



Il protocollo MIDI

C'era una volta un sequencer MIDI di nome Cubase...

Cubase nacque come sequencer MIDI *puro* e, in breve, divenne uno dei software di riferimento nel panorama musicale internazionale. Sono passati molti anni, e anche se oggi viene utilizzato principalmente per effettuare registrazioni o trattamenti audio, rimane uno dei software più potenti per la registrazione, l'editing e il trattamento dei dati MIDI.

Se questo manuale fosse stato scritto dieci anni fa, probabilmente non ci sarebbe stato bisogno di dedicare una sezione alla descrizione del protocollo MIDI, ma l'attuale sbilanciamento delle conoscenze a favore dell'audio comporta la necessità di un approfondimento sul protocollo di interfacciamento MIDI. Ovviamente, gli utenti esperti possono saltare la sezione e proseguire oltre, anche se, a parere nostro, un ripassino non fa mai male.

Perché il MIDI

Il MIDI (*Musical Instrument Digital Interface*) costituisce un insieme di messaggi e di regole che definiscono un protocollo di comunicazione tra strumenti tipicamente musicali e non solo. Il MIDI venne concepito per venire incontro alle esigenze pratiche dei poveri tastieristi che, fino agli inizi degli anni '80, erano circondati da castelli di synth di dimensioni tali da richiedere doti quasi circensi per essere suonati contemporaneamente. La tecnologia digitale nella musica era agli albori e i sintetizzatori offrivano poche note di polifonia e pochi preset monotimbrici: per poter suonare accordi ricchi di note e costituiti da più timbri sovrapposti, occorreva suonare più sintetizzatori simultaneamente. Agli inizi degli anni '80, qualcuno cominciò a domandarsi se non fosse possibile controllare la polifonia e la timbricità dei sintetizzatori in maniera più funzionale.

Yamaha, Sequential Circuits e Roland, tre grandi case costruttrici di sintetizzatori, sfruttando le possibilità dell'elettronica digitale, collaborarono alla costruzione di un linguaggio comune di comunicazione, presentato ufficialmente nell'autunno del 1982.

Nonostante svariati tentativi di cambiamento, la struttura del protocollo si presenta ancora oggi pressoché invariata, e questo per parecchi buoni motivi: è estremamente semplice da programmare, incredibilmente potente, non richiede grandi risorse tecnologiche ed è ormai ampiamente diffuso a livello planetario.

Struttura e connessioni

I messaggi MIDI sono costituiti da parole (*word*) formate da uno o più byte che trasferiscono le intenzioni tecniche ed espressive musicali dell'esecutore da uno strumento musicale, definito *master*, a un altro, chiamato *slave*, che offre il supporto timbrico.

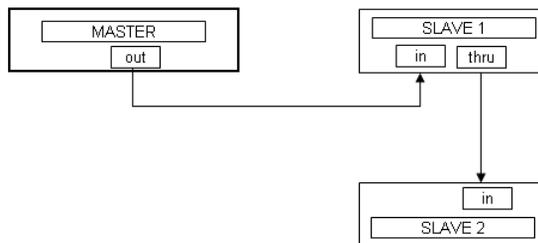
I messaggi MIDI riguardano tipicamente azioni proprie della pratica esecutiva, per esempio: "premi il pedale del volume", "suona una nota della tastiera", "aggiungi un vibrato muovendo la manopola della modulation wheel" e così via. Questi comandi, tradotti in messaggi digitali MIDI, viaggiano dal master allo slave, dove vengono interpretati e associati a un timbro. È importante capire che il messaggio MIDI non è un formato audio, ovvero non contiene il suono richiesto nel master, ma riguarda solamente le azioni del musicista; sarà poi lo slave a fornire il timbro.

I messaggi MIDI sono valori discreti compresi tra 0 e +5 volt, e sono quindi costituiti da sequenze di 0 e di 1 organizzate in byte.

