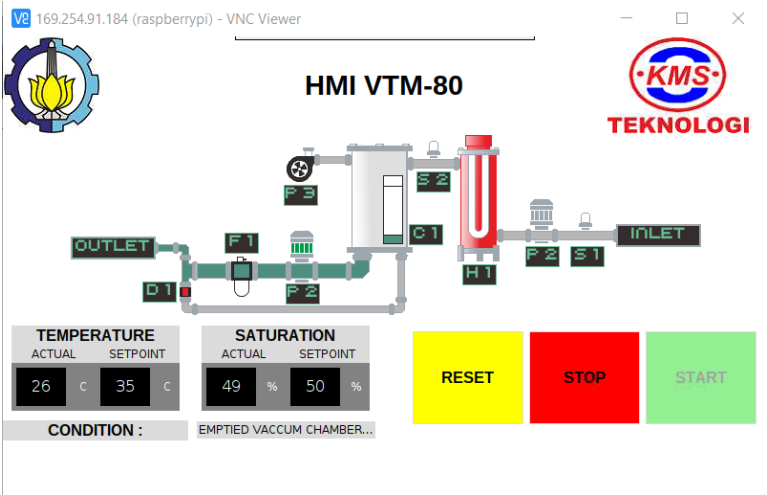
Gambar 4.9 Tampilan HMI Saat Pengecekan



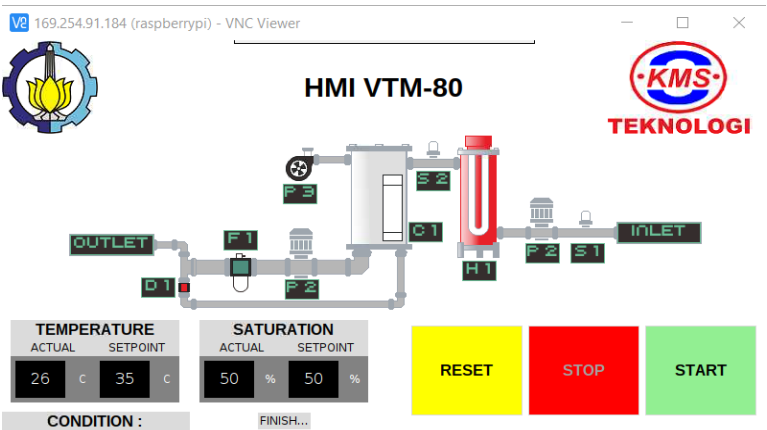
Gambar 4.10 Tampilan HMI Saat Kondisi Pengecekan dan Heater Mati

Setelah melewati proses pengecekan *evaquation pump* memompa sisa oli didalam *vacuum chamber* ke tangki oli dan plant

VTM-80 selesai bekerja. Tampilan HMI seperti pada Gambar 4.11 saat proses pengosongan oli di dalam *vacuum chamber* dan Gambar 4.12 saat proses selesai dijalankan.



Gambar 4.11 Tampilan HMI Saat Pengosongan Oli di *Vacuum Chamber*



Gambar 4.12 Tampilan HMI Saat Sistem Selesai Bekerja

4.1.4 Tampilan HMI Saat Mesin Bekerja Dengan Kondisi Error

Pengujian ini bertujuan untuk mengetahui respon HMI terhadap kondisi-kondisi *error* yang menghentikan proses kerja plant VTM-80. Kondisi-kondisi tersebut meliputi kondisi *switch level 1 error*, *vacuum chamber overload* dan *filter clogged*.

