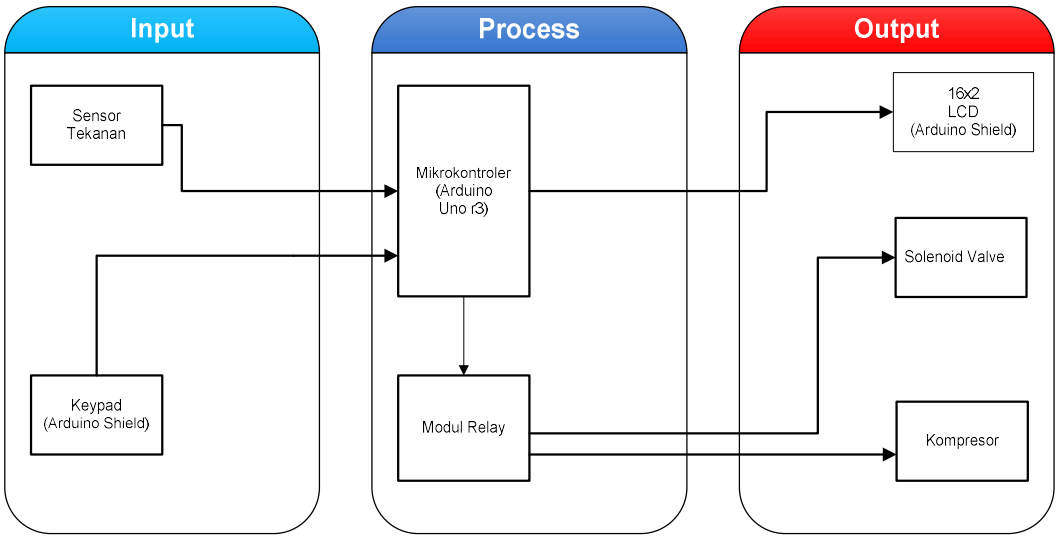
BAB III

PERANCANGAN ALAT

Perancangan merupakan bagian yang penting dari pembuatan alat yang akan dirancang dalam tugas akhir ini. Perancangan sistem dirancang secara terkonsep agar alat yang dibuat lebih baik berdasarkan teori dan referensi yang berkaitan dengan alat yang dirancang, sehingga akan lebih mempermudah dalam pengerjaan baik dalam analisanya.

3.1 Perancangan sistem

Pada perancangan sistem pengisi ban otomatis ini secara umum terdapat 3 bagian utama yaitu, masukan (*input*), proses(*process*) dan keluaran(*Output*). Tiga bagian inilah yang menjadi dasar dari kinerja alat yang penulis rancang.



Gambar 3. 1 Perancangan alat pengisi ban otomatis

Secara garis besar proses kerja dari alat pengisi ban otomatis yang akan dirancang dalam tugas akhir ini adalah ketika alat dinyalakan pengguna harus terlebih dahulu memasukan mulai setpoint melalui *keypad* yang sudah ada kemudian pengguna menghubungkan selang kompresor ke ban dimana pada kondisi ini arduino mulai membandingkan nilai tekanan ban dengan nilai setpoint yang disetting dan selama berlangsung proses pengisian, Arduino akan selalu membandingkan nilai tekanan dengan nilai setpoint hingga mendekati nilai setpoint atau sesuai harapan. Apabila nilai setpoint telah tercapai, maka solenoid valve akan tertutup dan kompresor akan mati secara otomatis.

3.1.1 Fungsi masing-masing Blok

Berikut urutan fungsi dari masing-masing bagian utama blok diagram sistem alat pengisi ban otomatis ini.

1. Masukan (Input)

   Dalam bagian Input ini terdapat dua komponen yang menjadi sumber masukan untuk mikrokontroler, diantaranya:

   a. Keypad

   Keypad disini berfungsi sebagai tempat memasukan nilai setpoint yang sudah menjadi satu dengan LCD 16x2 modul ini dinamakan dengan Arduino LCD with *keypad* Shield, model kerja dari *keypad* jenis ini menggunakan sistem kerja ADC (*Analog to Digital Converter*) kelebihan sistem ini adalah tidak perlu menggunakan banyak port sehingga lebih efisien dalam penggunaan *port*-nya, terlebih jika Arduino UNO yang biasanya tidak menyediakan terlalu banyak port I/O. Namun sistem ini juga memiliki

kelemahan yakni apabila *supply* tidak stabil maka pembacaan nilai ADC-nya juga akan terganggu sehingga efek *bounching* akan sering terjadi.

b. Sensor tekanan

Sensor tekanan berfungsi untuk mendeteksi tekanan dari ban dan keluaran dari sensor ini akan diproses di Arduino, sensor tekanan ini mengubah besaran fisik yang dalam hal ini adalah tekanan udara yang di konversi menjadi besaran listrik.

2. Proses

Mikrokontroler memproses data masukan dan mengeksekusi perangkat output sesuai dengan instruksi program yang telah dibuat oleh perancang. Pada bagian ini mikrokontroller Arduino Uno digunakan sebagai pusat memproses data.

