

LAUNCHCONTROL XL

Programmer's Reference



novation[®]
www.novationmusic.com

1 Introduction

This manual describes Launch Control XL's MIDI communication format. This is all the proprietary information you need to be able to write patches and applications that are customised for Launch Control XL.

It is assumed that you already have a basic knowledge of MIDI, and some appropriate software for writing interactive MIDI applications (for example, Max for Live, Max/MSP, or Pure Data).

Numbers in this manual are given in both hexadecimal and decimal. To avoid any ambiguity, hexadecimal numbers are always followed by a lower-case h.

2 Launch Control XL MIDI Overview

Launch Control XL is a class-compliant USB device that boasts 24 pots, 8 faders and 24 programmable buttons. The 16 'channel' buttons each contain a bi-coloured LED with a red element and a green element; the light from these elements can be mixed to form amber. The four directional buttons each contain a single red LED. The 'Device', 'Mute', 'Solo' and 'Record Arm' buttons each contain a single yellow LED. Launch Control XL has 16 templates: 8 user templates, which can be modified, and 8 factory templates, which cannot. User templates occupy slots 00h-07h (0-7), whereas factory templates occupy slots 08-0Fh (8-15). Use the Launch Control XL Editor (available on the Novation website) to modify your 8 user templates.

Launch Control XL has a single MIDI port named 'Launch Control XL n ', where n is the device ID of your unit (not shown for device ID 1). The button LEDs for any template can be controlled via System Exclusive messages. Alternatively, button LEDs for the currently selected template can be controlled via MIDI note-on, note-off, and control change (CC) messages, as per the original Launchpad protocol.

Launch Control XL uses a System Exclusive protocol to update the state of any button on any template, regardless of the currently selected template. In order to maintain compatibility with Launchpad and Launchpad S, Launch Control XL also adheres to the traditional Launchpad LED lighting protocol via note-on, note-off and CC messages. However, such messages will only be acted upon if the currently selected template contains a button/pot whose note/CC value and MIDI channel match those of the incoming message. Users are therefore advised to adopt the new System Exclusive protocol.

In addition, Launch Control XL also supports the original Launchpad double-buffering, flashing and set-/reset-all LED messages, where the MIDI channel of the message defines the template for which the message is intended. These messages can therefore be sent at any time, regardless of which template is currently selected.

The state of each LED is stored when the template is changed and will be recalled when the template is reselected. All LEDs can be updated in the background via Sysex.

3 Computer-to-Device Messages

LEDs on the Launch Control XL can be set via two different protocols: (1) the traditional Launchpad MIDI protocol, which requires the currently selected template to contain a button whose note/CC and MIDI channel correspond to the incoming message; and (2) the Launch Control XL System Exclusive protocol, which will update the required button regardless of its note/CC value or MIDI channel.

In both protocols, a single byte is used to set the intensities of both the red and green LEDs. This byte also includes the *Copy* and *Clear* flags. The byte is structured as follows (those unfamiliar with binary notation can read on for the formula):

Bit	Name	Meaning
6		Must be 0
5..4	<i>Green</i>	Green LED brightness
3	<i>Clear</i>	If 1: clear the other buffer's copy of this LED
2	<i>Copy</i>	If 1: write this LED data to both buffers Note: this behaviour overrides the <i>Clear</i> behaviour when both bits are set
1..0	<i>Red</i>	Red LED brightness

The *Copy* and *Clear* bits allow manipulation of the Launch Control XL's double-buffering feature. See the 'Control double-buffering' message and the Appendix for details about how this can be used.

Each LED can therefore be set to one of four values:

Brightness	Meaning
0	Off
1	Low brightness
2	Medium brightness
3	Full brightness

If the double-buffering features are not in use, it is good practice to keep the *Copy* and *Clear* bits set when turning LEDs on or off. This makes it possible to use the same routines in flashing mode without re-working them. A formula for calculating velocity values is:

$$\text{Hex version} \quad \text{Velocity} = (10\text{h} \times \text{Green}) \\ + \text{Red} \\ + \text{Flags}$$

$$\text{Decimal version} \quad \text{Velocity} = (16 \times \text{Green}) \\ + \text{Red} \\ + \text{Flags}$$

$$\text{where} \quad \text{Flags} = \begin{array}{l} 12 \quad (0\text{Ch in hex}) \text{ for normal use;} \\ 8 \quad \text{to make the LED flash, if configured;} \\ 0 \quad \text{if using double-buffering.} \end{array}$$

The following tables of pre-calculated velocity values for normal use may also be helpful:

Hex	Decimal	Colour	Brightness
0Ch	12	Off	Off
0Dh	13	Red	Low
0Fh	15	Red	Full
1Dh	29	Amber	Low
3Fh	63	Amber	Full
3Eh	62	Yellow	Full
1Ch	28	Green	Low
3Ch	60	Green	Full

Values for flashing LEDs are:

Hex	Decimal	Colour	Brightness
0Bh	11	Red	Full
3Bh	59	Amber	Full
3Ah	58	Yellow	Full
38h	56	Green	Full

Launchpad Protocol

Note On — Set button LEDs

Hex version	<i>9nh, Note, Velocity</i>
Dec version	<i>144+n, Note, Velocity</i>

A note-on message changes the state of all buttons in the currently selected template whose note/CC value matches that of the incoming *Note* value and whose zero-indexed MIDI channel matches the MIDI channel *n* of the incoming message. *Velocity* is used to set the LED colour.

