

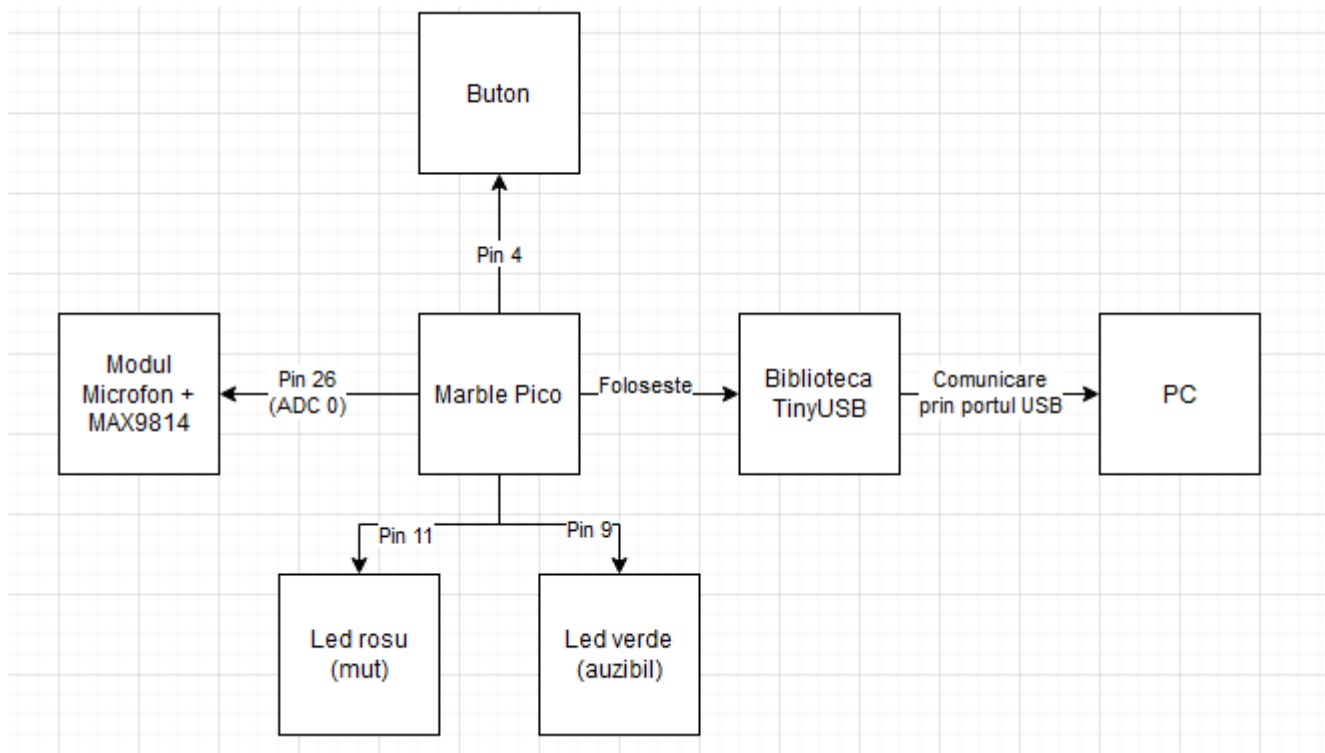
Good(ish) Microphone

Introducere

- Proiectul “Good(ish) Microphone” reprezintă dezvoltarea unui microfon USB cu calitate audio decentă, dotat cu un buton de mute dedicat. Microfonul capturează semnale audio folosind un convertor analog-digital cu o rezoluție de 16 biți ADC-ul intern de 12 biți și le transmite prin USB unui calculator.
- Acesta captează semnale audio, le convertește în format digital și le transmite prin USB către calculator, oferind și posibilitatea de întrerupere rapidă a captării prin butonul de mute.
- Am vrut să creez unui microfon accesibil și personalizabil care să ofere o calitate audio superioară microfoanelor de buget, dar la un cost mai redus decât soluțiile profesionale și pe care pot să îl montez pe casca și să îl pot ajusta cum doresc. Implementat este doar functionalitatea mecanică și nu poate fi montat pe casca, deoarece nu am putut să fac PCB pentru proiect atât de repede.
- Proiectul oferă o alternativă mai economică la soluțiile comerciale precum Antlion ModMic USB, având în plus avantajul personalizării și extinderii ulterioare.

