

# Aero Beat

## Introducere

Prezentarea pe scurt a proiectului vostru:

- ce face
- care este scopul lui
- care a fost ideea de la care ați pornit
- de ce credeți că este util pentru alții și pentru voi

**Aero Beat** este un instrument muzical virtual care simulează o baterie reală folosind două bețe de tobă echipate cu senzori de mișcare. Fiecare bată conține un modul giroscop + accelerometru (GY-521 / MPU-6050) montat pe vârf, care detectează direcția și intensitatea loviturii în aer. În funcție de gestul efectuat, sistemul identifică una dintre cele 3 tobe virtuale asociate fiecărei bețe (stânga, mijloc, dreapta) și redă în timp real sunetul corespunzător printr-un amplificator de chitară. Simultan, un LED RGB se aprinde și o aplicație mobilă evidențiază vizual toba lovită sau chiar un visualiser.

**Ce face:** Cântărețul ține câte o bată în fiecare mână și efectuează lovituri în aer. Sistemul detectează lovitura, identifică toba vizată pe baza direcției de mișcare, redă sunetul corespunzător, aprinde LED-ul RGB și trimite informația către o aplicație mobilă care afișează vizual toba activată. Există și un buton care schimbă genul tobelor (metal, rock etc.) și unu care pune pauza.

**Scopul proiectului:** Crearea unui instrument de percuție virtual care să ofere experiența cântatului la tobe fără a necesita o baterie fizică, fără zgomot mecanic, fără spațiu ocupat și fără a deranja pe nimeni din jur.

**Ideea de la care am pornit:** Mereu mi-am dorit să cânt la tobe, însă locuiesc la apartament și am vecini cu copil mic. O baterie reală este scumpă, voluminoasă și, cel mai important, extrem de zgomotoasă. Mi-am propus să construiesc un sistem care să îmi ofere aceeași experiență fără să deranjez pe nimeni.

### De ce este util:

- Pentru muzicieni începători sau avansați care vor să exerseze fără investiție majoră în echipament și fără probleme de zgomot
- Pentru persoanele care locuiesc la apartament sau în spații unde zgomotul este o problemă reală (copii mici, vecini, program de noapte)

## Descriere generală

O schemă bloc cu toate modulele proiectului vostru, atât software cât și hardware însoțită de o descriere a acestora precum și a modului în care interacționează. Exemplu de schemă bloc:

<http://www.robs-projects.com/mp3proj/newplayer.html>



## Hardware Design

Aici puneți tot ce ține de hardware design:

- listă de piese
- scheme electrice (se pot lua și de pe Internet și din datasheet-uri, e.g. <http://www.captain.at/electronic-atmega16-mmc-schematic.png>)
- diagrame de semnal
- rezultatele simulării

### Listă de piese:

- ATmega328P Xplained Mini — microcontroler principal
- 2 x GY-521 (MPU-6050) — senzori giroscop + accelerometru 3 axe, montați pe vârful betelor
- 2 x DFPlayer Mini + card SD — modul redare audio, fișiere .mp3 cu sunete de tobă
- Modul Bluetooth HC-05
- Display LCD1602 HD44780
- LED RGB (anod comun sau catod comun)
- Amplificator de chitară — output audio
- Modul jack 3.5mm + adaptor jack 3.5mm → 6.35mm — conectare la amplificator
- 2 x bețe de tobă (suport fizic pentru senzori)
- 2 x Modul de buton fara retinere
- Rezistoare, fire, breadboard, 2 carduri de memorie microSD

### Schema electrica:



Componenta	Pin componenta	Conectare
ATmega328P Xplained Mini	PC4 / SDA	Bus I2C Date (MPU-6050 #1, #2 și LCD)
ATmega328P Xplained Mini	PC5 / SCL	Bus I2C Ceas (MPU-6050 #1, #2 și LCD)
ATmega328P Xplained Mini	PD2 (UART TX)	RXD HC-05 Bluetooth (prin divizor 1kΩ + 2kΩ)
ATmega328P Xplained Mini	PD3 (UART RX)	TXD HC-05 Bluetooth (direct)
ATmega328P Xplained Mini	PD4 (SoftSerial TX)	RX DFPlayer Mini #1 (prin rezistor 1kΩ)
ATmega328P Xplained Mini	PD5 (SoftSerial RX)	TX DFPlayer Mini #1 (direct)

