REAL WORLD INTERFACES

Hardware and Software — Design and Consulting

rwi@firstpr.com.au http://www.firstpr.com.au/rwi/

DX7 MOD 7.1F Addendum to Manual

Robin Whittle 11 October 1994

B-Mod-7.1-User-Manual-Addendum-2002-08-14--2021-02-11.doc 2012-09-26 updated 2021-02-11

2021-02-11 notes: Between 1987 and 1998 I worked on DX7s (the original ones) to install this modified Mod 7.1 firmware, extra memory and sound improvement modifications. One or two of the memory mods involved pushbuttons and a 7 segment LED display. The rest used the Mod 7.1 firmware to select the bank, via Function 12. I also modified ROM carts with an extra box built on to be 16 bank RAM carts. Originally these had NiCad batteries, but these turned out to be a really bad idea because they leaked. Those with lithium batteries have a CR 1/2 AA battery which should last for decades. The 16 bank memory systems had Panasonic BR 2/3 AE lithium batteries, which should also last for decades. I don't think I used NiCad batteries for the internal 16 memory bank systems.

I also sold EPROMs containing the Mod 7.1 firmware, for machines I did not work on myself. Generally these machines have only the Mod 7.1 firmware. Most of the machines I worked on myself have Mod 7.1 and the sound mods, and many of them have 16 banks of memory.

Each EPROM has a section with the owner's name and phone number encoded in a tricky manner which resists attempts to change it.

Please see **`https://www.firstpr.com.au/rwi/dx7/`** for this and the other documentation for the DX7 mods.

This addendum is intended for the August 89 version of the manual. If you are reading this for the first time, please read the main body of the manual and then refer back to this addendum. If you have just purchased MOD7.1 through the mail and wish to install it, please turn to page 3 of the original manual - which follows this addendum.

Version F of MOD 7.1 contains fixes for a number of bugs and three new features. The new features are :

1 Control over MIDI OUT transmission of Sustain Pedal messages.

2 Control over MIDI OUT transmission of Foot Modulation control change messages.

3 Reception of controller 6 to "drive" the DX7's data entry slider in Play, Edit and Function modes.

Bug Fixes

MIDI Garbage at Power Up

Previously, the DX7 and MOD 7.1 put out lots of note off events when it was powered up. This had the effect of crashing any Atari which the DX7 was plugged into and which was powered up at the same time. MOD 7.1B removed these note off events, but the MIDI line was driven low for several seconds at power up and it seems that the Atari erroneously interpreted this as a string of 00s.

MOD 7.1F fixes this. There is usually a spike on the MIDI OUT line when the power is applied, but it remains high and no spurious data is transmitted.

Key Velocity Anomaly

There was a problem when the Velocity Sensitivity parameter was set to 0, and a key was pressed very softly - the MIDI OUT note event would have a velocity of 126! I did not encounter this bug until recently and even disbelieved someone who reported it to me over a year ago.

Split Mode

If MOD 7.1 is put into Split Mode 3, it no longer makes a sound when you hit the keys, nor does it visibly respond to patch change buttons. If the machine was powered off and then on, then it would leave the owners name message on the display and remain silent. This caused many people to believe their DX7 to be dead, and I received far too many desperate calls about it. In many cases, the musician had put it into Split Mode 3 and forgotten about it. In other cases it got to 3 by accident.

The problem is clearly explained on page 22 of the manual, but the moral of the story is that a machine must never seem to be defective when it is turned on. I apologise to those who have missed a heartbeat minutes before a gig when they have fallen into this trap. Consequently, MOD 7.1 F sets Split Mode to 0 every time it is turned on.

You could still get into trouble if you left Function Mode "pointing" to Function 13 - Split Mode and moved the data entry slider or pressed the "+" button after you turned the machine on. The solution here is to get into the habit of selecting Function 14 "Battery Check" whenever you have finished doing what you want with any of the function parameters. This is the only way to ensure that a bump on the data entry slider will not upset something crucial.

"Not Used"

There were nine unused parameters in Function 13 - all of them called "NOT USED". Some people got lost in there and did not press "Internal Protect" enough times to get out. MOD 7.1C fixed that by stopping these from ever being selected.

New Features
Sustain Pedal MIDI OUT

Previous versions of MOD 7.1 transmitted the Sustain pedal on whichever channel was selected by the CONT L-U parameter. If you were transmitting to two MIDI channels with the two halves of the keyboard, then you could only send sustain pedal messages to one of those channels. I did make a special version which transmitted to both channels.

