



## LE PROTOCOLE MIDI

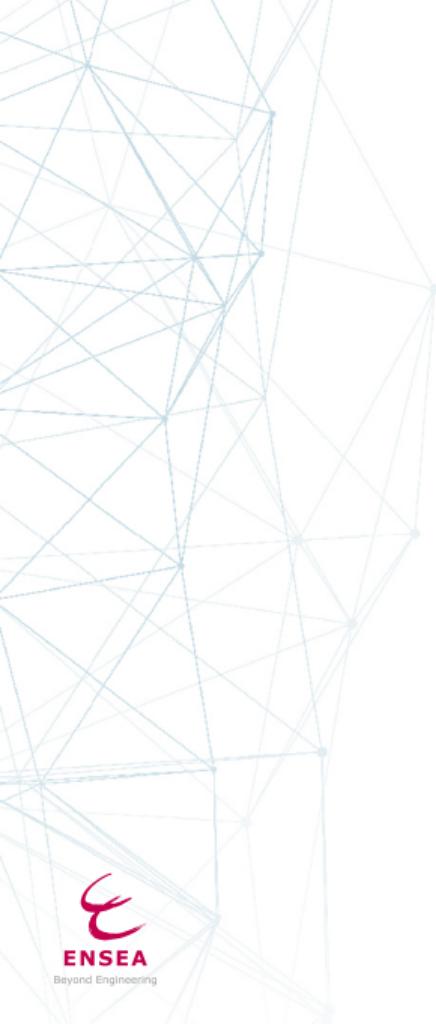
Connecter des trucs musicaux entre eux

Laurent Fiack – [laurent.fiack@ensea.fr](mailto:laurent.fiack@ensea.fr)