ATmega328P Xplained Mini	PD6 (SoftSerial TX)	RX DFPlayer Mini #2 (prin rezistor 1kΩ)
ATmega328P Xplained Mini	PD7 (SoftSerial RX)	TX DFPlayer Mini #2 (direct)
ATmega328P Xplained Mini	PB0 (pin 8)	OUT Buton 1 — pauză/resume
ATmega328P Xplained Mini	PB1 (pin 9)	OUT Buton 2 — schimbare stil ROCK/METAL
ATmega328P Xplained Mini	PB2 (pin 10)	LED RGB — canal Roșu (PWM)
ATmega328P Xplained Mini	PB3 (pin 11)	LED RGB — canal Verde (PWM)
ATmega328P Xplained Mini	PB4 (pin 12)	LED RGB — canal Albastru
ATmega328P Xplained Mini	PB5 (pin 13)	LED on-board
ATmega328P Xplained Mini	5V (J202-5)	Alimentare module
ATmega328P Xplained Mini	3.3V (J202-4)	Pin AD0 MPU-6050 #2
ATmega328P Xplained Mini	GND (J202-6)	Masă comună

Componenta	Pin componenta	Conectare
MPU-6050 #1 (Bâta stângă)	VDD	5V
MPU-6050 #1 (Bâta stângă)	GND	Masă comună (GND)
MPU-6050 #1 (Bâta stângă)	SDA	PC4 ATmega328P
MPU-6050 #1 (Bâta stângă)	SCL	PC5 ATmega328P
MPU-6050 #1 (Bâta stângă)	AD0	GND → adresă I2C 0x68

Componenta	Pin componenta	Conectare
MPU-6050 #2 (Bâta dreaptă)	VDD	5V
MPU-6050 #2 (Bâta dreaptă)	GND	Masă comună (GND)
MPU-6050 #2 (Bâta dreaptă)	SDA	PC4 ATmega328P
MPU-6050 #2 (Bâta dreaptă)	SCL	PC5 ATmega328P
MPU-6050 #2 (Bâta dreaptă)	AD0	3.3V ATmega328P → adresă I2C 0x69

Componenta	Pin componenta	Conectare
DFPlayer Mini #1 (Bâta stângă)	VCC	5V
DFPlayer Mini #1 (Bâta stângă)	GND	Masă comună (GND)
DFPlayer Mini #1 (Bâta stângă)	RX	PD4 ATmega328P (prin rezistor 1kΩ)
DFPlayer Mini #1 (Bâta stângă)	TX	PD5 ATmega328P (direct)
DFPlayer Mini #1 (Bâta stângă)	DAC_R	1kΩ → nod mixare → 100Ω → Jack R
DFPlayer Mini #1 (Bâta stângă)	DAC_L	1kΩ → nod mixare → 100Ω → Jack L

Componenta	Pin componenta	Conectare
DFPlayer Mini #2 (Bâta dreaptă)	VCC	5V
DFPlayer Mini #2 (Bâta dreaptă)	GND	Masă comună (GND)
DFPlayer Mini #2 (Bâta dreaptă)	RX	PD6 ATmega328P (prin rezistor 1kΩ)
DFPlayer Mini #2 (Bâta dreaptă)	TX	PD7 ATmega328P (direct)
DFPlayer Mini #2 (Bâta dreaptă)	DAC_R	1kΩ → același nod cu DF1 DAC_R → 100Ω → Jack R
DFPlayer Mini #2 (Bâta dreaptă)	DAC_L	1kΩ → același nod cu DF1 DAC_L → 100Ω → Jack L

Componenta	Pin componenta	Conectare
Modul Jack TRRS 3.5mm	R (Ring)	Nod mixare DAC_R (DF1+DF2 prin 1kΩ) → Rezistor 100Ω
Modul Jack TRRS 3.5mm	L (Tip)	Nod mixare DAC_L (DF1+DF2 prin 1kΩ) → Rezistor 100Ω
Modul Jack TRRS 3.5mm	GND (Sleeve)	GND DFPlayer (izolare ground loop)