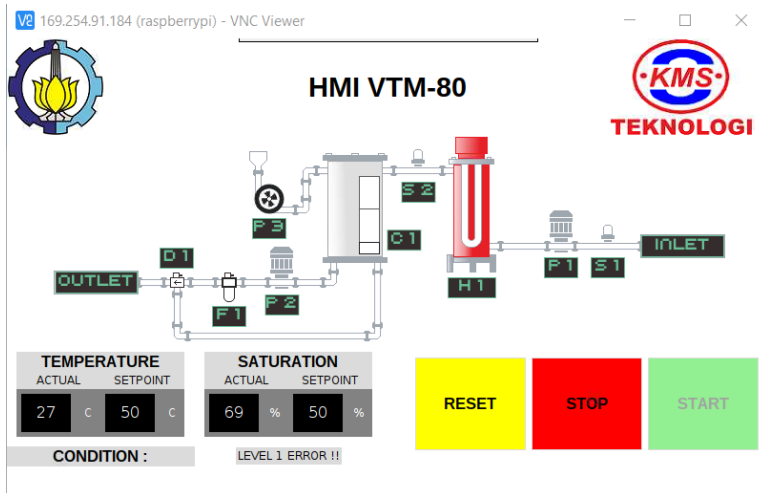
Switch level 1 error terjadi pada saat memulai proses pengisian *oil pump* menyala tetapi *switch level 1* tetap mati dalam jangka waktu 5 menit. Hal ini diakibatkan oleh *oil pump* tidak memompa oli dengan benar atau *switch level 1* mengalami kerusakan sehingga proses harus berhenti.

Pengujian dengan miniature plant dilakukan dengan menjalankan secara normal tetapi *switch* untuk *level 1* tidak diaktifkan sehingga mendapatkan kondisi yang sesuai dengan yang akan terjadi plant sesungguhnya. Data hasil pengujian terdapat pada Tabel 4.2 dengan tampilan hm1 seperti pada Gambar 4.13. Dengan peringatan pada *text condition : level 1 error*.

Setelah terjadi kondisi *level 1 error*, plant VTM-80 harus dilakukan pengecekan terhadap *oil pump*, *switch level* yang didalam *vacuum chamber*, dan tangki penampungan oli.

Tabel 4.2 Pengujian Kondisi Komponen Plant VTM-80 Saat Sistem Mengalami Error Pada Switch Level 1

Waktu	Kadar Air (%RH)		C1	P1	P2	H1	P3	D1
	Set Point	Aktual						
11.20	50	80	OFF	ON	OFF	ON	OFF	ON
11.22	50	80	OFF	ON	OFF	ON	OFF	ON
11.23	50	80	OFF	ON	OFF	ON	OFF	ON
11.26	50	80	OFF	OFF	OFF	OFF	OFF	OFF

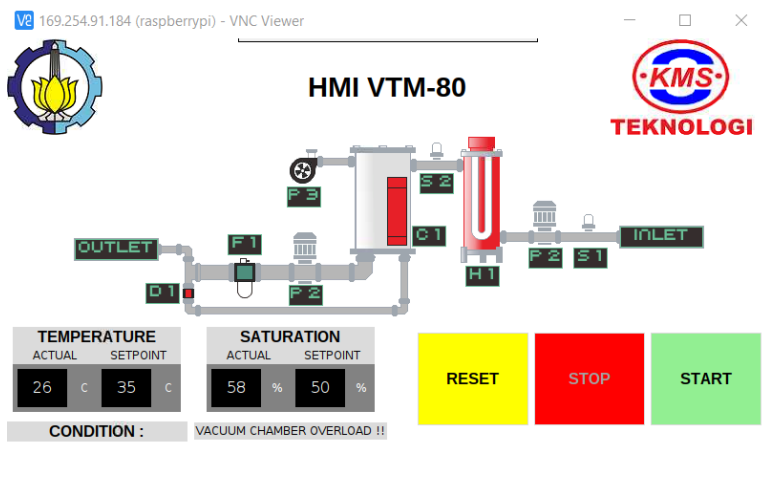


Gambar 4.13 Tampilan HMI Dengan *Contition Level 1 Error*

Pengujian selanjutnya adalah pengujian kondisi *error 'vacuum chamber overload'*. Hal ini diakibatkan oleh kerusakan pada *switch level 2* yang mengakibatkan *directional valve* dan *oil pump* tetap terus mengalirkan oli sampai oli mengaktifkan *switch level 3*.

pengujian menggunakan miniatur plant dengan menjalankan secara normal tetapi *switch level 3* dinyalakan sehingga mendapatkan kondisi yang sesuai dengan plant yang sesungguhnya. Setelah *switch level 3* plant berhenti bekerja dan *conditon* pada HMI menampilkan peringatan '*vacuum chamber overload !*' seperti pada Gambar 4.14. Data hasil pengujian terdapat pada Tabel 4.3.

setelah kondisi ini terjadi, operator harus melakukan pengecekan terhadap *switch level* yang berada didalam *vacuum chamber* dan *evaquation pump* apakah terjadi kerusakan pada komponen komponen tersebut sebelum program dijalankan kermbali.



