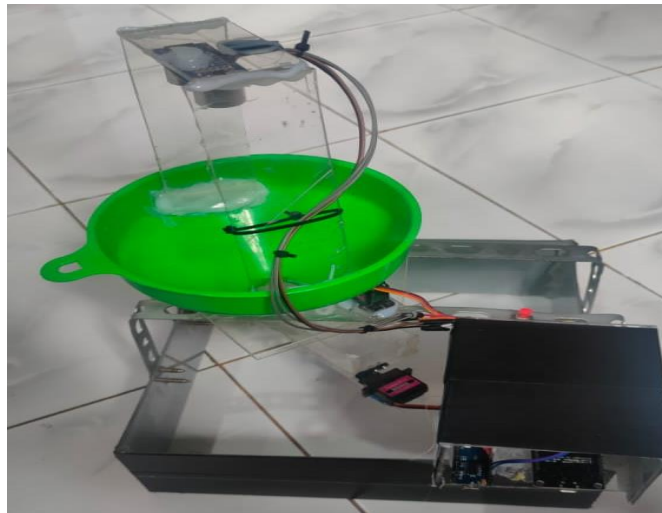


## BAB V

### PENGUJIAN SISTEM DAN IMPLEMENTASI

#### 5.1 HASIL IMPLEMENTASI

Pada tahap ini penulis mengimplementasikan hasil perancangan yang telah dibuat. Adapun hasil implementasi dari penelitian ini adalah sebagai berikut



Gambar 5.1 Prototype pemberian pakan ikan

Gambar di atas merupakan *prototype* alat pemberian pakan ikan berbasis IoT yang telah dirancang oleh penulis. Yang mana pada sistem tersebut terdapat NodeMCU, motor servo, Ultrasonic HC-SR04, ATmega328, dan rangkaian keseluruhan sistem pemberian pakan ikan berbasis IoT tersebut . digunakan telegram untuk menampilkan perintah menghidupkan dengan jarak jauh serta menampilkan pesan yang di kirim pada alat tersebut

## 5.2 PENGUJIAN SISTEMM

Alat yang telah dirancang harus diuji baik dari segi perangkat keras maupun perangkat lunak,tujuannya untuk mengetahui sejauh mana keberhasilan alat yang telah dirancang serta membandingkan apakah sudah sesuai dengan *spesifikasi* yang di inginkan.

Pengujian alat atau perangkat keras dilakukan dengan berbagai cara tergantung dengan alat apa yang diuji. sedangkan pengujian perangkat lunak dalam hal ini beupa program dilakukan dengan *software* arduino ide,setelah itu akan diadakan pengujian program menggunakan alat yang telah dibuat.

Sebelum melakukan pengujian ada baiknya dilakukan pengecekan terlebih dahulu apakah alat yang dirancang telah benar,pemeriksaan ini bisa secara fisik atau *visual* dengan melihat atau membandingkan alat dengan rangkaian, pemeriksaan sebelum pengujian sangat penting untuk meminimalisir keruskan.

Pengujian pemberian pakan ikan berbasis IoT ini memiliki bebrapa tahap, tahap ini bertujuan untuk memperkecil kemungkinan alat tidak bekerja saat di lakukan uji coba atau perbedaan hasil yang di inginkan.

