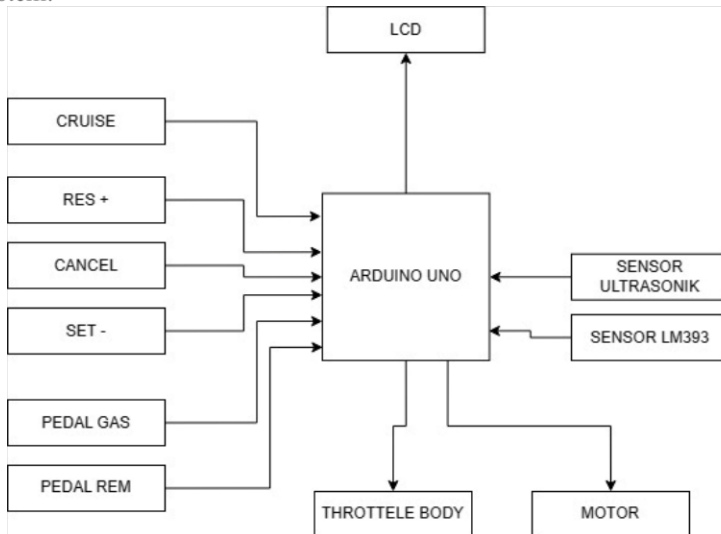


BAB III METODOLOGI DAN PERANCANGAN

3.1 Perancangan *Trainer Sistem Adaptive Cruise Control*

3.1.1 Blok Diagram *Trainer Sistem Adaptive Cruise Control*

Gambar 3.1 dibawah merupakan blok diagram *Trainer Sistem Adaptive Cruise Control* yang menjelaskan siklus atau proses kerja sistem.



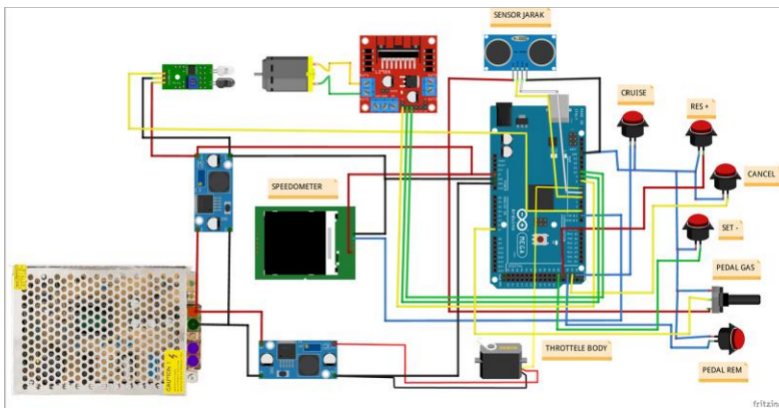
Gambar 3. 1 Blok Diagram *Trainer Sistem Adaptive Cruise Control*

Pada gambar 3.1 dapat dijelaskan sebagai berikut :

- a. Sensor Ultrasonik untuk membaca jarak kendaraan yang ada didepan.
- b. Sensor LM393 untuk membaca kecepatan pada roda.
- c. Potensiometer Geser (Pedal Gas) untuk mengatur kecepatan motor dan mengatur derajat servo.
- d. *Push Button* (Pedal Rem) untuk menonaktifkan sistem adaptive cruise control pada saat sistem sedang aktif.
- e. *Push Button* (Cruise) untuk mengaktifkan sistem adaptive cruise control.

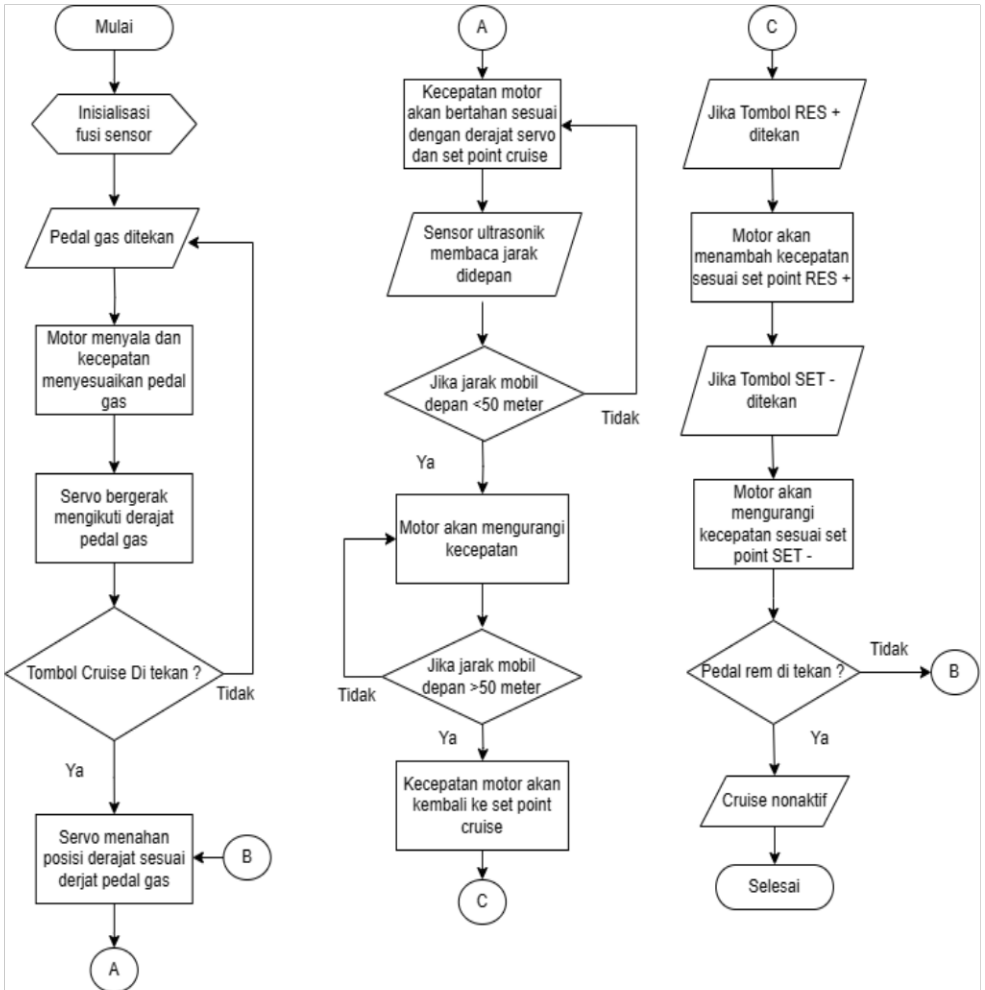
- f. *Push Button* (Res +) untuk menambah kecepatan kendaraan saat sistem adaptive cruise control aktif.
- g. *Push Button* (Set -) untuk mengurangi kecepatan kendaraan saat sistem adaptive cruise control aktif.
- h. *Push Button* (Cancel) untuk menonaktifkan sistem adaptive cruise control pada saat sistem sedang aktif.
- i. Servo MG966 untuk mengatur bukaan throttle body dan mengatur kecepatan motor.
- j. Lcd untuk indikator kecepatan, rpm, indikator *cruise control* aktif atau tidak.

