ENSEA

Beyond Engineering

APLM

(**A**utomated **P**iano **L**earning **M**odule)

Project semester 7 2024-2025

Contents

ENSEA
Beyond Engineering

I. Project Selection and Definition

ENSEA
Beyond Engineering

I. Project Selection and Definition

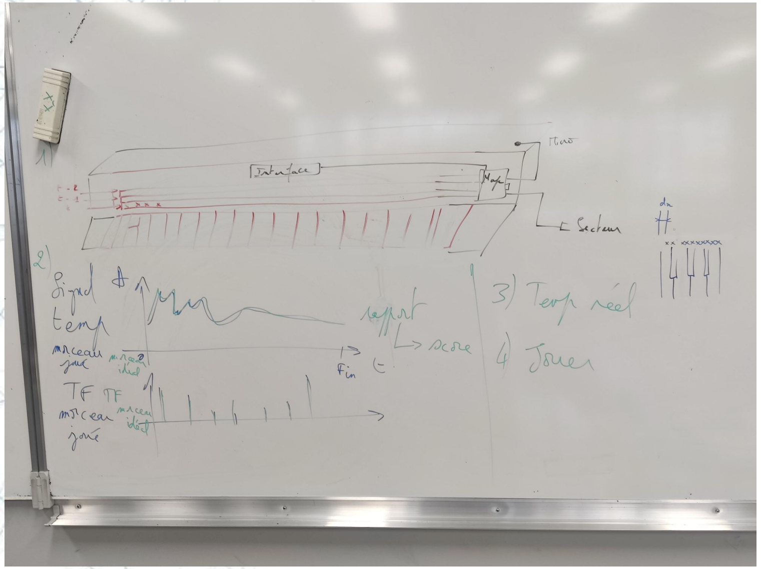


Figure 1: General outline of the project

Figure 2: LED ARGB strip

II. Distribution of Roles

II. Distribution of Roles

Justine Hazan: PCB, Component research and control

Mély Galvez : Audio signal processing code

Jiajing Li: PCB, component research and control

Marin Kerboriou: MIDI file processing code

III. Component Search and Order

ENSEA
Beyond Engineering

III. Component Search and Order

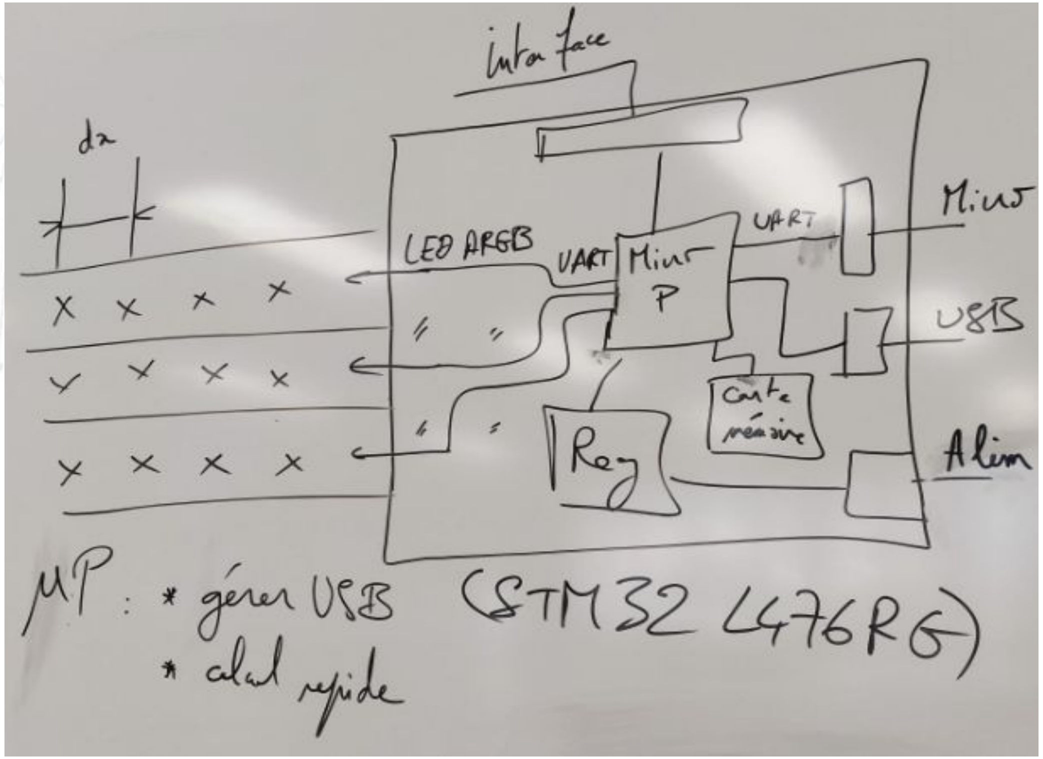


Figure 3: Main components and their place

III. Component Search and Order

Figure 4: Microprocessor

Purpose of the component:

- Order the PCB components

III. Component Search and Order

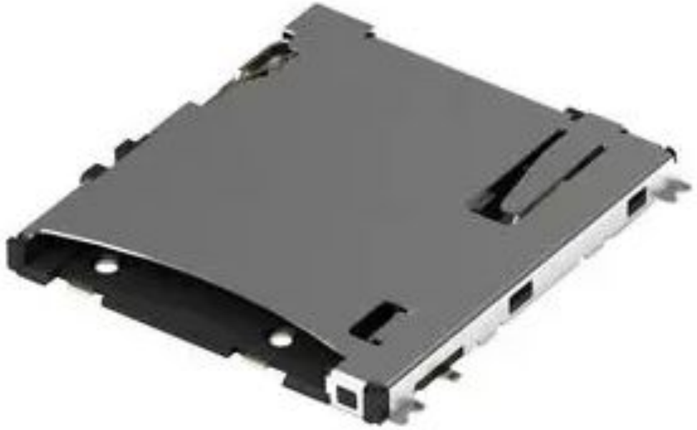


Figure 5: SD card port

Purpose of the component:

- Store data

III. Component Search and Order

Figure 6: AC/DC adapter

Figure 7: Power Jack DC

Purpose of the component:

- To have the correct alimentation for the PCB

ENSEA
Beyond Engineering

III. Component Search and Order

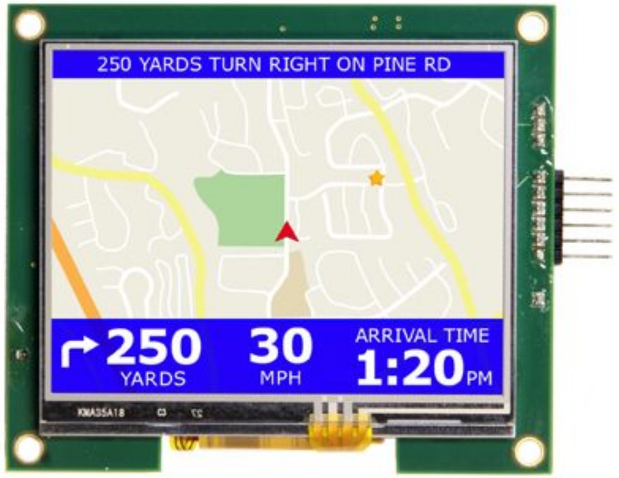


Figure 8: interface

Purpose of the component:

- Displays information

III. Component Search and Order

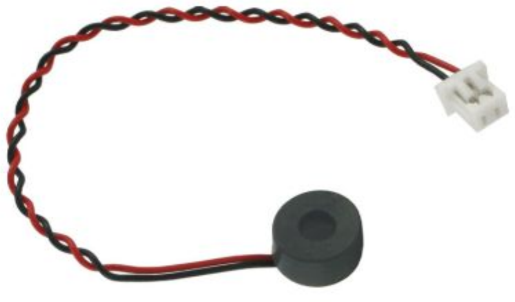


Figure 9: micro

Purpose of the component:

- Captures sound

ENSEA
Beyond Engineering

III. Component Search and Order

Figure 10: Led ARGB

Purpose of the component:

- Shows the note when they should be played

IV. Schematic and PCB Design

ENSEA
Beyond Engineering

V. Conception du Schéma et du PCB

INCOMING!

