

## MIDI / USB DECODER MODEL 225e

## EDIT FACILITY

MIDI  
inINTERNAL  
BUSSESA  
B  
C  
DMIDI  
resetremote  
enable

bus select

CONTROLLER  
BUSSES (a)

J

K

L

M

N

P

(b)

## NOTE MESSAGE BUSSES

pitch velocity note on  
E  
F  
G  
H

## PRESET MANAGER

current  
preset  
not  
storedcurrent  
storerecall  
stored

last next

global  
name  
select

DATA

225e MIDI Decoder / Preset Manager  
Purchase through authorized Buchla Dealers.

The 225e MIDI Decoder / Preset Manager translates incoming MIDI messages into analog voltages and pulses, which appear on three kinds of busses for subsequent distribution to 200e-series modules.

Four internal busses route signals to those 200e series modules that respond directly to MIDI messages. Each bus conveys messages appropriate to the destination module: Pulses are routed to a 281e, velocities to a 292e, and pitches are directed to up to four 259e or 261e modules.

Four note-message busses decode the incoming MIDI signal and convert it into voltages, accessible from the 225e front panel, that represent pitches, velocities, and pulses with sustain information.

Six Controller busses are provided, each supporting two independent controllers. Control information is output as standard (10V range) control voltages.

Controller output bus P(b) is a dedicated MIDI Clock output.

Any number of busses can be assigned to a single MIDI channel. Note messages so assigned can be directed to respond polyphonically.

Built into the 225e, but independent of the MIDI Decoder, is the Preset Manager, which is used to store and recall the settings of nearly all system variables.

The 225e can also poll and display the current firmware version of any 200e-series module in a system and, in conjunction with a Macintosh computer running OSX 10.5 or above, can be used to download / update 200e systems with the latest firmware.

SKU: 225e MIDI Decoder / Preset Manager Category: Modules

## **Description - model 225e MIDI Decoder / Preset Manager**

The 225e MIDI decoder translates incoming MIDI messages into analog voltages and pulses. These control signals appear on three kinds of "busses" for subsequent distribution to 200e series modules. Four busses are routed internally to 200e series modules that respond directly to MIDI messages. Four more handle only note messages, decoding them into front panel voltages that represent pitches, velocities and pulses with sustain information. An additional six busses handle controller and clock messages (two controllers each), which are translated into standard 0-10 volt control voltages.

### **Busses: Note and Controller message assignments to Pulses and CVs.**

Note messages are applied to busses A,B,C,D,E,F,G, and H;

Continuous Controller and Clock messages are handled by busses J, K, L, M, N, and P.

Each bus responds to one selected MIDI channel. But more than one bus may be assigned to the same channel, allowing multiple controllers CVs and note messages to be derived from a single MIDI channel. First select a bus for assignment use "bus select" to illuminate the LED next to the desired bus.

### **MIDI NoteOn and pitch translation: Busses A-H**

There are two sets of Note messages busses: A-D and E-H. Internal busses will be explained in more detail later. For these busses the following display will appear:

<b>Bus:A Ch:<u>15</u> <u>P</u>oly</b> <b>Tr:24 Ft:75 V1: /</b>
---

With the cursor buttons, select a data type. Use the knob to the right of the cursor buttons to enter data for the following fields:

**Ch: MIDI Channel.** The first is the channel number; Incoming MIDI messages on the selected MIDI channel 1-16 will be transmitted on this bus. Channel number 0 leaves the bus inactive.

**Poly/Mono: (note on)** If Mono is selected in this field, then new note messages will cut off the previous notes and new pulses or notes will be generated out the assigned bus. This is the default and best if you have a limited number of oscillators or want to assign each oscillator to a specific MIDI channel. But if you

enable polyphony, the bus operates polyphonically between A-D or E-H as notes overlap. In Poly mode, up to four note messages (note on, pitch, and velocity) are distributed between A-D or E-H busses but ONLY IF they are addressed to the same MIDI channel number.

**Tr: (or Xpose:) Transpose (pitch).** This field causes incoming MIDI note-on messages to be transposed in semitones (chromatically) by as much as + or – 2 octaves. (-24 to +24).

**Ft: Fine Tuning (pitch).** is for fine tuning. This field is applicable to only internal busses A-D. The number 0 translates to the standard A = 440 Hz. “An” means the fine tuning will be done with the fine tuning knob on the oscillator itself. Numbers 1-99 translate upwards in cents. (FYI 100 cents equals one semitone)

**Vi: Velocity.** MIDI note messages include velocity information. Normally, this information is sent unaltered, but if desired, non-linear transformations can be applied to velocity. Velocity curves are displayed graphically, with the normal linear response displayed as /, and other responses displayed as slanted or bent lines. Reversed velocity responses are particularly useful for implementing voltage controlled cross-fading.