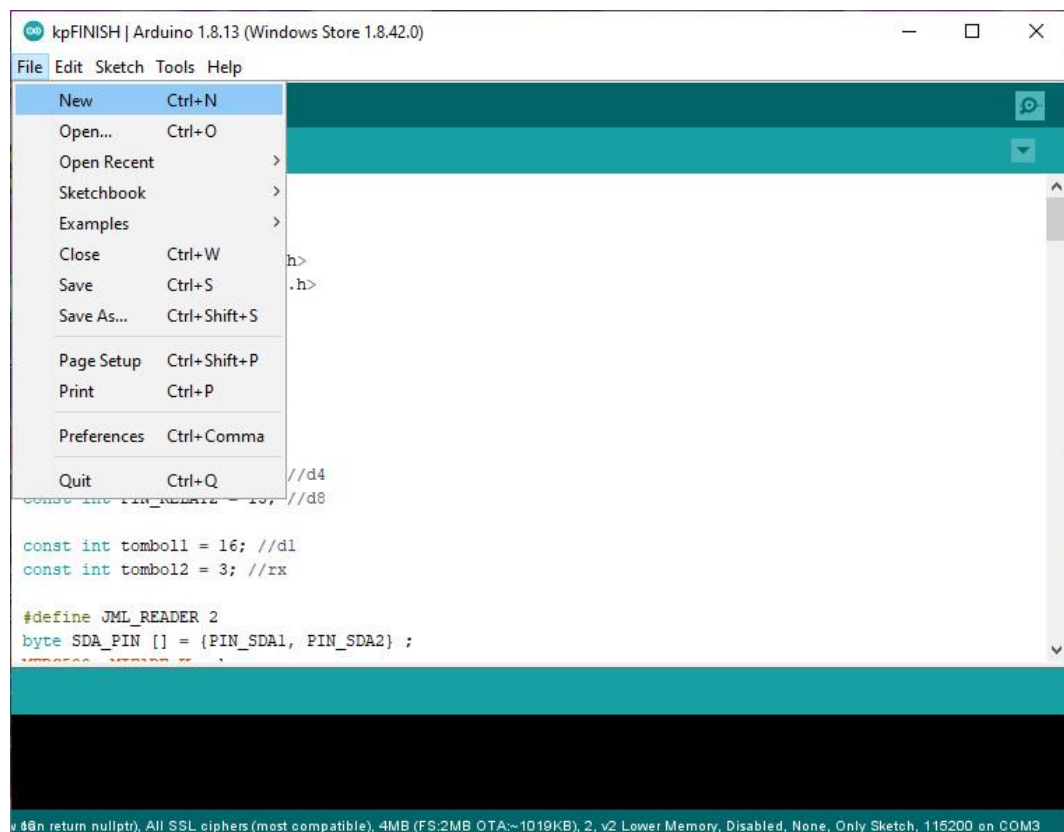
1. Pengujian dengan perangkat Lunak (*Software*)
2. Pengujian dengan perangkat Keras(*Hardware*)

### 5.2.1 Pengujian Perangkat Lunak (Software)

Hal yang dilakukan pertama kali dalam pengujian perangkat lunak (*software*) adalah menentukan aplikasi (*software*) yang akan digunakan,untuk mikrokontroler dengan menggunakan apklikasi Arduino ide.

### 5.2.1.1 Arduino

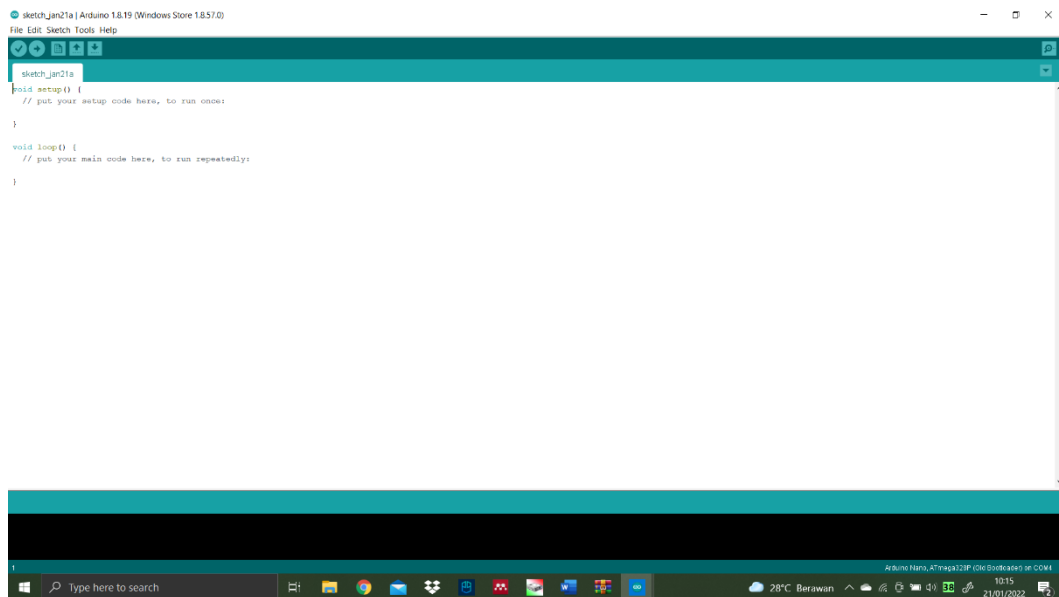
Untuk mikrokontroller NodeMCU sinkron dengan bahasa pemrograman seperti bahasa C, dalam pembuatan alat ini penulis menggunakan software Arduino IDE v1.8.13, software Arduino IDE ini digunakan untuk memprogram mikrokontroller arduino. Pengujian software berikutnya yaitu pembuatan listing program baru klik file new atau tekan CTRL+N. Maka dapat lihat seperti gambar :