Componenta	Pin componenta	Conectare
HC-05 Bluetooth	VCC	5V
HC-05 Bluetooth	GND	Masă comună (GND)
HC-05 Bluetooth	RXD	PD2 ATmega328P (prin divizor tensiune 1kΩ + 2kΩ)

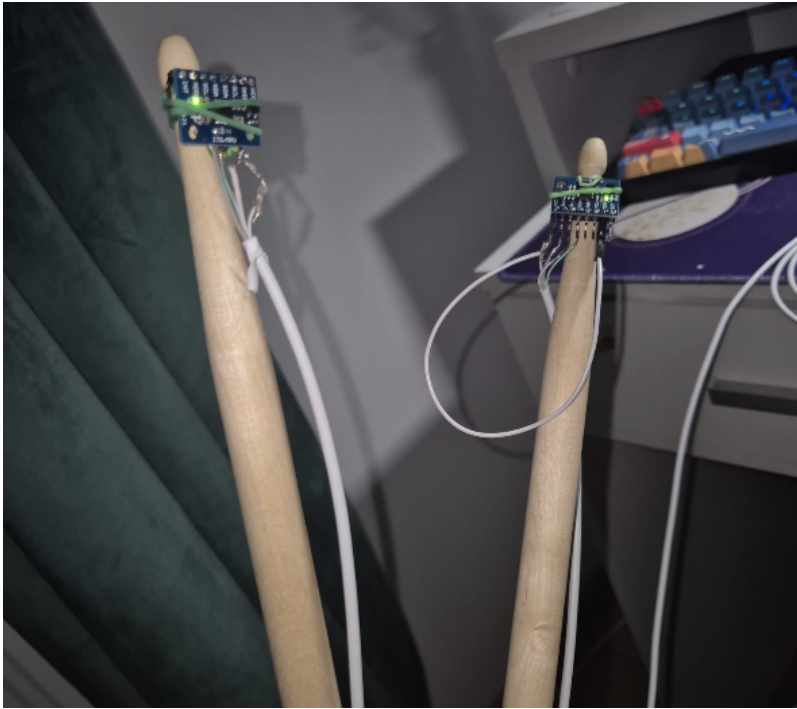
HC-05 Bluetooth	TXD	PD3 ATmega328P (direct)
-----------------	-----	-------------------------

Componenta	Pin componenta	Conectare
LCD I2C 16x2	VCC	5V
LCD I2C 16x2	GND	Masă comună (GND)
LCD I2C 16x2	SDA	PC4 ATmega328P (partajat pe magistrala I2C)
LCD I2C 16x2	SCL	PC5 ATmega328P (partajat pe magistrala I2C)

Componenta	Pin componenta	Conectare
Buton 1 (pauză/resume)	VCC	5V
Buton 1 (pauză/resume)	GND	Masă comună (GND)
Buton 1 (pauză/resume)	OUT	PB0 ATmega328P (pin 8)
Buton 2 (stil ROCK/METAL)	VCC	5V
Buton 2 (stil ROCK/METAL)	GND	Masă comună (GND)
Buton 2 (stil ROCK/METAL)	OUT	PB1 ATmega328P (pin 9)

Componenta	Pin componenta	Conectare
LED RGB (Anod Comun)	VCC (anod comun)	5V
LED RGB (Anod Comun)	R (catod roșu)	PB2 ATmega328P — pin 10 (PWM)
LED RGB (Anod Comun)	G (catod verde)	PB3 ATmega328P — pin 11 (PWM)
LED RGB (Anod Comun)	B (catod albastru)	PB4 ATmega328P — pin 12