Bureau D212/D060

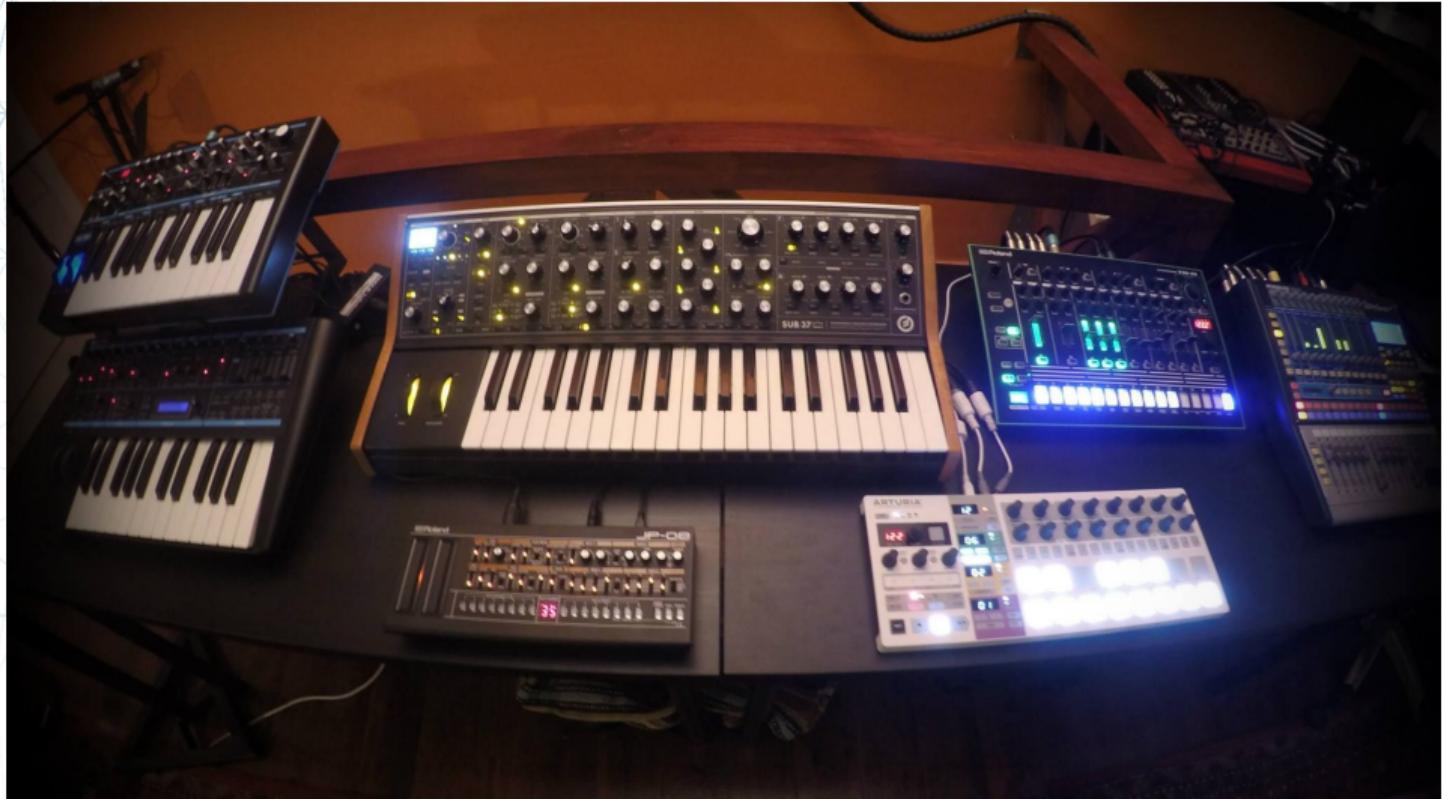
# Sommaire

- 1 Pourquoi le MIDI?**
- 2 Le protocole**
  - La couche physique
  - Les messages MIDI
- 3 USB MIDI**
  - USB Class Compliant
  - USB MIDI Class
- 4 TD Machine**
  - Mise en place d'un device USBMidi sur carte STM32F746 Discovery

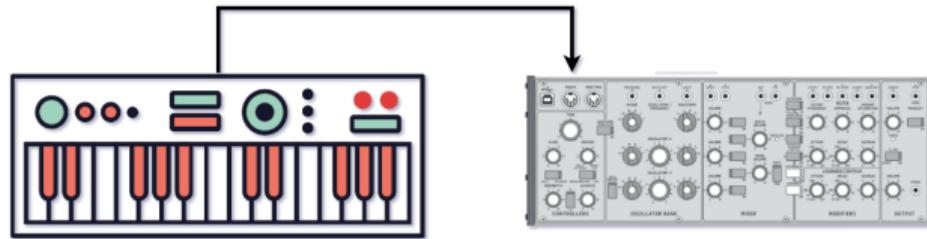


# Pourquoi le MIDI?

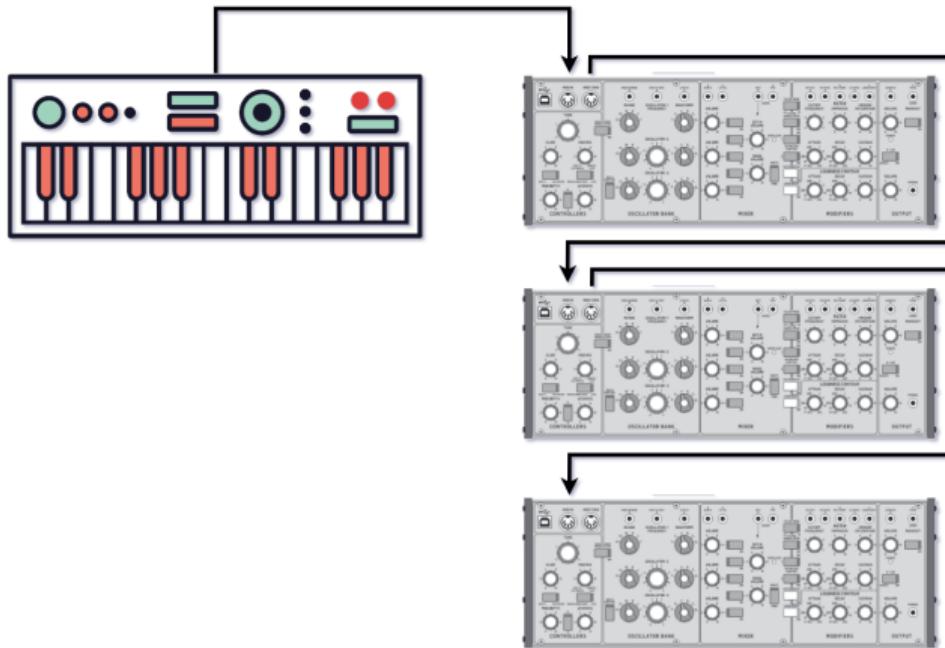
# Connecter des trucs musicaux entre eux