Descriere generală

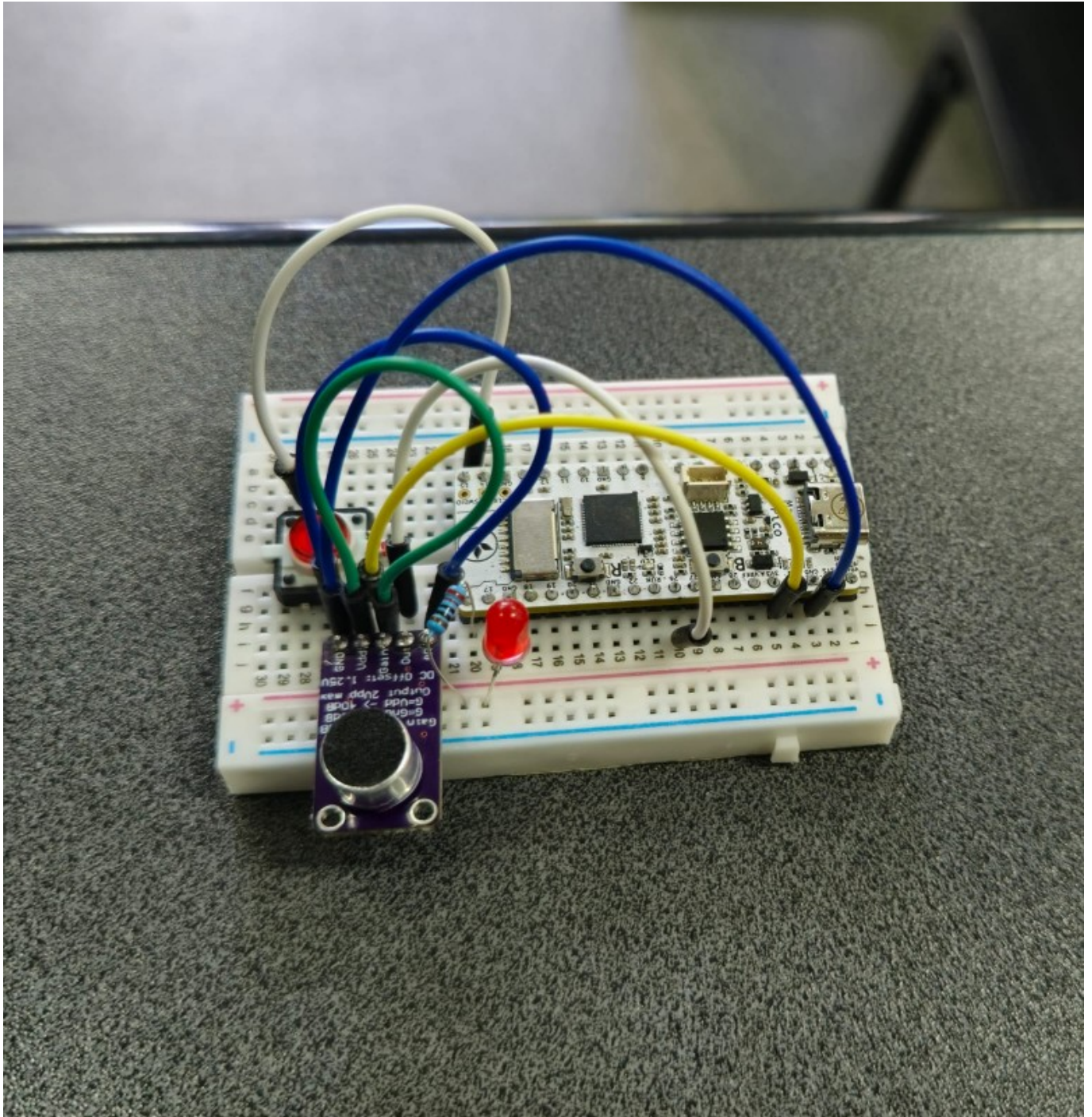
~~IC-ul de ADC~~ ADC-ul din pico poate să capteze cu rezoluție de 12 biți. Acesta preia sunetul de la un modul cu un amplificator MAX9814, care la rândul lui amplifică semnalul de la o capsulă electret și trimite datele la microcontroller. Pico-ul controlează led-ul de mut în funcție de starea butonului și transmite datele la calculator prin biblioteca TinyUSB.



