

WERKING EN GESCHIEDENIS VAN MIDI¹

INLEIDING

Robert Moog, één van de grondleggers van de elektronische muziekinstrumenten, beschrijft het werken met electronica en muziek in het 'analoge' tijdperk:

“Normaalgesproken spendeerde je uren om het geluid te krijgen dat je nastreefde maar dat je niet kon opslaan. Je moest al je opnames met dat specifieke geluid op datzelfde moment doen omdat je anders, wanneer je de volgende dag terug zou komen, hetzelfde geluid nooit weer zou terug kunnen krijgen. (...) Toen kwam er verlossing (...).”

De volgende uiteenzetting behandelt de geschiedenis en de werking van de *Musical Instrument Digital Interface (MIDI)*. Aan de orde zullen komen welke verbeteringen MIDI in historisch opzicht bracht en op welke manier MIDI toegepast kan worden. Door het verkregen inzicht in de werking van MIDI wordt uiteindelijk een conclusie getrokken over de plaats van MIDI in de muziekinstrumentenkunde.

WAT IS MIDI?

Simpel gezegd is MIDI: een conventie van codes waardoor elektronische muziekinstrumenten met elkaar kunnen communiceren. Het bestaat uit:

- a. Een verzameling *commando's*, ook wel de MIDI 'taal' genoemd
- b. Details over hoe deze *commando's* worden verzonden en ontvangen door MIDI instrumenten

De MIDI specificatie is een canon. Deze is terug te vinden in het historische document *MIDI 1.0 Detailed Specification* (1981), kortweg *spec* genoemd.² MIDI was oorspronkelijk bedoeld voor muziekuitsvoeringen en compositie maar begeeft zich nu ook in aanverwante gebieden zoals *audio editing* en - productie, podiumverlichting en andere aspecten van *real-time* uitvoeringen. De zogenaamde MIDI taal omvat, zoals gezegd, een reeks *commando's*, die een auditieve gebeurtenis beschrijven. MIDI op zich is nog geen muziek en ook nog geen geluid. MIDI beschrijft alleen de voorwaarden die nodig zijn om geluid te produceren. Paul Lehrman van de Universiteit van Massachusetts verwoordt het als volgt:

“MIDI is geen muziek in de zin dat het hetzelfde is als een opname van bijvoorbeeld een Engelse hoorn. Niettemin kunnen MIDI *commando's* de uitvoering van de speler op een dergelijke manier beschrijven dat een synthesizer een accurate uitvoering kan reconstrueren.”³

Hoewel MIDI ook wel de MIDI standaard wordt genoemd, is het niet een door de staat geautoriseerde standaard. De voorwaarden zoals die in de spec worden genoemd zijn niet dwingend en worden vrijwillig door producenten van MIDI apparatuur (grotendeels) nagevolgd. Het gevolg hiervan is dat veel apparatuur *MIDI-compatible* is en dus in netwerken gebruikt kan worden.

HOE IS MIDI ONTSTAAN?

Electronische muziek is een fenomeen dat zo rond het midden van de twintigste eeuw opgeld heeft gedaan. Het is op kleine schaal begonnen en door de opkomst van massaproductie en micro electronica uitgegroeid

¹ Dit essay werd in principe geschreven voor de cursus 'muziekinstrumenten' aan de Universiteit Utrecht. Doel was om op een begrijpelijke manier de geschiedenis en werking van MIDI voor de geïnteresseerde digibeeft inzichtelijk te maken.

² Gepubliceerd door de *International MIDI Association*. Deze werd eerst de *Universal Synthesizer Interface* genoemd. De spec is de definitieve naam geworden.

³ Lehrman & Tully (1993): 11

tot een wereldwijd fenomeen. Voorbeelden van de eerste elektronische instrumenten zijn de Theremin⁴ en de ondes Martenot, die door Olivier Messiaen gebruikt werd in zijn beroemde Turanglila Symfonie (1978).

De eerste studio's voor elektronische muziek werden gebouwd met componenten die niet specifiek voor de productie van muziek gemaakt werden, maar voor bijvoorbeeld audiologie laboratoria, telefonie of radiotest apparatuur zoals frequentie oscillatoren. Deze niet voor muziek bedoelde apparatuur was het enige voor handen zijnde materiaal voor de eerste elektronische composities zoals bijvoorbeeld *Klangfiguren* (1955/1956) van Karlheinz Stockhausen.

