



TUGAS AKHIR TF 145565

**RANCANG BANGUN SISTEM *MONITORING* PADA
PRODUKSI *BROWN'S GAS* MENGGUNAKAN
GENERATOR *HHO TYPE DRY CELL***

DIMASWARI EGI ASMIRESI
NRP. 1051150000043

Dosen Pembimbing I
Arief Abdurrahman, S.T., M.T.
NIP. 19870712 201404 1 002

Dosen Pembimbing II
Dwi Oktavianto Wahyu Nugroho, S.T., M.T.
NIP. 1983201711054

**PROGRAM STUDI DIII TEKNOLOGI INSTRUMENTASI
DEPARTEMEN TEKNIK INSTRUMENTASI
FAKULTAS VOKASI
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2018**



TUGAS AKHIR - TF 145565

**RANCANG BANGUN SISTEM *MONITORING* PADA
PRODUKSI *BROWN'S GAS* MENGGUNAKAN
GENERATOR HHO *TYPE DRY CELL***

DIMASWARI EGI ASMIRESI
NRP 1051150000043

Dosen Pembimbing I
Arief Abdurrahman, S.T., M.T.
NIP. 19870712 201404 1 002

Dosen Pembimbing II
Dwi Oktavianto Wahyu Nugroho, S.T., M.T.
NIP. 19832017111054

**PROGRAM STUDI DIII TEKNOLOGI INSTRUMENTASI
DEPARTEMEN TEKNIK INSTRUMENTASI
FAKULTAS VOKASI
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2018**



FINAL PROJECT - TF 145565

***DESIGN OF MONITORING SYSTEM ON
BROWN'S GAS PRODUCTION USING
GENERATOR HHO TYPE DRY CELL***

**DIMASWARI EGI ASMIRESI
NRP 1051150000043**

**Advisor Lecturer I
Arief Abdurrakhman, S.T., M.T.
NIP. 19870712 201404 1 002**

**Advisor Lecturer II
Dwi Oktavianto Wahyu Nugroho, S.T., M.T.
NIP. 19832017111054**

***STUDY PROGRAM OF D3 INSTRUMENTATION
TECHNOLOGY DEPARTMENT OF INSTRUMENTATION
ENGINEERING
FACULTY OF VOCATION
SEPULUH NOPEMBER INSTITUTE OF
TECHNOLOGY SURABAYA
2018***

LEMBAR PENGESAHAN

**RANCANG BANGUN SISTEM MONITORING PADA
PRODUKSI *BROWN'S GAS* MENGGUNAKAN
GENERATOR *HHO TYPE DRY CELL***

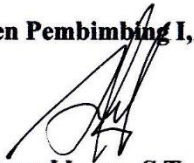
TUGAS AKHIR

Oleh :

DIMASWARI EGI ASMIRESI
NRP. 1051150000043

Surabaya, 24 Juli 2017
Mengetahui dan Menyetujui,

Dosen Pembimbing I,



Arief Abdurrahman, S.T., M.T.
NIP. 19870712 201404 1 002

Dosen Pembimbing II,



Dwi Oktavianto W. N., S.T., M.T.
NIP. 1983201711054

**Kepala Departemen Teknik
Instrumentasi ITS,**



Dr. Ir. Furwadi Agus D., M.Sc
NIP. 19620822 198803 1 001

LEMBAR PENGESAHAN

**RANCANG BANGUN SISTEM MONITORING PADA
PRODUKSI *BROWN'S GAS* MENGGUNAKAN
GENERATOR *HHO TYPE DRY CELL***

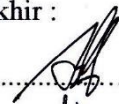
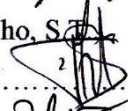

TUGAS AKHIR

Diajukan Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Ahli Madya
pada
Program Studi D3 Teknik Instrumentasi
Departemen Teknik Instrumentasi
Fakultas Vokasi
Institut Teknologi Sepuluh Nopember

Oleh :

**DIMASWARI EGI ASMIRESI
NRP. 1051150000043**

Disetujui oleh Tim Penguji Tugas Akhir :

1. Arief Abdurrakhman, S.T., M.T.....  (Pembimbing I)
2. Dwi Oktavianto Wahyu Nugroho, S.T.
M.T.....  (Pembimbing II)
3. Herry Sufyan Hadi, S.T., M.T.  (Penguji I)

**SURABAYA
JULI 2018**

RANCANG BANGUN SISTEM *MONITORING* PADA PRODUKSI *BROWN'S GAS* DENGAN MENGGUNAKAN GENERATOR *HHO TYPE DRY CELL*

Nama : Dimaswari Egi Asmiresi
NRP : 1051150000043
Jurusan : D3 Teknik Instrumentasi, ITS Surabaya
Pembimbing 1 : Arief Abdurrakhman S.T., M.T.
Pembimbing 2 : Dwi Oktavianto Wahyu Nugroho, S.T., M.T.

Abstrak

Energi merupakan kebutuhan dasar manusia yang terus meningkat, khususnya konsumsi energi pada sektor transportasi yang cukup tinggi hampir 95% dari bahan bakar fosil. Kondisi ini mendorong manusia untuk mencari bahan bakar alternatif yang bersifat terbarukan yaitu Hidrogen. Cara mendapatkan gas hidrogen adalah dengan elektrolisa air dari campuran akuades dan KOH yang akan menghasilkan gas H₂ dan O₂, gas tersebut diberi nama *Brown's Gas*. *Brown's gas* sangat baik dalam mengurangi konsumsi bahan bakar, meningkatkan kinerja mesin, dan memperbaiki kualitas emisi gas buang mesin. Produksi *Brown's Gas* tersebut memerlukan sistem monitoring level, tekanan, suhu dan kandungan oksigen pada *Brown's Gas*. Hal tersebut diperlukan untuk memastikan keamanan dan melindungi peralatan serta untuk pemantauan kualitas *Brown's Gas*. Sistem monitoring ini menggunakan sensor *level float*, termokopel *type K*, MPX5700 AP yang diproses melalui Atmega32 dan ditampilkan pada LCD 20x4. SDcard memiliki kapasitas 8 GB dan dapat bertahan selama ± 4,5 tahun. Dari SDCard diperoleh data bahwa produksi *Brown's Gas* telah mencapai ideal. Produksi dilakukan selama 2 jam, nilai suhu mencapai 40°C-44°C, saat kondisi *level* larutan mencapai *low*, kandungan oksigen 15% serta tekanan pada storage 0.44 psi.

Kata kunci : *Brown's Gas*, monitoring, *level float two ball*, penyimpanan data

**DESIGN OF MONITORING SYSTEM ON BROWN'S GAS
PRODUCTION USING GENERATOR HHO TYPE DRY
CELL**

Nama : Dimaswari Egi Asmiresi
NRP : 1051150000043
Jurusan : *Diploma of Instrumentation Engineering, ITS
Surabaya*
Pembimbing 1 : Arief Abdurrahman S.T., M.T.
Pembimbing 2 : Dwi Oktavianto Wahyu Nugroho, S.T., M.T.

Abstract

Energy is an increasingly basic human need, especially energy consumption in the transport sector which is quite high at almost 95% of fossil fuels. This condition encourages people to look for alternative fuels that are renewable ie Hydrogen. The way to get hydrogen gas is by electrolyzing water from the mixture of aquades and KOH which will produce H₂ and O₂ gas, the gas is named Brown's Gas. Brown's gas is excellent in reducing fuel consumption, improving engine performance, and improving engine exhaust emissions. Brown's Gas production requires a level monitoring system, pressure, temperature and oxygen content in Brown's Gas. It is necessary to ensure the safety and protection of equipment and for Brown's Gas quality monitoring. This monitoring system uses float level sensors, K type thermocouples, MPX5700 AP processed through Atmega32 and displayed on 20x4 LCD. SD card has a capacity of 8 GB and can last for ±4.5 years. From SDCard obtained data that the production of Brown's Gas has reached the ideal. Production is done for 2 hours, temperature value reaches 40oC-44oC, when condition of solution level reach low, 15% oxygen content and pressure at storage 0,44 psi.

Keywords : *Brown's Gas, monitoring, level float two ball, data storage*

KATA PENGANTAR

Puji syukur kehadirat Allah SWT atas berkat, rahmat dan kebesaran-Nya sehingga penulis dapat menyelesaikan Tugas Akhir dengan judul “Rancang Bangun Sistem *Monitoring Level, Tekanan, Suhu, Oksigen* pada Produksi *Brown’s Gas* dengan menggunakan Generator *HHO type dry cell* “ tepat pada waktunya.

Selama menyelesaikan tugas akhir ini penulis telah banyak mendapatkan bantuan dari berbagai pihak. Oleh karena itu pada kesempatan ini penulis ingin mengucapkan terima kasih yang sebesar-besarnya kepada :

1. Orang tua yang telah memberikan segala dukungan baik moral maupun materil,
2. Bapak Dr. Ir. Purwadi Agus Darwito, M.Sc selaku Kepala Departmen D3 Teknik Instrumentasi ITS,
3. Bapak Arief Abdurrahman S.T., M.T. selaku dosen pembimbing 1 yang telah meluangkan waktu, arahan dan saran yang membantu selama proses pengerjaan tugas akhir,
4. Bapak Ir. Matradji, MI. Kom selaku dosen wali kami yang telah memberikan semangat dan motivasi kepada kami,
5. Bapak Dwi Oktavianto Wahyu Nugroho, S.T., M.T. selaku dosen pembimbing 2 yang telah meluangkan waktu, arahan dan saran yang membantu selama proses pengerjaan tugas akhir
6. Terimakasih kepada mas dan mbak yang telah mendukung, memberikan motivasi selama ini,
7. Terimakasih kepada tim *Brown’s Gas* (Vivi Normalia, Aditya Ctur dan Alkhabib Luthfi) yang bersama-sama berjuang dalam pengerjaan tugas akhir ini,

8. Teman-teman F50.3 yang selalu senantiasa memberikan semangat dan semua pihak yang telah membantu terselesaikannya Tugas Akhir ini,
9. Serta semua pihak yang tidak dapat penulis sebutkan satu persatu.

Penulis menyadari bahwa kesempurnaan hanya milik Allah SWT. Oleh sebab itu, penulis sangat berterimakasih atas segala masukan, kritik dan saran yang membangun dari pembaca agar laporan ini menjadi lebih baik untuk di kemudian hari. Demikian laporan ini penulis buat, semoga laporan ini dapat memberikan manfaat selain bagi penulis sendiri, dan bagi pembaca sekalian.

Surabaya, 5 Juli 2017

Penulis

Dimaswari Egi Asmiresi
NRP. 10 51 15 000 00 043

DAFTAR ISI

HALAMAN JUDUL	i
LEMBAR PENGESAHAN I	iii
LEMBAR PENGESAHAN II	iv
ABSTRAK	v
ABSTRACT	vi
KATA PENGANTAR	vii
DAFTAR ISI	ix
DAFTAR GAMBAR	xi
DAFTAR TABEL	xiii

BAB I PENDAHULUAN

1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	3
1.3 Batasan Masalah	3
1.4 Tujuan.....	3
1.5 Manfaat.....	4

BAB II DASAR TEORI

2.1 <i>Brown's Gas</i>	5
2.2 Sistem Monitoring.....	6
2.3 Elektrolisis	14
2.4 Generator gas HHO	18
2.5 Netral Plat Generator HHO.....	20
2.6 Sensor <i>Level Float</i>	21
2.7 Mikrokontroler Atmega32	22
2.8 Software Code Vision AVR.....	23
2.9 LCD 20x4	23
2.10 RTC	25
2.11 <i>Openlog Open Source DataLogger</i>	27
2.12 Komunikasi data	28

BAB III PERANCANGAN DAN PEMBUATAN ALAT

3.1 <i>Flow Chart</i> / Diagram Alir Pembuatan Alat	33
3.2 Gambaran Umum Produksi Brown's Gas pada Sepeda Motor.....	34
3.3 Perancangan Sistem Monitoring	35
3.4 Perancangan <i>Hardware</i>	37
3.5 Perancanagn <i>Firmtware</i>	40

BAB IV ANALISIS DATA DAN PEMBAHASAN

4.1 Hasil Rancang Bangun Alat	47
4.2 Pengujian Sensor <i>Level Float</i>	48
4.3 Hasil RTC	49
4.4 Hasil Openlog Data Logger	50

BAB V KESIMPULAN DAN SARAN

5.1 Kesimpulan.....	53
5.2 Saran.....	53

DAFTAR PUSTAKA

xiv

LAMPIRAN

LAMPIRAN A (Data Sheet)

LAMPIRAN B (*Wiring Plant Brown's Gas*)

LAMPIRAN C (*Listing Program pada CVAVR*)

BIODATA PENULIS

DAFTAR GAMBAR

Gambar 2.1	Pemecahan molekul air menjadi gas HHO	5
Gambar 2.2	Diagram Block Sistem Pengukuran.....	6
Gambar 2.3	Tabel T student.....	11
Gambar 2.4	Respon orde nol	12
Gambar 2.5	Respon Orde Satu.....	13
Gambar 2.6	Respon Orde Dua	13
Gambar 2.7	Rangkaian dasar sistem elektrolit	14
Gambar 2.8	Hubungan Konsentrasi KOH dan Konduktivitas..	17
Gambar 2.9	Standard komposisi stainless steel (David Tabolt, 1998)	18
Gambar 2.10	Generator HHO Tipe <i>Dry Cell</i>	19
Gambar 2.11	Generator HHO Tipe <i>wet cell</i>	20
Gambar 2.12	Netral Plat.....	21
Gambar 2.13	Bentuk sensor Level Float	22
Gambar 2.14	IC Chip Atmega32	22
Gambar 2.15	LCD 20x4.....	24
Gambar 2.16	RTC DS1307.....	26
Gambar 2.17	Konfigurasi PIN RTC DS1307	26
Gambar 2.18	<i>Openlog data logger</i>	27
Gambar 2.19	AVR USART TX dan RX.....	29
Gambar 2.20	Cara kerja protocol SPI	30
Gambar 2.21	Konfigurasi sistem I2C.....	31
Gambar 3.1	Flowchart Pengerjaan Tugas Akhir	33
Gambar 3.2	Blok Flow Diagram Produksi <i>Brown's Gas</i>	35
Gambar 3.3	<i>Piping & Instrumentation Diagram</i> dalam Sistem Monitoring dalam produksi <i>Brown's Gas</i>	36
Gambar 3.4	Diagram Blok sistem <i>Monitoring level</i>	37
Gambar 3.5	Peletakkan sensor <i>level float</i>	38
Gambar 3.6	Konfigurasi display LCD 20x4	40
Gambar 3.7	Listing tampilan program	41

Gambar 3.8	Flowchart pembacaan RTC	43
Gambar 3.9	Tampilan Software Khazama	45
Gambar 4.1	Plant Produksi <i>Brown's Gas</i>	48
Gambar 4.2	Hasil RTC DS1307	49
Gambar 4.3	Data Paket Pengiriman SDCard	51

DAFTAR TABEL

Tabel 3.1 Konfigurasi Pada Atmega32	39
Tabel 4.1 Hasil Pengujian Sensor <i>Level Float</i>	48
Tabel 4.2 Hasil pengujian RTC	49
Tabel 4.3 Penyimpanan Data saat level kondisi <i>Low</i>	53
Tabel 4.3 Penyimpanan Data saat level kondisi <i>Medium</i>	54
Tabel 4.3 Penyimpanan Data saat level kondisi <i>High</i>	54

BAB I PENDAHULUAN

1.1 Latar Belakang

Energi merupakan kebutuhan dasar manusia, yang terus meningkat sejalan dengan tingkat kehidupannya. Energi nasional sangat membutuhkan Bahan Bakar Minyak (BBM) hingga memegang posisi yang sangat dominan (IPTEK, 2015). Data dari *Indonesian Petroleum Association* (IPA) dijelaskan bahwa permintaan energi di Indonesia pada tahun 2010 adalah 3,3 juta *barrel oil equivalent per day* (*boepd*). Pada tahun 2025, permintaan energi diperkirakan meningkat menjadi 7,7 juta *barrel oil equivalent per day*. *Indonesian Petroleum Association* (IPAdan pada saat ini terjadi gejala krisis energi di Indonesia. Tahun 2020 perkiraan Indonesia kekurangan pasokan minyak dan gas sebesar 2,4 - 2,5 juta *barrel oil equivalent per day* (Data IPA, 2010), karena terjadi penurunan produksi minyak bumi dari tahun ke tahun sekitar 4,41% per tahun sejak tahun 2014. Konsumsi energi pada meningkat pada sektor transportasi dilihat dari beberapa tahun belakangan ini, serta konsumsi energi Indonesia yang cukup tinggi hampir 95% dari bahan bakar fosil. Dari total tersebut, hampir 50%nya merupakan Bahan Bakar Minyak (BBM) (Kementerian ESDM, 2012). Sedangkan sampai saat ini, energi fosil masih menjadi pasokan energi yang utama.

Kondisi ini mendorong manusia untuk berusaha mencari bahan bakar alternatif yang bersifat terbarukan untuk menghilangkan ketergantungan manusia terhadap minyak bumi. Hidrogen merupakan salah satu energi baru dan terbarukan yang mempunyai nilai kalor bawah sebesar 119.950 kJ/kg. Nilai energi tersebut jauh lebih besar dibandingkan dengan bahan bakar LPG, premium, pertamax, solar, CNG, dan etanol pada massa yang sama. Salah satu cara untuk mendapatkan gas hidrogen adalah dengan cara elektrolisa air (Reaktom, 2016). Yull Brown seorang warga negara Australia pada tahun 1974 telah mendapatkan paten dari hasil proses elektrolisa dari air menghasilkan gas H₂ dan O₂ yang diberi nama *Brown's Gas* yang dapat digunakan untuk

menggerakkan mesin kendaraan. Di Indonesia pada akhir-akhir ini sudah mulai dikembangkan yaitu dengan mencampurkan *Brown's Gas* yang didapat dari elektrolisa air dengan bahan bakar pada mesin bensin dengan karburator dan ternyata dapat meningkatkan tenaga, mengurangi pemakaian bahan bakar, serta memperbaiki kualitas emisi gas buang. Memanfaatkan *Brown's Gas* ini masih secara sederhana, belum terkendali baik, masih hanya untuk mesin bensin dengan karburator.