Gli strumenti musicali elettronici, i mixer digitali e i processori di effetti, i controller, i sistemi DAW e i computer, per comunicare tra loro tramite protocollo MIDI devono essere provvisti di un'interfaccia adeguata, ovvero dell'hardware dotato di porte MIDI che consentano la trasmissione e la ricezione dei vari messaggi. L'interfaccia è costituita da tre tipologie di connessione e può variare da strumento a strumento:

1. la connessione MIDI IN serve per ricevere i messaggi inviati da un'altra macchina;
2. la connessione MIDI OUT è usata per inviare i messaggi ad altri strumenti;
3. la connessione MIDI THRU consente di reindirizzare un messaggio proveniente alla porta IN verso un altro strumento.

Nel caso avessimo un master e due slave, il collegamento MIDI andrebbe effettuato secondo lo schema riportato nella figura seguente.



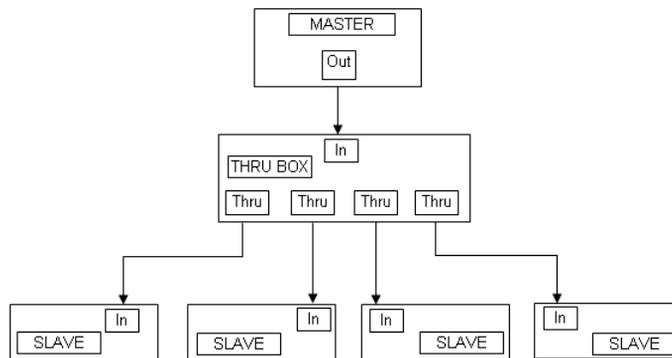
Pertanto, se inviamo un messaggio MIDI in uscita dalla master, questo sarà ricevuto da entrambi gli slave.

Supponiamo però di avere messaggi che devono essere interpretati solamente dallo Slave 1 e altri dallo Slave 2; come possiamo differenziare i messaggi? Semplice: attribuendo loro un numero di *canale* indipendente. In questo modo, possiamo indirizzare qualunque messaggio su uno dei 16 canali disponibili previsti dal protocollo; basta impostare per ciascuno degli Slave un numero di canale diverso (da 1 a 16) e il gioco è fatto. Impostiamo lo Slave 1 in ricezione sul canale MIDI 2 e lo Slave 2 sul canale 6; quando inviamo dalla master un messaggio MIDI sul canale 6, questo sarà interpretato dallo Slave 2 e ignorato dallo Slave 1.

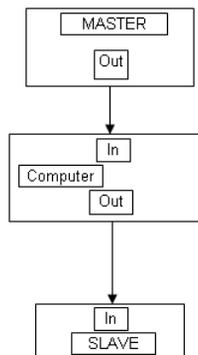
Il protocollo MIDI

Il collegamento illustrato nell'immagine precedente è detto *in cascata* o *seriale* ed è sconsigliato se si devono collegare numerosi dispositivi slave, in quanto potrebbero sorgere dei problemi e soprattutto delle latenze. La trasmissione dei dati MIDI avviene alla velocità di 31.250 baud al secondo, in maniera seriale e asincrona (ovvero occorrono in più un bit di start e uno di stop). Se si hanno numerosi slave, sarebbe meglio utilizzare delle *thru box*, ovvero interfacce dotate di una porta IN e più porte THRU.

In tal caso, i collegamenti vanno effettuati come nella figura seguente.



Utilizzando un computer, e quindi un sequencer, in grado di registrare e inviare dei messaggi MIDI, il suo inserimento nella catena MIDI avverrà tra il master e i vari slave.

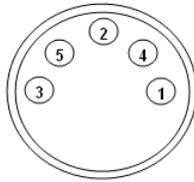


Una caratteristica importante: la porta MIDI OUT presente sul computer si comporta anche come porta THRU "virtuale" (l'opzione può essere nominata *patch thru*, *echo thru*, *thru on/off*) e, grazie a questa, è possibile controllare dalla master lo slave passando *attraverso* il sequencer.