Componenta	Pin componenta	Conectare
Rezistor 1kΩ (R1)	Capăt 1	PD4 ATmega328P
Rezistor 1kΩ (R1)	Capăt 2	RX DFPlayer Mini #1 (Protecție nivel logic 5V→3.3V)
Rezistor 1kΩ (R2)	Capăt 1	PD6 ATmega328P
Rezistor 1kΩ (R2)	Capăt 2	RX DFPlayer Mini #2 (Protecție nivel logic 5V→3.3V)
Rezistor 1kΩ (R3)	Capăt 1	PD2 ATmega328P
Rezistor 1kΩ (R3)	Capăt 2	RXD HC-05 + Nod divizor (Parte superioară divizor)
Rezistor 2kΩ (R4)	Capăt 1	Nod divizor
Rezistor 2kΩ (R4)	Capăt 2	GND (Parte inferioară divizor tensiune 5V→3.3V)
Rezistor 1kΩ (R5)	Capăt 1	DAC_R DFPlayer #1
Rezistor 1kΩ (R5)	Capăt 2	Nod mixare audio R (Izolare ieșire DAC)
Rezistor 1kΩ (R6)	Capăt 1	DAC_R DFPlayer #2
Rezistor 1kΩ (R6)	Capăt 2	Nod mixare audio R (Izolare ieșire DAC)
Rezistor 100Ω (R7)	Capăt 1	Nod mixare audio R
Rezistor 100Ω (R7)	Capăt 2	Pin R (Ring) Modul Jack TRRS (Protecție canal drept)
Rezistor 1kΩ (R8)	Capăt 1	DAC_L DFPlayer #1
Rezistor 1kΩ (R8)	Capăt 2	Nod mixare audio L (Izolare ieșire DAC)
Rezistor 1kΩ (R9)	Capăt 1	DAC_L DFPlayer #2
Rezistor 1kΩ (R9)	Capăt 2	Nod mixare audio L (Izolare ieșire DAC)
Rezistor 100Ω (R10)	Capăt 1	Nod mixare audio L
Rezistor 100Ω (R10)	Capăt 2	Pin L (Tip) Modul Jack TRRS (Protecție canal stâng)



## Descrierea Arhitecturii Hardware

Ne folosim de placa ATmega328P Xplained Mini ca microcontroller principal, aceasta e responsabilă cu procesarea semnalelor de la senzori, controlul redării audio, comunicația Bluetooth și afișarea pe ecran.

Detectarea loviturilor se realizează cu două module IMU MPU-6050 montate pe vârful bețelor, conectate pe același bus I2C (SDA/PC4, SCL/PC5). Diferențierea lor se face prin pinul AD0: senzorul bâtei stângi are AD0 la GND (adresă 0x68), cel al bâtei drepte la 3.3V (adresă 0x69). Pe același bus I2C este conectat și ecranul LCD 16x2 (adresă 0x27), care afișează stilul curent și ultima lovitură detectată.

Redarea sunetelor este asigurată de două module DFPlayer Mini cu card SD, câte unul per bătă, permițând redarea simultană și independentă a sunetelor. Comunicația se face prin SoftwareSerial: DF1 pe PD4/PD5, DF2 pe PD6/PD7. Pe liniile TX ale microcontrolerului spre RX-ul DFPlayer-elor sunt plasate rezistoare de 1k $\Omega$  pentru protecția intrărilor de 3.3V față de logica de 5V a ATmega-ului. Ieșirile audio DAC\_R și DAC\_L ale ambelor module sunt mixate pasiv prin rezistoare de 1k $\Omega$  și dirijate spre jack-ul TRRS 3.5mm prin rezistoare de 100 $\Omega$ .

Transmiterea datelor despre loviturile detectate spre aplicația mobilă se realizează prin modulul Bluetooth HC-05, conectat pe PD2/PD3. Linia RXD a modulului este protejată printr-un divizor de tensiune 1k $\Omega$ +2k $\Omega$ .

În cadrul testelor a fost identificată și rezolvată o problemă de ground loop audio: alimentarea ATmega-ului din laptopul conectat la priză, simultan cu amplificatorul de chitară, genera un zgomot continuu cauzat de diferența de potențial dintre cele două mase. Soluția aplicată a fost alimentarea plăcii din telefon sau power bank. GND-ul jack-ului audio este rutat direct la GND-ul DFPlayer-ului, izolând masa de semnal audio de restul circuitului.

