

BAB III

PERANCANGAN DAN PEMBUATAN ALAT

A. Identifikasi Kebutuhan

Untuk menunjang sistem alat ini dilakukan identifikasi komponen, alat dan bahan yang dibutuhkan:

1. Voltage Regulator
2. Mikrokontroler
3. Downloader
4. Printed Circuit Board (PCB)
5. Connector ECU
6. Komponen Electronika
7. Pengkondisi Sinyal
8. Cover

B. Analisis Kebutuhan

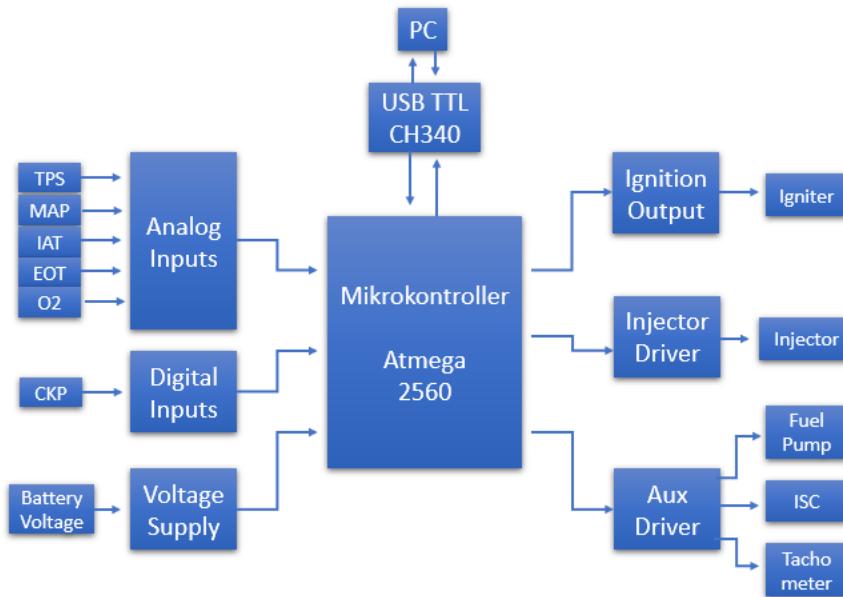
Berdasarkan dari identifikasi kebutuhan di atas maka diperoleh beberapa analisis kebutuhan terhadap pengembangan alat yang akan dibuat sebagai berikut:

1. Rangkaian *power supply* sebagai sumber tegangan untuk seluruh rangkaian sebesar 5V, serta 3,3V untuk sumber mikrokontroler.
2. Rangkaian sistem minimum menggunakan ATmega2560 sebagai pengontrol seluruh kinerja dari *ecu programmable*.
3. Rangkaian pengkondisi sinyal *crank position sensor*.
4. Rangkaian input sensor analog sebagai pengkondisi sinyal analog.
5. Rangkaian pada sistem *driver actuator*.
6. Penambahan *connector ecu* Yamaha 33 pin, untuk menghubungkan antara *ecu programmable* dengan kabel *wiring* pada sensor maupun aktuator.
7. Penambahan cover *ecu programmable* untuk melindungi seluruh rangkaian agar tidak mudah rusak.

C. Perancangan

1. Perancangan Blok Diagram Rangkaian

Perancangan blok diagram rangkaian berdasarkan analisis kebutuhan yang diperlukan alat. Susunan blok diagram rangkaian dapat dilihat pada Gambar 27.



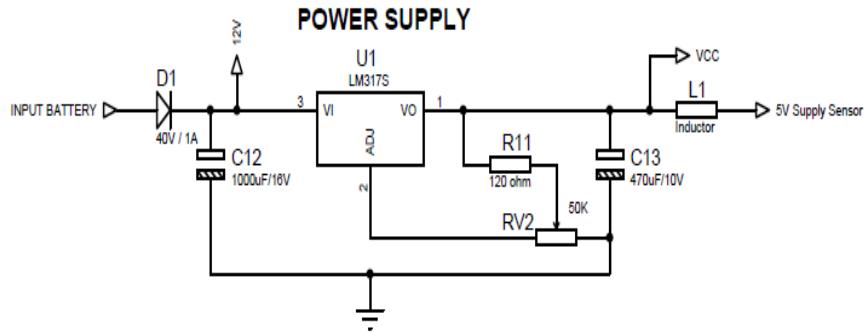
Gambar 27. Blok Diagram Perancangan Rangkaian

Keterangan blok diagram perancangan rangkaian ecu programmble pada Gambar 27, sebagai berikut:

- a. Input TPS (Throttel Position Sensor) berfungsi mendeteksi perubahan posisi dari throttle gas, dan kemudian merubah sumber *voltage 5 volt* menjadi sinyal analog.
- b. Input MAP (Manifold Absolute Pressure) berfungsi mendeteksi tekanan pada *intake manifold*, dan kemudian merubah sumber *voltage 5 volt* menjadi sinyal analog.
- c. Input IAT (Intake Air Temperature) berfungsi mendeteksi suhu udara masuk pada intake manifold, kemudian merubah sumber *voltage 5 volt* menjadi sinyal analog.

- d. Input EOT (Engine Oil Temperature) berfungsi mendeteksi suhu pada oli mesin, kemudian merubah sumber *voltage 5 volt* menjadi sinyal analog.
 - e. Input O2 sensor (Oxygen Sensor) berfungsi mendeteksi hasil pembakaran pada *exhaust manifold*, yang kemudian memberikan *feedback* pada mikrokontroller untuk mengontrol perbandingan campuran bahan bakar dengan udara.
 - f. Analog input berfungsi mengkondisikan sinyal analog dari sensor untuk masukan pada mikrokontroller.
 - g. Input CKP (Crank Position Sensor) berfungsi mendeteksi sudut crankshaft serta mengukur kecepatan mesin.
 - h. Digital Input adalah rangkaian pengkondisi sinyal dari CKP, untuk masukan ke mikrokontroller dalam bentuk sinyal kotak (*square wave*).
 - i. Voltage Supplay (*power supply*) adalah rangkaian voltage regulator yang merubah tegangan 12 *volt* menjadi 5 *volt*, kemudian sebagai sumber tegangan serta arus listrik untuk mengaktifkan seluruh komponen pada *ecu programmable*.
 - j. USB TTL CH340 berfungsi sebagai konverter komunikasi serial mikrokontroler dengan USB komputer.
2. Perancangan Voltage Supplay

Rangkaian *power supply* merupakan rangkaian yang digunakan sebagai sumber listrik pada setiap perangkat elektronika yang terdiri dari voltage regulator, dioda, inductor, dan capasitor. Secara keseluruhan rangkaian *power supply* berfungsi untuk memberikan tegangan yang telah diturunkan oleh *voltage regulator* dari +12V menjadi 5V, dan menstabilkan tegangan DC.



Gambar 28. Rangkaian Power Supply

Gambar 28 menunjukkan rangkaian *power supply* dengan *voltage regulator* LM317. Rangkaian *power supply* mengeluarkan *Output* 5V DC, sebagai sumber tegangan untuk sistem ATmega 2560 *robot dyn promini* dan untuk supply sensor 5V. Besarnya tegangan *Output* pada *voltage regulator* dengan IC LM317 diatas dapat dihitung dengan rumus berikut.

$$V_{out} = V_{ref}(1 + R_2/R_1) + I_{adj}R_2$$

Di sirkuit ini, nilai Vref adalah tegangan referensi antara terminal penyesuaian dan *Output* diambil sebagai 1,25 Volt. Nilai Iadj akan sangat kecil dan juga akan memiliki nilai konstan (ditiadakan/dianggap tidak ada). Dengan demikian persamaan di atas dapat dituliskembali sebagai berikut.

$$V_{out} = 1.25V(1 + R_2/R_1)$$

LM317 Voltage Regulator

R1 (Ohms)	R2 (Ohms)	R1 & R2 (Ohms)	LM 317 (Voltage Out)
120	1000	107.5	11.67
120	950	106.5	11.15
120	900	105.9	10.63
120	850	105.2	10.10
120	800	104.3	9.58
120	750	103.4	9.06
120	700	102.4	8.54
120	650	101.3	8.02
120	600	100.0	7.50
120	550	98.5	6.98
120	500	96.8	6.46
120	450	94.7	5.94
120	400	92.3	5.42
120	350	89.4	4.90
120	300	85.7	4.38
120	250	81.1	3.85
120	200	75.0	3.33
120	150	65.7	2.81
120	100	54.5	2.29
120	50	35.3	1.77
120	1	1	1.26

Gambar 29. Tabel *resistance adjustable* LM137

(<https://biohazardbrewery.wordpress.com/instruments/brewery-stirplate/>)

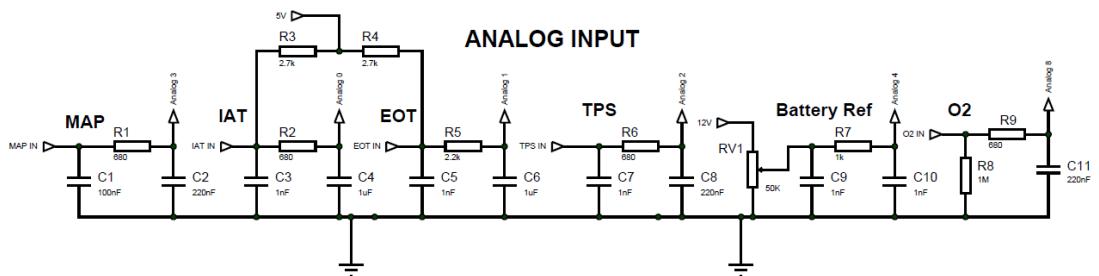
Untuk mencari nilai tegangan keluar sebesar 5V, maka dapat melihat pada tabel gambar 14 untuk menetapkan nilai tahanan pada R1 dan R2.

$$\begin{aligned}
 V_{out} &= 1.25V(1 + R_2/R_1) \\
 &= 1.25V(1 + 360/120) \\
 &= 1.25V(4) \\
 &= 5V
 \end{aligned}$$

Dengan perhitungan tersebut, untuk mencari tegangan 5V maka R1 senilai 120Ω dan R2 senilai 360Ω . Akan tetapi pada *ecu programmable* ini, R2 diganti dengan Variabel Resistor. Varibel Resistor digunakan untuk mencari kepresision nilai tahanan untuk menghasilakan tegangan keluar yang sangat presisi.

3. Perancangan Rangkaian Analog Input

Rangakaian analog input berfungsi untuk mengkondisikan serta memfilter sinyal *Output* dari sensor, agar tegangan analog yang masuk ke pin analog mikrokontroller menjadi stabil.



Gambar 30. Rangkaian Analog Input

Rangkaian analog input mempunyai beberapa rangkaian sistem sebagai berikut :

a. Rangkaian analog input untuk sensor MAP

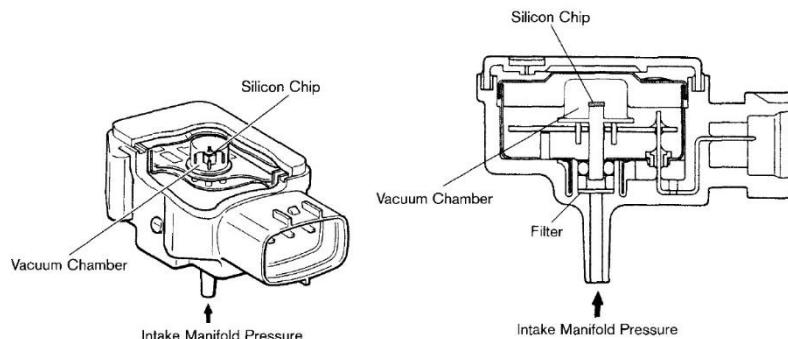
MAP (Manifold Absolute Pressure) berfungsi mendeteksi tekanan pada *intake manifold*. Pada map sensor terdiri dari dua ruangan yang dipisahkan oleh membran tipis dengan elemen semikonduktor terintegrasi pada membran.



