

PROIECT PMP

Nume: Kiraly Nicole Elena

Grupa: 30237

1. **Descriere proiect**

Proiectul pe care am ales sa-l implementez este o masinuta care comunica prin Bluetooth cu un alt modul Bluetooth legat la o alta placuta pentru a uda florile care au nevoie de apa. Se foloseste de asemenea un senzor de umiditate pentru a masura nivelul de umiditate din sol.

1. **Lista componente**

-4BO Motors

-4 roti

-baterii de aprox 7,2V

-2 placute Arduino Uno

-1 senzor umiditate

-1 Motor Shield

-1 motor Servo

-1 pompa de apa

-2 module Bluetooth

1. **Functii proiect+descriere**

//MASTER

#include <SoftwareSerial.h>

SoftwareSerial BTSerial(2,3);

int sensor\_pin1 = A0;

int sensor\_data1;

String s ="";

void setup()

{

BTSerial.begin(9600);

Serial.begin(9600);

pinMode(sensor\_pin1, INPUT); //Soil moisture sensor1

}

void loop()

{

soilWatering1();

delay(1000);

Serial.println(s);

}

void soilWatering1(){

sensor\_data1 = analogRead(sensor\_pin1);

if(sensor\_data1 > 750 )

{

BTSerial.write("DRY1");

s="DRY1";

}

else if(sensor\_data1 <= 600)

{

BTSerial.write("MEDIUM1");

s = "MEDIUM1";

}

}

Acesta este fisierul pentru bluetooth-ul configurat ca master, acesta trimite date spre celalalt modul bluetooth care porneste masinuta cand solul este uscat, iar cand solul este mediu masinuta nu face nimic. Am folosit biblioteca SoftwareSerial care functioneaza aproximativ la fel ca Serial. Senzorul de umiditate este legat la A0 si citeste o valoare intre anumite valori. Cand solul este uscat aceasta valoare este peste 750, iar cand solul este ud are o valoare sub 600. Datele ajung la masinuta dupa cum urmeaza:

//SLAVE

#include <SoftwareSerial.h>

#include <AFMotor.h>

#include <Servo.h>

Servo myservo;

AF\_DCMotor right\_motor(1, MOTOR12\_8KHZ);

AF\_DCMotor left\_motor(2, MOTOR12\_8KHZ);

AF\_DCMotor pump\_motor(3, MOTOR12\_8KHZ);

SoftwareSerial BTSerial(2,3);

String readData="";

boolean stringComplete = false;

int pos = 0;

void setup() {

BTSerial.begin(9600);

right\_motor.setSpeed(255);

left\_motor.setSpeed(255);

pump\_motor.setSpeed(255);

myservo.attach(10);

myservo.write(90);

}

void loop() {

myservo.write(90);

readBluetooth();

if (stringComplete){

if (readData == "DRY1"){

moveToSensor1();

}

}

stringComplete = false;

readData="";

}

void readBluetooth(){

if(BTSerial.available()){

readData = BTSerial.readString();

}

stringComplete = true;

}

void moveToSensor1(){

//forward left

delay(500);

right\_motor.run (FORWARD);

left\_motor.run (FORWARD);

delay(2000);

right\_motor.run (FORWARD);

left\_motor.run (BACKWARD);

delay(500);

right\_motor.run (FORWARD);

left\_motor.run (FORWARD);

delay(2000);

right\_motor.run (RELEASE);

left\_motor.run (RELEASE);

waterOn();

delay(100);

initialState1();

}

void waterOn(){

pump\_motor.run (FORWARD); //start water

for (pos = 50; pos <= 130; pos += 1) {

myservo.write(pos);

delay(10);

}

for (pos = 130; pos >= 50; pos -= 1) {

myservo.write(pos);

delay(10);

}

delay(2000);

pump\_motor.run (RELEASE);

}

void initialState1(){

delay(10);

right\_motor.run (BACKWARD);

left\_motor.run (BACKWARD);

delay(2000);

right\_motor.run(BACKWARD);

left\_motor.run(FORWARD);

delay(500);

right\_motor.run (BACKWARD);

left\_motor.run (BACKWARD);

delay(2000);

