Ajustarea volumului prin recunoasterea degetelor cu Pi Pico 2 W

**Nume studenti:**

*Chilimon Ana-Maria*

*Chiriac Raluca-Ștefania*

*Stoian Mario-Daniel*

**Adrese de email:**

[ana-maria.chilimon@student.tuiasi.ro](mailto:ana-maria.chilimon@student.tuiasi.ro)

[raluca-stefania.chiriac@student.tuiasi.ro](mailto:raluca-stefania.chiriac@student.tuiasi.ro)

[mario-daniel.stoian@student.tuiasi.ro](mailto:mario-daniel.stoian@student.tuiasi.ro)

**Motivația alegerii temei:**

Motivația alegerii acestei teme constă în dorința de a demonstra un exemplu practic de interacțiune intuitivă între om și tehnologie, folosind recunoașterea gesturilor pentru controlul volumului. Într-o eră în care interacțiunea cu tehnologia devine tot mai naturală, am dorit să explorăm o metodă alternativă și inovatoare de ajustare a volumului melodiilor, folosind recunoașterea poziției degetelor prin detectarea imaginii captate de camera laptopului și procesată cu un program Python dezvoltat în PyCharm.

Prin utilizarea plăcuței Raspberry Pi Pico W și a componentelor suplimentare (display, LED-uri, amplificator, difuzoare, rezistente), proiectul nostru se încadrează în tendințele actuale din domeniul IoT(Internet of Things) și al interfețelor inteligente. Alegerea acestei teme este determinată de dorința de a dezvolta abilități tehnice și de a stimula gândirea inovatoare și creativă, atât în realizarea componentelor software, cât și în proiectarea hardware-ului aplicației.

**Componentele principale folosite in realizarea proiectului:**

* **Camera laptopului:** Captează imaginea necesară pentru recunoașterea poziției degetelor utilizatorului.
* **Raspberry Pi Pico 2 W:** Reprezintă unitatea de control principală care procesează datele și gestionează componentele sistemului.  
    
  O imagine care conține electronice, Componentă electronică, Componenta circuitului, Componentă de circuit pasiv

  Conținutul generat de inteligența artificială poate fi incorect.
* **Breadboard 830 + fire jumper:** Asigură conectarea facilă a componentelor electronice și organizarea circuitului.  
    
  A white electronic board with red and blue lines

  AI-generated content may be incorrect.
* **Display LED 4x7 segmente:** Afișează nivelul volumului în timp real, oferind feedback vizual clar utilizatorului.

O imagine care conține ceas, text, Ceas digital

Conținutul generat de inteligența artificială poate fi incorect.

* **Amplificator 3W I2S - MAX98357A:** Amplifică semnalul audio pentru redarea clară a sunetului în difuzoare.

O imagine care conține text, captură de ecran, Font, proiectare

Conținutul generat de inteligența artificială poate fi incorect.

* **2 difuzoare 50mm – 2W – 30 ohm:** Redau sunetul melodiei, fiind controlate din aplicație în funcție de gesturile detectate.

O imagine care conține difuzor, cerc

Conținutul generat de inteligența artificială poate fi incorect.

* **Set rezistori 1/4W:** Folosiți pentru limitarea curentului în diverse părți ale circuitului.

A close up of a device

AI-generated content may be incorrect.

* **LED-uri:** Indică starea volumului   
  
* **Buzzer (opțional):** Utilizat ca soluție de rezervă în cazul în care difuzoarele nu funcționează corespunzător.

O imagine care conține cablu

Conținutul generat de inteligența artificială poate fi incorect.

**Funcționarea sistemului este următoarea:**

1. **Captarea imaginii**: Camera laptopului captează în timp real imaginea mâinii utilizatorului, iar programul dezvoltat în Visual Studio procesează poziția și numărul degetelor vizibile.
2. **Determinarea nivelului volumului**: Pe baza numărului de degete detectate, sistemul ajustează automat volumul melodiei – de exemplu, mai multe degete ridicate indică un volum mai ridicat.
3. **Feedback vizual**: Nivelul actual al volumului este afișat pe un display LED cu 4 cifre din 7 segmente, oferind utilizatorului informații clare și imediate.
4. **Feedback prin LED-uri:** În funcție de nivelul volumului, se aprinde un LED în parte.
5. **Feedback auditiv**: Sunetul este redat prin difuzoare conectate la un amplificator audio. În cazul unei erori sau a lipsei sunetului, buzzer-ul poate fi activat ca alternativă pentru semnalizare auditivă.