The new scheme gives you four options. The new SUST MODE parameter in function 13 controls whether the Sustain Pedal messages are transmitted on the two channels. SUST MODE is the very left hand (or top) parameter - just before VOLUME EN. Here are the four options.

SUST MODE	Sustain Pedal data sent out on -	
	Lower Channel	Upper Channel
0		
1	Yes	
2		Yes
3	Yes	Yes

The new Sustain Pedal scheme means that these MIDI OUT messages are independent of the CONT L-U parameter.

Foot Controller MIDI OUT

There is a common need for the DX7 to be able to send MIDI Volume commands to a slave device. The Volume Pedal of the DX7 is not connected to the internal computer in any way, so it is not possible to use it as a source of MIDI data. Instead I use the Foot Modulation pedal.

The Foot Modulation Pedal still has its original function inside the DX7 for the purpose of controlling internal sounds. The Modulation section of the DX7 - Mod-Wheel/Foot/Breath/Aftertouch - is a poorly understood part of the DX7 and I hope to explain it better in the new manual. MOD 7.1 is not involved in this area.

Whenever the Foot Modulation Pedal is moved, I take its new value and scale it because the actual output of the pedal is rather non-linear (at least in the case of the pedal I have). The pedal value can be sent as a control change to both the Lower Channel and the Upper Channel. The transmission to each channel can be disabled and most importantly, you can select which controller number it is sent as.

A MIDI Control Change message contains three items of information.

First Byte Specifies that this message is a Control Change on a particular MIDI channel.

Second Byte Specifies what "Type" of control change it is. This byte can be between 0 and 127, and this number is also called the Controller Number. Page 26 and 27 of the MOD 7.1 manual list some of the controller numbers which are generally used.

Third Byte Specifies the value of this Control Change. This is between 0 and 127.

Normally the Foot Modulation Pedal sends a control change 4, but I think it is only Yamaha instruments which receive control change 4. In the new scheme, you can make it send on controller 4 if you wish to keep its original function. However if you make it send a control change 7, then you can use this to control the volume of the MIDI device you are driving. Another possibility is control change 10 which is pan for many instruments.

Foot Controller MIDI Out is controlled by two similar parameters at the right (bottom) end of the Function 13 parameters - just past Slider No. They are called "FC LO" and "FC HI". Each one controls transmission to the MIDI channel of the Lower/Upper keyboard as specified in "LOWER CHAN" and "UPPER CHAN".

If you set "FC LO/HI" to 0, then no control change will be sent on this channel. If you set it to between 1 and 99, then a control change will be sent using the controller number you specify. This gives a great deal of flexibility and I am sure it will be useful for many musicians.

If the Lower and Upper Channels happen to be the same then there will be two control changes sent on that channel. If you set "FC LO" and "FC HI" to different numbers, then in fact you would be sending two different control changes to the one channel.

When you are sending controller 7 (Volume) in a sequencing situation, this could come back to the DX7 and control its volume. This may or may not be what you desire.

You could use the Foot Controller control changes - using Control Change 6 - to effectively drive the data entry slider of a MIDI slave device. Now the DX7 can receive Control Change 6 to perform the same operations as the Data Entry Slider - see the next section. The new Foot Controller scheme means that these MIDI OUT messages are independent of the CONT L-U parameter.

Data Entry Slider MIDI Receive

I had always assumed that when the DX7 received Control Change 6, the value (0 to 127) in that message would do the same job as moving the data entry slider to the equivalent position. I discovered that this was not the case - it only worked in Function Master Tune. Even here it did not really replicate the function of the slider - the value was written directly into the tune buffer.

MOD 7.1F can now receive Control Change 6 to do exactly the same job as moving the Data Entry Slider. This works for all Edit and Function parameters. When in Play mode - selecting and playing Internal and Cartridge voices - Control Change 6, like the physical slider, drives the last function parameter you selected. The only difference is that when a Control Change 6 is received, the Data Entry Slider operation is performed without transmitting a Control Change 6 to MIDI.

This feature opens up many possibilities for sequence controlled tweaking of Voice Edit parameters. Since the Foot Controller can now be made to transmit Control Change 6 messages, these could be sent back to the DX7 MIDI IN and used to do anything you can do with the Data Entry Slider.

