highly_liquid

MSA

Firmware Version 3.2

User Manual

Updated 2013-03-23

Additional documentation available at:

http://highlyliquid.com/support/

Table of Contents

© 2013 Sonarcana LLC

1.0 Overview

Users are encouraged to read ***MSA Hardware User Manual*** before continuing with this document.

1.1 About The MSA

The MSA is a multipurpose MIDI decoder. It can activate its 8 outputs in response to incoming MIDI messages of virtually any type.

MSA outputs can be controlled by either or both of the following:

- **Individual Output Control:** Each MSA output responds independently to MIDI input.
- **Preset Control:** The MSA stores up to 128 “presets”, each consisting of a compound output state that includes up to 8 individual outputs. Presets are recalled using program change, CC, or note commands.

To specify the desired behavior for either of the above control types, the user can employ one of two configuration methods:

- **“Learn” Method:** MSA behavior is specified via a mixture of switch input and MIDI messages.
- **SysEx Method:** MIDI System Exclusive (SysEx) messages are used to specify MSA behavior.

MSA configuration is retained when the MSA is disconnected from a power supply.

The MSA ships with a “factory default” configuration which can be described in terms of equivalent SysEx messages. See Factory Default Configuration.

1.2 About This Document

Throughout this document, SysEx message contents are shown in hexadecimal format. Hexadecimal numbers either are described as “hex” or are followed by the 'h' suffix. Decimal format should be assumed for all other numbers.

Sections of the manual that have changed significantly since the previous firmware version are highlighted in yellow.

1.3 Additional Resources

Support inquiries should be posted at the Highly Liquid Forum. Project examples and other information are posted at the forum.

http://forum.highlyliquid.com/

2.0 Individual Output Control

Each MSA output can respond independently to a specific MIDI message. Configure the MIDI response of each individual output via the Learn Method or SysEx Method.

2.1 Learn Method

A user-supplied "program" (PRGM) switch must be attached to the MSA "PRGM" terminals. If no switch is available, a piece of wire can be used to temporarily connect the terminals. See ***MSA Hardware User Manual*** for wiring details.

Use the following procedure to configure the individual outputs of the MSA:

1. Set the MSA DIP Switch for the desired MIDI channel.
2. Connect power to the MSA while pressing the PRGM switch.
3. Release the PRGM switch. The activity LED will blink slowly to indicate that the MSA is ready to learn.
4. Send a series of 8 MIDI messages to the MSA MIDI In port. As each message is received, the MSA output being assigned to the message will temporarily activate. Allow each output to deactivate before proceeding with the next message.
5. After the 8th message is received, the configuration is stored. To resume normal operation, disconnect, then reconnect power to the MSA.

Table 2-a shows the MIDI message types that can be learned. A single message type or a mixture of message types can be used. If different output modes are desired, the SysEx Method method must be used.

Table 2-a: Individual Output Configuration by Message Type

MIDI Message Type	Resulting Configuration
Note	Note Trigger
CC	CC On/Off
Program Change	Program Change "Match Only"

If the MSA does not receive 8 qualifying messages, any partially learned configuration will be discarded, and the prior configuration will remain.

The learn method can be repeated as desired. Configuration is retained when the MSA is disconnected from a power supply.

© 2013 Sonarcana LLC

2.2 SysEx Method

Figure 2-1 shows the format of the *Individual Output Configuration SysEx Message*. The message is a string of exactly 34 bytes. The body of the message consists of a Unit ID, a 3-byte configuration for each of the 8 MSA outputs, and a 2-byte pulse length.

Upon success, the MSA activity LED blinks 3 times and normal operation begins.

Figure 2-1: Individual Output Configuration SysEx Message

Section	Format (Hex)
Fixed Header	F0 00 01 5D 05 01
Unit ID	*id*
Output 0 Configuration	*mm d0 d1*
Output 1 Configuration	*mm d0 d1*
Output 2 Configuration	*mm d0 d1*
Output 3 Configuration	*mm d0 d1*
Output 4 Configuration	*mm d0 d1*
Output 5 Configuration	*mm d0 d1*
Output 6 Configuration	*mm d0 d1*
Output 7 Configuration	*mm d0 d1*
Pulse Length	*ph pl*
Fixed Footer	F7

2.2.1 Output Mode (*mm*) and Data (*d0, d1*)

Parameter *mm* specifies the output mode for the corresponding MSA output. For “normal” modes, the output is “off” by default and is activated in response to incoming MIDI events. For “inverted” modes, the output is “on” by default with an inverted MIDI response.