Gambar 31. Manifold Absolute Pressure

(<https://www.efi-parts.co.uk/index.php?productID=120>)

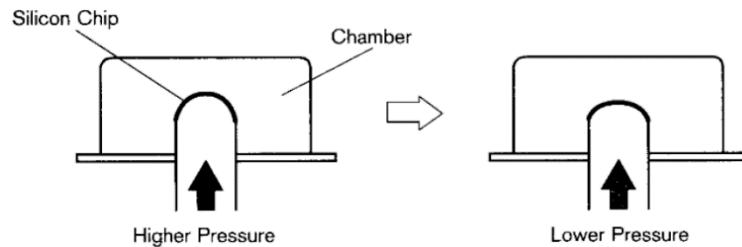
Ketika terjadi perubahan tekanan pada saluran vakum sensor maka membran akan terangkat, hal ini menjadikan perubahan tahanan dari elemen semikonduktor yang telah diintegrasikan pada membran.



Gambar 32. Elemen semikonduktor pada map sensor

(<http://kit2011.blogspot.com/2011/12/manifold-absolute-pressure-map-sensor.html>)

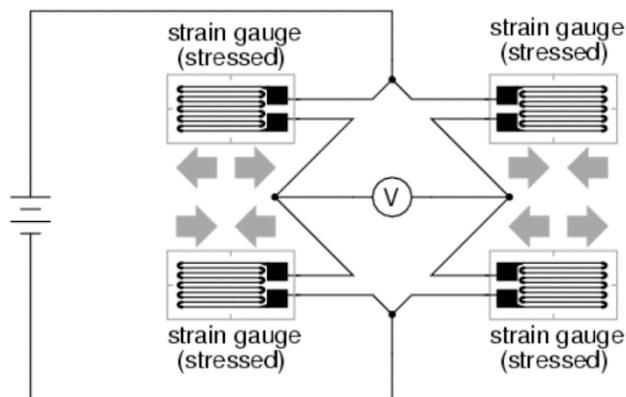
Ketika tekanan meningkat di dalam intake manifold, chip silikon melentur, dan bertindak seperti pengukur regangan dan mengubah tahanannya (*resistance*).



Gambar 33. Chip Silicon map sensor

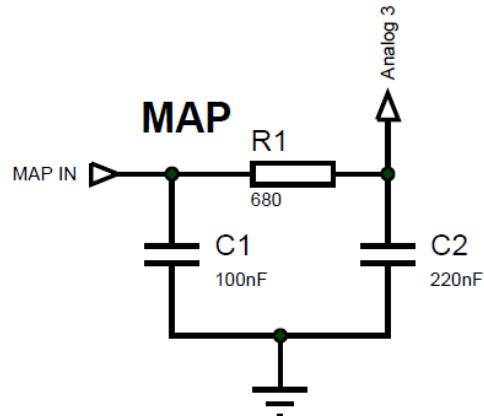
(<https://fenix.tecnico.ulisboa.pt/downloadFile/395139489176/Engine%20Control%20Unit.pdf>)

Pada map sensor terdapat rangkaian *wheatstone bridge* yang menciptakan perbedaan tegangan yang sebanding dengan regangan pada chip silikon.



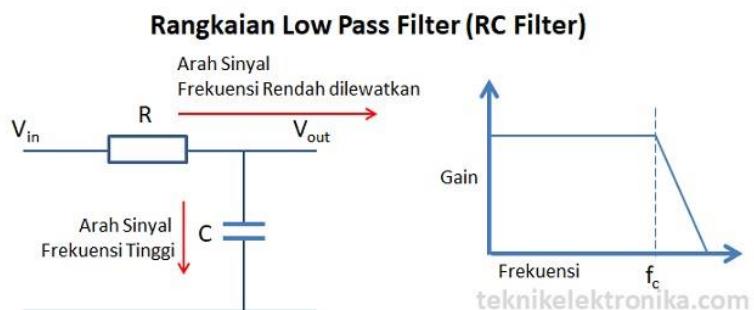
Gambar 34. Rangakain *wheatstone bridge* pada map sensor
(<https://fenix.tecnico.ulisboa.pt/downloadFile/395139489176/Engine%20Control%20Unit.pdf>)

Sinyal yang dihasilkan dari *Output* map sensor adalah berupa sinyal analog. Tahanan *chip silicon* yang berubah akan mempengaruhi tegangan *Output* pada map sensor.



Gambar 35.Rangkaian pengkondisi sinyal sensor MAP pada *ecu programmable*

Gambar 35 menunjukkan bahwa sinyal output map sensor harus di *filter* dahulu sebelum masuk ke pin analog pada mikrokontroller. Untuk menyaring sinyal output map sensor dibutuhkan rangkaian *low pass filter* (LPF), dimana rangkaian ini berfungsi untuk menyaring sinyal yang lewat pada frekuensi rendah, dan menghambat atau memblokir sinyal frekuensi tinggi.



Gambar 36. Rangkaian Low Pass RC Filter (LPF)
[\(https://teknikelektronika.com/pengertian-low-pass-filter-lpf-atau-tapis-lolos-bawah/\)](https://teknikelektronika.com/pengertian-low-pass-filter-lpf-atau-tapis-lolos-bawah/)

Rumus yang digunakan untuk menemukan titik cut-off Frekuensi dari rangkaian RC *filter* adalah $f = 1/2\pi RC$.

Dimana:

f = Frekuensi dalam satuan Hz

$\pi = 3.14$

R = Nilai Resistor dalam satuan Ohm (Ω)

C = Nilai Kapasitor dalam satuan Farad (F)

Pada rangkaian pengkondisi sinyal pada map sensor dihitung bahwa:

$$\pi = 3.14$$

$$R = 680\Omega$$

$$C = 220\text{nF} \text{ atau } 10 \times 10^{-9} \text{ atau } 0.000000022\text{F}$$

Dihitung:

$$f = 1/2\pi RC$$

$$f = 1/2 (3.14) (680) (0.000000022)$$

$$f = 1,064.409\text{Hz} \text{ atau sekitar } 1.064\text{KHz}$$

b. Rangkaian analog input untuk sensor IAT

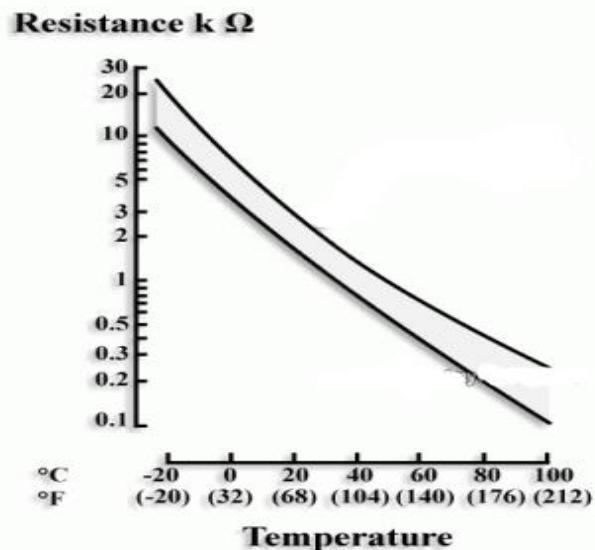
IAT (*intake air temperature sensor*) berfungsi mendeteksi suhu udara masuk pada *intake manifold*. Pada sensor IAT terdapat sebuah resistor yang perubahan nilai hambatannya dipengaruhi oleh suhu yang mengenainya.



Gambar 37. Intake Air Temperature Sensor

(http://www.ssdislesupply.com/p80_intake_air_temperature_sensor__iat_sensor.html)

Intake Air Temperature Sensor adalah sebuah resistor dengan jenis NTC (*negative temperature coefficient*) yaitu jika temperatur tinggi maka nilai tahanan pada resistor kecil, sebaliknya jika temperatur rendah maka nilai tahanan resistor besar.



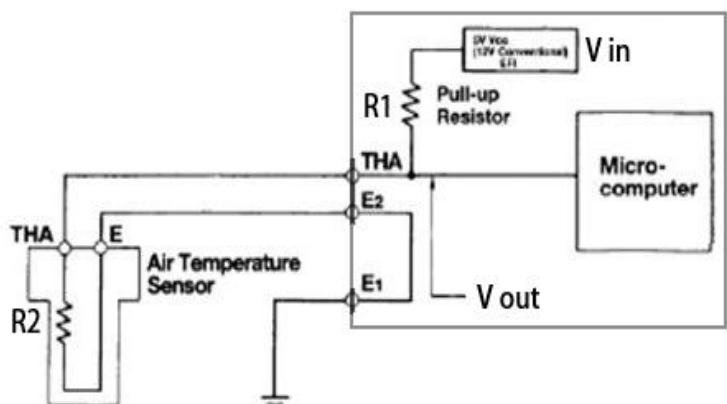
Gambar 38. Grafik Perbandingan Temperature dengan Resistance pada IATS

(<http://affifhaikalanggalaya.blogspot.com/2017/06/sistem-fuel-injection-pada-kendaraan.html>)

Pada dasarnya prinsip pada rangkaian sensor IAT adalah menggunakan rangkaian pembagi tegangan (*voltage divider*) dengan tambahan rangkaian *low pass filter* (LPF) , dimana sensor IAT bertindak sebagai R2 yang langsung terhubung ke *ground*.

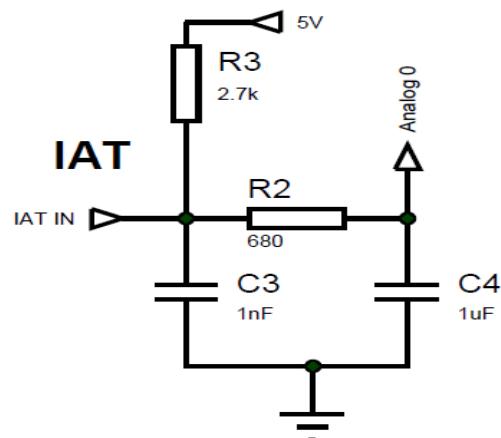
Rumus pembagi tegangan (*voltage divider*) :

$$V_{out} = (R_1 / (R_1 + R_2)) \times V_{in}$$



Gambar 39. Wiring Sensor IAT
(<https://www.teknik-otomotif.com/2017/01/sensor-temperatur-udara-intake-air.html>)

Pada gambar 40 menunjukkan *wiring intake air temperature sensor* dengan menggunakan rangkaian pembagi tegangan (*voltage divider*) untuk masukan analog input pada mikrokontroller.

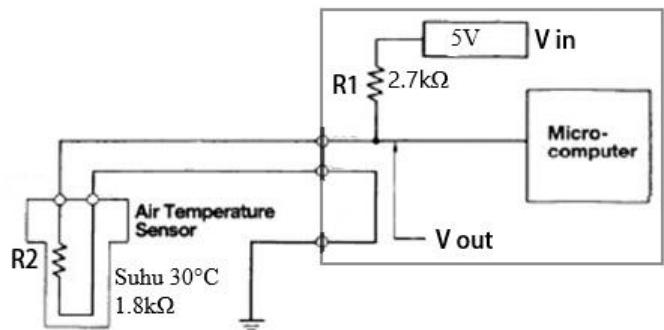


Gambar 40. Rangkaian Pengkondisi Sinyal IAT pada *ecu programmable*

Gambar 40 menunjukkan rangkaian pengkondisi sinyal IAT pada *ecu programmable*, dengan menggunakan rangkaian pembagi tegangan (*voltage divider*) serta tambahan rangkaian *low pass filter* (LPF) untuk menyaring sinyal analog hasil dari rangkaian pembagi tegangan tersebut.