# Software Design

## Mediu de dezvoltare:

- PlatformIO (VS Code) cu framework Arduino pentru ATmega328P
- Board: ATmega328P Xplained Mini, upload via xplainedmini programmer
- Monitor serial la 57600 baud pentru debug în timp real

## Librării folosite:

- Wire.h — comunicație I2C nativă cu MPU-6050 #1, #2 și LCD
- SoftwareSerial.h — emulare UART pe pini digitali pentru DFPlayer #1, DFPlayer #2 și HC-05 Bluetooth
- DFRobotDFPlayerMini — librărie 3rd-party pentru controlul modulelor MP3 prin protocolul serial DFPlayer (comenzi play, stop, loop, volume, EQ)
- LiquidCrystal\_I2C — librărie 3rd-party pentru controlul ecranului LCD 16×2 prin interfața I2C

## Algoritmi și structuri de date:

Detectarea loviturilor se bazează pe o mașină de stări cu trei stări per bătă (READY → DETECTING → COOLING), implementată în funcția process(). La fiecare iterație de 5ms, microcontrolerul citește accelerația brută de la MPU-6050 și calculează magnitudinea vectorului de accelerație ( $|dx| + |dy| + |dz|$ ). Dacă aceasta depășește pragul HIT\_THRESHOLD (6000 LSB, corespunzând la aproximativ 1.5g în modul  $\pm 8g$ ), starea trece în DETECTING. În fereastra de detecție (6 samples  $\times$  5ms = 30ms), firmware-ul urmărește peak-ul pozitiv și negativ pe axa Y. Direcția loviturii (STANGA, MIJLOC, DREAPTA) se determină prin compararea celor două peak-uri la finalul ferestrei. Starea COOLING previne detectarea multiplă a aceleiași lovituri, cu un timeout de 250ms sau 2 samples calme consecutive.

Fiecare bătă este reprezentată printr-o structură Bat care grupează adresa I2C, un pointer la instanța DFPlayer corespunzătoare, offset-urile de calibrare, starea mașinii și valorile de peak.

Sunetele redade depind de stilul activ (ROCK sau METAL), calculat dinamic ca offset față de numărul de fișier de bază: stilul ROCK folosește fișierele 1-3 pe ambele carduri, stilul METAL folosește fișierele 4-6 pe DF1 și 5-7 pe DF2.

LED-ul RGB este controlat printr-un algoritm de fade non-blocant bazat pe funcția stepTo(), care interpolează liniar valorile curente spre valorile țintă cu 8 pași per iterație (aproximativ 160ms per tranziție completă).

## Surse și funcții implementate:

- mpu\_init(addr) — inițializează MPU-6050: dezactivare sleep, configurare  $\pm 8g$ , DLPF 44Hz
- calibrate(bat) — calculează offset-urile de repaus prin medierea a 200 de samples la pornire
- mpu\_read(addr, ax, ay, az) — citește cei 6 bytes de accelerometru prin I2C
- process(bat) — rulează mașina de stări de detecție pentru o bătă
- classify(bat) — determină direcția loviturii, redă sunetul corespunzător, actualizează LCD, trimite date pe Bluetooth și setează culoarea LED-ului
- setLedColor(r, g, b) / updateLed() — controlul non-blocant al LED-ului RGB cu fade smooth
- setup() — inițializare hardware: LED, butoane, I2C, LCD, ambele DFPlayer-e cu debug complet,

Bluetooth, MPU-uri, calibrare

- loop() — buton pauză/resume cu melodie pe loop, buton schimbare stil, procesare senzori, actualizare LED