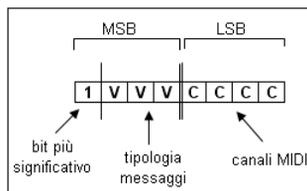
Da qualche anno, esistono inoltre dei Virtual Instrument, programmi residenti nel computer, ovvero degli "slave virtuali" i cui collegamenti MIDI sono di natura virtuale (logica) e che quindi vengono gestiti direttamente all'interno del sequencer senza necessità di cablaggi hardware.

Messaggi

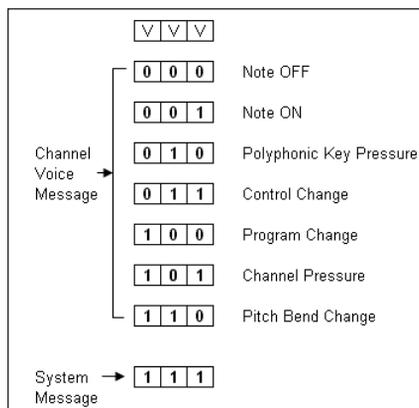
I messaggi MIDI vengono inviati su cavi a 5 pin di cui ne vengono utilizzati solamente 3.



Il pin 2 è collegato a massa in modo da evitare possibili interferenze. Il pin 5 è utilizzato per la trasmissione dei dati, mentre il pin 4 porta il cavo che fornisce l'alimentazione a +5V. Il messaggio MIDI è costituito da una parola (word) costituita da diversi byte divisi in due tipologie: lo *status byte* e il *data byte*. Lo status byte oltre a definire il comando MIDI e quindi rappresenta il tipo di informazione (nota suonata, volume, pitch bend e così via), indica il numero di canale su cui viaggerà. Il data byte rappresenta invece il valore richiesto per il parametro specificato nello status byte. Come è possibile distinguere lo status dal data byte? Tutti gli status hanno il bit più significativo pari a 1, i data pari a 0; per di più, un messaggio deve sempre iniziare con uno status byte. Questa è la struttura di uno status byte.



Come possiamo vedere, è diviso in due *nibble* (gruppo di 4 bit) chiamati MSB (*Most Significant Bit*) e LSB (*Least Significant Bit*). Nel primo è presente il bit più significativo, utilizzato per il riconoscimento dello status byte, mentre i tre bit rimanenti, indicati con la lettera V, sono utilizzati per specificare i vari messaggi. Di seguito, sono riportate le otto combinazioni possibili.



Il protocollo MIDI

I messaggi MIDI si dividono in due grandi categorie: *Channel Voice Message* (messaggi di canale) e *System Message* (messaggi di sistema). I primi possono essere indirizzati su uno dei 16 canali MIDI e sono divisi in *Voice Message* e *Mode Message*.

I secondi non contengono informazioni di canale ma sono indirizzati globalmente all'intero *sistema* di strumenti collegati nel nostro network musicale e si dividono in *Common Message*, *Real Time Message* ed *Exclusive Message*.

Il secondo nibble (LSB), quando è utilizzato nei messaggi di canale, serve a indicare il canale MIDI. Avendo a disposizione 4 bit, possiamo disporre di 16 combinazioni e pertanto di altrettanti canali.

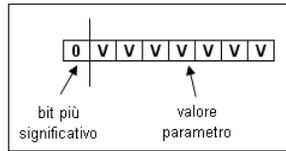
C	C	C	C	
0	0	0	0	canale 1
0	0	0	1	canale 2
0	0	1	0	canale 3
0	0	1	1	canale 4
0	1	0	0	canale 5
0	1	0	1	canale 6
0	1	1	0	canale 7
0	1	1	1	canale 8
1	0	0	0	canale 9
1	0	0	1	canale 10
1	0	1	0	canale 11
1	0	1	1	canale 12
1	1	0	0	canale 13
1	1	0	1	canale 14
1	1	1	0	canale 15
1	1	1	1	canale 16

Quando abbiamo a che fare con i messaggi di sistema, l'LSB si riferisce a 11 tipologie di messaggi (cinque combinazioni non sono ancora utilizzate), come possiamo osservare nell'immagine successiva.