Gambar 5.2 Menu membuat program baru

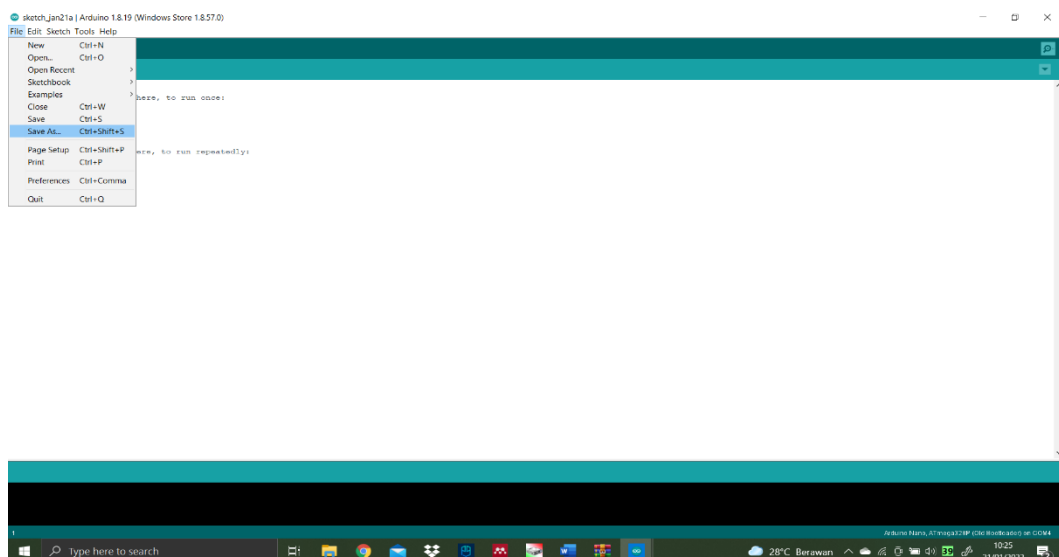
Setelah menyelesaikan listing program untuk membaca Sensor HMC5883L, save program yang telah dibuat dan kita compile terlebih dahulu jika program tidak ada pesan error maka listing program siap di-Upload seperti gambar 5.2 :

Maka selanjutnya akan muncul halaman yang baru dengan dua perintah program :