Gambar 4.14 Tampilan HMI Dengan *Contition : Vacuum Chamber Overlaod*

Tabel 4.3 Pengujian Kondisi Komponen Plant VTM-80 Saat *Vacuum Chamber Overload (C1 berada pada Level 3)*

Waktu	Kadar Air (%RH)		C1	P1	P2	H1	P3	D1
	Set Point	Aktual						
11.30	50	80	OFF	ON	OFF	OFF	OFF	ON
11.32	50	80	OFF	ON	OFF	OFF	OFF	ON
11.33	50	77	Level 1	ON	ON	OFF	ON	ON
11.34	50	72	Level 1	ON	ON	ON	ON	ON
11.36	50	70	Level 1	ON	ON	ON	ON	ON
11.38	50	68	Level 1	ON	ON	ON	ON	ON
11.39	50	68	Level 1	ON	ON	ON	ON	ON
11.41	50	67	Level 3	OFF	ON	OFF	OFF	OFF

Tabel 4.3 Pengujian Kondisi Komponen Plant VTM-80 Saat *Vacuum Chamber Overload* (C1 berada pada Level 3)

Waktu	Kadar Air (%RH)		C1	P1	P2	H1	P3	D1
	Set Point	Aktual						
11.44	50	67	Level 3	OFF	ON	OFF	OFF	OFF
11.47	50	67	Level 3	OFF	OFF	OFF	OFF	OFF

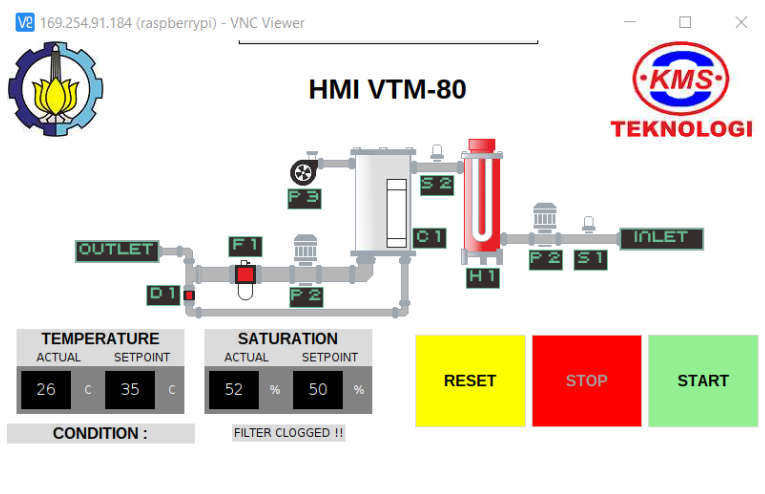
Pengujian kondisi *error* terakhir adalah pengujian saat *filter clogged* atau *filter unit* (F1) tersumbat sehingga *filter* harus diganti agar partikel partikel debu pada oli dapat disaring dan tidak memberatkan kerja *evacuation pump* (P2). Pengujian dengan miniatur plant dengan menjalankan secara normal dan ditengah tengah proses *switch filter* dinyalakan sehingga mendapatkan kondisi yang sesuai dengan plant yang sesungguhnya. Data hasil pengujian terdapat pada Tabel 4.4. Pada saat VTM-80 bekerja dan *switch* pada *filter unit* menyala, plant langsung berhenti bekerja dan *condition* pada hmi menampilkan peringatan *filter clogged* seperti pada Gambar 4.15

Tabel 4.4 Pengujian Kondisi Komponen Plant VTM-80 Saat *Filter Tersumbat* (F1 ON)

Waktu	Kadar Air (%RH)		C1	P1	P2	F1	P3	D1
	Set Point	Aktual						
13.10	50	70	OFF	ON	OFF	OFF	OFF	ON
13.12	50	70	OFF	ON	OFF	OFF	OFF	ON
13.14	50	68	Level 1	ON	ON	OFF	ON	ON
13.16	50	65	Level 1	ON	ON	OFF	ON	ON
13.19	50	60	Level 2	OFF	ON	OFF	ON	OFF
13.21	50	60	Level 2	OFF	ON	OFF	ON	OFF