Berikut adalah tampilan *full wiring* pada trainer sistem *adaptive cruise control* yang menggunakan sensor *infrared obstacle* sebagai sensor kecepatan motor, driver L298N sebagai penggerak motor, sensor ultrasonik HC-SR04 sebagai pembaca jarak, LCD Nextion sebagai tampilan speedometer, Arduino Mega 2560 sebagai mikrokontroler, 5 buah *push button* untuk sistem *adaptive cruise control*, servo sebagai penggerak *throttle body*, potensio geser sebagai pedal gas. Dibawah ini adalah tampilan *full wiring* pada trainer sistem *adaptive cruise control* :



Gambar 3. 2 Rangkaian *Full Wiring*

3.1.2 Flowchart Sistem Adaptive Cruise Control



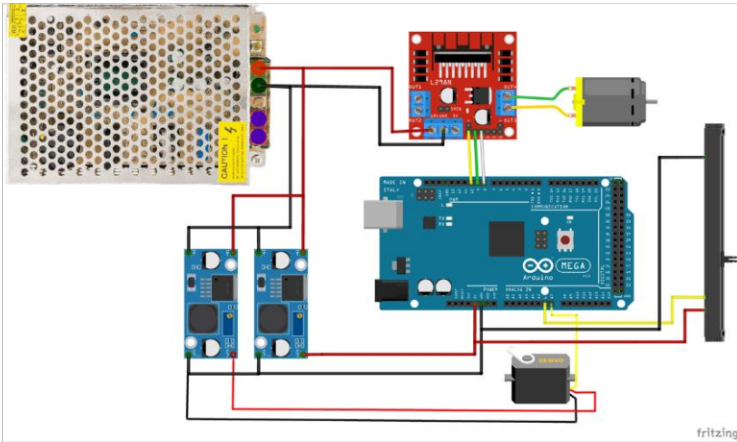
Gambar 3. 3 Flowchart Sistem Adaptive Cruise Control

Pada *Flowchart* di atas yang terdapat pada gambar dapat dijelaskan sebagai berikut :

- a. Pedal gas harus di tekan untuk mengaktifkan seluruh sistem.
- b. Motor menyala ketika pedal gas ditekan dan kecepatannya menyesuaikan posisi pedal gas.
- c. Tombol *Cruise Control* ditekan atau tidak, jika tidak di tekan akan kembali ke poin a dan b.
- d. Ketika tombol *Cruise Control* ditekan maka posisi derajat servo akan mengikuti derajat posisi pedal gas.
- e. Ketika pedal gas tidak di tekan maka servo akan tetap menahan posisi derajat nya sesuai dengan derajat pedal gas yang terakhir. Kecepatan motor akan bertahan sesuai dengan posisi terakhir dari pedal gas.
- b. Ultrasonik akan membaca jarak antara mobil yang sedang dikendari dengan mobil yang ada di depan nya.
- c. Jika ultrasonik membaca jarak mobil didepan kurang dari 100 meter maka motor akan otomatis mengurangi kecepatannya.
- d. Jika mobil didepan Kembali melaju atau jarak nya lebih dari 100 meter maka motor akan kembali ke kecepatan awal.
- e. Ketika tombol + ditekan maka posisi derajat servo akan bertambah dan posisi derajat servo terakhir dari pedal gas sudah tidak berlaku dan kecepatan motor akan bertambah.
- f. Ketika tombol – ditekan maka derajat posisi servo akan berkurang dan kecepatan motor akan berkurang.
- g. Ketika pedal rem di tekan maka akan menonaktifkan sistem *Adaptive Cruise Control*
- h. Jika pedal rem tidak ditekan maka sistem *Adaptive Cruise Control* masih tetap berjalan.

