

# Hardware

MIDI over USB

# Seriële communicatie

- Binaire communicatie, byte voor byte verstuurd
- 1 byte = 8 bits = nummer van 0-255 (want  $2^8 = 256$ )

01110111 01100001 01110010 01100100

\            /

byte

# Seriële communicatie: tekst

- Tekst → ASCII → Bytes om te versturen
  - tekst: hoi
  - ASCII: 104 111 105
  - bits: 01101000 01101111 01101001

# MIDI berichten

- Note On
- Note Off
- Control Change (CC)
- Pitch Bend
- After Touch
- Clock Start / Stop / Tick
- Program Change
- ...

# MIDI berichten

- MIDI heeft 16 kanalen
  - Voor bijv. 16 verschillende instrumenten/partijen
- Elk kanaal heeft:
  - 128 parameters (control change)
  - 128 toonhoogtes (note on/off)

# Seriële communicatie: MIDI

- Elk MIDI bericht is 1-3 bytes lang:
  - 1 status byte: wat voor bericht
  - 0-2 data bytes: inhoud van bericht
- 1e bit van elke byte onderscheid status van data byte
- 7 bits voor waarden:  $2^7 = 128$  = waarden van 0 t/m 127

# Seriële communicatie: MIDI

| status   | data     | data     |
|----------|----------|----------|
|          |          |          |
| 10010011 | 00111100 | 01100100 |

# Seriële communicatie: MIDI

note on

| channel 4 ( $2^4 = 16$  kanalen)

| |

10010011 00111100 01100100

# Seriële communicatie: MIDI

note on

| channel 4 ( $2^4 = 16$  kanalen)

| |

10010011 00111100 01100100

|

|

|

|

( $2^7 = 128$  waarden: 0-127)