Tabel 4.4 Pengujian Kondisi Komponen Plant VTM-80 Saat *Filter Tersumbat (F1 ON)*

Waktu	Kadar Air (%RH)		C1	P1	P2	F1	P3	D1
	Set Point	Aktual						
13.23	50	56	Level 1	OFF	ON	ON	OFF	OFF
13.24	50	56	Level 1	OFF	ON	ON	OFF	OFF
13.28	50	56	OFF	OFF	OFF	ON	OFF	OFF

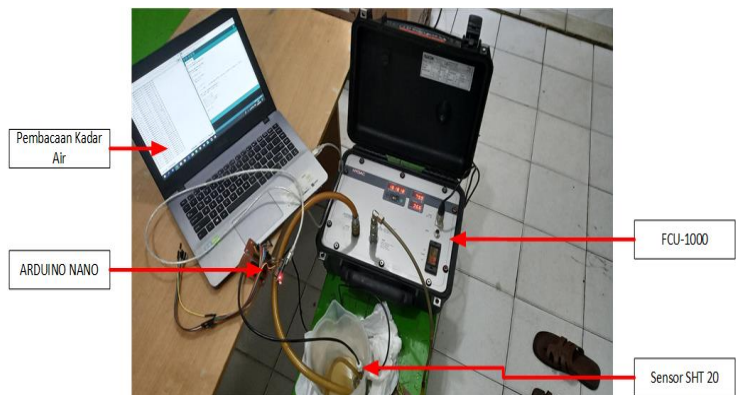


Gambar 4.15 Tampilan HMI Dengan *Contition : Filter Clogged*

4.2 Pengujian Nilai Kadar Air pada Sensor SHT-20

Pengujian ini dilakukan dengan menggunakan FCU (*fluid control unit*) – 1000 pada gambar 4.20 yang merupakan alat untuk membaca persentase kadar air (%RH) pada oli. Pengujian ini bertujuan untuk mengetahui keakuratan pembacaan oleh temperatur oleh sensor ds18b20 yang telah dikalibrasikan nilainya oleh program sistem kontrol otomatis yang menggunakan *Arduino Nano* yang dibahas pada buku

yang lain. Data hasil pembacaan tersebut merupakan data yang akan ditampilkan pada bagian nilai temperatur aktual didalam HMI. Data hasil pengujian terdapat pada Tabel 4.5 dan proses pengujian dilakukan seperti pada Gambar 4.16



Gambar 4.16 Pengujian Sensor SHT20



Gambar 4.17 Alat FCU1000

Tabel 4.5 Hasil Pengujian Nilai Kadar Air Sensor SHT-20
Dengan Alat Ukur FCU-1000

No.	FCU-1000 (%RH)	SHT-20 (%RH)	Error (%)
1	34,1	33,6	1,46
2	35,2	34,6	1,70
3	36,3	35,8	1,37
4	38,8	38,2	1,54
5z	39,3	38,4	2,29
6	42,1	41,3	1,92
7	43,5	42,6	2,06
8	45,4	44,4	2,20
9	47,2	46,3	1,90
10	46,8	46	1,70
11	52	50,9	2,11
12	55,3	54,4	1,62
13	58,4	57,4	1,71
14	63	62,1	1,42
15	67,9	67	1,32
16	69,4	69,6	2,88
17	71,9	71,9	0
18	76,8	77,4	0,52
19	77	78,1	1,69
20	77,6	78,1	0,64

Dengan persentase *error* yang didapat dengan menggunakan Persamaan 4.1 sebagai berikut :

$$(\%)Error = \frac{(\text{Nilai Aktual} - \text{Nilai Sensor})}{\text{Nilai Aktual}} \times 100\% \quad (4.1)$$

Dengan perhitungan rata rata *Error* sebesar 1,46 % yang dihitung melalui Persamaan 4.2:

$$(\%)Error \text{ rata - rata} = \frac{(d_1 + d_2 + d_3 + \dots d_n)}{n} \quad (4.2)$$

Dengan nilai persentase *error* rata rata sebesar 1,46 % merupakan persentase *error* yang sudah cukup kecil sehingga data hasil pembacaan sensor yang telah dikalibrasikan dengan FCU-1000 melalui program control otomatis dapat ditampilkan langsung pada HMI.