Note Off — Turn off button LEDs

Hex version	<i>8nh, Note, Velocity</i>
Dec version	<i>128+n, Note, Velocity</i>

This message is interpreted as a note-on message with the same *Note* value but with a velocity of 0. The *Velocity* byte is ignored in this message.

Reset Launch Control XL

Hex version $Bnh, 00h, 00h$
Dec version $176+n, 0, 0$

All LEDs are turned off, and the buffer settings and duty cycle are reset to their default values. The MIDI channel n defines the template for which this message is intended (00h-07h (0-7) for the 8 user templates, and 08h-0Fh (8-15) for the 8 factory templates).

Control double-buffering

Hex version $Bnh, 00h, 20-3Dh$
Dec version $176+n, 0, 32-61$

This message is used to control the double-buffering state of the buttons. The MIDI channel n defines the template for which this message is intended (00h-07h (0-7) for the 8 user templates, and 08h-0Fh (8-15) for the 8 factory templates). See the Appendix for more information on double-buffering. The last byte is determined as follows:

Bit	Name	Meaning
6		Must be 0.
5		Must be 1.
4	<i>Copy</i>	If 1: copy the LED states from the new 'displayed' buffer to the new 'updating' buffer.
3	<i>Flash</i>	If 1: continually flip 'displayed' buffers to make selected LEDs flash.
2	<i>Update</i>	Set buffer 0 or buffer 1 as the new 'updating' buffer.
1		Must be 0.
0	<i>Display</i>	Set buffer 0 or buffer 1 as the new 'displaying' buffer.

For those less familiar with binary, the formula for calculating the data byte is:

Hex version $Data = (4 \times Update)$
 $+ Display$
 $+ 20h$
 $+ Flags$

Decimal version $Data = (4 \times Update)$
 $+ Display$
 $+ 32$
 $+ Flags$

where $Flags = 16$ (10h in Hex) for *Copy*;
 8 for *Flash*;
 0 otherwise.

The default state is zero: no flashing; the update buffer is 0; the displayed buffer is also 0. In this mode, any LED data written to Launch Control XL is displayed instantly.

Sending this message also resets the flash timer, so it can be used to resynchronise the flash rates of all Launch Control XLs connected to a system.

Turn on all LEDs

Hex version	Bn , 00h, 7D-7Fh
Dec version	$176+n$, 0, 125-127

The last byte can take one of three values:

Hex	Decimal	Meaning
7Dh	125	Low brightness test.
7Eh	126	Medium brightness test.
7Fh	127	Full brightness test.

Sending this command resets all other data – see the *Reset Launch Control XL* message for more information. The MIDI channel n defines the template for which this message is intended (00h-07h (0-7) for the 8 user templates, and 08h-0Fh (8-15) for the 8 factory templates).

Launch Control XL System Exclusive Protocol

Set LEDs

System Exclusive messages can be used to set the LED values for any button or pot in any template, regardless of which template is currently selected. This is done using the following message:

Hex version	F0h 00h 20h 29h 02h 11h 78h <i>Template Index Value</i> F7h
Dec version	240 0 32 41 2 17 120 <i>Template Index Value</i> 247

Where *Template* is 00h-07h (0-7) for the 8 user templates, and 08h-0Fh (8-15) for the 8 factory templates; *Index* is the index of the button or pot (see below); and *Value* is the velocity byte that defines the brightness values of both the red and green LEDs.

Multiple LEDs can be addressed in a single message by including multiple *LED-Value* byte pairs.