3.2 Perancangan Kontrol Servo dan Motor DC

Pada perancangan kontrol servo dan motor ini bertujuan untuk mengatur kecepatan motor dan posisi derajat servo menggunakan sebuah potensiometer geser sebagai pedal gas nya. Komponen yang di gunakan untuk perancangan ini yaitu motor dc, driver L298N, servo MG996, potensiometer geser. Untuk wiring nya dapat di lihat pada gambar 3.3



Gambar 3. 4 Perancangan Kontrol Servo dan Motor

Konfigurasi dari perancangan koneksi servo dan motor dc dengan arduino mega dapat dilihat pada table 3.1 sebagai berikut :

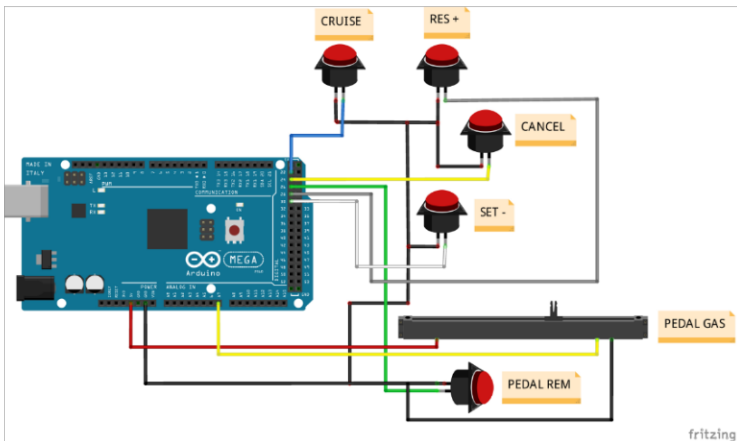
Tabel 3. 1 Konfigurasi Koneksi Servo dan Motor DC pada Arduino Mega

| Komponen | Koneksi | | | |
|--------------|--------------|--------------|------------------|---------------------|
| | Arduino Mega | Step Down | Power Supply 12V | Warna Jalur Koneksi |
| Servo | Arduino Mega | Step Down | Power Supply 12V | Warna Jalur Koneksi |
| VCC | - | OUT + 5V | - | Merah |
| Signal | Pin A6 | - | - | Kuning |
| GND | GND | OUT – GND | - | Hitam |
| Driver L298N | Arduino Mega | Step Down | Power Supply 12V | Warna Jalur Koneksi |
| VCC | - | - | 12 V | Merah |
| GND | GND | - | GND | Hitam |
| EN A | Pin D8 | - | - | Kuning |
| IN 1 | Pin D9 | - | - | Hijau |
| IN 2 | Pin D10 | - | - | Putih |
| Motor DC | - | Driver L298N | - | Warna Jalur Koneksi |

| | | | | |
|---------------------|---|--------------|---|---------------------|
| VCC | - | OUT 3 | - | Merah |
| GND | - | OUT 4 | - | Hitam |
| Potensiometer Geser | - | Arduino Mega | - | Warna Jalur Koneksi |
| Pin 1 | - | 5V | - | Merah |
| Pin 2 | - | GND | - | Hitam |
| Pin 3 | - | Pin A7 | - | Kuning |

3.3 Perancangan Pengaktifan Sistem *Adaptive Cruise Control*

Pada perancangan pengaktifan sistem *Adaptive Cruise Control* ini bertujuan untuk mengaktifkan atau menonaktifkan sistem *Adaptive Cruise Control* menggunakan push button. Komponen yang di gunakan untuk perancangan ini yaitu push button. Untuk wiring nya dapat di lihat pada gambar 3.4



Gambar 3. 5 Perancangan Pengaktifan Sistem Adaptive Cruise Control

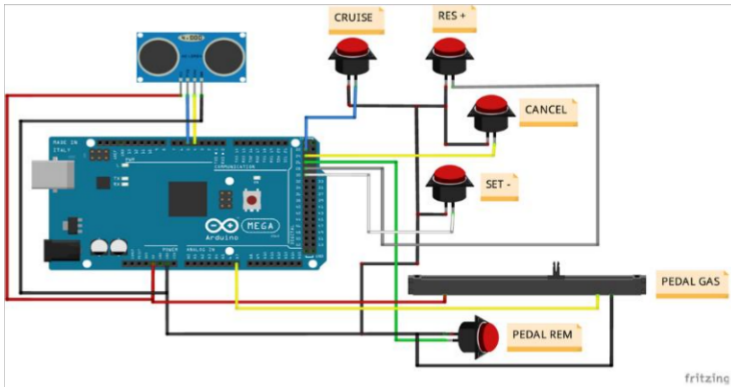
Konfigurasi dari perancangan pengaktifan sistem adaptive cruise control dengan arduino mega dapat dilihat pada table 3.2 sebagai berikut :

Tabel 3. 2 Konfigurasi Perancangan Pengaktifan Sistem Adaptive Cruise Control

| Komponen | Koneksi | |
|---------------------|--------------|---------------------|
| Button Cruise | Arduino Mega | Warna Jalur Koneksi |
| Digital Output | Pin D22 | Biru |
| GND | GND | Hitam |
| Button Cancel | Arduino Mega | Warna Jalur Koneksi |
| Digital Output | Pin D24 | Kuning |
| GND | GND | Hitam |
| Button Res + | Arduino Mega | Warna Jalur Koneksi |
| Digital Output | Pin D28 | Abu-Abu |
| GND | GND | Hitam |
| Button Set - | Arduino Mega | Warna Jalur Koneksi |
| Digital Output | Pin D30 | Putih |
| GND | GND | Hitam |
| Button Rem | Arduino Mega | Warna Jalur Koneksi |
| Digital Output | Pin D24 | Hijau |
| GND | GND | Hitam |
| Potensiometer Geser | Arduino Mega | Warna Jalur Koneksi |
| Pin 1 | 5V | Merah |
| Pin 2 | GND | Hitam |
| Pin 3 | Pin A7 | Kuning |