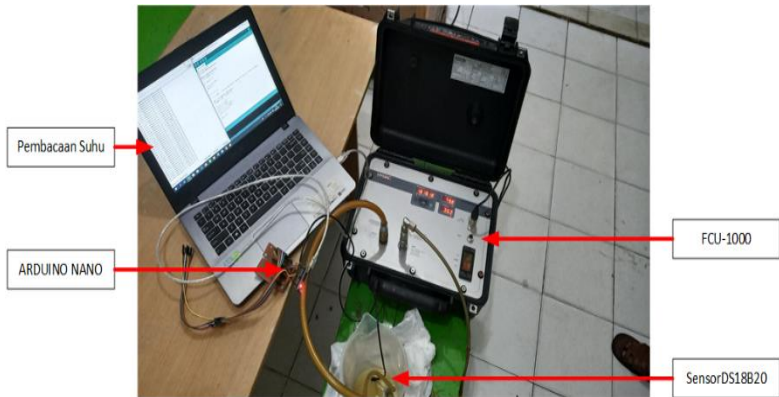
4.3 Pengujian Nilai Temperatur DS18B20

Pengujian ini dilakukan dengan menggunakan FCU (*Fluid Control Unit*) – 1000 yang terdapat termometer digital untuk mengukur temperatur pada oli. Pengujian ini bertujuan untuk mengetahui keakuratan pembacaan temperatur oleh sensor DS18B20 yang telah dikalibrasikan oleh program. Data hasil pembacaan tersebut merupakan data yang akan ditampilkan pada bagian nilai temperatur aktual didalam HMI. Data hasil pengujian terdapat pada Tabel 4.6 dan proses pengujian dilakukan seperti pada Gambar 4.18

Tabel 4.6 Hasil Kalibrasi Sensor DS8B20

No.	FCU1000 (°C)	DS18B20 (°C)	%Error
1	41,6	39,6	4,81
2	41,8	40	4,31
3	42,1	40,3	4,28
4	42,7	40,8	4,45
5	43,1	41,2	4,41
6	43,5	41,5	4,60
7	43,7	41,7	4,58
8	43,9	41,9	4,56
9	44	42,1	4,32
10	44,3	42,3	4,51
11	44,5	42,5	4,49
12	47,7	45,4	4,82
13	48,1	45,7	4,99
14	48,4	46,1	4,75
15	48,8	46,6	4,51
16	50	47	6,00
17	50,3	47,4	5,77
18	50,8	48,8	3,94
19	51	48,3	5,29
20	51,1	48,5	5,09
21	51,3	49	4,48
22	51,6	48,9	5,23
23	51,9	49,6	4,43
24	52	49,8	4,23

Dengan persentase *error* yang didapat dengan menggunakan Persamaan 4.1 dan rata rata *error* menggunakan Persamaan 4.2 dengan nilai rata rata *error* 5 % dari pengukuran menggunakan alat ukur FCU - 1000



Gambar 4.18 Pengujian Sensor DS18B20

---Halaman ini sengaja dikosongkan---

BAB V

PENUTUP

4.4 Kesimpulan

Dari hasil penelitian dan pembahasan mengenai penerapan *human machine interface* pada mesin *Filter* oli VTM-80 menggunakan *Raspberry Pi*. Dapat disimpulkan bahwa :

1. Antarmuka pada HMI dapat tertampil dan tombol – tombol fitur serta *monitoring* temperatur , kadar air dan plant VTM-80 dapat bekerja sesuai dengan fungsinya.
2. Pengujian antarmuka dari kondisi komponen VTM-80 dilakukan menggunakan *miniature plant* dengan cara melihan perubahan kondisi antarmuka dari HMI terdapat perubahan input yang diberikan
3. Pembacaan nilai aktual dari kadar air dan temperatur dilakukan melalui sensor SHT-20 untuk kadar air dan DS18B20 untuk temperature. Hasil pembacaan sensor tersebut akan dikalibrasikan oleh sistem kontrol otomatis yang menggunakan *Arduino* dan hasil pembacaan dikirim ke *Raspberry Pi* secara komunikasi serial untuk ditampilkan pada pembacaan HMI.