In de jaren zestig werden de specifiek voor muziekvoortbrenging bedoelde synthesizers geproduceerd. Deze synthesizers konden met elkaar verbonden worden met behulp van zogenaamde *patch cords*. Het verbinden van een aantal modules om een speciaal geluid te kunnen creëren werd aangegeven met het woord *patch*. Dit woord bestaat in de huidige muziekwereld ook wel als *voice*. Het hiervoor genoemde *patch cord* 'vervoerde' voltages. De modules communiceerden met elkaar door te reageren op verschillen in de voltages.

De werking van deze analoge apparatuur kan worden geïllustreerd door middel van een voltage gecontroleerde oscillator (of *Voltage Controlled Oscillator, VCO*). In plaats van een gewone draaiknop voor het bepalen van de voort te brengen frequentie, wordt de frequentie bepaald door een inkomend signaal. De producent van de oscillator bepaald vervolgens op welke manier de uitkomende frequentie (*output*) bepaald moet worden door het inkomend voltage (*input*). Dit kan bijvoorbeeld door de *1-volt-per-octave* verhouding. Binnen deze formule wordt een inkomend voltage van 1 volt een uitkomende frequentie van 100 Hz, twee volt resulteert in 200 Hz, drie volt in 400 Hz, vier volt 800 Hz etcetera.

Dit is slechts een eenvoudig voorbeeld van de toepassing van *voltage control*. Het principe kan gebruikt worden voor vele andere muzikale parameters dan alleen de frequentie zoals bijvoorbeeld dynamiek. Het principe van voltage-control werd erg populair onder synthesizer producenten. Het had echter zijn problemen. Ten eerste was er geen standaardisatie van formules; verschillende producenten hanteerden verschillende formules voor de wijze waarop voltages de *output* bepaalden. Zo kon de ene synthesizer met de *one-volt-per-octave* formule zijn uitgerust, terwijl de andere formule van de andere synthesizer met een verschillende formule was uitgerust. Zo waren ze dus moeilijk aaneen te schakelen. Een tweede nadeel was dat deze methode analoog werkte en dat absolute reproductie daarmee nooit was gewaarborgd. Met andere woorden: twee klanken van een zelfde *patch* op een verschillend moment, waren bijna altijd verschillend.

Het antwoord op deze nadelen kwam in de vorm van zogenaamde *triggers*. In plaats van een permanente beïnvloeding van de toon (ook wel *stream* genoemd) werd nu een binair systeem ontwikkeld dat op kort opeenvolgende momenten informatie verzond. Helaas werd hiermee het probleem van onverenigbaarheid (*incompatibility*) niet opgelost. Het oude probleem van producenten die verschillende formules gebruikten zoals in het tijdperk van de *voltage control* bleef ook hier bestaan. Op één of andere manier zou er een standaardisatie gemaakt moeten worden die ervoor zou zorgen dat alle synthesizers zonder problemen met elkaar verbonden konden worden.

⁴ Die overigens in productie is genomen door Robert (Bob) Moog (zie: www.moogmusic.com)

DE DIGITALE REVOLUTIE

In de late jaren zeventig begonnen producenten synthesizers met digitale electronica uit te rusten. Digitale synthesizers gebruiken mathematische algoritmen voor het produceren van digitale modellen van geluidsgolven. Deze mathematische constructies worden vervolgens omgezet met behulp van *digital-to-analog converters (DAC)* naar audio signalen. Verschillende modules – zoals de hiervoor genoemde oscillatoren – werden nu geïnternaliseerd in één instrument. Fysieke instrumenten waren software geworden die simpel geactiveerd konden worden. *Patches* werden louter informatie-technologische beschrijvingen van de specifieke constellatie van een *patch*. Met behulp van digitaal geheugen (of *RAM, Random Access Memory*) konden de meest complexe *patches* ook nog eens worden opgeslagen. Hiermee was exacte reproductie van bepaalde geluiden gewaarborgd. Het probleem van *incompatibility* was echter nog niet opgelost omdat producenten nog verschillende schema's hanteerden voor de wijze van transformatie van signaal. De eerste stappen voor het oplossen van dit probleem werden gezet door Dave Smith en Chet Wood van het bedrijf Sequential Circuits. De oplossing die zij aandroegen was een conventie die elektronische muziekinstrumenten volledig intercommunicatief moest maken. In oktober 1981 presenteerden zij de zogenaamde Universal Synthesizer Interface (USI). Deze conventie werd later veranderd in MIDI (*Musical Instrument Digital Interface*) en werd gepubliceerd in de reeds genoemde spec.