This feature could also give you a severe dose of MIDIitis, because the incoming MIDI data could upset things in ways you were not expecting. For this reason, there is a new parameter in Function 13 to enable the reception of Control Change 6 for "driving" the Data Entry Slider. Furthermore, this parameter is set to 0 (disable) whenever the machine is turned on. The new parameter is called DES REC EN. It is the last parameter on the list.

Other Matters

CONT L-U

This parameter no longer effects the MIDI OUT transmission of Sustain Pedal and Foot Controller messages. It still selects which channel (0 = LOWER CHAN or 1 = UPPER CHAN) is used to transmit the following messages. I have shown the MIDI bytes (in hexadecimal) which make up each message. n stands for MIDI Channel and vv stands for the value which is being transmitted :-

Program Change	(Also called Patch Change)	Cn	vv	
Pitch Bend		En	vv	
Aftertouch		Dn	vv	
Mod Wheel	(Controller 1)	Bn	01	vv
Breath Control	(Controller 2)	Bn	02	vv
Data Entry Slider	(Controller 6)	Bn	06	vv
+ Yes Button	(Controller 96)	Bn	60	7F
- No Button	(Controller 97)	Bn	61	7F

MIDIitis

Just plugging one MIDI instrument into another can lead to a regrettable experience of MIDIitis. Plugging one or more instruments into a sequencer can lead to further confusion - especially if the sequencer is echoing its received messages to MIDI OUT. When the instrument has many options as MOD 7.1 then there is almost certain to be some drama at some time. Now that MOD 7.1 can cause the function of the data entry slider to be controlled from MIDI in, all sorts of unexpected things could happen.

There is no instant cure for MIDI problems - it is your situation and you have to figure it out. By plugging all these pieces of equipment together, you are playing the role of the "systems-integrator" (or "zoo-keeper"), and only you can resolve the problems you encounter when you put complicated pieces of equipment together. Here are a few pointers :-

THINK

> Think about what MIDI messages are being generated, and where they are going. Be particularly aware of MIDI loops caused by the sequencer echoing messages back to the machine that sent them - and to other machines.

ISOLATE THE DX7 FROM MIDI

> If you have any problems which seem to be occurring within the DX7, you can make sure they are not caused by MIDI loops or any other external equipment by isolating the DX7 from MIDI. You can do this by pulling out the MIDI IN and OUT leads, but an easier way is to hold STORE and press INTERNAL PROTECT to disable MIDI IN and OUT. Holding STORE and pressing CARTRIDGE PROTECT will re-enable them.
>
> Whenever you are doing something to the DX7 which is intended only for the DX7, it is a good idea to disable MIDI. This way the patch changes, data entry slider and "+/Yes" and "-/No" buttons cannot generate MIDI out messages which will upset other items of equipment - or the DX7 via some MIDI loop. If you ever find the "+/Yes" and "-/No" buttons moving parameters in increments of two, then this is a sure sign that you have a MIDI loop. Pressing the "+" button

moves the parameter up by one, the DX7 sends a control change 96, and receives it back via the MIDI loop which moves the parameter up by one again.

DIVIDE AND CONQUER!

The best way to debug any complex situation is to simplify it until the problem goes away. In a MIDI situation, you should remove items of equipment until the problem goes away, then find out what is the minimum arrangement where the problem persists. If you have a good think at this stage, and change the settings of the pieces of equipment which remain, then you will soon understand the cause of the problem and hopefully be able to devise a way of working around it.

Version History

MOD 7.1 Prior to 28-10-87, MOD 7.1 had some serious bugs. Only a few people had it and I think they have all been updated. This version was current until February 1988.

MOD 7.1A Merged received MIDI timing messages with the MIDI OUT data. I think only one person has this.

MOD 7.1B February 1988. This fixed the surge of MIDI bytes at power on, and I tidied up the code to provide as much space as possible for future extensions. There may have been one other small problem I fixed but I cannot remember it now.

MOD 7.1B1 Transmitted controller Sustain Pedal messages on both MIDI channels. One or two people have this version. The same thing can now be accomplished with MOD 7.1F.

MOD 7.1BFV Transmitted controller 7 messages on both MIDI channels when the Foot Modulation Pedal was moved. One or two people have this version. The same thing can now be accomplished with MOD 7.1F.

