

VOORWOORD T.B.V. UNI-MAN VERSIE 3.0d

Hallo beste MIDI-gebruiker,

U heeft zojuist de versie 3.0d van het programma "UNI-MAN" – "the Most Extensive Generic SYSEX Toolkit" aangeschaft. Een software pakket dat letterlijk voor iedere gebruiker van MIDI-apparatuur handig is om permanent binnen handbereik te hebben, om de meeste problemen die zich op MIDI/SYSEX-gebied voordoen op te kunnen lossen.

Om iedere MIDI-gebruiker daartoe in de gelegenheid te stellen hebben wij naar wegen gezocht om UNI-MAN ook daadwerkelijk voor iedereen prijstechnisch toegankelijk te maken. U als verse gebruiker weet als geen ander dat dat ons is gelukt, want voor de prijs waarvoor u nu deze goed uitgeruste gereedschapskist heeft aanschaf kunt u amper een manager of editor voor slecht één synthesizer kopen. UNI-MAN is tot op heden zonder twijfel het meest krachtige pakket uit Zadok's 'PM-TOOLS Sound Series' software lijn. *(Wilt u meer informatie over de ZADOK 'PM-TOOLS' neem dan contact op met uw dealer of direkt met Zadok Products).*

Als overigens iemand het softwarepakket UNI-MAN nu nog niet kent dan heeft deze persoon, sorry dat ik het moet zeggen, toch kennelijk een tijdje op Mars ofzo gebivakkeerd, althans ergens waar geen MIDI-literatuur verkrijgbaar is en 'midi' hooguit de lengte van kledingstukken aangeeft. Het programma behoeft dan ook eigenlijk geen introductie. Sla de uitvoerige (vaak zeer positieve) kritieken en artikelen in de internationale vakpers er maar eens op na, zoals: Keyboards (D), Atari-ST Nieuws, Sound on Sound (E) en Fachblatt (D). Deze laatste noemt UNI-MAN zelfs "das Universal Genie!" Die test behandelde slechts de 2.2 versie met een klein aantal adaptors, terwijl het hier om de nog meer verbeterde ultieme 3.0(!) versie in ruim 50(!) Device-Adaptors gaat.

Effuh Rech' Zettuh

Deze UNIVERSELE MANAGER afgekort als UNI-MAN (het paradepaardje uit Zadok's 'PM-TOOLS' SOUND SERIES) is in de pers altijd afgestempeld als Universele Editor. Helaas is dat niet geheel de juiste naam voor dit pakket, anders hadden wij het wel UNI-EDI genoemd. Het plaatst UNI-MAN zelfs in een verkeerd daglicht.

Om te beginnen is het uitgangspunt van UNI-MAN niet het editeren van sounds, maar is als concept gekozen: "het oplossen van problemen die je tegenkomt bij het werken met sounds en/of patches". Dat o.a. het editeren van parameters van synthesizers (e.a. MIDI-apparatuur) daar deel van uitmaakt is een (gelukkig niet onbelangrijke) bijkomstigheid, maar wel mooi meegenomen.

Zo kun je bijvoorbeeld voor ieder MIDI-apparaat op z'n minst een dump afleggen. Maar b.v. ook met de bijbehorende MIDI-Monitor (UNI-MON) zien wat er over MIDI gebeurt en is UNI-DUMP een tool (PRG/ACC) waarmee je letterlijk elke data over MIDI kunt binnenhalen en versturen.

Er zitten overigens nog veel meer handige opties en hulpprogramma's in de UNI-MAN-gereedschapskist. Kortom, UNI-MAN is een SYSEX-TOOL-KIT waar geen enkele serieuze MIDI-gebruiker omheen kan.

UNI-MAN is dan ook niet voor niets over de hele wereld in gebruik bij recording studio's, muziekscholen en conservatoria, op diverse universiteiten als studiepakket, bij diverse soundwarehouses voor het universeel behandelen van al hun soundlibraries, bij MIDI-workshops en -seminars, home studio's, etc.

Wij wensen u veel plezier en succes met deze UNI-MAN toolkit versie 3.0!

Uw Zadok-team.

INHOUDSOPGAVE

	Pagina
VOORWOORD t.b.v. UNI-MAN 3.0d	1
Effuh Rech' Zettuh	2
INHOUDSOPGAVE	3
Inhoud van de programma diskette	4
Hoe moet je UNI-MAN 3.0d installeren	5
INLEIDING	6
UNI-MAN Update 3.0 Algemeen	7
Receive-, Send- en Save-Bank-Parameters	9
Voor de Programmeurs van Device-Adaptors	10
DIVERSE UPDATES 1	11
Andere Programmeertechniek	11
Uitbreiding Grafische Mogelijkheden	12
Extra Computer Keyboard Funkties	12
UNI-LINK VERSIE 3.0	13
UNI-MAN VERSIE 3.0d EN DE DUMMY-BANK (VOORAF)	14
Dummy-Bank achter de Exclusive-Bank	15
Hoe wordt in UNI-MAN zo'n Achteraf-Dummy-Bank gemaakt?	15
Overall-Data	16
Bank-Receive-Adaptor	17
Sound/Patch-Receive-Adaptor	17
Sound/Patch-Send-Adaptor	18
De Dummy-Bank Zelf	19
UNI-CON 3.0	20
DIVERSE UPDATES 2	24
Individual Parameter Handling	24
'Jump To' Fixed Parameter	24
Aanpassing Edit-Formulas	24

Inhoud van de programma diskette

UNI-MAN 3.0d, welke niet meer hardwarematig (d.m.v. een 'sleutel') beveiligd is, heeft op diverse fronten wijzigingen ondergaan. Een logisch gevolg hiervan is dat de diskette-inhoud is veranderd. Sinds UNI-MAN versie 3.0 zijn de voordelen van het programma UNI-BOY niet langer meer van toepassing en is als zodanig komen te vervallen.

De disk inhoud van UNI-MAN 3.0d is als volgt:

UNI-DUMFACC	UNI-CHAR.PRG
UNI-CON.PRG	UNI-DUMFPRG
UNI-LINK.PRG	UNI-MAN.PRG
UNI-MON.PRG	UNI-NUM.PRG
UNI-DUMFRSC	UNI-MON.RSC
UNI-NUM.RSC	

De folder 'RESOURCE':

ALL-MAN.RSC	ALL-RESTRSC
UNI-CON.RSC	UNI-LINK.RSC

De folder 'UNI-MAN':

INITD7V	STANDARD.DA
UNI-LINK.HF	STANDARD.NP
UNI-MAN.PIC	UNI-CHAR.SC1
UNI-CON.SC1	UNI-MAN.SC1
UNI-MAN.SC2	UNI-MAN.SC3
INITSEQ	

ZEER BELANGRIJK: Aan de inhoud van de programma-diskette mag niets worden gewijzigd of toegevoegd. Zelfs niet een nieuwe desk-top of een verandering van een extentie. Doet u dit wel, dan zal de disk zijn beschadigd en de werking van de programma's op zijn minst zijn verstoord. Zorg er dus voor dat de diskette altijd beveiligd is (nokje in de disk zodanig verschoven dat het gat zichtbaar is).

Hoe moet je UNI-MAN 3.0d installeren?

BELANGRIJK: Als je programma's als ACCESSORY gebruikt, of dit nu vanaf harddisk is of vanaf diskette, dan moet je voordat je een ander programma opstart eerst even de ACC in de desktop aanklikken (activeren) met de masterdisk in drive A en weer verlaten. Start daarna pas het gewenste programma op. Doe je dit niet dan zal het programma niet of slecht opstarten en zelfs kunnen vastlopen. Denk hier dus aan!

Gebruik vanaf floppy, met UNI-DUMP als accessory:

1. Doe de Masterdisk in drive A
2. Zet de monitor aan
3. Zet de computer aan

De accessory zal bij het opstarten automatisch worden geïnstalleerd en bevindt zich daarna, net als iedere andere ACC, onder het meest linker drop-down venster.