DE TOEPASSING VAN MIDI

MIDI werkt met bits en bytes. Bit is een afkorting voor *binary digit*. Een bit is een 0 of een 1, respectievelijk een 'aan of uit'. Deze bits zijn gegroepeerd in groepen van tien bits die tezamen een byte vormen. Deze groepen bits of bytes worden op een bepaalde manier gecombineerd en bepalen samen de inhoud van een commando of *message*. In de spec staat beschreven hoe MIDI bepaalde commando's ontcijfert en welke specifieke interpretatie bij een specifieke combinatie van bits en byte(s) hoort. De spec is dus bovenal een soort objectief referentiepunt bij de productie van elektronische muziekinstrumenten.

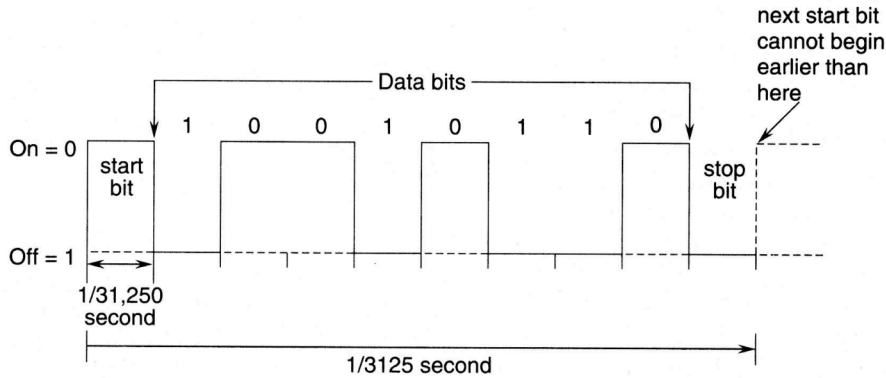
Een MIDI signaal is een puls die 31.250 maal per seconde wordt weggezonden. Omdat het MIDI signaal een digitaal signaal is, heeft deze puls twee tegengestelde momenten; een 0 en een 1. Een 0 staat voor een aan en een één voor een uit. De structuur van een MIDI byte ziet er daarmee als volgt uit: een byte begint altijd met een 0; 'aan' (de zogenaamde *start bit*). Hierop volgen acht zogenaamde *data bits* die zowel uit enen als nullen kan bestaan. Deze wordt weer opgevolgd door een *stopbit*, de 1 (uit). Hieruit volgt dat een MIDI byte altijd tien bits bevat waarvan er acht veranderlijke informatie bevatten en twee altijd hetzelfde zijn. Omdat de laatste bit een stop bit is kan de volgende byte niet onmiddellijk er na beginnen maar moet er 1 / 31.250ste seconde tussen zitten. Dit betekent in concreto dat er nooit twee absoluut simultane messages verzonden kunnen worden. Het verschil is namelijk altijd minimal zes tiende milliseconde tussen: $2 * 1 / 31.250 = 2 / 31.250 = ca. 0,00064$ seconde.

DE MIDI COMMANDO'S

Wanneer je een toon aanslaat op een MIDI keyboard (maar ook op andere MIDI instrumenten zoals een *drum pad* of *wind controller*), wordt er informatie gezonden in de vorm van *MIDI messages*. Zo'n *message* bevat verschillende bytes: de eerste byte beschrijft de aard van de actie (een toets is aangeslagen). De tweede byte beschrijft welke toon is aangeslagen en de derde byte beschrijft de aanslag (*velocity*). De eerste byte wordt ook wel de *command byte* of *status byte* genoemd omdat die aangeeft wat er gebeurt. De tweede en derde bytes zijn de *status of data bytes* omdat ze aangeven hoe het gebeurt. De spec geeft precies aan hoeveel en welke waarden van de status- of databytes zijn toegestaan.