3. Keluaran (*Output*)

Bagian keluaran atau output adalah bagian yang merupakan hasil dari eksekusi perangkat dan bertindak sebagai hasil dari kinerja perangkat sesuai dengan keinginan perancang. Terdapat tiga macam jenis perangkat yang digunakan pada perancangan alat ini, diantaranya.

a. *Liquid Crystal Display* (LCD)

Perangkat penampil yang digunakan alam perancangan pengisi ban otomatis ini adalah LCD 16x2 yang sudah terintegrasi dengan *keypad*. Perangkat ini berfungsi sebagai penampil tekanan saat ini, nilai setpoint dan status pengisian alat.

b. *Solenoid Valve*

Solenoid valve digunakan sebagai aktuator yang membuka saluran udara dari kompresor ke ban. *Solenoid valve* berfungsi untuk mengurangi tekanan angin ban. *Solenoid valve* ini terhubung dengan *driver* karena sebuah mikrokontroler tidak mungkin bisa mengendalikan sebuah *solenoid valve* secara langsung karena kebutuhan tegangan dan arus solenoid valve ini yang cukup besar.

c. Kompresor

Kompresor adalah sebuah alat yang berfungsi untuk meng-kompres udara sehingga dapat masuk ke dalam ban, dalam alat yang penulis rancang kompresor yang digunakan adalah kompresor mini yang menggunakan tegangan 12 Volt DC, kompresor ini juga menggunakan sebuah driver sehingga dapat dikendalikan oleh Arduino.

d. Modul Relay

Modul relay di buat sebagai penguat arus untuk mengendalikan *solenoid valve* dan mengendalikan kompresor.

3.1.2 Cara Kerja Sistem

Sistem kerja dari alat yang penulis rancang adalah pada saat alat dinyalakan pertama kalinya, pengguna akan memasukan nilai tekanan ban (*setpoint*) terlebih dahulu pada modul *interface*, setelah nilai tekanan ban sudah dimasukkan, maka langkah selanjutnya adalah menghubungkan selang kompresor ke ban. Pada saat itu Arduino akan mulai membandingkan nilai setpoint dengan tekanan ban saat ini, jika tekanan angin dalam ban dibawah nilai setpoint, maka Arduino akan

mengirimkan sinyal untuk menyalakan kompresor yang berfungsi untuk memompa udara ke ban hingga tekanan udara pada ban sama atau mendekati nilai setpoint, apabila nilai tekanan sama atau mendekati nilai setpoint, maka kompresor akan berhenti beroperasi. Namun jika nilai tekanan yang terbaca berada diatas nilai setpoint, maka Arduino akan mengirimkan sinyal ke *solenoid valve* yang berfungsi untuk mengurangi tekanan udara dalam ban ke udara luar. Begitu pula pada saat proses pengisian apabila terjadi kelebihan tekanan, maka mikrokontroler Arduino akan mengirimkan sinyal ke *solenoid valve* untuk segera mengembalikan tekanan udara ke daerah setpoint.

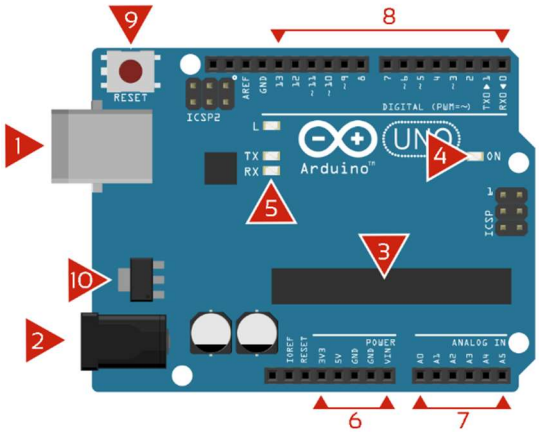
3.2 Pemilihan Komponen

Pemilihan jenis komponen yang digunakan dalam perancangan perangkat elektronik dilakukan bertujuan untuk menentukan kualitas bahan, desain alat dan biaya yang akan digunakan karena akan berdampak langsung pada tingkat efisiensi dan efektivitas perangkat yang akan dibuat.

3.2.1 Pemilihan Jenis Mikrokontroler

Jenis mikrokontroler yang digunakan pada perancangan alat pengisi ban otomatis harus berdasarkan kriteria sistem yang akan dibuat dalam hal ini penulis menggunakan mikrokontroler Arduino dengan jenis Arduino UNO R3.

Mikrokontroler jenis ini selain harganya terjangkau, pengaplikasian mikrokontroler jenis ini juga terbilang sangat luas, hampir penghobi elektronik yang berbasis pada *embeded system* menggunakan mikrokontroler jenis ini, selain itu pula pengisian program dari PC ke Arduino juga sangat mudah karena tinggal menghubungkan Arduino ke PC via kabel USB maka pemograman bisa dilakukan. Berikut adalah Bagian –Bagian Arduino:



Gambar 3. 2 Bagian bagian Arduino[3]

Keterangan dari papan Arduino adalah sebagai berikut:

1. Port USB

   Berfungsi untuk memasok sumber daya serta mengunggah kode ke Arduino. Tipe USB yang digunakan Adalah tipe B standar dan ujung satunya menggunakan USB tipe A standar.

2. Power Jack

   Selain mengambil *power* dari USB, Arduino juga memiliki Power Jack untuk memasok daya dari baterai 9 Volt maupun menggunakan adaptor. Perhatikan bahwa Arduino memiliki batas tegangan antara 6-20 Volt dengan rentang yang disarankan yaitu 7 hingga 12 volt.