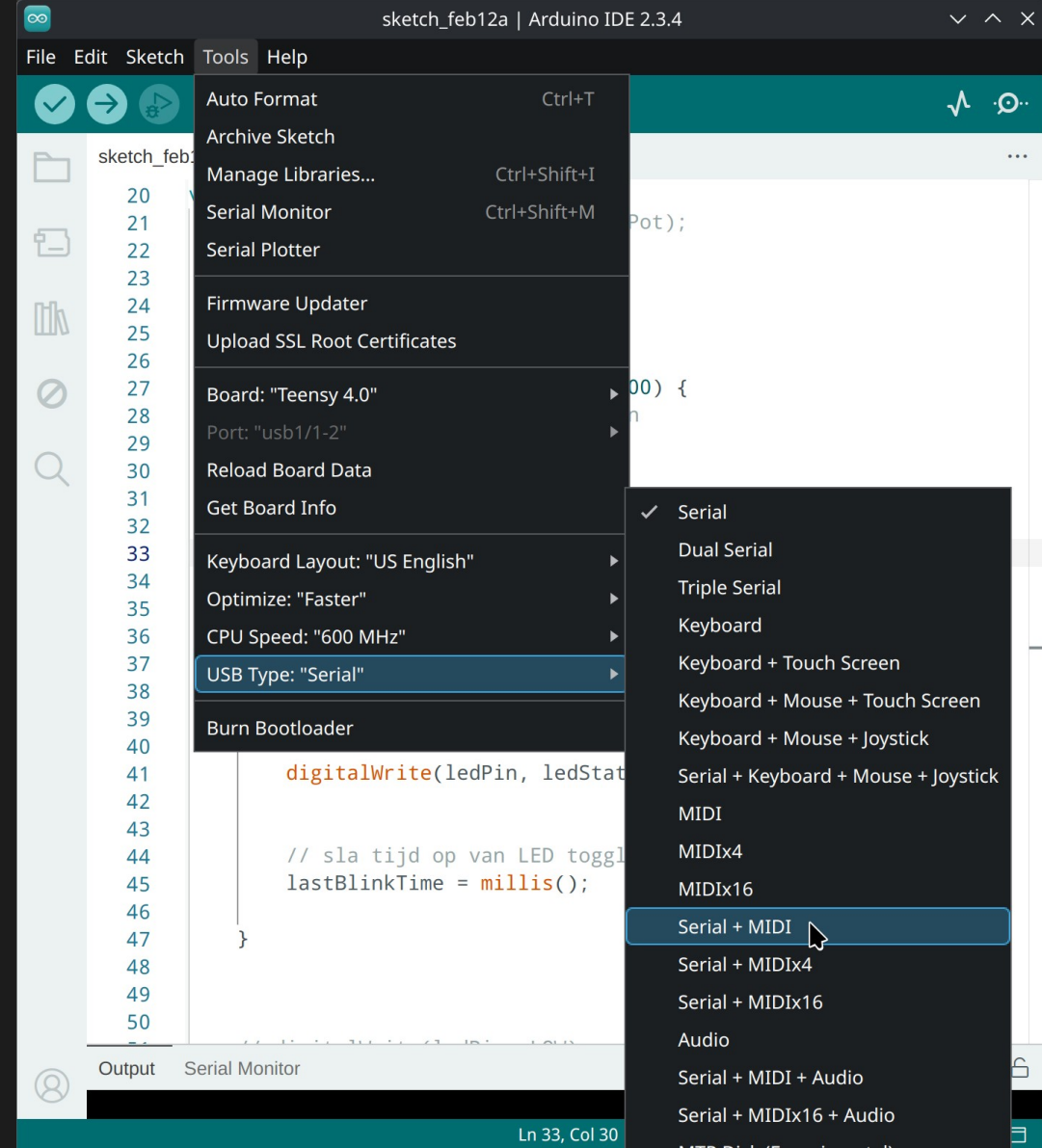
note 60

|

velocity 100

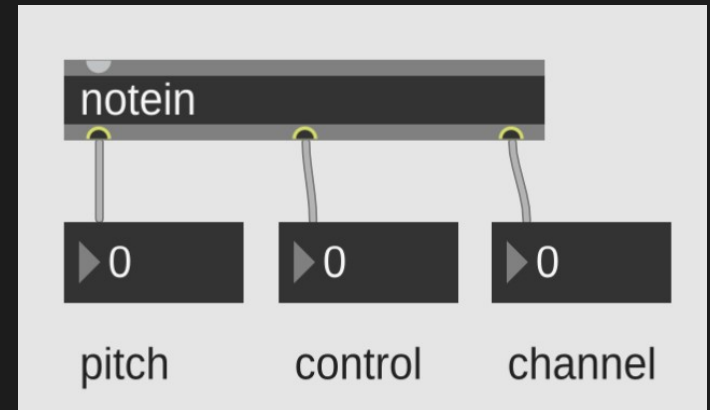
# Teensy: MIDI

- Zorgen dat USB MIDI kan sturen/ontvangen
  - **Tools > USB type > “Serial + MIDI”**
- Handige functies voor het sturen van MIDI berichten