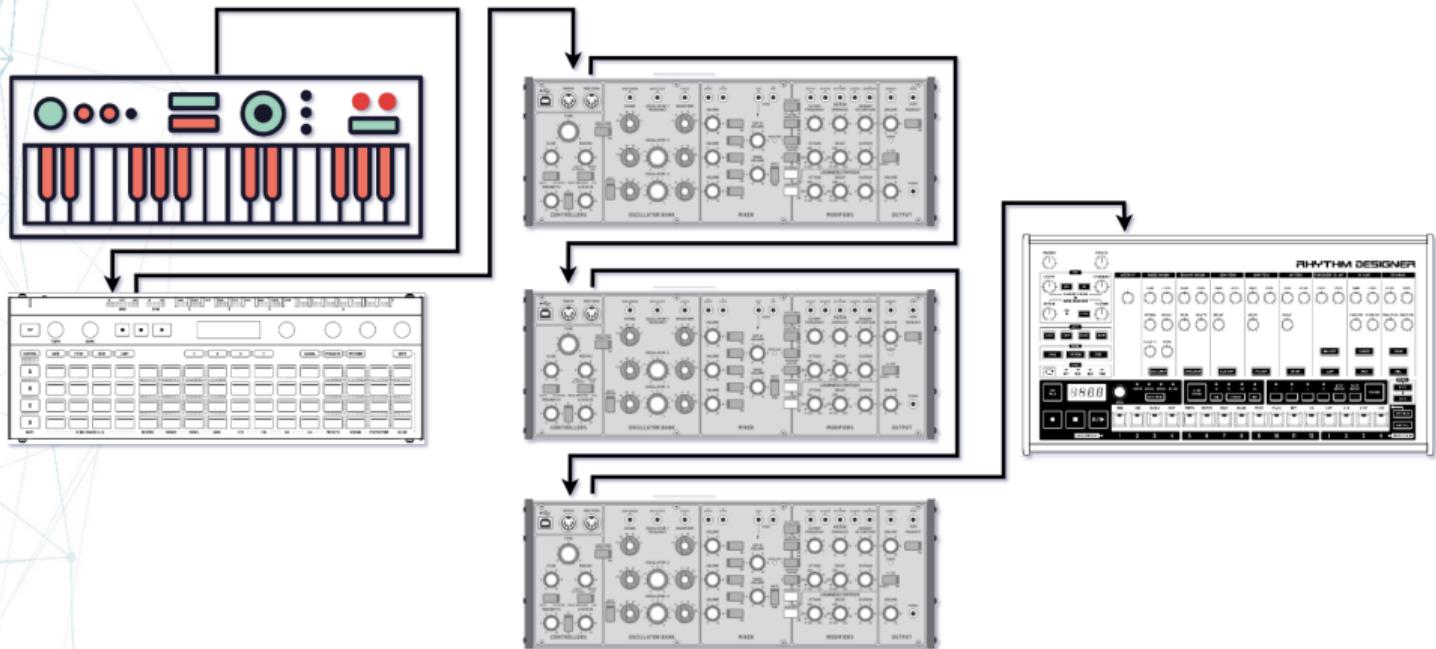
# Clavier maître → synthétiseur



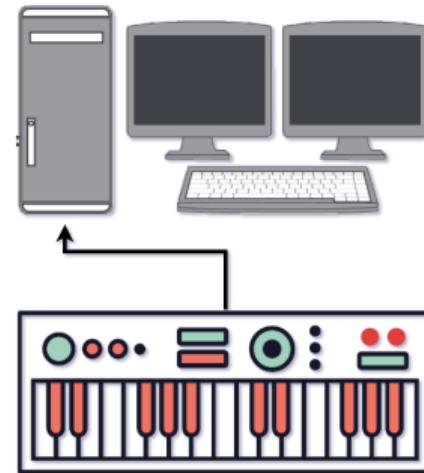
# Plusieurs synthétiseurs



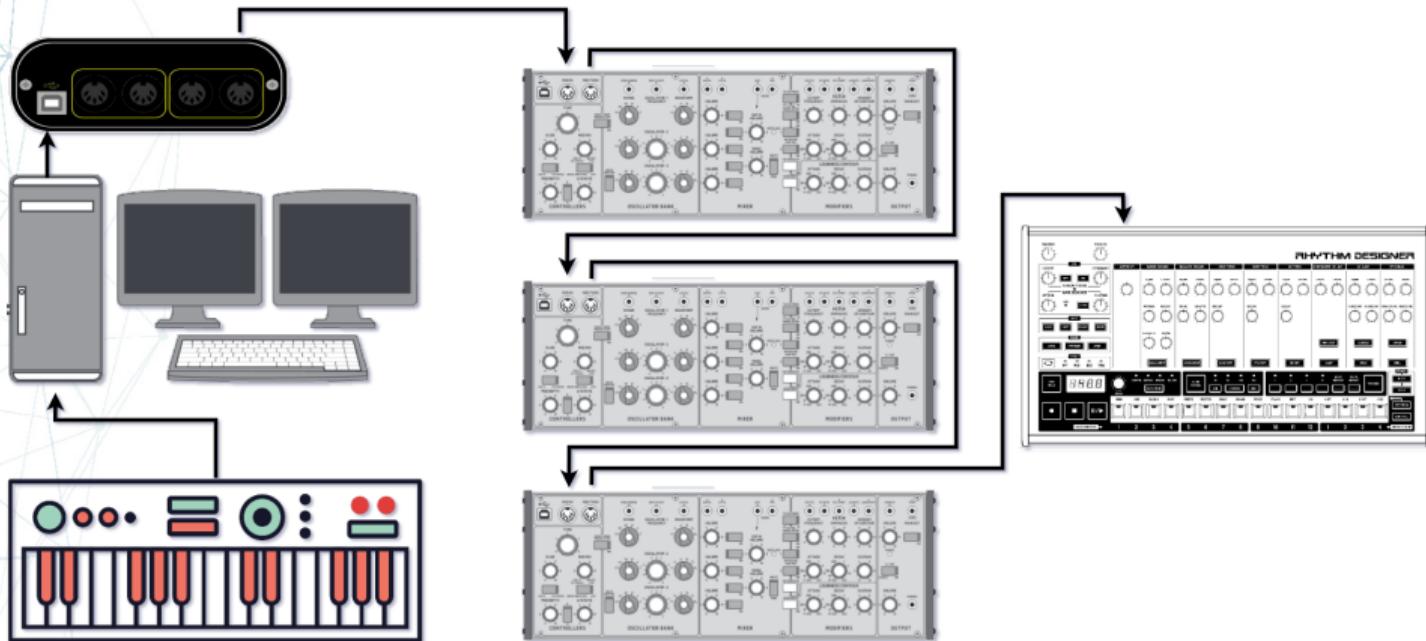
# Séquenceur



# Avec un PC

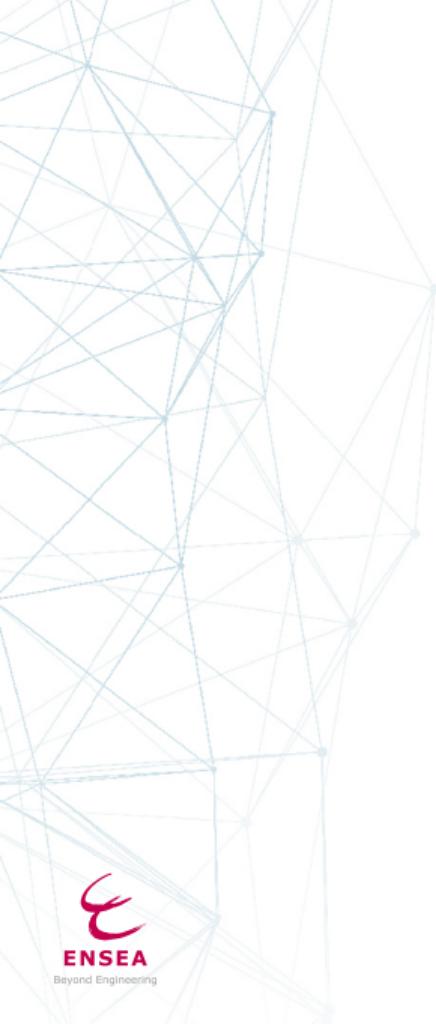


# Studio



# Historique

- Première spec en 1981 par Sequential Circuits et Roland
  - Rapidement accepté par les Japonais Yamaha, KORG, Kawai et les Americains Moog
- Standardisé par la MIDI Manufacturers Association (MMA) en 1983
- Connection entre Prophet 600 et Roland JP-6 en 83
- En 83 encore, TR-909 et séquenceur MSQ-700
- 1991 : format de fichier MIDI
- 1999 : USB Midi class compliant (USB 1.0 en 1996)
- 2020 : MIDI 2.0 bidir compatible MIDI 1.0