NOTE: UNI-MON.PRG en UNI-NUM.PRG kunnen ook als ACCESSORY worden gebruikt. Om ze als ACC te gebruiken moet je ze naar een andere diskette of harddisk kopiëren. Vergeet daarbij niet om ook de bijbehorende *.RSC- files mee te kopiëren.

Gebruik vanaf floppy, zonder accessories:

1. Doe iedere andere (geformatteerde) diskette in drive A
2. Zet de monitor aan
3. Zet de computer aan
4. Als de desk-top zichtbaar is en de floppy-drive klaar is (lampje niet meer brandt) diskette in drive A verwisselen door de UNI-MAN programma schijf.
5. Start (d.m.v. dubbel-klik) het gewenste programma op.

Gebruik vanaf harddisk:

1. Kopieer de totale inhoud van de UNI-MAN programma diskette naar drive C (de root directory) van de harddisk.
2. Start (d.m.v. dubbel klikken) het gewenste programma op. Zorg daarbij dat de programma disk zich ALTIJD in drive A bevindt.

Bij de eerstvolgende koude start zullen de accessoires automatisch worden geïnstalleerd. Zorg bij geactiveerde ACC's ervoor dat ook bij het opstarten van de harddisk de programma diskette zich in drive A bevindt.

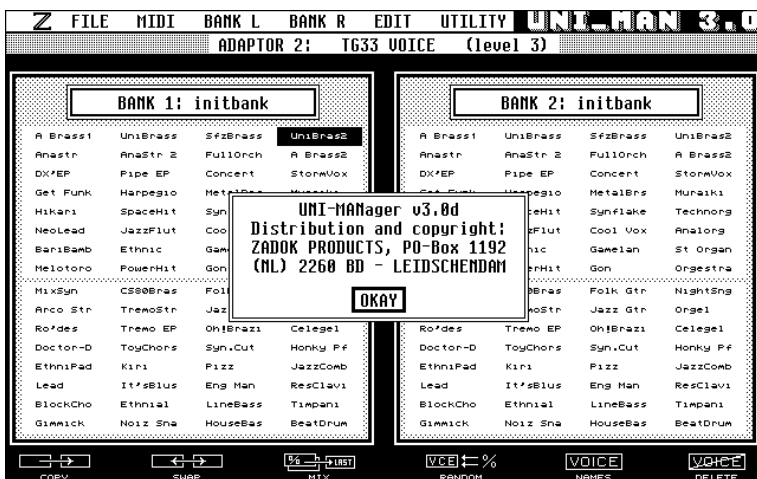
Zorg er ook nu weer voor dat je de ACCESSORIES eerst even activeert voordat je een PROGRAMMA opstart.

NOTE: Indien u geen gebruik wilt maken van de accessoires verander (uitsluitend op hard-disk, dus niet op diskette!) de extentie van ACC in b.v. ACX. Het betreffende accessoire-programma zal dan in het vervolg bij het opstarten niet meer in de desk-top verschijnen.

INLEIDING

Toen UNI-MAN (alweer een paar jaar geleden) als eerste volledig UNIVERSELE MANAGER op de markt kwam, heeft dat bij veel mensen naast bewondering ook wantrouwen opgeroepen. Omdat immers iedere synthesizer zijn eigen "dialect" sprak, was er weinig vertrouwen in dit fenomeen. Men zou weliswaar graag één programma willen hebben waarmee alle MIDI apparaten te managen/editen waren, maar dan moest dit toch eerst zijn bestaansrecht bewijzen.

Sinds de totstandkoming van het universele UNI-MAN systeem zijn tussen betrokkenen (programmeurs, pers en uiteraard UNI-MAN gebruikers) vele suggesties ter verbetering uitgewisseld. Ook het werken onder allerlei omstandigheden met een dergelijk programma heeft een schat aan ervaringen opgeleverd. Inderdaad bleken een aantal aanpassingen niet alleen wenselijk, maar soms ook noodzakelijk. Los daarvan, hebben krachtige updates voor een beter inzicht en het enorm toenemen van de mogelijkheden gezorgd. Vooral het implementeren van de librarian-functies met uitgebreide random, mix, search en sort mogelijkheden betekende een belangrijke stap in de richting van de gebruiker.



Dit alles heeft er toe geleid dat we nu een uitgerijpt totaal systeem in de versie 3.0 beschikbaar hebben dat beduidend minder weerstand heeft. In alle toepassingsgebieden blijkt UNI-MAN zich inmiddels een vaste plaats te hebben veroverd bij de dagelijkse werkelijkheid van professional en amateur. In tegenstelling tot de beginperiode bereiken ons steeds meer reacties van tevreden gebruikers. Hieruit is o.a. ook gebleken dat één van de sterke punten van UNI-MAN de mogelijkheid is om te kunnen werken met MIDI-apparatuur waar **nooit** software voor geschreven is of überhaupt voor geschreven zal worden.

In binnen en buitenland groeit de belangstelling. Het doet ons dan ook genoeg om UNI-MAN onlangs bij een POL (Publieke Opinie Lijst) in Duitsland van het gezaghebbende 'MUSIK FACHBLATT' (na enkel sequencers van C-lab en Steinberg) met stip werd gekozen tot beste en meest belangrijke MIDI-software. (Zie *Fachblatt no-3 van maart '91 en voor de 'UNIVERSAL GENIE'-test Fachblatt no-1 van januari '91*).

UNI-MAN Update 3.0 Algemeen

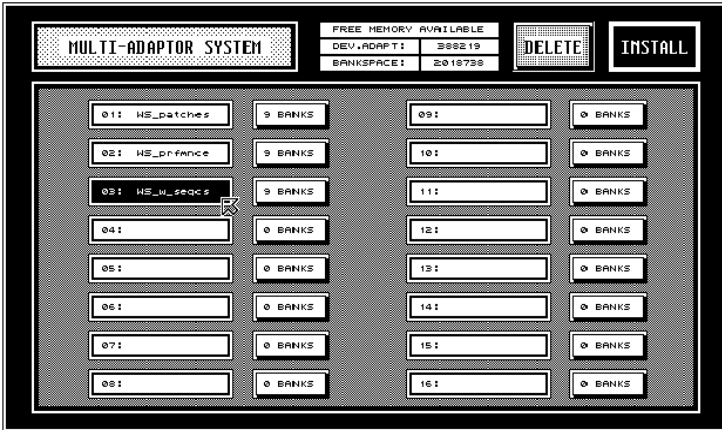
Doordat de programmering van UNI-MAN volgens een bepaalde (geplande) ontwikkelingslijn verliep, bleek het mogelijk om de handleiding zodanig te schrijven dat deze slechts weinig of geen verandering nodig zou hebben bij verdere groei van het systeem. Filosofie en concept blijven onaangetast, evenals reeds in gebruik zijnde Device-Adaptors. Dat is enkel mogelijk, omdat van meet af aan voldoende ruimte is gelaten voor latere ontwikkelingen. De te verwachten (voorziene) veranderingen naar versie 3.0 zijn zodoende voor een groot deel reeds omschreven en behoeven dus geen nadere uitleg. Voor zover het verbeteringen op gebruikersniveau betreft, zijn incidentele verklaringen noodzakelijk en enkele structurele omschrijvingen onvoorkomelijk. Programmeertechnische veranderingen (zoals geheugenbesparing e.d.) zullen ook kort worden genoemd.

De aanpassingen naar versie 3.0, zoals in deze aanvullende handleiding omschreven, zijn het direkte gevolg van veranderingen op de synthesizer-markt. Aanvankelijk was de geheugenstructuur van MIDI-devices vrij eenvoudig. Deze eenvoud was voor een universeel programma zeer welkom, zo niet een voorwaarde. Ook de gebruiker profiteerde van dit doorzichtige en overzichtelijke concept. De laatste jaren echter komen steeds meer apparaten met omvangrijke geheugen-configuraties en ingewikkelde structuren op de markt. Een en ander heeft ons doen besluiten voor een nog meer flexibele programmastructuur.