3. Chip Utama/Mikrokontroler

   Chip inilah yang menjadi otak dari Arduino. Chip yang digunakan biasanya ATMega yang merupakan buatan perusahaan ATMEL. Jenis chip yang digunakan berbeda-beda di beberapa jenis papan Arduino.

4. LED Indikator Daya

   LED yang berlabel ‘ON’ ini akan menyala ketika Arduino sedang beroperasi.

5. LED indikator Transfer Data

   Terdapat dua LED, yaitu TX(*Transmit*) dan RX(*Receive*). Kedua LED ini akan berkedip untuk memberitahukan telah terjadi komunikasi serial berupa pengiriman atau penerimaan data.

6. Pin Sumber Daya

   Pin memiliki fungsi yang vital, karena pin inilah yang menjadi *interface* dengan perangkat lainnya, seperti sensor atau aktuator. Digunakan kabel (*jumper*) untuk menghubungkan antara Arduino dan perangkat lain.

7. Pin Analog

   Semua pin analog hanya bersifat input dan digunakan untuk menerima sinyal analog dari, misalnya, sensor suhu. Sinyal analog yang telah diterima Arduino terlebih dahulu di ubah ke digital sebelum diproses lebih lanjut.

8. Pin Digital

   Pada Arduino UNO, terdapat 14 pin digital yang digunakan untuk input/output dari atau ke perangkat digital. Contohnya adalah input dari sensor gerak atau output ke lampu LED. Dua pin pertama (pin ke-0 dan 1) merupakan pin RXD dan TXD yang berfungsi sebagai *receive-transmit* digital.

9. Tombol Reset

Ketika tombol ini ditekan selama beberapa detik, maka Arduino akan mengulang program dari awal. Ini kita lakukan jika tidak memiliki fungsi perulangan pada kode yang telah diunggah ke Arduino atau jika kesalahan terjadi.

10. Regulator Tegangan

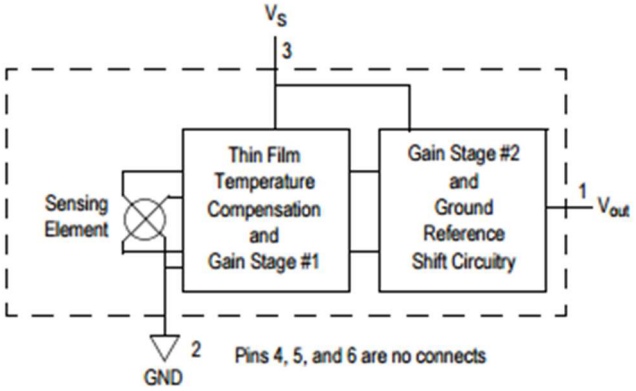
Regulator tegangan berfungsi sebagai penjaga gerbang yang berarti ia akan mengalirkan tegangan ter-regulasi ke Arduino dan membuang tegangan berlebih. Berikut ini adalah spesifikasi Arduino Uno:

Tabel 3. 1 spesifikasi Arduino UNO[9]

mikrokontroler	ATmega328
input tegangan min.	6 volt
input tegangan Maks.	20 volt
input tegangan yang disarankan	7-11 volt
pin I/O digital	14(6 bisa untuk PWM)
pin analog	6
arus tiap pin	50 mA
memori flash	32KB(ATmega328)
SRAM	2KB(ATmega328)
EEPROM	1KB(ATmega328)
Kecepatan clock	16 MHz

3.2.2 Sensor Tekanan

Sensor tekanan yang digunakan dalam perancangan ini adalah sensor tekanan semikonduktor dengan tipe MPX5700AP yang dimana sensor ini merupakan sensor yang terintegrasi dengan penguat sinyal, sehingga kita tidak perlu lagi menambahkan pengkondisi sinyal, untuk membaca keluaran dari sensor ini adalah kita hanya perlu mengkonversi sinyal ADC ke satuan Volt. Berikut adalah skema dari sensor tersebut.



Gambar 3. 3 skema rangkaian internal MPX5700AP[9]

Pengkonversian sinyal ADC ke satuan Volt dinyatakan dalam persamaan (3.1) berikut ini[7].

V_{out}= sensor value* (4,5/1023)..... (3.1)

Dimana Sensor value menunjukkan nilai ADC yang terbaca oleh Arduino dan 4,5 menunjukkan tegangan output maksimum sensor (Volt) serta 1023 menunjukkan ADC pada Arduino adalah 10 bit.

Selain itu juga sensor MPX5700AP ini memiliki spesifikasi sebagai berikut.