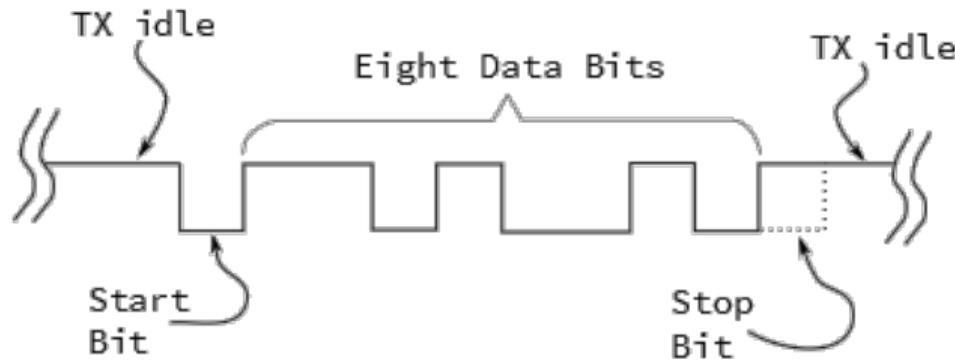


# Le protocole

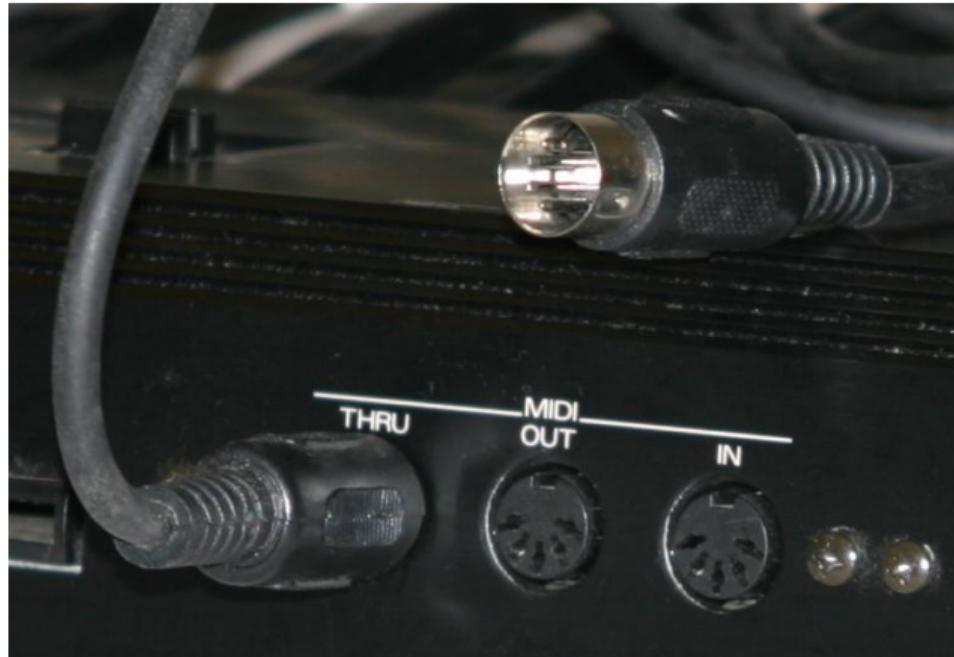
## La couche physique

# Protocole bas niveau

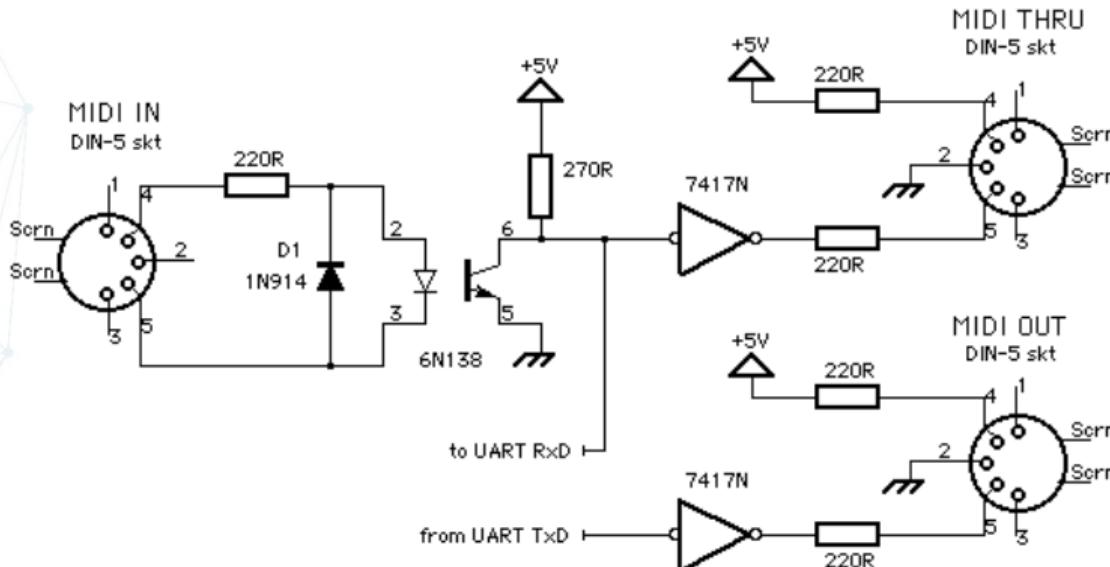
- Liaison Série Asynchrone (= UART)
- 31250 bits/seconde, 1 bit start, 1 bit stop, pas de parité
- Message MIDI = 1 octet de status (+ éventuellement data)
  - Octet Status : MSB = 1
  - Octet Data : MSB = 0
- Message classique = 3 octets : 1 Status, 2 Data
  - Environ 1 ms