**Importanța în domeniul Embedded Systems – SI**

* **Integrarea tehnologiilor moderne**: Proiectul demonstrează modul în care tehnologii actuale precum procesarea imaginii, microcontrolerele wireless și afișajele digitale pot fi integrate într-un sistem embedded interactiv, modern și intuitiv.
* **Aplicații practice**: Controlul volumului prin gesturi reflectă o tendință tot mai prezentă în domeniul interfețelor inteligente, cu potențial de aplicare în dispozitive multimedia, smart home sau asistenți personali.
* **Îmbunătățirea abilităților tehnice:** Realizarea sistemului necesită cunoștințe variate în programare, electronică, procesarea semnalului și integrarea hardware-software, dezvoltând competențe esențiale în domeniul embedded.
* **Feedback vizual și auditiv**: Sistemul integrează feedback vizual prin LED-uri și display, dar și auditiv prin difuzoare și buzzer, asigurând o interacțiune eficientă și accesibilă utilizatorului.
* **Extensibilitate și adaptabilitate**: Structura modulară a sistemului permite adaptarea sa pentru alte funcții, cum ar fi controlul media în automobile sau integrarea cu platforme IoT.
* **Educație și profesionalism:** Proiectul oferă o experiență practică valoroasă, dezvoltând competențe relevante pentru domeniul sistemelor embedded, utile atât în mediul academic, cât și pe piața muncii din industria tehnologică.

**Implementare:**

**1. Implementarea Hardware**

**Raspberry Pi Pico 2 W**

**Funcționalitatea principală:**

* Servește ca unitate de control centrală pentru toate componentele hardware
* Gestionează comunicarea wireless prin WiFi
* Controlează afișajul și LED-urile pentru feedback vizual
* Procesează și redă fluxul audio primit de la laptop

**Configurație Inițială:**  
• Conectarea plăcii Raspberry Pi Pico 2 W la laptop pentru programare.  
• Instalarea firmware-ului MicroPython pe placă pentru controlul componentelor hardware.

**Display 4x7 segmente (cu drivere)**  
• **Conexiuni:**  
• Conectarea pinilor de control ai display-ului la GPIO-uri ale Raspberry Pi Pico 2 W.  
• Alimentarea display-ului de la sursa de alimentare prin breadboard.  
• Afișarea nivelului de volum calculat în funcție de gesturi.

**LED-uri (verde, alb si galben)**  
• **Conexiuni:**  
• Conectarea fiecărui LED la câte un pin GPIO al plăcii, prin rezistori de protecție.

**Amplificator Audio MAX98357A**  
• **Conexiuni:**  
• Conectarea pinilor I2S (DIN, BCLK, LRCLK) la GPIO-uri dedicate de pe Raspberry Pi Pico 2 W.  
• Alimentarea amplificatorului din sursa de 3.3V și masă.

**Difuzoare (2x 50mm – 2W – 30 ohm)**  
• **Conexiuni:**  
• Legarea difuzoarelor la ieșirea audio a amplificatorului pentru redarea sunetului.

**Breadboard 830 + fire jumper + sursă alimentare**  
• Utilizate pentru interconectarea tuturor componentelor și distribuirea alimentării în mod organizat.  
  
**Protocolul de comunicare:**

* Utilizează două porturi UDP distincte: unul pentru datele audio (12345) și unul pentru comenzi de control (12346)
* Primește configurația audio (sample rate, bits per sample, channels) de la laptop
* Acceptă comenzi de control: PLAY, PAUSE, STOP și ajustări de volum
* Implementează timeout-uri pentru detecția pierderii conexiunii

**2. Implementarea Software**

**Firmware-ul Raspberry Pi Pico 2 W**  
 **Codul MicroPython:**  
• Configurarea pinilor GPIO pentru LED-uri, display, amplificator și comunicare cu laptopul.  
• Afișarea pe display a valorii volumului în timp real.  
• Aprinderea LED-urilor în funcție de nivelul volumului   
• Comandarea ieșirii audio pe baza semnalului transmis.