C	C	C	C	
0	0	0	0	System Exclusive Start
0	0	0	1	MTC Quarter frame
0	0	1	0	Song Position Pointer
0	0	1	1	Song Select
0	1	0	0	
0	1	0	1	
0	1	1	0	Tune Request
0	1	1	1	System Exclusive Stop
1	0	0	0	MIDI Clock
1	0	0	1	
1	0	1	0	Start
1	0	1	1	Continue
1	1	0	0	Stop
1	1	0	1	
1	1	1	0	Active Sensing
1	1	1	1	System Reset

La struttura di un *data byte*, invece, è la seguente:

Appendice A

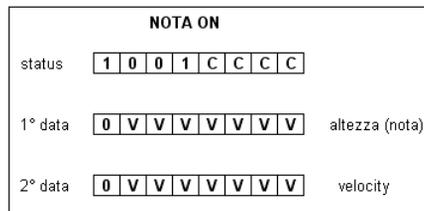


In questo caso, sono a disposizione sette bit e quindi 128 possibili valori a seconda dei casi; questa è una delle ambiguità lasciate dalla struttura di protocollo esprimibile come intervallo 1-128 e altre volte come 0-127.

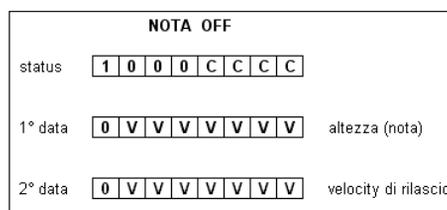
I messaggi MIDI fondamentali

Vediamo le varie tipologie di messaggi MIDI completi (Status e Data).

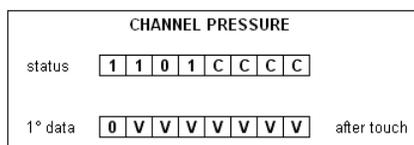
NOTA ON. Se si preme un tasto della tastiera della master, viene inviato un messaggio di Nota On che presenta due data byte; il primo è relativo all'altezza della nota e il secondo alla *velocity*, ovvero alla forza con cui è stato premuto il tasto.



NOTA OFF. Se si rilascia il tasto di nota premuto in precedenza, viene generato questo messaggio costituito anch'esso da due data byte; il primo specifica quale nota viene rilasciata e il secondo con quale *velocity*, cioè intensità di rilascio del tasto.

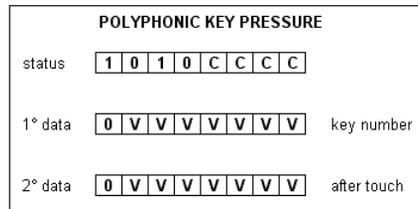


CHANNEL PRESSURE. Trasmette in tempo reale eventuali variazioni di pressione esercitate su di una nota premuta in precedenza. Questa pressione, unica per tutta la tastiera, è genericamente chiamata *after touch*, e di solito a essa viene assegnata una funzione che modifica il suono (la modulazione, l'intonazione, il volume e così via). Non ha il secondo data byte.

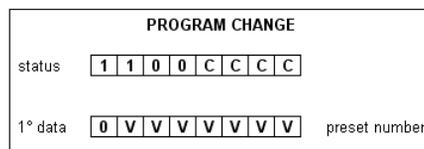


Il protocollo MIDI

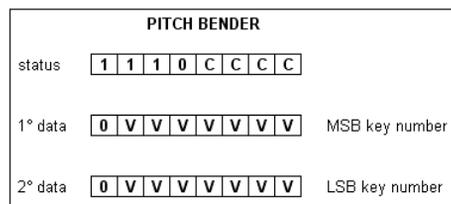
POLYPHONIC KEY PRESSURE. È simile al precedente ma si riferisce al più costoso, e quindi molto più raro, utilizzo di sensori di pressione indipendenti per ciascuna nota. Il primo data è riferito alla nota e il secondo al valore di after touch.



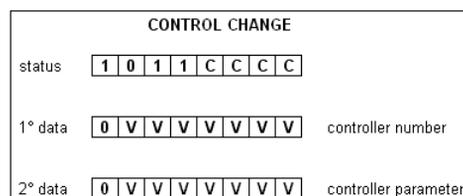
PROGRAM CHANGE. Viene utilizzato per richiamare il contenuto di una locazione di memoria nello strumento slave; in questo modo l'apparecchio può passare da un timbro all'altro. Non ha il secondo data byte.