Dari gambar diatas dapat dihitung tegangan keluar dari rangkaian pembagi tegangan, serta sinyal pada rangkaian *low pass filter* (LPF).

Sebagai contoh untuk rangkaian pembagi tegangan sensor IAT pada *ecu programmable*:



Gambar 41. Contoh Rangkaian pembagi tegangan sensor IAT

Jika suhu yang masuk pada intake manifold adalah 30°C, kemudian diketahui bahwa tahanan sensor IAT dengan suhu 30°C adalah 1800Ω. Dan tegangan pada Vin adalah 5V dengan resistor pull up senilai 2700Ω. Dengan rumus pembagi tegangan maka kita dapat menghitung tegangan yang keluar dari rangkian, yang kemudian masuk ke analog input mikrokontroller adalah sebagai berikut:

$$\begin{aligned}
 V_{out} &= (R_1 / (R_1 + R_2)) \times V_{in} \\
 &= (2700\Omega / (2700\Omega + 1800\Omega)) \times 5V \\
 &= (2700\Omega / 4500\Omega) \times 5V \\
 &= (0.6\Omega) \times 5V \\
 &= 3V
 \end{aligned}$$

Dari hasil perhitungan diatas dapat disimpulkan, bahwa ketika suhu udara masuk pada intake manifold sebesar 30°C, maka tegangan yang masuk pada pin analog input pada mikrokontroller sebesar 3V.

c. Rangkaian analog input untuk sensor EOT

ECT (*engine coolant temperature sensor*) berfungsi mendeteksi temperatur air pendingin mesin. Pada sensor ECT terdapat sebuah resistor yang perubahan nilai hambatannya dipengaruhi oleh suhu yang mengenainya.

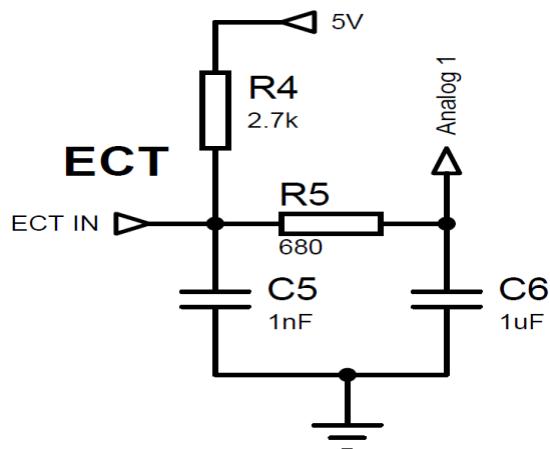


Gambar 42. Engine Coolant Temperature Sensor

(https://www.injetech.com/catalog/product/view/_ignore_category/1/id/6826/)

Sama seperti dengan IAT, Engine Coolant Temperature Sensor adalah sebuah resistor dengan jenis NTC (*negative temperature coefficient*) yaitu jika temperatur tinggi maka nilai tahanan pada resistor kecil, sebaliknya jika temperatur rendah maka nilai tahanan resistor besar.

Begitu juga dengan rangkaian pengkondisi sinyal, sama seperti pada rangkaian IAT. Berikut rangkaian pengkondisi sinyal ECT pada *ecu programmable*.



Gambar 43. Rangkaian pengkondisi sinyal ECT
pada *ecu programmable*

Gambar 43 menunjukkan rangkaian pengkondisi sinyal ECT pada *ecu programmable*, dengan menggunakan rangkaian pembagi tegangan (*voltage divider*) serta tambahan rangkaian *low*

pass filter (LPF) untuk menyaring sinyal analog hasil dari rangkaian pembagi tegangan tersebut.

d. Rangkaian analog input untuk sensor TPS

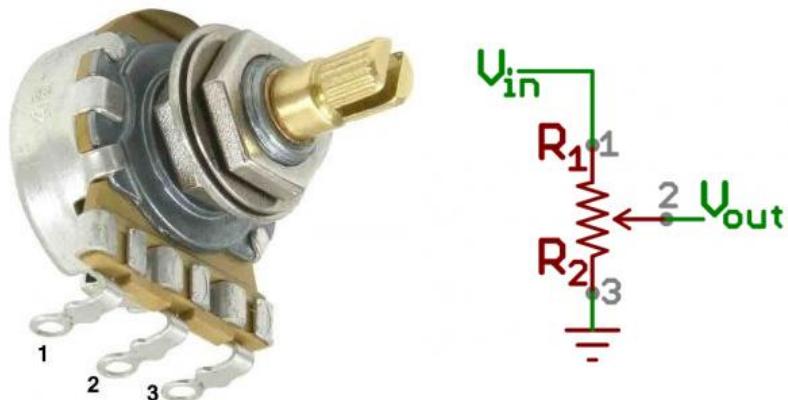
TPS (Throttel Position Sensor) berfungsi mendeteksi perubahan posisi katup throttle gas pada mesin pembakaran dalam.



Gambar 44. TPS (Throttel Position Sensor)

(https://www.autozone.com/engine-management/throttle-position-sensor-tps/duralast-throttle-position-sensor-tps/341570_0_0)

Pada sensor TPS (Throttel Position Sensor) terdapat komponen elektronika yaitu *potentiometer*. Potensiometer adalah komponen elektronika sejenis resistor yang nilai resistansinya dapat diatur sesuai keinginan (resistor variabel) dengan menggeser tuasnya.

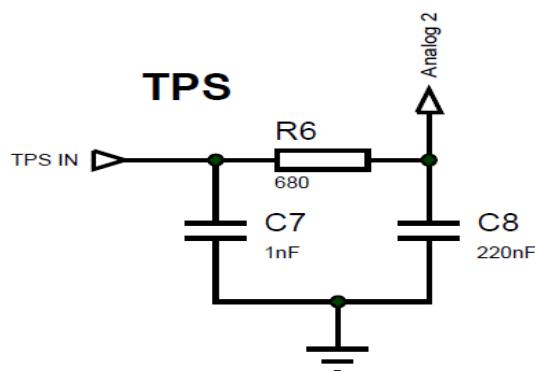


Gambar 45. Potensiometer
(<https://elektroku.com/fungsi-potensiometer-dan-cara-kerjanya/>)

Potensiometer terdiri dari sebuah elemen resistif yang membentuk jalur dengan terminal yang berada di kedua buah ujungnya. Sementara terminal satunya berada di tengah sebagai tuas atau penyapu untuk menentukan pergerakan pada jalur elemen resistif yang nantinya dapat mempengaruhi naik turunnya nilai resistansi.

Elemen resistif yang terdapat pada komponen potensiometer pada umumnya terbuat dari campuran logam dan keramik atau karbon. Jika potensiometer hanya digunakan 2 kaki (1 ujung dan 1 penyapu), dapat berfungsi sebagai resistor variabel atau rheostat. Namun jika yang digunakan 3 kaki, maka kaki yang satunya berfungsi sebagai V out sinyal keluaran.

Tegangan yang masuk pada sensor TPS adalah 5V. Sensor akan merubah tegangan suplai 5V menjadi sinyal, dimana nilai tegangan keluarnya setara dengan bukaan katup throttel. Sinyal keluaran dari sensor TPS berupa sinyal analog, dimana sinyal ini akan dikondisikan atau disaring sebelum masuk ke mikrikontroler. Berikut rangkaian pengkondisi sinyal TPS pada *ecu programmable*.



Gambar 46. Rangkaian pengkondisi sinyal TPS pada *ecu programmable*.

Pada gambar 46 rangkaian pengkondisi sinyal TPS sama seperti rangkaian pengkondisi sinyal MAP yang menggunakan rangkaian *low pass filter* (LPF), dimana rangkaian ini berfungsi untuk menyaring sinyal yang lewat pada frekuensi rendah, dan menghambat atau memblokir sinyal frekuensi tinggi.

e. Rangkaian analog input untuk sensor O2 Sensor

O2 sensor (Oxygen Sensor) berfungsi mendeteksi hasil pembakaran pada *exhaust manifold*, yang kemudian memberikan *feedback* pada mikrokontroller untuk mengontrol perbandingan campuran bahan bakar dengan udara.



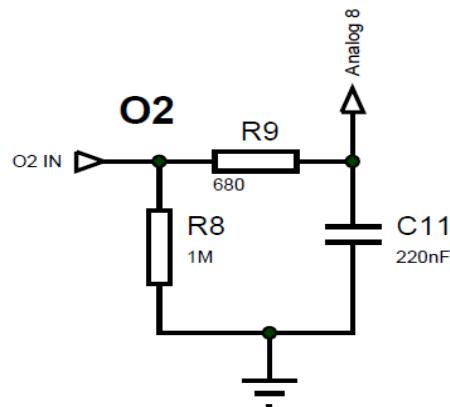
Gambar 47. O2 sensor (Oxygen Sensor).

(<http://www.counterman.com/o2-sensors-keep-watch-over-the-air-fuel-mixture/>)

Mengawasi campuran udara dan bahan bakar (Air/Fuel) merupakan fungsi penting dari sensor ini, karena mempengaruhi emisi, ekonomi bahan bakar, dan kinerja. Campuran bahan bakar yang sedikit kaya membuat lebih banyak tenaga tetapi juga meningkatkan emisi karbon monoksida (CO). Campuran yang lebih kurus membantu penghematan bahan bakar, tetapi jika campuran Air/Fuel terlalu kurus, mesin cepat rusak, kehilangan tenaga.

Sinyal yang dihasilkan dari O2 sensor berupa sinyal analog, dimana besarnya tegangan keluar dari sensor dipengaruhi oleh

hasil pembakaran pada *exhaust manifold*. Sinyal keluaran dari sensor harus dikondisikan dan disaring sebelum masuk ke analog input mikrokontroler. Berikut rangkaian pengkondisi sinyal analog dari O₂ sensor pada *ecu programmable*.



Gambar 48.Rangkaian pengkondisi sinyal O₂ Sensor pada *ecu programmable*.

Pada gambar 49 rangkaian pengkondisi sinyal O₂ Sensor yang menggunakan rangkaian *low pass filter* (LPF), dimana rangkaian ini berfungsi untuk menyaring sinyal yang lewat pada frekuensi rendah, dan menghambat atau memblokir sinyal frekuensi tinggi.

4. Perancangan Rangkaian Pengkondisi Sensor CKP tipe Hall

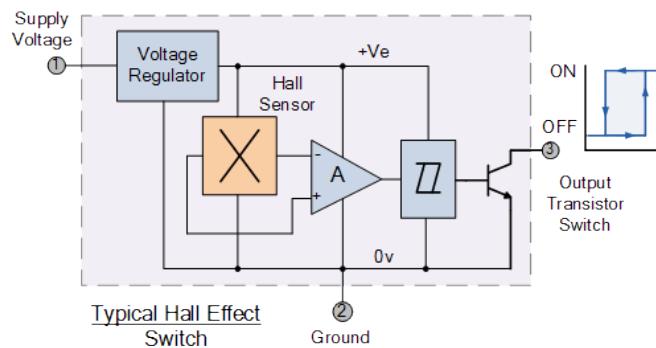
CKP (Crank Position Sensor) berfungsi mendeteksi sudut crankshaft serta mengukur kecepatan mesin. Terdapat 2 jenis tipe CKP yaitu:

a. CKP Tipe Variable Reluctor

Sensor Variable Reluctor terdiri dari magnet permanen yang dibungkus dengan gulungan kumparan kawat. Dimana arus yang dihasilkan dari tipe ini adalah arus AC (*alternating current*). Jika dilihat menggunakan Oscilloscope, sinyal pada sensor tipe ini menghasilkan sinyal sinus.