3.4 Perancangan Pembaca Jarak

Pada perancangan pembaca jarak ini bertujuan untuk membaca jarak antara kendaraan yang kita kendarai dengan kendaraan yang ada didepan. Komponen yang digunakan untuk perancangan ini yaitu sensor ultrasonik HC-SR04. Untuk wiring nya dapat di lihat pada gambar 3.5



Gambar 3. 6 Perancangan Pembaca Jarak

Konfigurasi dari perancangan pembaca jarak dengan Arduino mega dapat dilihat pada table 3.3 sebagai berikut :

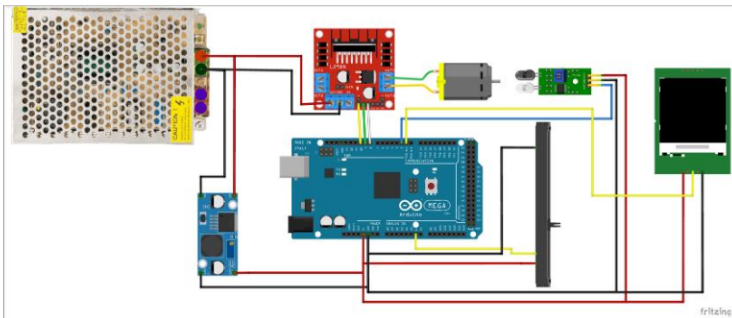
Tabel 3. 3 Konfigurasi Perancangan Pembaca Jarak

| Komponen | Koneksi | |
|----------------|--------------|---------------------|
| Button Cruise | Arduino Mega | Warna Jalur Koneksi |
| Digital Output | Pin D22 | Biru |
| GND | GND | Hitam |
| Button Cancel | Arduino Mega | Warna Jalur Koneksi |
| Digital Output | Pin D24 | Kuning |
| GND | GND | Hitam |
| Button Res + | Arduino Mega | Warna Jalur Koneksi |
| Digital Output | Pin D28 | Abu-Abu |
| GND | GND | Hitam |
| Button Set - | Arduino Mega | Warna Jalur Koneksi |
| Digital Output | Pin D30 | Putih |
| GND | GND | Hitam |
| | | |

| | | |
|---------------------|--------------|---------------------|
| Button Rem | Arduino Mega | Warna Jalur Koneksi |
| Digital Output | Pin D26 | Hijau |
| GND | GND | Hitam |
| Potensiometer Geser | Arduino Mega | Warna Jalur Koneksi |
| Pin 1 | 5V | Merah |
| Pin 2 | GND | Hitam |
| Pin 3 | Pin A7 | Kuning |
| Sensor Ultrasonik | Arduino Mega | Warna Jalur Koneksi |
| VCC | 5V | Merah |
| TRIG | Pin D5 | Biru |
| ECHO | Pin D4 | Kuning |
| GND | GND | Hitam |

3.5 Perancangan Pembaca Kecepatan Motor

Pada perancangan pembaca kecepatan motor ini bertujuan untuk membaca kecepatan motor dan rpm pada motor yang sedang bergerak. Komponen yang digunakan pada perancangan ini yaitu sensor *infrared obstacle* sebagai pembaca kecepatan motor, Lcd nextion sebagai penampil hasil dari bacaan sensor infrared obstacle. Untuk wiring nya dapat di lihat pada gambar 3.6



Gambar 3. 7 Perancangan Pembaca Kecepatan Motor

Konfigurasi dari perancangan pembaca kecepatan motor dengan Arduino mega dapat dilihat pada table 3.4 sebagai berikut :

Tabel 3. 4 Konfigurasi Perancangan Pembaca Kecepatan Motor

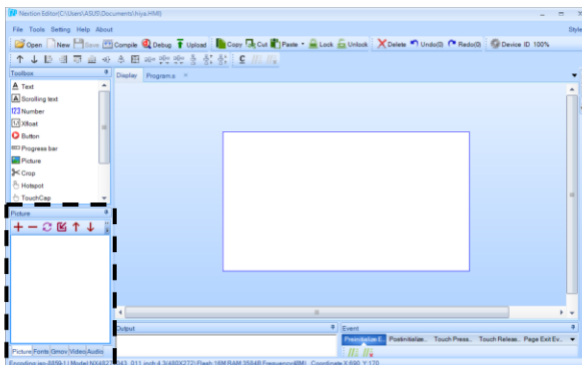
| Komponen | Koneksi | | | |
|---------------------------------|---------------------|---------------------|-------------------------|----------------------------|
| Sensor Infrared Obstacle | Arduino Mega | Step Down | Power Supply 12V | Warna Jalur Koneksi |
| VCC | 5V | OUT + 5V | - | Merah |
| Out | Pin D2 | - | - | Kuning |
| GND | GND | OUT – GND | - | Biru |
| Driver L298N | Arduino Mega | Step Down | Power Supply 12V | Warna Jalur Koneksi |
| VCC | - | - | 12 V | Merah |
| GND | GND | - | GND | Hitam |
| EN A | Pin D8 | - | - | Kuning |
| IN 1 | Pin D9 | - | - | Hijau |
| IN 2 | Pin D10 | - | - | Putih |
| Motor DC | - | Driver L298N | - | Warna Jalur Koneksi |
| VCC | - | OUT 3 | - | Merah |
| GND | - | OUT 4 | - | Hitam |
| Potensiometer Geser | - | Arduino Mega | - | Warna Jalur Koneksi |
| Pin 1 | - | 5V | - | Merah |
| Pin 2 | - | GND | - | Hitam |
| Pin 3 | - | Pin A7 | - | Kuning |
| Lcd Nextion | - | Arduino Mega | - | Warna Jalur Koneksi |
| VCC | - | 5V | - | Merah |
| GND | - | GND | - | Hitam |
| RX | - | Pin D2 | - | Kuning |