Tabel 3. 2 spesifikasi MPX5700AP[9]

Characteristic		Symbol	Min	Typ	Max	Unit
Pressure Range(1)	Gauge, Differential: MPX5700D Absolute: MPX5700A	P_{OP}	0 15	— —	700 700	kPa
Supply Voltage(2)		V_S	4.75	5.0	5.25	Vdc
Supply Current		I_O	—	7.0	10	mAdc
Zero Pressure Offset(3)	Gauge, Differential (0 to 85°C) Absolute (0 to 85°C)	V_{off}	0.088 0.184	0.2 —	0.313 0.409	Vdc
Full Scale Output(4)	(0 to 85°C)	V_{FSO}	4.587	4.7	4.813	Vdc
Full Scale Span(5)	(0 to 85°C)	V_{FSS}	—	4.5	—	Vdc
Accuracy(6)	(0 to 85°C)	—	—	—	±2.5	$\%V_{FSS}$
Sensitivity		V/P	—	6.4	—	mV/kPa
Response Time(7)		t_R	—	1.0	—	ms
Output Source Current at Full Scale Output		I_{O+}	—	0.1	—	mAdc
Warm-Up Time(8)		—	—	20	—	ms

Dari tabel diatas maka dapat kita simpulkan bahwa sensor tekanan jenis ini memiliki kemampuan membaca tekanan dari 15 hingga 700 kPa (2,175-101,5 Psi), dengan output maksimal 4,7 Volt pada tegangan supply 5 Volt. Selain itu juga, sensor ini memiliki tingkat akurasi ±2,5% pada tegangan outputnya dengan sensitivitas 6,4mV/kPa. Dengan spesifikasi tersebut, maka sensor tekanan model ini sudah memenuhi syarat untuk dapat digunakan dalam perancangan alat dalam penelitian ini.

3.2.3 Modul *Interface*

Interface merupakan sebuah perangkat yang menghubungkan suatu peralatan dengan penggunanya. Dalam alat yang penulis rancang, *interface* ini berupa layar LCD 16x2 yang sudah terintegrasi dengan papan tombol yang berfungsi sebagai penampil tekanan ban dan sebagai tempat untuk memasukan nilai setpoint. Pada pemilihan *interface* ini penulis menggunakan Arduino LCD with *keypad* shield, berikut ini adalah alasan mengapa penulis menggunakan modul ini:

1. Pemasangan mudah, karena hanya tinggal memasangkan diatas Arduino.
2. Port yang dipakai tidak banyak, sehingga port pada Arduino UNO yang tersisa masih bisa digunakan untuk keperluan lain sudah terdapat *keypad*, *keypad* ini pun hanya memakai satu port analog saja. Berikut adalah kegunaan dari masing-masing pin.

Tabel 3. 3 Fungsi Masing-masing Pin[7]

Pin	Fungsi
Analog 0	Tombol (*select, up, right, down dan left*)
Digital 4	DB0

Digital 5	DB1
Digital 6	DB2
Digital 7	DB3
Digital 8	RS(*Data or Signal Display Selection*)
Digital 9	Enable
Digital 10	Pengontrol cahaya LCD

Bentuk fisik dari Arduino Keypad Shield ditunjukkan pada gambar berikut ini.



Gambar 3. 4 Arduino LCD keypad shield

3.2.4 Kompresor

Kompresor adalah suatu alat yang biasa digunakan untuk meng-kompres/ memampatkan suatu gas dan disimpan dalam suatu tabung sebagai media penyimpanan tekanannya, kompresor terbagi tiga bagian yaitu, motor penggerak, pompa hisap dan tabung penyimpanan gas. Dalam perancangan alat yang penulis lakukan, penulis menggunakan kompresor yang berukuran kecil dan menggunakan tegangan 12 Volt DC, sehingga kompresor ini dapat menggunakan aki motor atau mobil sebagai sumber tenaganya. Sehingga dapat menunjang tingkat kepraktisan dari alat yang penulis rancang. Berikut ini adalah Gambar dari Kompresor mini.



Gambar 3. 5 Kompresor Mini

Kompresor ini memiliki spesifikasi sebagai berikut:

Tabel 3. 4 Spesifikasi Kompresor *mini*[13]

Merk	Coido
Model	6218
Tegangan Kerja	12-14 Volt
Daya maksimal	180 Watt
Arus Maksimal	12 Ampere
Tekanan Maksimal	150 Psi
Volume Udara	30 Liter per Menit (@ 0 Psi)
Durasi Pemakaian	20 Menit Max

3.2.5 Solenoid Valve

Solenoid valve merupakan katup yang dikendalikan oleh arus listrik AC maupun DC melalui kumparan. Solenoid valve ini merupakan elemen kontrol yang paling sering digunakan dalam sistem fluida ataupun gas. Seperti pada sistem pneumatik. Sistem hidrolik ataupun pada sistem kontrol mesin yang membutuhkan elemen kontrol otomatis. Pada perancangan ini penulis menggunakan solenoid valve yang tegangan kerja sebesar 12 volt dengan arus optimal 800mA, solenoid

valve ini memiliki batas tekanan maksimum sebesar 10bar. Berikut ini adalah gambar dari solenoid valve:



Gambar 3. 6 Solenoid Valve

3.3 Perancangan alat

Perancangan merupakan proses merancang atau mendesain suatu sistem yang akan dibuat serta di implementasikan, pada perancangan ini akan membahas mengenai proses perakitan alat dan implementasi dari komponen komponen yang dipakai. Dalam perancangan ini terbagi menjadi 2 proses yaitu, perancangan *hardware* dan perancangan *software*.

