

MIDIX30

MIDI Routing Matrix
with MIDI Recall™

Operating Manual

HINTON
INSTRUMENTS

Professional MIDI Management Systems

Contents

MIDIX SPECIFICATION	iii
1:MIDIX INSTALLATION	
<u>General/EMC Compliance</u>	1.1
<u>Earthing</u>	1.2
<u>MIDI Cables</u>	1.2
<u>Checking</u>	1.3
<u>Front Panel Controls</u>	1.4
<u>Rear Panel</u>	1.5
2:MIDIX OPERATION	
<u>Setting Up</u>	2.1
<u>Everyday Use</u>	2.3
<u>Advanced Use</u>	2.6
<u>VT100 Mode</u>	2.9
<u>Troubleshooting</u>	2.14
<u>MIDI Implementation Chart</u>	2.15
3:MIDIX COMMANDS	
<u>Summary</u>	3.1
<u>Single Character Commands</u>	3.4
<u>Double Character Commands</u>	3.6
<u>Compound Commands</u>	3.9
<u>MIDI Recall Commands</u>	3.20
<u>Optional Merger Commands</u>	3.22

©1992-2010 Hinton Instruments

MIDIX Operating Manual, Revision 3.0 (28 September 2010)

All Rights Reserved. Reproduction of any part of this manual in any form is forbidden. This documentation is provided solely to assist Hinton Instruments' clients achieving better usage of purchased equipment and additional copies will be provided free on request. All documentation, circuit diagrams, panel layouts, pcb layouts, software and screen layouts are copyright and remain Hinton Instruments' intellectual property. Disclosure to third parties is forbidden.

MIDIC, MIDIP, MIDIQ, MIDIX, MIDY, MIDZ, Z1, Z2, Z4, Long Haul Converter, MIDI Recall, VCA+ and the Hinton Instruments logo are Trademarks or Registered Trademarks of Hinton Instruments. All other manufacturers' trademarks are acknowledged.

MIDIX Specification

CASE

**3U 19" rack mounting Eurocard Case (W: 482 X H: 132 x D: 270 mm).
Anodised aluminium panels.**

POWER

85V-250V mains, 50/60Hz. 15W max.

FRONT PANEL

Patch Number Display
Input and Output MIDI Activity LEDs
Manual operation pushbuttons: Inc, Dec, Panel Lock, Panic, Hold.

REAR PANEL

MIDI Connections:
DB25S Multi-MIDI port = 5 MIDI ports per connector
6 x Matrix ports 1-30
2 x Merger ports 1-10

RS232 Control Ports: DB9 38k4Bd

Mains: IEC filtered inlet; 20 mm fuse holder

Earth: via Mains Inlet pin

MIDIX30 - MIDI ROUTING MATRIX

Maximum MIDI Inputs: 30
Maximum MIDI Outputs: 30
In to Out Delay: <200ns
Pulse Width Distortion: < 0.7%

MIDIX CONTROLLER WITH MIDI RECALL™

Software Version: 3.2 September 2010
Manual and RS232 control
Illegal Groups, Stop, Panic Clear and Clear Events in software
Hex and English MIDI Monitoring with Filtering
VT100 Paged displays
Number of Patches: 50
Number of Events: 50
Event List Size: 320 bytes
Chain Length: 50 steps

This specification and the information presented in this document may change without notice in the interests of continuing product improvement. While every effort is made to make this document accurate, Hinton Instruments cannot accept any responsibility for the interpretation of the information provided or any liability for any injury, loss, or damage, direct or consequential, caused by application or inability to use the equipment or information provided.

MIDIX Installation

GENERAL

The MIDIX MIDI Processing Rack takes 3U of rack space and requires a mains power supply with IEC connector on the left hand side. MIDIX will work on a mains voltage in the range of 85V-250V AC, 50 or 60 Hz.

The unit is cooled only by natural convection. If sited in a rack with vertical convection or fan cooling, spacing panels may be necessary to ensure that air flow is not obstructed.

Please take the normal precautions for siting an electronic instrument. Do not place on a source of direct heat, in direct sunlight, or near another instrument that may result in interference with either's operation.

The following statement is provided to comply with FCC regulations in the U.S.A., but the precautions and corrective measures are applicable worldwide:

WARNING: This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with this instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at their own expense.



MIDIX and its accessories have been tested to exceed the requirements of the above FCC rules, however this does not guarantee that interference will not occur, especially if connected to other devices that do not comply with these rules. If interference is experienced after installation of this equipment turn it off to determine whether it is the source. If interference is still experienced try to correct the situation by one or more of the following measures:

- **Relocate either MIDIX and its associated equipment and cabling or the equipment affected by the interference.**
- **Utilise different mains power outlets. MIDIX incorporates a mains line filter; try installing a filter in the mains of the other equipment.**
- **In the case of radio interference where 300 ohm ribbon antennas are used, try changing to 75 ohm coaxial cable.**

If for any reason you should need additional information relating to radio and TV interference, you may find a booklet prepared by the Federal Communications Commission helpful: "How to Identify and Resolve Radio/TV Interference Problems". This booklet is available from the U.S. Government Printing Office, Washington D.C. 20402, Stock #004-000-00345-4.



This product is fully compliant with the European EMC Directive 89/336/EEC which applies to apparatus which is "liable to cause electromagnetic disturbance or is itself liable to be affected by such disturbance".

Tests performed and passed are: EN50082-1 : 1992, EN55022 : 1994 Class B

This product may contain nuts.

EARTHING

The case is connected to mains earth and **must be grounded for safety**. The ground reference should be provided via the mains inlet. Do not remove the ground from the mains connector.

Do not provide a ground via the rack mounting bars. Use insulating mounting kits if this is the practise. For installations that have separate safety and technical earth systems please consult with Hinton Instruments before proceeding.

WARNING: Failure to observe this practise may result in hazard to life or damage to connected equipment.

After siting the unit, connect all permanent cabling.

MIDI CABLES

MIDIX has six Multi-MIDI port connectors on the rear panel. The MIDIY Mergers have an additional two ports. Each Multi-MIDI port has five MIDI In and Out Ports on a DB25 socket, the pin connections are given in the Appendix. Screened twisted pair cable should be used to break out to DIN MIDI connectors. The Out ground should be connected to the Out cable screen, the In cable screen should be taken up to the connector and sleeved so that it does not contact other screens.

Connect the MIDI Ins and Outs to all the MIDI equipment to be controlled. MIDIX may be inserted transparently in any MIDI connection scheme without any adverse delay effects. Use only quality cables intended for MIDI use and not hi-fi cables. Cable lengths should not exceed 15 metres. Longer cables, including special long run types, cannot be guaranteed to work in all circumstances as they have no control over the opto-isolator type used in the receiving equipment, its sensitivity, or its aging characteristic.

MIDI is an optically isolated system and the cable screens are connected to the driving circuitry ground, but not the receiver ground. Do not connect the cable screens to the plug shells as this will connect the case ground via the receptacle chassis and may cause grounding related problems.

RS232 CABLES

MIDIX has a DB9M connector conforming to the IBM PC AT pinout. There is an optional second DB9 for the MIDIY6 Merger which handles the additional merging.

The computer should be set to:

- 38k4Bd,
- 8 Data bits
- 2 Stop bits
- No Parity
- XON/XOFF handshaking

A “null modem” crossover cable is required.

Checking

When connected turn the power on to the MIDIX rack. The presence of internal power will be shown by the front panel numeric display lighting up. If not on check the mains fuse in the IEC inlet and that mains is present on the IEC cable. If the power supply fuse blows this should be changed and the power supply should be sent for repair. (Fuses in switch mode power supplies are blown if the supply develops a fault and if replaced may damage the supply and the circuitry it powers!)

If connected MIDI equipment is generating data the corresponding red Receive LEDs should flicker. Check that the three LEDs above the individual pushbuttons are off and that pressing the Panel Lock pushbutton turns it's LED on and off alternating. Leave in the off state.

Run your preferred terminal emulation software on your computer and set to VT100 emulation if it has it. If active when MIDIX is turned on a sign message should appear similar to:

```
MIDIX-30 Controller with MIDI Recall (TM)
Vers:3.2q (c) 1985-2010 Hinton Instruments

MIDIX Matrix Size:  30 Inputs into 30 Outputs
 50 Patches
 50 Event Lists of up to  320 MIDI bytes each
Maximum System Exclusive Block capture size: 16369 bytes
Buffer 1A Size: 16375
Buffer 1B Size: 16375
Buffer 1C Size: 16375
Buffer 1D Size: 16375

Block delay:  3
Running Status 1:  1
Active Sensing 1:  1
Memory Protect:  0
Front Panel enable:  1

Patch:  1:  title

If using a terminal with VT100 compatibility:
  first enable this mode and then type:
  T2 <ret>
```

The actual patch number and title will be the last one selected.

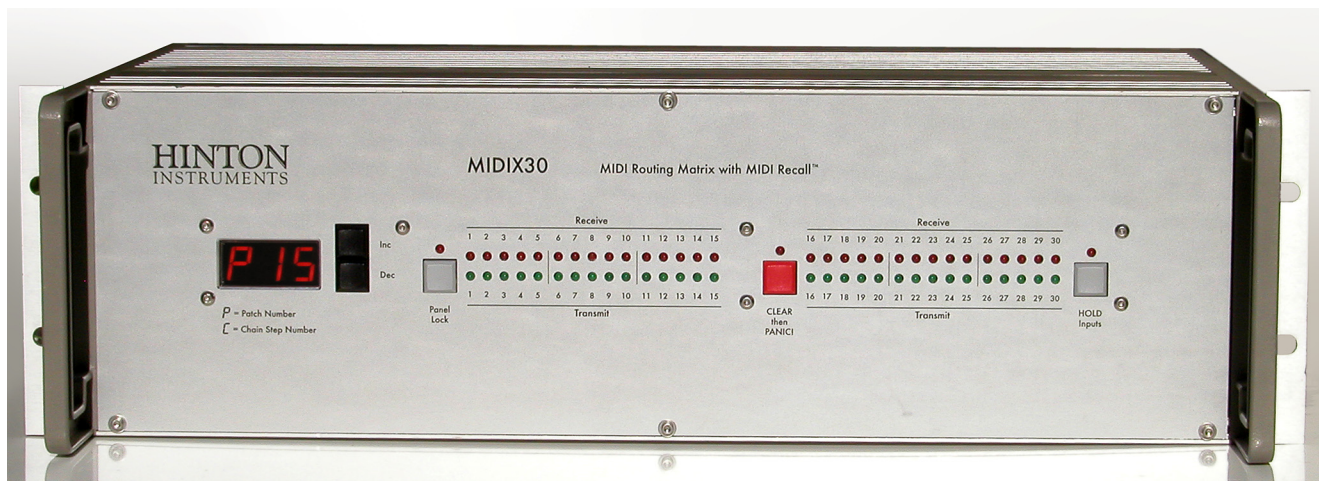
If MIDIX is already on when the terminal is started its window will be blank, but typing a character will induce a response. Typing ? should produce a short response showing the software and version number, typing H should produce a longer Help message. If random characters appear in the window, the computer is probably set to the wrong Baud rate. If messages appear, but with characters missing there may be a handshaking problem. Sometimes XON/XOFF handshaking may be slow to respond, in this case RTS/CTS hardware handshaking may be used.

