

## Computer Music 2

### - 第4回 -

担当：小林良穂

Computer Music 2

## 本日の内容

- Max for Live デバイスの作成
- MIDIとは
- MIDIの中身
- シーケンサー

Computer Music 2

## Max for Live

Computer Music 2

## Max for Liveとは

- Ableton LiveのデバイスをMaxで作成／編集できる
- ビルトインデバイスや市販のプラグインの組み合わせでは不可能な柔軟な制作環境が実現

Computer Music 2

## Max for Liveデバイスの使用

- “Max for Live”カテゴリーの中の  
Max Instrument  
Max MIDI Effect  
Max Audio Effect  
から選択
- Liveデバイスと同様に使用可能

Computer Music 2

## Max for Liveデバイスの編集

- デバイス右上の編集ボタンを押すとMaxが起動し、**Patcher Window** が開く




Computer Music 2

## デバイスの作成

- 新しいMax for Liveデバイスを作成するには
- “Max Instrument”, “Max MIDI Effect”, “Max Audio Effect”  
のいずれかを追加  
※既存デバイスは選ばない
- MIDI信号の入出力は **midlin / midiout**  
オーディオ信号の入出力は **plugin~ / pluginout~**  
の各オブジェクトに対応している

Computer Music 2

## 自作デバイスの使用

- 作成したデバイスを使用するには
  1. Maxのパッチを保存  
ファイル形式は **Ableton Max Patch**  
拡張子は **.amxd**
  2. Patcher 右下の  ボタンをオフにする

Computer Music 2

## デバイス表示の問題点1

- Liveデバイスのデザインには統一感がある
- Maxのオブジェクトを使うと統一感がなくなる  
→ **Live用に用意されたオブジェクトを利用**



Computer Music 2

## デバイス表示の問題点2

- Liveのデバイスチェーンは上下の表示範囲に限られる
- Maxのパッチをこの範囲に収めるのは困難
- 編集時と実行時のオブジェクトの配置を変えて解決  
→ **プレゼンテーションモードを利用**

Computer Music 2

## MIDIとは

## MIDIとは

- Musical Instrument Digital Interfaceの略
  - 電子楽器の**演奏情報**をやり取りするための規格
- ※**演奏情報**：  
楽器演奏時の「鍵盤を弾く」「弦を弾く」  
「ペダルを踏む」といった情報  
「音」そのもの(音響信号)は伝えない

Computer Music 2

Computer Music 2

## MIDIの歴史

- 1980年代初頭に日本の楽器メーカー(Roland, YAMAHA等)を中心として策定
- 5pin DINケーブルで簡単に接続
- デジタル技術を利用して様々な種類の情報を送れる

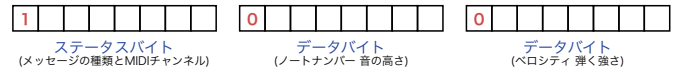


YAMAHA DX7 (1983年発売)  
FM音源によるフルデジタルシンセ  
誕生して間もないMIDIを採用

Computer Music 2

## MIDIの中身

- デジタル信号(0と1で構成)
- 8bit (= 1byte)を1つの単位として送受信
- 最初にステータスバイト、続けてデータバイトを送る
- 例：ノートオンメッセージの場合  
1つのステータスバイトに続けて2つのデータバイト



- 通信速度は約31.25kbps

Computer Music 2

## MIDIの中身

- 基本的に、各バイトの先頭の1bitはステータスバイト / データバイトの区別に使われる
- 残りの7bitで値を送る  
7bit → 2<sup>7</sup> (128通り)
- MIDIでは多くの値が 0~127 で表される

Computer Music 2

## ノートオン/オフ

- あらゆるMIDIメッセージを理解し制御するのは大変  
→ まずはノートオン/オフだけ扱う
- ノートオンメッセージ
  - 音を出す命令
  - ノートナンバーとベロシティで構成される
  - ベロシティ0のノートオンはノートオフと見なされる
- ノートオフメッセージ
  - 音を止める命令
  - ノートナンバーとベロシティで構成される