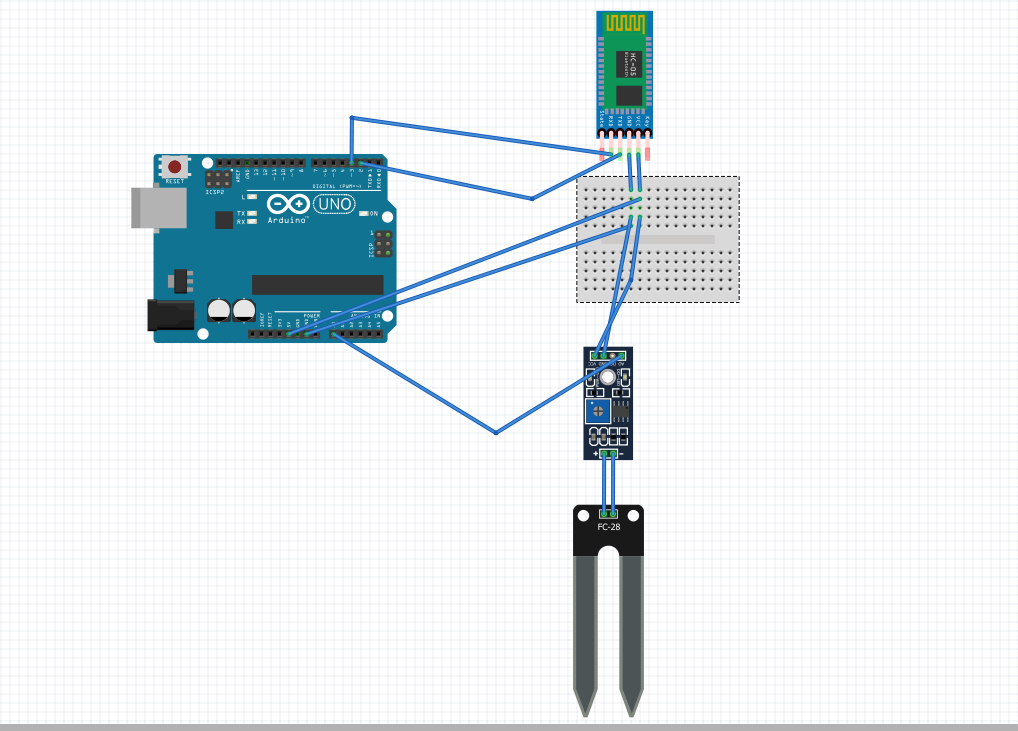
right\_motor.run (RELEASE);

left\_motor.run (RELEASE);

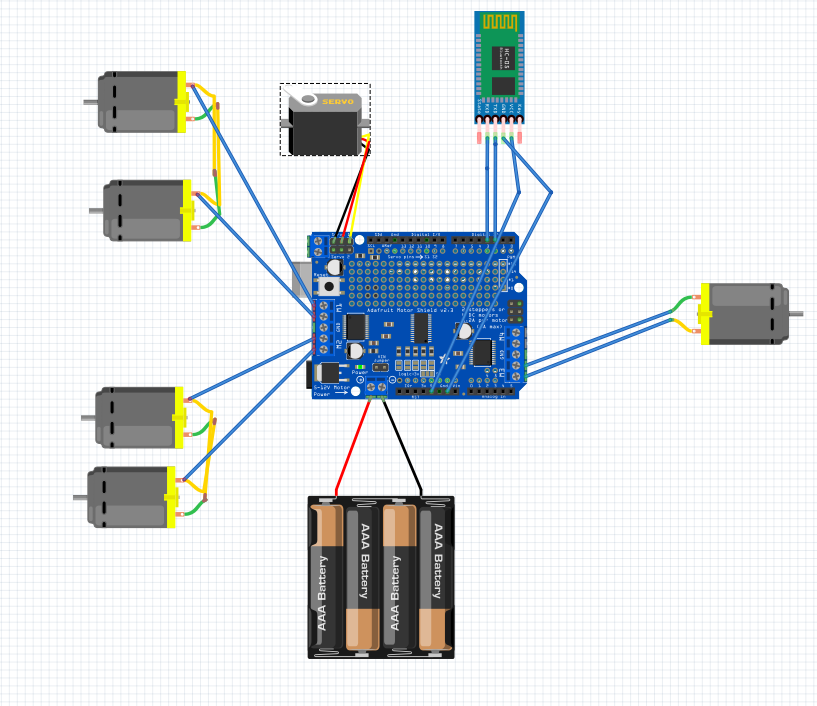
}

Masinuta primeste mesajul trimis de pe cealalta placuta si daca solul este uscat o sa-l ude, daca e mediu nu face nimic. Cand solul este uscat se apeleaza functia care muta masinuta spre directia unde e localizata planta, o uda si apoi se intoarce in starea ei initiala. Prodedeul se repeta cand planta e uscata iar. Am folosit 4 motoare pentru miscarea masinii, 2 fiind legate impreuna motor dreapta si inca 2 motor stanga. Mai exista si pompa de apa care trage apa din sticla cat timp solul e uscat si servo motorul care misca capatul pompei de apa stanga dreapta pentru imprastierea apei. Avand 5 motoare in total, toate sunt legate la motor shield-ul atasat placutei si voltajul si intensitatea necesare motoarelor au fost masurate pentru a alege bateriile corecte pentru a functiona in mod corespunzator.

1. **Schema electrica**

****

**senzor**

****

**Masinuta-bateriile sunt doar demonstrative, am folosit baterii de aprox 7V**

1. **Bibliografie**

* <https://docs.arduino.cc/learn/built-in-libraries/software-serial>
* <https://www.arduino.cc/reference/en/libraries/adafruit-motor-shield-library/>
* <https://mihai.utcluj.ro/wp-content/uploads/dmp/labs/pmp-lab08-Bluetooth-RO.pdf>
* <https://mihai.utcluj.ro/wp-content/uploads/dmp/labs/pmp-lab09.pdf>