If you experience any difficulty with the above operations or require any further information, please contact Hinton Instruments: support@hinton-instruments.co.uk.

More reference information and contact details are available on our website:

<http://www.hinton-instruments.co.uk>.

Front Panel



FRONT PANEL CONTROLS

MIDIX-30 has five front panel pushbuttons which allow limited standalone operation and emergency features.

The INC/DEC buttons to the right of the numeric display change the Patch number or Step number when in Chain mode. Both of these buttons have a slow auto-repeat and the new Patch or Step is selected on release. This allows the intervening patch or step numbers to be skipped. To cancel the effect of a button release while holding either button down, simply press the other button as well, then release both.

The PANEL LOCK button deactivates the other buttons to prevent accidental operation. The front panel may also be locked or unlocked with the F command.

The PANIC button has a delayed action to prevent accidental use. After holding for about 2 seconds a universal Clear Event will be sent to all Outputs, unless they are listed in Illegal Group 0. After a further 2 seconds a full Panic sequence, equivalent to the ! command) is transmitted.

The HOLD button disables all MIDI Inputs. The button has a delayed action to turn on, but not to turn off. This may be used in an emergency to break a *MIDI Feedback Loop* which will lock up the matrix from changes. The cause of the feedback should be dealt with before releasing the Hold. This is equivalent to the E command.

Rear Panels



The MIDIX-30 rear is divided into two panels: The MIDI I/O panel and the MIDIX Controller panel.

All MIDI I/O is via Multi-MIDI ports based on a DB25 socket comprising five MIDI Ins and five MIDI Outs. A multicore loom terminated with a DB25 plug with UNC 4-40 screwlocks is required for each Multi-MIDI port. A substantial cable tie bar is provided to support the looms which should be laced in position to prevent the weight of the cable pulling on the connectors. Do not rely on the screwlocks alone, the socket contacts may get stretched which will cause intermittent connections.

The Controller panel includes the RS232 ports and a fused IEC mains inlet. An Earth must be provided via the mains cord. DO NOT USE with two pin mains sockets.

Panels should not be removed while the unit is powered. Please consult the service manual before doing so.

MIDIX Operation

IN USE

The MIDIX-30 MIDI Routing Matrix is a sophisticated digital switching system for the interconnection of large numbers of MIDI devices in a professional environment. MIDIX is not a simple devices and has many advanced features, but has been designed to by easy to use, to be reliable and to enhance rather than obstruct.

All the MIDI equipment is connected to the MIDIX matrix where they are inter-routed under the direction of the MIDIX Controller which is connected both to the matrix via MIDI and to an external computer via RS232. All settings are stored in the MIDIX Controller's nonvolatile memory and may also be loaded and dumped by the external computer.

Please refer to the installation section for full information on setting up the MIDIX hardware and connecting to the computer. This section is intended to familiarise an operator with the basic concepts and day to day use and assumes that you have powered up the rack and have a Terminal Emulator running on the computer. More detailed descriptions of the commands are given in the Commands section.

The MIDIX-30 matrix has thirty two input buses and outputs of which thirty are available for external connections. The Controller has its own special MIDI inputs and outputs to the matrix and does not take up user ports. Each MIDI input is a full specification optical isolated circuit with fast turn on and off times. Each MIDI output is a current loop driver which is disabled at power on to prevent false data being sent.

Most equipment has a MIDI In and a MIDI Out so would logically be associated with the same port number, but does not have to be. Some devices have an unequal number of Ins and Outs so some thought is needed to allocate them to the matrix in a manageable fashion. Equipment with no Outs may be connected in a chain to another by its Thru output. If you require further advise on connecting please contact Hinton Instruments.

MEMORY PROTECT

The first task to be done with a new system or after equipment rearrangements is to name all the matrix inputs and outputs. Memory Protect must first be set to off by typing:

M0↵ (where ↵ is the Return character)

When all the changes have been made this may be set back on by typing:

M1↵

NAMING INPUTS AND OUTPUTS

Each connection may have up to a sixteen character long name which will be used on all displays giving routing information. There is no extra knowledge associated with the name, like identifying particular models of equipment, it is simply a label for the operator's convenience. All commands associated with naming use the quotation mark (") to start the string and a carriage return to end it, extra characters are ignored.

To name an Input, type:

I n "sixteencharacter"↵

where n is a number between 1 and 30, or to name an Output type:

```
O n "sixteencharacter"
```

Names may be changed at any time and, like all settings, it is recommended that a file is kept as a backup. To check what all the names currently are the *Dump Names* command may be used. Typing **D** will give a list of all the Inputs and Outputs in a system.

NAMING AND DEFINING EVENTS

Each MIDIX Output may have an *Event* associated with it that will be transmitted whenever it is disconnected from an Input. An Event is an arbitrary user definable list of MIDI codes up to 320 bytes in length. MIDIX is supplied with several common and useful ones already defined in its memory and these may be redefined as needed at any time. Events are referred to by the number sign (#) and a number from 1 to 50, they also have a sixteen character name.

To define an Event, type:

```
# n < hh, hh, hh, . . . hh>
```

where n is the event number and hh are the MIDI codes in hex. To name it, type:

```
# n "sixteencharacter"
```

To check an event, type

```
#n
```

and the name followed by the hex list will be given. All the events may be dumped with the *Dump Events* command, **D#**. This is quite a lot of information and will be more than a screen full and is really intended for saving to disc.

CLEAR EVENTS

To associate an Event with an Output, type:

```
CL n # nn
```

where n is the output number and nn is the event number. Notice how the command is expanded when you type it and the current setting is shown, if you do not wish to change it simply type Return. The complete *Clear List* may be obtained using the *Dump Clear List* command, **DC**.

If no event is required to clear an Output, enter 0 for the event number. Events may be used for other purposes which will be covered later.

ILLEGAL GROUPS

Each Output can have a defined *Illegal Group*. This is a list of Inputs that cannot be connected to that Output. Certain MIDI device such as sequencers and mergers can lock up if they have their output connected to their input forming a *MIDI Feedback Loop* and this is the first line of defence. To define an Illegal Group, type:

```
X n ( n, . . . n )
```

where the first n is the Output number and the ns in the list are Input numbers that should not be connected to that Output. Of course, this cannot prevent two MIDI devices being connected in series in a loop, but please refer to the Hold command for use in that eventuality.

An Illegal Group may be cleared by typing:

```
X n ( )↵
```

or inspected by typing

```
X n↵
```

and saved with the Dump Illegal Groups command, DX.

There is a special Illegal Group 0 which is a list on the Outputs which the Clear and Panic Events should not be sent to.

EVERYDAY USE

Now that we have defined Names, some Clear Events and Illegal Groups we can use MIDIX to make some routing connections. MIDIX is always in edit mode and routing may be changed at any time.

ROUTING

There are several Output commands, we have already used the Output Name (O”). All routing is accomplished with the Output equals command (O=).

```
O n = m↵
```

where n is the destination Output number and m is the source Input number to be routing to it. For no output m may be 0. After typing ↵ (Return) MIDIX will report the connection made using the already defined Input and Output names. To see what source an output is set to type:

```
O n↵
```

and to see what destination(s) an input is routed to (there may be more than one), type:

```
I n↵
```

The complete current routing may be shown with the *Dump Current Status* command:

```
D↵
```

PATCHES

Once a routing configuration has been set up it may be stored in one of the 50 Patch Memories. The current setting is kept in an *Edit Buffer* and it is possible to toggle between changes and the last Patch with a compare/recall action like synthesizer program memories.

To *Store* a Patch, type:

```
S nn↵
```

where nn is the memory number between 1 and 50 and defaults to the last recalled

memory. To *Recall* a Patch, type:

P nn↵

and to toggle between the Edit Buffer and this Patch after subsequent editing simply type

P↵

NAMING PATCHES Each Patch may have up to a twenty character name.

Patches may be named directly by typing:

P nn "twentycharactername"↵

CHAINS

Patches may be joined into a Chain to create a cycle for working. A Chain is a list of Patch numbers in sequence, a number may appear more than once, up to a total of 50 steps and is defined by typing:

CH [nn, nn, nn, . . . nn]↵

The Chain is entered by typing open bracket ([), advanced with a slash (/), retarded with a backslash (\) and exited with a close bracket (]). When active the front panel display will have C as the first character, instead of P.

To show the current Chain List, type:

CH↵

DUMPS

All the information that we have set up may be dumped separately or altogether with the Dump commands. Dump Names, Dump Events, Dump Clear List, Dump Illegal Groups and Dump Current Status have already been introduced. The remaining commands are Dump Patch (Dnn), Dump All Patches (DP), Dump Chain List (D[), Dump Filters (D%) and Dump Everything (D&). Most of these commands are executed immediately the second character is typed without waiting for a Return or allowing a correction.

All the dumps are in MIDIX command format so that all that is necessary to save and restore a complete setup is to keep the ASCII data as a text file.

DUMPING SETUPS TO DISC

To save any MIDIX setup first enable your Terminal Emulator to receive data for filing, then type the appropriate dump command or commands and watch the information in the window. When it is finished, close reception and save the file.

All the information is printable ASCII and may be edited with any text editor. It may be preferable to design Event Lists "offline" with an editor than to type into MIDIX directly.

RELOADING SETUPS FROM DISC

To restore any setup or partial setup select the file for transmission with the Terminal Emulator and send it. You should see all the information on screen confirming that MIDIX has accepted it. Note that the information has been expanded from the minimum condensed format used for dumping.

MIDIX is completely independent of computer types and disc formats. When MIDIX was first designed 8 bit 6502 based personal computers were still considered seriously and Windows and MacOS were barely on the horizon. Now many versions of these platforms are obsolete and unsupported, but MIDIX continues to perform the same role with the same data regardless of where it is stored.

There are two special commands to deal quickly with problem situations, These are *PANIC* and *HOLD*.

PANIC !

The major fear in any MIDI setup, especially for live work, is getting notes stuck on. The Clear List and Safe Switching features of MIDIX go a long way towards preventing this, but if it does happen a Panic Sequence may be sent out to all equipment simultaneously by simply typing ! or by pressing and holding the front panel button. The sequence comprises Stop, All Notes Off on all channels, then centre all Pitchbend Wheels, zero all Modulation Wheels and Reset All Controllers on all channels, then every Note Off on all Channels and finally a single Active Sensing Clock. There are pauses within this massive stream of data to avoid overflowing the buffers in all the receiving equipment. The pause time is set with the Block Delay command, Bn, where n = 0 to 9.