Tot en met UNI-MAN versie 2.2 was het niet mogelijk (en nodig) om met een specifiek stuk geheugen te communiceren. Tot dan toe was de grondgedachte achter UNI-MAN, dat alle "system exclusive data" binnen het programma onveranderd bleven. In de praktijk betekende dit, dat een eenmaal ontvangen BANK te allen tijde de system-exclusive data bleef behouden die door het MIDI-device zelf werden afgegeven.

Tijdens het ingeven van het request in de 'BANK RECEIVE ADAPTOR' moest dus ook worden bepaald c.q. vastgelegd waar deze bank tijdens het zenden ('COMPUTER ==> MIDI-DEVICE') naar toe ging. Dat was even eenvoudig als doeltreffend. Als er in een enkel geval (bv. KAWAI K1) sprake was van meer dan één bankgeheugen, dan werd dezelfde adaptor nogmaals ingeladen en moest daarna worden voorzien van een kleine wijziging in het request. Er moesten dus evenveel adaptors beschikbaar zijn, als dat er bankgeheugens in het MIDI-apparaat waren. Iedere Device-Adaptor correspondeerde dan met het overeenkomstige bankgeheugen.

Op die manier ontstond een comfortabele en overzichtelijke manager/editor die, vanuit het M.A.S., als een database werkte. Deze werkwijze kon worden gevolgd, omdat Device-Adaptors en banken voor dergelijke MIDI-apparatuur vrij klein waren. Het nadeel daarbij was dat alleen de banken (en daarmee de sounds/patches) van de bijbehorende Device-Adaptor ter beschikking stonden. Uitwisselen van sounds/patches tussen verschillende adaptors moest dan via de disk gebeuren.



De laatste tijd is, zoals eerder gezegd, de omvang van soundbanken van MIDI-devices steeds toegenomen en is de geheugenconfiguratie ook steeds ingewikkelder geworden. Bovendien is het, in tegenstelling tot voorheen, bijna altijd mogelijk om via MIDI ook rechtstreeks met de data-CARD(S) te communiceren. Ook is het in toenemende mate mogelijk om via MIDI diverse codes, zoals bijvoorbeeld voor conversie-doelende of program-changes per device-nummer mee te geven, enz. Vooral bij toepassing van meerdere soortgelijke apparaten, die slechts via een ander device-nummer kunnen worden aangesproken en bovendien meer dan één bank bezitten, is de coördinatie moeilijker.

Tenslotte blijkt de zendsnelheid bij grotere banken gevoeliger te zijn dan voorheen. Grote data-hoeveelheden die met hoge (lees: maximum) snelheid worden verstuurd, kunnen niet door ieder MIDI-apparaat even makkelijk worden bijhouden. Bovendien verlangt het ene MIDI-device **wel** een korte pauze tussen twee bij elkaar behorende dumps en het andere juist **niet**.

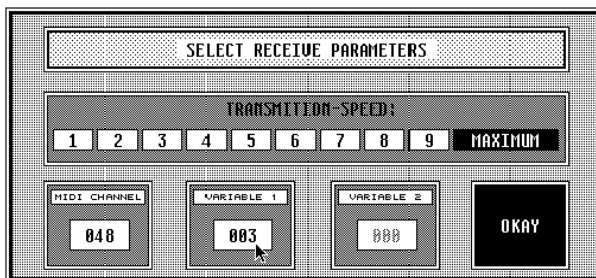
Deze ontwikkelingen hebben er toe geleid dat UNI-MAN structureel is veranderd. Dankzij de in deze aanvullende handleiding beschreven veranderingen is het nu **wél** mogelijk om als gebruiker invloed uit te oefenen op de RECEIVE- en SEND-PARAMETERS. Daardoor is vanaf nu voor ieder MIDI-device slechts één Device-Adaptor per parametergroep (b.v. de PATCHES) nodig.

Zowel bij het ontvangen als het verzenden van een bank kan er worden bepaald waar deze vandaan dient te komen of naar toe gaat. Bovendien kan het MIDI-kanaal c.q. device-nummer worden ingesteld. Om tenslotte invloed te kunnen hebben op de 'SEND-PARAMETERS' van de bank die op disk staat, geldt hetzelfde tijdens 'SAVE-BANK'. Zodoende kan de gebruiker bepalen met welk MIDI-kanaal en waar bijvoorbeeld 'UNI-DUMP' de bank heen stuurt. Omdat de header tijdens 'RECEIVE-', 'SEND'- en 'SAVE-BANK' wordt aangepast, kan bovendien elke bank (en daarmee elke sound/patch) in dezelfde Device-Adaptor worden geladen. Dat laatste feit is, naast de enorme geheugenbesparing, misschien wel het grootste voordeel van deze werkwijze. Anderzijds blijft de werkwijze zoals voorheen gebruikelijk was, onverkort mogelijk.

Voor de versie 3.0 is een meer uitgebreide Device-Adaptor noodzakelijk. De gebruiker zal daar echter geen problemen van ondervinden. Om een bestaande adaptor om te schrijven naar het nieuwe formaat is het voldoende deze eenmaal in UNI-MAN 3.0 in te laden en vervolgens (met de juiste RECEIVE- en SEND-PARAMETERS) naar disk te schrijven.

Receive-, Send- en Save-Bank-Parameters

Zodra één van deze opties wordt gekozen verschijnt er automatisch een box in beeld, waarin de genoemde parameters zijn in te stellen.



In principe kan de zendsnelheid maximaal zijn. Sommige MIDI-devices echter, kunnen grote hoeveelheden data niet met grote snelheid verwerken en verlangen dus een lagere zendsnelheid. Het apparaat maakt dit doorgaans kenbaar middels een 'CHECKSUM ERROR' of andere mededelingen zoals 'COMMUNICATION ERROR', e.d.. Soms blijft het MIDI-device zelfs 'HANGEN' (TOTAL ERROR) en weigert verder iedere dienst. Ongeveer 10 seconden uitschakelen en nogmaals (met een lagere zendsnelheid proberen) is dan de enig juiste oplossing. De juiste waarde kan het beste proefondervindelijk worden vastgesteld. Als een bulkdump uit meerdere data-blokken is opgebouwd, dan is het mogelijk dat er een korte pause tussen elk blok nodig is. Door een lager getal te kiezen, wordt automatisch een pause ingelast die overeenkomt met de lagere zendsnelheid.

In feite behoort het vaststellen van de juiste zendsnelheid bij het maken van de Device-Adaptor. UNI-MAN onthoudt in de nieuwe adaptor-versie 3.0 in de praktijk dit gegeven, zodat de gebruiker zich daarover geen zorgen hoeft te maken.

Beneden in de box kunnen drie parameters worden ingesteld die betrekking hebben op het REQUEST van de te ontvangen bank en de HEADER van de te verzenden bank. In principe bestrijken alle drie de parameters hetzelfde instelbereik en kunnen ze voor elk willekeurig doel worden ingezet. Voor het gemak van de gebruiker zijn hieraan wel de meest voorkomende namen gegeven, respectievelijk: 'MIDI-CHANNEL', 'VARIABLE-1' en 'VARIABLE-2'. Het bereik loopt van '0' tot en met '207' in het decimale stelsel. Deze range is zo uitgebreid gekozen om in het request ook b.v. een program-change over elk gewenst MIDI-kanaal te kunnen meegeven. Een voorbeeld hiervoor vormt de DW/EX 8000 van KORG. Omdat sommige MIDI-devices dat verlangen, kunnen naast het MIDI-kanaal nog **twee** variabelen worden geïnstalleerd (zie b.v. *KAWAI K1*).

