

TP - Envoi d'une trame MIDI avec Teensy

Objectif

Comprendre et mettre en œuvre l'envoi d'un message MIDI à l'aide d'un microcontrôleur Teensy.

Le but est de générer un message NoteOn et NoteOff pour piloter un instrument virtuel (logiciel MAO) ou un synthétiseur externe.

Matériel nécessaire

- 1 carte Teensy 4.0
- 1 câble USB → PC
- Arduino IDE
- Le logiciel de MAO Ableton pour visualiser les messages

Connexions

La Teensy 4.0 envoie directement du MIDI USB → rien à câbler, seulement brancher la carte en USB.

Dans l'IDE Arduino, sélectionner « USB Type » : « Serial + MIDI ».

Welcome to Teensy® 4.0

32 Bit Arduino-Compatible Microcontroller

To begin using Teensy, please visit the website & click [Getting Started](http://www.pjrc.com/teensy).
www.pjrc.com/teensy

Digital Pins
digitalRead
digitalWrite
pinMode

Analog Pins
analogRead

PWM Pins
analogWrite

Digital Audio
Audio Library

Serial Ports
Serial1 - Serial7

I²C Port
Wire Library

SPI Port
SPI Library

CAN Bus
FlexCAN_t4 Library

Red LED
Loading Status
dim: Ready
bright: Writing
blink: No USB

Vin (3.6 to 5.5 volts)
3.3V (250 mA max)
GND

On/Off Program
GND
3.3V
VBat

All digital pins have interrupt capability.

Teensy® 4.0 Back Side

Additional pins and features available on the back side

Use 3V coin cell for Date & Time and power management features

Teensy 4.0 signal pins are not 5V tolerant. Do not apply more than 3.3 volts to any pin, except VIN or VUSB.

For solutions to the most common issues and technical support, please visit:
www.pjrc.com/help

Teensy 4.0 System Requirements:
PC computer with Windows 7, 8, 10 or later
or Ubuntu Linux 14.04 or later
or Macintosh OS-X 10.8 or later
USB Micro-B Cable

7 14833 87948 0

Envoi d'une note :

Code Arduino

Voici un exemple de code qui envoie une trame MIDI avec une note unique répété en boucle :

```
// Programme Teensy : Envoi périodique d'une note MIDI sur USB
// Compatible mode USB : Serial + MIDI

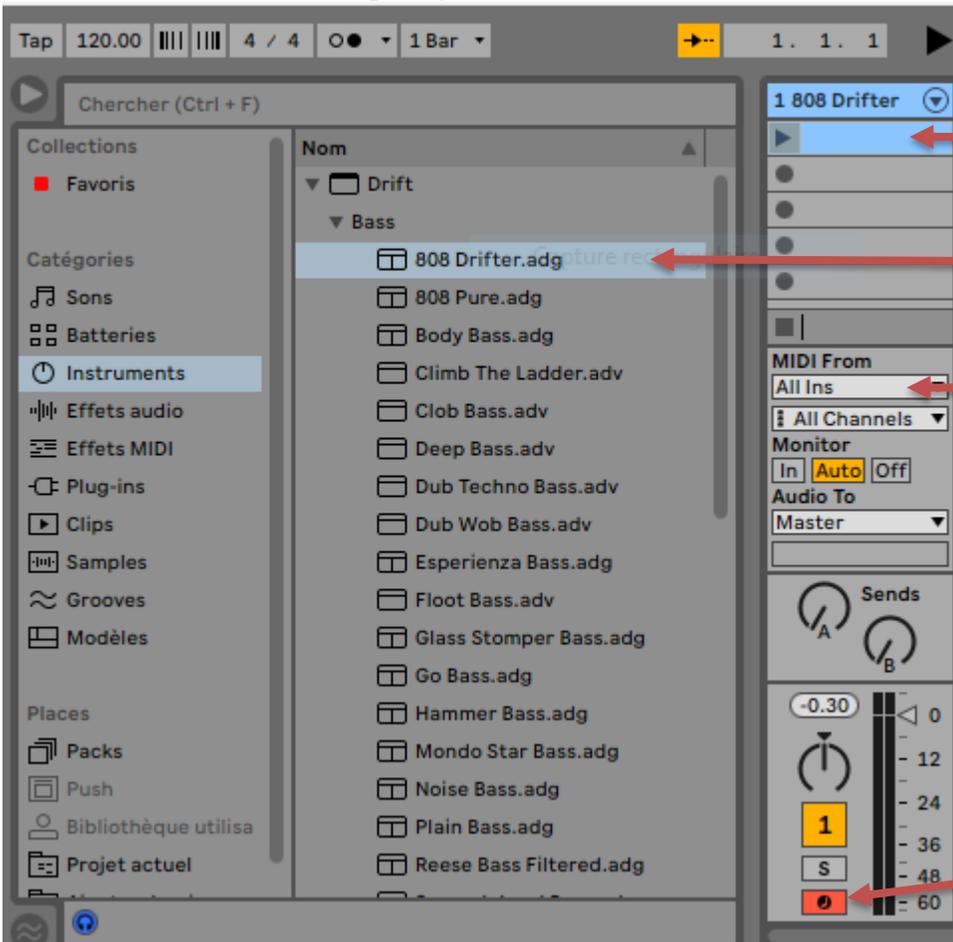
void setup() {
  Serial.begin(9600);
  Serial.println("=== Début du test MIDI USB ===");
}

void loop() {
  // Envoi Note On
  usbMIDI.sendNoteOn(60, 100, 1); // Note 60 = Do3
  Serial.println("Note ON envoyée (Do3)");
  delay(1000);

  // Envoi Note Off
  usbMIDI.sendNoteOff(60, 0, 1);
  Serial.println("Note OFF envoyée");
  delay(1000);
}
```