There is also an Output Panic command that sends the same sequence to one specified Output, type:

O n !↵

where n is the Output number, Don't forget the carriage return as this is not a single character command.

The other problem area is MIDI Feedback Loops formed by connecting the output of a merger or sequencer back to its own input, directly or indirectly. Nothing will happen until a MIDI byte is transmitted and then it will circulate endlessly causing the matrix to appear locked. This is because MIDIX has Safe Switching which only changes routings in the gaps between MIDI codes and when feedback occurs there are no gaps. The Illegal Group definitions should be used to prevent foreseeable loops being routed, but several devices may get connected in a loop or a loop may be formed external to the matrix. If feedback happens one or more of the I/O activity LEDs will be fully on and the only way to break the loop is to press the front panel Hold button or type the Emergency Hold command:

HOLD

E

All MIDI Inputs will be disabled and the routing may be changed to a safe state. Press the Hold button again or type ↵ to continue normal operation.

ADVANCED USE

There is one final Output command that is, perhaps, the most powerful or them all: Output Message. The format is:

MESSAGES

`O n <hh, hh, . . . hh>↵`

where n is the Output number and hh are MIDI data in hex. From the start of the opening bracket (<) to the closing bracket (>) the selected Output is switched over to the MIDIX Controller instead of whatever it was connected to. Any amount of data may be sent and when finished the patch is restored.

It is hardly likely that anyone would ever attempt to type anything more than a Program Change or two by hand, so where is all this data originated?

MONITORING

The MIDIX Controller can function in two modes of acquiring MIDI data. It can Monitor any MIDI source and translate the data stream into either Hex or English, subject to selectable Filtering, or it can capture specific manufacturers' System Exclusive blocks.

All available RAM not used for parameter and system storage is used as a buffer for MIDI input and in the current version is about 16Kbytes. This is equivalent to over 5 seconds worth of solid head to tail MIDI transmission and should be adequate for most situations.

Monitoring is started by typing:

`I n H`

for Hex format, where n is the source Input number, or

`I n E`

for English. Note that this command starts immediately on the H or E and stops on the next character typed so it cannot be given from a file. In Hex format the data is bracketed with < and > ready for saving and converting into an *Event Define* or an *Output Message* command.

FILTERS

Every different type of MIDI data may be independently filtered to eliminate unwanted data such as Aftertouch, Clocks, MIDI Time Code or Active Sensing. The filters operate before placing data into the buffer and may be set or cleared by typing:

`% <hh> n↵`

where hh is the MIDI Status and n is 0 for pass (i.e. no filter) and 1 for remove or filter. All filters may be cleared by typing:

`% <0>↵`

The filters are only removing data from the MIDIX Controller input for selective display. If you need processing of data within the matrix you will require a MIDIC Processor or MIDIY Merger.

In Hex mode the Input Monitor display is an exact translation of the incoming data showing every nuance of Running Status and Undefined data types. This makes it possible to diagnose obscure fault conditions of equipment incompatibility due to manufacturers

either misunderstanding the MIDI specification or simply being incompetent. The display has a new line inserted at every Status change to improve readability, but otherwise is in the exact form received. If you are receiving large amounts of data at high rates, like MTC, eventually the buffer will be overloaded and the data lost, but this will appear obvious by the irregular pattern. Start the Input Monitor command again to reset the buffer.

In English monitor mode only Pitchbend and Song Position Pointer, which are both 14 bit data types, are *interpreted* into a single decimal number. All other data is still a strict translation and is in decimal except for System Exclusive and MIDI Time Code which is more readable in hex.

INPUT EXCLUSIVE The *Input Exclusive* command is intended for use with equipment where bulk dumps may only be initiated manually. To enable capture, type:

```
I n <hh>↵
```

where n is the source Input number and hh is the manufacturer's ID in hex. 0 may be used for don't care or three byte IDs. The MIDIX Controller will switch over the matrix and wait for the next block with the specified, or unspecified, ID. All commands and front panel controls are inoperative until the block is completed to allow time to move to the source equipment and manually persuade it to transmit its System Exclusive data. This command may be aborted with {Esc} and so should not be driven from a file as it will read all following characters looking for an {Esc}.

MIDI RECALL™ The two Recall commands are similar, one may use an Event to request a System Exclusive block and the other uses a hex list. To execute an automatic *Capture*, type:

```
R n <hh> m <hh, hh, . . . hh>↵
```

or

```
R n <hh> m #nn↵
```

where n is the source Input number, hh is the ID, m the Output number to transmit the request on and <hh etc> is the request in hex bytes or #nn is the Event number containing the request. Neither command may be aborted once completed, but they will time out if a reply is not received in about 30 seconds.

The next block captured will be formatted as an Output Message command using Output m. All that is necessary to do to save and restore a complete setup is to have one file containing all the Recall commands for each piece of equipment and transmit it with text reception enabled. The resulting data is already in the correct format for retransmitting as Output Message commands.

The saved file may be edited with a text editor and different dumps may be cut and pasted together. The System Exclusive data within the < > brackets should not normally be altered without careful consultation of the manufacturer's documentation as it often contains *Checksums* that require a hexadecimal calculator to compute.

The necessary information to construct System Exclusive requests is usually, and should be, provided with the equipment manual. If this is lacking in any way Hinton Instruments

will obtain and provide this information ready for use upon request.

Now that all major operations performed by MIDIX have been covered and the general style and scope of the command set introduced it should be possible to perform a complete system setup of MIDIX and all connected equipment from one disc file. To make such archived data MIDIX has a *Comment* command, any line starting with a semicolon (;) is ignored until the following Return. Descriptions maybe freely inserted within files provided that they begin with a semicolon. Tabs, spaces, carriage returns and line feeds are also ignored.

COMMENTS

HELP

Online help may be obtained by typing H and the summary of commands in this manual will be paged onto the screen. This is intended as an aide memoire rather than a complete built in manual. Section 3 gives a detailed description of each command individually.

TERMINAL MODES

Everything described so far in this section has assumed that MIDIX was connected to a Dumb Terminal as this is the power up default state and all the command dialogue has been conducted on a scrolling screen simulating a paper roll. This may be altered with the Terminal command with has two other settings. Typing:

T 0↵

will turn off all expansion of commands and is intended for driving MIDIX with a dedicated computer application. If you terminal emulator has VT100 capability type:

T 2↵

VT100 MODE

and the screen will change a fixed page display that may be controlled using the cursor and numeric keypad keys as well as all the commands. When using a terminal emulator with VT100 capability comprehensive paged screens of all the MIDIX settings may be viewed. This technique is common to a lot of our products and ensures immunity from the hectic changes of the computer industry. In ten years time this software will still function on some computer with some emulator, just like it did ten years ago.

All information on the current state of MIDIX is shown as a series of tabular displays starting on the *Main Status* page. The screen is divided into three parts: the *Header* which shows the page name in the top right corner, the *Display Window* taking up most of the screen and the *Command Window* at the bottom. The *Command Window* behaves exactly as the Dumb Terminal mode and all commands may still be entered here.

When MIDIX is altered by command or front panel button operation the displays will update to show the correct current state.

The *Current Parameter* is always shown in reverse video and may be moved around the screen with the Cursor Keys. *Parameter Names* are shown ending with a colon (:) and are not active, the *Values* of parameters are to the right of the Names and may be decreased with the minus (-) key or increased with the plus (+) or comma (,) keys. (This makes sense when looking at a real VT100 application keypad which has a comma instead of the plus used by most computer keyboards.) Using these keys the value may

be cycled through its range or the numeric keypad may be used for short cuts. Simply navigate the Selected item (shown here on STATUS) around the screen with the Cursor Keys and then use + or - to scroll through the available options for each parameter.

When the current parameter is the Page Name the page may be incremented or decremented.

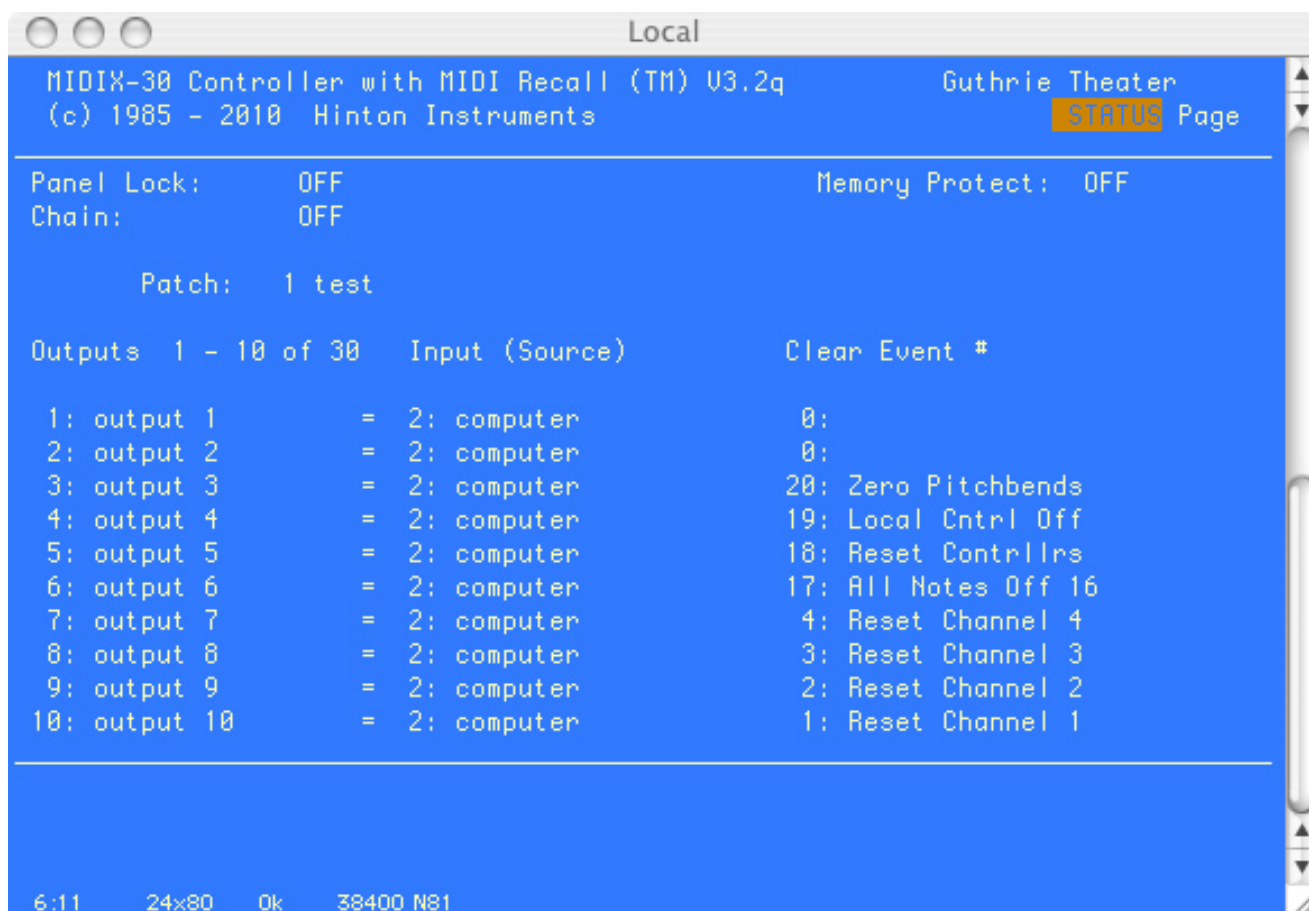
KEYPAD OPERATION

The *Application Keypad* section of the keyboard has special usage: The Decimal Point key will always home to the *Page Name* parameter. The Keypad numbers may be used to directly set a decimal value terminated by the Keypad Enter key ($\overline{\wedge}$), the screen will not update until the number is terminated. If the parameter is non-numeric the number will select a list member and if the number is out of range it will be ignored. The top four keys of a real VT100 keypad are known as the Programmable Function keys (PF1, PF2, PF3 and PF4) and these are usually the same on emulator programs or an equivalent is provided. PF1 and PF2 correspond to the real front panel buttons and the action of all may be configured on the Main Status Page.