# Connectique : DIN 5 broches

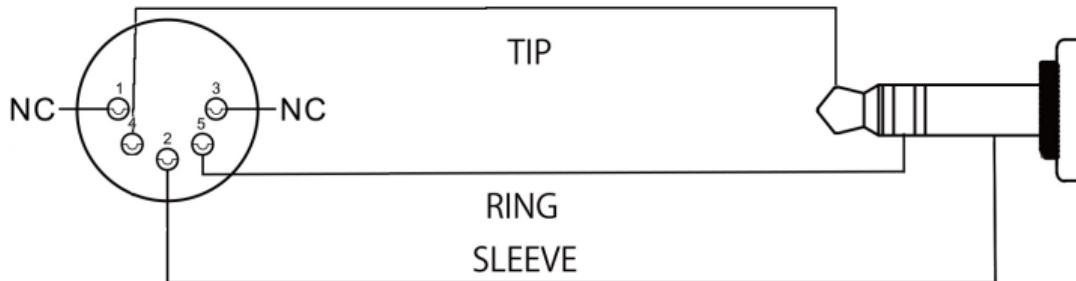


# Câblage



- Utilisation d'un optocoupleur
- Évite les boucles de masse

# Connectique alternative : Jack TRS 3.5mm



Tip —— Pin 4

Ring —— Pin 5

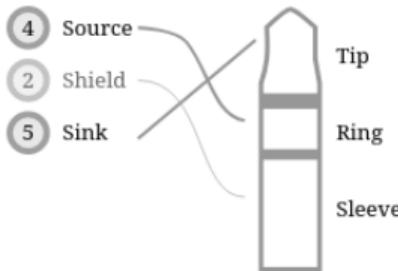
Sleeve —— Pin 2

- Standard recommande 2.5mm pour éviter la confusion avec un connecteur audio
- Assez peu respecté

# Connectique alternative : Jack TRS 3.5mm

## Type A (MIDI standard)

Korg, Make Noise



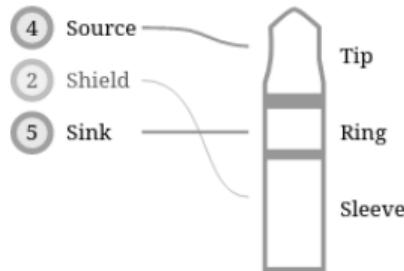
MIDI 4 (Source) > TRS RING  
 MIDI 2 (Shield) > TRS SLEEVE  
 MIDI 5 (Sink) > TRS TIP

Used by manufacturers like Korg, Make Noise, Akai, and IK Multimedia (iRig MIDI 1). In 2018, it became the [MIDI standard](#)

The first known use of MIDI over TRS (the Line 6 MIDI Mobilizer in 2010) connected this way.

## Type B

Arturia, Novation, 1010music



MIDI 4 (Source) > TRS TIP  
 MIDI 2 (Shield) > TRS SLEEVE  
 MIDI 5 (Sink) > TRS RING

Used by manufacturers like Arturia (BeatStep Pro), Novation (Launchpad Pro), and 1010music.

## ■ Deux standards



# Le protocole Les messages MIDI

# Messages MIDI

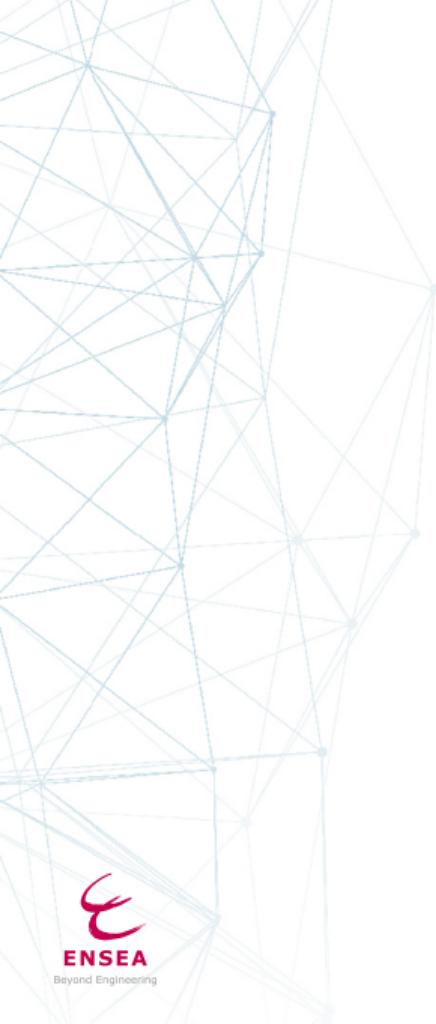
- 16 canaux

Message	Status	Data 1	Data 2
Note OFF	1000 CCCC	Note number	Velocity
Note ON	1001 CCCC	Note number	Velocity
Polyphonic Aftertouch	1010 CCCC	Note number	Pressure
Control Change (CC)	1011 CCCC	Control number (0-119)	Value (0-127)
Program Change	1100 CCCC	Program number	N/A
Channel Pressure	1101 CCCC	Pressure	N/A
Pitch Bend	1110 CCCC	LSB	MSB
System	1111 XXXX	Variable	Variable