Indices are as follows:

- 00-07h (0-7) : Top row of knobs, left to right
- 08-0Fh (8-15) : Middle row of knobs, left to right
- 10-17h (16-23) : Bottom row of knobs, left to right
- 18-1Fh (24-31) : Top row of 'channel' buttons, left to right

20-27h (32-39) : Bottom row of 'channel' buttons, left to right
 28-2Bh (40-43) : Buttons Device, Mute, Solo, Record Arm
 2C-2Fh (44-47) : Buttons Up, Down, Left, Right

Toggle button states

The state of buttons whose behaviour is set to 'Toggle' (rather than 'Momentary') can be updated by System Exclusive messages. This is done using the following message:

Hex version	F0h 00h 20h 29h 02h 11h 7Bh	<i>Template</i>	<i>Index</i>	<i>Value</i>	F7h
Dec version	240 0 32 41 2 17 123	<i>Template</i>	<i>Index</i>	<i>Value</i>	247

Where *Template* is 00h-07h (0-7) for the 8 user templates, and 08h-0Fh (8-15) for the 8 factory templates; *Index* is the index of the button (see below); and *Value* is either 00h (0) for off or 7Fh (127) for on. Messages for buttons not set to 'Toggle' will be ignored.

Multiple buttons can be addressed in a single message by including multiple *Index-Value* byte pairs.

Indices are as follows:

00-07h (0-7) : Top row of 'channel' buttons, left to right
 08-0Fh (8-15) : Bottom row of 'channel' buttons, left to right
 10-13h (16-19) : Buttons Device, Mute, Solo, Record Arm
 14-17h (20-23) : Buttons Up, Down, Left, Right

Change current template

The following message can be used to change the current template of the device:

Hex version	F0h 00h 20h 29h 02h 11h 77h	<i>Template</i>	F7h
Dec version	240 0 32 41 2 17 119	<i>Template</i>	247

Where *Template* is 00h-07h (0-7) for the 8 user templates, and 08h-0Fh (8-15) for the 8 factory templates.

4 Device-to-Computer messages

Button pressed

Hex version	9nh, Note, Velocity
--------------------	---------------------

Dec version $144+n$, *Note*, *Velocity*

OR

Hex version Bnh , *CC*, *Velocity*

Dec version $176+n$, *CC*, *Velocity*

Buttons can output either note messages or CC messages on a zero-indexed MIDI channel n . A message is sent with velocity 7Fh when a button is pressed; a second message is sent with velocity 0 when it is released. The editor can be used to change each button's note/CC value and velocity value on press/release.

Template changed

Launch Control XL sends the following System Exclusive message out on changing template:

Hex version	F0h 00h 20h 29h 02h 11h 77h <i>Template</i> F7h
Dec version	240 0 32 41 2 17 119 <i>Template</i> 247

Where *Template* is 00h-07h (0-7) for the 8 user templates, and 08h-0Fh (8-15) for the 8 factory templates.

Appendix — LED double-buffering and flashing

The Launch Control XL has two LED buffers, 0 and 1. Either one can be displayed while either is updated by incoming LED instructions. In practice, this can enhance the performance of Launch Control XL in one of two ways:

1. By enabling a large-scale LED update which, although it could take 100 milliseconds to set up, appears to the user to be instantaneous.
2. By automatically flashing selected LEDs.

To exploit double-buffering for the first purpose requires very little modification to existing applications. It can be introduced in the following way:

1. Send $Bnh, 00h, 31h (176+n, 0, 49)$ on start-up, where n defines the template for which this message is intended (00h-07h (0-7) for the 8 user templates, and 08h-0Fh (8-15) for the 8 factory templates). This sets buffer 1 as the displayed buffer, and buffer 0 as the updating buffer. Launch Control XL will cease to show new LED data that is written to it.
2. Write LEDs to the Launch Control XL as usual, ensuring that the *Copy* and *Clear* bits are not set.
3. When this update is finished, send $Bnh, 00h, 34h (176+n, 0, 52)$. This sets buffer 0 as the displayed buffer, and buffer 1 as the updating buffer. The new LED data will instantly become visible. The current contents of buffer 0 will automatically be copied to buffer 1.
4. Write more LEDs to the Launch Control XL, with *Copy* and *Clear* bits set to zero.
5. When this update is finished, send $Bnh, 00h, 31h (176+n, 0, 49)$ again. This switches back to the first state. The new LED data will become visible, and the contents of buffer 1 will be copied back to buffer 0.
6. Continue from step 2.
7. Finally, to turn this mode off, send $Bnh, 00h, 30h (176+n, 0, 48)$.

Alternatively, chosen LEDs can be made to flash. To turn on automatic flashing, which lets Launch Control XL use its own flashing speed, send:

Hex version	$Bnh, 00h, 28h$
Dec version	$176+n, 0, 40$

If an external timeline is required to make the LEDs flash at a determined rate, the following sequence is suggested:

Turn flashing LEDs on	$Bnh, 00h, 20h$	(decimal version $176+n, 0, 32$)
Turn flashing LEDs off	$Bnh, 00h, 20h$	(decimal version $176+n, 0, 33$)

As mentioned previously, it is good practice to keep the *Clear* and *Copy* bits set while addressing LEDs generally, so that an application can easily be expanded to include flashing. Otherwise, unintended effects will occur when trying to introduce it later.