**Aplicația pe Laptop – Visual Studio + OpenCV:**  
 Utilizează MediaPipe pentru detectarea și urmărirea mâinii în timp real

 Procesează poziția degetelor pentru identificarea gesturilor specifice

 Calculează distanța între degete pentru controlul fin al volumului  
  
**Gesturile recunoscute:**

* **Palma deschisă** (toate degetele extinse): Pornește redarea muzicii
* **Pumnul strâns** (degetele strânse, policing ascuns): Pune pe pauză
* **Index și mijlociu ridicat**: Trece la piesa următoare
* **Index și deget mic ridicat**: Revine la piesa anterioară
* **Policing și index extinse**: Controlul volumului bazat pe distanța dintre degete

**Gestionarea audio:**

* Încarcă și gestionează o listă de fișiere WAV dintr-un director specificat
* Implementează streaming audio în timp real către Pico prin UDP
* Controlează volumul prin scalarea amplitudinii semnalului audio

## Folosește threading pentru a separa streaming-ul audio de procesarea video **Fluxul de Funcționare**

**Inițializarea Sistemului**

1. **Configurarea hardware**: Pico se conectează la rețeaua WiFi și inițializează toate componentele (display, LED-uri, amplificator)
2. **Încărcarea melodiilor**: Laptopul scanează directorul specificat și încarcă lista de fișiere audio
3. **Stabilirea conexiunii**: Se stabilește comunicarea UDP între laptop și Pico
4. **Configurarea audio**: Prima melodie este deschisă pentru a determina parametrii audio care sunt trimiși către Pico

**Bucla Principală de Funcționare**

1. **Captarea imaginii**: Camera laptopului captează continuu cadre video
2. **Procesarea gesturilor**: MediaPipe detectează poziția mâinii și determină gestul actual
3. **Interpretarea comenzilor**: Gesturile sunt traduse în comenzi specifice (play, pause, volum, etc.)
4. **Transmiterea comenzilor**: Comenzile sunt trimise către Pico prin UDP
5. **Actualizarea feedback-ului**: Pico actualizează display-ul și LED-urile în funcție de starea curentă
6. **Streaming audio**: Datele audio sunt transmise continuu în chunk-uri către Pico pentru redare

**Controlul Volumului**

Sistemul implementează un control fin al volumului prin:

* **Detectarea distanței**: Măsurarea distanței în pixeli între vârful policelui și indexului
* **Maparea valorilor**: Convertirea distanței (20-220 pixeli) în procent de volum (0-100%)
* **Scalarea audio**: Modificarea amplitudinii semnalului audio înainte de transmitere
* **Feedback vizual**: Afișarea valorii pe display și aprinderea LED-urilor corespunzătoare

**Caracteristici Tehnice Avansate**

**Sincronizarea Audio**

Sistemul implementează o sincronizare precisă prin:

* Calcularea duratei reale a fiecărui chunk audio
* Implementarea unei pauze corespunzătoare în thread-ul de streaming
* Evitarea supraîncărcării buffer-ului receptor
* Detectarea timeout-urilor pentru pierderea conexiunii

**Gestionarea Erorilor**

* **Reconectare automată**: Gestionarea pierderii conexiunii WiFi
* **Validarea datelor**: Verificarea integrității comenzilor și datelor audio
* **Fallback-uri**: Buzzer ca alternativă la difuzoare în caz de eroare
* **Cleanup automat**: Eliberarea resurselor la închiderea aplicației

**Optimizări de Performanță**

* **Multiplexarea display-ului**: Actualizare rapidă (240Hz) pentru eliminarea flicker-ului
* **Thread-uri dedicate**: Separarea procesării video de streaming-ul audio
* **Buffer-ing inteligent**: Gestionarea optimă a buffer-elor I2S
* **Cooldown-uri**: Prevenirea comenzilor duplicate prin implementarea întârzierilor