Figure 10: KiCad schematic

V. Processing a MIDI file

ENSEA
Beyond Engineering

VI. Processing a MIDI file

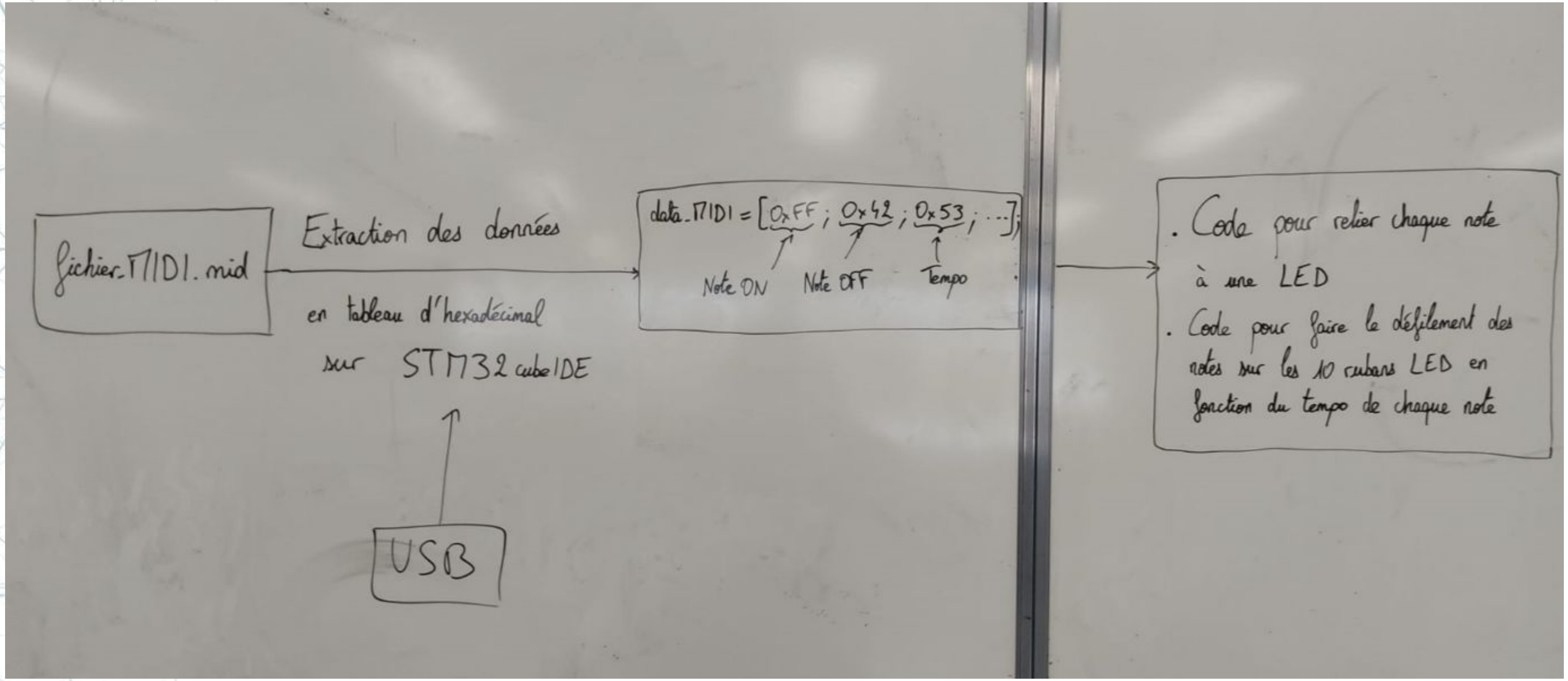

fichier_MIDI.mid
Extraction des données
en tableau d'hexadécimal
sur STM32 cubeIDE
USB
data_MIDI = [0xFF ; 0x42 ; 0x53 ; ...];
Note ON
Note OFF
Tempo
. Code pour relier chaque note à une LED
. Code pour faire le défilement des notes sur les 10 rubans LED en fonction du tempo de chaque note