To abort any command type the *Escape* key (ESC) twice, if no command has been started this will redraw the screen. This is also useful if the terminal has been disconnected or changed and is out of step with the MIDIX state.

Online Help pages are available by typing H.

THE MIDIX STATUS PAGE



Page: STATUS/CONFIG/MONITOR/FILTER, changes to other screen pages.

Panel Lock: On/Off, disables front panel controls.

Memory Protect: On/Off, prevents settings being altered.

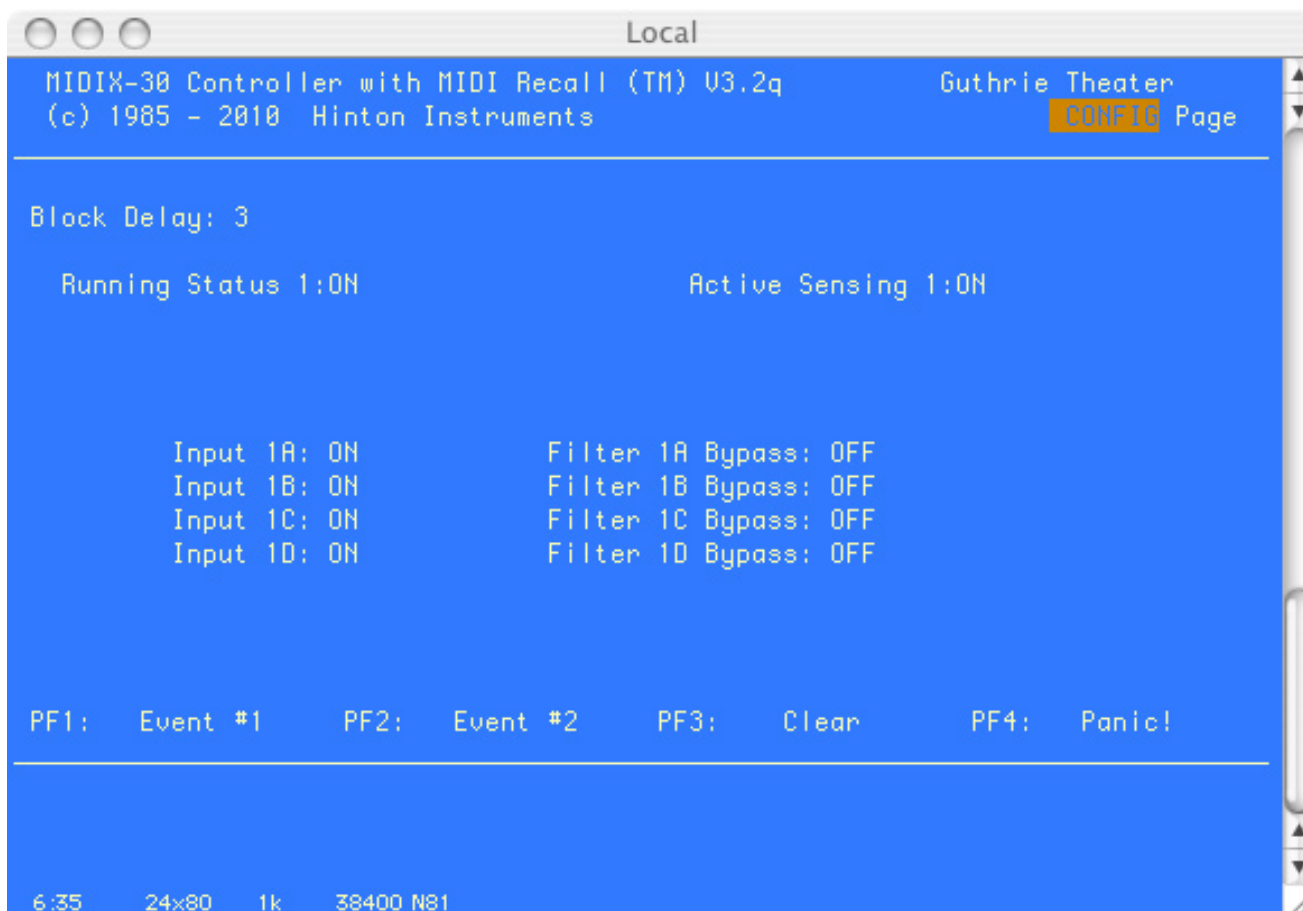
Chain: On/Off

Patch: 1 to 50, Patch number and title.

Outputs: 1-10 of 30/11-20 of 30/21-30 of 30, selects group of ten Outputs shown below.

Inputs: 1-30, shows the source number and title connected to the Output on the same line. 0 = off.

Clear Event: 1-50, shows the Event number which will be sent to the Output on the same line when it is disconnected from a source. 0 = no Event associated.



Block Delay: 0 - 9, sets the delay between large generated blocks of data to prevent overflowing the data buffers of receiving equipment.

Running Status 1: On/Off, determines whether Merger Output 1 is converted to Running Status.

Active Sensing 1: On/Off, automatically generates Active Sensing Clocks.

Input 1A (1B, 1C, 1D): On/Off, any Merger input may be locked out.

Filter 1A (1B, 1C, 1D) Bypass: On/Off, enables the individual filter sets on each input.

PF1, PF2, PF3 & PF4: PF1-4 are the VT100 keys which have a programmable function:

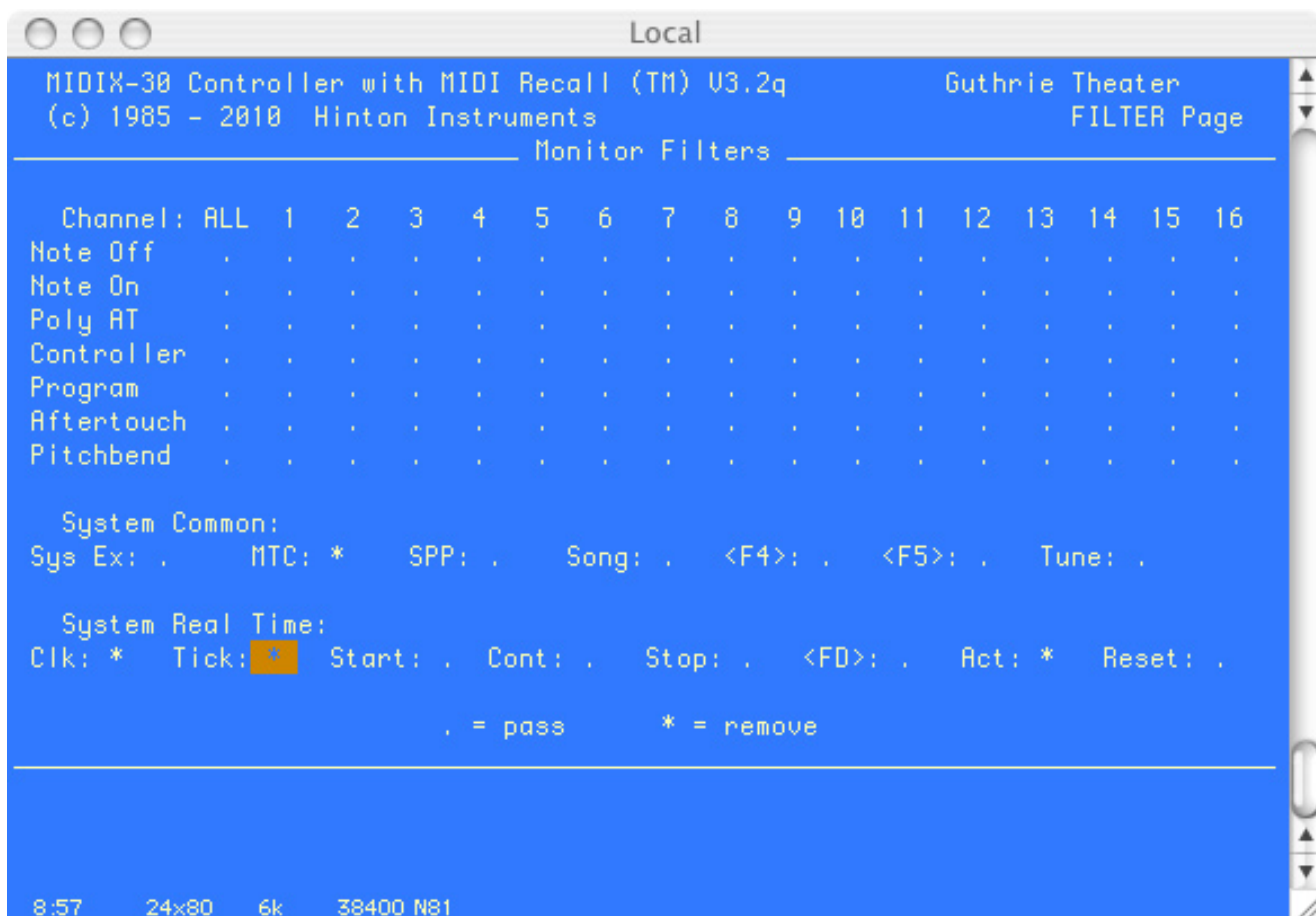
0 (off)

1: Event #1: Transmits user defined Event #1.

2: Event #2: Transmits user defined Event #2.

3: Clear: Transmits a Clear Event.

4: Panic!: Generates Stop, All Notes Off, Reset All Controllers, All Sound Off, Pitchbend = centre, Modulation Wheel = 0, Hold = 0, Volume = max, Channel Aftertouch = 0; then individual Note Offs for every note on every MIDI Channel.



This page has five sub-pages, one for the Monitor Filter and one for each of the Merger Input Filters, 1A, 1B, 1C, 1D.