**Aplicații și Extensibilitate**

**Utilizări Practice**

* **Sisteme multimedia**: Control hands-free al audio player-elor
* **Smart home**: Integrare cu sisteme de automatizare casnică
* **Prezentări**: Control muzicii de fundal fără întreruperea prezentării
* **Persoane cu dizabilități**: Interfață alternativă pentru controlul audio

**Posibilități de Extindere**

* **Gesturi adiționale**: Implementarea de noi comenzi prin gesturi personalizate
* **Recunoașterea vocală**: Combinarea cu comenzi vocale pentru control hibrid
* **Integrare IoT**: Conectarea la platforme cloud pentru control remote
* **Interfețe multiple**: Suport pentru mai multe utilizatori simultan

**Importanța în Domeniul Embedded Systems**

Proiectul demonstrează integrarea modernă a tehnologiilor embedded prin:

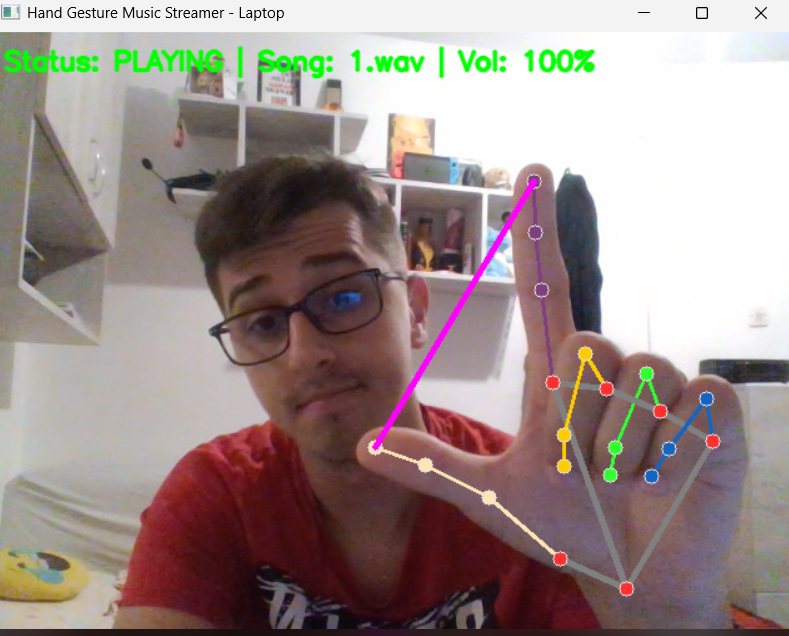
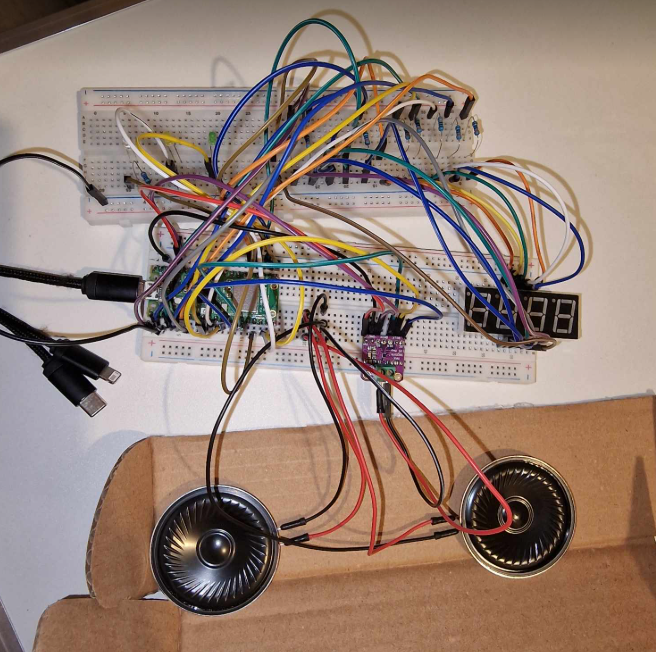
* **Procesarea în timp real**: Gestionarea simultană a multiple fluxuri de date
* **Comunicare wireless**: Implementarea protocoloalelor de rețea în sisteme embedded
* **Interfețe intuitive**: Crearea de experiențe utilizator naturale
* **Optimizarea resurselor**: Utilizarea eficientă a hardware-ului limitat
* **Scalabilitate**: Arhitectură modulară pentru extensii viitoare