4.5 Saran

Untuk pengembangan dan penyempurnaan penerapan *human machine interface* pada mesin *filter* oli VTM-80 menggunakan *Raspberry Pi* maka bisa diberikan beberapa saran yang harapannya bisa diaplikasikan kelak di kemudian hari sebagai berikut :

1. Pemrograman ditambahkan dengan *multi-threading* agar pembacaan data yang dikirim oleh *Arduino* tidak terganggu oleh program penampilan antarmuka.
2. Lakukan perbandingan dengan *fluid control unit* yang sudah ada agar dapat menjadi perbandingan atau acuan nilai dari hasil pengukuran temperatur dan kadar air

---Halaman ini sengaja dikosongkan---

DAFTAR PUSTAKA

- [1] KMS Teknologi, “*Vacuum Technology Oil Purifier (VTM-80 SERIES)*”. **Brosur Produk**, Surabaya, 2016.
- [2] *UK DTL, “Contamination Control in Fluid Power Systems, Vol 1 Field Studies”*. Dept. Trade and Industry, N.E.L., East Kilbride, Glasgow, UK, 1981.
- [3] *Mike Day Pall Europe Ltd., UK, “Setting Control Limits for Water Contamination In Hydraulic and Lube Systems”*, ASLE Transactions, American Society of Lubrication Engineers, Volume 20, No. 3, pp. 244-248, 1977, from a presentation at the 31st Annual ASLE Meeting, Philadelphia, PA
- [4] *IP 386/90, “Water In Crude Oil By Coulometric Karl Fischer Titration”*, Institute of Petroleum, London, UK, 1990.
- [5] Eugene C. Lister, “**Mesin dan Rangkaian Listrik Edisi keenam**”, Erlangga, Yogyakarta, 1993.
- [6] Sumbodo, Wirawan dan Sigit Poedjiono, “**Pneumatik dan Hidrolik**”. Deepublish, Yogyakarta, 2017.
- [7] Aripriharta, S.T., M.T. 2014. “**Smart Relay dan Aplikasinya**”, Yogyakarta: Graha Ilmu
- [8] *S.K Datta. 1995. “Power Electronics and Control”. Virginia: Ristor Publisng Company.*
- [9] Mochamad Fajar Wicksono dan Hidayat, “**Mudah Belajar Mikrokontroler Arduino**”, Informatika, Bandung, 2017
- [10] Robertson, John B. 1992. “**Keterampilan Teknik Listrik Praktis**”. Bandung: YRAMA WIDYA.
- [11] adafruit, “**ADS1115 16-BIT ADC - 4 CHANNEL**,” adafruit, New York.
- [12] W Zamora, R., Wildian, & Harmadi. (2015). “**Perancangan Alat Ukur TDS (Total Dissolved Solids) Air dengan Sensor Konduktivitas Secara Real Time**”, VII(1), 11–15.
- [13] Anita Rahmawati, Slamet Winardi, d. t. (2012). “**Rancang Bangun Alat Pengukur Suhu Tubuh Dengan Tampilan Digital Dan Keluaran Suara Berbasis Mikrokontroller AVR At Mega 8535**”. monitor, 1, 1.
- [14] Pujiono, “**Rangkaian Listrik**”, Graha Ilmu, Yogyakarta, 2013.
- [15] Andi Dinata, “*Physical Computing dengan Raspberry Pi*”, Gramedia, Jakarta, 2017.

- [16] R. H. Sianipar, "*Algoritma, Struktur Data dan Pemrograman GUI*", Informatika, Bandung, 2013.