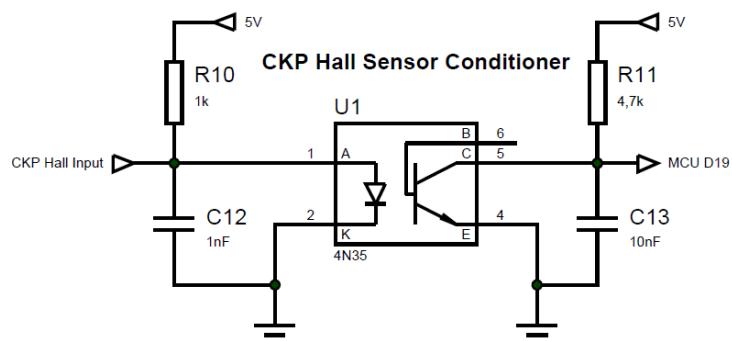
b. CKP Tipe Hall Effect

Sensor ini bekerja seperti Switch dimana arus keluar yang dihasilkan berupa Arus DC (*direct current*). Jika dilihat dengan menggunakan Oscilloscope, sinyal yang dihasilkan berupa sinyal kotak (square wave).



Gambar 49. Rangkaian *internal* Hall Effect Sensor
<https://www.electronics-tutorials.ws/electromagnetism/hall-effect.html>

Pada *ecu programmable* berbasis Atmega2560 ini, tipe sensor CKP yang digunakan berjenis Hall Effect Sensor. Berikut rangkaian pengkondisi sinyal CKP tipe Hall Effect pada *ecu programmable*.



Gambar 50. Rangkaian Pengkondisi Sensor CKP Hall Effect

Rangkaian pada gambar 50 berfungsi untuk mengkondisikan sinyal dari sensor CKP. Mikrokontroller membaca sinyal digital yang dihasilkan dari rangkaian pada gambar 50, oleh karena itu perlu ditambahkan rangkaian Pull Up dimana rangkaian ini berfungsi untuk

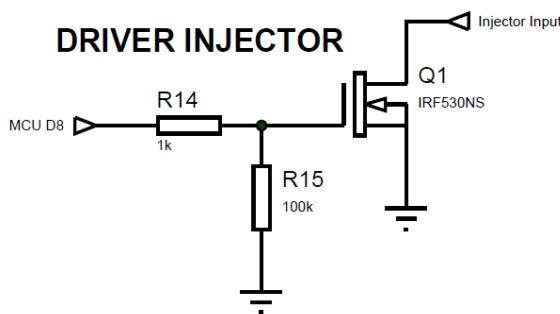
mengatasi kondisi “*floating*” yang terjadi pada rangkaian agar menjadi terdefenisi ke sinyal “High” atau “Low”.

5. Perancangan Rangkaian Aktuator

Injector dan Fuel Pump berperan sebagai aktuator, dimana untuk mengontrol aktuator diperlukan rangkaian *driver* agar bisa mengaktifkan aktuator tersebut.

a. Rangkaian Driver Injector

Injector berfungsi untuk mengabutkan bahan bakar ke *intake manifold* atau langsung keruang bakar. Injector adalah sejenis *solenoid* dimana jika dialirkan tegangan sebesar 12V maka nozzle akan membuka. Tegangan *Output* mikrokontroler hanya sebesar 5V, tegangan 5V tidak akan mampu mengaktifkan *solenoid* pada *injector*. Maka dari itu, dibutuhkan rangkaian penguatan dengan sebuah Transistor ataupun sebuah Mosfet. Berikut rangkaian *driver injector* pada *ecu programmable*.



Gambar 51. Rangkaian Driver Injector pada *ecu programmable*.

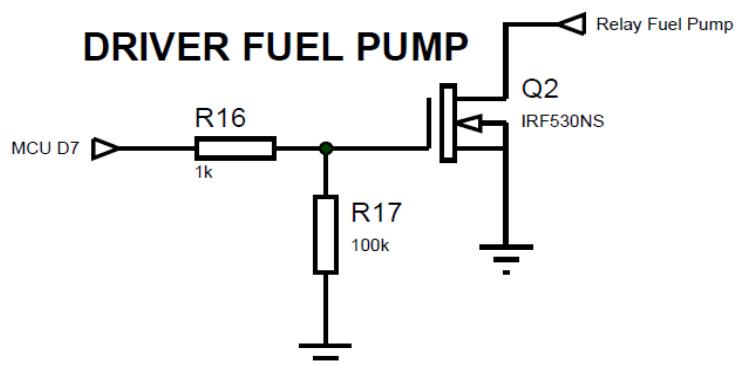
Rangkaian pada gambar 51 ini berfungsi untuk mengaktifkan Injector. Fungsi Mosfet (*metal oxide semiconductor field effect transistor*) pada gambar 51 adalah sebagai *switch electronic*. Dimana ketika ada tegangan keluar dari mikrokontroller (Logika High) yang mengaktifkan pin *gate* pada Mosfet, maka pin *drain* dengan *source* dalam kondisi *saturation*, yang artinya pin *drain* dengan *source* terhubung. Jika pin *drain* dengan *source*

terhubung, maka tegangan 12V yang melewati kumparan solenoid akan terhubung dengan *ground* akibatnya injector aktif.

Dan sebaliknya jika tegangan output mikrokontroler 0V (Logika Low) pada gate Mosfet, maka pin *drain* dengan *source* dalam kondisi *cut off* yang artinya antara pin *drain* dengan *source* tidak terhubung.

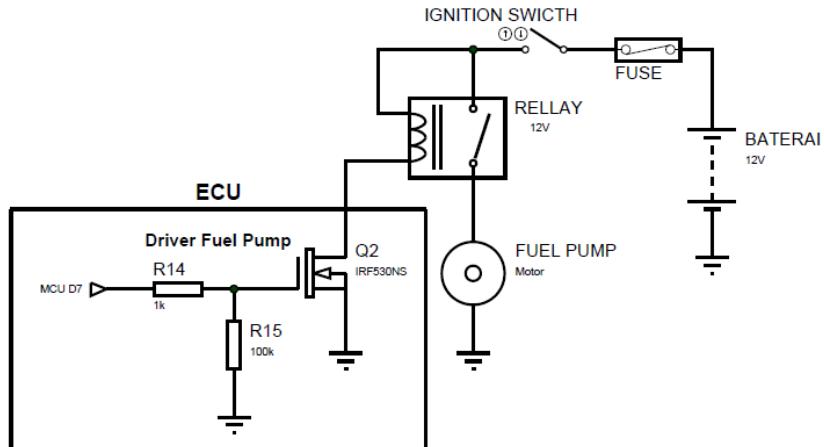
b. Rangkaian Driver Fuel Pump

Fuel Pump berfungsi menghisap bahan bakar dari tangki kemudian di pompa ke injektor, serta membangkitkan tekanan bahan bakar supaya bahan bakar dapat dikabutkan pada injektor. Fuel Pump adalah sejenis motor listrik yang beroperasi pada tegangan 12V. Rangkaian pengontrolan Fuel Pump pada *ecu programmable* sama seperti dengan rangkaian pada injektor, yang berbeda hanya pengontrolannya diberi relay 12V untuk mengaktifkan motor listrik.



Gambar 52.Rangkaian Driver Relay Fuel Pump

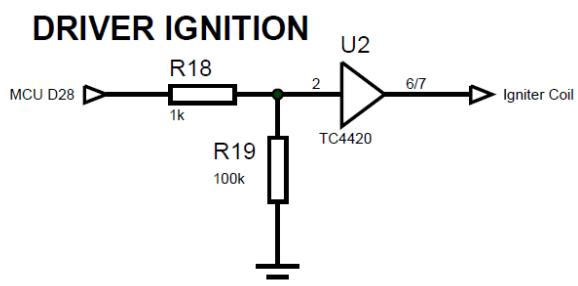
Transistor Mosfet pada *ecu programmable* tidak mampu mengaktifkan langsung motor listrik, sehingga perlu diberi tambahan *relay*. Relay ini berfungsi seperti Switch, memutuskan dan menghubungkan *kontak point* pada terminal relay. Sehingga dapat mengaktifkan motor listrik dengan arus yang tinggi.



Gambar 53.Wiring Fuel Pump dengan Relay.

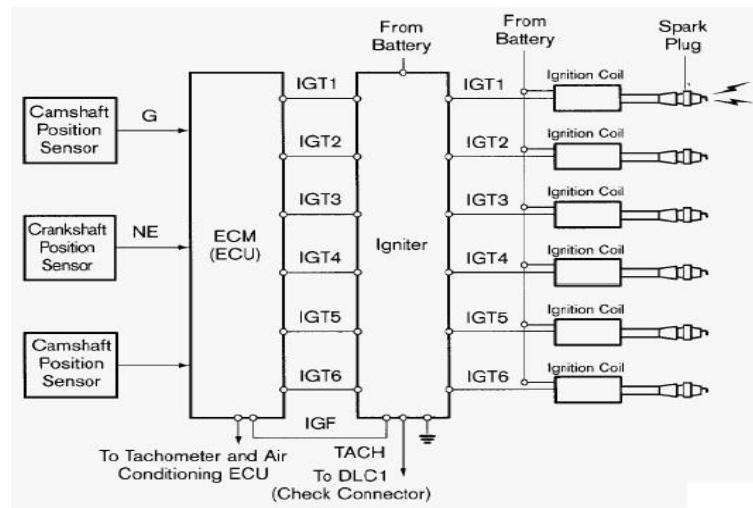
6. Rangkaian Driver Ignition

Sistem pengapian berfungsi menyediakan percikan bunga api bertegangan tinggi pada busi untuk membakar campuran udara dan bahan bakar di dalam ruang bakar.Untuk membuat tegangan tinggi dibutuhkan suatu Coil, dimana fungsi dari Coil adalah membangkitkan tegangan listrik dari 12 Volt menjadi Ribuan Volt. Pada sistem pengapian electronic, untuk mengaktifkan Coil dibutuhkan suatu rangkaian dengan menggunakan Transistor Power.Transistor Power ini biasa disebut IGBT (*insulated gate bipolar transistor*), transistor ini biasa digunakan pada Igniter pengapian.Untuk mengontrol Igniter pengapian, cukup dengan memicu (*trigger*) dengan tegangan dari ECU.Berikut rangkaian Driver Ignition pada *ecu programmable*.



Gambar 54.Rangkaian Driver Pengapian dengan IC TC4420

IC TC4420 adalah komponen *driver mosfet* yang berfungsi untuk menyetabilkan tegangan yang keluar dari mikrokontroler, dan mencegah tegangan balik ke mikrokontroler serta untuk memicu (*trigger*) pada Igniter Coil.



Gambar 55.Wiring Sistem Pengapian DIS (*direct ignition system*)
<http://green-elektronik.blogspot.com/2016/03/sistem-pengapian-terkontrol-komputer.html>)

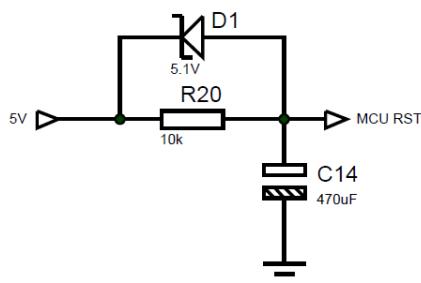
Pada gambar 55 ECU memicu (*trigger*) Igniter sesuai *firing order*, kemudian Igniter memutus dan menghubung arus pada *primer coil* berdasarkan sinyal dari ECU. Ketika arus pada *primer coil* diputus secara tiba-tiba oleh Igniter, maka akan terjadi induksi pada Coil. Induksi pada Coil ini yang menyebabkan naiknya tegangan listrik dari 12 Volt menjadi ribuan Volt.