That covers the display fields. Now to clarify what an internal bus is.

### **Busses A-D: Internal Busses**

The A, B, C, and D busses are dedicated internal busses. All modules within a system share internal communication and some of these modules will respond to note, controller, and clock messages that are appropriate to their function. These modules are the 259e, 261e, 281e, 281h, 292e, 292h 223e, 250e, 251e and 252e. Additionally, the 226h can also send note messages on these busses.

### **Patching vs internal bus:**

Using patch cables on the 225e it is typical to patch noteon pulses using busses E-H (out the red banana jack) to the pulse input of a 281e, pitch CV (out the blue banana jack) to an oscillator, and velocity CV (out the green banana jack) to the velocity input on the 292e.

But you can save your patch cables if you use the internal bus. The internal bus sends noteons to the 281h, velocity to a 292e and accurate pitch information to oscillators such as the 259e or 261e. See below for how to do this.

### **Assigning the internal bus.**

Up to four 259e or 261e Complex Oscillators can be assigned to the four internal MIDI busses A – D. Plus each of the four sections of one 292e Quad Dynamics Manager and one 281e Quad Function Generator can be assigned these A - D busses if their module ID is set to A.

The MIDI messages recognized and potentially routed to appropriate modules on the internal bus include:

- 1) Note On and off messages to 281e/h (if Module ID A)
- 2) Velocity information to the 292e/h (if Module ID A)
- 3) Pitch/Note number information to all 259e's and 261e's according to their module ID. This includes PitchBend Messages.
- 4) MIDI Clock Messages, sent to the 223e, 250e's, 251e, and 252e's

*To repeat: the "module ID" used to distinguish multiple modules The oscillators must be assigned to the same bus name (A-D) as the Module ID. The 292e/h and 281e/h must be assigned to Module ID A. (Unlike the oscillators, the 292e and 281e the Module ID is not the same as the Bus name; but instead bus A-D corresponds to the top to bottom on one module.) To read more about module ID's and how to read them and change them, read "the module ID" section in the preset manager description below.*

### **Controllers: Busses J-P**

To select a bus for MIDI Continuous controller information or pitchbend or aftertouch select a buss from the range J - P, the following appears:

<b>Bus:J Channel:15</b> <b>Cntrlrs:PW:125</b>
--

Busses J has two outputs (a) and (b) corresponding to the two numbers after Cntrlrs: Select between PW (for "PitchWheel"/pitchbend), AFT (for Aftertouch) or Continuous Controller (CC) numbers 0-125.

*(Note: Typical preassigned CC numbers found on most keyboard controllers include 1 for Modulation/modwheel, 7 for volume, and 64 for sustain switches)*

When using PW keep in mind that pitchbend on channels assigned to busses A - D may be already routed to oscillator modules internally. (See description of internal busses.)

### **Clock:**

If Channel is set to "Clock" (after 16 and before 0) a different interface is displayed:

<b>Bus:J Chan:Clock</b>
<b>Div: 1/4 : 3/8</b>

The Div fields allow the 225e to generate 10 volt pulses based on rhythmic divisions of an incoming MIDI clock. The numerator represents a count; valid values are 1-8. The denominator represents a note length. Valid values are:

1:	Whole note	8:	Eighth note
2:	Half note	12:	Eighth note triplet
3:	Half note triplet	16:	Sixteenth note
4:	Quarter note	24:	Sixteenth note triplet
6:	Quarter note triplet	32:	Thirty-second note

A pulse is issued after every count of note lengths. In the example above, 1/4 means "every quarter note" and 3/8 means "every third eighth note".

### **Additional switches:**

**MIDI Reset button:** To restore all voltages and internal parameters to their nominal values, hit "MIDI reset". This also sends "all notes off" and "all controllers off" messages to all channels.

**Remote Enable:** To enable storage and retrieval of your settings as part of a preset, engage "remote enable".

The MIDI interface's settings may be locally stored and subsequently recalled by the accompanying preset manager.

To connect or disconnect this module from the central library function, push the remote switch. The lighted red LED signifies a connection.

