

# Tobe Electronice

## Introducere

Acest proiect constă în realizarea unui controller de tobe electronice bazat pe placa ATmega328P XMINI și senzori piezoelectrice, capabil să detecteze loviturile aplicate pe diferite pad-uri și să le convertească în semnale MIDI transmise către calculator. Sistemul include funcționalități precum detectarea intensității loviturii, metronom vizual și audio, afișarea BPM-ului și gestionarea mai multor preseturi de configurare.

- **Ce face:** Sistemul citește semnalele analogice provenite de la senzorii piezoelectrice, detectează loviturile și transmite evenimente MIDI către calculator prin comunicație serială USB. Pe lângă partea de trigger MIDI, proiectul include afișaj pentru BPM și preset, metronom cu LED și buzzer, precum și control prin butoane.
- **Scopul lui:** Scopul proiectului este demonstrarea integrării mai multor periferice hardware și protocoale de comunicație într-un sistem embedded interactiv cu aplicație practică în domeniul instrumentelor muzicale electronice.
- **Ideea de pornire:** Ideea proiectului a pornit de la dorința de a construi un kit de tobe electronice modular folosind componente accesibile și cunoștințe acumulate în cadrul laboratoarelor.
- **Utilitate:** Proiectul oferă o modalitate practică de studiere a senzorilor analogici, procesării semnalelor și interacțiunii dintre hardware și software. În plus, sistemul poate fi extins și adaptat pentru diferite configurații de instrumente MIDI electronice.

## Descriere generală

Arhitectura proiectului se bazează pe o unitate centrală de procesare reprezentată de microcontrollerul ATmega328P XMINI, care gestionează citirea senzorilor piezoelectrice, generarea semnalelor MIDI și controlul perifericelor auxiliare. Sistemul este organizat în mai multe module hardware care comunică prin interfețe analogice și digitale.

## Module și interacțiuni

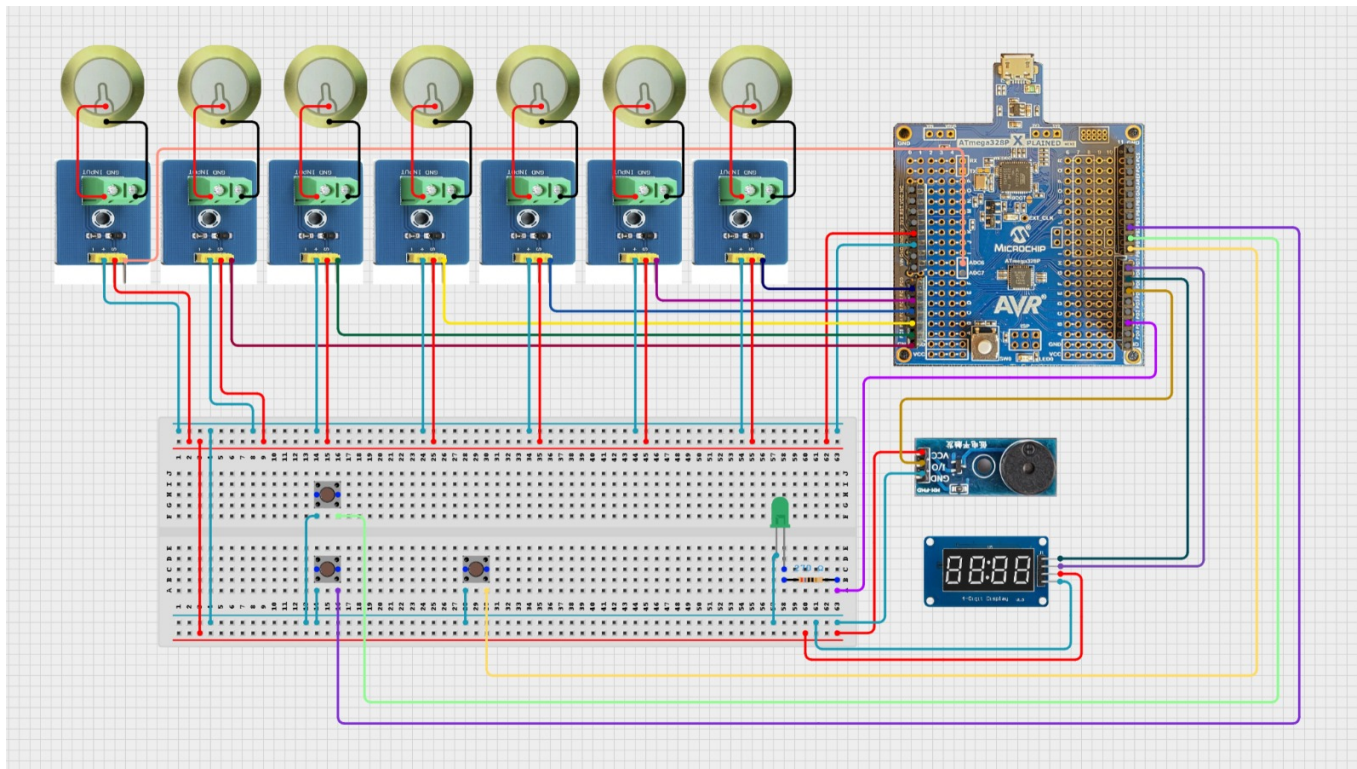
1. **Unitatea de procesare (ATmega328P):** Microcontrollerul reprezintă nucleul sistemului și coordonează toate componentele proiectului. Acesta citește valorile analogice provenite de la senzorii piezoelectrice, detectează loviturile și convertește evenimentele în mesaje MIDI transmise către calculator prin interfața serială USB (USART).
2. **Sistemul de intrare analogică (ADC):** Kitul de tobe utilizează mai mulți senzori piezoelectrice analogici montați pe pad-uri. Fiecare senzor generează o tensiune proporțională cu intensitatea