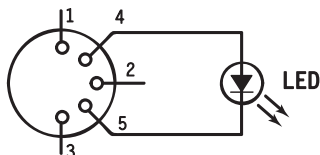
Every MIDI Status has an individual filter and there is additional filter for All Channels of Channel Messages which overrides any individual Channel filters.

In the example shown the sub-page parameter is set to Monitor Filters. MIDI Time Code, Clocks and Active Sensing are being filtered.

Note: Although Active Sensing and Reset are shown on each sub page they are always filtered on Merger Input regardless of the setting.

TROUBLESHOOTING For diagnosing MIDI transmission and distribution related problems we recommend fitting both a 5 pin DIN plug and a 5 pin DIN line socket with an LED soldered across pin 4 and 5, with the anode to pin 4, so that the LED body just protrudes through the cable support.

DIN-5 LINE SOCKET



These testers may be used to check that MIDI data is being transmitted by substituting them for the receiving equipment. They should not be used in parallel.

Starting with a MIDI source, plug the line plug into its MIDI Out and observe the LED flicker when MIDI data should be sent. Replace the normal MIDI cable and plug the line socket into the other end to check that the cable is not faulty. Pin 4 shorted to pin 5 or to the screen will cause this LED to be permanently off, whereas pin 5 shorted to screen will cause the LED to be fully on.

Work along the MIDI data route until you are sure that data is being presented to the final MIDI input.

The MIDIX-30 Routing Matrix should give years of fault free operation—many Hinton Instruments products have been in continuous use for over twenty years or used on major international tours and events—but it has been designed for ease of servicing in the event of any component failure.

Spare components and sub assemblies are available for critical maintenance applications. Please enquire if you need this support.

MIDI Implementation Chart

Function		Transmitted	Recognised	Remarks
Basic Channel	Default Changed	1-16 1-16	X X	
Mode	Default Messages Altered	O	O	
Note Number	True Voice	O	O	* Note 1
Velocity	Note On Note Off	O O	O O	* Note 1 * Note 1
Aftertouch	Key Channel	O O	O O	* Note 1 * Note 1
Pitch Bend		O	O	* Note 1
Control Change		O	O	* Note 1
Program Change		O	O	* Note 1
System Exclusive		O	O	* Note 1
System Common	MTC Song Position Song Select Tune	O O O O	O O O O	* Note 1 * Note 1 * Note 1 * Note 1
System Real Time	Clock Commands	O O	O O	* Note 1 * Note 1
Aux Messages	Local ON/OFF All Notes Off Active Sense Reset	O O O O	O O O O	* Note 1 * Note 1 * Note 1

Notes: 1. Every MIDI Byte is passed without processing.

O : Yes
X : No

MIDIX Commands

MIDIX may be completely controlled from the RS232 port by either a terminal (dumb or smart) or a computer either emulating a terminal or running a program specific to MIDIX. The commands are given as readable lines of ASCII characters and are structured for entry by hand typing or by transmitting from a saved text file.

There is one command per line determined by the first character after a Carriage Return {CR}, Line Feeds are ignored. Upper and lower case are treated as the same and any illegal command will generate an error message:

H for Help

There are three types of command:

Single Character commands that take immediate effect

Double Character commands

and Compound Commands ending with a Carriage Return

Apart from the single character commands all may be aborted by sending the ASCII Escape character {ESC}, <1Bh>, once or, if in VT100 mode, twice. It is only necessary to type the minimum number of characters to make a command unambiguous, e.g. I1 for Input 1 or I2 for Input 2, any following characters will be ignored until a valid argument is entered.

Some arguments may be defaulted by entering a {TAB} or {CR}, but not a {Space}. This has been done for ease of manual entry and for keeping commands saved in ASCII files.

MIDIX 30 Commands

Conventions:	{symbol}	= a key on the computer/terminal keyboard
	{CR}	= Carriage Return (Line Feeds are ignored)
	n	= small decimal number, usually 0 to 16
	nn	= decimal number
	nnnn	= large decimal number
	hh	= hexadecimal number, 00 to FF
	hhhh	= 16 bit hexadecimal number
	< ... >	= a byte string in hex
	" "	= an alphanumeric string in ASCII
	{BS} and {DEL}	both delete backwards
	{ESC}	- aborts command

Commands may be entered manually or transmitted from an ASCII file. All Dumps and Recalls are in the correct command format for saving and retransmitting with Terminal Emulation software. Command files may be edited with a text editor.

In VT100 mode the cursor keys move parameter selection shown in reverse.

{+} or {,}	increments the selected parameter and {-} decrements it.
{Keypad numbers}	followed by {Enter} set the parameter directly
{Keypad .}	Homes to Page Selector
{ESC} twice	to abort command or redraw screen

Single Character Commands:

?	- Show Status
^	- Interrogate Vers/Rev
!	- Global Panic
[- Enter Chain
]	- Exit Chain
/	- Advance Chain
\	- Retard Chain
E	- Emergency Hold until {CR}
H	- Help

Double Character Commands:

DP	- Dump all Patches
DC	- Dump Clear List
DX	- Dump Illegal Groups
D%	- Dump Filters
D#	- Dump Events
D"	- Dump Input/Output Names
D[- Dump Chain List
D&	- Dump Everything

Commands terminated with {CR}:

D{CR}	- Dump current status
Dnn	- Dump Patch Number nn
Bn	- Block Delay (n: 0 - 10)
Mn	- Memory Protect (n: 0 = off, 1 = on)
Fn	- Front Panel enable (n: 0 = off, 1 = on)
PFt=n	- Define Function (t: 1, 2, 3, 4; n = 0 -5)*
Tn	- Terminal type (n: 0 = off, 1 = Dumb, 2 = VT100)
Wnn	- Wait nn x 100ms delay
"...	- set unit ID
;...	- Comment

MIDIX 30 Commands

Patch Control Commands:

Snn - Store Patch nn
Lnn(n,n, ... n) - Load Patch nn
Pnn - Recall Patch nn
P - Recall/Compare Patch
Pnn" - Name Patch nn
CH[nn,...nn] - define Chain List
Xnn(nn,...nn) - define Illegal Group

Event Commands:

#nn - show Event List nn
#nn " - name Event List nn
#nn <hh,...hh> - define Event List nn

Output Commands:

Onn - show Input connection
Onn =mm - select Input mm, 0 = stop
Onn"... - name Output
Onn! - panic clear Output
Onn#mm - send Event List mm to Output nn (nn = 1-30)
Onn<hh, hh, ... hh> - send message to Output nn (nn= 1-30)
CLnn#mm - clear Output nn with Event List mm

Input Commands:

Inn - show Input name and connections
Inn"... - name Input
Inn<hh> - show next System Exclusive block with ID = hh, 0 = any
Inn H - monitor Input nn in Hex
Inn E - monitor Input nn in English
Inn #mm<...> - capture Event # mm from Input nn

%M<hh>n - Filter Monitor Status hh (n: 0 = off, 1 = on)

MIDI Recall™ Commands:

Rnn<ID>mm<hh, ... hh>
- Recall from Input nn the next System Exclusive block with ID = hh
after transmitting message on Output mm
Rnn<ID>mm#nn
- Recall from Input nn the next System Exclusive block with ID = hh
after transmitting Event List nn on Output mm

Optional Merger Commands:

A1n - Active Sensing 1 (n: 0 = off, 1 = on)
R1n - Running Status 1 (n: 0 = off, 1 = on)
Yn=s - Input n switch (n: A - D; s: 0 = off, 1 = on)*
%n=s - Filter n Bypass (n: A - D; s: 0 = off, 1 = on)*
%A<hh>n - Filter Input 1A Status hh (n: 0 = off, 1 = on)
%B<hh>n - Filter Input 1B Status hh (n: 0 = off, 1 = on)
%C<hh>n - Filter Input 1C Status hh (n: 0 = off, 1 = on)
%D<hh>n - Filter Input 1D Status hh (n: 0 = off, 1 = on)

SINGLE CHARACTER COMMANDS

? - Show Status
^ - Interrogate Vers/Rev
! - Global Panic Clear
[- Enter Chain
] - Exit Chain
/ - Advance Chain
\ - Retard Chain
E - Emergency Hold
H - Help

The single character commands are all operations requiring a single keystroke for fast manual entry.

? - SHOW STATUS Also invoked at power on/reset, shows the title, software version and revision and the current status. e.g.:

```
MIDIX-30 Controller with MIDI Recall (TM)
Vers:3.2q (c) 1985-2010 Hinton Instruments

MIDIX Matrix Size: 30 Inputs into 30 Outputs
50 Patches
50 Event Lists of up to 320 MIDI bytes each
Maximum System Exclusive Block capture size: 16369 bytes
Buffer 1A Size: 16375
Buffer 1B Size: 16375
Buffer 1C Size: 16375
Buffer 1D Size: 16375

Block delay: 3
Running Status 1: 1
Active Sensing 1: 1
Memory Protect: 0
Front Panel enable: 1
```

^ - INTERROGATE VERSION/REVISION

This character returns the program title and software version and revision number followed by the serial number. It is intended to be used by a computer program to check that it is connected to the correct equipment and firmware, but is useful if sharing one terminal between several units.

! - GLOBAL PANIC CLEAR

will transmit a generated MIDI Panic sequence comprising Stop, All Notes Off and Reset all Controllers; a delay followed by centre Pitchbend, zero Modulation Wheel, maximum Volume and zero Channel Aftertouch; another delay and then 128 Note Offs for all sixteen MIDI Channels and finally one Active Sensing message to guarantee clearing any stuck notes in any equipment. The sequence is transmitted on all MIDI Outputs apart from those listed in Illegal Group 0.

The sequence is sent as a series of blocks, with a variable delay between them, to avoid overflowing the input buffers of the connected equipment. This delay may be altered with the **B** command.

Panic Clear

will appear on the screen and the front panel Output activity LEDs will show the operation in progress. All buffers are cleared. When completed the matrix will revert to its previous routing.

CHAIN CONTROL A chain of patch selections set by the Chain Define command may be sequenced by four commands:

[- ENTER CHAIN selects the first patch defined in the chain and enables advance/retard action. If the first patch is already selected then it will be unchanged. The patch number and title appearing on the terminal screen shows that the patch has been changed.

] - EXIT CHAIN disables further advancing or retarding of the chain sequence. The last selected patch will remain unchanged.

/ - ADVANCE CHAIN & \ - RETARD CHAIN

selects the next patch, forwards or backwards, in the Chain. The Chain may be cycled indefinitely until the Exit command.

E - EMERGENCY HOLD

is to allow the breaking of *MIDI feedback loops*. This command should only be used to get out of the lockup condition caused by a MIDI data processing unit accidentally having its outputs connected to its input and continuously circulating data. As MIDIX employs a safe switching technique so that routing is only changed between MIDI messages, if there are no longer and gaps then any routing changes will never be made. If this condition arises some MIDI activity LEDs will become solidly on and panel and command operations will appear inoperative.