MOD 7.1BF A special version with a DAC and other circuitry to provide fast, smooth response to MIDI volume commands. There is only one of these - it is used by a clarinet player who uses a WX1 to control the DX7.

MOD 7.1C August 1989. This limited access to the Function 13 parameters so that no-one could get lost in the "NOT USED" parameters at the end of the list.

MOD 7.1D November 1991. SPLIT MODE is set to 0 at power on to avoid potential dramas. Several people have this and I will be updating them all to 7.1F.

MOD 7.1E I skipped this name to avoid confusion with the "E!" board from Grey Matter Response.

MOD 7.1F This version.

Limitations on possible features

MOD 7.1F is severely restricted by the limitations of the 16K byte EPROM chip which it must reside in. There is no extra space to put in any new features. This problem would not exist if I used a 32K EPROM, but this would entail a small circuit board and hardware modifications to the DX7 to make it work. I have elected to stay within these limitations to simplify the installation procedure of the chip. About half of the 130 MOD 7.1 users installed it themselves.

Function 13 Parameters

Use Internal Protect and Cartridge Protect buttons to scroll through the parameters.

Parameter Name	**Default Value**	**Range**	**Brief Description**	**See page**
SUST MODE	3	3	Enables transmission of Sustain Pedal messages :- 0 Nothing sent. 1 Lower 2 Upper 3 Lower Upper	A2
VOLUME EN	0	1	1 enables the interpretation of voice volumes.	14
# EN	0	1	1 enables the execution of # voices. Set it to 0 when you want to edit some # voices.	14
SPLIT POINT	60	97	MIDI note number of lowest key in upper section of keyboard. 60 is Middle C.	14
SPLIT MODE	0	3	Controls whether the DX7 plays notes locally when you play the keyboard. Set to 0 at power on. 0 Lower Upper 1 Upper 2 Lower 3 "Local Control Off". No notes are played locally and controller and patch change buttons drive MIDI but do not affect DX7.	15
LOWER CHAN	0	16	1 to 16 selects MIDI transmit channel for notes played below the split point. 0 selects the same channel as MIDI IN (func 8).	15
UPPER CHAN	0	16	As above but for keys on and above the split point.	15
CONT L-U	1	1	0 selects lower channel as the MIDI channel for transmission of controller and patch messages. 1 selects upper channel.	16
CHAR SET	0	2	Controls which set of characters to use when writing the voice name. 0 = normal. 1 = lowercase and special characters. 2 = weirdo random mode.	16
EXTRA NOTES	0	3	How many extra notes to play. This is for "normal" DX7 voices. Plus + voices have their own extra notes parameters and are not affected by this or the following three parameters.	16
X1 TUNE	53	99	Detune of the first extra note. 50 is normal. Each step is 1.17 cents.	16
X2 TUNE	44	99	Detune of the second extra note.	16
X3 TUNE	57	99	Detune of the third extra note.	16
STRETCH	0	8	Piano like stretch tuning.	17
PCH ADD	0	96	Number which is added to patch number in patch change message sent to MIDI out.	17
PCH OUT	1	1	0 disables the transmission of patch changes.	18
AFT OUT	1	1	0 disables the transmission of aftertouch messages.	18

Function 13 Parameters continued

Use Internal Protect and Cartridge Protect buttons to scroll through the parameters.

Parameter Name	Default Value	Range	Brief Description	See page
KEY SENS	10	19	0 is the standard keyboard sensitivity. 10 gives a better response with the full velocity range available.	18
PROG ADV	0	1	1 enables program advance with the portamento footswitch.	18
ANOFF IN	0	1	1 enables the reception of the controller 123 All Notes Off message.	19
SLOW TIMER	0	2	0 = normal, but patch changes are slower than a standard DX7. 1 = LFO and pitch envelopes run at 2/3 their normal speed but patch changes are fast. 2 = LFO and pitch envelops run at 1/2 speed and patch changes are even faster.	19
SLIDER No.	6	99	The data entry slider transmits a control change. This is the number of the controller. 6 is the standard controller number for the data entry slider.	20
FC LO	4	99	This is the controller number for the control change transmitted by the Foot Modulation Pedal, when it is being transmitted on the lower channel. 0 means do not send a control change at all.	A3
FC UP	4	99	As above but for the control change message sent on the upper channel.	A3
DES REC EN	0	1	1 enables the DX7 to receive control change 6 to perform the same function as the data entry slider. This is reset to 0 at power on.	A4