Figure 11: Outline for processing a MIDI file

VI. Processing a MIDI file

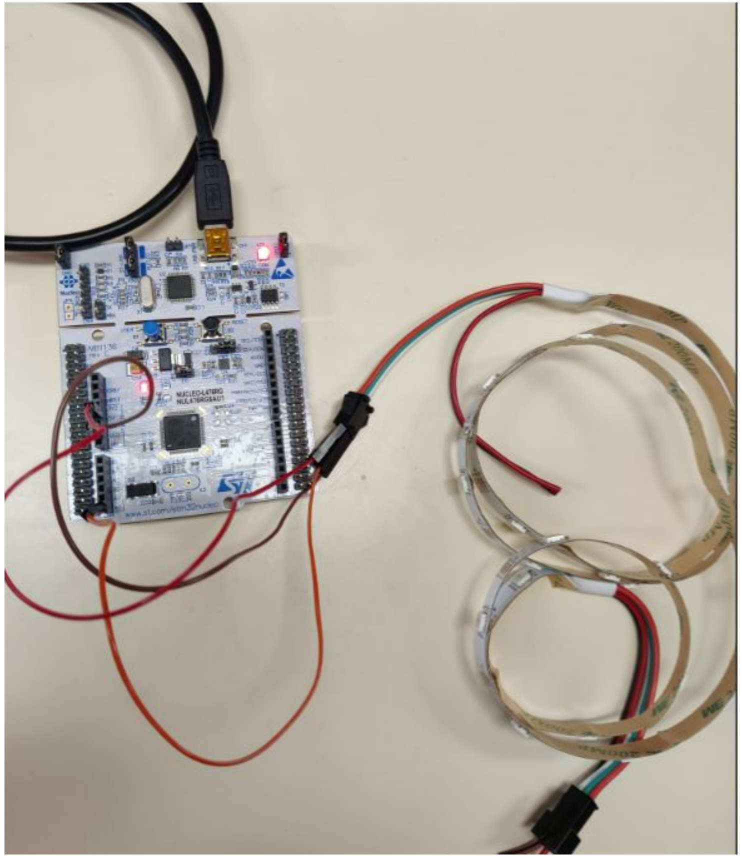


Figure 12: Connexion STM32 with LED strip

VI. Digital audio signal processing

ENSEA
Beyond Engineering

VI. Digital audio signal processing

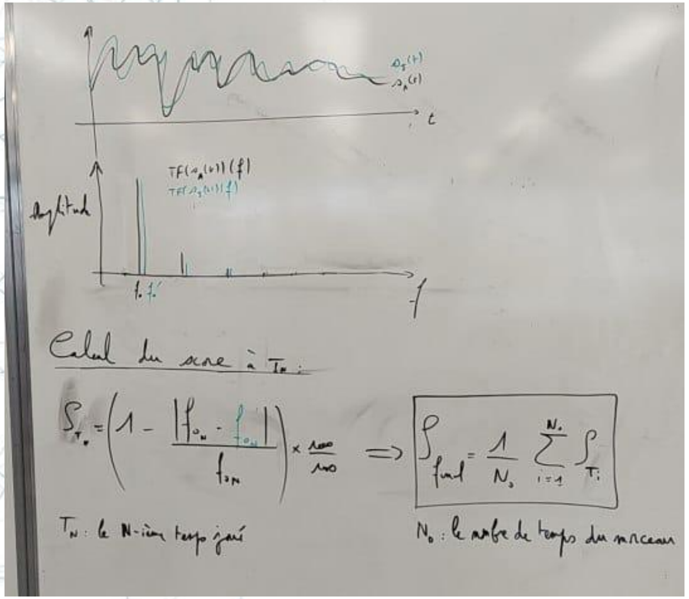

t
Amplitude
TF(s_A(t))(f)
TF(s_J(t))(f)
f
Calcul du score à T_N :
$S_{T_N} = \left(1 - \frac{|f_{0N} - f_{0N}|}{f_{0N}}\right) \times \frac{100}{100}$
$\Longrightarrow \boxed{S_{final} = \frac{1}{N_0} \sum_{i=1}^{N_0} S_{T_i}}$
T_N : le N-ième temps joué
N_0 : le nombre de temps du morceau