When E is typed all matrix routing is disconnected and the cause of the feedback should be dealt with before typing {CR} to resume normal operation. Hold may also be activated/deactivated with the front panel switch.

H - HELP displays several pages of concise descriptions of every available command. The information is similar to the summary of commands at the beginning of this section.

After each page the message:

More ... ESC to abort, any key to continue ...

will be shown until the end of the information. In VT100 mode the help information will be shown in the central window, rather than the command window, which will be restored when completed or aborted with a double {ESC}.

DOUBLE CHARACTER COMMANDS

- D{CR} - Dump current status
- D{nn} - Dump Patch number {nn}
- DP - Dump all Patches
- DC - Dump Clear List
- DX - Dump Illegal Groups
- D" - Dump Input/Output Names
- D[- Dump Chain List
- D% - Dump MIDI Input Filters
- D# - Dump Event Lists
- D& - Dump Everything

All the commands in this group invoke a partial or complete dump in readable ASCII format. These commands may be given to view the current setups or to save them to disk via a computer. When using a terminal emulator with disk filing, enable capture reception before issuing the Dump command and then save the capture buffer to disk when the dump (seen on the screen) has finished.

All dumps are in MIDIX command format so that the file may be simply transmitted back to restore that state. As the dumps are in printable ASCII they may be edited with any word processor and cut and pasted to make new setup files. Comments may be added to the files for future reference by preceding with a semicolon, e.g.:

```
;patch dump on Tuesday afernoon↵
```

All D commands cause the message

```
;Dump
```

to appear on the screen to acknowledge that the command has been started. The message starts with a semicolon to turn it into a safe comment in case the dump is being save to disk, otherwise it would be interpreted as a command when sent back. This line may be deleted when editing files.

When shipped MIDIY contains "factory presets" of useful events. It is recommended that a complete dump is saved to disk and used as a template to produce new files.

D{CR} - DUMP CURENT STATUS

produces a dump of the current settings as a series of abbreviated commands. e.g.:

```
B 2  
R1:0  
R2:0  
S1:0  
S2:0  
I 1= 1  
...etc
```

When reloaded these commands will update or restore the operational settings for Block Delay, Running Status, Active Sensing, Input Switches, Filtering Bypass and the

Programmable Functions.

D{NN} - DUMP PATCH NUMBER {NN}

(Not strictly a Double Character command, but included here with the other dumps) requests a specific patch to be dumped in the form of a Patch Title and a Load command (see P and L sections). If the number {nn} is out of range an error message/bellring will be given and the command ignored/

e.g. **D9**↵

might dump:

```
Patch: 9 "Magic hot formula
L 9 ( 1, 2, 3, 4, 5, 6, 7, 8,)
```

By editing the number 9 in the save file this patch could be reloaded in a different patch location. When reloaded only the patch memory will be updated, the current status will be unaltered unless that patch number is currently selected.

DP - DUMP ALL PATCHES

as above, but dumps all patches as Name Patch and Load Patch commands. Again, the dumped file may be edited and the patch numbers and data changed to produce new setups offline. If there are repeated patch number in a file the last one will overwrite the other.

DC - DUMP CLEAR LIST

Each MIDIX Output may have an Event List associated with it that will be transmitted if it becomes disconnected from an input. The Clear List is a set of commands to establish the correspondence of an Event List with an output.

DX - DUMP ILLEGAL GROUPS

dumps all the Illegal patching Group for each output as a set of define Illegal Group commands.

D" - DUMP INPUT/OUTPUT NAMES

will output a list of Name Input and Name Output commands for the matrix size.

D[- DUMP CHAIN LIST

will output a Chain Define command for the current Chain, If there is no Chain set up a clear chain command is dumped.

i.e. **Chain: []**

D% - DUMP MIDI FILTERS

dumps the current state of the Input filters as an Open All Filters command followed by

a series of Status filter definitions for each closed filter.

e.g. if only one Pitchbend Filter for Channel 7 were closed the dump would be:

```
%A<0>  
%A<E6>1  
%B<0>  
%C<0>  
%D<0>
```

The command to open all the filters first saves having a command for every single filter.

D# - DUMP EVENT LISTS

gives all the Events as Name Event and Define Event commands.

D& - DUMP EVERYTHING

is equivalent to all of the above Dump commands in turn.

COMMANDS TERMINATED WITH {CR}:

Bn - Block Delay
Mn - Memory Protect
Tn - Terminal type
Fn - Front Panel enable
Wn - Wait
PFt=n - Define Function
"... - set unit ID
;... - Comment

BN - BLOCK DELAY n may be in the range 0 to 10 and sets the duration of the delay between data blocks in the Panic and Clear sequences. The value should be adjusted to be the smallest compatible with the MIDI equipment being used. Too small a value may cause this sequence to overload some of the MIDI Receive buffers and so have the opposite effect to that intended. Too large a value will increase the total time taken unnecessarily.

MN - MEMORY PROTECT

n may be 0 to turn protection of or 1 to turn it on. A default argument may be entered by typing {CR} to show the current state without altering it. When set On all commands that alter the stored memory will be treated as errors.

TN - TERMINAL TYPE

n may be 0 for off, 1 for "dumb" terminal and 2 for intelligent terminal (VT100) and defaults to 1 after power up or reset.

When n=1 MIDIX assumes that it is communicating with a dumb terminal and expands all responses to make them readable to the user. When n=0 responses are minimal and terse and are intended for communication with a computer program that does not require superfluous information.

The dumb terminal mode will work with any terminal or emulator on any computer and uses only a single scrolling screen.

The intelligent terminal mode uses a subset of the VT100 protocol that will work on any hardware or software emulation of the VT100 i.e. it does not use dim, underline or double size attributes that are often unavailable. This mode features a number of paged displays for better visual presentation and it is recommended for manual setting up and control. For dumps and restoring files it may be preferable to work in dumb terminal mode.

If changing from VT100 mode to another a VT100 reset terminal command is sent which will clear the screen, but may cause some emulators to reload their settings from disk so do not be alarmed if you notice your disk activity at this point.

FN - FRONT PANEL ENABLE

n may be 0 to disable the front panel manual controls or 1 to enable them. The default

state upon reset is enabled. The Panel Lock status is shown by an LED.

W_{NN} - WAIT

Wait for nn 100ms delays, where nn = 0 to 255. This command may be inserted between Output data commands in setup files when using with equipment that has a dead time after just receiving a block of data. If inserted in a Request file it will be transferred to the correct place in the Dump file.

PFT=N - DEFINE FUNCTION KEY

t = 1-4; n = 0-4

A VT100 terminal has four programmable function keys (PF1 to PF4) at the top of the keypad. The functions of these controls is set by an argument in the range 0 - 4:

0 (off)

1: Event #1: Transmits user defined Event #1 globally

2: Event #2: Transmits user defined Event #2 globally

3: Clear: Transmits both Event #1 and Event #2.

4: Panic!: Generates Stop, All Notes Off, Reset All Controllers, All Sound Off, Pitchbend = centre, Modulation Wheel = 0, Hold = 0, Volume = max, Channel Aftertouch = 0; then individual Note Offs for every note on every MIDI Channel.

“...

Up to twenty characters may be entered after the quote to identify the unit. This will be shown at the top right of the Header Window in VT100 mode.

;...

All characters following a semicolon are treated as comments until an {ESC} or {CR}. This may be used to annotate commands within a text file or for disabling commands.

PATCH CONTROL COMMANDS:

Snn - Store Patch nn
Lnn - Load Patch nn
Pnn - Recall Patch nn
P - Recall/Compare Patch
Pnn" - Name Patch nn
CH[nn,...nn] - define Chain List
Xnn(nn,...nn) - define Illegal Group

SNN - STORE PATCH

Loads the current matrix routing in the edit buffer to the specified patch memory, nn. If no number is specified the current patch selected will be overwritten, including any edits made since it was recalled. An error will occur if the patch number is out of range or Memory Protect has been set on.

On typing S

Store: nn

will appear in the screen, where nn is the current patch number. A new patch number may be entered terminated by any non-numeric character, usually a {CR}.

LNN - LOAD PATCH

Directly loads a patch memory without effecting the current routing or edit buffer. This is a quicker way of loading a set of memories as it does not involve setting every output in the edit buffer and then storing into a memory.

On typing L the screen will show:

Load 1

where 1 is a default memory number. A new number may be entered and then all characters are ignored until an open parenthesis (followed by a list of input selections in strict output order and ending with a close parenthesis).

e.g. **Load 1 (1, 2, 0, 0, 0, 6, 7, 8,)**

0 means an output is set to Stop and is the default condition. Any remaining outputs after the) termination will be set to Stop. A null list will Stop all outputs.

An error occurs if a number is out of range. All patch dumps output in this format, but if a dump from a larger matrix is loaded only the relevant number of outputs will be accepted and the remainder of the command will be treated as a comment.

PNN - RECALL PATCH NN

Sets the matrix routing and the edit buffer to the patch store in memory nn. On typing P a P will appear on the screen as this is an ambiguous command (see PF command above) until qualified by a number. As soon as a number is entered, say 4, the screen will show:

Patch: 4

which may be further entered or deleted, upon typing {CR} the screen will display the following line:

Patch: 4 "Patch 4 title

The first line, entering the command, is a request to select Patch 4. The second line is an acknowledgement that the patch has been updated. If the same patch were already selected there would be no response to the request. Patch acknowledgements will also appear on the screen when the patch is changed by front panel manual control or by Chaining.

P - RECALL/COMPARE PATCH

If a patch has been edited, either manually or by using the O= command, entering P with no argument and typing {CR} will toggle between the routing in the edit buffer and the memory store. Extending the previous example, typing P↵ three times after editing the patch would produce on the screen:

```
Patch: 4
Recall Patch: 4 "Patch 4 title
Patch: 4
Edit Patch: 4 "Patch 4 title
Patch: 4
Recall Patch: 4 "Patch 4 title
```

This action will also result from Chained patch changes if the edit buffer has been altered. Starting editing when in the Recall state will lose the previous edit.

PNN" - PATCH NAME

Directly sets the title of the specified patch number to be up to twenty characters following the quote, terminated by a {CR}. If more than twenty characters are entered they will be ignored. To produce the title in the examples above enter:

```
P4 "Patch 4 title↵
the screen will actually show:
Patch: 4 "Patch 4 title
```

CH[NN,...NN] - DEFINE CHAIN LIST

C is an ambiguous command (see CL below), qualifying with an H will produce on the screen:

Chain define:

all characters are then ignored until an open bracket [and then a sequence of patch numbers may be entered terminated by a close bracket]. The patch numbers may be in any order up to a maximum chain length of fifty. The patches in the chain will be sequenced using the [and] commands. Tabbing will automatically increment the patch numbers in the chain. E.g. entering:

```
CH[6→1→1→14→1→1]↵
will result in:
Chain define: [ 6, 7, 8, 14, 15, ]
```

Note that the first tab each time delimits the patch number. A chain may be cleared by entering the close bracket as the first step. i.e. CH[]

XNN(NN,...NN) - DEFINE ILLEGAL GROUP

An Illegal Group is a list of all inputs that cannot be connected to a certain output. On typing X the screen will show:

```
Illegal Group: 1
```

and, like all commands, the argument may be altered and the current state shown or redefined.