PITCH BENDER. Le tastiere master sono provviste di slider o joystick con i quali è possibile variare l'altezza di una nota in esecuzione. Occorrono due data byte per definire i valori di incremento o diminuzione dell'altezza sufficientemente fluidi e privi di *zippering* all'ascolto. I due data byte forniscono in totale 16.384 valori possibili.



CONTROL CHANGE. Questo messaggio in realtà può comprendere 128 diverse variazioni di messaggio che si riferiscono alle informazioni di espressione, e che sono appunto chiamati *control change*. Il primo data indica il numero del controller (tra i 128 disponibili), mentre il secondo ne specifica il valore.



Appendice A

Alcuni cc non sono stati definiti e la loro attribuzione è genericamente rispettata; ciò non esclude che, in determinate situazioni, costruttori diversi possano liberamente *appropriarsi* di un codice control change per soddisfare determinate esigenze.

0	Bank Select	45	Effect control 2
1	Modulation wheel	46..47	Non definito
2	Breath control	48...51	General #1,2,3,4
3	Non definito	52...63	Non definito
4	Foot control	64	Dumper pedal
5	Portamento time	65	Portamento on/off
6	Data Entry	66	Sostenuto on/off
7	Channel Volume	67	Soft pedal on /off
8	Balance	68	Legato Footswitch
9	Non definito	69	Hold 2
10	Pan	70...79	Sound Controller 1-10
11	Expression Controller	80...83	General Purpose 5,6,7,8
12	Effect control 1	84	Portamento control
13	Effect control 2	85...90	Non definito
14...15	Non definito	91...95	Effects 1,2,3,4,5 Depth
16...19	General #1,2,3,4	96	Data entry +1
20...31	Non definito	97	Data entry -1
32	Bank Select	98	NRPN LSB
33	Modulation wheel	99	NRPN MSB
34	Breath control	1000	RPN LSB
35	Non definito	101	RPN MSB
36	Foot control	102...119	Non definito
37	Portamento time	120	All Sound Off
38	Data Entry	121	Reset All Controllers
39	Channel Volume	122	Local control on/off
40	Balance	123	All note off
41	Non definito	124	Omni mode off
42	Pan	125	Omni mode on
43	Expression	126	Poly mode on/off
44	Effect control 1	127	Poly mode on

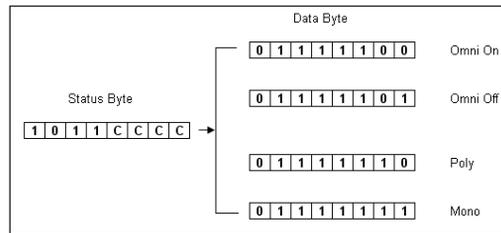
Gli ultimi quattro control change si riferiscono ai *Channel Mode Message*, che stabiliscono possibili configurazioni di funzionamento per gli strumenti MIDI, ottenute in base alla combinazione di tre stati operativi.

1. *OMNI*: indica che lo strumento è in grado di rispondere a tutti i messaggi inviati indifferentemente su tutti e sedici i canali MIDI.
2. *POLY*: lo strumento gestisce le proprie voci rispettando richieste polifoniche tese alla realizzazione di accordi.
3. *MONO*: lo strumento lavora in monofonia.

I quattro modi disponibili combinando i tre stadi operativi sono i seguenti.