7. Rangkaian Reset Protection

Pin RESET pada mikrokontroler berfungsi yaitu untuk merestart program, sehingga kembali ke program awal. Jika program yang sudah berjalan pada ECU kembali ke program awal, akibatnya pengontrolan mesin tidak sempurna bahkan mesin tidak dapat hidup. Pin RESET pada mikrokontroler Atmega2560 adalah aktif Low. Jika sebuah sinyal Low diaplikasikan pada pin ini, maka mikrokontroler akan direset. Gangguan pin RESET sendiri terjadi akibat *noise* pada

mesin dan juga tegangan tinggi yang dihasilkan oleh koil pengapian akan terjadi gangguan frekuensi radio suara (radio interference) yang mengganggu kinerja mikrokontroler. Oleh karena itu dibutuhkan rangkaian *reset protection* untuk mencegah terjadinya *reset program* pada mikrokontroler. Berikut rangkaian RESET pada *ecu programmable*.

RESET PROTECTION



Gambar 56.Rangkaian Reset Protection pada *ecu programmable*.

Fungsi resistor pada gambar 56 sebagai *pull up* agar pin RESET mikrokontroller selalu dalam kondisi High. Sedangkan dioda Zener berfungsi untuk penstabil tegangan masuk agar tetap 5V.

D. Peralatan yang digunakan

Dalam Pembuatan ECU Programmable berbasis mikrokontroler Atmega2560 ini memerlukan beberapa alat yang harus disediakan, yaitu :

1. Solder
2. Oscilloscope
3. Multimeter
4. Tang Potong
5. Cutter
6. Gunting
7. Penjepit PCB dan Kaca Pembesar
8. Pemotong Akrilik
9. Obeng set
10. Pinset
11. Ekstrator

12. Solder Pump

Sedangkan bahan-bahan yang dibutuhkan, yaitu :

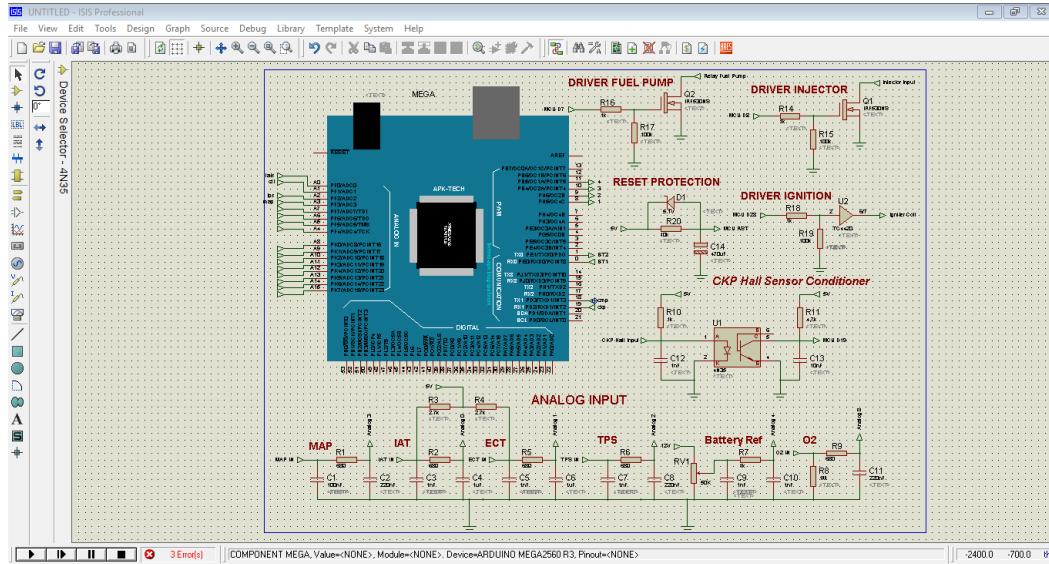
1. Komponen untuk Power Supply dan sistem mikrokontroller, seperti Modul Robotdyn Promini 2560, IC LM 317, IC 4N35, IRF 530, IC TC4420, Connector ECU Yamaha 33 pin, dan komponen pendukung misalnya resistor, resistor variabel, capasitor, dioda, LED, pin holder, inductor, dioda zener dll.
2. PCB (*printed circuit board*)
3. Timah (Tenol)
4. Fluks
5. Lem G
6. Box acrylic
7. Amplas
8. Kabel

E. Langkah Pembuatan Alat

Pembuatan alat proyek akhir ini terdiri dari pembuatan rangkaian ECU Programmable, membuat *program firmware*, pembuatan desain *printed circuit board* (PCB) untuk rangkaian *power supply*, rangkaian pengkondisi sinyal CKP, rangakaian *analog input*, rangkaian *driver injector* dan *driver fuel pump*, dudukan modul *robotdyn promini* 2560, serta pemasangan komponen, pengujian laporan, pembuatan desain box rangkaian, dan perakitan pada box.

1. Membuat Rangkaian ECU Programmable

Pembuatan rangkaian ini menggunakan aplikasi ISIS Proteus 7.09 Professional. Pembuatan rangkaian meliputi, rangkaian pengkondisi sinyal CKP, rangakaian *analog input*, rangkaian *driver injector* dan *driver fuel pump*, dudukan modul *robotdyn promini* 2560. Pembuatan rangkaian ini bertujuan untuk mensimulasikan rangkaian dan program firmware agar mengetahui kinerja rangakaian.



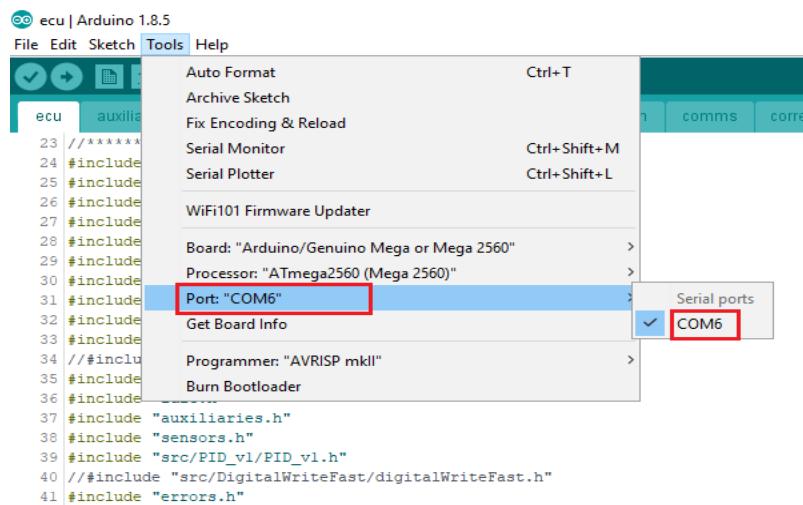
Gambar 57.Pembuatan rangkaian ECU Programmble menggunakan aplikasi ISIS Proteus 7.09 Professional.

2. Pembuatan *program firmware* menggunakan Arduino

Program firmware merupakan serangkaian instruksi yang ditulis untuk melakukan suatu fungsi spesifik pada mikrokontroller. Berikut langkah-langkah untuk memasukan program ke dalam mikrokontroller:

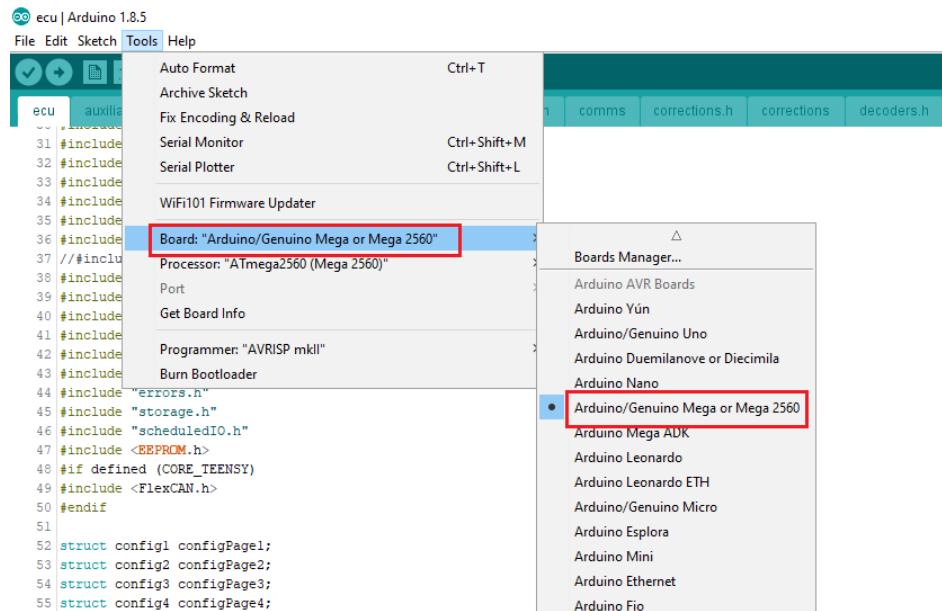
- Menghubungkan modul *robotdyn promini 2560* dengan perangkat komputer.
- Pilih menu Tools pada aplikasi Arduino

- Pilih dan klik Port, kemudian pilih port komputer yang digunakan pada mikrokotroller.



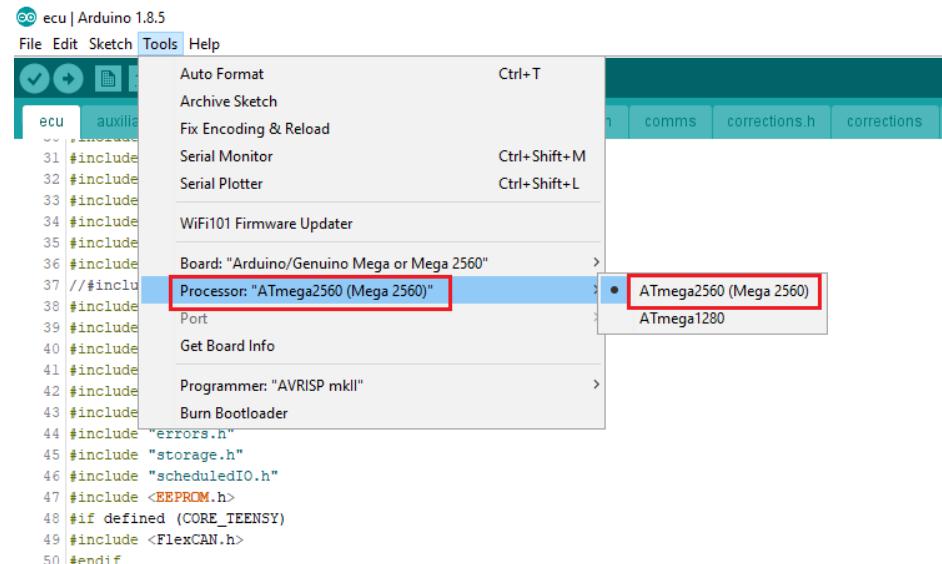
Gambar 58.Konfigurasi Port Mikrokontroler pada Komputer

2) Pilih dan klik Board, kemudian pilih “Arduino/Genuino Mega or Mega 2560”