## **The Preset Manager (225e or 206e):**

Sharing the same panel on a 225e or 206e, but completely independent of the mixer or the MIDI Interface is a facility dedicated to Managing Presets and to Firmware Updating. First, we'll deal with Managing Presets:

### **How it works:**

Each 200e series module has a built-in processor that can read that module's knob settings and switch positions. On command from the preset manager this data is stored in the module's LOCAL memory and associated with a preset number. That stored data is retrieved and the settings are restored when the module gets a recall command with that preset number.

Even though the modules preset parameters are stored on the module itself, if you have multiple modules--all with remote enabled--they respond as a global preset. A user can store up to 30 presets with each being retrieved by simply transmitting its number to all the modules in a system.

Resolving differences in visible knob settings with recalled patches:

Now obviously, a knob (without a motor) cannot indicate the value of its associated parameter. It has a mechanically (indicated) value, but what we experience sonically is the underlying (virtual) value most recently received from the manager. The dilemma is resolved by persuading the virtual value to "catch up" with the indicated value. When you move a knob, this virtual value will change faster or slower than normal, but in the same direction as you turned the knob, eventually causing the virtual and indicated values to agree. (Some modules implement an earlier version of the knob catcher, doing nothing until the user catches the underlying value. We'll update these.

Switch settings are simpler; LED's show the true status of each switch.

DATA MODE SWITCH:

Press the switch under the word "DATA" to cycle through the various programming modes and displays. Now choose "select".

### **Select mode: Storing the preset:**

Say you found that mouth-watering sound you want to save for posterity (or at least tomorrow's gig). Here's how to save it as a preset:

The first line of the LCD display will contain the current (last recalled) preset number followed by a name (if named). The second line points to a memory

location (from 1 to 30). Use the data wheel to select a memory location for your preset---and with the system's modules remotes enabled--press the store button.

If the 206e / 225e's "store" switch is pressed, the current state of modules with remotes enabled will be read into the current memory location. Parameters can then be altered by moving knobs or switches, but this will not alter the underlying stored values until store is pressed again. An LED indicates that the current preset is not stored.

### Selecting/Recalling a Preset:

Selecting a preset is equally simple. With remotes enabled, choose a desired preset with the data wheel and hit "recall". The chosen preset will appear in the first line of the display (the "current" preset), and your system will assume the appropriate state.

The switches "last" and "next" can also be used to instantly recalls the preset from memory.

Pulse inputs can also but used in this same way. (See "Pulse input modes" in global settings. This can be use to create chain of presets.

*Note: You can selectively decide NOT to store or recall parameters on a particular module by NOT having that module remote enabled. This is very useful when coming up with variations of one preset.*

### Name mode: Naming the Preset:

Once stored in memory, a preset may be named (or renamed). To name the preset, hit the mode switch, illuminating the "name" LED. Use the "last" and "next" switches to direct the cursor position, and the data wheel to select desired characters. When naming is complete, typically you return to "select". But let's explore the global setting.

### Global mode: Global settings and memory card functions

You can use the mode switch to display three displays of that global option that apply to the system. (*This section assumes you have the Primo version of the hardware and software.*)



Global settings: The first global display controls the setting and clearing of the remote status of ALL the modules in a system. The command is sent when the cursor is in the status field and the data wheel is turned. This is a convenient way to change the system status with a single knob. Use the "next" key to cursor to the Disp field and set the LCD contrast contrast for both the preset manager and the MIDI decoder for best viewing.

**Rem Sstats: Disp:2  
Base Channel: 16**

The next global entry is the base channel. The preset manager responds to MIDI program change messages received on the specified base channel. A blank here disables this response to program changes.

Pulse input modes: Push the mode switch again. You can now choose how the pulse inputs work. The simplest selection is <-Last\_Next->. Pulse inputs will act the same as the button press for "last" and "next". If instead you choose <-C.V.\_Samp-> then the left pulse input acts as a CV input for random access of presets based on the voltage when the left pulse input receives a pulse. (We are overripening the orange banana into functioning like a black banana. Your results may vary. It is recommended to use <-Last\_Next-> when possible.)

**<- Last Next ->  
Format Card**

### Memory card functions:

Formatting a card: A facility exists to store presets on a memory card. If the card is new, it must be formatted, accomplished by inserting it in the memory socket, moving the cursor to "format card" and rotating the data switch. Formatting a card also erases the card, so be careful when formatting cards.

Push the mode switch again: The next entry allows one to move presets or groups of presets. Pick the desired range for the source presets and the desired range for the target.