MIDI commando's kunnen geschreven worden in de zogenaamde binaire notatie. Dit zijn de wel bekende nullen en enen. Toch is dit niet een voor mensen bruikbare notatie omdat mensen in het dagelijks leven de zogenaamde decimale notatie gebruiken. Op die manier wordt *1001 1101* min *0111 0101*, *157 – 117*. Afgezien van de binaire en decimale notatie van getallen is er nog een derde. Dit is de zogenaamde hexadecimale notatie, afgekort hex. In hexadecimale notatie is het rechter cijfer van een tweecijferig getal het 1 nummer en het linker cijfer het 16 nummer. De definitie van hexadecimale notatie in decimale notatie

wordt dan: $hex. = 16n + 1x$ (met n als linker cijfer en x als rechter cijfer). Op die manier wordt het getal $hex. 14 = (16*1) + (1*4) = 20$ in decimale notatie. Het voordeel van de hexadecimale notatie is dat alle voor MIDI benodigde getallen uitgedrukt kunnen worden als twee-cijferige getallen. Het linker cijfer geeft het daarbij linker deel van de acht *status bits* weer, terwijl het rechter cijfer het rechter deel weergeeft (zie onderstaand figuur).



Een probleem met het hexadecimale systeem vormen de getallen hoger dan 9. Deze worden achtereenvolgens weergegeven door de letters A t/m F (10 t/m 15).

De eerste helft (de eerste vier getallen) van de *status byte* bepaalt de aard van het commando. Omdat de hoogste bit altijd een 1 is, is het eerste hex cijfer (het linker) altijd tussen de 8 en F (in binair tussen 1000 en 1111, en decimaal 128 en 255⁵). Dat levert de volgende mogelijk commando's op met hun bijbehorende vorm van de eerste vier bits. De ordening zoals die hier gepresenteerd is de letterlijke ordening uit de spec.

⁵ En niet 256 omdat de 0 als getal geldt.

EEN VOORBEELD VAN DE WERKING VAN MIDI: *PITCH BEND*

$9nH^6$ is het *Note On* commando. Wanneer deze *message* wordt ontvangen, wordt er een noot gespeeld. Dit commando heeft twee bytes: één voor het bepalen van het nootnummer en één voor de aanslag (*velocity*). De ambitus die gebruikt wordt door MIDI is door het volgende principe bepaald: $0000\ 0000$ tot $0111\ 1111$.⁷ Dat levert de volgende omvang op: *Hex. 00* tot *Hex. 7F*, in decimale notatie 0 tot en met 127 . Dit is de omvang voor zowel nootnummers als aanslagen. Vanwege het feit dat meestal met kleine secondes als kleinste interval wordt gewerkt, levert dit een mogelijk omvang van tien-en-een-half octaaf op.⁸ Ter oriëntatie, 64 is het nootnummer voor de ‘middelste *c*’ op een piano.

EnH is het *Pitch Bend* commando. *Pitch Bend* wordt normaalgesproken gecontroleerd door een soort wiel waarvan de ene helft boven het toetsenbord uitsteekt en die na beroering terugkeert naar het middelpunt met behulp van een veer. Omdat MIDI niet met negatieve nummers werkt, is een waarde van nul het laagste punt van de ambitus. Hieruit volgt dat 64 de standaardpositie is (*Hex. 40* en de middelste *c* op een piano).

Bij de meeste apparaten kan de mate van glissando ingesteld worden door de gebruiker. Op die manier kan bijvoorbeeld een arpeggio verkregen worden op de volgende wijze. Door een ambitus van 2 te nemen kan een bepaalde toon twee halve noten (ofwel een grote secunde) naar boven en naar beneden gebogen worden. Bij een ambitus van twaalf, is er een *bend range* van twee octaven, één hoger en één lager dan de originele noot. Bij sommige programma's is het ook mogelijk *steps* in te voeren in combinatie met een *range*. Op die manier zou je kunnen beslissen om een ambitus van twee octaven te combineren met kleine tertsen (wat verminderd septiem akkoorden zou opleveren).