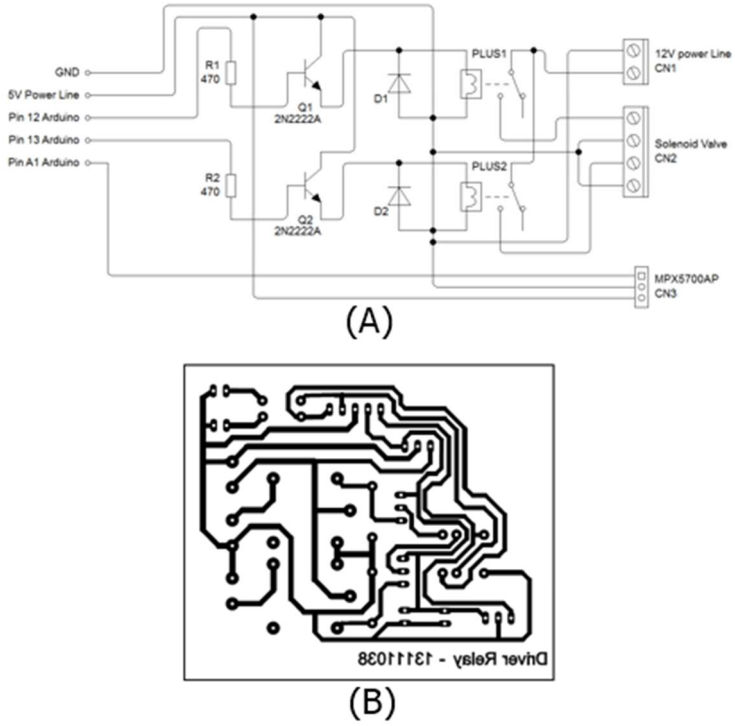
3.3.1 Perancangan Hardware

Pada perancangan hardware ini meliputi pembahasan mengenai, *microcontroller*, sensor, aktuator dan *interface*.

3.3.1.1 Modul Relay

Modul Relay berfungsi sebagai penguat sinyal keluaran dari Arduino Uno, seperti yang kita ketahui sinyal keluaran arduino sebesar 5 Volt dengan arus tidak lebih dari 50mA sedangkan sebuah *solenoid valve* membutuhkan tegangan 12 Volt dengan arus minimal sebesar 900mA sehingga untuk dapat bekerja dengan

semestinya, *solenoid valve* harus di kendalikan dengan menggunakan sebuah *driver*. Dalam pembuatan skema dan *layout* driver ini, penulis menggunakan program Livewire dan PCB Wizard, program ini digunakan karena kemudahan dan kepraktisan dalam penggunaannya. Berikut adalah skema dari modul relay tersebut:



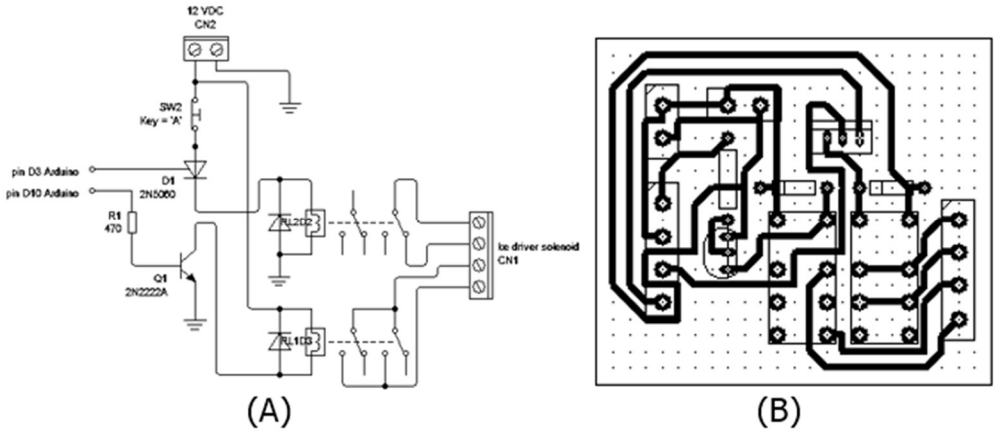
Gambar 3. 7 Skema(A) dan Layout(B) Rangkaian modul Relay

Pada modul ini terdapat dua buah komponen utama, yaitu Relay dengan Transistor yang dimana dua komponen ini saling berhubungan satu sama lainnya. Transistor disini berfungsi untuk menguatkan arus dari Arduino karena arus yang dikeluarkan oleh Arduino hanya sebesar 40mA saja, sehingga kurang kuat untuk mengendalikan sebuah relay secara langsung, dimana relay ini memerlukan setidaknya 50mA untuk dapat menarik armature yang berada di dalam relay tersebut. Sedangkan relay disini berfungsi untuk menguatkan arus yang mengalir

ke solenoid valve dimana sebuah solenoid valve membutuhkan arus sebesar 2A untuk bisa membuka katup magnetiknya.

