

23-MIDI de base

introduction

Ce tutoriel couvrira les bases de l'utilisation du protocole de communication MIDI (Musical Instrument Digital Interface). Nous verrons comment recevoir les informations MIDI à partir des périphériques d'entrée, les envoyer à d'autres matériels ou logiciels et sélectionner le périphérique qui recevra les informations envoyées.

Alors que la prévalence des systèmes de musique exclusivement logiciels a réduit le besoin de dispositifs MIDI, le protocole MIDI est toujours nécessaire pour travailler avec des synthétiseurs et des échantillonneurs matériels et logiciels. Le MIDI est également de plus en plus utilisé par les artistes travaillant avec des interfaces physiques, car il fournit un protocole de communication compact et facile à utiliser pour recevoir des informations de capteurs et de produire des messages de contrôle pour les moteurs et autres dispositifs contrôlables.

Pour ouvrir le patch du didacticiel, cliquez sur le bouton vert **Ouvrir didacticiel** dans le coin supérieur droit de la fenêtre de documentation.

Les objets *midiiin* et *midiiout*

Le patch du didacticiel contient un certain nombre de petits patches spécifiques au MIDI. Le patch le plus à gauche modifie une boîte de *nombre* à chaque fois qu'une information est reçue d'un périphérique d'entrée MIDI (via l'objet *midiiin*), puis envoie cette information à un autre périphérique MIDI (via l'objet *midiiout*). Cependant, vous devez sélectionner le périphérique qui sera utilisé pour l'entrée et la sortie MIDI. Une façon simple de sélectionner un dispositif MIDI consiste à double-cliquer sur l'objet MIDI que vous souhaitez assigner. En double-cliquant sur l'objet *midiiin* vous obtenez un menu des entrées MIDI disponibles, tandis qu'en double-cliquant sur l'objet *midiiout* pour afficher un menu de sorties MIDI.

En plus des périphériques MIDI physiques qui sont connectés à votre ordinateur, certains ports MIDI virtuels sont également visibles, en fonction de votre système d'exploitation et de sa configuration. L'un des ports virtuels les plus utiles est la sortie du synthétiseur intégrée - elle vous permet de générer des sons à partir de messages MIDI sans avoir à connecter un synthétiseur physique ou un échantillonneur physique.

Si vous sélectionnez un clavier d'entrée MIDI pour l'objet *midiiin* et le synthétiseur intégré pour la sortie, vous disposez d'une application MIDI *thru* simple avec très peu de patching. Les objets *midiiin* et *midiiout* attendent des messages MIDI bruts non formatés. Si vous acheminez la sortie de l'objet *midiiin* vers un objet *print*, vous verrez que vous disposez d'un flux de nombres en série qui peut être difficile à interpréter. Max contient un certain nombre d'objets qui nous permettent de mieux contrôler l'utilisation des données MIDI dans notre programme en sélectionnant les types d'événements MIDI (notes, contrôleurs continus, etc.) avec lesquels nous souhaitons travailler.

Tout sur les notes: les objets *notein* et *noteout*

Les objets *notein* et *noteout* sont un exemple d'objets MIDI *spécifiques aux messages*. Ils acceptent l'entrée d'un flux MIDI et ignorent tous les types de message autres que les messages de *note*, ceux qui sont généralement provoqués par la frappe des touches d'un clavier MIDI ou les pads d'un trigger de batterie. Comme pour les objets *midiiin* et *midiiout*, nous pouvons sélectionner le port MIDI en double-cliquant sur l'objet.

Le deuxième patch montre un affichage de base des notes; nous pouvons sélectionner un port MIDI, puis visualiser les notes reçues sur ce port. L'objet *notein* affiche trois informations pour chaque message de note entrant : le numéro (ou hauteur) de la note, la vélocité de la note et le canal MIDI sur lequel elle a été transmise. Jouez sur un clavier MIDI ou un autre contrôleur qui crée des messages de note MIDI, et voyez comment les informations de note sont affichées par les boîtes de *nombres*.