Computer Music 2

## ノートオンの中身

- ステータスバイトは16進数で9n(nはチャンネル番号0~15)  
10進数では144(ch1)~159(ch16)
- |   |   |   |   |  |  |  |  |
|---|---|---|---|--|--|--|--|
| 1 | 0 | 0 | 1 |  |  |  |  |
| g |   | n |   |  |  |  |  |
- 第1データバイトはノートナンバー (音の高さ)  
(16進数で00~7F、10進数で0~127)
    - 中央ド(Middle C; C4)が60、半音毎に1ずつ上下する
  - 第2データバイトはベロシティ (音の強さ)  
(16進数で00~7F、10進数で0~127)
    - 1が最弱、127が最強
    - 0はノートオフ(音を止める)を意味する

Computer Music 2

## 補足：リストの扱い

- リストとは
  - 複数の値をまとめたメッセージ
  - packやpakで作成
  - unpackで分解

Computer Music 2

## シーケンサー

Computer Music 2

## シーケンサーとは

- ・プログラムに基づいて自動演奏をする装置
- ・現在はコンピュータ上のソフトウェアが主流



Computer Music 2

## notein / noteout

- ・ノートオン/オフメッセージのみを扱いたい場合は **notein / noteout** オブジェクトが便利

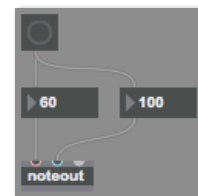
**notein :**

- ・第1 outlet : **ノートナンバー** (0~127)
- ・第2 outlet : **ペロシティ** (0~127; 0は**ノートオフ**)
- ・第3 outlet : **MIDIチャンネル** (1~16)

Computer Music 2

## step1. 音を出す

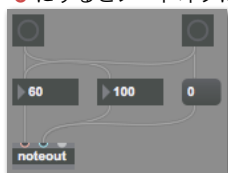
- ・noteoutオブジェクトにノートナンバー(音の高さ)とペロシティ(音の強さ)を送る



Computer Music 2

## step2. 音を止める

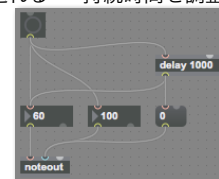
- ・**ノートオン**メッセージだけだと音が鳴り続ける
- ・**ノートオフ**を送ることで音が止められる
- ・ペロシティが **0** にするとノートオフになる



Computer Music 2

## step3. オン/オフの連動

- ・**delay**オブジェクトを使えばノートオンの一定時間後にノートオフを送れる → 持続時間を調整できる



- ・**makenote**オブジェクトを使えばさらに細かく調整が可能

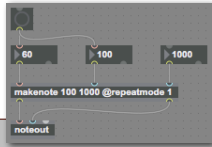
Computer Music 2

## makenote

- ノートオンを受けてから指定時間後にノートオフを送ることで**音の長さ**を決められる

**makenote :**

- 第1 inlet : **ノートナンバー**
- 第2 inlet : **ベロシティ**
- 第3 inlet : **音の長さ** (単位 : ms)

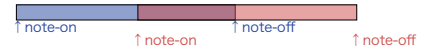


Computer Music 2

## repeatmode

- makenoteには@repeatmodeというアトリビュートが用意されている

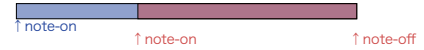
**@repeatmodeなし :**



**@repeatmode 1 :**



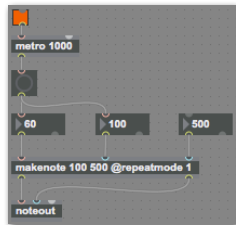
**@repeatmode 2 :**



Computer Music 2

## step4. 定期的に音を鳴らす

- 発音のきっかけに**metro**オブジェクトを使う



Computer Music 2

## step5. 曲のテンポに合わせる

- Max for Liveのデバイスを曲のテンポに合わせるには
  - metroやdelayの時間の部分に**ノートバリュー**を入れる

	通常	付点(3/2倍)	3連譜(2/3倍)
全音符	1n	1nd	1nt
2分音符	2n	2nd	2nt
4分音符	4n	4nd	4nt
8分音符	8n	8nd	8nt
16分音符	16n	16nd	16nt