A null list enables an output to be connected to any input to the MIDIX matrix:

```
Illegal Group: 1 ( )
```

MIDI devices that should always be prevented from being connected to themselves include Sequencers and Mergers. Less obvious are certain expander modules with soft Thru or data repeating functions and modules that may get confused receiving a System Exclusive block dump at the same time they are transmitting it.

It is also good practise to limit MIDI Clocks and MIDI Time Code to only those devices that actually require them. Although the MIDI specification states that all devices should ignore all MIDI data codes that are not used, most common equipment has poor handling of this type of data. If a SMPTE timecode to MIDI Clock/MTC converter is being used, include its input number in every Illegal Group list except the computer or sequencer and any automation devices.

Illegal Group 0 is a special group that defines the outputs that global events should not be transmitted to.

EVENT COMMANDS:

#nn {CR} - show Event List nn
#nn " - name Event List nn
#nn <hh,...hh> - define Event List nn

#NN{CR} - SHOW EVENT LIST

Typing #will produce on the screen:

```
Event #: 1
```

the event number may now be entered followed by a {CR} and if within range the event title will be shown followed by the event list data on the next lines:

```
Event #: 1 "All Notes Off  
# 1 < B0, 7B, 00, B1, 7B, 00, B2, 7B, 00, B3, 7B, 00, B4, 7B, 00, B5, 7B, 00, B6, 7B, 00, B7,  
7B, 00, B8, 7B, 00, B9, 7B, 00, BA, 7B, 00, BB, 7B, 00, BC, 7B, 00, BD, 7B, 00, BE, 7B,  
00, BF, 7B, 00, >
```

This command is intended for use by an operator to check the contents of an Event List, the first line should not be saved to disk as it is not a command. Use **D#** to dump events.

If the event type is not delimited with a {CR} the event data may be entered with the following commands:

#NN "..." - NAME EVENT LIST NN

Up to sixteen printable characters may be entered following the quote character, further characters or trailing comments will be ignored. Delete and Backspace both delete backwards and the string may be aborted by typing {ESC} or stored by typing {CR}

#NN <HH, ... HH> - DEFINE EVENT LIST

After typing the < character to start hexadecimal entry up to 320 bytes may be given to define the Event List. {CR} will not terminate this command until the hex entry is ended by typing >. Leading Spaces, {TAB}s and {CR}s will be ignored so that the event list may go over more than one line in a file and be formatted for clarity.

The data defining an Event List will be used whenever that event is invoked. The validity of the MIDI protocol within the list is not checked either on entry or use. MIDIX is shipped with preset Event Lists which may be dumped and used as a template for editing into new events.

All MIDI equipment should include all the MIDI codes it recognises in its manual and if any difficulty is encountered translating this information into Event Lists or such information is not given by the manufacturer please consult Hinton Instruments web site or contact technical support.

The [MIDI Protocol Guide](#) gives a synopsis of MIDI structure with Controller numbers and System Exclusive IDs.

O - OUTPUT COMMANDS:

Onn{CR}	- Show Input
Onn = mm	- Select Input/Stop
Onn "...	- Name Output
Onn !	- Panic Clear Output
Onn #mm	- send Event List mm to Output
Onn <hh, hh, ...hh>	- send Message to Output

ONN{CR} - SHOW INPUT

Typing O will produce on the screen:

Output: nn

where nn is the default of the last output number used by a command. A different number may be entered or, if the default is correct, simply type {CR} and the source and destination names will be shown following as a comments. E.g. typing

01↵

might result in:

Output: 1 = 7 ; CZ101 to Fairlight III

if Output 1 were named "Fairlight III" and Input 7 were named "CZ101". If that output were in the Stop state

Output: 1 = 0 ; CZ101 to Fairlight III

would result with the names showing the connection that would be made if Stop is changed.

ONN = MM - SELECT INPUT/STOP

After entering the output number if = is typed the current selected input will be shown. This may be overtyped with a new input number (1 to 30) or 0 to Stop. After entering {CR} the new connection will be shown.

E.g. **07 = 1↵**

might result in:

Output: 7 = 1 ; Fairlight III to CZ101

ONN "... - NAME OUTPUT

Up to sixteen printable characters may be entered following the quote, further characters or trailing comments will be ignored. Delete and Backspace both delete backwards and the string may be aborted by typing {Esc} or entered by typing {CR}. This name will be used in displaying all connections with this output.

The remaining Output commands all transmit MIDI data from the specified output. After typing the qualifying character (!, # or <) following the output number the output will be switched exclusively to the MIDIX Controller. At the completion of the command the patch will be restored.

ONN ! - OUTPUT PANIC SEQUENCE

Transmits a Panic Sequence as used in the Global Panic command (!), but only to the specified Output. This command needs a {CR} before entering the next command.

ONN #MM - SEND EVENT LIST MM TO OUTPUT NN

Transmits any of the stored Event Lists to the current output. This may be used for specific initialisation, request or preset. Out of range arguments produce an error message, null events will be ignored.

ONN <HH, ... HH> - SEND MIDI MESSAGE TO OUTPUT NN

This is a free format version of the previous command. Each MIDI byte is transmitted from the selected Output as soon as it is delimited so that {Delete} and {Backspace} only function within each byte rather than the whole message. {CR} will not end this command until the hex entry is first ended by typing >. Leading Spaces, {TAB}s and {CR}s will be ignored so that data may go over more than one line in a file.

e.g. `05 <90,3C,40,3C,0>↵`

will transmit a middle c Note On and Off to a unit connected to Output 5.

`03 <C0,0,C,5>↵`

will transmit a Program Change 1, MIDI Channel 1 and Program Change 6, MIDI Channel 3 to Output 3.

While this command is in use normal connection is suspended to the selected Output and incoming data is lost. There is no limit to the number of bytes transmitted so it may be used to send large System Exclusive blocks and complex equipment setups. All Recall commands (see below) save data in this format. A series of these commands in a file for all outputs can completely (or partially) configure all connected MIDI equipment in a setup.

CLNN #MM - CLEAR OUTPUT NN WITH EVENT LIST MM

Each output may have an Event associated with it that will be sent to connected equipment before any patch changes that would result in that equipment becoming disconnected from a source take place, i.e. selecting a new input or Stopping the output.

On typing CL (C is ambiguous) the screen will show:

`Clear output: nn`

where nn is the last output addressed by a command. On entering or defaulting the

output number the screen will show:

Clear output: nn with Event #: mm

where **mm** is the event number already set up for that output. If **mm** = 0 no event will be used. A new event number may be entered or {CR} will leave the setup intact.

This action is not only required by keyboards and sound modules to prevent droning notes. Effects units and MIDI data processors that map different controllers to various functions may also require setting to a safe state.

I - INPUT COMMANDS

Inn	- Show Input name and connections
Inn "...	- Name Input
Inn <ID>	- Show next System Exclusive block from Input nn
Inn #mm <...>	- Capture Event mm from Input nn
Inn H	- Monitor MIDI data in Hex
Inn E	- Monitor MIDI data in English
%M<hh>n	- Filter Monitor Status hh (n: 0 = off, 1 = on)

Typing **I** will produce on the screen:

Input: nn

were **nn** defaults to the last input addressed by a command. A different number may be entered or if the default is correct simply type {CR} and the source and destination names will be shown following as a comment.

E.g. **I7↵**

might result in:

Input: 7 ;CZ101 not connected

or

**Input: 7 ;CZ101
; to 1 Fairlight III**

with any further connections shown in the same format on following lines.

INN "... - NAME INPUT

Up to sixteen printable characters may be entered following the quote. Further characters or trailing comments will be ignored. Delete and Backspace both delete backwards and the string may be aborted by typing {Esc} or entered by typing {CR}. This name will be used in displaying all connections with this input.

INN <ID> - SHOW NEXT SYSTEM EXCLUSIVE BLOCK FROM INPUT NN

This command really belongs with the MIDI Recall commands and is a special case for capturing system exclusive dumps from older equipment where the dump could only be manually initiated. After qualifying the input number with a < the manufacturer's ID should be entered in hex. An ID of "00" may be used for any ID, including three byte types.

The next single system exclusive block received from the specified input will be

captured (up to the maximum capture block size) and then formatted on the screen in ASCII hex within ◀ and ▶. This data may be copied to a file, but will require Onn entered before the ◀ to convert it into a send MIDI message command or a #nn to convert it into a define event command.

Once started only receiving the block or typing {Esc} to abort will end this command, {CR} will be ignored.

INN #MM <...> - CAPTURE EVENT #MM FROM INPUT NN

This command allows any MIDI data to be captured to define an event. After typing the input number and event number capture is started on typing ◀ and stopped on typing ▶ or the event memory being filled. Data is shown in hexadecimal on the screen as it is being captured. There is no abort for this command, once started the previous data for the event number is lost.

MIDI MONITOR The MIDIX Controller can monitor any MIDIX input and interpret the MIDI data into either hexadecimal or English on the terminal screen. Any MIDI Status and its Data may be selectively filtered to make the display easier to comprehend.

The commands function differently between the dumb and intelligent terminal modes as there are also dedicated VT100 pages for monitoring and filtering.

INN E - MONITOR INPUT NN DATA IN ENGLISH

If an E is typed after setting an input the screen will show:

```
Input:    1 English Monitor
```

and any MIDI codes received on that input will be displayed until another key is pressed.

If a keyboard is connected to the MIDIX input and a Program Change, slight Pitchbend movement or a slight Modulation wheel movement are made the screen will show something similar to:

```
Ch1:Program:    8,  
Ch1:Pitchbend:+ 260,  
Ch1:Pitchbend:    0,  
Ch1:Controller: 1/Modulation,    1,    1/Modulation,    0,
```

showing that the data is transmitted on MIDI Channel 1 and that the Pitchbend is transmitted with Running Status off while the Modulation has Running Status on.

Pitchbend and Song Position Pointer which both have two byte, 14 bit data are shown as a single decimal number, with Pitchbend centre as zero. All other data types are shown as transmitted. Please refer to our [MIDI Protocol Guide](#) for details of MIDI data structure.

INNE - MONITOR INPUT NN DATA IN HEX

If an H is typed after setting an Input the screen will show:

```
Input:    1 Hex Monitor  
<
```

and any MIDI codes received on that input will be displayed as hex bytes until another key is pressed. Note that the translation is preceded with a < and terminated with a > so that it may be copied into Event and Output Message definitions.

Using the previous example, if the same data were sent the screen would now show:

```
C0,07,  
E0,04,42,  
E0,00,42,  
B0,01,01,01,00,
```

which is exactly the bytes transmitted. A newline is inserted at every Status change to improve readability. Every nuance of the MIDI protocol is displayed as it is often violations of this that cause compatibility problems between equipment.