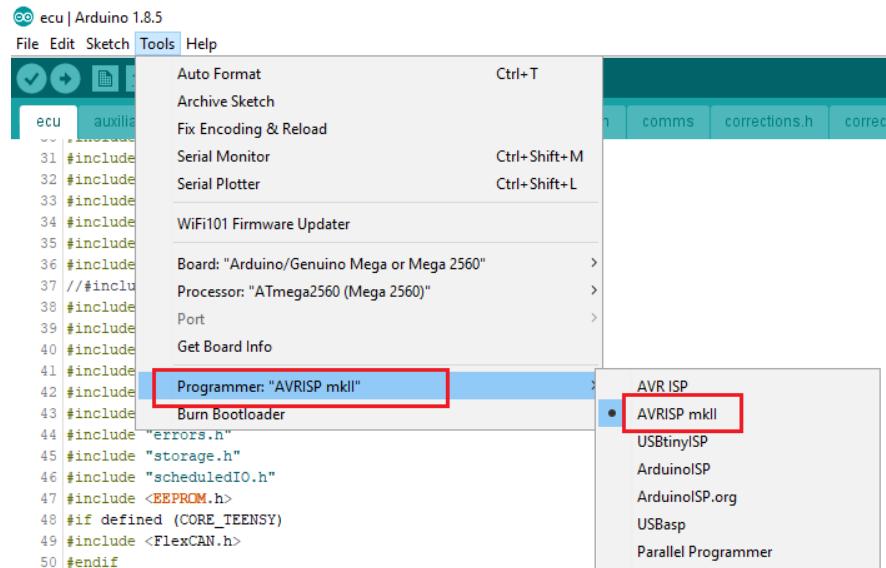
Gambar 59.Konfigurasi Board Mikrokontroler pada Komputer

3) Pilih dan klik Processor, kemudian pilih “Atmega2560 (Mega 2560)”



Gambar 60.Konfigurasi Processor pada Mikrokontroler

4) Pilih dan Klik Programmer, kemudian pilih “AVRISP mkll”



Gambar 61.Konfigurasi *Programmer* pada Mikrokontroler

- Buka program yang sudah dibuat, kemudian klik “Verify” pada toolbars untuk memastikan program tersebut tidak *error*, tunggu sampai proses selesai dan muncul “Done Compile”

```

ecu | Arduino 1.8.5
File Edit Sketch Tools Help
ecu auxiliaries auxiliaries cancomms.h cancomms commss.h commss corrections.h corrections decoders decoders display display errors.h errors globals.h idia.h
31 #include "scheduler.h"
32 #include "cancomm.h"
33 #include "cancomm.h"
34 #include "matha.h"
35 #include "corrections.h"
36 #include "timers.h"
37 //#include "display.h"
38 #include "decoders.h"
39 #include "auxiliaries.h"
40 #include "sensors.h"
41 #include "soc/FID_v1/FID_v1.h"
42 #include "soc/DigitalWriteFast/digitalWriteFast.h"
< <
Done compiling.
C:\Program Files (x86)\Arduino\hardware\tools\avr\bin\avr-gcc-ar rcs "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core.a" "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core.o" "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\core.a" "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\core.o" "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\matha.o" "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\matha.o" "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\corrections.o" "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\corrections.o" "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\timers.o" "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\timers.o" "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\display.o" "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\display.o" "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\decoders.o" "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\decoders.o" "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\auxiliaries.o" "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\auxiliaries.o" "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\sensors.o" "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\sensors.o" "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\soc\FID_v1\FID_v1.o" "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\soc\FID_v1\FID_v1.o" "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\soc\DigitalWriteFast\digitalWriteFast.o" "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\soc\DigitalWriteFast\digitalWriteFast.o" "C:\Users\user\AppData\Local\Temp\arduino_cache_343804\core\core_arduino_avr mega_cpu_atmega2560_0c11237ac70e4a9b93505fb077f54c.a
Archiving built core (caching) in C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\core.a "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\core.a" "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\core.o" "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\matha.o" "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\corrections.o" "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\timers.o" "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\display.o" "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\decoders.o" "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\auxiliaries.o" "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\sensors.o" "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\soc\FID_v1\FID_v1.o" "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\soc\DigitalWriteFast\digitalWriteFast.o"
Linking everything together...
C:\Program Files (x86)\Arduino\hardware\tools\avr\bin\avr-gcc -w -Os -g -fno-fuse-linker-plugin -Wl,-T,linker-script -s --relax -mmcu=atmega2560 -o "C:\Users\user\AppData\Local\Temp\arduino_build_391488\sketch\sketch_20200120\sketch_20200120.ino.elf" "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\core.a" "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\core.o" "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\matha.o" "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\corrections.o" "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\timers.o" "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\display.o" "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\decoders.o" "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\auxiliaries.o" "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\sensors.o" "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\soc\FID_v1\FID_v1.o" "C:\Users\user\AppData\Local\Temp\arduino_build_391488\core\soc\DigitalWriteFast\digitalWriteFast.o" "C:\Users\user\AppData\Local\Temp\arduino_cache_343804\core\core_arduino_avr mega_cpu_atmega2560_0c11237ac70e4a9b93505fb077f54c.a
Using library EEPROM at version 2.0 in folder: C:\Program Files (x86)\Arduino\hardware\arduino\avr\libraries\EEPROM
Sketch uses 72458 bytes (28%) of program storage space. Maximum is 253952 bytes.
Global variables use 4529 bytes (55%) of dynamic memory, leaving 3663 bytes for local variables. Maximum is 8192 bytes.

```

Gambar 62.Verifikasi *Sketch Program*

- Klik Upload pada toolbars untuk memasukan program kedalam mikrokontroller, tunggu sampai proses selesai dan muncul “Done Uploading”

```
Done uploading.

avrduke: writing flash (86004 bytes):
Writing | ##### | 100% 12.83s

avrduke: 86004 bytes of flash written
avrduke: verifying flash memory against C:\Users\user\AppData\Local\Temp\ard
avrduke: load data flash data from input file C:\Users\user\AppData\Local\Te
avrduke: input file C:\Users\user\AppData\Local\Temp\arduino_build_892066\ec
avrduke: reading on-chip flash data:

Reading | ##### | 100% 9.85s

avrduke: verifying ...
avrduke: 86004 bytes of flash verified

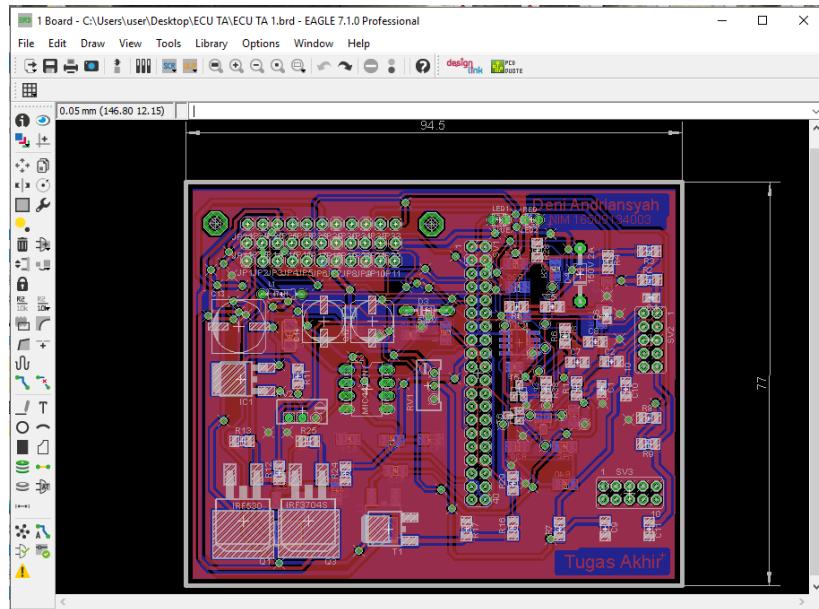
avrduke done. Thank you.
```

Gambar 63. Proses *Uploading Program*

3. Pembuatan *Printed Circuit Board* (PCB)

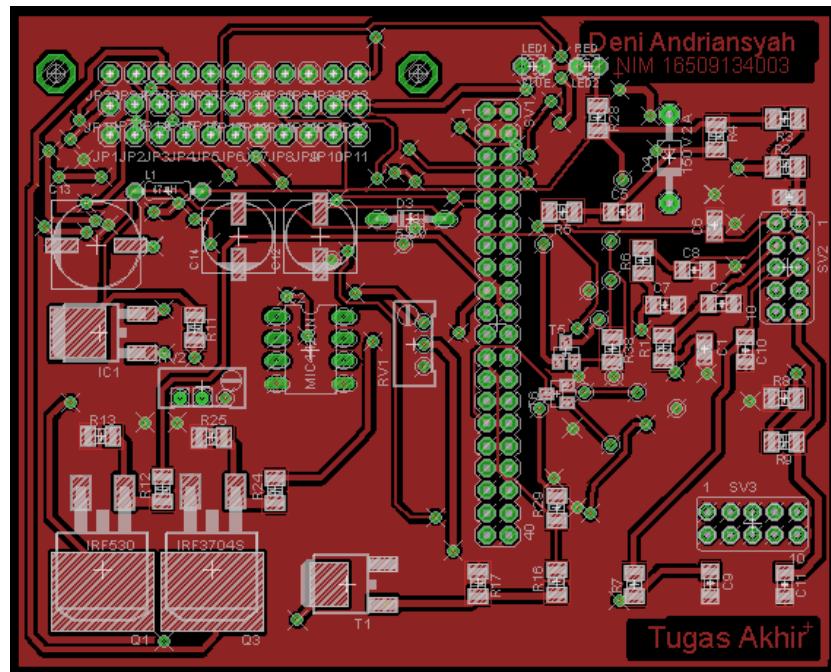
a. Pembuatan *layout* PCB

Langkah awal pembuatan PCB adalah menggambar layout rangkaian dengan perangkat lunak. Perangkat lunak yang digunakan yaitu EAGLE. Berikut proses desain *layout* PCB menggunakan aplikasi EAGLE.

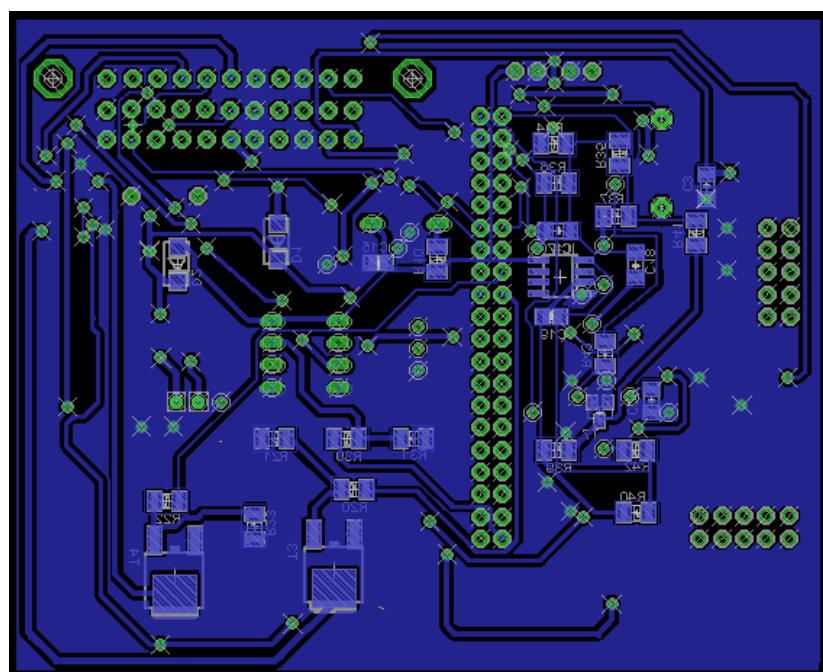


Gambar 64. proses desain *layout* PCB pada ecu programmable menggunakan aplikasi EAGLE.