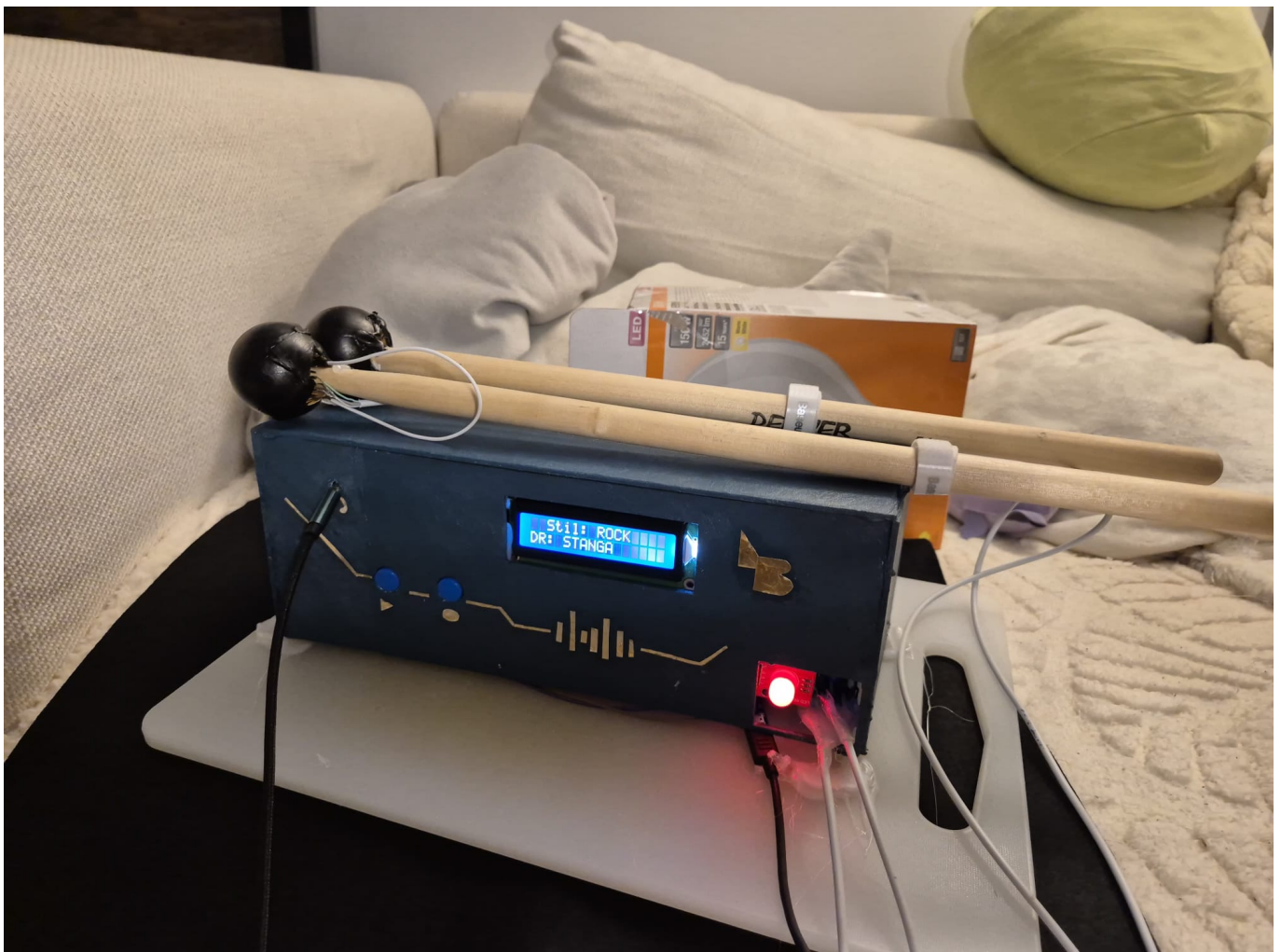
## Rezultate Obținute

Proiectul funcționează. Lovești în aer cu două bețe și se aude ca și cum ai fi un toboșar, doar că nu ești, și nici nu ai fi fără acest sistem.

Concret:

- Ambele bâte detectează corect direcția loviturii (stânga, mijloc, dreapta) în timp real
- Sunetele celor două bâte se redau simultan fără interferențe
- Schimbarea stilului între ROCK și METAL funcționează instantaneu
- LED-ul RGB confirmă vizual fiecare lovitură cu tranziții de culoare smooth
- Aplicația Android afișează în timp real toba lovită și vizualizatorul audio reacționează la fiecare impact
- Ecranul LCD afișează stilul activ și ultima lovitură detectată
- Butonul de pauză oprește detecția și pornește o melodie pe loop, util când vrei să te prefaci că știi să cânti la chitară

Scheletul produsului final:



## Concluzii


Proiectul demonstrează că un instrument muzical de percuție virtual poate fi construit cu componente accesibile și un microcontroler de clasă entry-level. ATmega328P s-a dovedit suficient pentru a gestiona simultan două magistrale I2C, trei instanțe SoftwareSerial și un algoritm de detecție în timp real, fără întârzieri perceptibile.