Note that for every MIDI byte received at 31k25Bd three RS232 bytes are generated at 38k4Bd. This may cause the data to lag behind with large bursts. Even more data is generated in English mode so use with caution.

%M<HH>N - FILTER MIDI MONITOR STATUS

To reduce the amount of data being processed by the MIDIX MIDI Monitor there are independent filters for every MIDI Status type. Any filter may be set to pass or remove a MIDI Status with:

```
%M<hh>n↵
```

where hh is the MIDI Status in hex and n is 0 for pass and 1 for remove. When a Filter is "on" it removes both that Status and its associated Data from the MIDI stream.

A special form of this command, %M<0>, will clear all filtering. hh must be between 80h and FFh and F0h operates both System Exclusive Start and End i.e. there is not an F7h filter.

e.g. To clear the Monitor Filters and then setup filtering for Polyphonic Aftertouch on Channel 5 and MIDI TimeCode:

```
%M<0>  
%M<B4>1  
%M<F1>1
```

R - MIDI RECALL™ COMMANDS

Rnn <ID> mm <hh, ... hh>

Rnn <ID> mm #nn

Rnn <ID> mm <hh, ... hh>

Recall from Input nn the next System Exclusive block with ID after sending on output mm the MIDI message <hh, ... hh>. On typing R the screen will show:

```
;Recall from Input: nn
```

where nn is the last output addressed. The command has been preceded with a semicolon turning the line into a comment so that it is safe to store to disc without being interpreted as a command when reloaded. The input number may now be entered for the equipment that the data dump is expected on. The system exclusive ID is then entered in hex, an ID of "00" may be used for any ID. The screen may then show:

```
;Recall from Input: nn <ID = 0> after sending Output: nn
```

the output number will default to the same number as the input, but if equipment is not connected to the matrix on a one to one basis a different number may be entered.

The command may now be qualified with a free format ASCII hex string containing the system exclusive dump request data. Information should be provided in the equipment manuals on how to request data dumps. If in doubt please contact Hinton Instruments technical support.

The MIDI message will be sent exclusively to the specified output. Any outputs connected to the dumping equipment will not be Stopped to allow programmed transfers of data between different units using the matrix. If this is not intended make sure such outputs are stopped before issuing the recall command.

The next single system exclusive block received from the specified input will be captured (up to the maximum block size) and then shown formatted on the screen in ASCII hex within < and > preceded by an Omm to make it into an Output Message command. It is assumed that the block will be sent back to the same output as the request was transmitted on, but may obviously be edited offline for different use.

N.B. system exclusive blocks often include checksums so are not suitable for editing in ASCII format. As long as everything between an F0 and an F7 is kept intact different system exclusive blocks may be cut and pasted in the stored files.

Once the MIDI message is terminated with a > the screen will show:

```
. . . > Waiting . . .
```

and will time out after approximately 30 seconds if nothing is received and ring the terminal bell. This is the one situation where typing {Esc} will not abort a command which is necessary to ensure correct operation running from prepared files.

Rnn <ID> mm #nn

Recall from Input nn the next System Exclusive block with ID after sending on output mm the Event #nn.

This is very similar to the previous command except that one of the defined Event Lists may be used instead of the MIDI message.

Optional Four Input Merger Commands

A1n - Output 1 Active Sensing (n: 0 = off, 1 = on)
R1n - Output 1 Running Status (n: 0 = off, 1 = on)
Yn=s - Input n switch
%n=s - Filter n Bypass
%A<hh>n - Filter Input 1A Status hh
%B<hh>n - Filter Input 1B Status hh
%C<hh>n - Filter Input 1C Status hh
%D<hh>n - Filter Input 1D Status hh

A1n - Output 1 Active Sensing

n: 0 = off, 1 = on.

When set On a MIDI Active Sensing Clock (FEh) is automatically generated at the Merger Output approximately every 90ms whenever no MIDI data is being transmitted. This feature is essential for some MIDI devices that require Active Sensing and will shut down if not received within 300ms of the last MIDI byte received.

R1n - Output 1 Running Status

n: 0 - off, 1 = on.

When set On redundant Channel Status bytes are removed from the Merger Output data according to Running Status protocol to reduce possible accumulative delays. This is reset after any 90ms pause to ensure that Data does not become separated from its Status.

Yn=s - Merger Input Switch

n = A, B, C, D; s: 0 - off, 1 = on.

Each of the four Merger Inputs may be turned completely on or off.

Filter Commands To reduce the amount of data being processed and unnecessary delays each Input has independent filters for every MIDI Status type (except Active Sensing and Reset which are not merged). Any Filter set may be switched in or out with the following command:

%n=s, where n: A, B, C, D; s: 0 - off, 1 = on.

Any filter may be set to pass or remove a MIDI Status with:

%n<hh>s←7

where n specifies the set for Input numbers A, B, C, D, hh is the MIDI Status in hex and s is 0 for pass and 1 for remove. When a Filter is "on" it removes both that Status and its associated Data from the MIDI stream.

A special form of this command, Fn<0>, will clear all filtering on Input n. hh must be between 80h and FDh and F0h operates both System Exclusive Start and End i.e. there is not an F7h filter.

e.g. To clear Input 1C Filters and then setup filtering for Polyphonic Aftertouch on Channel 5 and MIDI TimeCode:

```
%C<0>  
%C<B4>1  
%C<F1>1
```

Appendix

CONNECTORS

Multi-MIDI port: DB25S

	1	MIDI In 1+
14		MIDI In 1-
	2	MIDI Out 1+
15		MIDI Out 1-
	3	MIDI Out 1 Screen
16		MIDI In 2+
	4	MIDI In 2-
17		MIDI Out 2+
	5	MIDI Out 2-
18		MIDI Out 2 Screen
	6	MIDI In 3+
19		MIDI In 3-
	7	MIDI Out 3+
20		MIDI Out 3-
	8	MIDI Out 3 Screen
21		MIDI In 4+
	9	MIDI In 4-
22		MIDI Out 4+
	10	MIDI Out 4-
23		MIDI Out 4 Screen
	11	MIDI In 5+
24		MIDI In 5-
	12	MIDI Out 5+
25		MIDI Out 5-
	13	MIDI Out 5 Screen

RS232 port: DB25P

	1	DCD	Data Carrier Detect
6		DSR	Data Set Ready
	2	RxD	Receive Data
7		RTS	Request to Send
	3	TxD	Transmit Data
8		CTS	Clear to Send
	4	DTR	Data Terminal Ready
9		RI	Ring Indicator
	5	COM	Signal Ground

VT100 EMULATION

In Terminal Mode 2 the Controller uses a subset of the VT100 Intelligent Terminal protocol to provide a friendlier and computer independent user interface. The subset used has been chosen to give correct results with emulation software on computers not capable of simulating the dim, double size, underline or flashing characters.

To enable checking or emulation to be implemented the following code sequences are the only ones used:

VT100 CODES RECOGNISED

ANSI Mode Normal

Cursor Control Keys:

Cursor Up	{Esc}	[A
Cursor Down	{Esc}	[B
Cursor Right	{Esc}	[C
Cursor Left	{Esc}	[D

Auxiliary Keys:

PF1	{Esc}	[P
PF1	{Esc}	[Q
PF1	{Esc}	[R
PF1	{Esc}	[S

VT100 CODES USED

C0 (ASCII Control Characters:

BEL, BS, HT, CR, LF, FF, DC1, DC3, ESC
DC1 is used as Xon and DC3 as Xoff.

Escape Sequences:

Only 7-bit characters are used in both 7-bit and 8-bit environments. 8-bit Control Sequences are not used.

Hard terminal reset (RIS) {Esc} c

Cursor position (CUP) {Esc} [*pl* ; *pc* H
pl = line number, 1 to 24, decimal ASCII
pc = column number, 1 to 80, decimal ASCII

Set Top and Bottom Margins (DECSTBM)
{Esc} *pt* ; *pb* r
pt = top line number to be scrolled, 1 to 24, decimal ASCII
pb = bottom line number

Erase in display (ED) {Esc} [J
Erase in line (EL) {Esc} [K

Save cursor (DECSC) {Esc} 7
Restore cursor (DECRC) {Esc} 8

Text cursor enable (DECTCEM)
Set on {Esc} [? 25 h
Set off {Esc} [? 25 l

Select Graphic Rendition (SGR)
All attributes off {Esc} [
Display negative {Esc} [7 m

Index

A

Active Sensing 2.11, 3.22

B

Block Delay 2.11

C

Capture Event 3.18
Chains 2.4
Chain Commands 3.5
Clear 3.16
Clear Events 2.2, 2.10
Commands Summary 3.2
Comments 2.8, 3.10
Compound Commands 3.2, 3.9
Config Page 2.11
Connectors A-1

D

Define Chain List 3.12
Define Event List 3.14
Define Illegal Group 3.13
Double Character Commands 3.2, 3.6
Dumps 2.4

E

Earthing 1.2
Event Commands 3.3, 3.14

F

Filters 2.6, 3.19
Filter Page 2.13
Front Panel 1.4

H

Help 2.8, 3.5
Hold 2.5, 3.5

I

ID string 3.10
Illegal Groups 2.2
Input Commands 3.3, 3.17
Input Exclusive 2.7, 3.17
Interrogate Version/Revision 3.4

K

Keypad Operation 2.9

L

Load Patch 3.11

M

Memory Protect 2.1, 2.10
Merger Filters 2.13, 3.22
Merger Input Switch 3.22
Messages 2.6
MIDI Cables 1.2
MIDI Implementation Chart 2.15
MIDI Monitor 3.18
MIDI Recall 2.7, 3.20
MIDI Recall Commands 3.3, 3.20
Monitoring 2.6
Monitor Filter 2.13, 3.19
Monitor Page 2.12

N

Name Event List 2.2, 3.14
Name Input 2.1, 3.17
Name Output 2.1, 3.15
Name Patch 2.4, 3.12
Nuts 1.1

O

Optional Merger Commands 3.3, 3.22
Output Commands 3.3, 3.15
Output Event 3.16
Output Message 3.16
Output Panic Sequence 3.16

P

Panel Lock 2.10
Panic ! 2.5, 3.4
Patches 2.3
Patch Control Commands 3.3, 3.11
PF Keys 2.11, 3.10

R

Rear Panel 1.5
Recall/Compare 3.12
Recall Patch 3.11
Routing 2.3
RS232 Cables 1.2
Running Status 2.11, 3.22

S

Select Input/Stop 3.15
Send Event List 3.16
Send MIDI message 3.16
Show Event List 3.14
Show Input 3.15
Show Status 3.4
Single Character Commands 3.2, 3.4
Status Page 2.10
Store Patch 3.11

T

Terminal Modes 2.8
Troubleshooting 2.14

V

VT100 codes A-2
VT100 Mode 2.8

W

Wait 3.10