**Rezumat:**

Acest proiect are ca scop dezvoltarea unui sistem de control al volumului audio prin recunoașterea gesturilor mâinii, folosind o placă Raspberry Pi Pico 2 W și o cameră pentru detectarea poziției degetelor. Volumul melodiei poate fi ajustat intuitiv prin mișcările degetelor, iar nivelul acestuia este afișat pe un display. În plus, nouă LED-uri care indică starea volumului: cand volumul este la 10% se aprinde un LED, cand se face 20% se aprinde alt LED s.a.m.d pana la 100%. Sistemul oferă o metodă modernă și interactivă de control, integrând atât partea software, cât și componenta hardware.  
  
***Prezentare:***A digital clock with wires

AI-generated content may be incorrect.

**A circuit board with wires and speakers

AI-generated content may be incorrect.**  
  


**Schema hardware implementata**

**Initial:**

A computer circuit board with wires and a digital display

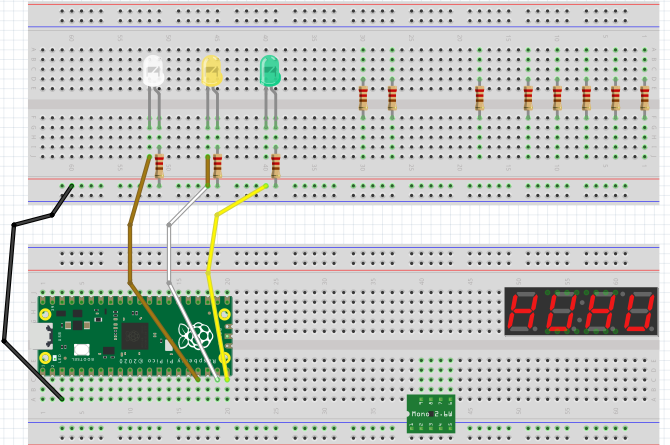
AI-generated content may be incorrect.

**Conectarea pentru display**

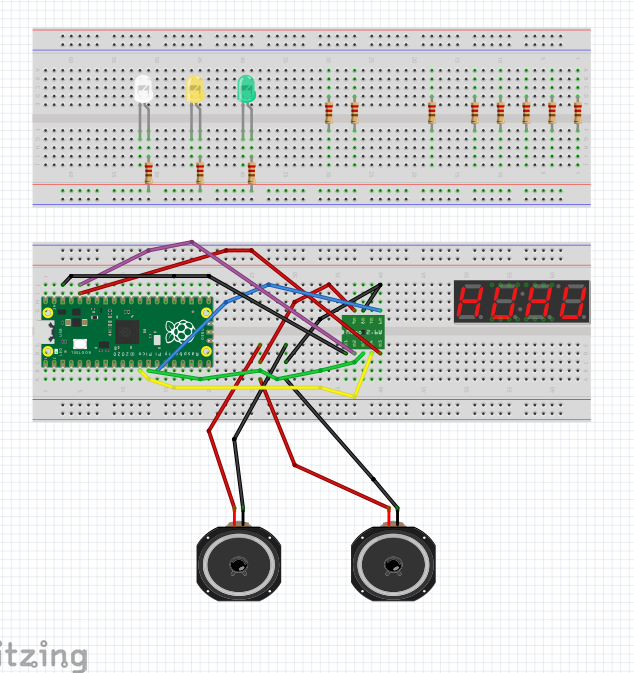
**A circuit board with wires and a couple of round speakers

AI-generated content may be incorrect.**

**Conectarea pentru LED-uri:**



**Conectarea cu difuzoare:**

****

Bibliografie:

[**https://github.com/ChiriacRaluca/Proiect-SM**](https://github.com/ChiriacRaluca/Proiect-SM)

[**https://datasheets.raspberrypi.com/picow/pico-2-w-pinout.pdf**](https://datasheets.raspberrypi.com/picow/pico-2-w-pinout.pdf)

**https://chat.openai.com/**

**https://www.electronicsforu.com/resources/7-segment-display-pinout-understanding**