Brown's gas berpotensi sangat baik dalam mengurangi konsumsi bahan bakar baik untuk mesin bensin dan diesel, meningkatkan kinerja mesin, dan memperbaiki kualitas emisi gas buang mesin (Stanley Meyer, 1998). Pada penelitian yang dilakukan oleh Chadwell dan Dingle (2008) didapatkan bahwa pada mesin diesel, injeksi air dapat mengurangi emisi NOx sebesar 82% dan terjadi peningkatan pada torsinya. Injeksi H₂ pada mesin dapat menghilangkan detonasi dan mengurangi NOx lebih dari 50%; angka oktan naik lebih dari 50%; dan meningkatkan kerja mesin antara 30% - 50% (Lanzame. R, 1999).

Berdasarkan dari hasil penelitian yang telah diuraikan diatas menunjukkan peluang yang besar untuk membuat sebuah plant yang memproduksi *Brown's Gas*. Variable yang dikontrol dalam produksi *Brown's Gas* harus dijaga agar menghasilkan *Brown's Gas* yang ideal untuk masuk pada ruang bakar mesin. Variable-variable tersebut anatara lain temperature gas yang akan masuk pada ruang bakar mesin tidak boleh melebihi 60°C, level pada storage berupa campuran KOH dan Akuades, tekanan pada storage, dan kandungan oksigen gas buang, keempat variabel tersebut perlu dilakukan monitoring agar output sesuai dengan nilai yang dijaga. Metode yang digunakan dalam memproduksi *Brown's Gas* menggunakan metode elektrolisis yang akan menghasilkan *Brown's Gas* untuk meningkatkan efisiensi bahan bakar, kinerja mesin kendaraan, dan kualitas emisi gas buang mesin. Pada perancangan monitoring level tersebut terdapat lampu LED apabila level pada tanki berada posisi low. Oleh karena itu maka penulis mengajukan judul tentang "Rancang Bangun Sistem *Monitoring* Pada Produksi *Brown's Gas*

Menggunakan Generator HHO". Dengan hasil data monitoring yang diperoleh bisa digunakan untuk analisa karena data yang di dapatkan realtime dari hasil purifikasi.

1.2 Rumusan Permasalahan

Berdasarkan latar belakang yang dijelaskan diatas, maka rumusan masalah dalam Tugas Akhir ini adalah :

1. Bagaimana merancang sistem *monitoring* dalam proses produksi *Brown's Gas* dengan metode elektrolisis campuran akuades dan KOH?
2. Bagaimana proses sistem *monitoring* yang dapat menyimpan data, mengolah data dan menampilkan data sesuai dengan format yang diinginkan dalam proses produksi *Brown's Gas*?

1.3 Batasan Masalah

Perlu diberikan beberapa batasan permasalahan agar pembahasan tidak menyimpang dari tujuan. Adapun batasan permasalahannya yaitu adalah:

1. Fokus tugas akhir ini membahas tentang sistem monitoring temperature *Brown's Gas*, level pada storage, tekanan pada storage dan kandungan oksigen pada gas buang pada proses produksi *Brown's Gas*.
2. Sistem monitoring ini menggunakan Atmega32 untuk mengolah sinyal *output* sensor.
3. Bahan generator yang digunakan adalah *stainless Steel*.
4. *Engine* yang digunakan pengujian adalah motor *automatic transmission* 110 cc.
5. Bahan storage yang digunakan adalah PVC.
6. Bahan yang digunakan pengujian untuk produksi *Brown's Gas* adalah campuran KOH dan Aquades.

1.4 Tujuan

Tujuan utama dari rancang bangun alat ini adalah untuk memberikan solusi pada rumusan masalah yaitu:

1. Dapat merancang sistem monitoring dalam proses produksi *Brown's Gas* dengan metode elektrolisis campuran akuades dan KOH.
2. Dapat membuat proses sistem *monitoring* yang dapat menyimpan data, mengolah data dan menampilkan data sesuai dengan format yang diinginkan dalam proses produksi *Brown's Gas*.

1.5 Manfaat

Adapun manfaat yang dapat diperoleh dari tugas akhir ini yaitu:

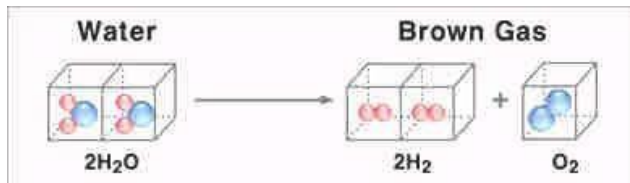
1. Tugas akhir ini dapat dijadikan untuk bekal bagi peserta untuk kedepannya dalam mengetahui energi alternatif bersifat terbarukan.
2. Tugas akhir ini dapat dijadikan sebagai ajang menambah pengetahuan bagi adik tingkat di program studi D3 Teknik Instrumentasi tentang energi alternatif.

BAB II DASAR TEORI

2.1 *Brown's Gas*

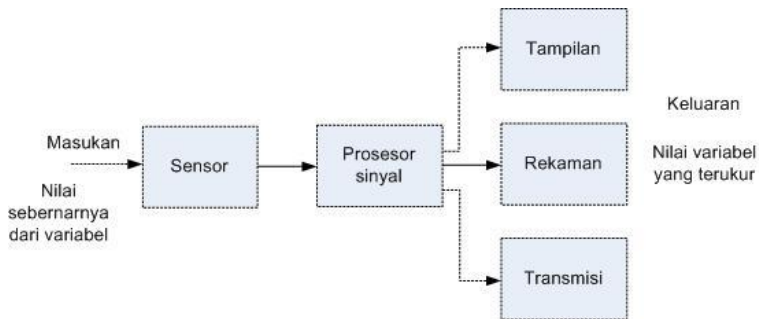
Gas HHO merupakan produk dari elektrolisa air murni (H_2O) dimana molekul air murni dipecah menjadi 2 molekul hidrogen (H_2) dan 1 molekul oksigen (O_2). Proses elektrolisis air dapat terjadi beberapa reaksi antara lain asam, basa maupun dengan setengah reaksi asam ataupun basa (*alkaline electrolysis*)[1].

Teknologi untuk memecah molekul air menjadi gas HHO dengan cara elektrolisis air sebenarnya telah ditemukan sejak tahun 1800 oleh William Nicholson dan Johann Ritter. Kemudian pada tahun 1805, Isaac de rivaz (1752-1828) menggunakan gas hidrogen dari hasil elektrolisis air sebagai bahan bakar mesin pembakaran internal yang ia rancang dan ia buat sendiri. Pada saat itu bahan bakar fosil belum ditemukan. Namun gas hasil dari elektrolisis air tersebut baru diberi nama dan dipatenkan oleh Dr. Yull Brown, pada tahun 1974. Gas hasil dari elektrolisis air tersebut diberi nama *Brown's Gas*. Selain menggunakannya sebagai suplemen bahan bakar pada mesin, Dr. Yull Brown juga menggunakan *Brown's Gas* untuk pengelasan (*cutting and welding torch*)[2].



Gambar 2.1 Pemecahan molekul air menjadi gas HHO[1]

2.2 Sistem Monitoring



Gambar 2.2 Diagram Block Sistem Pengukuran[3]

Monitoring merupakan pemantauan pada suatu proses atau sistem yang mencapai hasil yang diharapkan baik secara kualitas dan kuantitas dengan efektif. Monitoring dilakukan dengan tujuan untuk meminimalkan gangguan jika terdeteksi suatu gangguan atau kegagalan dalam suatu sistem. Monitoring selain berfungsi sebagai pengawasan juga berfungsi untuk merekam apa yang terjadi pada sistem yang dimonitor dalam bentuk data tabel maupun grafik yang ditampilkan dalam bentuk *display*.

Untuk mempermudah dalam melakukan pengukuran maka perlu diketahuimengenai pronsip dasar pengukuran seperti gambar 2.2 diatas. Sensor adalah elemen sistem yang secara efektif berhubungan dengan proses dimana suatu variabel sedang diukur dan menghasilkan suatu keluaran dalam bentuk tertentu tergantung pada variabel masukannya, dan dapat digunakan oleh bagian sistem pengukuran yang lain untuk mengenali nilai variabel tersebut. sebagai contoh adalah sensor termokopel yang memiliki masukan berupa temperatur serta keluaran berupa gaya gerak listrik (GGL) yang kecil. GGL yang kecil ini oleh bagian sistem pengukuran yang lain dapat diperkuat sehingga diperoleh pembacaan pada alat ukur.

Prosesor sinyal merupakan elemen sistem instrumentasi yang akan mengambil keluaran dari sensor dan mengubahnya menjadi suatu bentuk besaran yang cocok untuk tampilan dan transmisi selanjutnya dalam beberapa sistem kontrol. Seperti pengondisi sinyal (*signal conditioner*) merupakan salah satu bentuk *prosesor sinyal*.

Penampil data Elemen ini menampilkan nilai-nilai yang terukur dalam bentuk yang bisa dikenali oleh pengamat, seperti melalui sebuah alat penampil (*display*), misalnya sebuah jarum penunjuk (*pointer*) yang bergerak disepanjang skala suatu alat ukur. Selain ditampilkan, sinyal tersebut juga dapat direkam, misalnya pada kertas perekam diagram atau pada piringan magnetik, ataupun ditransmisikan ke beberapa sistem yang lain seperti sistem kontrol/kendali[3].

2.2.1 Karakteristik Statis

Merupakan karakter yang menggambarkan parameter instrument dalam keadaan *steady*. Adapun yang termasuk dalam bagian dari karakteristik static adalah sebagai berikut[4]:

- a. Akurasi (ketelitian)
Ketepatan alat ukur dalam memberikan hasil pengukuran.
- b. Presisi (*Repeatability*)
Kemampuan sistem pengukuran untuk menampilkan ulang output yang sama pada pengukuran berulang singkat.
- c. Toleransi
Menunjukkan kesalahan maksimum.
- d. Range (span)
Selisih nilai maksimum dan minimum yang dapat diukur oleh alat.
- e. Sensitivitas
Perubahan output instrument yang terjadi ketika kualitas pengukuran berubah.

f. Linieritas

Pengukuran yang baik adalah jika input pengukuran (nilai sesungguhnya) memberikan output (nilai yang ditunjukkan alat ukur) yang sebanding lurus. Sehingga bisa dikatakan bahwa linieritas adalah terjadinya penyimpangan garis linier[4].

Ketidakpastian Pengukuran

Suatu parameter yang berhubungan dengan hasil pengukuran yang mengkarakteristikan (memberikan sifat) penyebaran nilai-nilai layak dikaitkan dengan besaran ukur. Dalam perhitungan ketidakpastian pengukuran ada beberapa langkah yang harus dihitung antara lain[4]:

a. Koreksi

koreksi dapat diperoleh dengan persamaan berikut

koreksi = pembacaan standart – pembacaan alat

b. Standart Deviasi

$$\sigma = \frac{\sqrt{\sum(y_i - \bar{y})^2}}{n - 1}$$

Dimana:

Y_i = koreksi alat ukur

\bar{y} = rata-rata koreksi

N = banyak range pengukuran

c. Analisa Type A, (U_a)

Pada analisa tipe A ini hasilnya diperoleh dari data pengukuran. Adapun persamaannya adalah sebagai berikut :

$$U_{a1} = \frac{\sigma}{\sqrt{n}}$$

d. Ketidakpastian U_{a2}

Ketidakpastian U_{a2} dapat dihitung dengan persamaan :

$$U_{a2} = \sqrt{\frac{SSR}{n-2}} \text{ (Ketidakpastian regresi)}$$

e. Analisa type B, U_B

Analisa type B ini diperoleh berdasarkan sertifikat kalibrasi atau spesifikasi dari alat ukur. Adapun persamaannya adalah sebagai berikut :

U_{B1} (Ketidakpastian resolusi) :

$$\frac{\frac{1}{2} \times \text{Resolusi}}{\sqrt{3}}$$

U_{B2} (Ketidakpastian resolusi) :

$$\frac{a}{k}$$

f. Ketidakpastian Kombinasi, U_c

U_c merupakan ketidakpastian kombinasi dari ketidakpastian tipe A dan ketidakpastian tipe B. Adapun persamaan dari ketidakpastian kombinasi adalah :

$$U_c = \sqrt{U_{A1}^2 + U_{A2}^2 + U_{B1}^2 + U_{B2}^2}$$

Dimana :

U_c = Nilai ketidakpastian kombinasi

U_{A1} = Nilai ketidakpastian hasil pengukuran

U_{A2} = Nilai ketidakpastian regresi

U_{B1} = Nilai ketidakpastian resolusi

U_{B2} = Nilai ketidakpastian kalibrator

g. Ketidakpastian di perluas

U_c merupakan ketidakpastian kombinasi dari ketidakpastian tipe A dan ketidakpastian tipe B. Adapun persamaan dari ketidakpastian kombinasi adalah :

$$U_c = \sqrt{U_{A1}^2 + U_{A2}^2 + U_{B1}^2 + U_{B2}^2}$$

Dimana :

U_c = Nilai ketidakpastian kombinasi

U_{A1} = Nilai ketidakpastian hasil pengukuran

U_{A2} = Nilai ketidakpastian regresi

U_{B1} = Nilai ketidakpastian resolusi

U_{B2} = Nilai ketidakpastian kalibrator

Hasil akhir kalibrasi adalah ketidakpastian diperluas sehingga alat ukur tersebut dapat diketahui ketidakpastiannya melalui *Uexpand*. Persamaan *Uexpand* adalah:

$$U_{expand} = K \cdot U_c$$

Dimana :

k = Faktor cakupan

U_c = Nilai ketidakpastian kombinasi

Untuk mencari nilai k, maka melihat table t student sesuai dengan *confidence level 95%*. Tabel T *student* dapat dilihat pada gambar berikut :

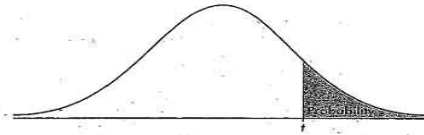


TABLE B: F-DISTRIBUTION CRITICAL VALUES

df	Tail probability p											
	.25	.20	.15	.10	.05	.025	.02	.01	.005	.0025	.001	.0005
1	1.000	1.376	1.963	3.078	6.314	12.71	15.89	31.82	63.66	127.3	318.3	636.6
2	.816	1.061	1.386	1.886	2.920	4.303	4.849	6.965	9.925	14.09	22.33	31.60
3	.765	.978	1.250	1.638	2.353	3.182	3.482	4.541	5.841	7.453	10.21	12.92
4	.741	.941	1.190	1.533	2.132	2.776	2.999	3.747	4.604	5.598	7.173	8.610
5	.727	.920	1.156	1.476	2.015	2.571	2.757	3.365	4.032	4.773	5.893	6.869
6	.718	.906	1.134	1.440	1.943	2.447	2.612	3.143	3.707	4.317	5.208	5.959
7	.711	.896	1.119	1.415	1.895	2.365	2.517	2.998	3.499	4.029	4.785	5.408
8	.706	.889	1.108	1.397	1.860	2.306	2.449	2.896	3.355	3.833	4.501	5.041
9	.703	.883	1.100	1.383	1.833	2.262	2.398	2.821	3.250	3.690	4.297	4.781
10	.700	.879	1.093	1.372	1.812	2.228	2.359	2.764	3.169	3.581	4.144	4.587
11	.697	.876	1.088	1.363	1.796	2.201	2.328	2.718	3.106	3.497	4.025	4.437
12	.695	.873	1.083	1.356	1.782	2.179	2.303	2.681	3.055	3.428	3.930	4.318
13	.694	.870	1.079	1.350	1.771	2.160	2.282	2.650	3.012	3.372	3.852	4.221
14	.692	.868	1.076	1.345	1.761	2.145	2.264	2.624	2.977	3.326	3.787	4.140
15	.691	.866	1.074	1.341	1.753	2.131	2.249	2.602	2.947	3.286	3.733	4.073
16	.690	.865	1.071	1.337	1.746	2.120	2.235	2.583	2.921	3.252	3.686	4.015
17	.689	.863	1.069	1.333	1.740	2.110	2.224	2.567	2.898	3.222	3.646	3.965
18	.688	.862	1.067	1.330	1.734	2.101	2.214	2.552	2.878	3.197	3.611	3.922
19	.688	.861	1.066	1.328	1.729	2.093	2.205	2.539	2.861	3.174	3.579	3.883
20	.687	.860	1.064	1.325	1.725	2.086	2.197	2.528	2.845	3.153	3.552	3.850
21	.686	.859	1.063	1.323	1.721	2.080	2.189	2.518	2.831	3.135	3.527	3.819
22	.686	.858	1.061	1.321	1.717	2.074	2.183	2.508	2.819	3.119	3.505	3.792
23	.685	.858	1.060	1.319	1.714	2.069	2.177	2.500	2.807	3.104	3.485	3.768
24	.685	.857	1.059	1.318	1.711	2.064	2.172	2.492	2.797	3.091	3.467	3.745
25	.684	.856	1.058	1.316	1.708	2.060	2.167	2.485	2.787	3.078	3.450	3.725
26	.684	.856	1.058	1.315	1.706	2.056	2.162	2.479	2.779	3.067	3.435	3.707
27	.684	.855	1.057	1.314	1.703	2.052	2.158	2.473	2.771	3.057	3.421	3.690
28	.683	.855	1.056	1.313	1.701	2.048	2.154	2.467	2.763	3.047	3.408	3.674
29	.683	.854	1.055	1.311	1.699	2.045	2.150	2.462	2.756	3.038	3.396	3.659
30	.683	.854	1.055	1.310	1.697	2.042	2.147	2.457	2.750	3.030	3.385	3.646
40	.681	.851	1.050	1.303	1.684	2.021	2.123	2.423	2.704	2.971	3.307	3.551
50	.679	.849	1.047	1.299	1.676	2.009	2.109	2.403	2.678	2.937	3.261	3.496
60	.679	.848	1.045	1.296	1.671	2.000	2.099	2.390	2.660	2.915	3.232	3.460
80	.678	.846	1.043	1.292	1.664	1.990	2.088	2.374	2.639	2.887	3.195	3.416
100	.677	.845	1.042	1.290	1.660	1.984	2.081	2.364	2.626	2.871	3.174	3.390
∞	.675	.842	1.037	1.282	1.646	1.962	2.056	2.330	2.581	2.813	3.098	3.300
∞	.674	.841	1.036	1.282	1.645	1.960	2.054	2.326	2.576	2.807	3.091	3.291
	50%	60%	70%	80%	90%	95%	96%	98%	99%	99.5%	99.8%	99.9%
	Confidence level C											

Gambar 2.3 Tabel T student[4]

h. **V efektif**

$$V_{eff} = \frac{(U_c)^4}{\sum (U_i)^4 / V_i}$$

Dimana

V_{eff} : Derajat kebebasan efektif dari ketidakpastian kombinasi

V_i : Derajat kebebasan dari komponen ketidakpastian ke- i

U_i : Hasil ketidakpastian tipe A dan B[4]

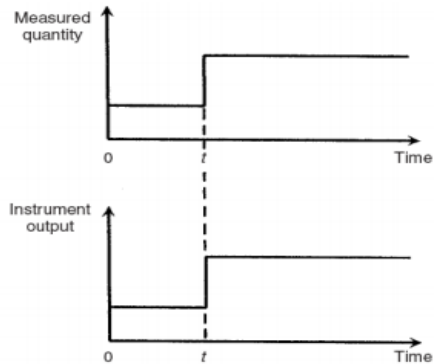
2.2.2 Karakteristik dinamik

Karakteristik dinamik dari sebuah alat ukur menggambarkan perilakunya antara waktu yang terukur dengan perubahan nilai dan waktu ketika sebuah alat output mencapai nilai stabil. Nilai karakteristik dinamik dikutip dalam lembaran instrumen data hanya berlaku pada saat instrumen yang digunakan dalam kondisi lingkungan tertentu[4].