3.3.1.2 Modul kontrol tegangan

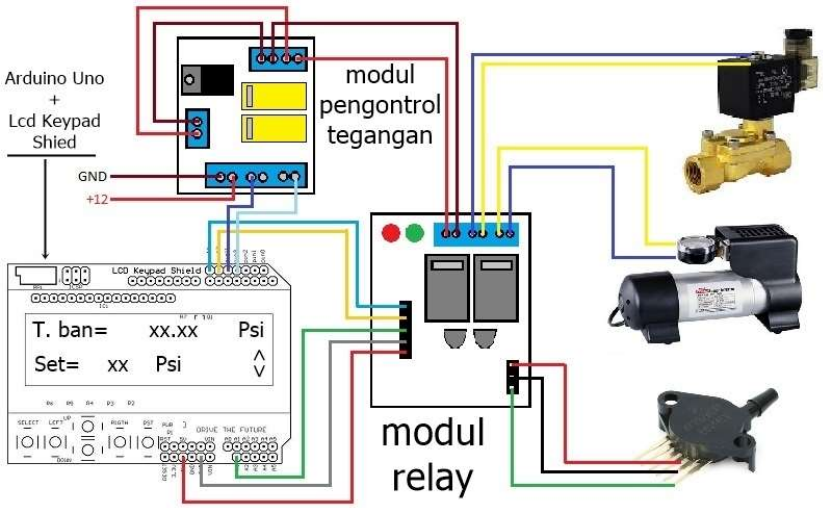
Modul ini berperan sebagai pengaman agar kompresor dan *solenoid valve* tidak menyala tanpa beban. Modul ini juga berperan sebagai *switch* otomatis yang memutuskan aliran listrik pada saat nilai tekanan ban tidak terdeteksi. Dalam pembuatan skema dan *layout* driver ini, penulis menggunakan program Livewire dan PCB Wizard, program ini digunakan karena kemudahan dan kepraktisan dalam penggunaannya Berikut adalah skema dan layout dari rangkaian driver kompresor tersebut.



Gambar 3. 8 Skema(A) dan layout(B) modul pengontrol tegangan

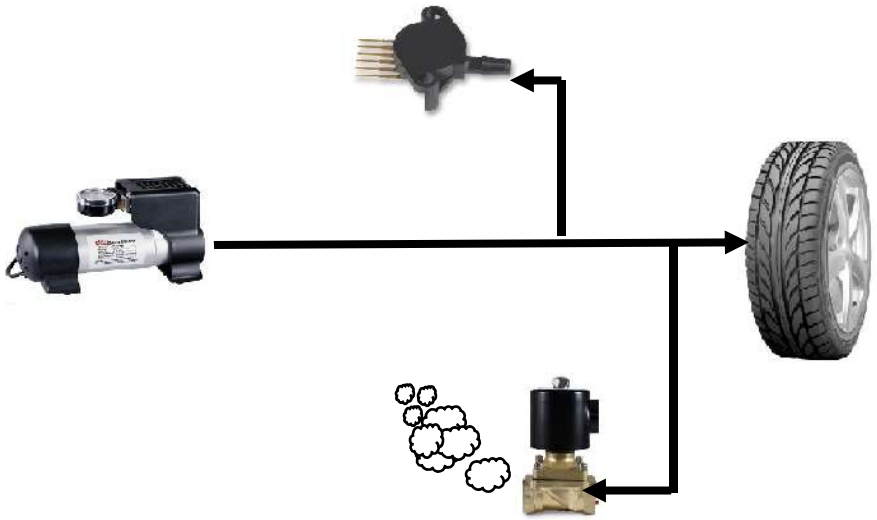
Pada modul ini terdapat tiga komponen utama yaitu relay, transistor dan sebuah SCR. Dimana transistor terhubung dengan relay 1 yang berfungsi menghubungkan dan memutuskan tegangan dari sumber tegangan ke modul Relay, relay 1 ini terhubung secara *Normally Open*, sehingga pada saat kaki basis transistor ini mendapatkan logika 1 (*High*), maka arus mengalir ke kumparan pada relay, sehingga akan menghubungkan sumber tegangan dengan driver solenoid. Sedangkan SCR terhubung dengan relay 2 yang yang memiliki fungsi yang sama

dengan relay 1 hanya saja relay ini terhubung secara *Normally Close* sehingga pada saat relay di aliri arus listrik, maka akan memutuskan sumber tegangan ke driver solenoid. SCR digunakan karena prinsip kerjanya yang apabila pada bagian kaki *Gate* diberi tegangan maka arus akan tetap mengalir hingga SCR diputuskan dari sumber tegangannya (*Reset*). Berikut adalah jalur kabel elektrik pada alat yang penulis rancang.



Gambar 3. 9 Skema Jalur kabel elektrik

Adapun skema jalur aliran udaranya ditujukan pada gambar dibawah ini.



Gambar 3. 10 skema jalur aliran udara

Berikut adalah gambar dari kedua modul yang sudah terpasang.



Gambar 3. 11 modul terpasang

3.3.2 Perancangan Software

Perancangan software bertujuan supaya alat yang sudah dirakit dan terhubung satu sama lain dapat bekerja seperti yang kita harapkan, semua pemrosesan data input dan output akan terjadi pada perancangan software ini. Pada perancangan software ini akan terdapat dua bagian, yaitu software untuk pembacaan keypad dan perancangan sistem kerja keseluruhan

3.3.2.1 Perancangan Software Untuk Pembacaan *Keypad*

Arduino LCD *Keypad* Shield digunakan sebagai *interface* menggunakan metode pembagian tegangan sebagai output dari *keypad*nya, sehingga pada bagian arduino sebagai pemroses datanya akan membaca nilai ADC sebagai pendeteksi tombol yang ditekan.

Pada alat yang penulis rancang, berikut adalah nilai ADC yang digunakan untuk pendeteksian tombol-tombol *keypad*nya.