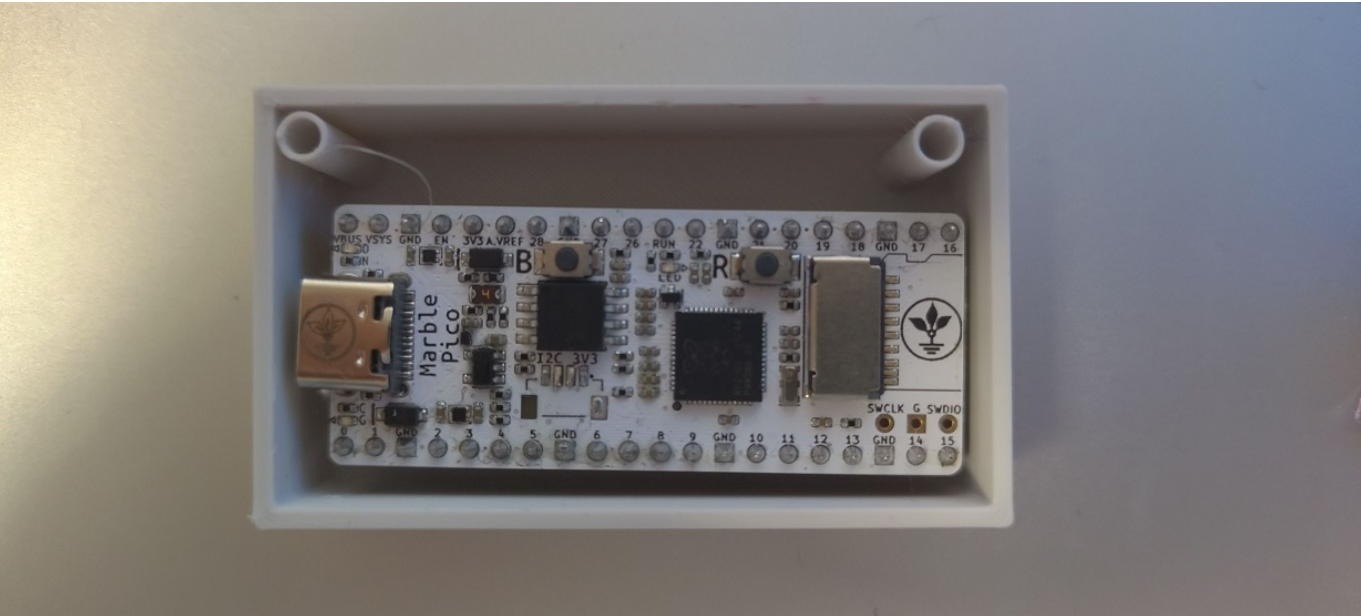
Hardware Design

- Marble Pico (ales pentru ca are Type C si un ADC mai precis decat pico normal)
- Modul Microfon cu Amplificator
- Buton
- LED rosu si LED verde
- Cabluri (USB C si de legat componentele intre ele)
- Carcasa (modelata prost :))

Acesta este primul prototip unde am testat pe breadboard functionalitatea software.