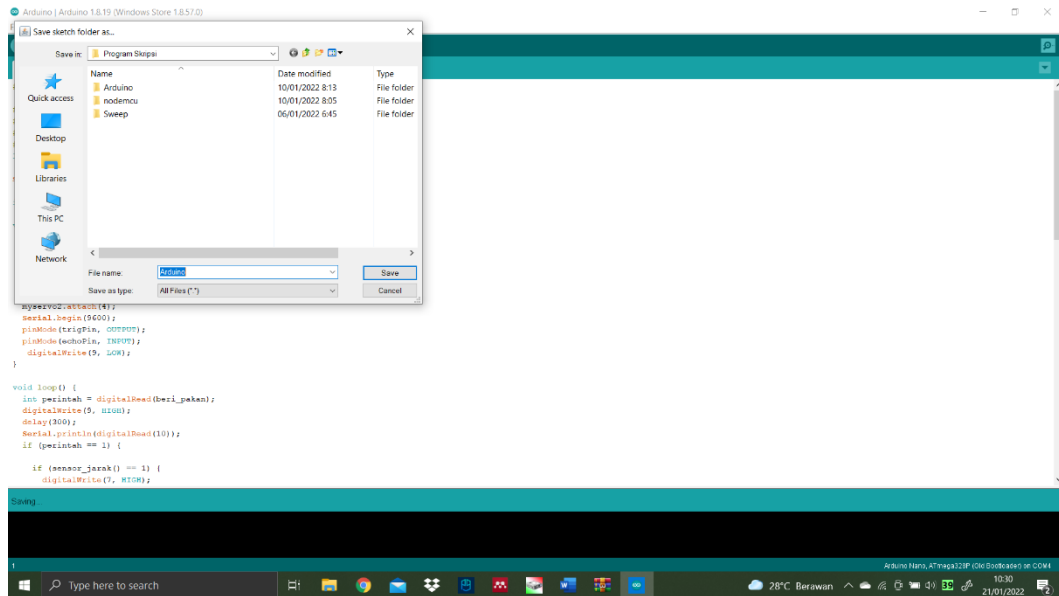
Gambar 5.3 Tampilan Arduino IDE

Kemudian Ketika sudah menulis listing programnya maka file haruslah di simpan terlebih dahulu dengan memilih file kemudian pilih save as, bertujuan untuk menepatkan program di folder yang di inginkan :



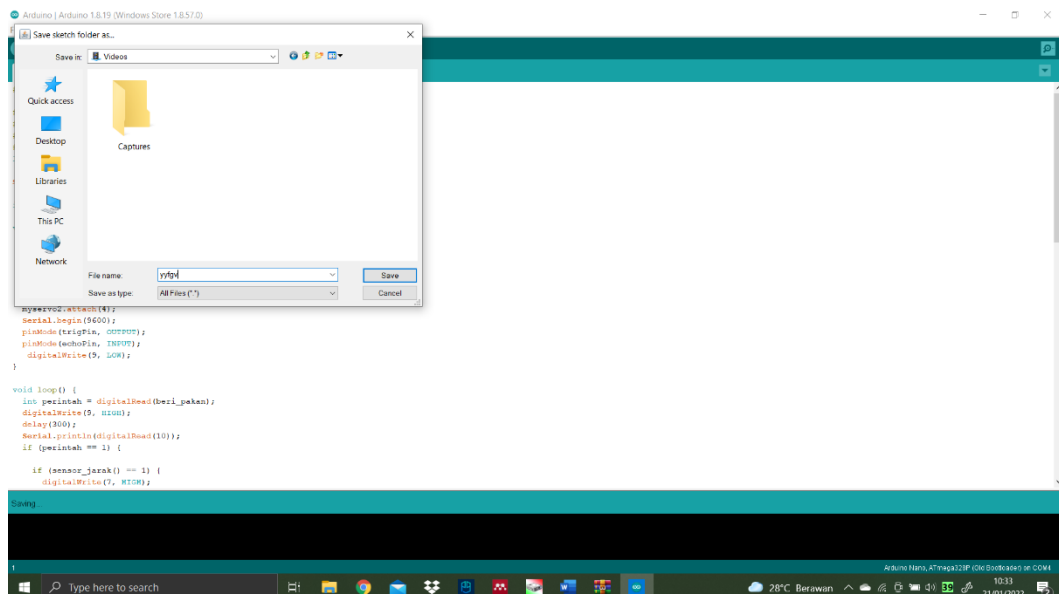
Gambar 5.4 Tampilan Save as

Berikutnya muncul jendela baru untuk menyimpan listing program pada folder yang diinginkan perhatikan gambar berikut :



Gambar 5.5 Pemilihan folder save

Sebelum menekan tombol save pada jendela yang muncul, berikan nama pada listing program yang ingin di simpan dengan proyek yang di buat.



