DOCUMENTATIE

PROIECT

APLICATIE IN OPENGL CONTROLATA DE PLACA ARDUINO

PROIECTARE CU MICROPROCESOARE

UNIVERSITATEA TEHNICA DIN CLUJ-NAPOCA

FACULTATEA DE AUTOMATICA SI CALCULATOARE

SPECIALIZAREA CALCULATOARE SI TEHNOLOGIA INFORMATIEI

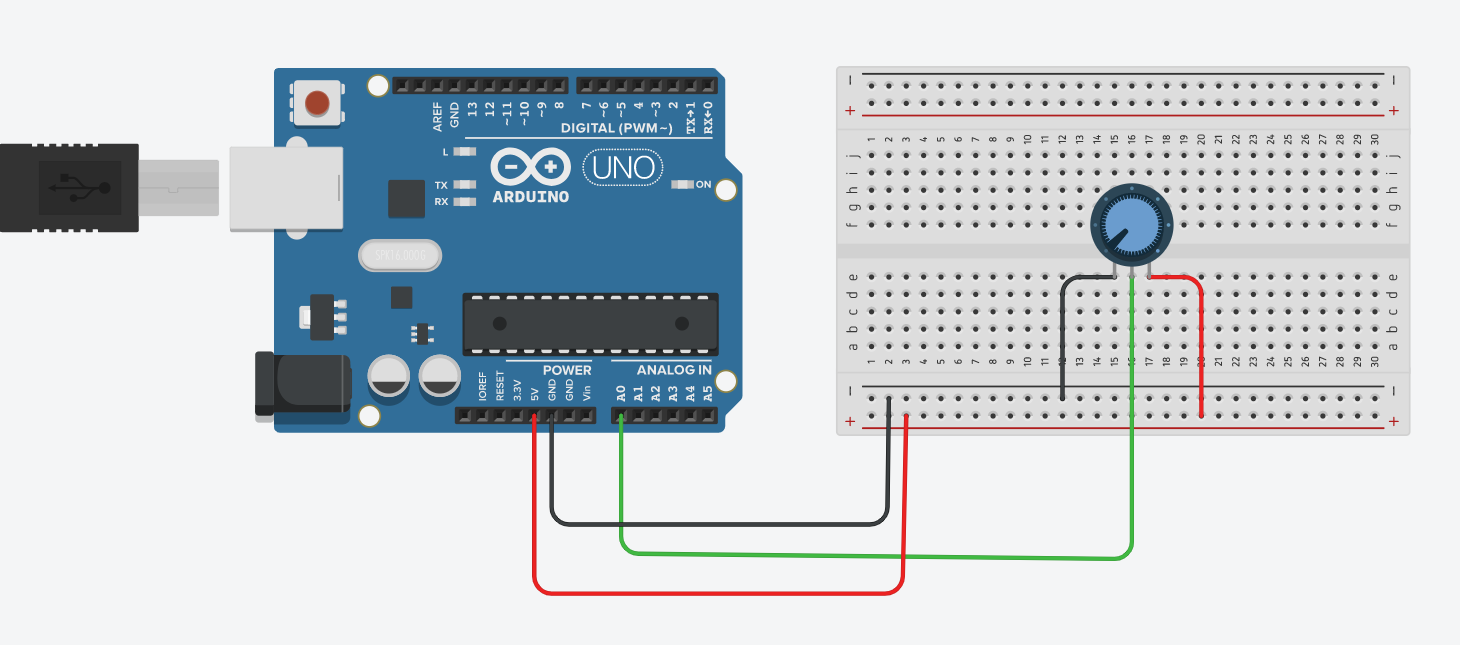
Negruț Ioana-Raluca

Grupa 30226

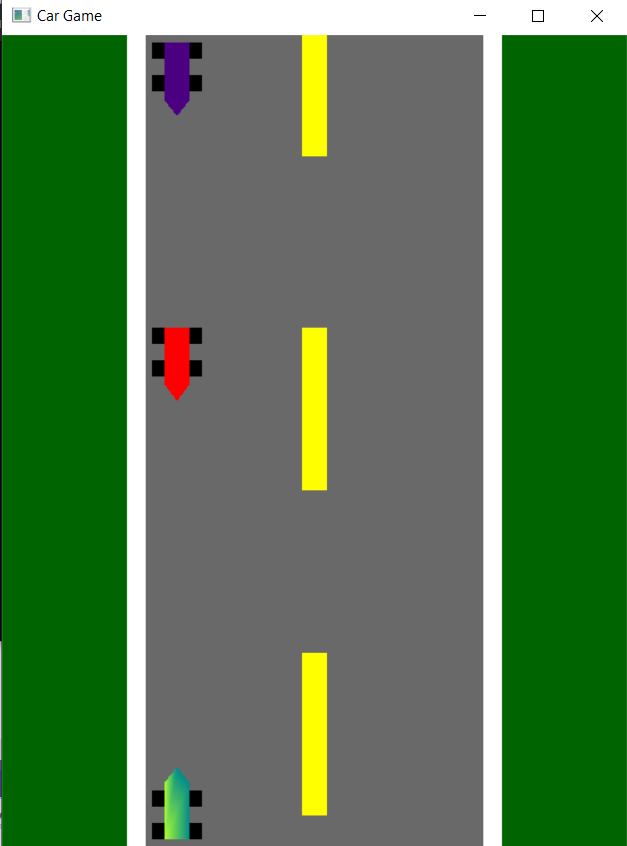
Componente:

* Placa de dezvolatare Arduino Uno R3
* Breadboard
* Fire tata-tata
* Potentiometru liniar(10k)
* Rezistor(1k)

Schema circuitului:

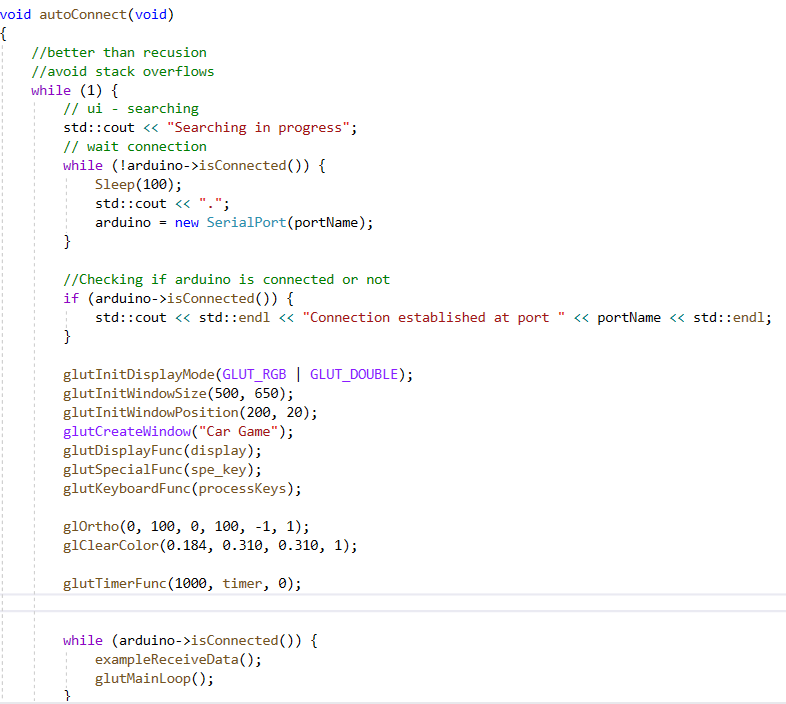


Aplicatia in openGL:



In opengl se foloseste libraria GLUT. GLUT are grijă de toate sarcinile specifice sistemului necesare pentru crearea ferestrelor, inițializarea contextelor OpenGL și gestionarea evenimentelor de intrare, pentru a permite programele OpenGL foarte portabile.





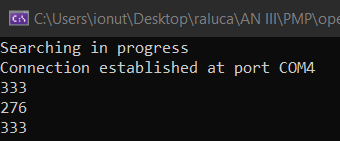
Prin functia autoConnect() se va verifica conexiunea cu portul COM4 si de asemenea si va porni aplicatia in opengl. Tot in aceasta functie se apeleaza functia de citire a datelor transmise de la potentiometru prin comunicarea seriala (exampleReceiveData()). Functia de citire a datelor foloseste arduino->readSerialPort() pentru citirea datelor, apoi acestea se vor converti in valori de int (sscanf\_s()) pentru a putea fi comparate cu valorea de referinta (< deplasare stanga, > deplasare dreapta). Se va apela functia spe\_key care controleaza coordonatele unde se va muta masina in fereastra opengl.

Descrierea proiectului:

Proiectul este format din doua parti: aplicatia in Visual Studio(folosind openGL) si partea hardware reprezentata de placa Arduino.

Aplicatie este un joc cu masini. Masina principala(verde) va trebui sa se fereasca de celelalte masini care vin din partea opusa a ferestrei jocului(masina rosie, mov etc.). Odata ce masina principala nu se fereste de celelalte masini, jocul este pierdut.

Placa Arduino este folosita ca si controller pentru jocul explicat mai sus. Cu ajutorul potentiometrului de pe placuta se va stabili directia in care masina principala va merge (stanga,dreapta).



Legatura dintre Visual Studio si Arduino IDE se va face prin folosirea monitorului serial, de unde aplicatie va lua datele citite de potentiometru. Se va face conexiunea cu arduino prin folosirea claselor Serial.cpp si Serial.h unde exista resurse pentru conectarea, citirea si procesarea datelor ( functiile: arduino->isConnected(), arduino->readSerialPort(), sscanf\_s) transmise prin portul COM4.