- loviturii, iar semnalul este citit de convertorul Analog-to-Digital (ADC) al microcontrollerului.
3. **Interfața de control (GPIO):** Sistemul include trei butoane tactile utilizate pentru controlul BPM-ului metronomului și schimbarea preseturilor. Butoanele utilizează rezistențele interne de tip pull-up ale microcontrollerului și comută semnalul în LOW la apăsare.
  4. **Sistemul de afișare (7-Segment Display):** Afișajul cu 4 cifre este utilizat pentru afișarea BPM-ului. Modulul comunică printr-o interfață serială simplificată utilizând doi pini digitali ai microcontrollerului.
  5. **Sistemul de feedback audio și vizual:** Metronomul este implementat utilizând un LED de mare intensitate și un buzzer activ. Acestea oferă feedback vizual și audio pentru menținerea ritmului în timpul utilizării sistemului.
  6. **Interfața de comunicație serială (USART):** Datele MIDI generate de microcontroller sunt transmise către calculator prin conexiunea microUSB integrată pe placă. Un software de tip MIDI bridge interpretează datele seriale și le convertește într-un port MIDI virtual utilizat de aplicațiile audio.
  7. **Sistemul de alimentare:** Întregul sistem este alimentat prin conexiunea microUSB a plăcii XMINI. Alimentarea de 5V este distribuită către senzori și modulele auxiliare prin liniile comune ale breadboard-ului.



## Hardware Design

Designul hardware al proiectului este conceput pentru a realiza un controller de tobe electronice modular și ușor de extins, utilizând senzori piezoelectrice pentru detectarea loviturilor și comunicație serială pentru transmiterea semnalelor MIDI către calculator. Sistemul este organizat în jurul microcontrollerului ATmega328P XMINI și utilizează periferice dedicate pentru feedback audio-vizual și controlul utilizatorului.



## Lista de piese:

- 1 x Placă de dezvoltare ATmega328P XMINI
- 7 x Module senzori piezoelectrice analogici
- 1 x Display TM1637 4-digit 7-segment
- 1 x Modul Active Buzzer pentru feedback audio
- 1 x Breadboard MB102 830 puncte
- 3 x Butoane momentary push button
- 1 x LED verde ultra bright 5mm
- 1 x Rezistență 270Ω (pentru LED)
- Jumper Wires
- Cablu siliconic multifilar 24AWG roșu/negru

## Detalii de design electric:

Senzorii piezoelectrice sunt conectați la intrările analogice ale microcontrollerului prin intermediul modulelor dedicate de condiționare a semnalului. Fiecare senzor generează o tensiune proporțională cu intensitatea loviturii detectate. Microcontrollerul citește aceste valori utilizând convertorul ADC intern și generează mesaje MIDI corespunzătoare.