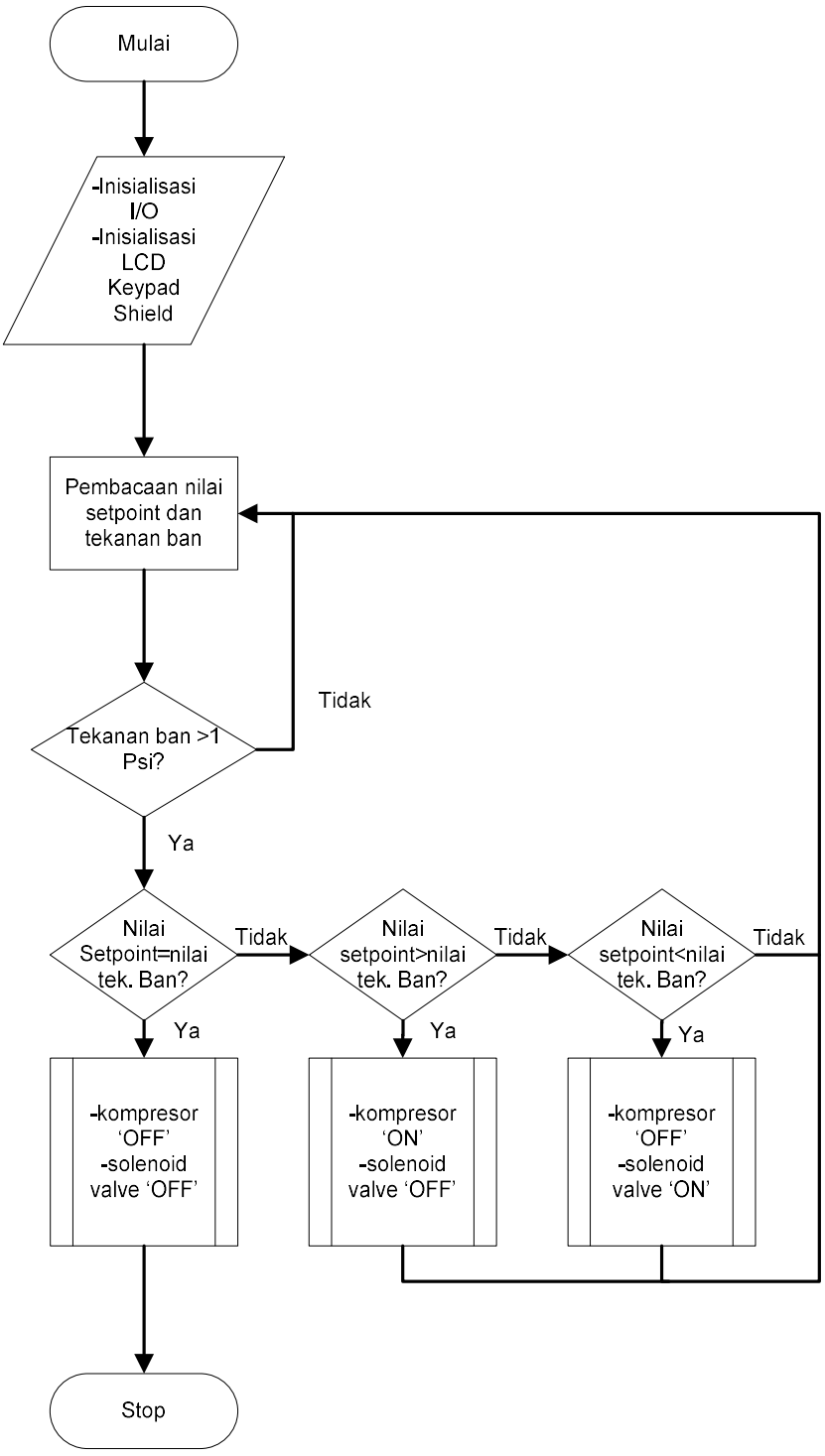
Tabel 3. 5 Fungsi tombol dan nilai ADC yang digunakan

Tombol Pada Shield	Nilai ADC	Kegunaan
Left	< 555	Setpoint -10
Up	< 150	Setpoint +1
Down	< 385	Setpoint -1
Right	< 50	Setpoint +10