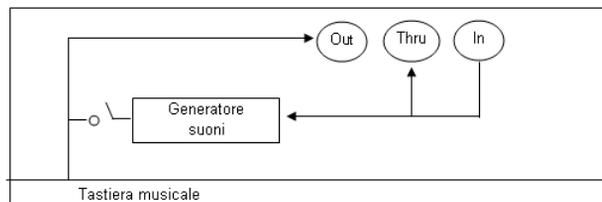
1. Modo 1 – *Omni On/Poly*: lo slave interpreta tutti i messaggi su tutti i canali e in maniera polifonica; è il cosiddetto modo operativo *a prova di idiota*: lo slave suona in ogni caso e in ogni condizione.
2. Modo 2 – *Omni On/Mono*: lo slave risponde su tutti i canali ma monofonicamente; chiaramente non è una condizione operativa molto funzionale.
3. Modo 3 – *Omni Off/Poly*: lo slave risponde in maniera polifonica solamente ai messaggi che giungono sul suo canale MIDI (specificato in ricezione). È detto sinteticamente *Poly*.
4. Modo 4 – *Omni Off/Mono*: lo slave risponde monofonicamente ai messaggi che giungono sul suo canale MIDI. È detto sinteticamente *Mono*.

Questi messaggi hanno solamente un data byte.

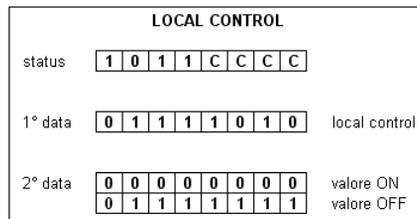


Local Control

Tra i diversi control change, ne esiste uno molto particolare chiamato *Local Control* che è utilizzato quando la master possiede al suo interno un modulo sonoro. In tal caso, per evitare che la tastiera musicale invii il messaggio di Nota ON anche al generatore di suoni interno, occorre impostare la master su Local OFF; conseguentemente, il messaggio MIDI sarà inviato solamente alla porta MIDI Out, come mostrato nella figura.



In questa modalità, il generatore interno di suoni si comporta esattamente come uno slave e pertanto potrà essere sfruttato rinviando da un sequencer il messaggio nella porta MIDI In della master. Ecco la struttura del messaggio.

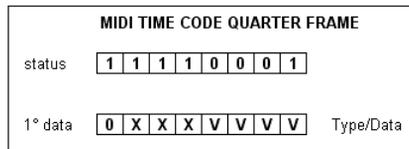


I System Message

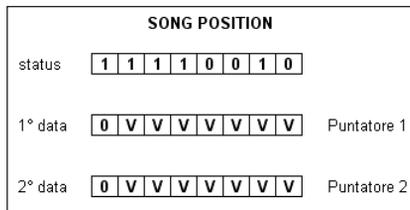
Si dividono, come abbiamo visto, in tre categorie.

1. *Common Message*: in totale quattro, danno istruzioni generali a tutto il sistema; sono:
 - *MIDI Time Code Quarter Frame* è un messaggio di time code nel formato ore:minuti:secondi:frame. Ogni otto messaggi di *MIDI Time Code* (MTC) si identifica una locazione del codice SMPTE.

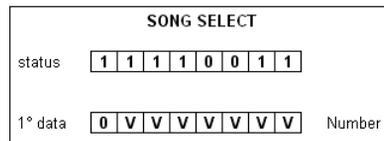
Appendice A



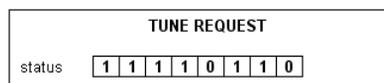
- *Song Position* indica la posizione che il cursore deve assumere durante la riproduzione di un file MIDI, assegnando un indirizzo univoco a ognuno dei beat calcolati dall'inizio della sequenza MIDI.



- *Song Select* consente di selezionare una song all'interno di un sequencer. È costituito da un solo data byte che indica il numero della song.

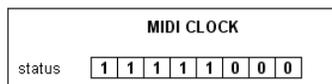


- *Tune Request* viene utilizzato per accordare lo strumento. È sprovvisto di data byte.



2. *Real Time*: sono 6 messaggi di sincronizzazione privi di data byte:

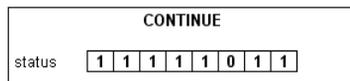
- *MIDI Clock* è utilizzato per sincronizzare strumenti MIDI; viene inviato 24 volte per ogni nota del valore di un quarto.



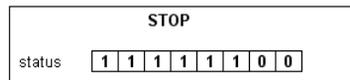
- *Start* viene utilizzato per portare tutti gli strumenti esterni alla posizione iniziale della song e a farle partire in riproduzione o registrazione.