Display-ul TM1637 este utilizat pentru afișarea BPM-ului și comunică prin doi pini digitali dedicați. Feedback-ul utilizatorului este completat de un LED de mare intensitate și un buzzer activ utilizat pentru implementarea metronomului.

Întregul sistem este alimentat prin conexiunea microUSB a plăcii XMINI, iar tensiunea de 5V este distribuită către module prin liniile comune ale breadboard-ului.

## Documentația pinilor și a conexiunilor hardware

Sistemul utilizează microcontrollerul ATmega328P XMINI ca unitate centrală de procesare. Organizarea pinilor a fost realizată pe criterii funcționale, separând componentele analogice de perifericele digitale pentru a obține o arhitectură clară și ușor de extins.

### Organizarea porturilor

Port	Rol principal în proiect
PORTC (PC0-PC5 + ADC6)	Interfață analogică pentru senzorii piezoelectrice
PORTB (PB0-PB2)	Intrări digitale pentru butoane
PORTD (PD2, PD5-PD7)	Periferice digitale de ieșire

### Senzori piezoelectrice

Sistemul utilizează 7 senzori piezoelectrice analogici pentru detectarea loviturilor. Fiecare senzor este conectat la un modul analogic, care realizează preprocesarea și protecția semnalului înainte de conversia ADC.

Componentă	Pin MCU	Rol
Piezo 1	PC0	Kick
Piezo 2	PC1	Hi-Hat Pedal / Kick 2
Piezo 3	PC2	Hi-Hat
Piezo 4	PC3	Snare
Piezo 5	PC4	Tom
Piezo 6	PC5	Crash
Piezo 7	ADC6	Ride / Crash 2

**Motivația alegerii:** Pinii PORTC sunt conectați direct la convertorul Analog-to-Digital al ATmega328P, fiind alegerea naturală pentru citirea semnalelor analogice generate de discurile piezoelectrice.

### Butoane de control metronom

Sistemul include 3 butoane tactile pentru controlul BPM-ului și schimbarea preset-urilor. Butoanele utilizează rezistențele interne de tip pull-up ale microcontrollerului și comută semnalul în LOW la apăsare.

Funcție	Pin MCU	Configurație
BPM Up	PB0	GPIO input + pull-up
BPM On/Off	PB1	GPIO input + pull-up
BPM Down	PB2	GPIO input + pull-up

**Motivația alegerii:** PORTB a fost ales pentru intrările digitale deoarece oferă pini GPIO grupați compact și ușor de citit simultan prin registrul PINB.

## Butoane de selecție preset

Sistemul utilizează butonul și LED-ul intern al plăcii pentru selectarea și indicarea presetului activ al kit-ului de tobe.

Componentă	Pin MCU	Rol
Buton intern	PB7	Schimbare preset
LED intern	PB5	Feedback vizual preset

La apăsarea butonului intern conectat pe PB7, sistemul va realiza ciclarea între cele 4 preseturi disponibile. Fiecare preset reprezintă o configurație diferită a kit-ului, permițând modificarea parametrilor utilizați pentru semnalele MIDI generate.

LED-ul intern conectat pe PB5 oferă feedback vizual asupra presetului selectat printr-un număr de impulsuri luminoase egal cu indexul presetului activ.

## Display TM1637

Afișajul cu 4 cifre este utilizat pentru afișarea BPM-ului.

Semnal TM1637	Pin MCU
CLK	PD6
DIO	PD7

**Motivația alegerii:** PD6 și PD7 sunt pini digitali liberi și permit organizarea clară a conexiunilor fără a interfera cu intrările analogice.

## Buzzer activ

Buzzerul activ oferă feedback audio pentru funcția de metronom.

Componentă	Pin MCU	Tip
Buzzer activ	PD5	GPIO / PWM

**Motivația alegerii:** PD5 suportă funcții PWM, utile pentru generarea tonurilor și a semnalelor audio.

## LED feedback

LED-ul oferă feedback vizual pentru metronom.

Componentă	Pin MCU	Configurație
LED verde	PD2	GPIO output

LED-ul este conectat în serie cu o rezistență de limitare de  $270\Omega$ .

## Alegerea rezistenței

Având:

- tensiune alimentare: 5V
- tensiune directă LED: 3V - 3.4V
- rezistență:  $270\Omega$

Curentul rezultat este:

$$I = (5V - 3.2V) / 270\Omega \approx 6.7mA$$

Această valoare este suficientă pentru obținerea unei luminozități ridicate fără suprasolicitarea LED-ului sau a pinului microcontrollerului.