Karakteristik dinamik dikelompokkan menjadi tiga orde diantaranya:

1. Instrument orde nol

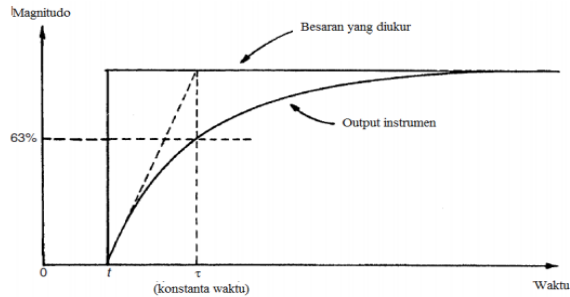
Pada Instrument orde nol, ketika ada perubahan input pengukuran, output akan bergerak cepat menuju nilai baru sehingga mendekati respon *step*. Berikut ini merupakan respon output orde nol :



Gambar 2.4 Respon orde nol[4]

2. Instrument orde satu

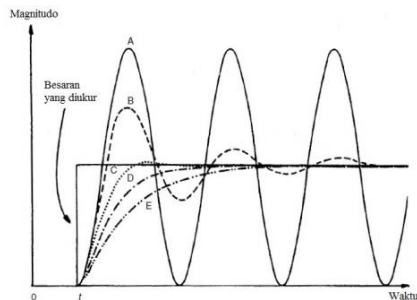
Pada instrument orde satu, saat ada perubahan step input pengukuran, output instrumen berubah secara gradual (tidak secara tiba-tiba seperti instrumen orde nol) dan membutuhkan waktu untuk mencapai kondisi yang sama dengan nilai besaran yang diukur. Pada orde ini nilainya dipengaruhi oleh karakteristik statik instrumen.



Gambar 2.5 Respon Orde Satu

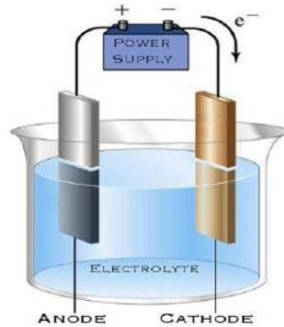
3. Instrument orde dua

Pada instrument orde dua ini dipengaruhi oleh beberapa faktor diantaranya rasio redaman, sensitivitas statik, dan frekuensi natural tak teredam. Redaman sangat mempengaruhi respon terhadap perubahan step input. Bentuk respon step besaran output yang diperoleh bergantung pada nilai parameter rasio redaman.



Gambar 2.6 Respon Orde Dua[4]

2.3 Elektrolisis



Gambar 2.7 Rangkaian dasar sistem elektrolit[2]

Elektrolisis adalah suatu proses untuk memisahkan senyawa kimia menjadi unsur-unsurnya atau memproduksi suatu molekul baru dengan memberi arus listrik. Sedangkan elektrolisis air adalah proses elektrolisa yang dimanfaatkan untuk memecah molekul air (H_2O) menjadi Hidrogen (H_2) dan Oksigen (O_2). Elektrolisis air pada dasarnya dilakukan dengan mengalirkan arus listrik ke air melalui dua buah elektroda (Katoda dan Anoda). Agar proses elektrolisa dapat terjadi dengan cepat maka air tersebut dicampur dengan elektrolit sebagai katalis.

Proses elektrolisis air dapat terjadi dengan setengah reaksi asam ataupun basa (*alkaline electrolysis*) ataupun keduanya. Terjadinya reaksi asam ataupun basa tergantung oleh kondisi lingkungan/jenis elektrolit yang digunakan (Robert B. Dopp, 2007). Jika elektrolit yang digunakan berupa larutan asam seperti HCl dan H_2SO_4 maka reaksi yang terjadi adalah reaksi asam. Pada reaksi ini reaksi reduksi terjadi pada elektroda negatif (katoda), dimana elektron (e^-) dari katoda diikat oleh kation H^+ untuk membentuk gas Hidrogen ($H_{2(g)}$). Sedangkan pada elektroda positif (anoda), molekul H_2O kehilangan elektron (e^-) sehingga

terpecah menjadi gas Oksigen ($O_{2(g)}$) dan kation H^+ . Untuk rangkaian dasar proses elektrolisis dapat dilihat pada gambar 2.7.

- Reaksi oksidasi di anoda (+): $2 H_2O_{(l)} \rightarrow O_{2(g)} + 4 H^+_{(aq)} + 4 e^-$
- Reaksi reduksi di katoda (-): $2 H^+_{(aq)} + 2 e^- \rightarrow H_{2(g)}$
- Reaksi keseluruhan : $2 H_2O_{(l)} \rightarrow 2 H_{2(g)} + O_{2(g)}$

Jika elektrolit yang digunakan adalah larutan basa seperti KOH, NaOH (basa dari golongan periode IA, alkali tanah) maka akan terjadi reaksi basa. Pada reaksi basa, reaksi reduksi terjadi di katoda dimana molekul air mengikat elektron (e^-) sehingga terpecah menjadi gas Hidrogen ($H_{2(g)}$) dan anion OH^- . Anion OH^- tersebut kemudian tertarik kesisi anoda dan terpecah menjadi gas oksigen dan molekul $H_2O_{(l)}$, sebagaimana dapat dilihat pada persamaan reaksi kimia berikut:

- Reaksi reduksi di katoda (-): $2 H_2O_{(l)} + 2 e^- \rightarrow H_{2(g)} + 2 OH^-_{(aq)}$
- Reaksi oksidasi di anoda (+): $4 OH^-_{(aq)} \rightarrow O_{2(g)} + 2 H_2O_{(l)} + 4 e^-$
- Reaksi keseluruhan: $2 H_2O_{(l)} \rightarrow 2 H_{2(g)} + O_{2(g)}$

Akan tetapi jika elektrolit yang digunakan dari jenis garam seperti NaCl, KCl, dan Na_2CO_3 , maka akan terjadi reaksi asam dan basa. Dari kedua reaksi asam ataupun basa dapat dilihat bahwa pada kedua reaksi tersebut produk yang dihasilkan dari elektrolisa 2 mol H_2O memiliki komposisi yang sama yaitu 2 mol gas Hidrogen dan 1 mol gas Oksigen. Pada kedua jenis reaksi diatas gas Hidrogen juga dihasilkan pada elektroda negatif (katoda) dan gas oksigen dihasilkan pada elektroda positif (anoda)[2].

Adapun Faktor yang mempengaruhi elektrolisis antara lain adalah:

2.3.1 Penggunaan Katalisator

Misalnya H_2SO_4 dan KOH berfungsi mempermudah proses penguraian air menjadi hidrogen dan oksigen karena ion-ion katalisator mampu mempengaruhi kesetabilan molekul air menjadi menjadi ion H^+ dan OH^- yang lebih mudah di elektrolisis karena terjadi penurunan energi pengaktifan. Zat tersebut tidak mengalami perubahan yang kekal (tidak dikonsumsi dalam proses elektrolisis). Penggunaan asam sulfat sebagai katalis dalam proses elektrolisis menjadi pilihan utama dibandingkan KOH . Karena asam sulfat melepaskan H^+ yang memudahkan membentuk gas hidrogen. Sedangkan KOH melepaskan OH^- yang menghambat pembentukan gas hidrogen.

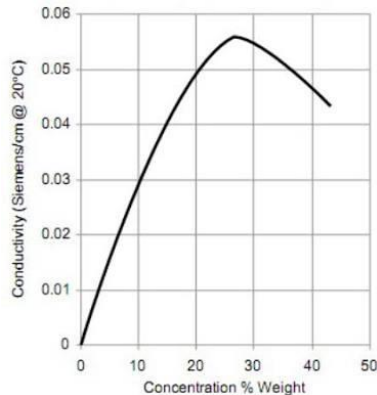
2.3.2 Luas Permukaan Tercelup

Semakin banyak luas yang semakin banyak menyentuh elektrolit maka semakin mempermudah suatu elektrolit untuk mentransfer elektronnya. Sehingga terjadi hubungan sebanding jika luasan yang tercelup sedikit maka semakin mempersulit elektrolit untuk melepaskan elektron dikarenakan sedikitnya luas penampang penghantar yang menyentuh elektrolit. Sehingga transfer elektron bekerja lambat dalam mengelektrolisis elektrolit. Pada tugas akhir ini luasan yang terelektrolisis sekitar 60% dan cukup dibatasi dengan o-ring atau seal yang berdiameter 70 mm pada setiap plat yang digunakan. Selain itu pada setiap plat terdapat dua lubang berdiameter 12 mm untuk saluran gas HHO yang berada di bagian atas dan di bawah.

2.3.3 Konsentrasi Pereaksi

Semakin besar konsentrasi suatu larutan pereaksi maka akan semakin besar pula laju reaksinya. Hal ini dikarenakan dengan prosentase katalis yang semakin tinggi dapat mereduksi hambatan pada elektrolit. Sehingga transfer electron dapat lebih cepat

mengelektrolisis elektrolit dan dapat ditarik garis lurus bahwa terjadi hubungan sebanding terhadap prosentase katalis dengan transfer electron. Penggunaan katalis untuk paling optimum adalah 27%. Hal tersebut dibuktikan dengan gambar 2.8.



Gambar 2.8 Hubungan Konsentrasi KOH dan Konduktivitas[2]

2.3.4 Bahan Stainless Steel sebagai Elektroda

Elektroda merupakan salah satu komponen yang sangat penting pada proses elektrolisis air. Elektroda berfungsi sebagai penghantar arus listrik dari sumber tegangan ke air yang akan dielektrolisis. Material serta luasan elektroda yang digunakan sangat berpengaruh terhadap gas HHO yang dihasilkan dari proses elektrolisis air. Stainless Steel merupakan logam paduan yang memiliki konduktivitas dan ketahanan terhadap korosi yang relatif lebih baik di banding logam-logam paduan ataupun logam murni lainnya dan harganya juga relatif lebih terjangkau. Sehingga Stainless steel menjadi pilihan yang tepat untuk digunakan sebagai elektroda pada proses elektrolisis.

Compositions of AISI-SAE Standard Stainless Steels									
AISI Type (UNS)	Typical Composition (%)								
	Cr	Ni	C	Mn	Si	P	S	Mo	N
316 (S31600) Austenitic	16–18	10–14	0.08	2.0	0.75	0.045	0.030	2.0–3.0	0.10
302 (S30200) Austenitic	17–19	8–10	0.15	2.0	0.75	0.045	0.030	---	0.10
316L (S31603) Austenitic	16–18	10–14	0.03	2.0	0.75	0.045	0.030	2.0–3.0	0.10
303 (S30300) Austenitic	17–19	8–10	0.15	2.0	1.0	0.20	0.015 min	0.60 optional	---
304 (S30400) Austenitic	18–20	8–10.50	0.08	2.0	0.75	0.045	0.030	---	0.10
304L (S30403) Austenitic	18–20	8–12	0.03	2.0	0.75	0.045	0.030	---	0.10
321 (S32100) Austenitic	17–19	9–12	0.08	2.0	0.75	0.045	0.030 [Ti, SiC + Ni max, 0.20 max]	---	0.10 max
430 (S43000) Ferritic	16–18	0.75 Ni	0.12	1.0	1.0	0.040	030	---	---
416 (S41600) Martensitic	12–14		0.15	1.25	1.0	0.060	0.15min	0.060 optional	---

Gambar 2.9 Standard komposisi stainless steel (David Tabolt, 1998)

2.4 Generator gas HHO

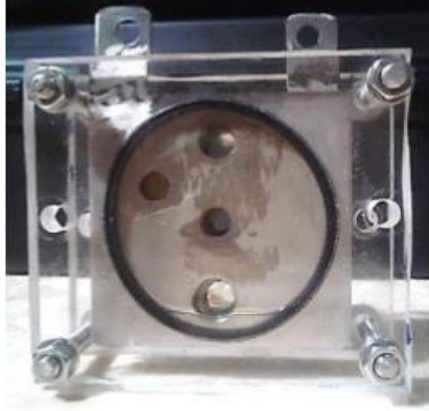
Generator gas HHO tersusun atas 2 komponen dasar, yaitu tabung generator yang terdiri atas tabung, sepasang elektroda dan elektrolit. Dan sumber tenaganya yang berupa baterai ataupun aki. Generator ini bekerja dengan prinsip elektrolisa air. Generator gas HHO diklasifikasikan menjadi dua tipe, yakni sebagai berikut:

1. Generator HHO Tipe dry cell

Generator HHO dimana sebagian elektrodanya tidak terendam elektrolit. Keuntungan generator HHO tipe dry cell adalah:

- a. Penggunaan air untuk proses elektrolisa hanya sedikit, yaitu hanya air yang terjebak diantara lempengan cell.
- b. Ada sirkulasi air dengan tambahan reservoir, dimana cukup untuk menurunkan temperature kerja dari generator itu sendiri.

- c. Konstruksinya yang simpel, tidak memerlukan space yang banyak[5].



Gambar 2.10 Generator HHO Tipe Dry Cell[5]

2. Generator HHO Tipe wet cell

Generator HHO Tipe wet cell Generator HHO dimana semua elektrodanya terendam cairan elektrolit di dalam sebuah bejana air. Keuntungan generator gas HHO tipe wet cell adalah:

- a. Produksi yang dihasilkan lebih banyak dikarenakan luasan elektroda yang sepenuhnya terendam larutan elektrolit.
- b. Perawatan generator yang lebih ringkas.
- c. Pembuatan generator tipe wet cell lebih mudah dan cepat[5].



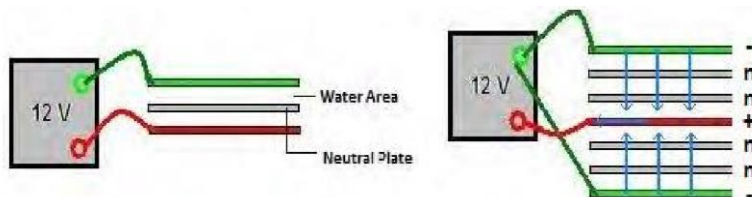
Gambar 2.11 Generator HHO Tipe wet cell[5]

2.5 Netral Plat Generator HHO

Netral plat adalah suatu plat elektroda pada generator gas HHO yang tidak dialiri oleh listrik, sehingga elektroda ini bukan merupakan suatu kutub negatif maupun positif, oleh karena itu disebut sebagai elektroda netral. Netral plat menyebabkan tegangan drop antar plat, dimana netral plat memberikan luas permukaan tambahan untuk produksi gas HHO serta menurunkan panas yang diakibatkan dari proses elektrolisa air.

Netral plat berada di antara elektroda positif dan negatif dan tidak dialiri oleh arus listrik. Ketika netral plat ditambahkan, maka water area (kolom air diantara plat) juga akan bertambah sesuai dengan jumlah netral plat yang ditambahkan. Ketika netral plat ditambahkan di antara elektroda positif maupun negatif maka besar arus listrik yang melalui netral plat juga sama dengan arus yang melewati elektroda positif dan negatif. Jika diukur besar tegangan antara satu cell kutub positif dan negatif ialah 12volt namun bila diukur besar tegangan pada salah satu kutub elektroda dengan netral plate maka terjadi penurunan tegangan sesuai dengan penambahan netral platnya. Disebabkan penambahan plat netral menaikkan nilai resistansi atau hambatan pada arus listrik

yang bekerja pada elektroda, sehingga penambahan netral plate dapat menurunkan tegangan listrik pada plat elektroda. Semakin rendah tegangannya, maka semakin rendah pula panas yang dihasilkan, karena tegangan yang rendah dengan besar arus yang sama, serta luas permukaan yang bertambah dikarenakan penambahan netral plat, maka laju produksi gas HHO pada generator juga akan semakin meningkat[5].