De parameter 'MIDI-channel' correspondeert in het request met het symbool 'FC'. Voor de parameters 'Variable-1' en 'Variable-2' worden respectievelijk de symbolen 'FA' en 'FB' gebruikt. Door deze symbolen in het request te plaatsen, kan de gebruiker op eenvoudige wijze vastleggen op welke plaats de overeenkomstige parameter instelbaar dient te zijn.

Voor de Programmeurs van Device-Adaptors

Deze paragraaf is speciaal bedoeld voor de groep van gebruikers die zich interesseert voor het maken van Device-Adaptors. Vanaf UNI-MAN 3.0 weet het programma (automatisch) op welke plaats en met welke waarde de header van de in het geheugen aanwezige banken, moet worden aangepast. Voor het instellen van bovengenoemde parameters, hoeft de gebruiker ook geen rekening te houden met zaken als: packet-formaat, al of geen checksum over de header, het binnenhalen van banken als één geheel, of juist als een aantal sounds/patches, enz.. Dit alles wordt volledig en automatisch correct verwerkt. Omdat **sounds/patches** corresponderen met het temporary-buffer van het MIDI-device (of geheugenplaats die als zodanig dienst doet) zijn deze voorzieningen hierop niet van toepassing. Ook de zendsnelheid speelt dan geen rol.

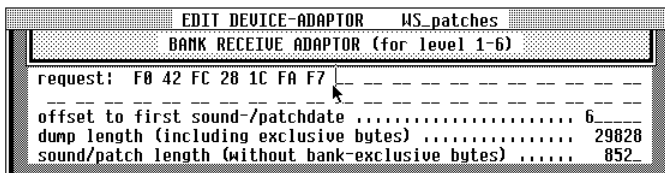
Voor alle duidelijkheid wordt in dit geval als voorbeeld de KORG WAVESTATION genomen. We beschouwen eerst het request van bv. een hele bank van patches: 'F0 42 3M 28 1C 0B F7'

F0 = system exclusive status byte
 42 = KORG id(entication number)
 3M = format id (M = MIDI channel)
 28 = wavestation id
 1C = all patch dump
 0B = bank number
 F7 = end of exclusive

Zoals de meesten wel bekend zal zijn, staat het MIDI-kanaal doorgaans op de derde plaats. Zo ook in dit geval. Dat is dus de plaats waar de programmeur van de Device-Adaptor 'FC' moet invullen. De getallen in het request zijn hexadecimaal. Ze moeten worden vertaald naar waarden die de muzikant in de dialog-box kan instellen. Voor MIDI-kanaal '1' (waarde '0') dient de gebruiker dus het decimale getal '48' in te stellen. Immers, $3M = 30H (3 \times 16 + 0) = 48$ (**look te controleren met de UNI-NUM accessory**). Bij het naar disk wegschrijven van de 3.0 Device-Adaptor, wordt deze waarde onthouden en verschijnt tijdens het gebruik automatisch op de juiste plaats.

Op de zesde plaats staat een onlangs door het I.M.A. (the International MIDI Association) in het leven geroepen bank-parameter. De waarde '0' correspondeert met RAM-1, '1' met RAM-2, '2' met het ROM en tenslotte '3' met de CARD. In dit geval is één variabele dus voldoende om de geheugenplaats aan te duiden. Op deze plaats dient de Device-Adaptor-programmeur 'FA' in te voeren.

In het geval dat er meer dan één request nodig is, zal deze aanpassing ook meer dan één maal moeten plaatsvinden.



DIVERSE UPDATES 1

Andere Programmeertechniek

Door toepassing van de nieuwste programmeertechnieken is het mogelijk gebleken om het programma UNI-MAN belangrijk in te korten. UNI-MAN is op diskette zelfs bijna 200 Kbyte kleiner. Dit betekent dat voortaan alle programma's, die behoren tot het UNI-MAN SYSTEM, op één disk staan. Bovendien betekent het een belangrijke geheugenwinst binnen de computer. De netto winst is weliswaar niet zo spectaculair als bij de programma-lengte, maar toch aanzienlijk. Als bijkomend voordeel zijn beide programma's (afhankelijk van de optie) gemeten tussen 66 en 100 procent versneld. D.w.z. dat de meeste opties nog slechts de helft van de tijd duren, dan voorheen nodig was. Dit is zeer welkom, omdat krachtige universele programma's nu eenmaal niet de snelste kunnen zijn. Dit vanwege de ingewikkelde scanner-lussen die met "1001" mogelijkheden rekening dienen te houden. Bovendien moet er (in het belang van de muzikant) vaak worden "gewoekerd" met het geheugen, waardoor slechts een klein gedeelte resteert voor het bedrijfssysteem en het programma. De beste demonstratie hiervoor is de optie 'EDIT DEVICE-ADAPTOR'. Tijdens de eerste keer doorstappen van de verschillende resourceboxen neemt de computer enkele seconden wachttijd (zgn. 'garbage collection'). De tweede en volgende keer gaat dit echter vliegensvlug. Het geheugen is dan, zoals dat heet, "geactualiseerd".

Uitbreiding Grafische Mogelijkheden




In het kader van het streven naar volledigheid, is de lijngrafiek uitgebreid met een mogelijkheid tot het weergeven van negatieve waarden. Het kan namelijk zeer wenselijk zijn om b.v. alle detune-waarden in beeld te hebben. Vanaf de versie 3.0 kan in het GRAPHICAL WINDOW voor dat doel een negatieve offset worden ingegeven, achter de parameter 'GRAPHICAL ZERO'. Een voorbeeld is te vinden in de Device-Adaptor 'PATCHES' van de KORG WAVESTATION.

ENTER GRAPHICAL PARAMETERS

GRAPHICAL WINDOW NUMBER:

GRAPHICAL WINDOW NAME :

GRAPHICAL ZERO (+/-127):

OKAY

CANCEL

RECIPROCAL VALUE	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	
VERTICAL-AXIS →	<input type="text" value="0"/>	<input type="text" value="60"/>	<input type="text" value="142"/>	<input type="text" value="224"/>	<input type="text" value="306"/>	<input type="text" value="1"/>	<input type="text" value="-1"/>	<input type="text" value="-1"/>	<input type="text" value="-1"/>	
GRAPHIC POINT	<input type="text" value="01"/>	<input type="text" value="02"/>	<input type="text" value="03"/>	<input type="text" value="04"/>	<input type="text" value="05"/>	<input type="text" value="06"/>	<input type="text" value="07"/>	<input type="text" value="08"/>	<input type="text" value="09"/>	<input type="text" value="10"/>
VALUE/HOR-AXIS →	<input type="text" value="0"/>	<input type="text" value="20"/>	<input type="text" value="20"/>	<input type="text" value="20"/>	<input type="text" value="20"/>	<input type="text" value="20"/>	<input type="text" value="-1"/>	<input type="text" value="-1"/>	<input type="text" value="-1"/>	
RECIPROCAL VALUE	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	
POINT-NAME →	<input type="text" value="DetA"/>	<input type="text" value="DetB"/>	<input type="text" value="detC"/>	<input type="text" value="detD"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	

LEVEL/DEPTH/
GAIN/ETC.

TIME/RATE/
FREQ./ETC.

(e . s . R D S R)

Extra Computer Keyboard Funkties

Zowel in de MANAGER, het M.A.S. en de EDITOR zijn nieuwe computer keyboard-funkties geïmplementeerd:

In de MANAGER:

Toets	Funktie
[F1]	Copy sound/patch or bank
[F2]	Swap sound/patch
[F3]	Mix sound/patch or bank
[F4]	Random sound/patch or bank
[F5]	Names sound/patch
[F6]	Delete sound/patch
[F7]	Bank to Librarian
[F8]	Librarian to bank
[F9]	Compare sound/patch or bank
[F10]	Sequencer
[L]	Load funkties
[S]	Save funkties
[E]	Edit aan te klikken sound/patch
[M]	M.A.S. (zie ook [ESCAPE] toets)
[F]	Format disk
[D]	Delete/Show file
[P]	Print bank
[T]	MIDI THRU
[ESC]	M.A.S.