The meaning of bytes *d0* and *d1* depend on the value of *mm.*

Output modes are described in **Table 2-b**.

2.2.2 Pulse Length (*ph, pl*)

Certain output modes generate fixed-length output pulses. The length of the output pulse is set “globally” via the values *ph* and *pl*, according to the following formula:

$$\text{pulse length} = ((ph \times 128) + pl + 1) \times 0.5 \text{ ms}$$

The valid range for both *ph* and *pl* is 00h–7Fh. Thus, the maximum configurable pulse length is approximately 8.2 seconds. Actual pulse length is accurate to 0.5 ms of configured value.

Output modes which do not generate fixed-length pulses are not affected by the values *ph* and *pl*.

Table 2-b: Output Modes

***mm* (Hex)**		**Mode Description**	***d0* Meaning**	***d1* Meaning**
Normal	**Inverted**			
00	40	**Disabled** Output does not respond individually to MIDI input.	Ignored	Ignored
01	41	**Note Trigger** Output is "on" for the duration of the corresponding MIDI note with velocity greater than or equal to *d1*.	Note Number	Velocity Threshold
02	42	**Note Trigger – Fixed Length Pulse** Matching note on messages with a velocity equal to or greater than *d1*generate a pulse with length determined by the values of *ph* and *pl*.	Note Number	Velocity Threshold
03	43	**Note Toggle** Output state is toggled & latched upon the receipt of a matching Note On message.	Note Number	Velocity Threshold
08	48	**CC On/Off** Matching CC messages with a value equal to or greater than *d1* latch output "on". Matching CC messages with a value less than *d1* latch output "off".	CC Number	Value Threshold
09	49	**CC – Fixed Length Pulse** Matching CC messages with a value equal to or greater than *d1*generate a pulse with length determined by the values of *ph* and *pl*.	CC Number	Value Threshold
0A	4A	**CC Toggle** Output is toggled & latched upon the receipt of matching CC messages with a value equal to or greater than *d1*.	CC Number	Value Threshold
0B	4B	**CC "Match Only"** Output is latched "on" when a matching CC message is received. All other CC messages latch output "off".	CC Number	Ignored
10	50	**Program Change "Match Only"** Output is latched "on" when a matching program change message is received. All other program change messages latch output "off".	Program Number	Ignored
11	51	**Program Change – Fixed Length Pulse** Matching program change messages generate a pulse with length determined by the values *ph* and *pl*.	Program Number	Ignored
18	58	**Sync – Run** Output is latched "on" by MIDI Start or MIDI Continue messages, and is latched "off" by MIDI Stop messages.	Ignored	Ignored
19	59	**Sync – Clock** MIDI Clock messages generate pulses with length determined by *ph* and *pl*.	Ignored	Ignored

3.0 Preset Control

The MSA stores up to 128 “presets”, each consisting of an output state for up to 8 individual outputs. Presets are recalled using program change, CC, or note commands. The output state for each preset can be stored via the Learn Method or SysEx Method.

3.1 Learn Method

If desired, a user-supplied “program” (PRGM) switch can be attached to the MSA “PRGM” terminals. If no switch is available, a piece of wire can be used to temporarily connect the terminals. See ***MSA Hardware User Manual*** for wiring details.

Use the following procedure to store the output states for a preset:

1. Set the MSA DIP Switch for the desired MIDI channel.
2. Connect power to the MSA. Allow the activity LED self-test to complete.
3. Recall the preset to be stored using the corresponding recall message.
4. Change each MSA output to the desired state using MIDI messages for Individual Output Control.
5. Send the preset "store command" to the MSA using either of the following methods:
   a) Temporarily close the PRGM switch, or
   b) Send the MSA the MIDI "store command" as specified in the Preset Configuration SysEx Message.
6. The MSA activity LED will blink twice to acknowledge the storage of the preset state.
7. Additional preset states can be stored by repeating steps 3 - 5.

Important Note:

The following settings are determined by the Factory Default Configuration unless changed using the SysEx Method:

1. the type of MIDI message used to recall presets
2. the MIDI message used to store a preset ("preset store" command)
3. which individual MSA outputs are controlled via preset control

3.2 SysEx Method

Figure 3-1 shows the format of the *Preset Configuration SysEx Message*. Upon success receipt of the message, the MSA activity LED blinks 3 times normal operation begins.