## Alimentare

Întregul sistem este alimentat prin conexiunea microUSB a plăcii XMINI.

Linie alimentare	Tensiune
VCC	+5V
GND	0V

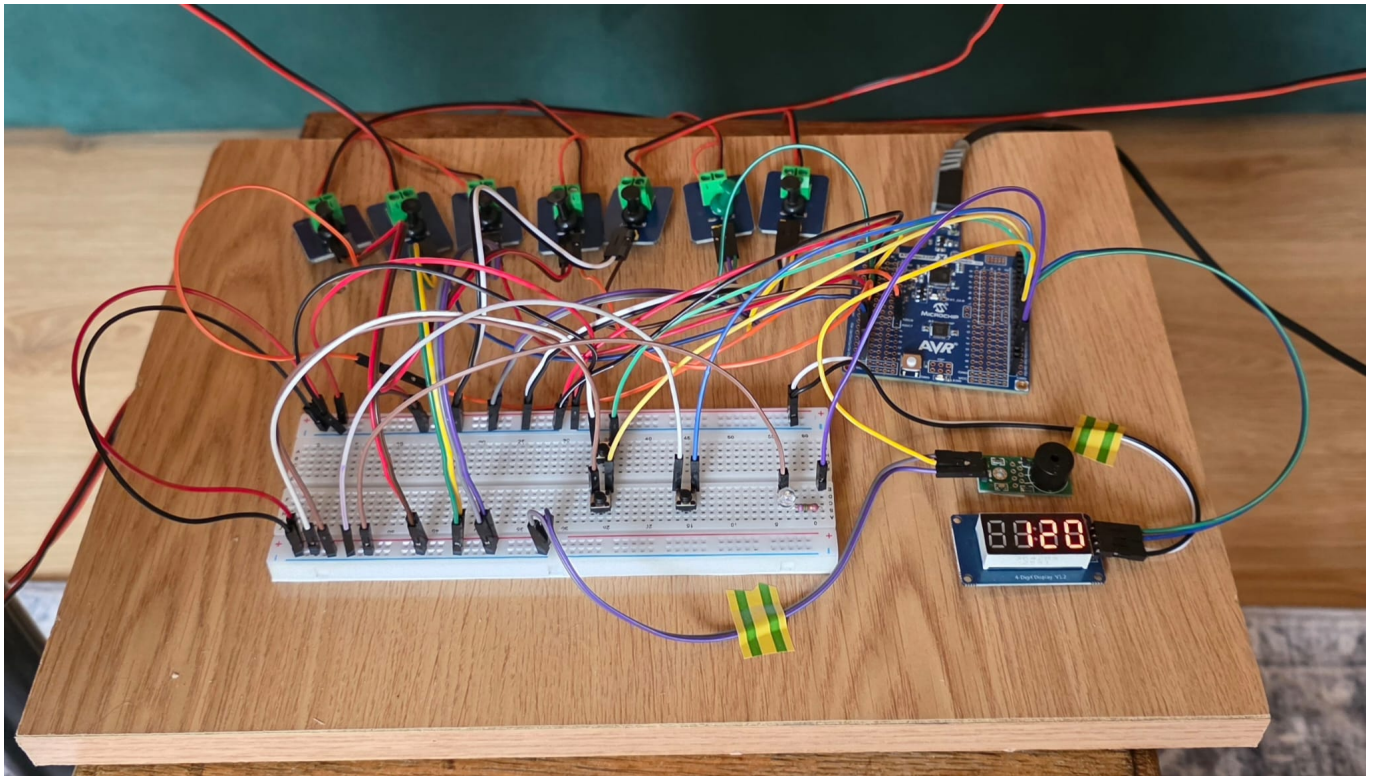
Liniile de alimentare sunt distribuite către toate modulele prin rail-urile breadboard-ului.

## Interfață serială USB

Comunicarea MIDI cu calculatorul se realizează prin interfața USART-over-USB integrată pe placă.

Datele seriale generate de microcontroller sunt interpretate de un software MIDI bridge și convertite într-un port MIDI virtual utilizat de aplicațiile audio.

## Circuitul Fizic



## Software Design

### Descrierea codului aplicației

Firmware-ul proiectului a fost dezvoltat utilizând limbajul C/C++ pentru platforme embedded și are rolul de a gestiona citirea senzorilor, procesarea semnalelor și generarea mesajelor MIDI.