3.6 Perancangan *Interface* Pada LCD Nextion

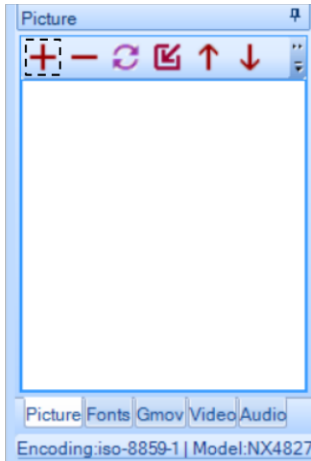
Pada perancangan *interface* yang tampil pada LCD Nextion untuk menunjukkan kecepatan rpm motor, kecepatan km/j motor, sistem *cruise control* sudah aktif atau belum, serta pembaca jarak antar kendaraan di depan, maka digunakan lah *software* untuk membuat *interface* tersebut bernama *Nextion Editor*.

Nextion Editor adalah perangkat lunak yang dirancang khusus untuk memprogram dan merancang antarmuka pengguna pada layar LCD Nextion. Dengan menggunakan *Nextion Editor*, pengguna dapat membuat antarmuka yang menarik dan interaktif dengan mudah, tanpa memerlukan pengetahuan pemrograman yang mendalam. Berikut adalah cara pembuatan *interface* melalui *software Nextion Editor* :

1. Langkah awal masukan gambar yang sebelumnya telah dibuat kedalam *software Nextion Editor* dengan cara pada kotak *picture* pojok kiri bawah klik logo + untuk memasukan gambar yang telah di buat

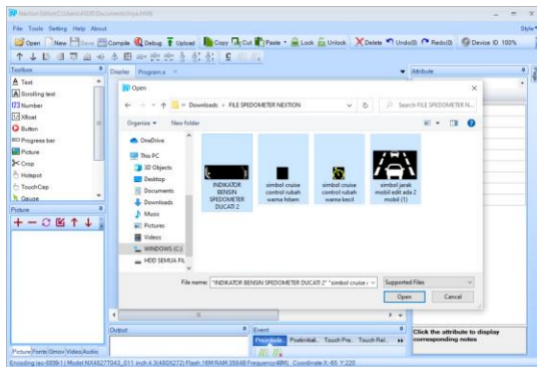


Gambar 3. 8 Tampilan utama mendesain lcd



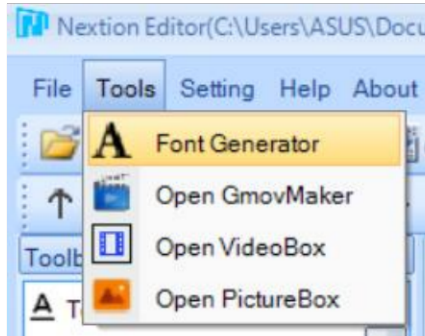
Gambar 3. 9 Tampilan untuk menambahkan foto

2. Lalu cari lokasi file penyimpanan gambar yang telah dibuat sebelumnya, dan blok gambar yang dipilih lalu klik open



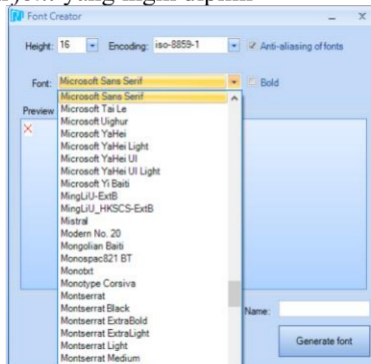
Gambar 3. 10 Tampilan pemilahan gambar

3. Kemudian klik *Tools* pada pojok kiri atas lalu klik *font generator* untuk memasukan *font* atau tulisan kedalam *Nextion Editor*



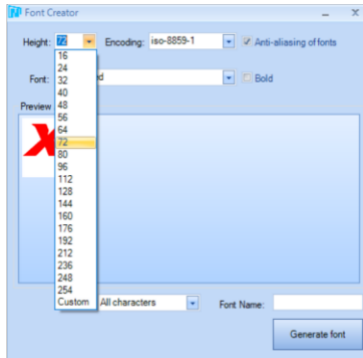
Gambar 3. 11 Tampilan toolbar nembahkan teks

4. Kemudian pilih *font* yang ingin dipilih



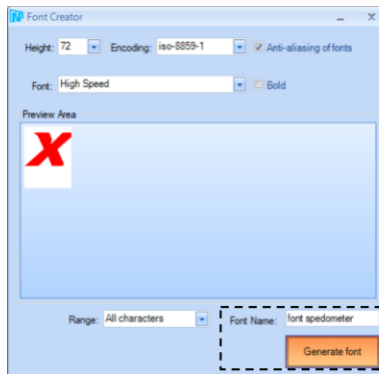
Gambar 3. 12 Tampilan pemilihan jenis teks