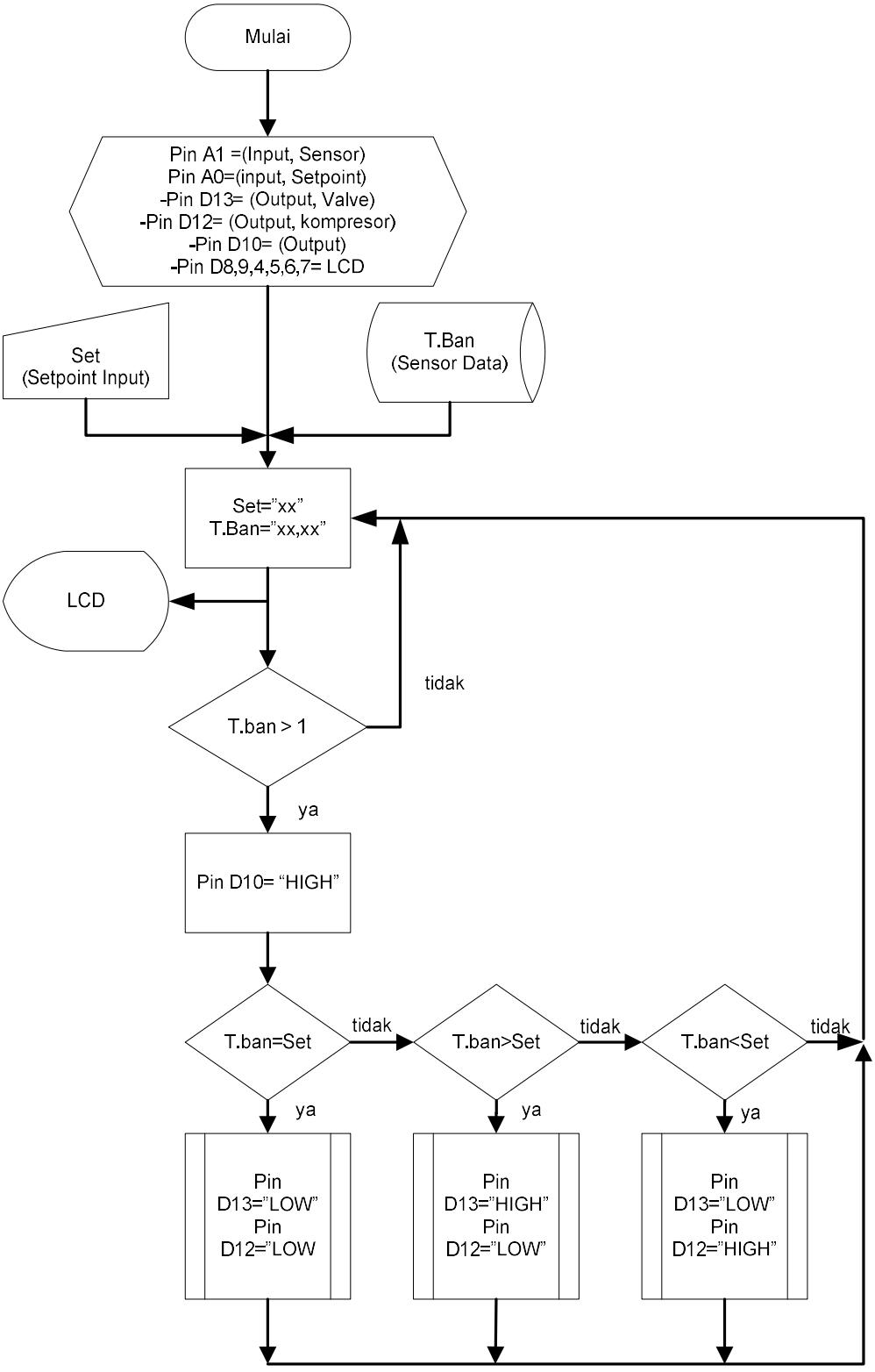
Berdasarkan tabel diatas dapat diketahui bahwa dengan adanya perbedaan nilai ADC, maka Arduino akan dapat mendeteksi tombol mana yang ditekan. Hasil dari pendeteksian tombol akan ditampilkan pada panel LCD dan menujukan nilai setpoint yang di inginkan oleh pengguna.

3.3.2.2 Perancangan Software Untuk Proses Kerja Keseluruhan

Perancangan pada bagian ini akan dijelaskan bagaimana alur kerja dari keseluruhan proses pada alat yang penulis rancang. Mulai dari pembacaan tekanan, pembacaan nilai setpoint, komparasi data tekanan dengan nilai setpoint dan proses eksekusi pada bagian keluaran. Berikut adalah alur kerja sistem (flowchart) dari alat yang penulis rancang.



Gambar 3. 12 diagram alir kerja sistem



Gambar 3. 13 Algoritma program

Pada **Gambar 3.12** merupakan alur kerja dari keseluruhan sistem pada alat yang penulis rancang. Ketika pada saat alat dinyalakan Arduino Uno akan memulai proses inisialisasi dan proses pembacaan tekanan ban dengan nilai setpoint yang telah diinputkan sebelumnya yang kemudian akan di komparasi antara keduanya. Pada **Gambar 3.12**, penulis menggunakan perintah "Tekanan ban >1 Psi?" pada awal proses, hal ini dikarenakan supaya tidak terjadi kesalahan yang membuat kompresor menyala sebelum waktunya. Atau dengan kata lain perintah tersebut dimaksudkan untuk pengaman dari alat itu sendiri. Sehingga alat yang penulis rancang hanya dapat beroperasi apabila selang sudah terhubung dengan ban. Kemudian apabila sudah terhubung dengan ban kendaraan maka proses selanjutnya dapat di eksekusi., proses selanjutnya adalah proses pembandingan nilai setpoint dengan tekanan ban. Pada **Gambar 3.12** diatas alat akan mendeteksi apakah tekanan ban sudah setara dengan setpoint, apabila sudah tercapai, maka arduino Uno akan membuat kompresor dan solenoid valve berhenti beroperasi. Namun apabila belum, maka arduino akan membandingkan kembali apakah tekanan ban lebih tinggi atau lebih rendah dari setpoint. Setiap proses pengisian atau pengurangan tekanan ban akan berlangsung kira-kira satu detik sebelum hasilnya akan dibandingkan kembali oleh Arduino Uno. Proses itu akan terus berulang hingga nilai tekanan ban setara atau mendekati nilai setpoint.