Gambar 2.12 Netral Plat[5]

2.6 Sensor *Level Float*

Sensor *level float* merupakan switch yang akan memberikan informasi atau tanda bahwa *level* dari suatu liquid (*liquid level*) atau *powder level* sudah mencapai pada setting yang ditentukan. *Level switch* ini bisa bekerja pada media liquid normal seperti untuk water level, untuk oil seperti fuel level atau *oil level* atau *solar level* Namun ada juga yang bekerja untuk matrial *not liquid* seperti *powder*, biji-bijian tau lumpur. Jenis dari *liquid* yang mungkin bersifat korosif atau yang lainnya akan menentukan jenis matrial untuk floating switchnya. Jenis *Level switch* yang paling banyak digunakan adalah jenis *float level switch* baik untuk oil maupun air. Untuk jenis *float level switch* solar menggunakan matrial float dari stainless steel sedangkan untuk *float level switch* yang diaplikasikan di air banyak menggunakan jenis non logam[6].

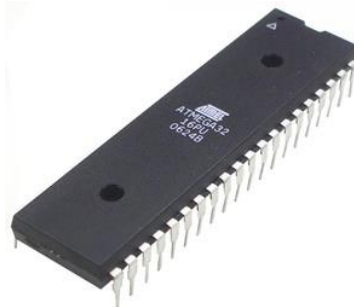


Gambar 2.13 Bentuk sensor Level Float[6]

2.7 Mikrokontroler Atmega32

Mikrokontroler adalah IC yang dapat diprogram berulang kali, baik ditulis atau dihapus. Biasanya digunakan untuk pengontrolan otomatis dan manual pada perangkat elektronika.

ATmega32 memiliki 32 *general purpose register*, dan *register* terhubung langsung dengan ALU (*Arithmetic Logic Unit*) sehingga dengan dua *register* dapat sekaligus diakses dalam satu intruksi yang dieksekusi tiap *clock*-nya. Sehingga arsitektur seperti ini lebih efisien dalam eksekusi kode program dan dapat mencapai eksekusi sepuluh kali lebih cepat dibandingkan mikrokontroler CISC(*Complete Instruction Set Computer*)[7].



Gambar 2.14 IC Chip Atmega32[7]

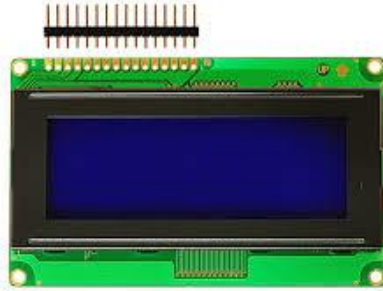
2.8 Software Code Vision AVR

Code Vision AVR merupakan sebuah cross-compiler C, Integrated Development Environment (IDE), dan Automatic Program Generator yang didesain untuk mikrokontroler buatan Atmel seri AVR. Code Vision AVR dapat dijalankan pada sistem operasi Windows 95, 98, Me, NT4, 2000, dan XP. Crosscompiler C mampu menerjemahkan hampir semua perintah dari bahasa ANSI C, sejauh yang diizinkan oleh arsitektur dari AVR, dengan tambahan beberapa fitur untuk mengambil kelebihan khusus dari arsitektur AVR dan kebutuhan pada sistem embedded. File object COFF hasil kompilasi dapat digunakan untuk keperluan debugging pada tingkatan C, dengan pengamatan variabel, menggunakan debugger Atmel AVR Studio. IDE mempunyai fasilitas internal berupa software AVR Chip InSystem Programmer yang memungkinkan untuk melakukan transfer program kedalam chip mikrokontroler setelah sukses melakukan kompilasi/assembly secara otomatis[8].

2.9 LCD 20x4

LCD (Liquid Crystal Display) ini merupakan suatu *device* untuk menampilkan data berupa tampilan (*display*). LCD adalah lapisan dari campuran organik antara lapisan kaca bening dengan elektroda transparan indium oksida dalam bentuk tampilan seven-segment dan lapisan elektroda pada kaca belakang. Ketika elektroda diaktifkan dengan medan listrik (tegangan), molekul organik yang panjang dan silindris menyesuaikan diri dengan elektroda dari segmen. LCD yang ada di pasaran terdiri dari berbagai ukuran, seperti 16x2, 16x4, 20x4 dan lain-lain. LCD (*Liquid Cristal Display*) berfungsi sebagai penampil data baik dalam bentuk karakter, huruf, angka ataupun grafik.

Gambar 2.16 tersebut merupakan contoh dari LCD dengan ukuran 20 x 4, yang artinya terdiri dari 20 baris dan 4 kolom. Masing-masing baris memiliki alamat memori tersendiri. LCD membutuhkan tegangan sebagai *power* yaitu sebesar 0 – 5 Volt DC. Tegangan tersebut masuk ke dalam pin 2 (VSS) sebagai GND dan pin 2 (VDD sebagai VCC. Ada pula pin 3 (VEE) untuk mengatur tingkat keterangan cahaya *backlight* yang biasa dilakukan dengan pengaturan resistansi menggunakan potensiometer.



Gambar 2.15 LCD 20x4

Keterangan:

1. Pin 1 dihubungkan ke Ground.
2. Pin 2 dihubungkan ke Vcc (+5V).
3. Pin 3 dihubungkan ke bagian tengah potensiometer sebagai pengatur kontras.
4. Pin 4 untuk Register Selection (RS). Jika diberi nilai logika 1 (High) = display data dan jika diberi nilai logika 0 (Low) = Write Operational.
5. Pin 5 digunakan untuk mengatur fungsi LCD. Jika diset ke logika 1 (high, +5V) maka LCD berfungsi untuk membaca data, jika pin ini di-set ke logika 0 (low, 0V) akan berfungsi untuk menulis data.

6. Pin 6 adalah terminal enable (Enable Signal). Berlogika 1 setiap kali pengiriman atau pembaca data.
7. Pin 7 – 14 adalah saluran dua arah (bi-directional) data 8 bit dan 4 bit bus data (untuk 4 bit pin data yang digunakan pin 11 – 14).
8. Pin 15 dan 16 adalah tegangan untuk menyalakan LCD[9].

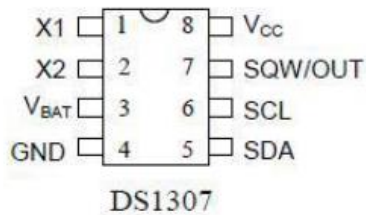
2.10 RTC

Istilah RTC (*real time clock*) merupakan sebuah istilah untuk jam elektronik dalam bentuk sebuah IC (*integrated circuit*) yang memiliki fungsi sebagai penyimpan data waktu dan tanggal. Data waktu dan tanggal tersebut akan disertakan dengan data guna menambah validitas proses pengambilan data. Jadi, akan diketahui detik, menit, jam, hari, bulan, dan tahun saat melakukan proses pengambilan data.

Salah satu jenis real time clock adalah DS1307 yang dapat menyimpan data- data berupa detik, menit, jam, tanggal, bulan, hari dalam seminggu, dan tahun yang valid hingga tahun 2100. IC DS1307 ini mempunyai NV SRAM sebesar 56-byte, General Purpose RAM tanpa unlimited writes, antarmuka I2C, dan battery- backed sebagai sumber daya cadangan. Susunan pin DS1307 dapat dilihat pada Gambar 2.18. Bentuk komunikasi data dari IC RTC adalah I2C yang merupakan kepanjangan dari *Inter Integrated Circuit*. Komunikasi jenis ini hanya menggunakan 2 jalur komunikasi yaitu SCL dan SDA. Semua microcontroller sudah dilengkapi dengan fitur komunikasi 2 jalur ini, termasuk diantaranya Arduino Microcontroller[10].



Gambar 2.16 RTC DS1307



Gambar 2.17 Konfigurasi PIN RTC DS1307

Fungsi pin dari komponen RTC S1307 adalah sebagai berikut:

1. Pin V_{CC} (Nomor 8) berfungsi sebagai sumber energi listrik Utama. Tegangan kerja dari komponen ini adalah 5 volt, dan ini sesuai dengan tegangan kerja dari microcontroller.
2. Pin GND (Nomor 4) Anda harus menghubungkan ground yang dimiliki oleh komponen RTC.
3. SCL berfungsi sebagai saluran *clock* untuk komunikasi data antara Microcontroller dengan RTC.
4. SDA berfungsi sebagai saluran Data untuk komunikasi data antara Microcontroller dengan RTC.
5. X1 dan X2 berfungsi untuk saluran *clock* yang bersumber dari *crystal external*.
6. Vbat Berfungsi sebagai saluran energy listrik dari Battery external[10].

2.11 *Openlog Open Source Datalogger*

Openlog open source datalogger merupakan modul penyimpanan datalogger berbasis SD/MMC. Telah dilengkapi dengan soket SD/MMC jenis FAT16 dan FAT32 dan mampu menyimpan data sampai dengan 16 Giga Bytes. Modul ini berbasis mikrokontroler ATmega328. Proses penyimpanan data cukup mudah, hanya dengan mengirimkan perintah melalui antarmuka UART, maka sudah dapat melakukan create file, append file, write file, read file, create dir, dll. Data yang dikirim mikrokontroler secara serial akan dituliskan ke dalam file dengan format .TXT. File .TXT tersebut dapat dibuka dan dibaca melalui program NOTEPAD.

Adapun spesifikasi atau fitur dari OpenLog Open Source Datalogger adalah sebagai berikut:

- Catu daya: 3.3VDC - 12VDC.
- Berbasis mikrokontroler ATmega328. □ Antarmuka UART.
- Media penyimpanan berupa microSD (up to 16GByte).
- Bersifat open source.
- Dua LED mengindikasikan menulis statusnya.
- Dapat melakukan edit Fileconfig.txt dari komputer untuk mengubah *baudrate* dan pengaturan sistem lainnya[11].



Gambar 2.18 Openlog data logger[11]

2.12 Komunikasi data

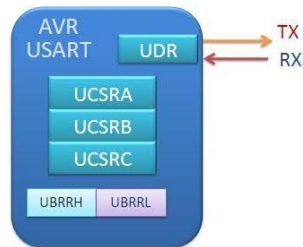
Komunikasi data adalah proses pengiriman dan penerimaan data/informasi dari dua atau lebih device (alat, seperti komputer/laptop/printer/dan alat komunikasi lain) yang terhubung dalam sebuah jaringan. Salah satu macam-macam dari komunikasi data adalah USART, SPI dan I2C.

2.12.1 USART

USART (*Universal Synchronous and Asynchronous Serial Receiver and Transmitter*) adalah protokol komunikasi serial antara PC dengan komponen elektronika lainnya, salah satunya adalah mikrokontroler. USART juga digunakan sebagai alat komunikasi antara satu mikrokontroler dengan mikrokontroler lain maupun dengan port expander. Hal lain yang perlu diperhatikan adalah setting baudrate dan clock yang digunakan mikrokontroler tersebut. USART pada mikrokontroler ATmega32 memiliki beberapa keuntungan diantaranya ialah operasi full duplex (memiliki register receiver dan transmitter yang terpisah), mendukung komunikasi multiprosesor dan kecepatan transmisi berorde Mbps.

- **USART I/O Data Register**

(UDR) digunakan sebagai penyangga (*buffer*) data yang akan dikirimkan dan penyangga data yang diterima. Transmitter data *buffer register* (TXB) berfungsi sebagai penyangga data yang akan ditulis ke dalam register UDR. Sedangkan *receiver data buffer register* (RXB) berfungsi sebagai penyangga data yang diterima oleh register UDR. Register UDR untuk penyangga *transmitter* hanya dapat ditulis ketika bit/*flag* UDRE dalam register UCSRA dalam keadaan set. Gambar 2.5 menunjukkan register USART *transmitter* dan *receiver* pada UDR sebagai berikut:



Gambar 2.19 AVR USART TX dan RX

- **Control and status register (UCSRA, UCSRB, dan UCSRC)**

Penggunaan USART untuk komunikasi antar mikrokontroler adalah dengan menghubungkan pin RX dan TX lawan pin TX pada RW lawan. Begitu pula jika mikrokontroler berkomunikasi dengan PC. Akan tetapi, karena level tegangan pada PC berbeda dengan mikrokontroler maka dibutuhkan suatu rangkaian untuk menyesuaikan. Biasanya digunakan konektor RS232. Dengan dimanfaatkannya sistem USART kita bisa menghubungkan mikrokontroler dengan PC, HP, GPS, atau bahkan modem dan banyak lagi yang dapat dihubungkan dengan mikrokontroler dengan menggunakan fasilitas USART[12].

2.12.2 SPI (*serial peripheral interface*)

SPI (*serial peripheral interface*) merupakan salah satu metode pengiriman data dari suatu devais ke devais lainnya. Metode ini merupakan metode yang bekerja pada metode *full duplex* dan merupakan standar sinkronasi serial *data link* yang dikembangkan oleh Motorola). Cara kerja protokol SPI ditunjukkan pada gambar 2.20. Pada SPI, device dibagi menjadi dua bagian yaitu master dan slave dengan master sebagai device yang menginisiasi pengiriman data. Dalam

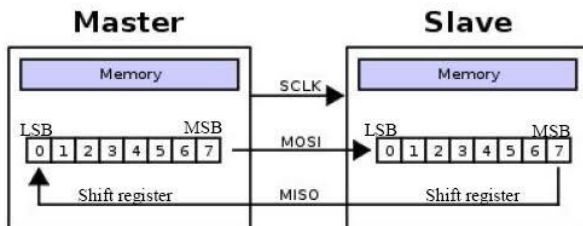
aplikasinya, sebuah master dapat digunakan untuk mengatur pengiriman data ke beberapa slave (*Multipoint*).

Komunikasi serial data antara master dan slave pada SPI diatur melalui 4 buah pin yang terdiri dari SCLK, MOSI, MISO, dan SS. Berikut ini adalah penjelasan singkat mengenai ke 4 pin tersebut: SCLK (*serial clock*) merupakan data biner yang keluar dari master ke slave yang berfungsi sebagai clock dengan frekuensi tertentu. *Clock* merupakan salah satu komponen prosedur komunikasi data SPI. Dalam beberapa devais, istilah yang digunakan untuk pin ini adalah SCK.

MOSI (*master out slave input*) merupakan pin yang berfungsi sebagai jalur data pada saat data keluar dari master dan masuk ke dalam slave. Istilah lain untuk pin ini antara lain SIMO, SDI, DI, dan SI.

MISO (*master input slave output*) merupakan pin yang berfungsi sebagai jalur data yang keluar dari slave dan masuk ke dalam master. Istilah lain untuk pin ini adalah SOMI, SDO, DO, dan SO.

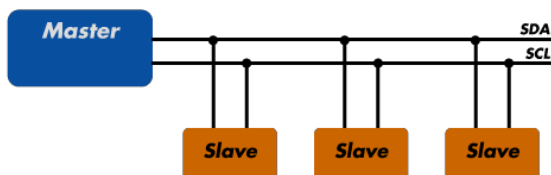
SS (*slave select*) merupakan pin yang berfungsi untuk mengaktifkan slave sehingga pengiriman data hanya dapat dilakukan jika slave dalam keadaan aktif (*active low*). Istilah lain untuk SS antara lain CS (*chip select*), CS, SS, dan STE (*slave transmit enable*).



Gambar 2.20 Cara kerja protocol SPI[12]

2.12.3 I2C (*Inter Integrated Circuit*)

Inter Integrated Circuit atau sering disebut I2C adalah standar komunikasi serial dua arah menggunakan dua saluran yang didesain khusus untuk mengirim maupun menerima data. Sistem I2C terdiri dari saluran SCL (*Serial Clock*) dan SDA (*Serial Data*) yang membawa informasi data antara I2C dengan pengontrolnya.. Pada komunikasi I2C terdapat perangkat master dan slave. Master adalah perangkat yang mengatur jalur clock SCL. Sedangkan slave adalah perangkat yang merespon perintah master. Slave tidak dapat mengirim sinyal untuk dapat mentransfer data pada jalur I2C, hanya master yang dapat melakukannya.



Gambar 2.21 Konfigurasi sistem I2C[12]

2.12.4 Perbedaan antara USART, SPI dan I2C

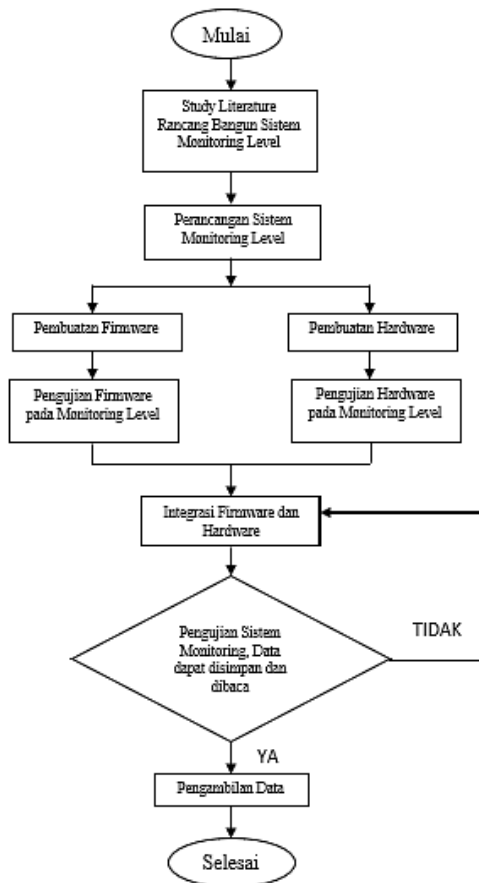
Adapun perbedaan antara USART, SPI dan I2C adalah pin yang digunakan pada ketiga komunikasi data tersebut. USART adalah protokol komunikasi data antara satu device ke device lainnya. Dengan pin TXD dan RXD. Sedangkan SPI adalah protokol data serial yang digunakan oleh mikrokontroler untuk berkomunikasi dengan satu atau lebih perangkat periferal cepat jarak pendek. Dengan pin SCLK, MOSI, MISO, dan SS. Dan untuk I2C adalah standar komunikasi serial dua arah menggunakan dua saluran yang didesain khusus untuk mengirim maupun menerima data. Dengan pin yang digunakan SCL (*serial clock*) dan SDA (*serial data*). Pada I2C tidak perlu mengatur baud rate seperti pada USART.