Figure 3-1: Preset Configuration SysEx Message

Section	Format (Hex)
Fixed Header	F0 00 01 5D 05 02
Unit ID	*id*
Recall Type	*rr*
Store Command	*s0 s1*
Outputs Under Preset Control	*bb*
Preset State (Optional)	*nn k0 k1 k2 k3 k4 k5 k6 k7*
Additional Preset State (Optional)	*nn k0 k1 k2 k3 k4 k5 k6 k7*
. . .	. . .
Additional Preset State (Optional)	*nn k0 k1 k2 k3 k4 k5 k6 k7*
Fixed Footer	F7

3.2.1 Recall Type (*rr*)

Byte *rr* specifies which type of MIDI message will be used to recall the presets during operation. See **Table 3-a**.

Table 3-a: Recall Types

rr (Hex)	Message Type
01	Program Change
02	CC
04	Note

3.2.2 Store Command (*s0, s1*)

The *Store Command* is the MIDI message used to trigger storage of a preset state when using the Learn Method. *s0* specifies the message type as defined in **Table 3-b**. *s1* specifies the program number, CC number, or note number.

Table 3-b: Store Command Types

s0 **(Hex)**	**Message Type**
00	No MIDI store command.
01	Program Change
02	CC
04	Note

3.2.3 Outputs Controlled by Preset

Byte *bb* specifies which MSA outputs will be under both preset control and individual output control, and which MSA outputs will be under individual output control only. See **Table 3-c**.

Table 3-c: Outputs under Preset Control

bb **(Hex)**	**Outputs under both preset and individual output control**	**Outputs under individual output control only**
00	none	0, 1, 2, 3, 4, 5, 6, 7
01	0	1, 2, 3, 4, 5, 6, 7
02	0, 1	2, 3, 4, 5, 6, 7
03	0, 1, 2	3, 4, 5, 6, 7
04	0, 1, 2, 3	4, 5, 6, 7
05	0, 1, 2, 3, 4	5, 6, 7
06	0, 1, 2, 3, 4, 5	6, 7
07	0, 1, 2, 3, 4, 5, 6	7
08	0, 1, 2, 3, 4, 5, 6, 7	none

3.2.4 Preset State (*nn, k0-k7*)

Byte *nn* specifies the number of the preset defined by the following 8 bytes. *nn* corresponds to the MIDI program, CC, or note number used to recall the preset. The MSA can store up to 128 presets. The valid range for *nn* is 00h-7Fh.

Byte *k0-k7* specify the states of MSA outputs 0-7. For these bytes, a value of 00h sets the output to off, and a value of 01h sets the value to on.

© 2013 Sonarcana LLC

4.0 Factory Default Configuration

By default, MSA outputs 0-7 respond individually to MIDI notes 60-67. Presets are recalled by Program Change messages. The default configuration can be expressed as one Individual Output Configuration Sysex message and one Preset Configuration Message as shown in **Figure 4-1** and **Figure 4-2**.

Figure 4-1: Factory Default Individual Output Configuration SysEx Message

SysEx Data (Hex)	Meaning
F0 00 01 5D 05 01	Fixed header
00	Unit ID (00h = all units)
01 3C 01	Output 0 configuration
01 3D 01	Output 1 configuration
01 3E 01	Output 2 configuration
01 3F 01	Output 3 configuration
01 40 01	Output 4 configuration
01 41 01	Output 5 configuration
01 42 01	Output 6 configuration
01 43 01	Output 7 configuration
07 67	Pulse length: 500ms (Not used by factory default output modes.)
F7	Fixed footer

Figure 4-2: Factory Default Preset Configuration SysEx Message

SysEx Data (Hex)	Meaning
F0 00 01 5D 05 02	Fixed header
00	Unit ID (00h = All Units)
01	Recall type: Program Change
02 7F	Store command: CC #127
08	Outputs under both preset and individual output control: all
00 00 00 00 00 00 00 00 00	All outputs off for presets 0-127.
. . .	
7F 00 00 00 00 00 00 00 00	
F7	Fixed footer

5.0 Configuration Retrieval

5.1 Identity Request / Identity Reply

The MSA responds to the standard "Identity Request" SysEx command. If the MSA receives the SysEx message described in Figure 5-1, it will respond with the standard "Identity Reply" message described in Figure 5-2.

Figure 5-1: Identity Request SysEx Message

SysEx Data (Hex)	Meaning
F0 7E 7F 06 01 F7	Fixed Message

Figure 5-2: Identity Reply SysEx Message

SysEx Data (Hex)	Meaning
F0 7E 7F 06 02	Fixed Header
00 01 5D	Manufacturer ID
00 00	Family Code (MIDI Decoder)
00 05	Model (MSA Rev K)
00 *vv*	Version, where *vv* is the firmware version number as described in **Table 5-a**.
id	Unit ID
tt	Reserved for future use.
F7	Fixed Footer

Table 5-a: Firmware Version

vv (Hex)	Firmware Version
00	3.0
01	3.1
02	3.2 beta
03	3.2 production
04 thru 7F	Reserved for future versions.