#### Mediu de dezvoltare:

- Visual Studio Code
- PlatformIO IDE
- Framework Arduino pentru ATmega328P

PlatformIO a fost utilizat pentru gestionarea compilării, configurării proiectului și administrarea bibliotecilor externe.

#### Librării și surse third-party utilizate:

- TM1637 Library (smougenot/TM1637) - utilizată pentru controlul afișajului cu 4 cifre și 7 segmente;
- Arduino Framework - utilizat pentru funcționalități de bază precum comunicația serială (`Serial`) și structura programului (`setup()`, `loop()`).

Funcționalitățile principale precum configurarea GPIO, ADC, timerele și întreruperile au fost implementate direct prin accesarea registrelor microcontrollerului, fără utilizarea unor biblioteci suplimentare.

## Citirea Butoanelor

Sistemul utilizează patru butoane pentru interacțiunea cu utilizatorul:

- BPM UP – creșterea vitezei metronomului;
- BPM DOWN – scăderea vitezei metronomului;
- Metronome Toggle – activarea/dezactivarea metronomului;
- Change Preset – schimbarea configurației kit-ului.

Butoanele sunt conectate pe pini configurați ca intrări și utilizează rezistențele interne de tip pull-up ale microcontrollerului. În această configurație, pinul are valoarea logică HIGH în stare normală și trece în LOW la apăsarea butonului.

Starea butoanelor este citită utilizând registrul PINB, iar pentru identificarea unei apăsări noi se utilizează compararea stării curente cu starea anterioară.

Pentru eliminarea efectului de debounce, care poate produce mai multe tranziții rapide la o singură apăsare, a fost implementată o perioadă minimă între două citiri consecutive.

De asemenea, pentru butoanele BPM UP și BPM DOWN a fost implementată funcționalitatea de apăsare lungă (hold detection), permițând modificarea rapidă a valorii BPM atunci când utilizatorul menține butonul apăsat.

Preseturile pentru kit modifica următoarele sunete:

Preset	Pedala 2	Cinelul 2
Preset 1	Hi-Hat Control	Ride
Preset 2	Hi-Hat Control	Crash 2
Preset 3	Kick 2	Ride
Preset 4	Kick 2	Crash 2

## Metronomul

Metronomul a fost implementat folosind Timer1 al microcontrollerului ATmega328P și întreruperea de tip Compare Match. Alegerea unei întreruperi hardware permite generarea impulsurilor metronomului la intervale precise, fără utilizarea funcției de `delay()` și fără blocarea buclei principale a programului.

Timer1 este configurat în modul CTC (Clear Timer on Compare Match). În acest mod, timerul numără până la valoarea definită în registrul de comparație, moment în care este generată întreruperea și contorul este resetat automat.

Perioada de apariție a unui beat este calculată în funcție de BPM folosind formula:

$$\text{ticks\_per\_beat} = (\text{F\_CPU} / \text{Prescaler}) \times (60 / \text{BPM})$$

unde:

- `F_CPU` reprezintă frecvența microcontrollerului;
- `Prescaler` reprezintă factorul de divizare al timerului;

- BPM reprezintă viteza metronomului.

Funcția `updateMetronomeTimer()` este responsabilă pentru actualizarea valorii registrului de comparație de fiecare dată când BPM-ul este modificat. Astfel, perioada de funcționare a metronomului se adaptează automat la viteza selectată de utilizator.

Întreruperea asociată `Timer1` nu gestionează întreaga durată a impulsului generat de metronom, ci marchează doar începutul unui nou tick. În momentul apariției întreruperii sunt activate LED-ul și buzzerul, iar o variabilă globală semnalizează apariția unui nou beat.

Stingerea LED-ului și oprirea buzzerului nu sunt realizate în rutina de întrerupere, ci în funcția `loop()`. Această alegere reduce timpul petrecut în cadrul ISR și menține rutina de întrerupere cât mai scurtă și eficientă.

În bucla principală este determinat momentul la care impulsul trebuie încheiat, utilizând timpul curent și durata definită a pulsului. După expirarea intervalului stabilit, LED-ul și buzzerul sunt dezactivate.

Prin această abordare, metronomul funcționează independent de citirea senzorilor și de procesarea butoanelor, iar bucla principală rămâne disponibilă pentru scanarea continuă a pad-urilor piezoelectrice.