Vous remarquerez qu'il n'y a pas d'affichage «on ou off» pour différencier les touches jouées (une «note activée») et les touches relâchées (une «note-off») - au contraire, un message de note-off est affiché avec une vélocité de zéro (**0**). Il s'agit d'une convention MIDI courante utilisée par de nombreux objets Max comme moyen privilégié d'afficher un message de note-off. Nous avons également utilisé une boîte de *nombres* dont l'attribut **Format d'affichage** est réglé sur MIDI. Cela nous donne une indication facile à lire quant à la hauteur de la note jouée.

Le patch suivant est un exemple simple du processus inverse: générer des messages de note qui seront jouées par un dispositif MIDI. L'objet *noteout* s'attend à ce que les nombres des *hauteur*, de *vélocité* et de *canal* soient reçus dans ses entrées gauche, centrale et droite, respectivement. L'entrée de gauche (**pitch**) est *chaude* alors que les autres sont *froides*. Dans ce cas, nous utilisons la capacité de Max à décoder une liste de messages comme entrée pour toutes les entrées d'un objet. Ainsi, la boîte de *message* étiquetée **64 100 0** est traitée comme trois messages distincts adressés à chaque entrée: il s'agit des canaux «canal 0», «vélocité 100» et «note 64». Cela enverra un message de note-on pour la hauteur 64 (E3) à une vélocité de 100 vers le canal 0. La deuxième boîte de *message* est presque identique, mais utilise une vélocité de 0, qui générera un message note-off. Double-cliquez sur l'objet *noteout* et sélectionnez un périphérique de sortie MIDI valide. Cliquez ensuite sur la première boîte de message. Vous devriez entendre le périphérique émettre la note. Elle devrait être maintenue jusqu'à ce que vous cliquiez sur la deuxième boîte de message, ce qui l'éteindra. Il s'agit d'une structure d'événement de note MIDI de base, qui peut être utilisée comme base pour générer des messages MIDI qui jouent de la musique.

Contrôleurs et sélection de ports: *ctlin* et *ctlout* et *midinfo*

Le patch le plus à droite est une version simple d'une application qui remappe les données du contrôleur MIDI: elle change toute valeur de contrôle de molette de modulation (par convention, le contrôleur MIDI **1**) en une valeur de contrôle panoramique (contrôleur **10**). Dans ce cas, les objets *ctlin* et *ctlout* sont utilisés pour recevoir et envoyer des messages de contrôleur continu MIDI. Ce sont des objets MIDI spécifiques aux messages dédiés aux valeurs de contrôleur MIDI, souvent envoyés par les boîtes de faders MIDI et contrôleurs rotatifs, ainsi que par des interfaces plus axées sur la musique, telles que les pédales de sustain de clavier. Comme avec les autres objets, vous pouvez double-cliquer dessus pour définir un port MIDI à utiliser.

Les valeurs entrantes sont utilisées pour configurer le système de contrôle. Le patch utilise le numéro de contrôleur entrant (la sortie centrale de *ctlin*) pour déterminer le routage d'un objet de porte graphique (*gswitch2*). Lorsque le contrôleur **1** est vu, l'objet == génère un **1**, ce qui active le commutateur pour permettre à la valeur du contrôleur (sortie de gauche) d'aller dans la boîte de *message* suivante. Cette valeur est ensuite assemblée dans un message valide pour *ctlout*, qui enverra la valeur (assignée au contrôleur MIDI **10** sur le canal **0**) à la sortie MIDI sélectionnée.

C'est certes la manière la plus difficile pour détecter les messages de contrôle 1. Si nous donnons un argument de 1 à un objet ctlin, il ne répondra qu'aux messages de contrôle 1.