Gambar 5.6 pemberian nama proyek

Setelah proses penyimpanan selesai maka hendaklah memverifikasi program sebelum di uploading ke mikrokontroler, caranya dengan mengklik tombol centang pada tolls yang di sediakan Arduino IDE :

The screenshot displays the Arduino IDE environment. The top window shows a sketch with the following code:

```

#include <Servo.h>

#define beri_pakan 6
#define deteksi_pakan 7
#define subPin 2
#define trigPin 3
long duration, distanca;

Servo myServo, myServo2;

int pos = 0;

void setup() {
  pinMode(deteksi_pakan, OUTPUT);
  pinMode(beri_pakan, INPUT);
  pinMode(5, OUTPUT);
  pinMode(10, INPUT);
  myServo.write(170);
  myServo.attach(10);
  myServo2.attach(4);
  Serial.begin(9600);
  pinMode(trigPin, OUTPUT);
  pinMode(subPin, INPUT);
  digitalWrite(5, LOW);
}

void loop() {
  int perintah = digitalRead(beri_pakan);
  digitalWrite(5, HIGH);
  delay(200);
}

```

The bottom window shows the compilation output, which includes the following text:

```

Done compiling.
"C:\Program Files\WindowsApps\ArduinoLC.ArduinoIDE_1.8.57.0_x86_mdqqa336fwt\hardware\tools\avr-gcc" -x -c -g -fno-common -mmcu=atmega328p -o "C:\Users\ACER\AppData\Local\Temp\arduino_build_521856\Arduino.ino.o" "C:\Program Files\WindowsApps\ArduinoLC.ArduinoIDE_1.8.57.0_x86_mdqqa336fwt\hardware\tools\avr-gcc" -c -g -mmcu=atmega328p -o "C:\Users\ACER\AppData\Local\Temp\arduino_build_521856\Arduino.ino.o" "C:\Program Files\WindowsApps\ArduinoLC.ArduinoIDE_1.8.57.0_x86_mdqqa336fwt\libraries\Servo\Servo.cpp"
Sketch uses 402 bytes (15%) of program storage space. Maximum is 2620 bytes.
Global variables use 24 bytes (11%) of dynamic memory, leaving 1014 bytes for local variables. Maximum is 2048 bytes.