Codul incarcat pe placuta:

Graphical user interface, text, application, email

Description automatically generated

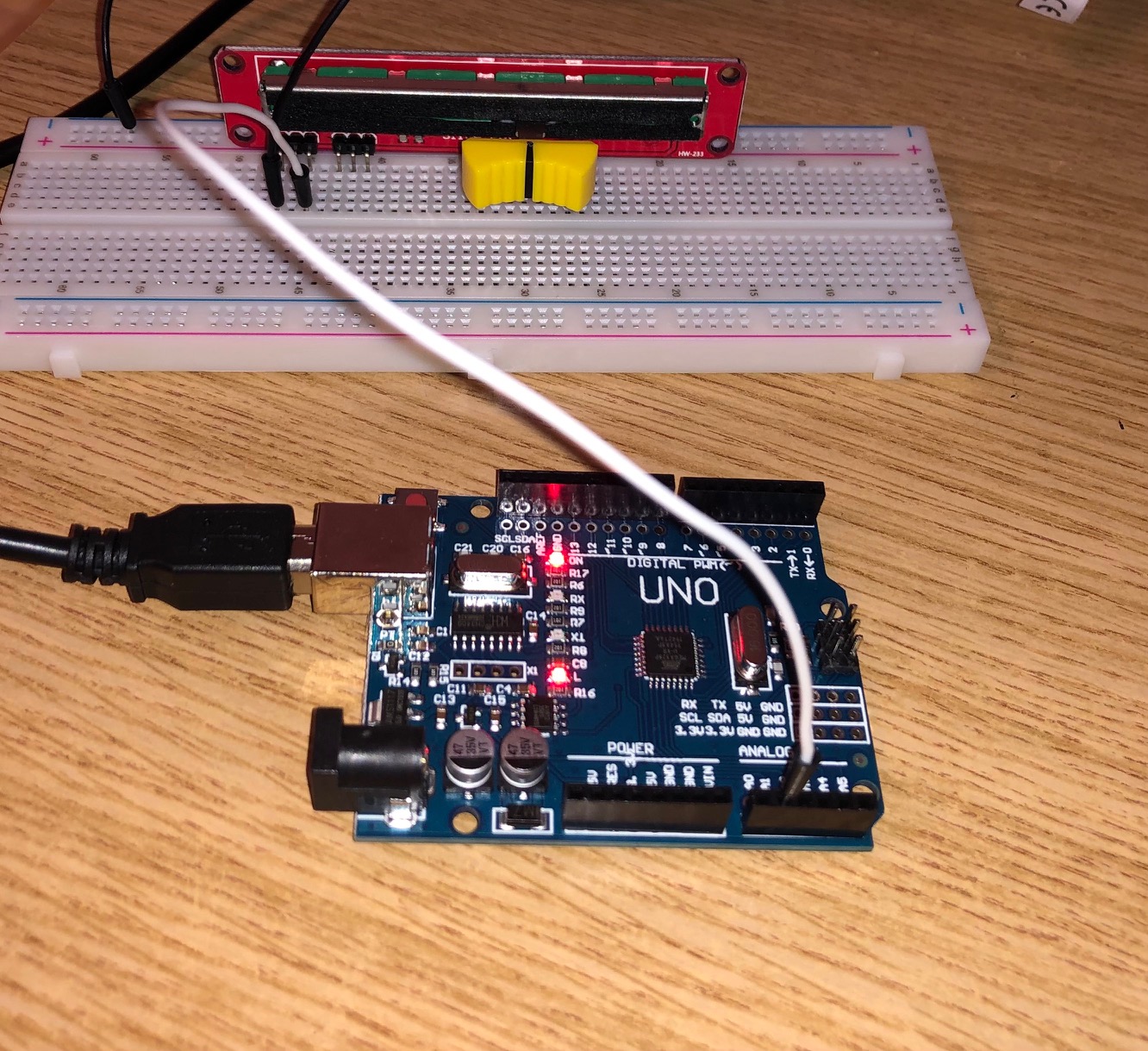
Configuram voltajul de referinta pentru intrarea analogica, ca DEFAULT.

Se seteaza pinul A1 ca si intrare.

Prin Serial.begin(9600) se va începe comunicarea seriala, astfel incat Arduino sa poata trimite comenzi prin conexiunea USB. Valoarea 9600 se numește „rata de transfer” a conexiunii. Acesta va stabili cât de rapid se vor transmite datele.

In variabila var2 se vor citi cu analogRead() valorile potentiometrului si deasemenea se vor afisa in monitorul serial pentru verificare (inainte sa folosim datele in visual studio). Se pune si o intarziere a transmiterii datelor pentru vizibilitate si procesare mai usoara.

Montajul complet:



**Bibliografie**

<https://medium.com/manash-en-blog/serial-communication-with-an-arduino-using-c-on-windows-d08710186498>

<https://www.tinkercad.com/things/eIhtvNElahh-amazing-lahdi-leelo/editel?tenant=circuits>

<https://www.instructables.com/Video-Game-Controller-With-Arduino/>

<https://www3.ntu.edu.sg/home/ehchua/programming/opengl/cg_introduction.html>