Aici se poate vedea cat de mica este carcasa pe care am facut-o :(. Am dezlipit (si am rupt si pad-ul) si conectorul I2C special cu care vine placuta ca sa fie mai mica.





Produsul final:

Aici am inclus si un “cablu” facut de mine care are o sarma in el. Acest lucru permite microfonului sa se deformeze plastic in functie de cum vrem sa fie asezat.

Singurele detalii care nu sunt mentionate in diagrama de sus sunt cum am setat gain-ul de la microfon: AR la GND si Gain la VDD. Datasheet-ul la MAX9814 era destul de straight forward pe aceasta tema.

Software Design

~~VSCode cu extensia oficiala pentru Raspberry PI Pico~~ Arduino Studio TinyUSB (fork-ul lui pschatzmann)

Pentru arduino studio a fost nevoie de setarile de pe pagina Ardushop a Marble Pico si importarea bibliotecii. Pentru compilare trebuie setat USB Stack ca TinyUSB.

Pentru ca sa pot ajunge la un sampling rate foarte aproape de maximul teoretic din datasheet (500 kHz/s), am folosit transferuri DMA pentru a citi de la ADC, deoarece acestea nu ocupa timpul procesorului si notifica atunci cand are date noi printr-o intrerupere. DMA-ul foloseste doua buffere in care pune date si schimba intre ele si permite functiei de callback la USB sa preia datele direct de acolo.

Rezultate Obținute

Microfonul se aude destul de bine, doar ca mai trebuie lucrat la intreruperile DMA si USB pentru a elimina stutter-urile. Overall pretty good pentru ~40 lei.

Concluzii

Foarte mult timp a fost pus pe:

- pe modalitatea initiala de sampling (cu FAN3852) deoarece nu puteam sa lipesc fire pe IC;
- incercarea folosirii directe a TinyUSB fara Arduino ca sa fie detectat dispozitivul ca microfon (ex bug: se bloca calculatorul cand incercam sa selectez microfonul)
- calibrarea imprimantei 3d pentru a printa carcasa si alte mulaje;
- incercarea a mai multor modalitati de creare a cablului care se deformeaza deoarece nu am reusit sa gasesc astfel de cablu gata facut;
- depanarea codului (DMA + USB) etc.

Am creat acest proiect cu scopul de a face pe viitor un microfon miniaturizat in stilul la Antlion ModMic, pe care sa il folosesc eu. Motivul pentru care am ales sa il fac un microfon care sta pe masa este faptul ca era prea mare un pico care atarna de o casca :). O sa iau componentele direct de pe placuta aceasta si le pun pe alt pcb (si alte extinderi).

Download

- carcasa + codul pentru arduino studio [proiect_andrei_l.zip](#). Trebuie setat TinyUSB ca USB Stack in setarile Arduino Studio

Jurnal

- 29.04.2025: Draft OCW
- 30.05.2025: Final Document

Bibliografie/Resurse

- <https://www.farnell.com/datasheets/2736016.pdf>
- <https://www.analog.com/media/en/technical-documentation/data-sheets/MAX9814.pdf>
- <https://github.com/raspberrypi/pico-examples> (in special ADC si DMA)
- <https://ardushop.ro/ro/groundstudio/1590-groundstudio-marble-pico-6427854000682.html>
- <https://datasheets.raspberrypi.com/rp2040/rp2040-datasheet.pdf>
- <https://www.pschatzmann.ch/home/2024/10/13/tinyusb-audio-on-an-rp2040-in-arduino/>

[Export to PDF](#)

From:

<http://ocw.cs.pub.ro/courses/> - **CS Open CourseWare**

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2025/rnedelcu/andrei.lipan>



Last update: **2025/05/30 07:40**