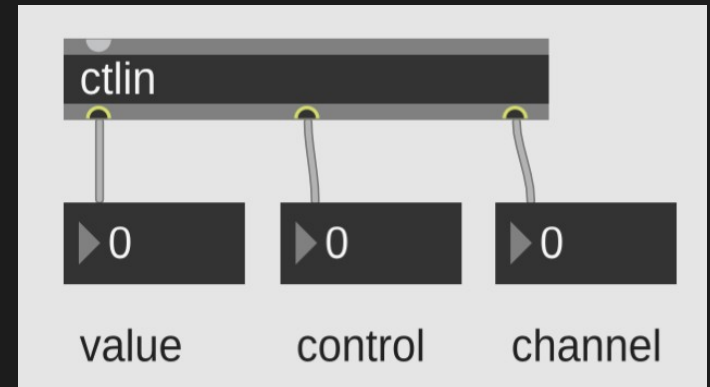
# Teensy: MIDI verzenden

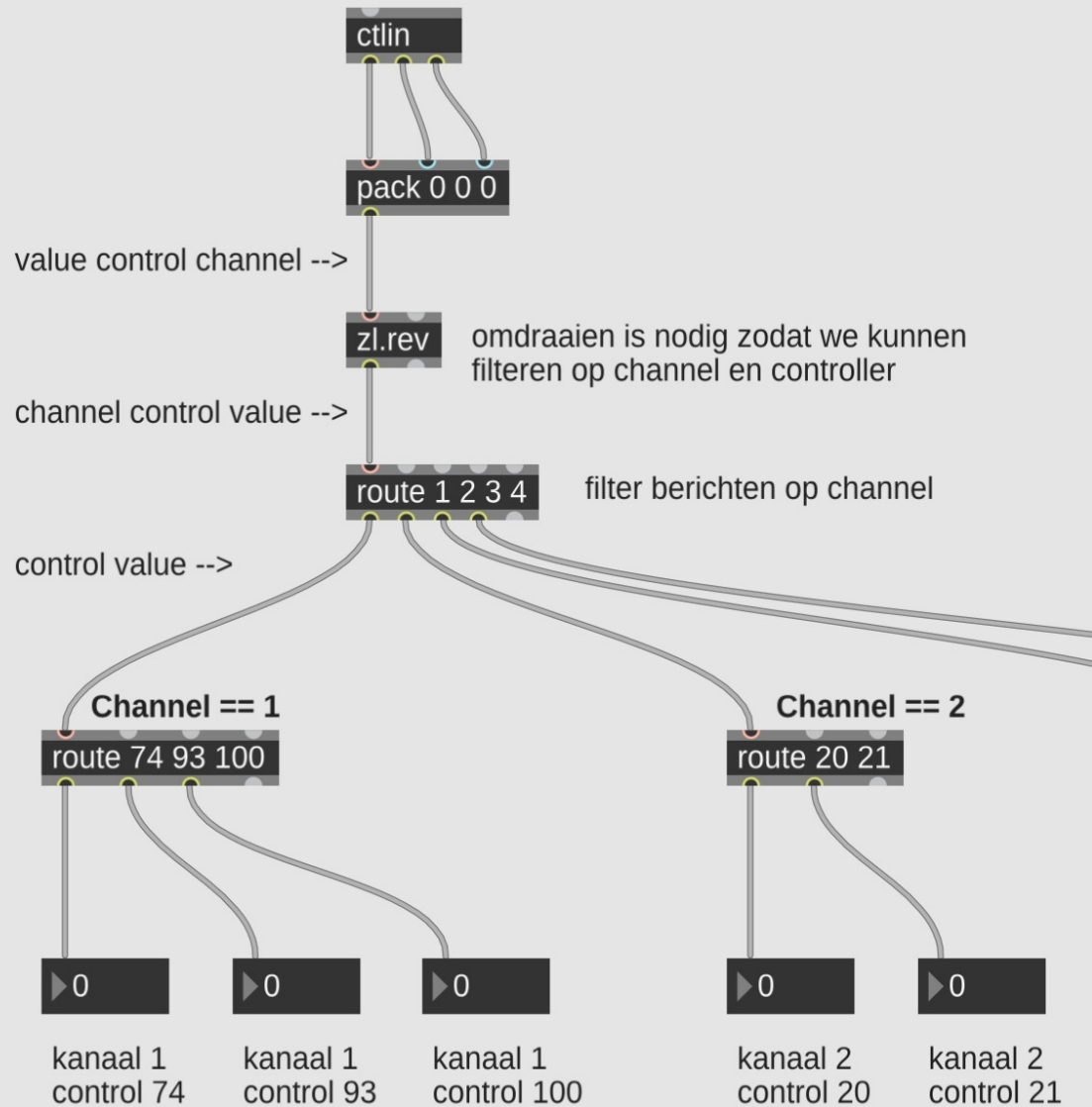
- `usbMIDI.sendControlChange(control, val, chan);`
  - control = welke control is veranderd?
  - val = nieuwe waarde van de control
  - chan = kanaal van de control
- Ontvangen in Max met **[ctlin]**



# Teensy: MIDI verzenden

- `usbMIDI.sendNoteOn(pitch, velocity, chan);`
- `usbMIDI.sendNoteOff(pitch, velocity, chan);`
  - pitch = toonhoogte van de noot
  - velocity = hoe hard is de toets ingedrukt?
  - chan = kanaal van de noot
- Ontvangen in Max met **[notein]**





# Teensy: MIDI ontvangen

- Functie(s) aanmaken waarin je iets met binnenkomende bericht doet
- In de `setup( )` aangeven welke functie(s) dat zijn
- In de `loop( )` checken of er al weer MIDI ontvangen is
- Sturen vanuit Max met `[ctlout]` of `[noteout]`

# Teensy: MIDI ontvangen

```
void setup() {  
    usbMIDI.setHandleControlChange(controlHandler);  
}  
  
void loop() {  
    usbMIDI.read();  
}  
  
void controlHandler(byte channel, byte control, byte value) {  
    if (channel == 1 && control == 74) {  
        digitalWrite(13, HIGH);  
    }  
}
```

# Demonstratie

- Potmeter uitlezen
- Waarde via MIDI versturen