Il y a un peu de programmation supplémentaire attachée à l'objet *ctlout*: un *button* est connecté à un objet *midinfo*, qui charge un objet *umenu* avec certaines données lorsqu'il reçoit un message **bang**.

Ces données sont introduites dans l'objet *ctlout*. De quoi s'agit-il ?

Parfois, vous ne voudrez pas que l'utilisateur final de votre patch doive double-cliquer sur les objets MIDI pour définir un port, ou vous voudrez peut-être vouloir définir le port sur plusieurs objets simultanément. L'objet *midiinfo* fournit de nombreuses informations sur votre configuration MIDI actuelle, que vous pouvez utiliser pour définir les informations de port sur vos objets MIDI. Dans ce cas, lorsque l'objet *midiinfo* reçoit un message **bang** dans son entrée de *gauche*, il produit les noms des ports de *sortie* MIDI disponibles actuellement définis sur votre système. Ceci est utilisé pour charger l'objet *umenu* avec votre configuration MIDI actuelle. Lorsque vous sélectionnez un élément du menu, il est envoyé à l'objet *ctlout* comme identifiant du port que vous souhaitez utiliser.

Ceci n'est qu'une des utilisations de l'objet *midiinfo*; nous verrons d'autres utilisations dans les prochains tutoriels. Si vous voulez interroger votre ordinateur pour obtenir des informations sur les ports MIDI disponibles et les contrôleurs disponibles, consultez les fichiers d'aide et les pages de référence de *midiinfo*.

Création d'un moniteur MIDI de base

Afin de s'exercer aux outils de messagerie MIDI, créons une application de surveillance MIDI de base. Nous pouvons commencer par créer une version de la combinaison *midiinfo* / *umenu* qui affichera les entrées MIDI disponibles. Afin de voir les entrées, nous devons envoyer un message **bang** dans l'entrée *droite* de l'objet *midiinfo*. Vous pouvez le faire avec un objet *button* ou utiliser un objet *loadbang* pour que le menu soit chargé à chaque fois que le patch est chargé.

Afin de suivre les informations sur les notes et les contrôleurs, nous pouvons envoyer la sortie de l'*umenu* à un objet *notein* et *ctlin*. Ceci configure les deux objets pour qu'ils reçoivent des informations MIDI à partir du même port MIDI. Maintenant, connectez les boîtes de *nombres* à chacune des sorties des objets *notein* et *ctlin*. Nous pouvons probablement faire en sorte que les messages «channel» soient partagés par une seule boîte de *nombres*, car nous n'aurons ni notes ni contrôleurs générés en même temps. Enfin, en changeant la boîte de *nombres* de note pour utiliser un **format d'affichage** MIDI nous permettra de voir facilement la touche de la note enfoncée.

Bien qu'il s'agisse d'une application de moniteur MIDI assez simple, c'est le type d'application qui peut vous aider à déboguer une configuration MIDI complexe. Vous pouvez étendre ce patch en utilisant d'autres objets spécifiques au message MIDI, tels que *pgmin* (changements de programme MIDI) et *bendin* (messages de pitch bend MIDI), afin de voir encore plus le flux de données MIDI reçu.

Résumé

Le contenu de base d'un flux MIDI entrant peut être récupéré à l'aide de l'objet *midiin* et un flux MIDI peut être envoyé à l'aide de l'objet *midiout*. Cependant, dans la plupart des cas, il est plus facile de traiter les types de message individuels en utilisant leurs objets spécifiques. Les objets *notein* / *noteout* et *ctlin* / *ctlout* fournissent un moyen facile de traiter les messages sans avoir à décoder l'entrée MIDI brute ou à créer une sortie MIDI brute.

L'objet *midiinfo* est un outil puissant pour examiner l'état actuel de votre configuration MIDI. L'utiliser pour charger un menu de ports MIDI disponibles peut aider à rendre votre patch plus facile à utiliser et c'est aussi la meilleure façon de régler un certain nombre d'objets simultanément.