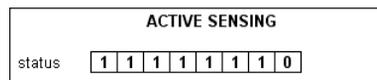
- *Continue* permette di riprendere la song dal punto in cui è stata interrotta.



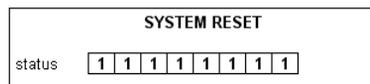
- *Stop* è utilizzato per fermare la registrazione o la riproduzione degli strumenti collegati al sequencer.



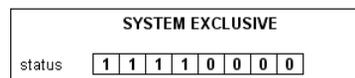
- *Active Sensing*, inviato ogni 300 millisecondi, è utilizzato per tenere attiva la connessione tra master e slave.



- *System Reset* ripristina le condizioni di default dello slave.

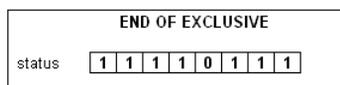


3. *System Exclusive (SysEx)*: sono messaggi personalizzabili strumento per strumento a seconda delle esigenze proprie degli sviluppatori; i codici di sistema esclusivo possono essere interpretati solo da categorie di apparecchi compatibili tra loro. Se lo strumento non è in grado di interpretare uno o più messaggi di sistema esclusivo, restituirà un messaggio di errore. Sono costituiti da una sequenza il cui inizio è rappresentato dal seguente messaggio.



Il messaggio di inizio è seguito da un byte contenente il Manufacturer ID, che identifica il costruttore. La sequenza può essere lunga o breve a seconda delle necessità e della complessità di quanto si trasmette; in ogni caso, viene sempre conclusa dal messaggio *end of exclusive*.

Appendice A



I parametri di una macchina che si possono modificare tramite il sistema esclusivo sono generalmente indicati nel manuale fornito con lo strumento, ma se non si possiede questa informazione, esistono programmi di utilità in grado di produrre la sequenza di sistema esclusivo che si ottiene collegando lo strumento e variando uno dei parametri.

Gli standard MIDI

Partiamo dal *General MIDI*; è stato introdotto nel 1991 per semplificare le operazioni di riproduzione delle sequenze MIDI mediante parametratura dei vari timbri. Successivamente, i timbri sono stati ordinati all'interno di una lista i cui numeri sono associati al messaggio di program change rispettando sempre la stessa posizione, a prescindere dalla marca dello strumento.

In questo modo, è molto semplice riprodurre una sequenza MIDI che necessita di timbri particolari: basta attivare il banco General MIDI e leggere la sequenza per caricare automaticamente i timbri (questo perché all'inizio si trovano i messaggi di program change). Inoltre, per convenzione, lo standard dedica il canale MIDI numero 10 ai suoni di batteria e percussione. Per esempio, lo strumento virtuale Hypersonic di Steinberg fornisce un ottimo banco General MIDI.

Successivamente, Roland creò lo standard *GS (General System)*, che permette di utilizzare un numero superiore di timbri rispetto ai 128 del General MIDI grazie all'introduzione di nuovi messaggi MIDI come il Bank Select (Control Change), che consente di selezionare i vari banchi dove sono memorizzati i timbri.

Nel 1994 Yamaha introdusse lo standard *XG*, in grado di controllare ulteriori banchi e strumenti, nonché drum kit ed effetti, e che è compatibile con il General MIDI.

MIDI File

Un *MIDI File* è una sequenza di dati MIDI costituiti da diversi messaggi distribuiti nei vari canali in grado di essere interpretata da qualsiasi strumento dotato di porte MIDI. Lo *standard MIDI File (SMF)* è stato realizzato nel 1986 da Opcode System; ha come estensione .mid ed è un file di testo codificato ASCII, e pertanto occupa poco spazio. Lo standard MIDI File è diviso in tre formati.

1. Formato 0: racchiude tutti i dati MIDI su un'unica traccia.
2. Formato 1: propone i vari dati MIDI su più tracce.
3. Formato 2: raro da incontrare ma molto potente in quanto consente di mantenere una certa indipendenza sui dati conservati nelle singole tracce.