## Citirea Senzorilor și Generarea Sunetului

Sistemul utilizează senzori piezoelectrice pentru detectarea loviturilor aplicate pe diferite componente ale kit-ului de tobe. Fiecare senzor produce un semnal analogic a cărui amplitudine este proporțională cu intensitatea impactului. Aceste semnale sunt conectate la intrările analogice ale microcontrollerului și sunt prelucrate folosind convertorul Analog-to-Digital (ADC).

ADC-ul este configurat pentru a utiliza tensiunea de referință  $V_{cc}$  și convertește semnalele analogice în valori digitale pe 10 biți, rezultând valori în intervalul 0-1023. Pentru fiecare pad, microcontrollerul selectează canalul ADC corespunzător, pornește conversia și citește valoarea rezultată.

Procesarea semnalelor provenite de la senzori este realizată prin funcția `processPad()`. Această funcție reprezintă componenta principală a sistemului de detectare și generare a evenimentelor MIDI.

Funcția primește următorii parametri:

- `channel` - reprezintă canalul ADC asociat unui senzor piezoelectric;
- `note` - reprezintă nota MIDI transmisă către calculator și determină sunetul redat;
- `threshold` - reprezintă pragul minim de detecție a loviturii. Valorile sub acest prag sunt considerate zgomot și sunt ignorate;
- `cap` - reprezintă valoarea maximă permisă pentru amplitudinea semnalului. Această limitare previne variațiile foarte mari produse de lovituri directe asupra senzorului;
- `cooldown` - reprezintă intervalul minim dintre două detectări consecutive și este utilizat pentru eliminarea declanșărilor multiple produse de vibrațiile reziduale.

În momentul în care valoarea ADC depășește pragul de detecție și perioada de cooldown a expirat, sistemul consideră că a fost detectată o lovitură validă.

După validarea loviturii, este calculată intensitatea acesteia (`velocity`). Parametrul `velocity`

reprezintă intensitatea sunetului transmis către aplicația audio și poate avea valori între 0 și 127 conform standardului MIDI.

Valoarea este calculată utilizând relația:

$$\text{velocity} = ((\text{adcValue} - \text{threshold}) \times 127) / (\text{cap} - \text{threshold})$$

Prin această formulă, valorile citite de ADC sunt transformate într-un interval compatibil MIDI, astfel încât loviturile mai puternice produc un sunet mai intens, iar loviturile mai slabe generează un volum redus.

După calcularea intensității, sistemul transmite prin USART un mesaj MIDI către calculator, unde acesta este interpretat de aplicația software și convertit în sunetul corespunzător componenteii lovite.

## Sistemul de Procesare MIDI de pe Calculator

Pentru funcționarea completă a sistemului de tobe electronice a fost necesară utilizarea unor aplicații software suplimentare pe calculator. Aceste aplicații realizează conversia mesajelor seriale în evenimente MIDI și redarea sunetelor corespunzătoare instrumentelor muzicale.

Fluxul complet al datelor este următorul:



### EA Serial MIDI Bridge

[EA Serial MIDI Bridge](#) reprezintă componenta care realizează conversia datelor seriale generate de microcontroller în mesaje MIDI compatibile cu sistemul de operare. Microcontrollerul transmite mesaje MIDI prin interfața USART utilizând conexiunea USB a plăcii. Deoarece sistemul de operare nu interpretează direct aceste date ca dispozitiv MIDI, este necesară utilizarea unui software intermediar. Aplicația citește fluxul serial de date și transformă secvențele transmise în mesaje MIDI standard, care pot fi utilizate de aplicațiile audio.

În cadrul proiectului a fost utilizată implementarea open-source disponibilă pe GitHub.

## loopMIDI

[loopMIDI](#) este utilizat pentru crearea unui port MIDI virtual în sistemul de operare. Mesajele MIDI generate de EA Serial MIDI Bridge nu sunt transmise direct către aplicația audio, ci sunt direcționate către un port virtual. Acest port funcționează ca o conexiune între aplicațiile software și permite transferul datelor MIDI între acestea.

În cadrul proiectului a fost creat un port virtual dedicat utilizat pentru transferul evenimentelor MIDI generate de sistemul de tobe.

## REAPER

[REAPER](#) este utilizat ca Digital Audio Workstation (DAW) și reprezintă componenta centrală de procesare audio a sistemului. Acesta primește evenimentele MIDI de la portul virtual creat prin loopMIDI și le direcționează către plugin-ul de instrument virtual.