5. Kemudian pilih ukuran *font* yang ingin digunakan, untuk tugas akhir ini menggunakan font ukuran 72



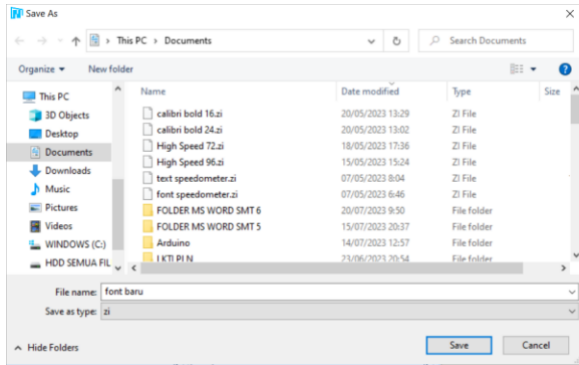
Gambar 3. 13 Tampilan pemilihan ukuran teks

6. Kemudian masukan nama file untuk *font* yang telah di pilih pada kotak *Font Name* untuk disimpan kedalam file komputer, lalu klik *Generate font*



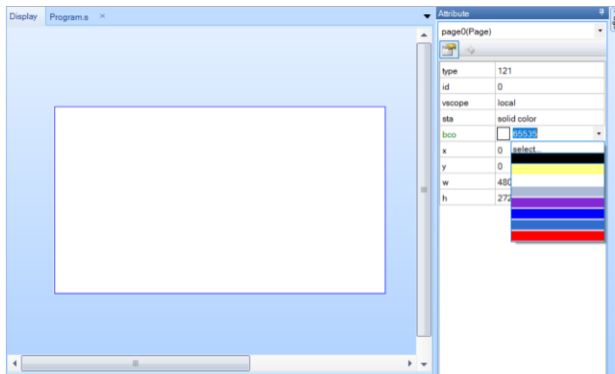
Gambar 3. 14 Tampilan Genrate font

7. Kemudian tentukan lokasi penyimpanan *font* yang telah dibuat tadi, lalu klik *save*



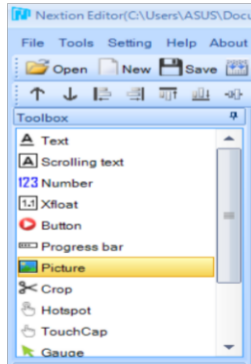
Gambar 3. 15 Tampilan lokasi penyimpanan teks

8. Kemudian ulangi seperti langkah nomor 8 – 12 dengan ukuran *font* nya 16
9. Kemudian rubah warna *background* menjadi warna hitam dengan cara klik pada kotak bco pojok kanan lalu pilih warna hitam



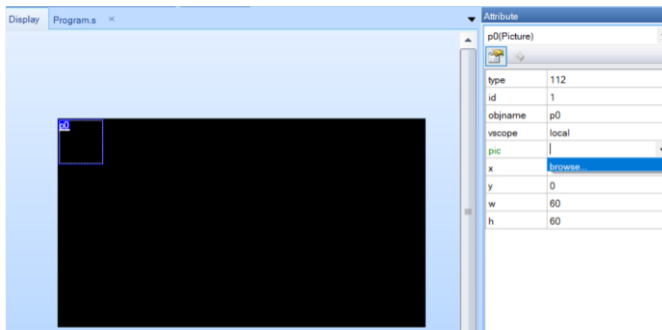
Gambar 3. 16 Tampilan merubah warna background

10. Kemudian klik *Picture* pada kotak *Toolbox* yang berada pada pojok kiri atas

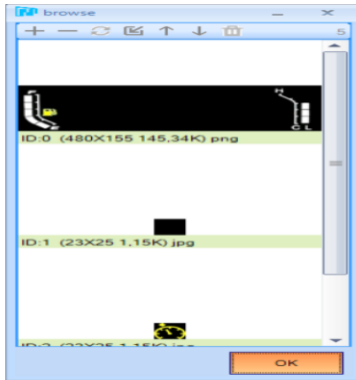


zGambar 3. 17 Tampilan toolbar picture

11. Kemudian pada kotak *Attribute* yang berada pada pojok kanan cari tulisan *pic* lalu klik pada kotak kosong yang berada disebelah tulisan *pic* lalu pilih *browser* kemudian pilih foto yang ingin digunakan lalu klik ok.

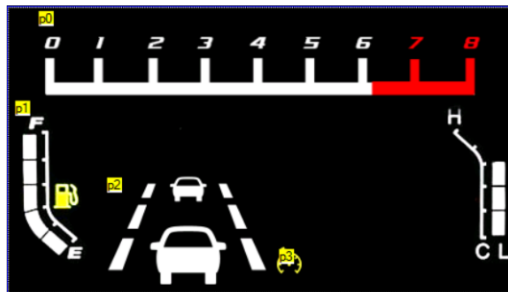


Gambar 3. 18 Tampilan toolbar picture pada halaman desain



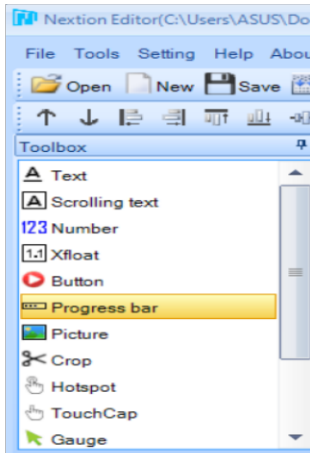
Gambar 3. 19 Tampilan toolbar browser picture