- Ex: Jouer un C3 (60, 0x3C) sur le channel 2
  - 0x92 0x3C 0x7F
  - 0x82 0x3C 0x7F

# Notes number

Octave	Note Names / Numbers (C3=60)											
	C	C#	D	D#	E	F	F#	G	G#	A	A#	B
-2	0	1	2	3	4	5	6	7	8	9	10	11
-1	12	13	14	15	16	17	18	19	20	21	22	23
0	24	25	26	27	28	29	30	31	32	33	34	35
1	36	37	38	39	40	41	42	43	44	45	46	47
2	48	49	50	51	52	53	54	55	56	57	58	59
3	60	61	62	63	64	65	66	67	68	69	70	71
4	72	73	74	75	76	77	78	79	80	81	82	83
5	84	85	86	87	88	89	90	91	92	93	94	95
6	96	97	98	99	100	101	102	103	104	105	106	107
7	108	109	110	111	112	113	114	115	116	117	118	119
8	120	121	122	123	124	125	126	127	—	—	—	—



# USB MIDI

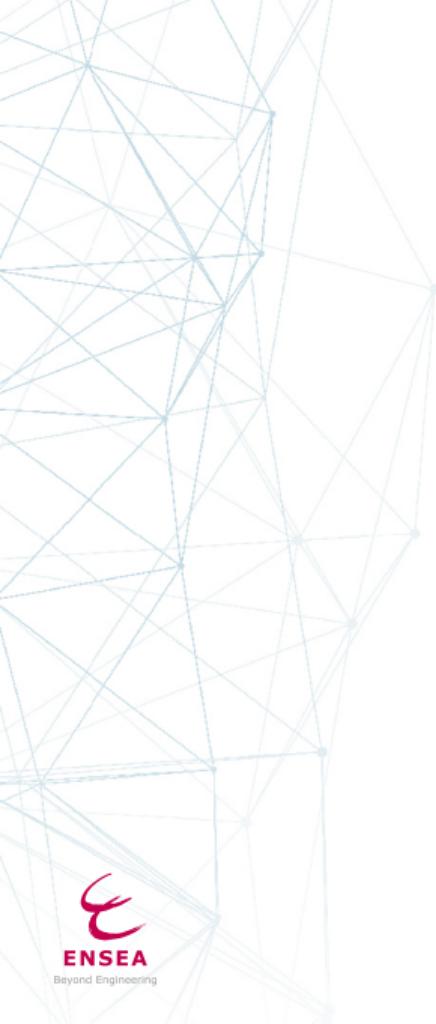
## USB Class Compliant

# Classes USB

- Périphériques USB regroupés en types ou *classes*
  - Drivers intégrés dans la plupart des OS
- Exemple de classes :
  - Audio Device Class
  - Communication Device Class (CDC)
  - Download Firmware Update Class (DFU)
  - Human Interface Device Class (HID)
  - Mass Storage Class
  - etc...

# Classe USB et driver

- Périphériques USB définis par deux descripteurs :
  - Vendor ID
  - Product ID
- Visualisable avec la commande `lsusb`
- Chargement de drivers `sudo dmesg`



# USB MIDI

## USB MIDI Class

# MIDIStreaming Interface Class

- "Hundreds of time faster"
- Sous-classe de Audio Interface Class
- Virtual cable support → Plus de limite de 16 canaux
- Event packet USB-MIDI sur 32-bit
- Octet supplémentaire par rapport à la norme MIDI

# Trame USBMidi



- Cable Number sur 4 bits
- Code Index Number sur 4 bits, même valeur que les 4 premiers bits du message midi
- Ex: Jouer un C3 (60, 0x3C) sur le channel 2, cable 1
  - 0x19 0x92 0x3C 0x7F
  - 0x19 0x82 0x3C 0x7F