MIDI ALS MUZIEKINSTRUMENT

Op welke manier is MIDI eigenlijk een instrument? In de keten

Componist -> *Muziek (geschreven)* -> *Uitvoerende* -> *Muziekinstrument* -> *Muziek (klinkend)* -> *Luisteraar*

bevindt MIDI zich tussen muziekinstrument en de klinkende muziek. Via een zogenaamde *pitch-to-MIDI-converter* kan de door een instrument geproduceerde klank via een microfoon omgezet naar MIDI data. Deze data (een wiskundige weergave van de gespeelde klank) kan vervolgens met behulp van software bewerkt worden. Dit proces kan *real-time* gebeuren. Op die manier kan MIDI gebruikt worden in uitvoeringen. In de meeste software kunnen spelers parameters zoals bijvoorbeeld *pitch bend* toewijzen aan een *controller*. *Controllers* hebben nummers binnen de MIDI conventie gekregen zodat ze universeel toepasbaar zijn. Wanneer, in het meest simpele geval, bijvoorbeeld een blokfluitist *pitch bend* wil toepassen op het geluid dat hij produceert, moet hij een *controller* hebben die hij met zijn voet kan bedienen. Een voorbeeld hiervan is een pedaal. In de software wijst hij dan *pitch bend* toe aan het specifieke *controller* nummer. Dit is een simpel voorbeeld omdat het ook uitgevoerd zou kunnen worden zonder MIDI maar met een *pitch bend* pedaal.

Een meer geavanceerde toepassing van MIDI is door met behulp van toonhoogteherkenning software een stukje muziek te laten spelen. Met behulp van de toonnummers kan software zo geprogrammeerd worden dat het spelen van een bepaalde toon, als startsein fungeert voor het afspele van een stukje muziek. Nog een stap verder is de software zo te programmeren dat door een andere toon te spelen, een stukje muziek van een te bepalen lengte wordt opgenomen. De speler heeft nu een *PLAY* en een *REC* functie toegewezen aan twee verschillende tonen. Op die manier ontstaat een interactieve situatie doordat de speler

⁶ n representeert de tweede helft van de acht *digits* en specificceert het nummer van het kanaal waar op audio wordt afgespeeld.

⁷ Het eerste cijfer van de data bits bij MIDI moet altijd een 0 zijn (daarom niet: $1111\ 1111$).

⁸ Het is bij sommige programma's ook mogelijk kleinere intervallen dan een kleine secunde specificeren. Het programma LiSa is hiervan een voorbeeld.

zelf kan bepalen op welk moment er een stukje muziek wordt opgenomen en op welk moment dat wordt afgespeeld. Dit is een voorbeeld van het zogenaamde *live-sampling*.

CONCLUSIE

Nu de primaire werking van MIDI bekend is, kunnen we het beschouwen in het licht van de traditionele indeling van muziekinstrumenten in excitator, oscillator en resonator. Bij digitale muziekinstrumenten fungeert de stroomtoevoer als excitator. De primaire toonvorming vindt plaats op de door Rasch genoemde 'virtuele' manier of digitaal gedefinieerde elektronische oscillatie.⁹ Ten laatste is duidelijk dat de luidsprekers fungeren als de resonatoren omdat zij de audio informatie overbrengen in de vorm van luchttrillingen.

MIDI valt naar mijn mening in deze indeling niet onder de muziekinstrumenten. MIDI is een conventie of internationale afspraak die communicatie van digitale muziekinstrumenten faciliteert. In die zin is het niet verschillend van de conventie die er bestaat over de manier waarop computers met printers communiceren. Op die manier denkend is MIDI zelfs te vergelijken met muzieknotatie. Ook dat is een conventie die communicatie via muzikaal schrift faciliteert. Guido's toepassing van gekleurde C en F sleutels binnen een vierlijnen-systeem bleek de ontwikkeling van complexere geschreven muziek te faciliteren en werd, evenals MIDI, al snel een internationale standaard.¹⁰

Dit gezegd hebbende kan MIDI dus niet ingedeeld worden in categorie 532, de digitale elektronische instrumenten maar eerder bij een nieuw gebied: Muzikale Conventies!

BIBLIOGRAFIE

Paul D. Lehrman & Tim Tully, *MIDI voor the Professional* (New York, 1993)

Rudolf Rasch, *Muziekinstrumenten* (Utrecht, 2003)

Joseph Rothstein, *MIDI, A Comprehensive Introduction* (Oxford, 1992)

Joseph Smits van Waesberghe, 'The Musical Notation of Guido of Arezzo'. In: *Musica Disciplina* (1951), pp.15 – 53.

Jos Zwaanenburg, *Composition and Performance with Live Electronics* (Hilversum, 1998)

⁹ Rasch (2003): 10

¹⁰ We zouden, analoog aan de publicatie van de spec, Guido van Arezzo's *Prologus in antifonarium* dan als beginpunt kunnen nemen.