În plus, REAPER oferă:

- configurarea intrărilor MIDI;
- gestionarea dispozitivelor audio;
- încărcarea plugin-urilor VST;
- controlul latenței și al bufferelor audio;
- înregistrarea și procesarea semnalelor audio.

## MT Power Drum Kit

[MT Power Drum Kit](#) este un plugin virtual de tip VST utilizat pentru generarea sunetelor de tobe. Plugin-ul primește notele MIDI transmise de REAPER și le convertește în sunete corespunzătoare fiecărei componente a kit-ului de tobe. În funcție de nota MIDI și de parametrul *velocity*, plugin-ul poate reda diferite intensități și variații ale sunetului, simulând comportamentul unui set de tobe reale.

Acesta a fost ales deoarece oferă un kit complet de instrumente, latență redusă și compatibilitate directă cu aplicațiile DAW.

## Rezultate Obținute

În urma realizării proiectului am obținut un sistem funcțional de tobe electronice bazat pe microcontrollerul ATmega328P și senzori piezoelectrice, capabil să detecteze loviturile și să genereze evenimente MIDI în timp real.

Funcționalitățile implementate și validate includ:

- detectarea loviturilor aplicate pe cele 7 pad-uri piezoelectrice;
- determinarea intensității loviturii și generarea parametrului `velocity`;
- transmiterea mesajelor MIDI către calculator prin comunicație serială;
- implementarea unui metronom cu feedback audio și vizual;
- controlul BPM-ului prin intermediul butoanelor;
- afișarea valorii BPM pe display-ul TM1637;
- implementarea mai multor preseturi pentru configurarea kit-ului.

În timpul testării, sistemul a demonstrat un răspuns rapid și o funcționare stabilă în condiții normale de utilizare. Integrarea cu aplicațiile software utilizate pe calculator a permis redarea corectă a sunetelor și simularea funcționării unui set de tobe electronice.

În urma testelor am observat faptul că în cazul unor fill-uri rapide și complexe pad-ul de snare poate omite ocazional anumite lovituri consecutive. Acest comportament apare ca rezultat al mecanismelor de filtrare și cooldown implementate pentru eliminarea triggerelor false. Am preferat obținerea unei funcționări stabile și a unui comportament predictibil pe durata utilizării generale, chiar dacă acest lucru implică un compromis în cazul secvențelor foarte rapide de lovituri.

## Video Demo Procesare Audio

## Video Demo Metronom

## Concluzii

Proiectul a demonstrat posibilitatea realizării unui sistem funcțional de tobe electronice utilizând un microcontroller ATmega328P și senzori piezoelectrice. Prin integrarea ADC-ului, comunicației USART, întreruperilor și perifericelor digitale, a fost obținut un sistem capabil să detecteze lovituri și să genereze evenimente MIDI în timp real.

Rezultatele obținute confirmă faptul că funcționalitățile studiate în cadrul laboratoarelor pot fi combinate într-o aplicație practică și extensibilă, cu utilizare în domeniul instrumentelor muzicale electronice.

## Download

[GitHub Repo](#)

[Cod Sursă și Schema Electrică](#)

## Jurnal

Data	Activitate
1 Mai	Comandarea componentelor necesare proiectului
8 Mai	Ridicarea componentelor și realizarea documentației introductive
14 Mai	Realizarea schemei electrice
15 Mai	Documentarea pinilor și începerea asamblării circuitului
19 Mai	Finalizarea circuitului și testarea componentelor hardware
20 Mai	Implementarea logicii pentru citirea și procesarea butoanelor
21 Mai	Implementarea funcționalității de metronom
22 Mai	Implementarea citirii senzorilor piezoelectrice și ajustarea parametrilor

## Bibliografie/Resurse

[ATMega328P XMINI Documentation](#)

[7 Segments Display Library](#)

[EA Serial MIDI Bridge](#)

[loopMIDI](#)

[REAPER](#)

[MT Power Drum Kit](#)

From:

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

Permanent link:

[http://ocw.cs.pub.ro/courses/pm/prj2026/tarik\\_ilhan.omer/gabriel.paduraru](http://ocw.cs.pub.ro/courses/pm/prj2026/tarik_ilhan.omer/gabriel.paduraru)



Last update: **2026/05/25 06:18**