Principalele provocări tehnice întâlnite au fost gestionarea conflictelor dintre instanțele SoftwareSerial, eliminarea ground loop-ului audio și calibrarea pragurilor de detecție pentru a obține un echilibru între sensibilitate și rata de detecții false. Fiecare problemă a fost diagnosticată și rezolvată iterativ direct pe hardware.

Ca direcții de dezvoltare ulterioară, sistemul poate fi extins cu un al doilea canal Bluetooth dedicat transmisiei audio wireless, înlocuirea SoftwareSerial cu hardware UART pentru o comunicație mai robustă și adăugarea unui mod de înregistrare a secvențelor de lovituri.

În ansamblu, proiectul acoperă cu succes noțiunile din laboratoarele de GPIO, UART, Timere/PWM și I2C, integrându-le într-un sistem funcțional și, cel mai important, unul pe care îl poți folosi efectiv la o petrecere dacă oamenii sunt suficient de politicoși să nu comenteze.

## Download

O arhivă (sau mai multe dacă este cazul) cu fișierele obținute în urma realizării proiectului: surse, scheme, etc. Un fișier README, un ChangeLog, un script de compilare și copiere automată pe uC crează întotdeauna o impresie bună . Fișierele se încarcă pe wiki folosind facilitatea **Add Images or other files**. Namespace-ul în care se încarcă fișierele este de tipul **:pm:prj20??:c?** sau **:pm:prj20??:c?:nume\_student** (dacă este cazul). **Exemplu:** Dumitru Alin, 331CC → **:pm:prj2009:cc:dumitru\_alin**.

Link github: [https://github.com/cclaryc/Aero\\_Beat.git](https://github.com/cclaryc/Aero_Beat.git)

## Jurnal

Puteți avea și o secțiune de jurnal în care să poată urmări asistentul de proiect progresul proiectului.

- Săptămâna 1: Alegerea componentelor, testare conectivitate I2C cu MPU-6050
- Săptămâna 2: Implementare algoritm de detecție lovitură, calibrare pe axa Y
- Săptămâna 4: Integrare DFPlayer Mini, LED RGB, aplicație mobilă

- Săptămâna 5: Integrare Ecran LED, buton pentru schimbarea genului tobelor.
- Last 100m : Buton pentru pause/start, muzica de fundal pause

## Bibliografie/Resurse

### Resurse Hardware:

- MPU-6050 Product Specification — InvenSense
- MPU-6050 Register Map and Descriptions — InvenSense (RM-MPU-6000A)
- ATmega328P Xplained Mini User Guide — Microchip (DS50002659B)
- ATmega328P Datasheet — Microchip
- DFPlayer Mini Datasheet — DFRobot
- GY-521 Module Schematic
- HC-05 Bluetooth Module AT Command Set
- LiquidCrystal I2C — HD44780 Controller Datasheet

### Resurse Software:

- Arduino Wire Library Documentation — <https://www.arduino.cc/en/reference/wire>
- Arduino SoftwareSerial Library — <https://www.arduino.cc/en/Reference/SoftwareSerial>
- PlatformIO Documentation — <https://docs.platformio.org>
- DFRobotDFPlayerMini Library — <https://github.com/DFRobot/DFRobotDFPlayerMini>
- LiquidCrystal\_I2C Library — [https://github.com/johnrickman/LiquidCrystal\\_I2C](https://github.com/johnrickman/LiquidCrystal_I2C)
- Android Bluetooth SPP Development — <https://developer.android.com/guide/topics/connectivity/bluetooth>
- ATmega328P PWM and Timer Registers — <https://www.arduino.cc/en/Reference/AnalogWrite>

[Export to PDF](#)

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

Permanent link:  
<http://ocw.cs.pub.ro/courses/pm/prj2026/ciprian.popescu0411/clarisse.braviceanu>



Last update: **2026/05/25 02:23**