Computer Music 2

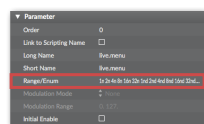
## ノートバリュー

- ノートバリューに関する注意点
- あくまで西洋音楽における**音価**(譜面上の時間)を表す
- そのため**1.5n**や**3n**といった指定は無効
- 1n, 2n, 4n, 8n, .....とそれぞれの付点(nd)、3連譜(nt)が指定可能
- 数値として扱えないため、変更には**live.menu**が有効

4n

**live.menu**オブジェクト

第2outletからメニュー内容をそのまま出力

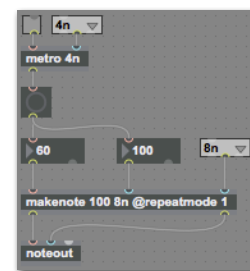


**live.menu**のinspector

Range/Enumのところに項目名を登録

Computer Music 2

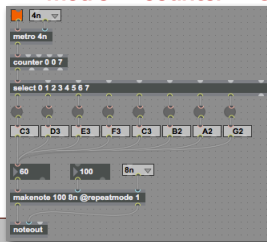
## ノートバリュー



Computer Music 2

## step6. メロディを作る

- 設定したピッチを順番に鳴らせばメロディが作れる
- metro → counter → select の組み合わせ

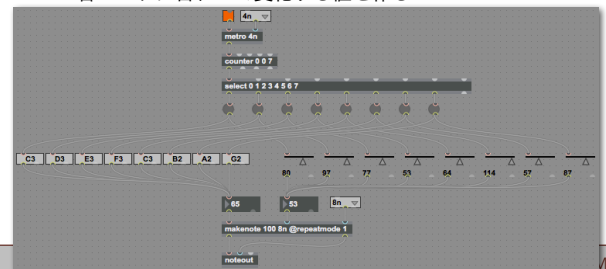


live.numboxの表示をMIDIにした例

Computer Music 2

## step7. ベロシティの調整

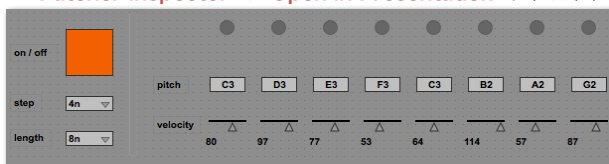
- 各ノートに合わせて変化する値を作る



Music 2

## step8. 見た目の整理調整

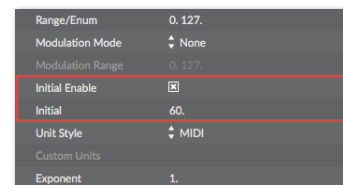
- プレゼンテーションモードを利用
- 必要なオブジェクトだけをプレゼンテーションモードに含めてデバイスの指定範囲内に配置
- Patcher Inspector で “Open in Presentation” にチェック



Computer Music 2

## step9. 初期値の設定

- liveオブジェクトは初期値が保存できる
- Inspector で Initial Enable にチェック
- Initial に値を入れる



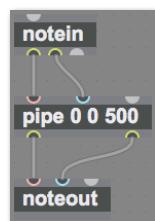
Computer Music 2

## ノートメッセージの遅延

- bangを遅延させるにはdelayを使う
- 値やリストを遅延させるにはpipeを使う

pipe :

- 値、リストを遅延させる
- アーギュメント増やすことで複数の値を入力に使える
- 最後のアーギュメントは遅延時間



Computer Music 2

## 第4回課題

### シーケンサーの作成

- MIDIトラックを1つだけ残す
- 好きなドラムの音を選ぶ
- Max for Liveでシーケンサーを作成
- オートメーションは使わないで、値の変化は全てMaxでコントロールする
- オーディオエフェクトは使わない

Computer Music 2

## 第4回課題

### <提出方法>

Liveのファイルを“**cm2-04 学籍番号.als**”で保存  
([すべてを集めて保存]を実行)  
プロジェクトフォルダを**ZIPで圧縮**して提出

### <提出期限>

10月31日(火) 23:59

### <質問>

[ryoho@sfc.keio.ac.jp](mailto:ryoho@sfc.keio.ac.jp)