Halaman sengaja dikosongkan

BAB III PERANCANGAN DAN PEMBUATAN ALAT

3.1 *Flow chart/ Diagram Alir Pembuatan Alat*

Pada subbab ini dijelaskan mengenai prosedur tahapan dalam penelitian tugas akhir yang dilakukan. Dengan *flowchart* dibawah ini:



Gambar 3.1 Flowchart Pengerjaan Tugas Akhir

Adapun keterangan gambar 3.1 mengenai digram alir dapat dijelaskan sebagai berikut:

a. Studi Literatur

Dilakukan kajian terhadap metode-metode, konsep, atau teori yang terkait dengan penelitian yang dilakukan, baik yang bersumber dari jurnal, laporan penelitian, maupun buku-buku yang memiliki bahasan yang sesuai dengan tema penelitian.

b. Perancangan Sistem Monitoring

Dilakukan perancangan sistem monitoring pada produksi Brown's Gas, perancangan dilakukan dengan membuat desain mekanik alat serta desain visualisasi data yang akan ditampilkan.

c. Pembuatan Alat

Dilakukan pembuatan sistem monitoring pada produksi Brown's Gas, pembuatan sistem monitoring dilakukan dengan membuat *Hardware* dan *software* dari sistem monitoring, pembuatan *hardware* meliputi pembuatan mekanik alat dan pembuatan *software* meliputi pembuatan *visualisasi* data yang akan ditampilkan berdasarkan hasil ukur alat ukur.

d. Analisis Kinerja Sistem Monitoring

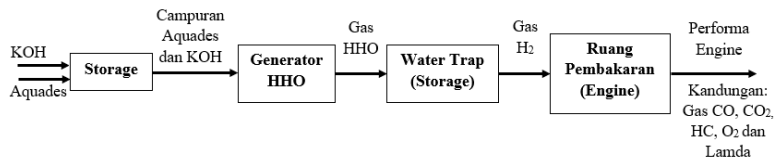
Analisis ini bertujuan untuk mengetahui apakah kinerja dari masing-masing komponen sesuai dengan perancangan mekanik alat serta bentuk *visualisasi* dari hasil pengukuran, apabila tidak sesuai maka akan dilakukan evaluasi terhadap perancangan sistem monitoring.

3.2 Gambaran Umum Produksi *Brown's Gas* pada Sepeda Motor

Pada perancangan tugas akhir ini dilakukan sistem monitoring level pada Brown's Gas, sistem monitoring temperatur ini terletak pada *storage* produksi Brown's Gas yang merupakan tempat pencampuran larutan Akuades dan KOH, pada *plant* ini terdapat beberapa proses elektrolisis yang terjadi pada Generator HHO, yang akan memproduksi Brown's Gas.

Pada gambar 3.2 merupakan proses yang ada pada Produksi *Brown's Gas* yang berawal dari campuran larutan Akuades dan KOH yang kemudian dialirkan ke Generator HHO ini terjadi proses elektrolisa antara campuran larutan Akuades dan KOH, ketinggian air pada *storage* harus dijaga sehingga terdapat sistem monitoring level pada *storage* produksi *Brown's Gas*, pada proses produksi gas juga terdapat sistem kontrol temperatur pada *storage* yang bertujuan untuk mengkontrol temperature dikarenakan *Brown's Gas* yang masuk pada ruang bakar tidak boleh melebihi suhu 60° .

Sebelum menuju *engine* (ruang pembakaran) gas H_2 akan menuju ke *water trap* agar yang masuk pada ruang bakar benar-benar mengandung gas H_2 tidak tercampur dengan larutan. Dari proses, pada ruang bakar tidak boleh terdapat kandungan O_2 , apabila terdapat kandungan O_2 berlebih, sensor O_2 akan mendeteksi kemudian akan mengatur aliran yang masuk ke ruang bakar.



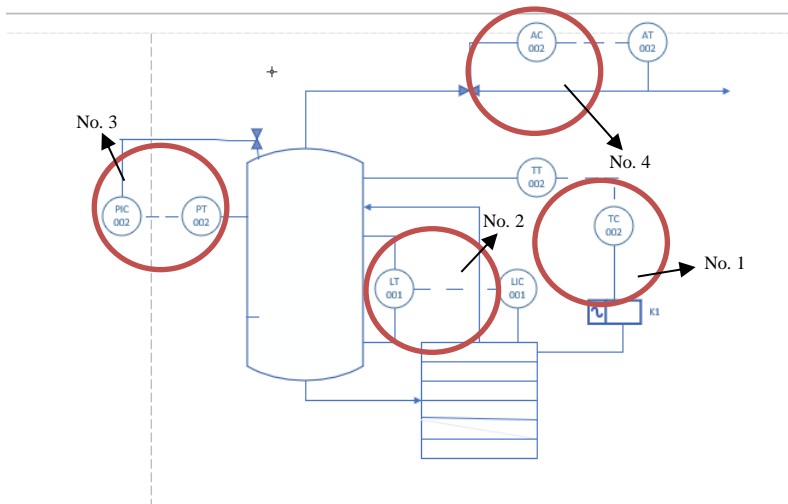
Gambar 3.2 Blok Flow Diagram Produksi *Brown's Gas*

3.3 Perancangan Sistem Monitoring

Pada tugas akhir ini dirancang Sistem monitoring level, tekanan, dan suhu pada *storage* serta O_2 pada gas buang. Pada produksi *Brown's Gas*, sistem ini bertujuan untuk menghasilkan produksi *Brown's Gas* yang ideal pada mesin kendaraan bermotor, pada *plant* ini proses untuk mengubah larutan campuran KOH dan akuades menjadi *Brown's Gas* yang digunakan untuk efisiensi bahan bakar pada sepeda motor.

Pada gambar 3.3 merupakan P&ID dari produksi *Brown's Gas* pada gambar tersebut sistem monitoring level, tekanan, dan suhu pada *storage* serta O_2 pada gas buang terdapat pada gambar

yang dilingkar merah, *Brown's Gas* merupakan hasil produksi dari suatu sistem elektrolisis pada Generator HHO. Pada gambar 3.4 dapat dilihat, ketika sensor mendeteksi level, tekanan, dan suhu pada storage serta O₂ pada gas buang maka sensor akan mengirimkan sinyal berupa tegangan listrik yang kemudian masuk ke pengkondisian sinyal dan diubah kedata digital agar dapat dikirim ke mikrokontroler. Mikrokontroler berfungsi memproses data untuk ditampilkan pada layar LCD, selain itu mikrokontroler juga berfungsi mengirim data ke data logger shield untuk proses pencatatan data. Dimana pada data logger shield terpasang memori SD Card yang berfungsi sebagai media penyimpan hasil pencatatan data output produksi *Brown's Gas*. Perangkat lunak yang digunakan adalah software CVAVR untuk memrogram data.

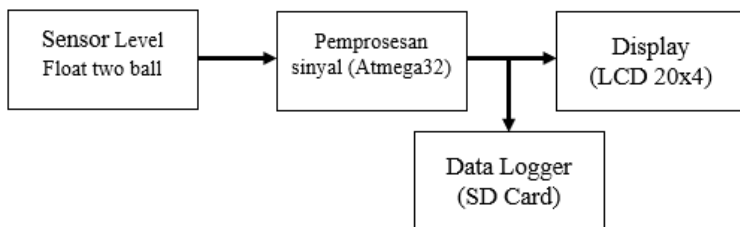


Gambar 3.3 *Piping & Instrumentation Diagram* dalam Sistem Monitoring dalam produksi *Brown's Gas*

Dari gambar P&id diatas menunjukkan bahwa terdapat 4 variabel yang akan dimonitoring, berikut adalah penjelasan untuk setiap variabel:

- Gambar No.1 adalah monitoring suhu Gas HHO yang akan menuju pada ruang pembakaran. Suhu ideal yang akan masuk pada ruang pembakaran adalah 45°C.
- Gambar No.2 adalah monitoring level dari ketinggian larutan aquades dan KOH pada storage. Campuran aquades dan KOH ini sangat penting dalam produksi *Brown's Gas*, jadi dapat mengetahui isi storage telah mencapai *low* agar segera mengisi kembali.
- Gambar No.3 adalah monitoring tekanan dalam storage selama produksi *Brown's Gas*. Tekanan pada storage harus tetap terjaga, tekanan pada storage tidak boleh melebihi *set point* yaitu 5,81 psi.
- Gambar No.4 adalah monitoring kandungan oksigen setelah ruang pembakaran. Agar mengetahui efisiensi dari produksi *Brown's Gas* setelah terjadi pembakaran pada *engine*.

Berikut adalah diagram blok dari sistem monitoring level serta suhu, tekanan dan oksigen:



Gambar 3.4 Diagram Blok sistem *Monitoring level*

3.4 Perancangan Hardware

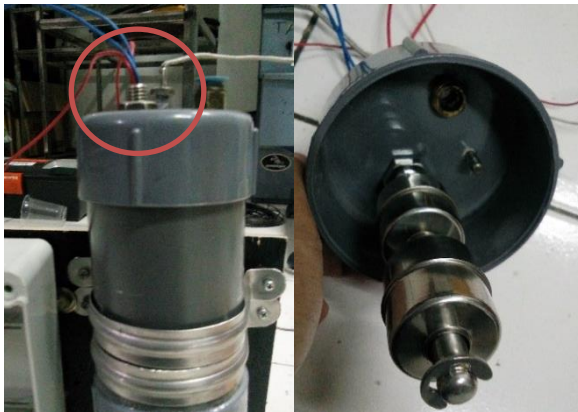
Pembuatan *hardware* untuk sistem monitoring ini terdapat perancangan pada rangkaian *sensing element*, perancangan pada

mikrokontroler Atmega32, dan display pada LCD 20x4. Perancangan Peletakan Mekanik Alat berikut ini merupakan penjelasan setiap perancangan tersebut:

3.4.1 Konfigurasi Sensor level Float

Dalam perancangan tugas akhir ini, sensor yang digunakan adalah sensor level float two ball. Sensor bisa digunakan pada saat 3 posisi *low*, *medium*, dan *high*. Sensor ini diletakkan di tutup *storage* yang tentunya larutan yang ada memiliki suhu paling tinggi 60°C dan tinggi level sekitar 15cm. Dan Panjang sensor 10mm. Berikut adalah spesifikasi sensor *level float*.

Bahan : *stainless steel*
Panjang kabel : cm
Panjang sensor: 100mm
Tegangan : maks. 220VDC
Arus : maks. 1.5A



Gambar 3.5 Peletakkan sensor *level float*

3.4.2 Perancangan Rangkaian Atmega32

Pada perancangan sistem monitoring ini menggunakan Atmega32. Mikrokontroler Atmega32 adalah mikrokontroler

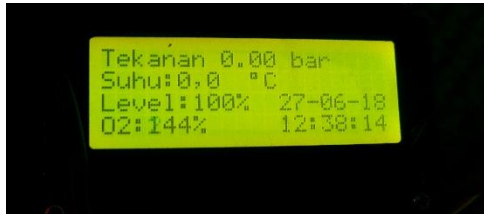
yang diproduksi oleh Atmel. Mikrokontroler ini memiliki *clock* dan kerjanya tinggi sampai 16 MHz, ukuran flash memorinya cukup besar, kapasistas SRAM sebesar 2 KiloByte, 32 buah port I/O yang sangat memadai untuk berinteraksi dengan LCD dan keypad. Hal ini digunakan karena sesuai dengan kebutuhan pengguna dalam menggunakan fungsi-fungsi mikrokontroler tersebut yang berfungsi sebagai tempat pengidentifikasi *input* dan *output data*. Pada mikrokontroler Atmega ini diaktifkan tiap port pada *power*, ADC, dan digital. Masing-masing port tersebut digunakan untuk tampilan LCD, dan led. Table 3.1 merupakan konfigurasi port yang digunakan untuk sistem ini. Sebagai aktivasi mikrokontroler Atmega32 ini mendapatkan sumber tegangan 5 volt.

Tabel 3.1 Konfigurasi Pada Atmega32

Komponen	Konfigurasi	Pin Pada Atmega32
Open Log Data Logger	TX	PIND.0
	RX	PIND.1
	VCC	VCC
	GND	GND
LCD	VDD	+5V
	VSS	GND
	RS	PINC.7
	EN	PINC.5
	RW	PINC.6
	D4	PINC.3
	D5	PINC.2
	D6	PINC.1
Level Switch	NO 1	PINA.0
	NO 2	PINA.1
RTC DS1307	SDA	PINB.0
	SCK	PINB.1
	VCC	VCC
	GND	GND

3.4.3 Display

Display merupakan suatu modul penampil. Untuk hal ini digunakan untuk menampilkan data yang terdeteksi oleh sensor sehingga dapat ditampilkan data berupa *digital* yang menunjukkan nilai suhu pada *steam output boiler*. Pada gambar 3.5 merupakan konfigurasi rangkaian LCD 20x4. LCD berfungsi untuk menampilkan suatu karakter huruf maupun angka. LCD 20x4 mempunyai tampilan 20 kolom dan 4 baris. Perancangan rangkaian LCD digunakan untuk menampilkan nilai persentase level, nilai tekanan, suhu dan kandungan O₂ pada produksi Brown's Gas. Digunakan mikrokontroler ATmega 32 untuk mengkoneksikan LCD 20x4.



Gambar 3.6 Konfigurasi display LCD 20x4

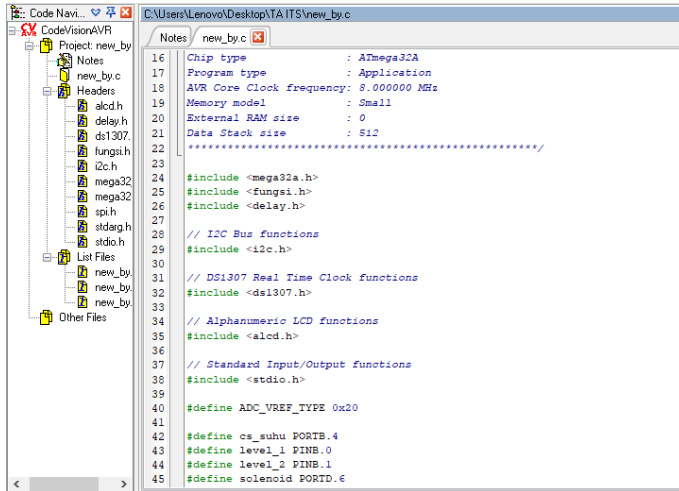
3.5 Perancangan Firmware

Pada perancangan alat ini terdapat 2 *software* yang masing-masing digunakan untuk membuat program sebelum di *compile* ke mikrokontroler dan digunakan sebagai *compiler* program ke mikrokontroler yaitu *Code Vision AVR* dan *Khazama*

3.5.1 Code Vision AVR

Pada pembuatan tugas akhir ini digunakan suatu *software* guna penunjang daripada sistem kontrol. *Software* yang digunakan untuk membuat program pada mikrokontroler ATmega 32 adalah *CodeVision AVR*. *listing* program atau koding dibuat untuk mengolah data yang telah dibaca oleh sensor untuk ditampilkan ke LCD. Gambar 3.7 ini merupakan cuplikan

dari listing program untuk instruksi mikrokontroler Atmega32 pada tugas akhir ini.



```

16 Chip type      : ATmega32A
17 Program type   : Application
18 AVR Core Clock frequency: 8.000000 MHz
19 Memory model   : Small
20 External RAM size : 0
21 Data Stack size : 512
22
23
24 #include <mega32a.h>
25 #include <fungsi.h>
26 #include <delay.h>
27
28 // I2C Bus functions
29 #include <i2c.h>
30
31 // DS1307 Real Time Clock functions
32 #include <ds1307.h>
33
34 // Alphanumeric LCD functions
35 #include <alcd.h>
36
37 // Standard Input/Output functions
38 #include <stdio.h>
39
40 #define ADC_VREF_TYPE 0x20
41
42 #define cs_suhu PORTB.4
43 #define level_1 PINB.0
44 #define level_2 PINB.1
45 #define solenoid PORTD.6

```

Gambar 3.7 Listing tampilan program

Setelah melakukan konfigurasi pada software CVAVR selanjutnya akan melakukan pemrograman untuk komponen yang akan digunakan.

a. Pembuatan sensor

Sensor Float two ball ini menggunakan PiN I/O biasa sehingga dapat menggunakan PINA.0 dan PINA.1. Berikut ini perintah untuk membaca data sensor level float pada storage produksi *Brown's Gas*.

```

void baca_level()
{
    if(level_1==0 && level_2==0){
        new_level=0;
        lcd_gotoxy(6,2);
        lcd_putsf("LOW ");
        level_low = on;
    }
}

```

} Kondisi 1

```

    }

    else if(level_1==1 && level_2==0){
        new_level=50;
        lcd_gotoxy(6,2);
        lcd_putsf("MEDIUM");
        level_low = off;
    }
    else if (level_1==1 && level_2==1)
    {
        new_level=100;
        lcd_gotoxy(6,2);
        lcd_putsf("HIGH ");
        level_low = off;
    }
    task_level();
    old_level=new_level;
}
void task_level()
{
    If(old_level != new_level)
        level = new_level;
}

```

} Kondisi 2

} Kondisi 3

Level float ini terdapat dua ball, untuk ball pertama atau ball yang berada diatas diinisiasikan menjadi level_2 dan ball kedua atau ball berada dibawah menjadi level_1. Berikut adalah penjelasan kondisi dalam sensor *level float*, level

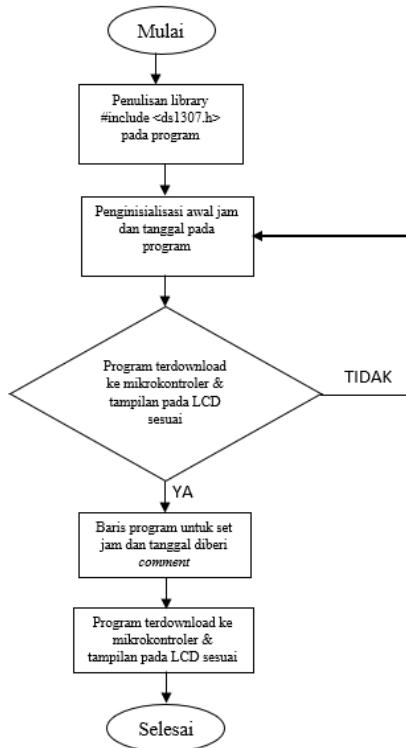
- Kondisi 1 adalah ketika level low. Terlihat ketika level_1 dan level_2 bernilai 0. Yang artinya kedua ball pada sensor level ini dalam kondisi off atau ke bawah.
- Kondisi 2 adalah ketika level medium. Terlihat ketika level_1 bernilai 1 dan level_2 bernilai 0. Yang artinya

ball diatas berada pada posisi off sedangkan ball dibawah berada pada posisi on atau ke atas.