Figure 13: Method for evaluating the user score

CMSIS
COMPLIANT
ARM® Cortex™ Microcontroller Software Interface Standard

VI. Digital audio signal processing

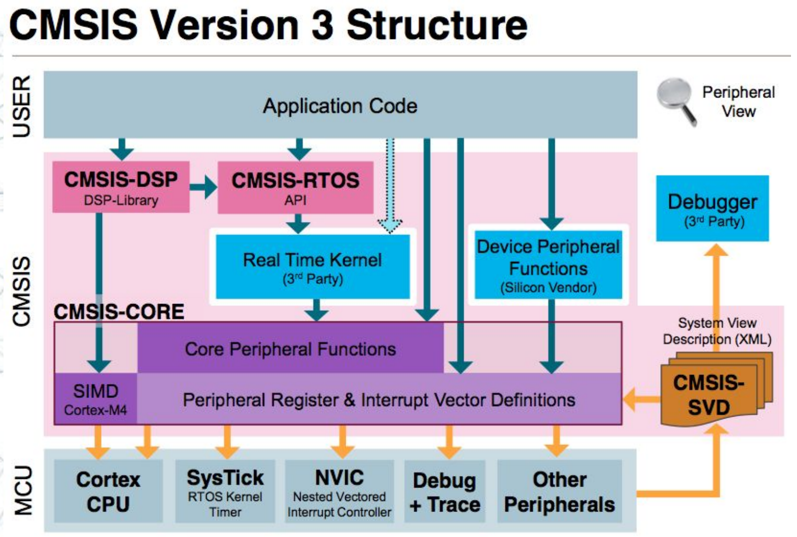
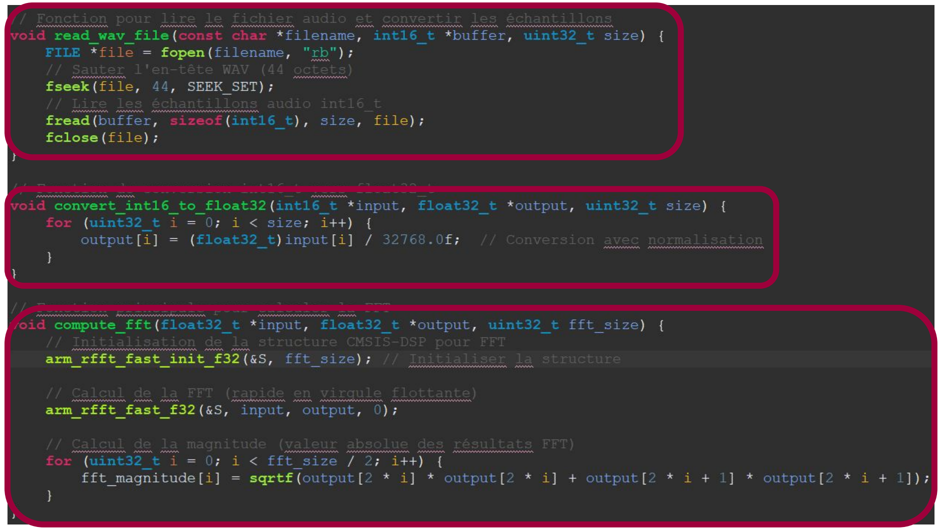


Figure 14: How CMSIS works



```c
// Fonction pour lire le fichier audio et convertir les échantillons
void read_wav_file(const char *filename, int16_t *buffer, uint32_t size) {
    FILE *file = fopen(filename, "rb");
    // Sauter l'en-tête WAV (44 octets)
    fseek(file, 44, SEEK_SET);
    // Lire les échantillons audio int16_t
    fread(buffer, sizeof(int16_t), size, file);
    fclose(file);
}

// Fonction de conversion int16_t vers float32_t
void convert_int16_to_float32(int16_t *input, float32_t *output, uint32_t size) {
    for (uint32_t i = 0; i < size; i++) {
        output[i] = (float32_t)input[i] / 32768.0f;  // Conversion avec normalisation
    }
}

// Fonction principale pour calculer la FFT
void compute_fft(float32_t *input, float32_t *output, uint32_t fft_size) {
    // Initialisation de la structure CMSIS-DSP pour FFT
    arm_rfft_fast_init_f32(&S, fft_size); // Initialiser la structure

    // Calcul de la FFT (rapide en virgule flottante)
    arm_rfft_fast_f32(&S, input, output, 0);

    // Calcul de la magnitude (valeur absolue des résultats FFT)
    for (uint32_t i = 0; i < fft_size / 2; i++) {
        fft_magnitude[i] = sqrtf(output[2 * i] * output[2 * i] + output[2 * i + 1] * output[2 * i + 1]);
    }
}
```

Figure 15: Code

VI. Digital audio signal processing

Problems encountered:

- importing the library
- inclusion problems

Possible solutions:

- follow an online step-by-step method for importing

ENSEA
Beyond Engineering

ENSEA

Beyond Engineering

Thank you for your time and attention!