LAMPIRAN A

DOKUMENTASI KEGIATAN

A.1 Diskusi sistem VTM-80 di CV. KMS TEKNOLOGI



A.2 Mengoperasikan VTM-80



A.3 Mengamati Wiring VTM-80

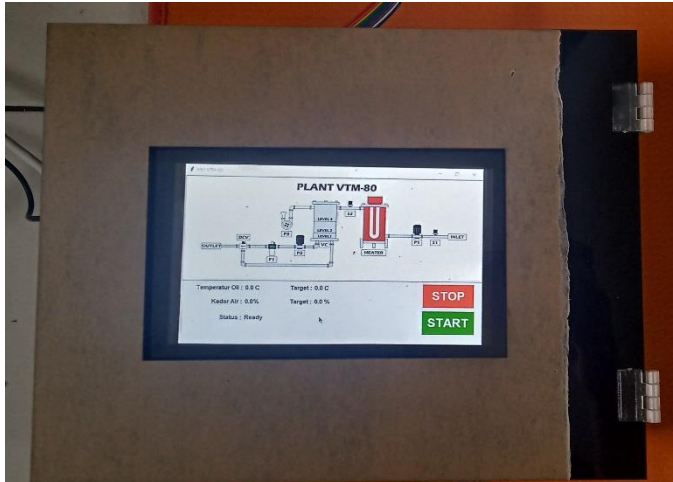


A.4 Kalibrasi Sensor SHT20 dan DS18B20 Menggunakan FCU1000

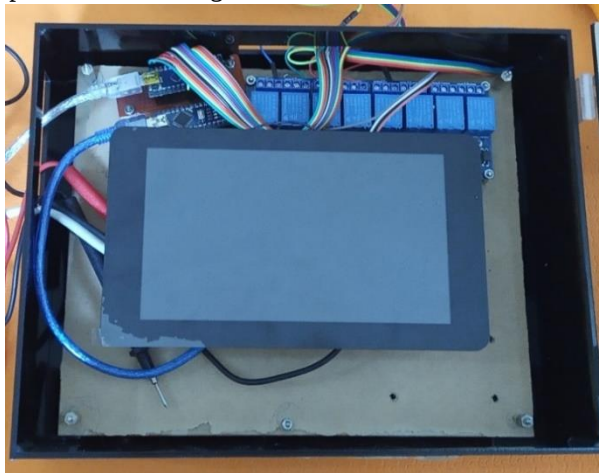


LAMPIRAN B TAMPILAN ALAT

B.1 Tampilan Alat HMI Bagian Luar



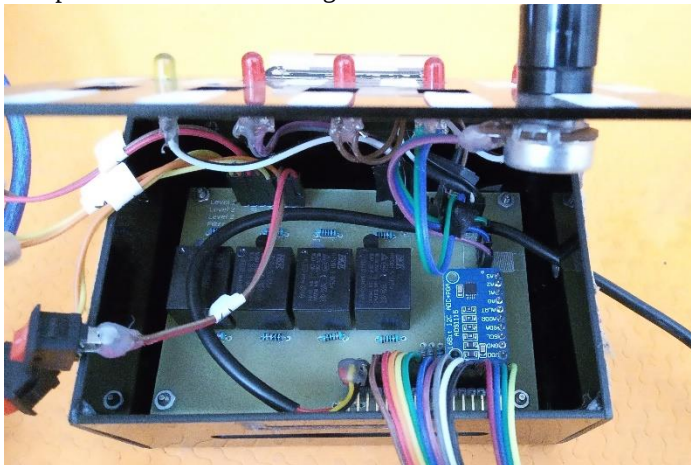
B.2 Tampilan Alat HMI Bagian Dalam



B.3 Tampilan *Miniature Plant* Bagian Luar



B.4 Tampilan *Miniature Plant* Bagian Dalam



RIWAYAT PENULIS



Nama : Muhammad Amin Rais
TTL : Boyolali, 08 Januari 2000
Jenis Kelamin : Laki - Laki
Agama : Islam
Alamat Rumah : Jln. Kancil IX, E/143,
Sertajaya, Cikarang Timur,
Kab. Bekasi, Jawa Barat
Telp/HP : 082298812916
E-mail : muhammad.amin.rais23
@gmail.com

RIWAYAT PENDIDIKAN

1. 2005 -2011 : SDN Mekarmukti 06
2. 2011 - 2014 : SMPN 04 Cikarang Utara
3. 2014 - 2016 : SMAN 01 Cikarang Utara
4. 2016 - Sekarang : Departemen Teknik Elektro Otomasi-Fakultas
Vokasi (FV) -Institut Teknologi Sepuluh
Nopember (ITS)

PENGALAMAN KERJA

1. Magang CV. KMS Teknologi, Surabaya JawaTimur
(Agustus 2018 – Desember 2018)