- Kondisi 3 adalah ketika level high. Terlihat ketika level_1 dan level_2 bernilai 1. Yang artinya kedua ball pada sensor level ini dalam kondisi on atau keatas.

b. Pemrograman RTC (*Real Time Clock*)

Konfigurasi akses data pada *real time clock* dengan TWI (I2C) sudah bisa memanfaatkan library dari ds1307_twi.h. Dari library tersebut sudah disediakan fungsi-fungsi yang dibutuhkan untuk membaca data dari *real time clock*.



Gambar 3.8 Flowchart pembacaan RTC

Berikut baris program untuk melakukan inisialisasi awal.

```
jm = 12;
mn = 37;
dt = 00;
ming = 01;
tg = 8;
bl = 07;
th = 18;
//rtc_set_time(jm,mn,dt);
//rtc_set_date(ming,tg,bl,th);
```

Jika inisialisasi awal sudah dilakukan, maka untuk membaca data dapat dilakukan dengan fungsi yang sudah disediakan oleh library TWI. Berikut menunjukkan baris perintah untuk membaca dari RTC.

```
void get_time()
{
    rtc_get_time(&jm,&mn,&dt);
    rtc_get_date(&ming,&tg,&bl,&th);
}
```

Kemudian untuk menampilkan waktu pada LCD menggunakan perintah sebagai berikut:

```
lcd_gotoxy(12,2);
sprintf (tampil,"%02d-%02d-%02d",tg,bl,th);
lcd_puts (tampil);
lcd_gotoxy(12,3);
sprintf (tampil,"%02d:%02d:%02d",jm,mn,dt);
lcd_puts (tampil);
```

c. Pembuatan Data Logger

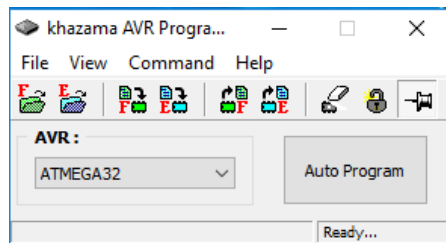
Berikut ini barisan program yang mengirimkan data pembacaan sensor-sensor, serta data jam dan tanggal:

```
void simpan_data()
{
    printf ("%02d-%02d-%02d",tg,bl,th);
    putchar(9);
```

```
printf ("%02d:%02d:%02d",jm,mn,dt);  
putchar(9);  
printf ("%0.2f",bar);  
putchar(9);  
printf ("%d,%d",suhu,koma);  
putchar(9);  
printf ("%d",level);  
putchar(9);  
printf ("%d",oksigen);  
putchar(10);  
putchar(13);  
}
```

3.5.2 Khazama

Software ini digunakan untuk meng-compile listing program yang dibuat pada code vision AVR ke mikrokontroler atmega8535. Selain itu, khazama juga dapat melakukan pengaturan clock dan proses eksekusi program ke mikrokontroler. Berikut ini adalah tampilan khazama.



Gambar 3.9 Tampilan Software Khazama

Halaman ini sengaja dikosongkan

BAB IV ANALISIS DATA DAN PEMBAHASAN

4.1 Hasil Rancang Bangun Alat

Setelah perancangan sistem monitoring *level* pada storage pada produksi Brown's Gas dibuat maka pengujian baik dari *hardware*, *controlling*, dan *software* perlu dilakukan. Pengujian dilakukan untuk mendapatkan data-data dari alat yang dirancang guna mengetahui spesifikasi serta performansi dari alat secara keseluruhan dan seberapa besar *error* atau kesalahan yang terjadi pada alat berdasarkan respon sistem dari nilai *set poin* yang diberikan.

Secara mekanisme kerja dari perancangan sistem monitoring *level* pada produksi Brown's Gas ini untuk mengetahui ketinggian air pada storage. Sensor *level float* digunakan untuk mengetahui ketinggian air pada produksi *Brown's Gas*. Ketika ketinggian level telah mencapai LOW akan menyalakan lampu LED hijau, agar memberi peringatan untuk mengisi kembali larutan campuran KOH dan akuades. Seperti pada gambar 4.1 merupakan hasil perancangan sistem monitoring *level*, suhu, tekanan, dan oksigen pada produksi Brown's Gas pada sepeda motor.

Pada sistem ini juga dilengkapi dengan RTC (*real time clock*) yang berfungsi untuk menampilkan waktu yang sebenarnya, openlog data logger yang digunakan untuk penyimpanan data ke memori SDCard dan LCD sebagai penampil data.



Gambar 4.1 Plant Produksi *Brown's Gas*

4.2 Pengujian Sensor *Level Float*

Untuk mengetahui keakuratan dari sensor level switch yang digunakan pada sistem, perlu adanya sebuah uji pada sensor tersebut. Pengujian dilakukan dengan menggunakan meteran sebagai acuan standar dengan satuan cm. Hasil dari pengujian sensor level float sebagai berikut:

Tabel 4.1 Hasil Pengujian Sensor *Level Float*

No	Level (cm)	Status Ball 1	Status Ball 2	Ketinggian (%)
1.	0	<i>Off</i>	<i>Off</i>	Low
2.	0,5	<i>Off</i>	<i>Off</i>	Low
3.	1,5	<i>Off</i>	<i>Off</i>	Low
4.	2,5	<i>Off</i>	<i>Off</i>	Low
5.	3,5	<i>Off</i>	<i>On</i>	Medium
6.	4	<i>Off</i>	<i>On</i>	Medium
7.	4,5	<i>Off</i>	<i>On</i>	Medium
8.	5,5	<i>Off</i>	<i>On</i>	Medium
9.	6,5	<i>Off</i>	<i>On</i>	Medium
10.	7	<i>On</i>	<i>On</i>	High

Hasil pengujian level tersebut menyebutkan hanya terdapat tiga keadaan level ketinggian air yaitu low, medium dan high. Pada saat keadaan low, indicator berupa LED hijau akan menyala menandakan storage larutan perlu untuk diisi kembali. Jika level pada storage kosong maka tidak akan terjadi produksi Brown's Gas.

4.3 Hasil Pengujian RTC

Hasil dari RTC DS1307 dengan waktu dan jam yang ditunjukkan pada gambar dibawah ini :



Gambar 4.2 Hasil RTC DS1307

Pada gambar diatas menunjukkan bahwa keterangan waktu pada RTC sama dengan keterangan waktu yang ditampilkan oleh handphone. Berikut hasil dari RTC yang ditunjukkan pada table dibawah ini:

Tabel 4.2 Hasil pengujian RTC

Jam dan waktu pada HP		Jam dan waktu pada LCD	
Date	Time	Date	Time
27/06/2018	12:38:51	27/06/2018	12:38:14
08/07/2018	14:43:47	08/07/2018	14:43:00
09/07/2018	12:38:10	09/07/2018	12:37:23
14/07/2018	19:25:45	14/07/2018	19:25:34
17/07/2018	01:58:09	17/07/2018	01:58:23

Hasil pengujian tersebut diambil dengan membandingkan tanggal dan jam yang diambil pada Handphone dan LCD 20x4 pada plant Brown's Gas. Hasil tersebut menunjukkan bahwa

tanggal pada HP dan LCD sama, sedangkan jam pada keduanya mempunyai selisih beberapa detik. Selisih tersebut disebabkan karena terdapat selang waktu ketika program didownload pada mikrokontroler, sehingga menyebabkan rtc kembali mengatur ulang tanggal dan jam pada mikrokontroler tersebut.

4.4 Hasil *Openlog Datalogger*

Penyimpanan data menggunakan openlog datalogger berhasil disimpan pada Microsoft excel. Pada modul ini menggunakan pin TX yang akan disambungkan pada RX mikrokontroler Atmega32 yaitu pada PIND.0 dan pin RX modul akan disambungkan pada pin TX mikrokontroler Atmega32 yaitu pada PIND.1.

Pengambilan data pada produksi Brown's Gas bertujuan untuk mengetahui nilai level larutan, suhu, dan tekanan pada storage produksi Brown's Gas serta mengetahui kadar oksigen yang keluar setelah ruang pembakaran. Pengambilan data ini diambil data dari level, tekanan, suhu dan oksigen disimpan pada sdcard dengan pemrosesan menggunakan Atmega32. Penyimpanan data menggunakan micro sdcard berhasil disimpan pada file.txt yang kemudian akan diubah menjadi file.csv. Untuk mengetahui waktu menggunakan Modul real Time Clock (RTC) DS1307, modul ini menggunakan pin Vcc Gnd, pin SDA disambungkan pada pin SDA mikrokontroler Atmega32 pada pin PD1 dan pin SCL pada modul RTC disambungkan pada pin SCL mikrokontroler Atmega32 pada pin PD0. Pada tabel 4.2 merupakan hasil penyimpanan data logger dalam bentuk data excel yang dilakukan beberapa pengambilan data.

Komunikasi data yang digunakan untuk modul Open log ini adalah USART, karena membutuhkan port RX untuk jalur perpindahan data dan TX. Untuk mengirim data Modul open log ini tidak menggunakan protokol untuk dapat menyimpan pada SDCard, mikrokontroler hanya perlu mengirim data secara serial ke open log sehingga bisa terbaca oleh SDCard.

Adapun urutan protokol pada proses penyimpanan data adalah sebagai berikut:

DATA PAKET PENGIRIMAN										
Date (9)	Time (9)	Tekanan (9)	Suhu (9)	Level (9)	Oksigen (10)	(13)				
A (9)	B (9)	C (9)	D (9)	E (9)	F (10)	(13)				

Gambar 4.3 Data Paket Pengiriman SDCard

Gambar 4.3 Menggambarkan data paket pengiriman yang dibahas lebih detail pada subbab berikutnya. Paket data A menyatakan

1. Menulis program untuk judul yang akan disimpan pada SdCard.

```
void tulis_judul()
{
    printf("Date");
    putchar(9);
    printf("Time");
    putchar(9);
    printf("Tekanan");
    putchar(9);
    printf("Suhu");
    putchar(9);
    printf("Level");
    putchar(9);
    printf("Oksigen");
    putchar(10);
    putchar(13);
}
```

Untuk untuk mengirimkan data hasil pembacaan sensor digunakan perintah printf(); yang ada pada library stdio.h. Sebelum hasil pembacaan sensor, dituliskan judul dari urutan Date, Time, Tekanan, Suhu, Level, dan Oksigen disesuaikan dengan urutan pembacaan sensor yang ada. Setiap judul terpisahkan oleh kolom dengan menggunakan kode ASCII control

characters dengan perintah “putchar(9)” yang artinya Horizontal Tab. Ketika sudah pada judul terakhir yaitu oksigen, kemudian menggunakan perintah “putchar(10)” untuk memerintah data selanjutnya agar berganti line bawahnya. Kemudian menuliskan perintah “putchar(13)” artinya Carriage return. Untuk void tulis_judul() dipanggil diluar while karena hanya dipanggil satu kali untuk judul pada penyimpanan.

2. Menulis program untuk mengirimkan data hasil pembacaan sensor.

```
void simpan_data()
{
    printf ("%02d-%02d-%02d",tg,bl,th);
    putchar(9);
    printf ("%02d:%02d:%02d",jm,mn,dt);
    putchar(9);
    printf ("%0.2f",bar);
    putchar(9);
    printf ("%d,%d",suhu,koma);
    putchar(9);

    if (new_level == 0)
        printf ("LOW");
    else if (new_level == 50)
        printf ("MEDIUM");
    else if (new_level == 100)
        printf ("HIGH");

    putchar(9);
    printf ("%d",oksigen);
    putchar('% ');
    putchar(10);
    putchar(13);
}
```

Sama halnya dengan penulisan judul pada penyimpanan SdCard, untuk urutan hasil pembacaan sensor serta tanggal dan

waktu juga sama mengikuti urutan dari penulisan judul tersebut. Yang pertama kolom penyimpanan tanggal kemudian diberikan perintah Horizontal Tab, dilanjutkan untuk penyimpanan waktu. Dari penyimpanan waktu kemudian diperintahkan kembali untuk Horizontal Tab yang dilanjutkan untuk penyimpanan pembacaan sensor tekanan dan untuk seterusnya sensor suhu, level, oksigen ditulis secara berurutan. Yang terakhir setelah pemanggilan sensor oksigen dilanjutkan dengan perintah line feed atau pindah line bawahnya. Kemudian dilanjutkan dengan perintah carriage return. Untuk void simpan_data() dipanggil di dalam while karena dipanggil berulang kali untuk pembacaan dari sensor serta waktu dan tanggalnya.

3. Dengan program seperti langkah 1 dan 2 pembacaan sensor serta waktu dan tanggal dapat tersimpan pada SDCard. Format yang telah tersimpan pada SDCard adalah *.TXT*. Untuk dapat melihat hasil dari penyimpanan file *.TXT* tersebut di save as dengan diubah menjadi *.CSV*. Maka data dapat dilihat dari Microsoft excel.

Tabel 4.3 Penyimpanan Data saat level kondisi Low

Date	Time	Tekanan	Suhu	Level	Oksigen
16/07/2018	22:30:00	0.44	42,12	LOW	15%
16/07/2018	22:31:00	0.44	43,1	LOW	15%
16/07/2018	22:32:00	0.44	42,31	LOW	15%
16/07/2018	22:33:00	0.44	43,1	LOW	15%
16/07/2018	22:34:00	0.44	42,12	LOW	15%
16/07/2018	22:35:00	0.44	44,2	LOW	15%
16/07/2018	22:36:00	0.44	43,21	LOW	15%
16/07/2018	22:37:00	0.44	41,22	LOW	15%
16/07/2018	22:38:00	0.44	41,13	LOW	15%
16/07/2018	22:39:00	0.44	44,2	LOW	15%
16/07/2018	22:40:00	0.44	44,1	LOW	15%

Tabel 4.4 Penyimpanan data saat level kondisi Medium

Date	Time	Tekanan	Suhu	Level	Oksigen
17/07/2018	14:44:00	0.44	39,34	MEDIUM	15%
17/07/2018	14:45:00	0.03	39,5	MEDIUM	2%
17/07/2018	14:46:00	0.03	39,5	MEDIUM	2%
17/07/2018	14:47:00	0.03	39,24	MEDIUM	2%
17/07/2018	14:48:00	0.00	38,35	MEDIUM	2%
17/07/2018	14:49:00	0.03	40,14	MEDIUM	1%
17/07/2018	14:50:00	0.03	41,13	MEDIUM	2%
17/07/2018	14:51:00	0.03	39,5	MEDIUM	1%
17/07/2018	14:52:00	0.03	42,22	MEDIUM	1%

Tabel 4.5 Penyimpanan data saat level kondisi High

Date	Time	Tekanan	Suhu	Level	Oksigen
14/07/2018	23:51:00	0.03	34,39	HIGH	1%
14/07/2018	23:52:00	0.03	36,37	HIGH	2%
14/07/2018	23:53:00	0.03	37,7	HIGH	2%
14/07/2018	23:54:00	0.03	38,16	HIGH	2%
14/07/2018	23:55:00	0.03	38,35	HIGH	2%
14/07/2018	23:56:00	0.03	39,5	HIGH	1%
14/07/2018	23:57:00	0.03	38,35	HIGH	2%
14/07/2018	23:58:00	0.03	40,4	HIGH	1%
14/07/2018	23:59:00	0.03	39,24	HIGH	1%
15/07/2018	00:00:00	0.03	39,34	HIGH	2%

Dari ketiga table diatas table 4.3 sampai 4.5. Pada table 4.3 data diambil ketika produksi dari *Brown's Gas* sudah sekitar 2 jam maka ditunjukkan bahwa level mencapai posisi low, hal tersebut dikarenakan campuran aquades dan KOH sudah digunakan digunakan untuk pengujian beberapa jam. Suhu dari *Brown's Gas* yang ditunjukkan kurang lebih telah mencapai sekitar 40°C, nilai tersebut adalah suhu minimum yang ideal dari *Brown's Gas* yang akan masuk pada *engine*. Tekanan pada

storage stabil 0.44 psi, nilai tekanan berbanding lurus dengan nilai suhu yang meningkat. Dan untuk kandungan Oksigen stabil mencapai 15% dikarenakan produksi *Brown's Gas* sudah stabil.