```

Gambar 5.7 Tampilan verifikasi

Jika setelah melakukan verifikasi pada listing program dan terdapat *library* atau *source code* yang tidak cocok maka akan muncul pemberitahuan pada Arduino IDE bahwa *source code* yang di tulis salah kemudian terdapat pada pemberitahuan *source code* mana yang salah akan di *block* dengan warna merah :

```

Arduino | Arduino 1.8.19 (Windows Store 1.8.57.0)
File Edit Sketch Tools Help

#include <servo.h>

#define beri_pakan 6
#define deteksi_pakan 7
#define sensor 2
#define trigPin 3
long duration, distance;

Servo myservo, myservo2;

int pos = 0;

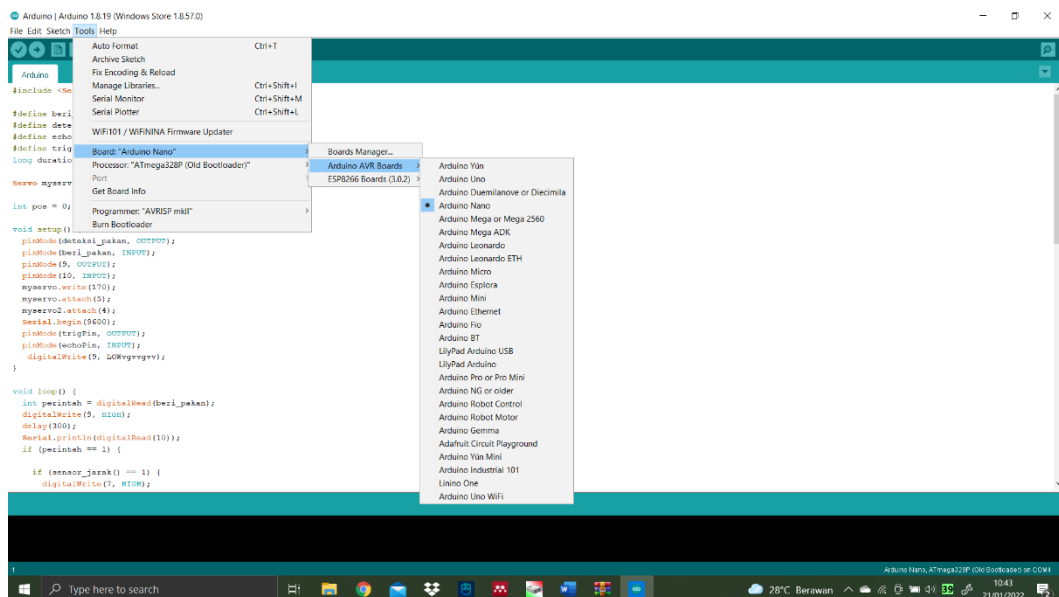
void setup() {
  pinMode(deteksi_pakan, OUTPUT);
  pinMode(beri_pakan, INPUT);
  pinMode(5, OUTPUT);
  pinMode(10, INPUT);
  myservo.write(170);
  myservo.attach(5);
  myservo2.attach(4);
  Serial.begin(9600);
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  digitalWrite(0, LOW);
}

void loop() {
  int perintah = digitalRead(beri_pakan);
  digitalWrite(0, HIGH);
  delay(100);
  LOWRegister was not declared in this scope
  Copy error message
  C:\Users\Aldi\Desktop\Program Ortopedi\Arduino\Arduino.ino: in function 'void setup()':
  Arduino:1:20: error: 'LOWRegister' was not declared in this scope
  digitalWrite(0, LOWRegister);
  ~~~~~
  Using library Servo at version 1.1.9 in folder: C:\Program Files\WindowsApps\ArduinoIDC.ArduinoIDC_1.0.57.0_x-0C_mdgms03d4wtb12librarie\servo
  last errors
  'LOWRegister' was not declared in this scope
  1
  Type here to search
  28°C Berawan
  10:40
  21/01/2022

```

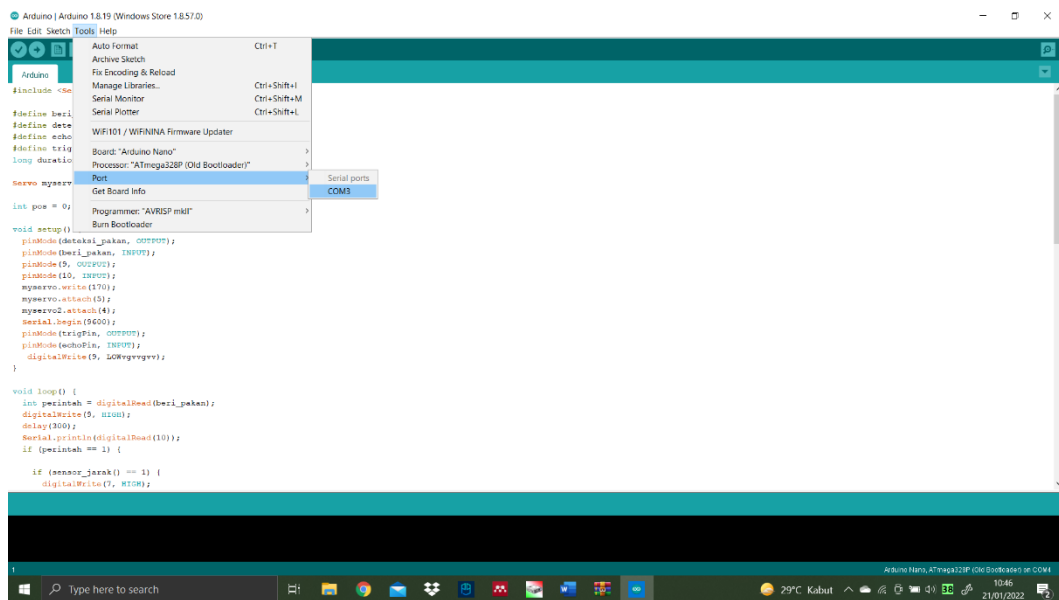
Gambar 5.8 Tampilan Code salah

Jika telah di temukan dan di perbaiki *source code* yang *error*, maka ketahap selanjutnya yaitu memilih board Arduino yang digunakan



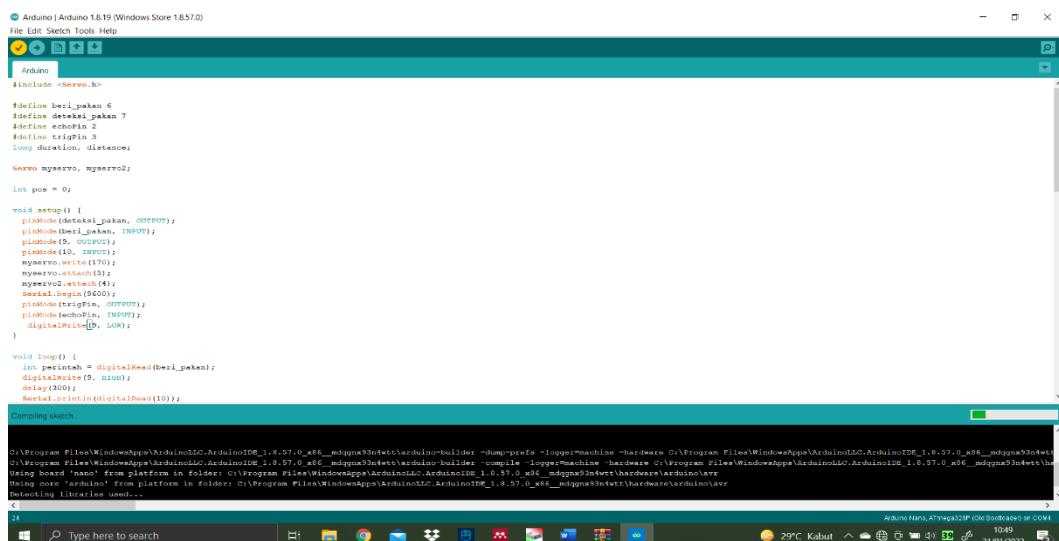
Gambar 5.9 Tampilan Bord Arduino

Jika telah selesai memilih *board* maka selanjutnya memilih *port* yang di gunakan untuk mengkoneksikan software Arduino IDE dengan mikrokontroler Arduino nano tahap ini penulis menggunakan *port* Com3 :



Gambar 5.10 Tampilan port

Jika telah terkoneksi antara Arduino IDE dengan Mikrokontroler Arduino akan upload ke mikrokontroler :



Gambar 5.11 upload sukses