In het M.A.S.:

[D]	Delete Device-Adaptor
[RETURN]	Install Device-Adaptor/aantal banken
[CURSOR] up/down	Selecteer Device-Adaptor
[CURSOR] left/right	Device-Adaptor ↔ banks
[-], [+] tijdens banks	Verminder/vermeerder aantal banken

In de EDITOR:

[SHIFT] + [J]	Fixeer Parametergroep Jump punt
[J]	'JUMP TO' gefixeerde parametergroep.

UNI-LINK VERSIE 3.0

Gezien de veranderingen die in de Device-Adaptor en de verwerking daarvan hebben plaatsgevonden, is het ook noodzakelijk gebleken om UNI-LINK aan te passen. Alleen UNI-LINK versie 3.0 verwerkt de adaptors 3.0 correct en levert weer een adaptor af die integraal is ingericht voor bovengenoemde functies.

Wellicht ten overvloede nog een voorbeeld ter illustratie van het één en ander. Als voorbeeld is weer de KORG WAVESTATION gekozen. Deze synthesizer heeft drie geheugens:

1. RAM-1
2. RAM-2
3. CARD

Elk van de drie geheugens bevat:

1. PERFORMANCES
2. PATCHES
3. WAVE-SEQUENCES

In de UNI-MAN versie tot en met versie 2.2 zou een volledige editering van de Wavestation negen (3 x 3) Device-Adaptors vereisen. Omdat de adaptors en de banken ook nog eens vrij groot zijn, was het onmogelijk om één set van drie Device-Adaptors binnen het geheugen een 1040ST te laden, laat staan te laten werken. Dankzij de geheugenbesparing, die door de nieuw toegepaste programmeertechniek tot stand is gekomen, is dat nu **wel** mogelijk.

In de WAVESTATION FOLDER bevinden zich de volgende adaptors:

1. Multi-Setup-WS
2. Patches-WS
3. Performances-WS
4. Wave-Sequences-WS
5. Total-WS

Bovendien zijn nog een aantal REQUESTS afgelegd (in de folder 'REQUESTS') en een tweetal CONFIGURATIES (MEGA-2-4.CFG en 1040-BOYCFG). De requests zijn bedoeld om ook een SYSTEM-SETUP en eventuele GLOBAL-data op disk te kunnen schrijven, middels de optie 'RECEIVE MIDI-FILE' of d.m.v. UNI-DUMP. Het is overigens heel goed mogelijk om, indien daar behoefte aan bestaat, een kleine LEVEL-6 adaptor te maken waarmee ook deze 'overall-data' kunnen worden geëditteerd. Voorbeelden daarvan zijn op de originele DATA-disks te vinden. In de praktijk zal deze behoefte niet zo groot zijn, omdat het doorgaans om een eenmalige instelling gaat.

Zoals iedereen die de Wavestation goed kent wel weet, is een bepaalde werkwijze noodzakelijk. Om de Wavestation handig te kunnen gebruiken, is het nl. nodig om binnen de PERFORMANCE-BANK (van bijvoorbeeld RAM-1) uitsluitend gebruik te maken van patches en wave-sequences uit RAM-1. Het is anders niet mogelijk om soundbanken uit te wisselen, tenzij men het hele geheugen (van ongeveer 180.000 bytes zonder CARD) vervangt.

Omdat het absoluut geen zin heeft om **alleen** patches of **alleen** performances te vervangen, werd met UNI-LINK versie 3.0 een Device-Adaptor gemaakt die de **hele** inhoud van RAM-1, RAM-2 of CARD vervangt. Zelfs adaptors die zoals in dit geval met **meerdere** banken werken hoeven voortaan slechts éénmalig te worden aangemaakt. Dit wordt mogelijk, omdat UNI-MAN na het opgeven van de hier omschreven parameters automatisch eerst alle afzonderlijke 'HEADERS' aanpast, voordat deze worden verstuurd c.q. ontvangen. Ondanks het feit dat de bijbehorende banken zo groot zijn, 'past' een dergelijke adaptor nu makkelijk in een 1040ST computer. Bij een eenmalige aanpassing van het MIDI-kanaal en/of geheugen-adres **vóór** het zenden of ontvangen van de data zal UNI-MAN dit automatisch (voor alle performances, patches en wave-sequences) doorvoeren, zodat ze allemaal binnen het opgegeven RAM-geheugen blijven.

Bij sommige oudere software-versies (van de Korg-WS1) kan het voorkomen dat de wavestation de (maximum) zendsnelheid van UNI-MAN niet kan bijhouden. In dat geval, is het nodig om een langzamere snelheid in te stellen. Ook bij gebruik van de optie 'SEND MIDI FILE' is deze zendsnelheid op te geven. Bij de meeste MIDI-devices zal een vertraging echter **niet** nodig zijn.

UNI-MAN 3.0d EN DE DUMMY-BANK (VOORAF)

Het gedeelte dat nu wordt besproken t.b.v. de update naar UNI-MAN versie 3.0c is in hoofdzaak van belang voor gebruikers die zich bezig (gaan) houden met het vervaardigen van Device-Adaptors. Diegenen welke UNI-MAN uitsluitend gebruiken als Manager, Editor, Data-File, Sound-Creator, Librarian e.d. hoeven dit gedeelte **niet** door te nemen. Voor iedereen die wat meer interesse heeft voor 'MIDI-System-Exclusive', kan het anderzijds geen kwaad om dit gedeelte **toch** te lezen. Het is misschien zelfs aan te bevelen!

Dummy-Bank Achter de Exclusive-Bank

Het maken van een DUMMY-BANK welke zich **achter** de eigenlijke 'exclusive'-bank bevindt wordt in deze paragraaf besproken. De hierna volgende bewerkingen zijn mogelijk vanaf de 3.0c versie van UNI-MAN. Met het oog op grotere universaliteit heeft het programma nl. omvangrijke wijziging ondergaan t.o.v. de 2.2 (en zelfs de 3.0b) versie.

In de handleiding van UNI-MAN is er diverse keren sprake van een zgn. 'DUMMY-BANK'. Dat is een bank die, zoals de meeste UNI-MAN gebruikers weten, data bevat die **niet** tot de sounddata behoren.

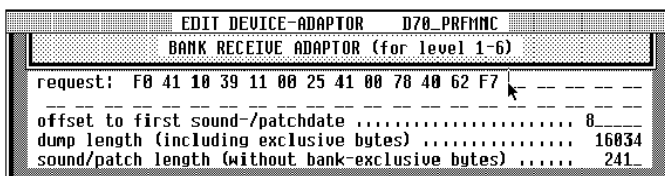
Het meest sprekende (tevens meest bekende) voorbeeld is de DUMMY-BANK welke aan de eigenlijke 'system exclusive bank' **vooraf** gaat met de bedoeling om bv. SOUND-NAMEN (of opmerkingen omtrent een sound) toe te voegen. Dit wordt vaak toegepast als het MIDI-device zelf geen sound/patch-namen heeft, of als de muzikant behoefte heeft aan een andere vorm van klassifikatie. Alle voorwaarden ivm. het correct functioneren van een dergelijke bank worden in de handleiding meer dan eens uit de doeken gedaan.

Een enkele keer kan het voorkomen dat het handig of wenselijk is om, **na** de eigenlijke sounddata, extra data te versturen. Dat is bv. het geval bij synthesizers die, nadat ze sounddata hebben ontvangen, nog een WRITE-REQUEST verlangen. (o.a. de D70 van ROLAND is zo'n apparaat.) Pas na ontvangst van een write-request zal in dat geval bv. de display correct worden ingelezen en de data als definitief worden beschouwd. Belangrijk is, dat vaak alleen de sounds (en niet de hele bank) zo'n write-request moeten meekrijgen. Daarbij kunnen zelfs die write-requests per sound verschillend zijn. Iedere sound kan immers zijn eigen geheugen-adres hebben.