Pada table 4.4 data diambil ketika produksi dari *Brown's Gas* sudah sekitar 1 jam maka ditunjukkan bahwa level mencapai medium dikarenakan campuran aquades dan KOH hanya digunakan pengujian beberapa menit. Suhu dari *Brown's Gas* yang ditunjukkan mengalami peningkatan dari suhu 39°C hingga 41°C. Ketika sistem telah memproduksi *Brown's Gas*, tutup storage dibuka yang menyebabkan tekanan pada storage semula menunjukkan nilai 0.44 psi kemudian turun mencapai 0.03 psi. Serta menyebabkan kandungan Oksigen yang semula mencapai 15% menurun hingga 2% dikarenakan *Brown's Gas* pada storage hilang, sehingga *Brown's Gas* yang masuk pada *engine* sangat sedikit.

Pada table 4.5 data diambil ketika produksi dari *Brown's Gas* baru beberapa menit, maka ditunjukkan bahwa level masih mencapai *high*, produksi dari *Brown's Gas* masih belum lama. Dengan begitu, menyebabkan Suhu, tekanan dan kandungan Oksigen yang tidak menunjukkan peningkatan angka. Dikarenakan produksi *Brown's Gas* akan maksimal apabila produksi sekitar 2 jam.

Untuk penyimpanan data dari level, temperature dan tekanan pada storage serta kadar oksigen di gas buang pada produksi *Brown's Gas* ini menggunakan modul openlog datalogger sebagai penyimpanan data dan SD Card yang berkapasitas 8 GB. Pencatatan data yang dilakukan selama satu menit membutuhkan ruang penyimpanan pada SD Card rata-rata sebesar $3 \text{ KB} = 0.003 \text{ MB}$ sehingga dalam waktu 1 hari akan membutuhkan kapasitas memory sebanyak 4.37MB, SD Card dengan kapasitas 8 GB memiliki nilai kapasitas maksimal yang bisa digunakan adalah 7390 MB, Sehingga jumlah pencatatan yang dapat dilakukan dengan menggunakan SD Card yang berkapasitas 8GB adalah sebagai berikut:

$$\text{Lama Waktu} = \frac{\text{Kapasitas SD Card}}{\text{Ukuran file per hari}} \quad (4.1)$$

Dari persamaan 4.1 maka penggunaan memory dapat digunakan selama 1691,075 hari atau kurang lebih selama 4,5 tahun.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil pengujian dan perancangan sistem monitoring level, suhu, tekanan, dan oksigen pada produksi *Brown's Gas* maka dapat didapatkan kesimpulan sebagai berikut :

1. Telah dibuat rancang bangun sistem monitoring level, tekanan, suhu pada storage serta oksigen pada keluaran ruang pembakaran , dengan adanya sistem monitoring tersebut maka level pada storage dapat selalu terjaga, agar produksi *Brown's Gas* tetap berjalan.
2. Telah berhasil dirancang sistem monitoring level, suhu, tekanan, dan oksigen dengan tampilan *real time* pada LCD 20x4 serta sistem monitoring ini dengan hasil yang dapat di-*record* dan disimpan langsung pada laptop menggunakan memori penyimpanan SD Card dengan format excel. Kapasitas memori Sd Card ini selama 4,5 tahun.
3. Dari data penyimpanan yang telah diperoleh bahwa produksi *Brown's Gas* telah mencapai ideal, ditunjukkan bahwa nilai suhu dari Gas HHO bernilai ideal 40°C-44°C, kandungan oksigen setelah ruang pembakaran mencapai 15% serta tekanan pada storage dalam keadaan stabil bernilai 0.44 psi. Produksi saat ini dilakukan selama 2 jam dan menunjukkan larutan campuran aquades dan KOH dalam kondisi low.

5.2 Saran

Berdasarkan penelitian yang telah dilakukan terdapat beberapa saran diantaranya sebagai berikut :

1. Penggunaan wiring perkabelan sensor dan controller pada plant lebih rapi, agar untuk mengintegrasikan akan lebih mudah

2. Penggunaan anti redam getaran pada plant untuk lebih diperhatikan lagi, agar tampilan pada display tidak mengacak dan lebih mudah untuk perbaiki.

DAFTAR PUSTAKA

- [1] Wirawan, Fahmi, Kawano, Djoko, S. 2014. “*Pengaruh Penambahan Gas HHO Terhadap Unjuk Kerja Mesin Diesel Putaran Konstan Dengan Variasi Massa Katalis KOH pada Generator Gas HHO*”. Surabaya, Teknik Mesin, FTI, ITS.
- [2] Abdurrakhman, A., Guntur, H. L. 2010. “*Studi Eksperimental Pengaruh Variasi Tegangan Listrik, Konsentrasi Katalis dan Temperatur Terhadap Optimalisasi Sistem Elektrolisis Brown’s Gas*”. Surabaya, Teknik Mesin, FTI, ITS.
- [3] Share ITS. 2013. Teknik Pengukuran dan Kalibrasi. ITS. Surabaya.
- [4] Bentley, John P. Principles of Measurement Systems – 4th edition. London: Pearson Education Limited
- [5] Agni, Brillyano. 2012, “*Studi Ekperimen Pengaruh Pencampuran Gas Hidrogen dari Generator HHO Tipe Kering dengan Bahan Bakar Kerosene Pada Distribusi Temperatur Nyakla Api Kompor Tekan Blowtorch*”, Tugas Akhir, Teknik Mesin Institut Teknologi Sepuluh Nopember, Surabaya.
- [6] Marsudi, 2009, *Desain Pengaturan Otomatis Pompa Air Listrik Berbasis Mikrokontroller AT89S51*, Universitas Mercu Buana, Yogyakarta.
- [7] Supriyadi, Mukh Yasir Nur. 2012. *Pengenalan Mikrokontroller AVR Atmega32* . Bandung : Laboratorium Hardware Program Study Teknik Komputer FMIPA UNPAD.
- [8] Supriyadi, Mukh Yasin Nur. 2014. *Modul I Pengenalan Mikrokontroler AVR ATMega8535*, Bandung: Laboratorium Hardware Program Studi Teknik Komputer FMIPA UNPAD.

- [9] <https://www.scribd.com/doc/185920131/LCD-20X4//>
diakses pada 26 Juni 2018
- [10] <https://proyekarduino.wordpress.com/2015/04/01/pengetahuan-dasar-rtc-ds1307/> diakses pada tanggal 01 April 2018
- [11] <https://www.sparkfun.com/products/13712SparkFunOpenLog>, diakses pada tahun tanggal 26 juni 2018
- [12] Natalindo, Nuncio. 2017. *Modul Komunikasi serial pada model simulator boiler berbasis mikrokontroler Atmega16*, Yogyakarta: Teknik Elektro Universitas Sanata Darma

LAMPIRAN A (Datasheet)

- **Data Sheet Sensor *Level Float*:**

Stem Mounted Multi-Point Level Switch FL Series



The FL Magnetic level switch is designed to meet demanding customer applications for liquid level sensing in rugged hostile environments. Each FL is manufactured to the users specifications making it fit to work precisely according to the application requirements.

Features

- Custom tailored to user specifications
- Long switch life
- Up to seven switch points

FL Series Specifications

Performance

- FLR: 7 Switch Points
Min. distance between levels: 1.75 in. (45mm)
- FLE: 3 Switch Points

Environmental

- Operating Temperature:
14° to 212°F (-10° to 100°C)



Certification

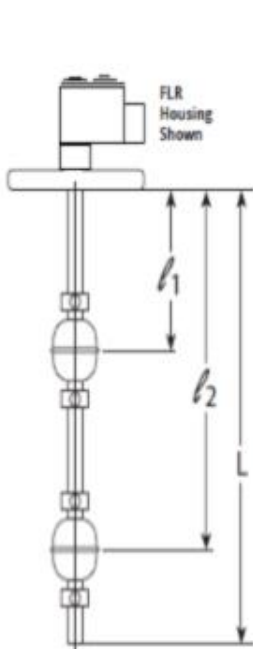
- FLE:
NEMA 4 & 7, IP65
IP43
- FLR:
NEMA 4 & 7, IP65

Electrical

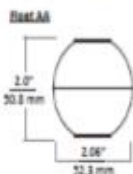
- Switch Rating:
FLE: 20 VA, 50 VA
FLR: 50 VA, 180 VA
- Max Current:
0.5 A AC
- Max Voltage:
220 VAC

Physical

- Skirt Length:
FLE: 24 in. (610 mm)
FLR: 153 in. (3900 mm)
- Cable Entry:
FLE: 3/4 in.
FLR: 3/4 in.



FLR Floats



FLE Floats



FLR Model Configuration Options

Model Number: FLR- _____
 A B C D E F G H I

A. Mounting Type

- 0 Flat face flange
- 3 Triclamp
- 4 Plug mounted from outside of tank

B. Mounting Option and Size

- A_* Ansi flange 150# (size=2, 2.5, 3)
- SF_* Triclamp (size=2)
- T_* NPT plug (size=1.5, 2, 2.5, 3)

*Note: Add an 'S' after mount size for Slide Connection

C. Material

- 56 316L SS

D. Housing

- H3 NEMA 4 & 7, IP65 (3/4 in. cable entry)
- W_ No housing, mounting option with 3/4 NPT plug and 12 or 36 in. lead wires

E. Reed Switch

- A 50 VA
- C 180 VA

F. Number of Switch Points

- 1-7 Select the number of switch points required

G. Number of Floats

- 1-7 Select the number of floats required

H. Float Type

- AA 316L SS (2.06 in. diameter, 0.59 SG)
- AA.92 316L SS (2.06 in. diameter, 0.92 SG)
- BB 316L SS (1.63 in. diameter, 0.607 SG)
- BB.92 316L SS (1.63 in. diameter, 0.92 SG)

I. Probe Length (in./mm)

- L_ in./mm (up to 153 in / 3900 mm)

Switch Point Location(s)

(Measured from process connection)

- 1 ___ in./mm (designate NO or NC position)
- 2 ___ in./mm (designate NO or NC position)
- 3 ___ in./mm (designate NO or NC position)
- 4 ___ in./mm (designate NO or NC position)
- 5 ___ in./mm (designate NO or NC position)
- 6 ___ in./mm (designate NO or NC position)
- 7 ___ in./mm (designate NO or NC position)

• Data Sheet Atmega32:

Features

- High-performance, Low-power Atmel®AVR® 8-bit Microcontroller
- Advanced RISC Architecture
 - 131 Powerful Instructions – Most Single-clock Cycle Execution
 - 32 x 8 General Purpose Working Registers
 - Fully Static Operation
 - Up to 16 MIPS Throughput at 16MHz
 - On-chip 2-cycle Multiplier
- High Endurance Non-volatile Memory segments
 - 32Kbytes of In-System Self-programmable Flash program memory
 - 1024Bytes EEPROM
 - 2Kbytes Internal SRAM
 - Write/Erase Cycles: 10,000 Flash/100,000 EEPROM
 - Data retention: 20 years at 85°C/100 years at 25°C⁽¹⁾
 - Optional Boot Code Section with Independent Lock Bits
 - In-System Programming by On-chip Boot Program
 - True Read-While-Write Operation
 - Programming Lock for Software Security
- JTAG (IEEE std. 1149.1 Compliant) Interface
 - Boundary-scan Capabilities According to the JTAG Standard
 - Extensive On-chip Debug Support
 - Programming of Flash, EEPROM, Fuses, and Lock Bits through the JTAG Interface
- Peripheral Features
 - Two 8-bit Timer/Counters with Separate Prescalers and Compare Modes
 - One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
 - Real Time Counter with Separate Oscillator
 - Four PWM Channels
 - 8-channel, 10-bit ADC
 - 8 Single-ended Channels
 - 7 Differential Channels in TQFP Package Only
 - 2 Differential Channels with Programmable Gain at 1x, 10x, or 200x
 - Byte-oriented Two-wire Serial Interface
 - Programmable Serial USART
 - Master/Slave SPI Serial Interface
 - Programmable Watchdog Timer with Separate On-chip Oscillator
 - On-chip Analog Comparator
- Special Microcontroller Features
 - Power-on Reset and Programmable Brown-out Detection
 - Internal Calibrated RC Oscillator
 - External and Internal Interrupt Sources
 - Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby and Extended Standby
- I/O and Packages
 - 32 Programmable I/O Lines
 - 40-pin PDIP, 44-lead TQFP, and 44-pad QFN/MLF
- Operating Voltages
 - 2.7V - 5.5V for ATmega32L
 - 4.5V - 5.5V for ATmega32
- Speed Grades
 - 0 - 8MHz for ATmega32L
 - 0 - 16MHz for ATmega32
- Power Consumption at 1MHz, 3V, 25°C
 - Active: 1.1mA
 - Idle Mode: 0.35mA
 - Power-down Mode: < 1µA



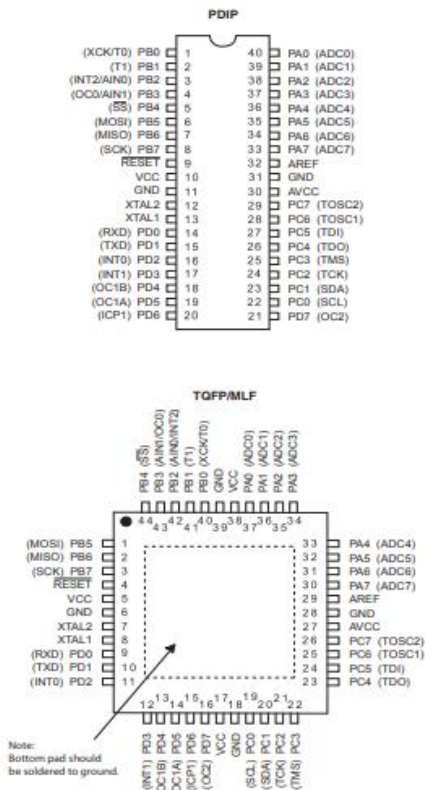
8-bit AVR[®]
Microcontroller
with 32KBytes
In-System
Programmable
Flash

ATmega32
ATmega32L

2503C-AVR-02/11



Pin Configurations

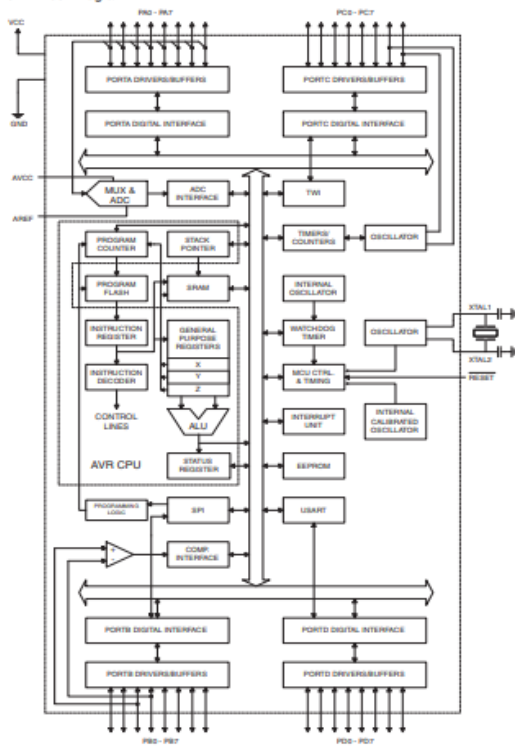
Figure 1. Pinout ATmega32


Overview

The Atmel®AVR®ATmega32 is a low-power CMOS 8-bit microcontroller based on the AVR enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega32 achieves throughputs approaching 1 MIPS per MHz allowing the system designer to optimize power consumption versus processing speed.

Block Diagram

Figure 2. Block Diagram



The Atmel® AVR® AVR core combines a rich instruction set with 32 general purpose working registers. All the 32 registers are directly connected to the Arithmetic Logic Unit (ALU), allowing two independent registers to be accessed in one single instruction executed in one clock cycle. The resulting architecture is more code efficient while achieving throughputs up to ten times faster than conventional CISC microcontrollers.

The ATmega32 provides the following features: 32Kbytes of In-System Programmable Flash Program memory with Read-While-Write capabilities, 1024bytes EEPROM, 2Kbyte SRAM, 32 general purpose I/O lines, 32 general purpose working registers, a JTAG interface for Boundary-scan, On-chip Debugging support and programming, three flexible Timer/Counters with compare modes, Internal and External Interrupts, a serial programmable USART, a byte oriented Two-wire Serial Interface, an 8-channel, 10-bit ADC with optional differential input stage with programmable gain (TQFP package only), a programmable Watchdog Timer with Internal Oscillator, an SPI serial port, and six software selectable power saving modes. The Idle mode stops the CPU while allowing the USART, Two-wire interface, A/D Converter, SRAM, Timer/Counters, SPI port, and interrupt system to continue functioning. The Power-down mode saves the register contents but freezes the Oscillator, disabling all other chip functions until the next External Interrupt or Hardware Reset. In Power-save mode, the Asynchronous Timer continues to run, allowing the user to maintain a timer base while the rest of the device is sleeping. The ADC Noise Reduction mode stops the CPU and all I/O modules except Asynchronous Timer and ADC, to minimize switching noise during ADC conversions. In Standby mode, the crystal/resonator Oscillator is running while the rest of the device is sleeping. This allows very fast start-up combined with low-power consumption. In Extended Standby mode, both the main Oscillator and the Asynchronous Timer continue to run.

The device is manufactured using Atmel's high density nonvolatile memory technology. The On-chip ISP Flash allows the program memory to be reprogrammed in-system through an SPI serial interface, by a conventional nonvolatile memory programmer, or by an On-chip Boot program running on the AVR core. The boot program can use any interface to download the application program in the Application Flash memory. Software in the Boot Flash section will continue to run while the Application Flash section is updated, providing true Read-While-Write operation. By combining an 8-bit RISC CPU with In-System Self-Programmable Flash on a monolithic chip, the Atmel ATmega32 is a powerful microcontroller that provides a highly-flexible and cost-effective solution to many embedded control applications.

The Atmel AVR ATmega32 is supported with a full suite of program and system development tools including: C compilers, macro assemblers, program debugger/simulators, in-circuit emulators, and evaluation kits.