Jika di bagian keterangan kiri bawah terdapat pesan *successfully done* itu tandanya program yang kita buat telah berhasil di masukan kedalam mikrokontroler dengan begitu program siap di uji coba.

### **5.2.2 Pengujian Perangkat Keras (Hardware)**

Pengujian ini dilakukan agar dapat mengetahui apakah alat atau perangkat keras berfungsi dengan baik, pengujian ini meliputi pengujian semua alat yang telah dirancang, pengujian dilakukan dengan cara menggerakkan alat yang telah dirancang dan melihat respon alat tersebut apakah berfungsi dengan sesuai yang diharapkan atau masih ada kesalahan yang lain.

Untuk pengujian rangkain di lakukan sebuah tes dengan menggunakan alat multy tesster, jika arus berada pada 5 Volt maka di pastikan rangkain sudah mempunyai arus yang cukup arus yang berlebihan ataupun berkurang akan berdampak pada mikrokontroller dan bisa juga menyebabkan kerusakan pada rangkaian tersebut.

#### **1. Pengujian Tegangan**

Rangkain ini merupakan otak dari seluruh rangkaian. Semua rangkaian yang ada di kendalikan input, outputnya oleh rangkaian mikrokontroller ini adapun hasil pengujian tegangan mikrokontroler dapat di lihat sebagai berikut

Tabel 5.1 Pengujian Mikrokontroler

Sumber	Tegangan input	Tegangan Ouput
Mikrokontroler	5 Volt	4,8Volt

## 2. Pengujian Tegangan sensor HC-SR04

Tahap pertama dilakukan tessting pada pengecekan arus yang berada pada sensor HC-SR04 tersebut tekanan yang baik pada rangkaian adalah 5Volt

Tabel 5.2 Pengujian Sensor HC-SR04

Sumber	Tegangan input	Tegangan Ouput
Hmc5883L	4,8Volt	4Volt

## 3. Pengujian motor servo

Tahap pertama dilakukan tessting pada pengecekan arus yang berada pada Motor servo tersebut tekanan yang baik pada rangkaian adalah 5Volt

Tabel 5.3 Pengujian Motor Servo

Sumber	Tegangan input	Tegangan Ouput
Motor Servo	4,8Volt	4.4Volt

### 5.3 ANALISA SISTEM SECARA KESELURUHAN

Untuk mendeteksi apabila terjadi kesalahan setelah uji coba, maka perlu dilakukan Analisa secara keseluruhan. Dari seluruh proses yang telah dilakukan, baik pengujian perangkat keras maupun perangkat lunak, proses dapat dikatakan bahwa alat ini dapat berfungsi sebagaimana yang penulis inginkan, sebagaimana proses pembacaan data pada rangkaian dan dapat berjalan sesuai dengan program yang telah dibuat