Het kan ook nodig zijn om het aangesloten MIDI-device, na ontvangst van exclusive data, in een bepaalde 'MODE' te zetten c.q. terug te plaatsen. In dat geval zijn de extra data doorgaans identiek.

Hoe wordt in UNI-MAN zo'n Achteraf-Dummy-Bank gemaakt?

Voor het gemak nemen we even aan dat het aangesloten MIDI-device één bank afgeeft waarin zich alle sounds/patches bevinden. Als voorbeeld geldt de D70 van ROLAND. Zoals gezegd, verlangt deze synthesizer, na ontvangst van een sound, een write-request van 13 bytes.



In een bank zijn 64 'PERFORMANCES' aanwezig. Omdat de rest breed-voerig in de handleiding wordt uitgelegd, besteden we op deze plaats alleen aandacht aan dat gedeelte wat betrekking heeft op de DUMMY-BANK.

Overall-Data

Om te beginnen dient onder 'OVERALL DATA', zowel bij het 'NUMBER OF DUMPS PER BANK' als bij het 'NUMBER OF DUMPS PER SOUND/PATCH', een '2' te worden ingevoerd. Vooral als **alleen** de sound/patch-data van een write-request moeten worden voorzien, is de neiging groot om slechts de tweede vraag met '2' te beantwoorden. Dat zou correct zijn als de betreffende data ook daadwerkelijk in de eerste bank aanwezig waren. In het geval van een extra DUMMY-BANK echter, dient UNI-MAN zijn additionele gegevens uit deze tweede bank te halen, vandaar het gekozen getal '2'.

```

EDIT DEVICE-ADAPTOR   D70_PRFMC
-----
OVERALL DATA (for all levels)
-----
device name ..... D70_PRFMC
file extender ..... PRF

number of snds/patches (max.128)... 64_
name-offset in sound/patch ..... 0__
length of sound-/patchname ..... 10
number of dumps per bank ..... 2_
number of dumps per sound/patch ... 2_
convert patch before send ..... 0__

level: 1=bank, 2=patch, 3=paramtr   3
preconvert to 8/7 bitformat ..... 0__
maximum param's/bytes in editor ... 400_

EXIT                               INSTALL
  
```

Zodoende ontstaan in een dergelijk geval:

```

Twee 'BANK-RECEIVE-ADAPTORS'
Twee 'SOUND/PATCH RECEIVE ADAPTORS'
Twee 'SOUND/PATCH SEND ADAPTORS'
  
```

In de laatste Device-Adaptor-boxen (in dit geval de tweede) moeten steeds de gegevens van de DUMMY-BANK worden ingevuld.

Bank-Receive-Adaptor

In principe gelden dezelfde regels als bij het maken van een DUMMY-BANK welke aan de sounddata **vooraf** gaat. Voorwaarde is, dat de DUMMY-BANK-LENGTH niet '0' (of kleiner dan '0') mag zijn en dat de 'OFFSET TO FIRST SOUND/PATCH DATA-BYTE' in de bank receive-adaptor gelijk is aan de lengte van de eerste bank (of de gezamenlijke lengte van de banken) die er aan vooraf gaat (c.q. gingen).

EDIT DEVICE-ADAPTOR D78_PRFMNC	
BANK RECEIVE ADAPTOR (for level 1-6)	
request: FE	-----
offset to first sound-/patchdate	16034
dump length (including exclusive bytes)	768
sound/patch length (without bank-exclusive bytes)	12

Om te voorkomen dat UNI-MAN tijdens 'RECEIVE BANK' op data voor de tweede (dummy-) bank gaat staan wachten, moet als DUMMY-BANK-request 'FE' worden ingegeven (*zie hoofdstuk 6.2.11 van de oorspronkelijke handleiding*).

Omdat de oorspronkelijke banklengte 16034 Kbyte is, geldt voor de tweede (dummy-) bank een offset van 16034. Het write-request omvat 13 bytes, waarvan de laatste (F7) door UNI-MAN zelf wordt gegenereerd. Zodoende is de lengte per data-string (van het write-request) slechts '12' bytes. Als DUMP-LENGTH geldt hier $64 \times 12 = 768$, terwijl de 'sound/patch-length' dus uiteraard '12' is.

Het 'dump-follow-number' van de DUMMYBANK is steeds '2', omdat deze (in dit geval) immers **achter** de oorspronkelijke exclusive-bank komt.

Sound/Patch-Receive-Adaptor

Ook voor de tweede 'sound/patch receive adaptor' geldt dat op de plaats van het request 'FE' moet worden ingevoerd. De tweede bank doet dan niet mee aan het ontvangen van sound/patch-data.

SOUND/PATCH RECEIVE-ADAPTOR (for level 2-6)	
request: FE	-----
dump-follow-number in the bank	2
offset to first sound/patch date-byte	0
sound/patch data-length (unconverted without exclusives)	12

Omdat de DUMMY-BANK geen request kent wordt achter OFFSET TO FIRST SOUND/PATCHDATA een '0' ingevuld.

De dump lengte is weer $64 \times 12 = 768$.

Als 'sound/patch-data-length' geldt weer '12' bytes.

Sound/Patch-Send-Adaptor

Bij de tweede 'sound/patch send adaptor' **moet** op de plaats van het request **00** worden ingevoerd, om UNI-MAN te vertellen dat er **geen** header is. Althans, voor de hier gekozen (makkelijke) oplossing.

SOUND/PATCH SEND ADAPTOR (for level 2-6)	
header: 00	_____
dump-follow-number in the bank	2
converting-table follow-number	2
checksum precount (pos, or neg, offset)	0
number of headerbytes to add -> checksum	0
program-change (0=no, 1=before, 2=after)	2
program-change channel	16
program-change sound-/patchnumber	1
send-message for data-upgrade by device (0=off, 1=on) ..	0
SOUND/PATCH SEND ADAPTOR: 2	
OKAY	

Een betere (moeilijkere) oplossing, die echter veel universeler is en gebruik maakt van comfortabele UNI-MAN-features, wordt iets verderop omschreven (*zie NOTE*). Omdat hier geen 'FE' voorkomt zullen de data van deze bank echter **wel** mee worden gestuurd. Op die manier wordt bereikt dat deze extra data **niet** met de bank en **wel** met de sounds worden verstuurd. Om een korte pause in te lassen, kan eventueel eerst nog de aanduiding 'FD' worden geplaatst. Dat is overigens alleen gewenst als het MIDI-apparaat dat verlangt. In alle andere gevallen betekent één of meer 'FD's' onnodig oponthoud.

De overige gegevens komen weer overeen met de 'SOUND/PATCH RECEIVE ADAPTOR'.

Vaak wordt er in UNI-MAN gewerkt met een zelf gedefinieerd edit-buffer. Voor het gemak kan dat bv soundplaats nummer '1' zijn. Dit is vooral handig bij MIDI-devices welke niet over een 'TEMPORARY- (EDIT-) BUFFER' beschikken. In dit geval zou het betekenen dat alle write-request-data precies hetzelfde kunnen zijn. Alle sounds worden dan nl. naar hetzelfde adres gestuurd. Voor degene die een DEVICE-ADAPTOR willen perfectioneren is het zeer goed mogelijk om, zoals we verderop zullen zien, elke sound/patch naar zijn eigen adres te sturen. De sound/patch-data staan dan meteen op de juiste plaats (in het apparaat).