Pin Descriptions

VCC Digital supply voltage.

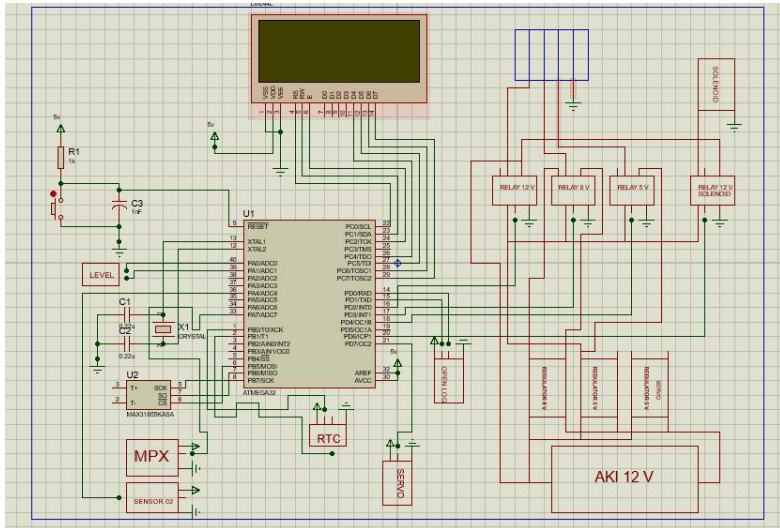
GND Ground.

Port A (PA7..PA0) Port A serves as the analog inputs to the A/D Converter.

Port A also serves as an 8-bit bi-directional I/O port, if the A/D Converter is not used. Port pins can provide internal pull-up resistors (selected for each bit). The Port A output buffers have symmetrical drive characteristics with both high sink and source capability. When pins PA0 to PA7 are used as inputs and are externally pulled low, they will source current if the internal pull-up resistors are activated. The Port A pins are tri-stated when a reset condition becomes active, even if the clock is not running.

Port B (PB7..PB0)	<p>Port B is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port B output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port B pins that are externally pulled low will source current if the pull-up resistors are activated. The Port B pins are tri-stated when a reset condition becomes active, even if the clock is not running.</p> <p>Port B also serves the functions of various special features of the ATmega32 as listed on page 57.</p>
Port C (PC7..PC0)	<p>Port C is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port C output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port C pins that are externally pulled low will source current if the pull-up resistors are activated. The Port C pins are tri-stated when a reset condition becomes active, even if the clock is not running. If the JTAG interface is enabled, the pull-up resistors on pins PC5(TDI), PC3(TMS) and PC2(TCK) will be activated even if a reset occurs.</p> <p>The TD0 pin is tri-stated unless TAP states that shift out data are entered.</p> <p>Port C also serves the functions of the JTAG interface and other special features of the ATmega32 as listed on page 60.</p>
Port D (PD7..PD0)	<p>Port D is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port D output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port D pins that are externally pulled low will source current if the pull-up resistors are activated. The Port D pins are tri-stated when a reset condition becomes active, even if the clock is not running.</p> <p>Port D also serves the functions of various special features of the ATmega32 as listed on page 62.</p>
<u>RESET</u>	<p>Reset Input. A low level on this pin for longer than the minimum pulse length will generate a reset, even if the clock is not running. The minimum pulse length is given in Table 15 on page 37. Shorter pulses are not guaranteed to generate a reset.</p>
XTAL1	<p>Input to the inverting Oscillator amplifier and input to the internal clock operating circuit.</p>
XTAL2	<p>Output from the inverting Oscillator amplifier.</p>
AVCC	<p>AVCC is the supply voltage pin for Port A and the A/D Converter. It should be externally connected to V_{CC}, even if the ADC is not used. If the ADC is used, it should be connected to V_{CC} through a low-pass filter.</p>
AREF	<p>AREF is the analog reference pin for the A/D Converter.</p>

LAMPIRAN B (Wiring Plant Brown's Gas)



LAMPIRAN C
LISTING PROGRAM PADA CVAVR

/******
*

This program was produced by the
CodeWizardAVR V2.05.3 Standard
Automatic Program Generator
© Copyright 1998-2011 Pavel Haiduc, HP InfoTech s.r.l.
<http://www.hpinfotech.com>

Project :
Version :
Date : 6/27/2018
Author : TEAM BROWN'S GAS
Company :
Comments:

Chip type : ATmega32A
Program type : Application
AVR Core Clock frequency: 8.000000 MHz
Memory model : Small
External RAM size : 0
Data Stack size : 512

/
#include <mega32a.h>
#include <funksi.h>
#include <delay.h>
// I2C Bus functions
#include <i2c.h>
// DS1307 Real Time Clock functions
#include <ds1307.h>

```

// Alphanumeric LCD functions
#include <alcd.h>
// Standard Input/Output functions
#include <stdio.h>
#define ADC_VREF_TYPE 0x20
#define cs_suhu PORTB.4
#define level_1 PINA.0
#define level_2 PINA.1
#define solenoid PORTD.6
#define level_low PORTD.5
#define relay_3 PORTD.4
#define relay_2 PORTD.3
#define relay_1 PORTD.2
#define off 1
#define on 0
#define servo OCR2
#define buka 30
#define buka_setengah 90
#define tutup 150

float v,psi,kpa,mmhg,bar;
unsigned char tampil[33];

unsigned char old_kondisi_suhu, kondisi_suhu;
signed char adc_tekanan=0;
signed char adc_oksigen=0,kondisi_servo;
int xcount=0;
signed int oksigen,oksi_ref=51;
int level=0,old_level=0,new_level=0;
unsigned char jm,mn,dt,tg,bl,th,ming,old_dt;
unsigned result,satuan,suhu,koma;
int ever_open=0,count_press=0;

```

```

float add_press=0;
// Read the 8 most significant bits
// of the AD conversion result
unsigned char read_adc(unsigned char adc_input)
{
ADMUX=adc_input | (ADC_VREF_TYPE & 0xff);
// Delay needed for the stabilization of the ADC input voltage
delay_us(10);
// Start the AD conversion
ADCSRA|=0x40;
// Wait for the AD conversion to complete
while ((ADCSRA & 0x10)==0);
ADCSRA|=0x10;
return ADCH;
}
// SPI functions
#include <spi.h>
// Declare your global variables here
//////////////////////////////////TEKANAN//////////////////////////////////
void baca_tekanan()
{
    adc_tekanan =(read_adc(3)-42);
    if(adc_tekanan <=0)
        adc_tekanan = 0;
    v = adc_tekanan * 0.01960784 ;
    kpa =(v/5)/0.0012858;
    psi = kpa*0.145;
    mmhg = kpa*7.5;
    bar = (psi * 0.0690);
    task_tekanan();
}
void task_tekanan()

```

```

{
  if(psi>=5.8)
  {
    solenoid = on;
    ever_open = 1;
  }
  else
  {
    solenoid = off;
  }
}
//////////////////////////////////SUHU ////////////////////////////////////
void baca_suhu()
{
  cs_suhu = on;
  result = (unsigned) spi(0)<<8;
  result |= spi(0);
  cs_suhu = off;

  result = (unsigned)((((unsigned long ) result * 5000)/4096L);
  satuan =result/40;
  koma = result%40;
  suhu = satuan+3;

  //  suhu = suhu + 1;
  //  if ( suhu >= 65 )
  //    suhu =0;

  if (suhu <=0)
    suhu = 0;
  delay_ms(500);
}

```

```

if ( suhu >= 45)
    kondisi_suhu = 0;
else if ( suhu >= 44)
    kondisi_suhu = 1;
else if ( suhu >= 43)
    kondisi_suhu = 2;
else
    kondisi_suhu = 3;

task_suhu();
old_kondisi_suhu = kondisi_suhu;
}
void task_suhu()
{
    if(old_kondisi_suhu != kondisi_suhu)
    {
        if(kondisi_suhu == 3)
        {
            relay_1 = off;
            relay_2 = off;
            relay_3 = on;
        }
        else if(kondisi_suhu == 2)
        {
            relay_1 = off;
            relay_2 = on;
            relay_3 = off;
        }
        else if(kondisi_suhu == 1)
        {
            relay_1 = on;
            relay_2 = off;

```



```

        relay_3 = off;
    }
    else if(kondisi_suhu == 0)
    {
        relay_1 = off;
        relay_2 = off;
        relay_3 = off;
    }
}
}
////////////////////////////////OKSIGEN //////////////////////////////////
void baca_oksigen()
{
    adc_oksigen = read_adc(2);
    task_oksigen();
}
void task_oksigen()
{
    oksigen = (adc_oksigen * 100) / oks_i_ref ;
    if(oksigen < 0)
        oksigen = oksigen * (-1);

    if(oksigen >= 15)
        oksigen = 15;

    if(oksigen <= 10){
        servo = buka;
        kondisi_servo = 100;
    }
    else if (oksigen <= 15){
        servo = buka_setengah;
        kondisi_servo = 50;
    }
}

```

```

    }
    else {
        servo = tutup;
        kondisi_servo = 0;
    }
}
////////////////////////////////LEVEL////////////////////////////////
void baca_level()
{
    if(level_1==0 && level_2==0){
        new_level=0;
        lcd_gotoxy(6,2);
        lcd_putsf("LOW ");
        level_low = on;
    }
    else if(level_1==1 && level_2==0){
        new_level=50;
        lcd_gotoxy(6,2);
        lcd_putsf("MEDIUM");
        level_low = off;
    }
    else if (level_1==1 && level_2==1){
        new_level=100;
        lcd_gotoxy(6,2);
        lcd_putsf("HIGH ");
        level_low = off;
    }
    task_level();
    old_level=new_level;
}
void task_level()
{

```

```

if(old_level != new_level)
    level = new_level;
}
////////////////////////tampilan data tekanan //////////////////////////
void tampil_lcd()
{
    lcd_gotoxy(0,0);
    lcd_putsf("Tekanan :");

    lcd_gotoxy(8,0);
    sprintf (tampil,"% .2f psi ",psi);
    lcd_puts (tampil);

//  lcd_gotoxy(17,0);
//  sprintf (tampil,"%d ",adc_tekanan);
//  lcd_puts (tampil);
//

////////////////////////tampilan data suhu////////////////////////
    lcd_gotoxy(0,1);
    lcd_putsf("Suhu:");

    lcd_gotoxy(5,1);
    sprintf (tampil,"%d,%d ",suhu,koma);
    lcd_puts (tampil);
    lcd_gotoxy(10,1);
    lcd_putchar(0xDF);
    lcd_gotoxy(11,1);
    lcd_putchar('C');

    lcd_gotoxy(13,1);
    sprintf (tampil,"%d",kondisi_suhu);
    lcd_puts (tampil);

```

```

////////////////////tampilan data oksigen////////////////
lcd_gotoxy(0,3);
lcd_putsf("O2:");

lcd_gotoxy(3,3);
sprintf (tampil,"%d ",oksigen);
lcd_puts (tampil);
lcd_gotoxy(5,3);
lcd_putchar('%');

lcd_gotoxy(7,3);
sprintf (tampil,"%d ",kondisi_servo);
lcd_puts (tampil);
////////////////////tampilan data level////////////////
lcd_gotoxy(0,2);
lcd_putsf("Level:");
////////////////////tampilan data waktu////////////////
lcd_gotoxy(12,2);
sprintf (tampil,"%02d-%02d-%02d",tg,bl,th);
lcd_puts (tampil);
lcd_gotoxy(12,3);
sprintf (tampil,"%02d:%02d:%02d",jm,mn,dt);
lcd_puts (tampil);
}
void get_time()
{
    rtc_get_time(&jm,&mn,&dt);
    rtc_get_date(&ming,&tg,&bl,&th);
}
void tulis_judul()
{
    printf("Date");
}

```

```

    putchar(9);
    printf("Time");
    putchar(9);
    printf("Tekanan");
    putchar(9);
    printf("Suhu");
    putchar(9);
    printf("Level");
    putchar(9);
    printf("Oksigen");
    putchar(10);
    putchar(13);
}
void simpan_data()
{
    printf ("%02d-%02d-%02d",tg,bl,th);
    putchar(9);
    printf ("%02d:%02d:%02d",jm,mn,dt);
    putchar(9);
    printf ("% .2f",bar);
    putchar(9);
    printf ("%d,%d",suhu,koma);
    putchar(9);
    if (new_level == 0)
        printf ("LOW");
    else if (new_level == 50)
        printf ("MEDIUM");
    else if (new_level == 100)
        printf ("HIGH");
    putchar(9);
    printf ("%d",oksigen);
    putchar('% ');
}

```

```

        putchar(10);
        putchar(13);
    }
void main(void)
{
// Declare your local variables here

// Input/Output Ports initialization
// Port A initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In
Func1=In Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T
State1=T State0=T
PORTA=0x03;
DDRA=0x00;

// Port B initialization
// Func7=Out Func6=In Func5=Out Func4=Out Func3=In
Func2=In Func1=In Func0=In
// State7=0 State6=T State5=0 State4=0 State3=T State2=T
State1=T State0=T
PORTB=0x00;
DDRB=0xB0;

// Port C initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In
Func1=In Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T
State1=T State0=T
PORTC=0x00;
DDRC=0x00;

```

```
// Port D initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In
Func1=In Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T
State1=T State0=T
PORTD=0xFC;
DDRD=0xFC;
```

```
// Timer/Counter 0 initialization
// Clock source: System Clock
// Clock value: Timer 0 Stopped
// Mode: Normal top=0xFF
// OC0 output: Disconnected
TCCR0=0x00;
TCNT0=0x00;
OCR0=0x00;
```

```
// Timer/Counter 1 initialization
// Clock source: System Clock
// Clock value: Timer1 Stopped
// Mode: Normal top=0xFFFF
// OC1A output: Discon.
// OC1B output: Discon.
// Noise Canceler: Off
// Input Capture on Falling Edge
// Timer1 Overflow Interrupt: Off
// Input Capture Interrupt: Off
// Compare A Match Interrupt: Off
// Compare B Match Interrupt: Off
TCCR1A=0x00;
TCCR1B=0x00;
TCNT1H=0x00;
```

```
TCNT1L=0x00;
ICR1H=0x00;
ICR1L=0x00;
OCR1AH=0x00;
OCR1AL=0x00;
OCR1BH=0x00;
OCR1BL=0x00;
```

```
// Timer/Counter 2 initialization
// Clock source: System Clock
// Clock value: 125.000 kHz
// Mode: Fast PWM top=0xFF
// OC2 output: Inverted PWM
ASSR=0x00;
TCCR2=0x7C;
TCNT2=0x00;
OCR2=0x00;
```

```
// External Interrupt(s) initialization
// INT0: Off
// INT1: Off
// INT2: Off
MCUCR=0x00;
MCUCSR=0x00;
```

```
// Timer(s)/Counter(s) Interrupt(s) initialization
TIMSK=0x00;
```

```
// USART initialization
// Communication Parameters: 8 Data, 1 Stop, No Parity
// USART Receiver: On
// USART Transmitter: On
```



```
// USART Mode: Asynchronous
// USART Baud Rate: 9600
UCSRA=0x00;
UCSRB=0x18;
UCSRC=0x06;
UBRRH=0x00;
UBRRL=0x33;

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

// ADC initialization
// ADC Clock frequency: 1000.000 kHz
// ADC Voltage Reference: AREF pin
// Only the 8 most significant bits of
// the AD conversion result are used
ADMUX=ADC_VREF_TYPE & 0xff;
ADCSRA=0x83;

// SPI initialization
// SPI Type: Master
// SPI Clock Rate: 2000.000 kHz
// SPI Clock Phase: Cycle Start
// SPI Clock Polarity: Low
// SPI Data Order: MSB First
SPCR=0x50;
SPSR=0x00;

// TWI initialization
```

```

// TWI disabled
TWCR=0x00;

// I2C Bus initialization
// I2C Port: PORTB
// I2C SDA bit: 1
// I2C SCL bit: 0
// Bit Rate: 100 kHz
// Note: I2C settings are specified in the
// Project|Configure|C Compiler|Libraries|I2C menu.
i2c_init();
// DS1307 Real Time Clock initialization
// Square wave output on pin SQW/OUT: Off
// SQW/OUT pin state: 0
rtc_init(0,0,0);

// Alphanumeric LCD initialization
// Connections are specified in the
// Project|Configure|C Compiler|Libraries|Alphanumeric LCD
menu:
// RS - PORTC Bit 7
// RD - PORTC Bit 6
// EN - PORTC Bit 5
// D4 - PORTC Bit 3
// D5 - PORTC Bit 2
// D6 - PORTC Bit 1
// D7 - PORTC Bit 0
// Characters/line: 20
lcd_init(20);

//set time rtc //
jm = 12;

```

```
mn = 37;
dt = 00;
ming = 01;
tg = 8;
bl = 07;
th = 18;
//rtc_set_time(jm,mn,dt);
//rtc_set_date(ming,tg,bl,th);
tulis_judul();
```

```
while (1)
{
    baca_tekanan();
    delay_ms(20);
    baca_oksigen();
    delay_ms(20);
    baca_level();
    delay_ms(20);
    tampil_lcd();
    baca_suhu();
    get_time();
//    lcd_clear();
    if(dt != old_dt)
    {
        simpan_data();
        old_dt = dt;
    }
}
```

BIODATA



Dimaswari Egi Asmiresi yang biasa dipanggil “Egi” ini merupakan mahasiswa dari daerah Nganjuk, Jawa Timur. Penulis menempuh pendidikan formal di SDN PLOSO 1 Nganjuk, SMP Negeri 1 Nganjuk, dan SMAN 2 Nganjuk. Pada tahun 2015 penulis mengikuti test ujian D3 ITS dan diterima pada jurusan D3 Teknik Instrumentasi ITS sekarang ini. Apabila ada pertanyaan mengenai tugas akhir penulis, dapat menghubungi 083854390782 atau email :asmiresidimaswari@gmail.com