Desain PCB ini menggunakan jenis *double layer*, dimana hubungan rangkaian pada sisi atas PCB dengan sisi bawah PCB terhubung. Berikut tampilan PCB pada posisi atas dan bawah:



Gambar 65. *Layout PCB* pada sisi atas.



Gambar 66. *Layout PCB* pada sisi bawah.

b. Pencetakan *Printed Circuit Board* (PCB).

Untuk mencetak layout PCB yang sudah di desain tadi, penulis menggunakan jasa pencetakan PCB yaitu JLC PCB. JLC PCB ini adalah industri pencetakan PCB yang berada di china, alamat websitenya adalah <https://jlpcb.com>.

Invoice
No.2367178A20180613081416Y6 

Ship From: **Ship To:** **Billing Address:**

JIALICHUANG
TECH.DEVELOPMENT Deni
CO.LTD. Andriansyah
Jalan Hasyim
30A, Area A, Shennan Tahir Dusun 1
Garden Bldg, Paya Gambar,
Nanshan Dist. Batang Kuis
Shenzhen 518000
China Deli Serdang
20372
+86 13738561292 ID

082370128989

Date Ordered: 2018-
06-08 14:29:42

Products

PCB	Unit	Quantity	Total Price
- File: ECU TA DENI_Y6	\$0,2	10	\$2
- 77mm *			
95mm;			
- Layers: 2;			
- PCB Thickness:			
1,6mm;			
- PCB Quantity: 10;			
- PCB Color: Green;			
- Surface Finish: HASL(with lead);			
- Copper Weight: 1 oz;			

Sub-Total: \$2
Express: \$8,51
Total: \$10,51

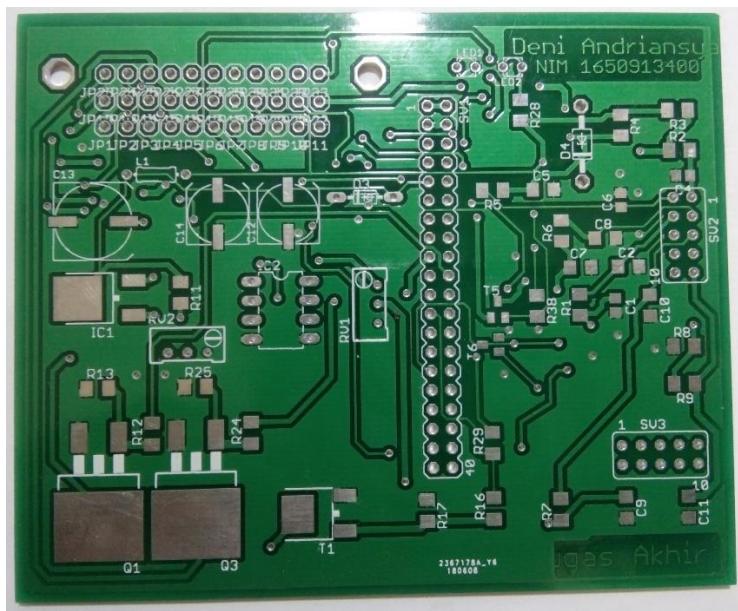
Gambar 67. *Invoice* pemesanan pencetakan PCB.

Process



1MI	2018-06-08 18:32:37
2Drilling	2018-06-08 19:12:49
3Copper Deposition	2018-06-08 22:35:03
4Image the outer layers	2018-06-08 23:30:02
5Pattern Plating	2018-06-09 01:25:09
6Automatic Optical Inspection(AOI)	2018-06-09 04:36:57
7Solder Mask	2018-06-09 04:49:16
8Silkscreen	2018-06-09 13:36:08
9Hot Air Solder Leveling(HASL)	2018-06-09 17:22:45
10Electrical Test	2018-06-09 23:19:37
11Profiling,V-cut scoring	2018-06-10 03:57:16
12Final inspection	2018-06-10 06:11:41
13Packaging,Delivery	2018-06-10 06:55:01

Gambar 68. Process penggeraan pencetakan *layout* PCB.

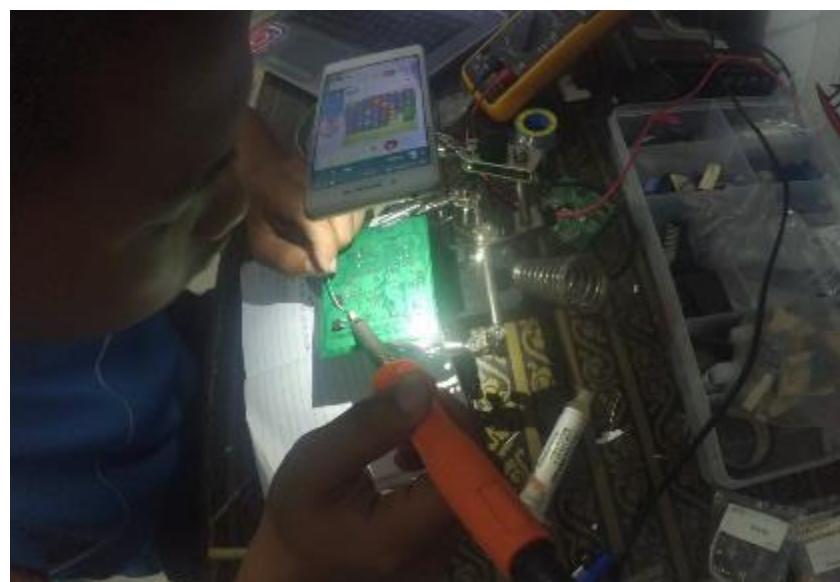


Gambar 69.Layout PCB ecu programmable

c. Perakitan Komponen

Jika PCB selesai di cetak, maka selanjutnya yaitu pemasangan seluruh komponen pada PCB, dengan urutan sebagai berikut :

- 1) Menyiapkan seluruh komponen yang dibutuhkan.
- 2) Memasang komponen dari jenis yang sama dengan urutan komponen yang benar.
- 3) Memberi *fluks* pada *through hole* PCB dan kaki komponen yang akan di solder, untuk melindungi cairan timah dari udara pada saat menyolderan.
- 4) Menyolder kaki komponen hingga semua komponen terpasang.
- 5) Memotong komponen yang berlebihan agar terlihat rapi.
- 6) Membersikan cairan *fluks* yang masih menempel.



Gambar 70.Proses perakitan komponen pada PCB.

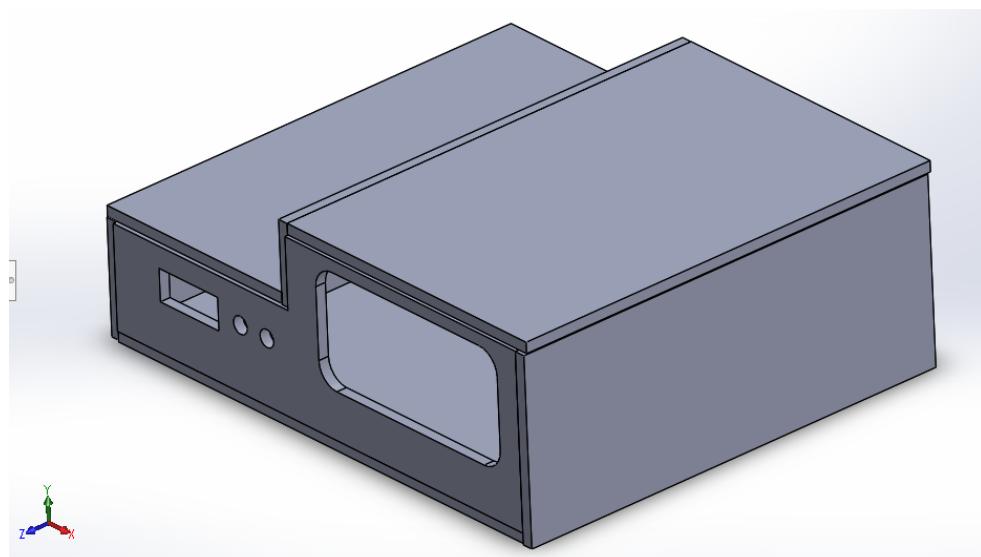
4. Pengujian Rangkaian

Setelah komponen telah terpasang semua, langkah selanjutnya yaitu melakukan pengujian rangkaian. Hal ini perlu dilakukan agar kita mengetahui bahwa cara kerja rangkaian sudah baik atau belum.

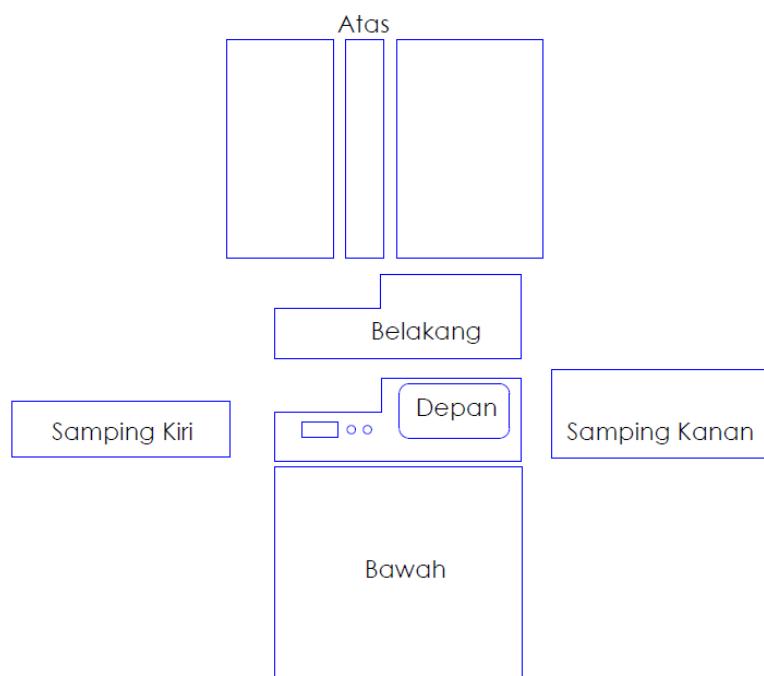
5. Pembuatan Box

a. Perencanaan Ukuran

Pembuatan box ini bertujuan untuk melindungi komponen rangkaian ECU Programmable dari gangguan yang akan mengakibatkan korosi atau kerusakan pada komponen. Berikut gambaran box ECU Programmable.