Activité

1. Téléverser le programme dans la carte Teensy.
2. Ouvrir le logiciel Ableton. Dans le menu Préférences (Ctrl + ,), aller dans Link / Tempo / MIDI.
3. Repérer la Teensy. Dans la section MIDI Ports, tu dois voir :
 - Input : Teensy MIDI
 - Output : Teensy MIDI
4. Activer la réception des notes
 - Dans la ligne Input : Teensy MIDI, activer « Piste » (clic sur le bouton jusqu'à ce qu'il devienne jaune).
 - Laisser Sync, Remote, MPE désactivés pour l'instant. Piste = indispensable pour que les notes arrivent dans une piste MIDI.
5. Sur une piste MIDI, double-cliquer sur un clip vide (voir capture ci-dessous).
6. Sur la même piste, dans MIDI From, choisir Teensy MIDI.
7. Activer le bouton Arm Recording (cercle rouge dans la piste).
8. Charger un instrument logiciel (par ex. 808 Drifter).
9. Quand la Teensy envoie *sendNoteOn(60, 100, 1)*, le petit indicateur MIDI IN (carré jaune en haut à droite de la fenêtre Ableton) clignote. L'instrument joue la note. Avec *sendNoteOff*, la note s'arrête.



Clip

Instrument

Choix du contrôleur

Arm recording

Changement d'un paramètre :

Code Arduino

Voici un exemple de code qui envoie une trame MIDI avec changement d'un paramètre en fonction du temps :

```
// Envoi périodique d'un message MIDI Control Change (CC)
// avec affichage Serial optimisé (une seule ligne de print)

int ccNumber = 1;    // Numéro du contrôleur (1 = modulation)
int channel  = 1;    // Canal MIDI
int value    = 0;    // Valeur envoyée
int step     = 10;   // Incrémentation

void setup() {
  Serial.begin(9600);
  Serial.println("CC Test");
}

void loop() {
  usbMIDI.sendControlChange(ccNumber, value, channel);

  // Affichage compact et rapide (1 seule ligne)
  Serial.printf("Ch:%d CC:%d Val:%d\n", channel, ccNumber,
value);

  // Fait varier la valeur entre 0 et 127
  value += step;
  if (value > 127 || value < 0) {
    step = -step;
    value += step;
  }

  delay(200);
}
```

Activité

1. Téléverser le programme dans la carte Teensy.
2. Dans le menu Préférences (Ctrl + ,) d'Ableton, aller dans Link / Tempo / MIDI.
3. Repérer la Teensy. Dans la section MIDI Ports, tu dois voir :
 - Input : Teensy MIDI
 - Output : Teensy MIDI
4. Activer la réception des contrôles
 - Dans la ligne Input : Teensy MIDI, activer « Remote » (clic sur le bouton jusqu'à ce qu'il devienne jaune).
 - Laisser Sync et MPE désactivés. Remote = indispensable pour que les changements de paramètres soient pris en compte.

5. En haut à droite de la fenêtre Ableton, cliquer sur « MIDI », on rentre ainsi dans l'assignation MIDI, le logiciel va repérer qu'un message MIDI de contrôle est envoyé sur un certain canal et va alors assigner ce canal au paramètre sur lequel nous allons cliquer. Cliquer alors sur Filter Cutoff.

Intégration d'un capteur :

Activité

Créer un programme permettant de modifier deux paramètres (effets) d'Ableton grâce à deux potentiomètres.

Remarques :

- Pour éviter que le programme Arduino envoie constamment des messages MIDI il est nécessaire de compléter le programme en envoyant un message MIDI seulement quand une valeur de capteur change.
- La carte Teensy fonctionne en 3,3V. Pour éviter toute perte d'information il faut alimenter les capteurs analogiques en 3,3V.