NOTE Een betere DUMMY-SEND-ADAPTOR, waarmee aan elk gewenst adres kan worden gestuurd en steeds (automatisch) is voorzien van het juiste checksum over het request, ziet er als volgt uit:

Op de plaats van de HEADER (waar nu '00' staat) kan het onveranderlijke deel (waarover bovendien geen checksum nodig is) worden ingevoerd. In het voorbeeld van de D70 performances is dat: 'F0 41 10 39 12'. De DUMMY-BANK bestaat dan uit 64 x het 'address' MSB en LSB, gevolgd door de 'size' MSB en LSB. Omdat in ons voorbeeld voor alle performances hetzelfde adres werd gekozen is dat steeds: '04 58 6A 00 00 01'.

Door in de tweede 'bank-receive-adaptor' de voor Roland juiste checksum-mode te installeren ('2') zal UNI-MAN automatisch bij ieder te verzenden performance een checksum genereren en met 'F7' afsluiten. Dit systeem zal ook korrekt werken als voor elke performance een ander adres werd ingegeven.

De Dummy-Bank zelf

Om o.a. zelf (zonder programmeertechnische kunsten) een DUMMY-BANK te kunnen samenstellen wordt er bij de UNI-MAN TOOL-KIT het hulpprogramma 'UNI-CON' geleverd. In bovenstaand voorbeeld bestaat de DUMMY-BANK uit 64 stuks write-requests á 12 c.q. 6 bytes. Voor het gemak gaan we uit van een DUMMY-BANK die alle data van het write-request bevat (12 bytes). In eerste instantie moeten deze 64 x 12 = 768 bytes achter elkaar worden geplaatst en naar disk geschreven. Vervolgens moet deze file dan **achter** de eigenlijke bank worden 'geplakt'.

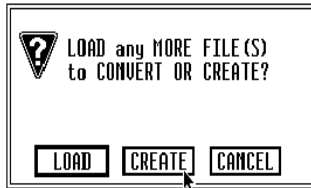
Deze bewerking hoeft slechts één keer te gebeuren. UNI-MAN onthoudt nl. deze data (achter de INIT-BANK) en schrijft ze met elke bank mee naar disk.

Het is ook mogelijk om de hele manipulatie in één keer te doen. Er is dan geen enkele tussenhandeling nodig. Hier kan de gebruiker zijn/haar eigen voorkeur laten gelden. We zullen beide mogelijkheden bespreken.

UNI-CON 3.0

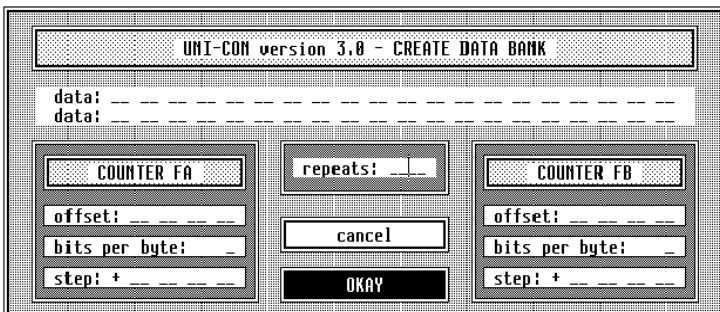
Start het programma UNI-CON op. Denk er om, dat het versienummer van UNI-CON 3.0 of hoger is. In vroegere versies is de hier te bespreken mogelijkheid namelijk niet voorhanden. Kies voor de optie 'SPECIALS'. Zoals te verwachten verschijnt er een item-selector. Hierin kan worden gekozen voor het inladen van een bank die aan de DUMMY-BANK vooraf gaat, of voor het aanmaken van de DUMMY-BANK. In ons geval gaat het om de INIT-BANK zoals de D70 die heeft afgelegd. Om alles in één sessie te doen klikt men de D70 bank aan en vervolgens 'OKAY'. De bank wordt dan ingeladen.

In het andere geval kiest men voor 'cancel'. In beide gevallen volgt dezelfde vraag: "LOAD any MORE FILE(S) to CONVERT or CREATE?".



Kies nu voor 'CREATE'. In de box die dan verschijnt (zie figuur 'CREATE DATA BANK') kan een DUMMY-BANK worden aangemaakt. Bijgaande figuur doet een zekere moeilijkheidsgraad verwachten. Het tegendeel is echter waar. Er is met opzet voor deze oplossing gekozen, omdat de gegevens van de MIDI-implementatie zonder meer hierin 'passen'. Om te voorkomen dat de maker van de Device-Adaptor moeizaam alle formaat-gegevens over meer dan één byte heen en terug moet rekenen, kunnen de waarden in hun oorspronkelijke gedaante blijven bestaan. Bedenk bovendien, dat we nog steeds praten over het maken van een Device-Adaptor! Het hier besprokene heeft dus niets te maken met het gebruik van UNI-MAN als MANAGER en/of EDITOR e.d.

In de bovenste twee regels kunnen 40 bytes tot een 'data-string' worden samengesteld. Doorgaans zullen hier de data worden ingegeven die, zoals in ons geval, een write-request vormen. Het kan echter ook een 'MODE-omschakeling', programchange achteraf, terugstelling naar uitgangssituatie, een display-update-bevel, of iets dergelijks zijn.



Op de plaats waar een 'counter' (teller-byte) in de opgebouwde data-string nodig is, dient 'FA' en/of 'FB' te worden ingevoerd. Links en rechts onder kunnen deze counters worden geïnstalleerd. Die counters werken volledig onafhankelijk van elkaar. Doorgaans zal één byte per counter voldoende zijn. Er zijn ook situaties denkbaar waarin twee, drie en soms zelfs vier bytes per counter noodzakelijk blijken. Dat laatste kan voorkomen bij het veranderen van een adres wat groter is dan de waarde die in één, twee of drie bytes kan worden ondergebracht. In het voorbeeld (dat in veel gevallen opgaat) hebben we steeds hetzelfde adres, zodat op de plaats van het adres geen counter nodig is. Ook elders is er geen counter nodig, zodat (in dit geval) alleen de data-string en het aantal herhalingen moet worden ingegeven. Naast de count-bytes 'FA' en 'FB' zijn er nog een viertal verschillende tellers beschikbaar: 'FC', 'FD', 'FE' en 'FF'. Ze vertegenwoordigen een waarde van **1 byte** en tellen automatisch op of af. Ze zijn bedoeld voor het **initialiseren** van MIDI-data-waarden binnen een exclusive-string. Vandaar vertegenwoordigen deze count-bytes de volgende waarden:

FC: van 127 naar 1
FD: van 127 naar 0
FE: van 1 naar 127
FF: van 0 naar 127

Elk van deze tellers kan (binnen een zelfde exclusive-string) tot vier keer worden gebruikt.

Het werken **met** counter is iets moeilijker, maar zoals gezegd kunnen de gegevens rechtstreeks uit de system exclusive implementatie (of gebruiksaanwijzing) van het MIDI-device worden overgenomen.

Per counter kan worden bepaald in welk formaat de teller dient te lopen. Vaak is dat in '7 bit Hexadecimaal'. Een eveneens veel voorkomend data-formaat is de 'nibble' (4 bits per byte). Dit data-formaat wordt doorgaans steeds in de MIDI-implementatie vermeld. Bovendien komen deze begrippen in de originele UNI-MAN handleiding uitvoerig aan de orde (**zie hoofdstuk 6.3 en 6.5**).

De grootte van de verandering per stap kan (geheel beneden) in eveneens maximaal vier bytes worden vastgelegd. Belangrijk is, dat de hier in te vullen waarde overeen dient te komen met het ingestelde formaat (**zie ook handleiding UNI-MAN hoofdstuk 6.3 en 6.5**). Rest nog om van elke teller de polariteit (+ of -) vast te leggen. De polariteit bepaalt uiteraard of de counter respectievelijk op- of aftelt. Ter plaatse aanklikken met de muis polariseert de variabele.

Geheel midden in beeld kan tenslotte worden vastgelegd hoe vaak deze string achtereenvolgens moet voorkomen. In de meeste gevallen komt dat overeen met het aantal sounds/patches. Zo ook nu.