# TD Machine

Mise en place d'un device USBMidi sur carte STM32F746 Discovery

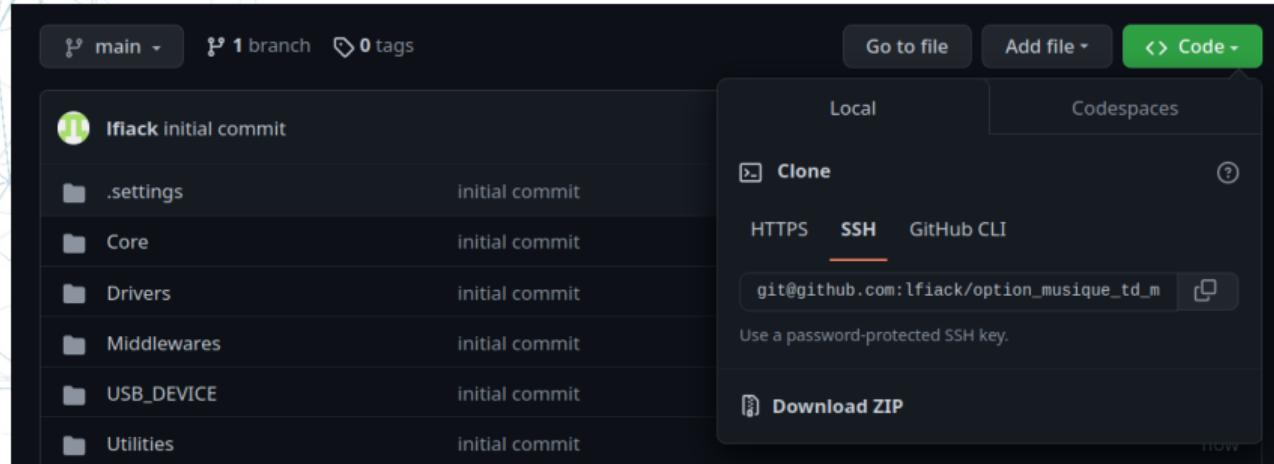
# Prérequis

- STM32CubeIDE 1.10.1
  - <https://www.st.com/en/development-tools/stm32cubeide.html#get-software>
- Ou STM32CubeMX 6.6.1 et VSCode
  - <https://www.st.com/en/development-tools/stm32cubemx.html#get-software>
- Renoise



# Récupération du projet

- [https://github.com/lfiack/option\\_musique\\_td\\_midi](https://github.com/lfiack/option_musique_td_midi)
- Clonez ou téléchargez le projet dans un dossier sans espace ni accent  
`git clone https://github.com/lfiack/option_musique_td_midi.git`



# Importer le projet dans STM32CubeIDE

- Lancez STM32CubeIDE
- File > Import
- General > Existing project into Workspace
- Select root directory > Browse...
- Choisir le dossier option\_musique\_td\_midi
- Cliquez sur Finish

# Tester le projet (1/3)

- Compilez le projet
- Programmez la carte
- Naviguez dans les différents onglets
  
- Sur le PC, ouvrez un terminal
- Lancez la commande `lsusb`
- Quelles sont les lignes correspondant à la carte?

# Tester le projet (2/3)

- Dans un terminal, lancez minicom  
`minicom -D /dev/ttyACM0`
  - Sous Windows, vous pouvez utiliser TeraTerm, Putty ou autre, débrouillez-vous...
- Redémarrez la carte (bouton poussoir noir), un prompt devrait apparaître dans minicom
- Tapez 'h' puis <Entrée>
- Testez les différentes fonctions

# Tester le projet (3/3)

- Lancez renoise
- Cliquez sur Edit > Preference, puis sur l'onglet MIDI
- Dans In device C, sélectionnez MIDI Device MIDI 1
- Fermez la fenêtre Preference
- Dans renoise, allez dans l'onglet MIDI
- Dans Input Device et dans Output Device, sélectionnez MIDI Device MIDI 1
- Appuyez sur une touche du clavier virtuel
- Quelque chose devrait apparaître dans minicom
- Si le prompt disparaît, vous pouvez le faire réapparaître en appuyant sur <Entrée>

# Analyse du code

- Vous pouvez faire Ctrl+Clic sur une fonction (ou appuyer sur F3)
- Quelles sont les fonctions appelables par le shell?
  - Où sont-elles définies?
- Comment ajouter une fonction au shell?
- Quelles fonctions sont appelées quand un message est envoyé depuis renoise?
- Quelle est la fonction appelée lorsque l'on bouge un slider?

# Ajout de fonctionnalités

- Écrivez une fonction shell permettant d'envoyer une commande Note On et Note Off
  - Testez dans renoise
- Modifiez le code pour envoyer des messages Control Change lorsque l'on touche à un slider
  - Testez dans renoise

# Aller plus loin

- Quelques suggestions :

- Dans renoise, créez un rythme, ajoutez des effets (delay, reverb...), jouez avec les paramètres de ces effets
- Ajoutez des onglets pour pouvoir contrôler plus de paramètres
- Créez un clavier dans un nouvel onglet
- Développez un synthé : utilisez le CODEC audio pour sortir du son