Gambar 71.Pandangan Isometric Box ECU Programmable

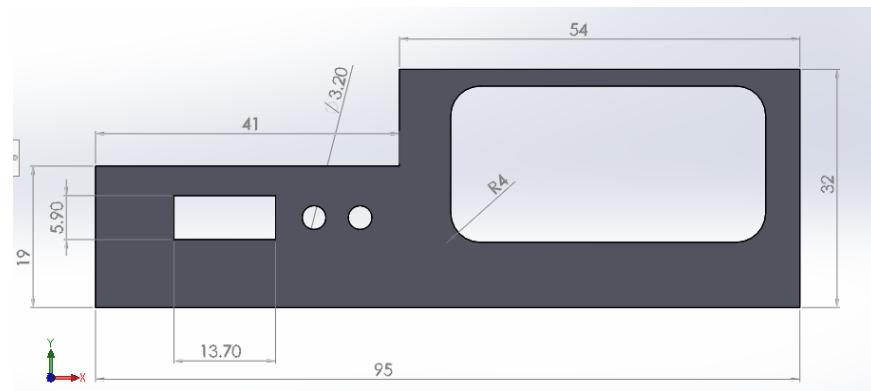


Gambar 72.Gambar Proyeksi Eropa

b. Pembuatan Box

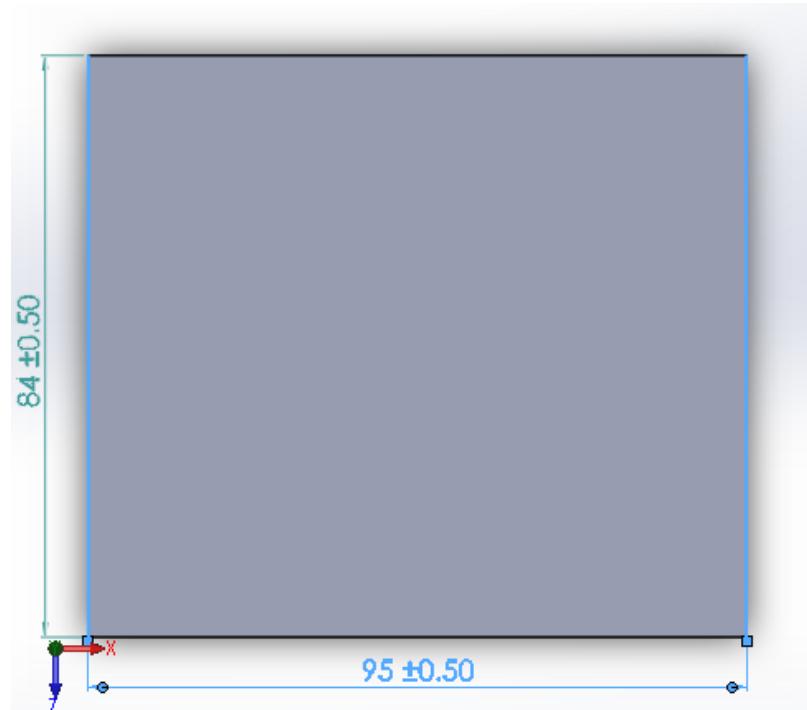
Pembuatan box dilakukan menggunakan bahan acrylic dengan menggunakan laser *cutting* agar mendapat hasil potongan yang lebih rapi dan teliti. Dikarenakan bahan menggunakan acrylic maka box terbagi menjadi beberapa bagian sebagai berikut.

1) Bagian Depan



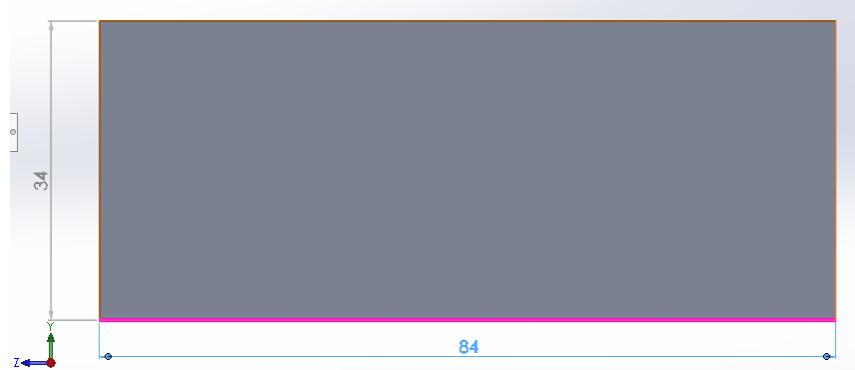
Gambar 73.Bagian depan box

2) Bagian Bawah



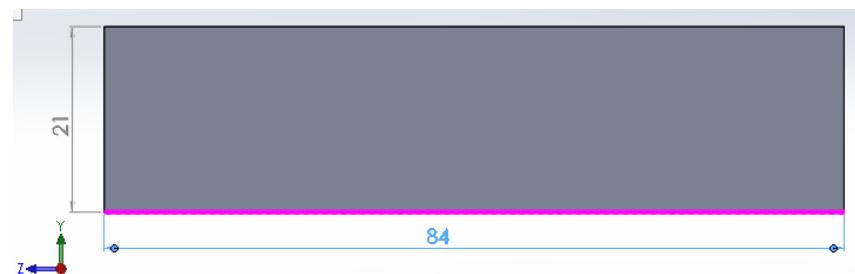
Gambar 74.Bagian bawah box

3) Bagian Samping Kanan



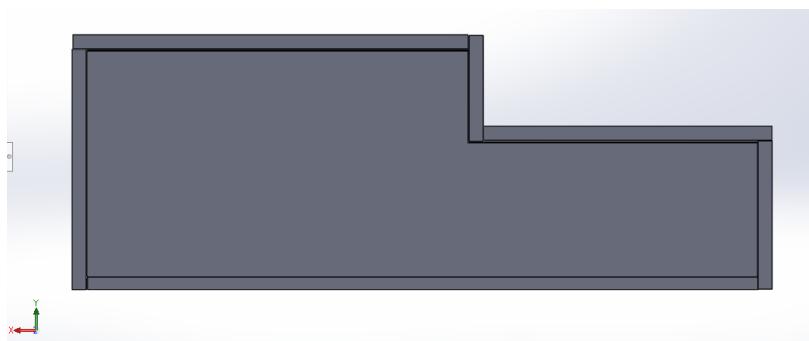
Gambar 75.Bagian samping kanan box

4) Bagian Samping Kiri



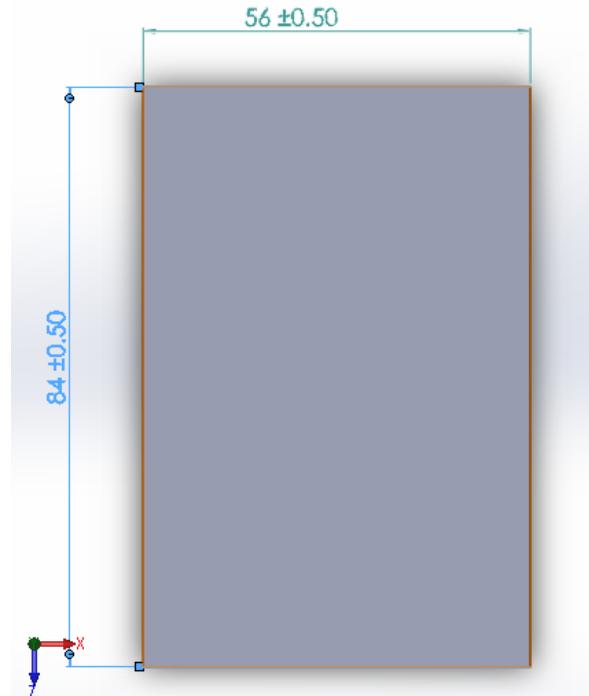
Gambar 76. Bagian samping kiri box

5) Bagian Belakang



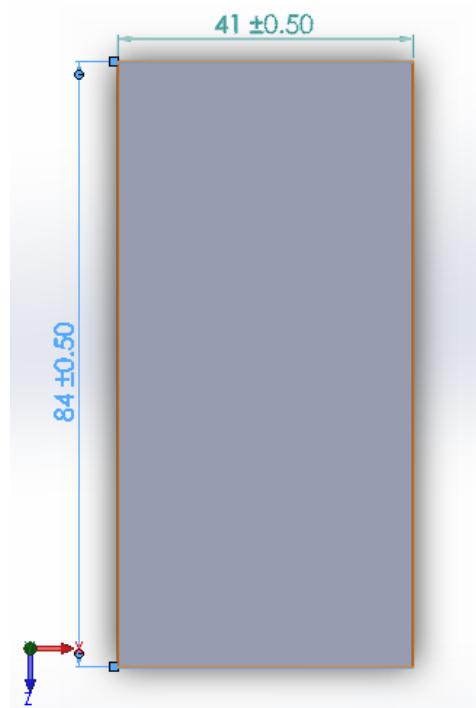
Gambar 77.Bagian belakang box

6) Bagian Atas Kanan



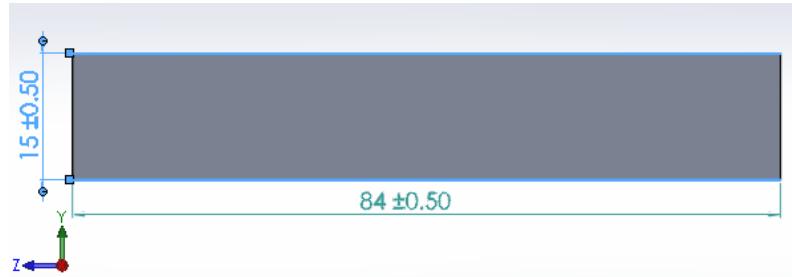
Gambar 78.Bagian atas kanan box

7) Bagian Atas Kiri



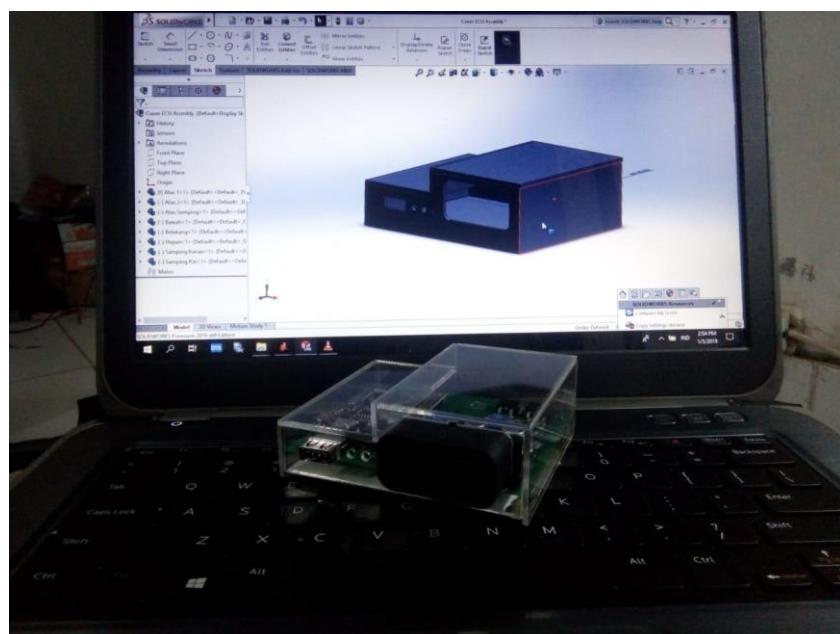
Gambar 79.Bagian atas kiri box

8) Bagian Atas Tengah



Gambar 80.Bagian atas tengah box

Berikut gambar box ketika sudah jadi.



Gambar 81.Box ECU Programmable

6. Pemasangan Rangkaian pada Box

Pemasangan rangkaian pada *box* dilakukan setelah semua bagian rangkaian selesai dirakit dan diuji sudah bisa bekerja dengan baik. Bagian rangkaian yang dirakit di dalam *box* yaitu rangkaian unit ECU Programmable.

F. Pengujian Alat

Pengujian alat dilakukan untuk mendapat data dari ECU Programmable bebasis Atmega2560 yang telah dibuat, Pengujian dilakukan dalam beberapa tahap sebagai berikut:

1. Uji fungsional

Pengujian yang dilakukan yaitu pengujian pada tiap bagian rangkaian. Pengujian ini dilakukan untuk memastikan pada setiap bagian rangkaian bekerja sesuai dengan fungsinya agar tidak terjadi kendala.

2. Uji seluruh sistem

Setelah menguji pada setiap bagian rangkaian hingga sesuai dengan fungsinya, langkah selanjutnya melakukan pengujian untuk seluruh sistem. Pengujian ini dilakukan dengan tujuan untuk mengetahui alat dapat bekerja sesuai fungsinya atau tidak. Hasil dari pembuatan alat tidak hanya mencapai keberhasilan, tetapi juga terdapat kekurangan. Kekurangan ini yang diharapkan dapat diperbaiki di lain kesempatan. Hasil pengujian dan pembahasan terdapat pada BAB IV.