In het geval er eerst een bank werd ingeladen, zal de gecreëerde 'DUMMY-BANK' naast de reeds aanwezige bank worden getoond. In het hoofdscherm verschijnen dan twee files die op de bekende wijze aan elkaar worden 'gehangen'. In het andere geval zal alleen de nu samengestelde data-bank in het geheugen zijn en in het hoofdscherm als 'CREATED' worden aangeduid. In dat laatste geval is het nodig om (na ze naar disk te hebben weggeschreven), in een tweede sessie, de twee banken tot één 'INIT-BANK' samen te voegen. Het is uiteraard ook mogelijk om de 'SYSTEM-EXCLUSIVE-BANK' in te laden **nadat** de 'DUMMY-BANK' werd gecreëerd. Let dan op de juiste volgorde (*zie hiervoor hoofdstuk 76*).

NOTE: In tegenstelling tot eerdere versies is het vanaf UNI-CON versie 3.0 mogelijk om met 'BYTE-STRINGS' te werken die kleiner zijn dan 17 bytes.

UNI-MAN zal er voor zorgdragen dat, tijdens het naar diskette schrijven van één enkele sound, het bijbehorende write-request wordt meegeschreven. Dat kan belangrijk zijn voor het gebruik van UNI-DUMP. Op die manier kunnen sounds/patches worden geprepareerd voor een bepaalde plaats in de bank. Indien dat niet wenselijk is, kan voor een andere werkwijze worden gekozen (*zie verderop bij behandeling van de bank*).

Om te voorkomen dat de sound/patch-adressen van de DUMMY-BANK worden overschreven, laat UNI-MAN (tijdens het inladen van een sound/patch) deze extra data buiten beschouwing. Zodoende blijft de adressering vanuit UNI-MAN steeds korrekt.

Als extra voorbeeld zijn de gegevens gekozen, zoals die gelden voor een DUMMY-BANK t.b.v. het write-request van de RA-50 (*zie figuur WRITE PATCHES RA-50*). Dat is een realtime-arranger van de firma ROLAND, welke is voorzien van een vrij uitgebreide MIDI-implementatie. De variabele 'FA' wordt hier ingezet om het patch-nummer op te tellen, terwijl 'FB' het 'CHECKSUM' over het request voorstelt.

UNI-CON version 3.0 - CREATE DATA BANK

data: F0 41 10 39 12 04 58 6A 00 00 01 39 ---
 data: -----

COUNTER FA	repeats: 64	COUNTER FB
offset: -----	cancel	offset: -----
bits per byte: --	OKAY	bits per byte: --
step: + -----		step: + -----

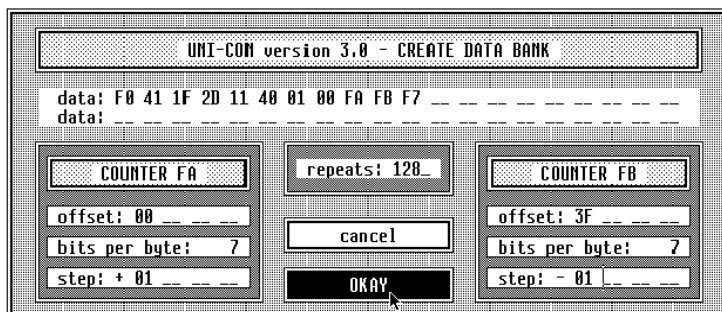
Omdat het patch-nummer steeds met '1' wordt opgehoogd, dient het checksum steeds met '1' te worden verminderd. Zodoende blijft de som (checksum) konstant. De checksum-waarde '3F' is de complementaire waarde naar 128 (80H) over het adres '40 01 00' en het patchnummer 'FA'. Ook deze gegevens werden direct aan de betreffende handleiding ontleend.

Voor sommige devices kan er een probleem ontstaan bij het gebruik van UNI-DUMP (in combinatie met een 'DUMMY-BANK' **achteraf**). Omdat UNI-MAN alle op deze bank betrekking hebbende data in één file bewaart, kan het voorkomen dat de data niet in de (voor het verzenden van een bank) gewenste volgorde staan. UNI-DUMP stuurt weliswaar geen data die niet behoren tot de exclusieve standaard, echter alle write-requests in één bank **kan** toch verkeerd uitpakken. Sommige MIDI-devices nl. vertonen de neiging om alle sound/patch-geheugenplaatsen steeds met dezelfde (laatste) sound/patch te vullen. Mocht dit probleem zich voordoen, dan is dat op verschillende manieren op te lossen:

Oplossing 1: Haal de Device-Adaptor (met bijbehorende INIT-BANK) door UNI-LINK. Dit programma ontdoet de bank van alle DUMMY-BANKEN. De INIT-BANK die dan wordt afgelegd kan moeiteloos (met UNI-DUMP) worden verstuurd.

Oplossing 2: Maak een Device-Adaptor welke per sound werkt. Daarbij komt achter elke sound/patch een write-request. Een dergelijke bank zal onder alle omstandigheden worden geaccepteerd. Ook deze bank hoeft, op de hierboven omschreven wijze, slechts eenmalig als INIT-BANK te worden aangemaakt (zie EX8000 en SPX900/1000). Dit is een heel degelijke oplossing.

Oplossing 3: Schrijf de bank vanuit UNI-MAN naar disk en verwijder daarna de DUMMY-BANK met behulp van UNI-CON.



DIVERSE UPDATES 2

Individual Parameter Handling

Een fikse uitbreiding/aanpassing heeft UNI-MAN in staat gesteld om niet alleen per bank of sound/patch te kunnen handelen, maar ook **per parameter**. Dat betekent universele REAL-TIME faciliteiten. Ook apparaten die uitsluitend via exclusive-strings per parameter editeerbaar zijn, vallen nu binnen het bereik van UNI-MAN.

'JUMP TO' Fixed Parameter

Belangrijkste aanvulling in de editor is een door velen gevraagde memory-functie voor bepaalde parameters. Het komt vaak voor dat sommige parameters alleen zinvol kunnen worden geëditeerd in samenhang met andere parameters. Als deze parameters niet dicht bij elkaar staan vraagt het enige tijd en ongemak om steeds de andere op te zoeken.

Voorbeeld: Een ROLAND D50 geluid bestaat uit één COMMON en twee PARTIALS voor UPPER en LOWER. Om een bepaald partial goed te kunnen beluisteren moeten de overige partials tijdelijk worden uitgeschakeld. Met de partial-mute functie (COMMON-parameter) kunnen partials worden aan- of uitgeschakeld. In dit geval is het dus wenselijk om tijdelijk de partial-mute functie onder een 'JUMP TO' button te hebben en wel zodanig dat deze parameter precies in de middelste regel staat. Pas dan is het mogelijk om, direct na de 'jump', de muis of met de [+] en [-] toetsen de gewenste mute-instelling te kiezen om vervolgens weer terug te springen.

Welnu, door met de **rechter** muisknop de button 'JUMP TO' aan te klikken fixeert men de (middelste) parameter waar naar toe wordt gesprongen, als deze button later met de **linker** muistoets wordt bediend. U zult begrijpen dat deze nieuwe functie (vooral bij intensief gebruik) een stuk comfort betekent.

Aanpassing Edit-Formulas

Een ander aspect van de samenhang tussen de diverse parameters is de mogelijkheid om middels formules (*zie paragraaf 6.5 'EDIT FORMULAS' van de handleiding*) terug te grijpen op vóórliggende parameterwaarden. Vanaf de versie 3.0c is het namelijk mogelijk om achter 'READ BYTE' een negatieve waarde in te geven en de daar gevonden waarde in de actuele parameter te verdisconteren. Zo is het ook mogelijk om met [SHIFT] + 'READ BYTE' de elders gevonden waarde te sommeren met de actuele waarde. De formule noteert dit b.v. als: +Rb-22 (tel hierbij op de waarde welke twee en twintig bytes terug wordt gevonden).