12. Lakukan langkah 15 – 16 sampai seperti pada gambar di bawah ini



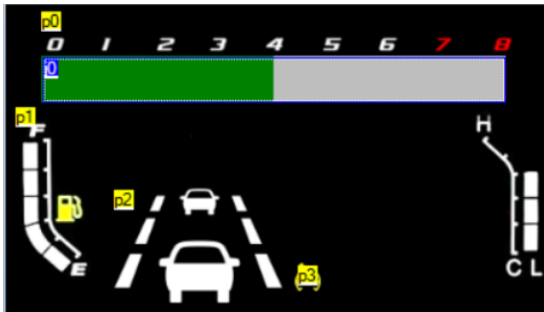
Gambar 3. 20 Tampilan desain yang sudah disusun

13. Kemudian klik *Progres bar* pada kotak *Toolbox* yang berada pada pojok kiri atas



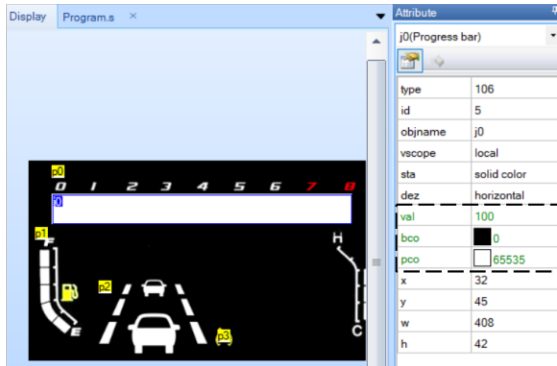
Gambar 3. 21 Tampilan toolbar progress bar

14. Kemudian letakan seperti pada gambar dibawah ini



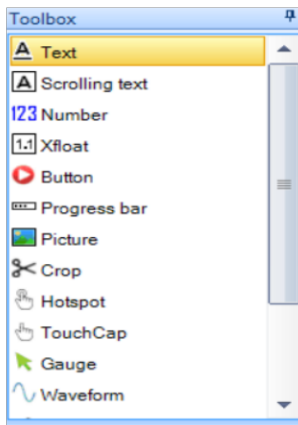
Gambar 3. 22 Tampilan setelah ditambahkan progress bar

15. Kemudian pada kotak *Attribute* rubahlah *val* menjadi 100, kemudian *bco* menjadi hitam, lalu *pco* menjadi putih.



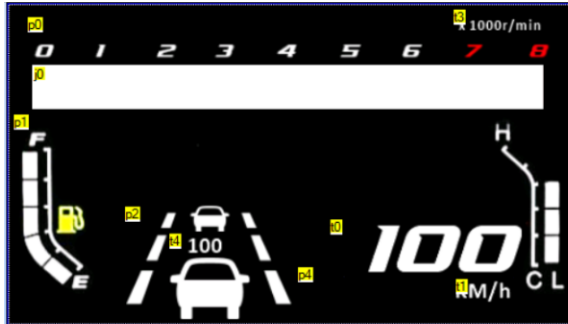
Gambar 3. 23 Tampilan merubah warna progress bar

16. Kemudian klik *Text* pada kotak *Toolbox* untuk memasukan tulisan



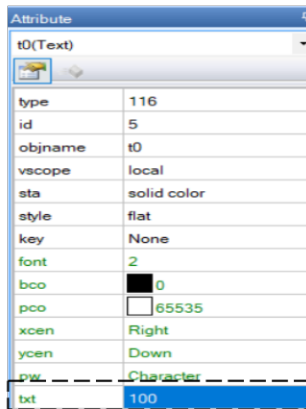
Gambar 3. 24 Tampilan toolbar text

17. Kemudian buatlah seperti pada gambar di bawah ini

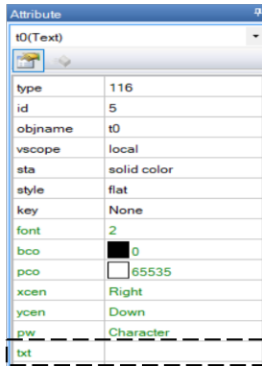


Gambar 3. 25 Tampilan setelah ditambahkan teks

18. Kemudian hilangkan angka 100 dengan cara pada kotak *Attribute* cari tulisan *txt* lalu *double* klik pada angka 100 kemudian hapus lalu tekan *enter*

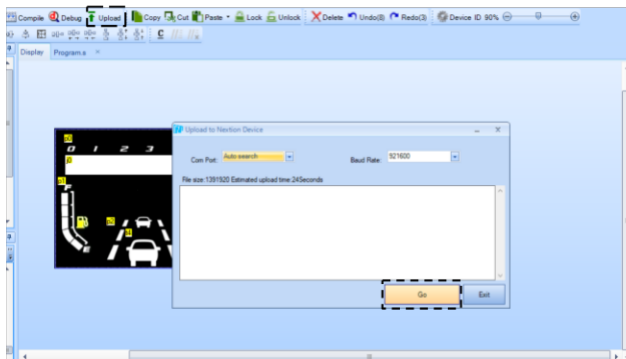


Gambar 3. 26 Tampilan attribute menghapus teks



Gambar 3. 27 Tampilan setelah teks dihapus

19. Hubungkan LCD Nextion dengan modul USB to TTL untuk melakukan upload gambar yang telah dibuat
20. Kemudian klik tulisan *Upload* pada bar yang ada diatas, lalu klik *Go* untuk melakukan upload gambar.