© 2013 Sonarcana LLC

5.2 Output Configuration Retrieval

The current MSA configuration can be retrieved via the MSA MIDI Out port.

If the MSA receives a SysEx message as described in **Figure 5-3**, it will send its configuration as an Individual Output Configuration SysEx Message.

If the MSA receives a SysEx message as described in **Figure 5-4** or **Figure 5-5,** it will send its configuration as a Preset Configuration SysEx Message.

Figure 5-3: Individual Output Configuration Request SysEx Message

SysEx Data (Hex)	Meaning
F0 00 01 5D 05 00 00 01 F7	Fixed Message

Figure 5-4: Preset Request (Single Preset) SysEx Message

SysEx Data (Hex)	Meaning
F0 00 01 5D 05 00 00 02	Fixed Header
pp	Preset #
F7	Fixed Footer

Figure 5-5: Preset Request (128-Preset Dump) SysEx Message

SysEx Data (Hex)	Meaning
F0 00 01 5D 05 00 00 04 F7	Fixed Message

6.0 DIP Switch

The MSA DIP switch serves two main purposes: to set the MIDI channel for MIDI input, and to set the "Unit ID". See **Table 6-a**.

6.1 MIDI Channel

The MSA DIP switch is used to specify the MIDI channel for incoming MIDI messages. During normal operation, the MSA will respond only to messages received on this channel.

6.2 Unit ID

Each configuration SysEx message contains a *Unit ID* byte.

If the Unit ID byte is 00h, any MSA unit receiving the message will configure itself as directed by the message.

If the Unit ID byte is between 1 and 16 (01h to 10h), any MSA receiving the message will update its configuration only if its DIP Switch setting matches the Unit ID. This allows the user to target & configure a single MSA within a chain of multiple units.

Table 6-a: MIDI Channel / Unit ID DIP Switch Settings

MIDI Channel / Unit ID	**SW1 Setting**			
	1	**2**	**3**	**4**
1	off	off	off	off
2	off	off	off	**on**
3	off	off	**on**	off
4	off	off	**on**	**on**
5	off	**on**	off	off
6	off	**on**	off	**on**
7	off	**on**	**on**	off
8	off	**on**	**on**	**on**
9	**on**	off	off	off
10	**on**	off	off	**on**
11	**on**	off	**on**	off
12	**on**	off	**on**	**on**
13	**on**	**on**	off	off
14	**on**	**on**	off	**on**
15	**on**	**on**	**on**	off
16	**on**	**on**	**on**	**on**

© 2013 Sonarcana LLC

7.0 Activity LED

The MSA activity LED performs several functions:

- **Self test:** Upon power-up or device reset, the activity LED lights briefly before normal operation begins.
- **Configuration update indication:** The LED blinks 3 times when the MSA configuration is updated via a MIDI SysEx message. If no LED activity is observed, then the attempted configuration was unsuccessful.
- **MIDI activity indication:** The LED blinks when a MIDI message is received that affects the MSA output state. Other MIDI messages will not cause an activity indication.
- **Individual output control learn mode indication:** The LED will blink slowly while the MSA awaits MIDI messages during Learn Mode.
- **Preset store:** The LED blinks twice when a preset state is stored using the Preset Control Learn Method.
- **Firmware update error:** The LED blinks continuously when the receipt of a firmware update SysEx message has failed, or when the firmware has become corrupt.

8.0 Firmware Update

MSA firmware can be upgraded via MIDI SysEx message. Firmware update files, when available, can be downloaded from the MSA product page at highlyliquid.com.

8.1 Firmware Update Procedure

1. Download and unzip the firmware update SysEx message. The resulting file should have the extension ".syx".
2. Disconnect power to the MSA.
3. Press and hold the PRGM switch. (See ***MSA Hardware User Manual***.)
4. Reconnect power to the MSA. The activity LED will remain lighted.
5. Send the firmware update SysEx message to the MSA. Transmission of the message may require several seconds.
6. Upon completion of the update, the activity LED will deactivate.
7. Release the PRGM switch.
8. Disconnect power to the MSA.

8.2 Notes

- Do not interrupt the transmission of the firmware update SysEx message.
- A continuously blinking activity LED indicates an error in the transmission of the firmware update SysEx message.
- If a continuously blinking activity LED is observed, repeat the procedure from the beginning.
- The current firmware version can be retrieved via Identity Request / Identity Reply SysEx messages.