**16-30 to 1-15  
Copy Exchange**

Make your choice, choose “Copy” or “Exchange” , and turn the data wheel.

Backing up your system: By far the simplest procedure to preserve the settings in your systems is to do a full backup of all the parameters in modules and their associated presets Do this by using “Restore from Card” or “Backup to Card”.

<p><b><u>R</u>estore from crd</b> <b><u>B</u>ackup to card</b></p>
--

Why consider a backup of your system’s parameters:

- 1) in unforeseen events leading to theft or damage of a single module or system, you still have a means of recovering your work.
- 2) If you need to firmware reprogram a single module and want to save it’s saved contents before reprogramming.
- 3) If you want to share your awesome sequence with a friend, you can save the contents of your module as yours but in a different system—by backing up your single module and restoring that single module in the other system. Of course this will only work if it’s set to the same module ID (see “The module ID” section below for a description)

## **Software versions, Module ID’s and Firmware updating.**

Reading the software version and module ID: A sustained press on most module’s remote enable switch\* will display the module’s current software version on the preset manager’s LCD.

*\*Exceptions: There are some exceptions to the remote enable switch. One exception is the Preset Manager. Lacking a remote switch, we use the “select“ switch to accomplish the same function. For the 281h Use the quadrature switch. Likewise the 281e – this module has two computers, both requiring the same software. For the upper computer, push the “Quadrature A-B” switch, and for the lower use the “Quadrature C-D” switch. Another special case is the 256e. This module also has two computers executing identical software. Access the top one by pressing the “A” breakpoint switch and the bottom one by pressing the “C” breakpoint switch. The 285e is an exception with 2 computers but with different software– use the reference select switches to select the section you’d like to deal with.*

In addition to the firmware version, this action displays a module’s I.D. (if applicable)

## The module ID:

The preset manager and the internal bus communicate with each module independently. Each module knows what kind of module it is and what internal messages to listen for. But we expect multiples of some modules in a system and for those modules there also is a module ID. (Having up to four 261e's are the most obvious example.) The only way to tell these modules apart is with a module ID. It's important for backups to memory cards so modules can be independently identified. The previous section also describes how module ID's are important for assignment on the internal bus. DIP switches on the back of these modules allow a technician to easily change the module ID if it has one. If you bought the module as part of a system, these are set at the factory for you, but it can be altered by the user as necessary. Contact Buchla for instructions.

*Note about conflicting/shared module IDs: The parameters of a module are saved with that module and associated with a preset number. Similar modules with shared/conflicting module ID's will not overwrite each others saved parameter if a memory card is not used. And if a crazy and highly unusual system required conflicting module ID's—a system with MORE than 4 oscillators of the same type (e.g. five 259e's) in one system, or any multiples of modules without a module ID—you can still use them independently in a system. But to save the contents of each module on a memory card, but you'd have to turn remote enable off for the conflicting module(s) IDs and use a 2<sup>nd</sup> memory card.*

*Also note that some h-series modules like the 292h 281h are simply half of their e-series counterpart (e.g. 292e, 281e) and therefore need unique module IDs if the h-series and e-series versions are together in one system.*

## Firmware Updating: When to consider reprogramming your module:

Now that you know the software version of your module, you can compare your software to the software versions compiled on the Buchla website. Consider updating your module if you have an older software version.

Or perhaps something has occurred that has caused your module to go haywire.

## Updating a module's firmware:

**You can find the programs and more complete instructions for downloading and updating a module's firmware on the Buchla website under "downloads".** Below is only a summary of what you might expect to perform once you've successfully downloaded your software updates.

Updating a module's firmware involves shutting down the system and inserting the Firmware card. Then, while pressing and holding the target module's remote enable button/switch (or its substitute\*), turning on the system power. After two

or three seconds, release the switch. The new firmware will be transferred to the selected module. The procedure will take up to two minutes. When finished, the module will assume its initial state. (If you have a 292e, you may see indication of any programming within the system on its blue LED.) Wait up to two minutes or until the module's lights return to their normal state.

*See \*Exceptions to the remote enable are mostly for modules with 2 processors and therefore 2 programming procedures. The same switch that when held displays the module's firmware and module ID also substitutes for the remote enable switch in Firmware updating procedures. One exception: The 225h does not report its firmware version but firmware can be loaded by connecting the input of the gate-to-pulse converter on the upper left of the module to a voltage >2.5v (e.g., the output of the Euro-to-Buchla CV processor on the 225h with the pot offset set to 5v and banana cable from the blue "out" jack to the white banana jack).*