Gambar 3. 28 Tampilan untuk proses upload desain ke lcd

3.7 Perancangan Bahan Pembuatan Casing

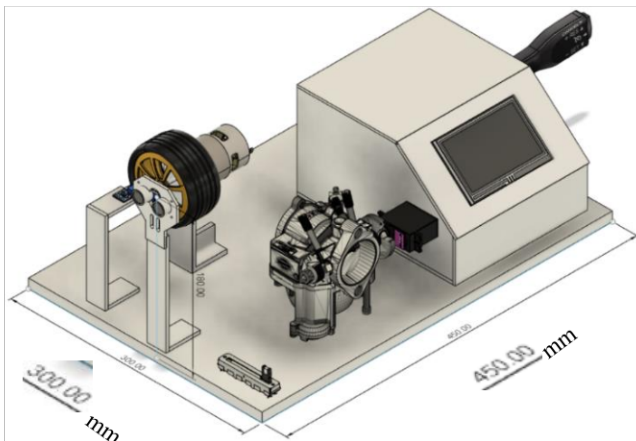
Pada perancangan bahan pembuatan casing ini ada beberapa bahan yang digunakan, antara lain :

1. Casing LCD Nextion : pada pembuatan casing LCD Nextion menggunakan printer 3D yang berukuran Panjang 16,3 cm x Lebar 18,7 cm x Tinggi 11,7 cm .
2. *Bracket* atau pegangan sensor ultrasonik : pada pembuatan *Bracket* atau pegangan sensor ultrasonik menggunakan akrilik berukuran 5 mm dengan ukuran Panjang 4 cm x Lebar 2,5 cm x Tinggi 16 cm.
3. *Bracket* atau pegangan sensor *infrared obstacle* : pada pembuatan *bracket* atau pegangan sensor *infrared obstacle* menggunakan akrilik berukuran 5 mm dengan ukuran Panjang 4,6 cm x Lebar 2 cm x Tinggi 11,5 cm.

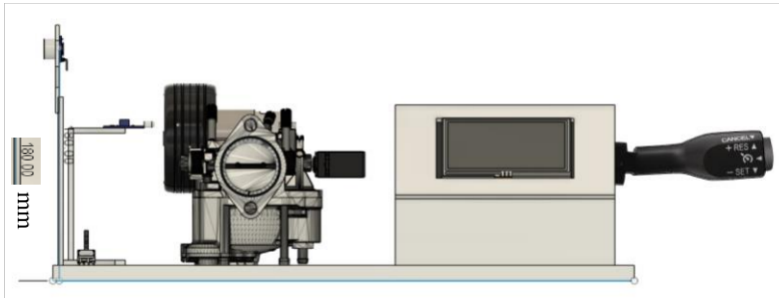
3.8 Perancangan Perangkat Keras

3.8.1 Perancangan Mekanik

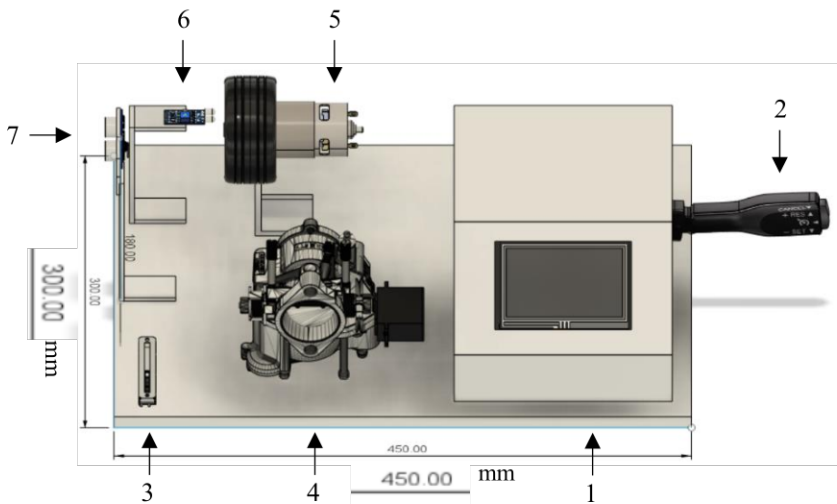
Berikut adalah perancangan mekanik alat yang dibuat meliputi tampak depan, tampak atas, dan tampak samping dari alat beserta keterangan ukuran dalam satuan milimeter. Berikut adalah perancangan mekanik trainer sistem *Adaptive Cruise Control* dapat dilihat pada Gambar 3.20 Gambar 3.21, Gambar 3.22 dan Gambar 3.23



Gambar 3. 29 Tampak samping



Gambar 3. 30 Tampak depan



Gambar 3. 31 Tampak atas

Berikut adalah penjelasan dari penomoran pada gambar diatas :

1. LCD Nextion digunakan sebagai tampilan speedometer.
2. Tombol Cruise Control digunakan untuk mengaktifkan sistem *Adaptive Cruise Control*, menambah kecepatan, mengurangi kecepatan, dan menonaktifkan sistem *Adaptive Cruise Control*.

3. Potensiometer digunakan sebagai pedal gas untuk mengatur posisi servo.
4. Servo digunakan untuk mengatur buka dan tutup *throttle body* yang akan menggerakkan motor
5. Motor digunakan sebagai penggerak roda
6. Sensor *Infrared Obstacle* digunakan sebagai pembaca kecepatan motor.
7. Sensor ultrasonik digunakan sebagai pendeteksi jarak kendaraan yang ada didepan.

~ Halaman ini sengaja dikosongkan ~