Pada keadaan pertama sensor HC-SR04 akan mendeteksi pakan ikan terlebih dahulu jika pakan ada alat akan berjalan sesuai dengan perintah yang telah diberikan, tetapi jika pakan habis maka akan memberikan pesan dalam telegram bahwa pakan habis dan alat tidak akan berjalan, tambakan pakan ikan menggunakan servo jika pakan ikan ada maka servo akan terbuka dan akan langsung memberikan pada tampungan kedua, tampungan kedua akan menuangkan pakan kedalam kolam jika sudah di berikan dari tampungan pertama, tampungan kedua juga memanfaatkan servo untuk system buka tutupnya, pengendalian telegram di program dengan NodeMCu dan pengendalian mekanik menggunakan Mikrokontroler Atmega.

Tabel 5.4 Uji coba keseluruhan

Uji coba	Sensor hc SR04	Servo 1	Servo 2	Hasil
1	Pakan Tersedia	1	1	Terisi
2	Pakan Tidak ada	0	0	Pakan Tidak Terisi
3	Pakan Tersedia	1	1	Terisi

Tabel 5.5 Uji coba memberikan pakan secara offline

Uji coba	Sensor hc SR04	Servo 1	Servo 2	Hasil
1	Pakan Tersedia	1	1	Terisi
2	Pakan Tidak ada	1	1	Alat akan tetap berjalan walaupun tidak ada isi
3	Pakan Tersedia	1	1	Terisi
4	Pakan tidak ada	1	1	Alat akan tetap berjalan walaupun tidak ada isi

Tabel 5.6 Uji coba memberikan pakan Menggunakan telegram

Uji coba	Sensor hc SR04	Servo 1	Servo 2	Hasil
1	Pakan Tersedia	1	1	Terisi
2	Pakan Tidak ada	0	0	Alat tidak akan hidup dan memberikan pesan pada Telegram
3	Pakan Tersedia	1	1	Terisi
4	Pakan Tidak ada	0	0	Alat tidak akan hidup dan memberikan pesan pada Telegram

Tabel 5.7 memberikan pakan saat menggunakan catu daya cadangan

Uji coba	Sensor	Tampungan	Pengisian	Hasil
1	Pakan Tersedia	1	1	Terisi
2	Pakan Tidak ada	0	0	Tidak terisi
3	Pakan Tersedia	1	1	Terisi
4	Pakan Tidak ada	0	0	Tidak terisi

Tabel 5.8 Uji coba sesuai ukuran pakan

Uji coba	Sensor	Tampungan	Pengisian	Hasil
1	Pakan full	1	1	Terisi
2	Pakan setengah	1	1	Terisi
3	Habis	0	0	Tidak

Tabel 5.9 Uji coba Delay alat

Uji coba	Alat	Delay
1	Arduino ke NodeMcu	2 Detik
2	Servo 1	1.5 Detik
3	Servo 2	3 Detik
4	NodeMcu ke Arduino	0.6 detik

Tabel 5.10 Uji coba Servo

Uji coba	Telegran	Servo Atas	Servo Bawah	Penampungan	Pemberian	Hasil
1	On	0	1	mati	Hidup	Bekerja
2	On	1	0	hidup	mati	Bekerja
3	Off	0	0	mati	mati	Tidak
4	Off	0	0	mati	mati	Tidak