LSC LIGHTING SYSTEMS
(AUST) PTY LTD

Bldg 3, 66-74 Micro Circuit
South Dandenong VIC
Australia 3175

T +61 (0)3 9702 8000
F +61 (0)3 9702 8466

info@lsclighting.com.au
www.lsclighting.com.au

ABN 21 090 801 675

# LSC Eplate Editor User Manual

**Date:** 11 Feb 2012

**Revision:** 1.1

## 1. INTRODUCTION

LSC Lighting's ePlates are wallplates that allow the user to alter or modify the lighting states when using LSC Lighting’s EKO, TEKO or E24 dimming systems. Each of these dimmer models incorporate “LNet”, which is LSC Lighting’s network protocol. Each ePlate is highly configurable and can be user altered as required. The ePlate Editor is the tool for altering the ePlates functions.

## 2. GETTING STARTED

The editor operates on a 'Project' basis. Each Project can have as many ePlates as required. A Project can be thought of as an installation, or a building, or even a room within a building. Although there is no limit as to how many Projects are created, only one Project can be opened in the Editor at a time.

To start a new Project, click the New button. The Editor will prompt for the type of the first ePlate to be added to the Project - select the one that matches your hardware. At this point, the Project name, and ePlate name should be changed from 'New Project' and 'New ePlate' to something meaningful. This is achieved simply by clicking on the New Project or New ePlate node and typing the required name into the respective fields on the right hand side of the page. Pressing Apply will automatically update the names.

When saving projects, it is recommended that an empty directory is created for every new project. When saving, the editor will save the project file (*.ini) and all ePlates (*.xml) into the selected directory. If the directory is not empty, the editor will issue a warning. When saving modified projects, the editor will overwrite the project file and existing ePlate configuration files (*.xml).

## 3. Adding, Importing and Deleting ePlates

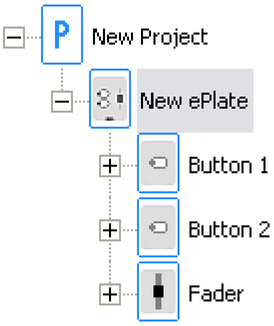
You can add a new ePlate to the project at any time by clicking the Add... button. You will be prompted to select an ePlate type.

You can also Import... an existing ePlate configuration (stored in an XML file on your hard disk or network location). The editor will automatically determine the ePlate type from the configuration file (if it has been properly constructed). While importing, the editor will try to correct any errors it encounters, and will issue an alert if any problems are found. Multiple ePlates may be imported at the same time.

You can also import a configuration from an existing ePlate already connected to LNet. Please refer to section 7.1 for further detail.

Any configured ePlate may be removed from the Project at any time by highlighting the unwanted ePlate and clicking the Delete... button. Please note that once deleted, all changes to that ePlate are lost.

## 4. Editing Properties



ePlates can have buttons, faders or a combination of both. Buttons and faders have their nodes displayed in the tree under the ePlate node. Buttons and faders detect events, and can perform actions in response to detected events. Event nodes are displayed under buttons and/or faders; action nodes are displayed under event nodes.

Each element of the Project (ePlates, buttons, faders, events and actions) can be re-configured by selecting the corresponding node in the tree. Once a node is selected, its alterable properties appear on the right hand side. No changes to node properties are applied until another tree node is selected or ✓ Apply button is clicked. Changes can be reverted before they are applied by clicking the ⊘ Discard button.

### 4.1 Clipboard

Properties of a tree node (and its sub-elements) can be copied to the clipboard and then pasted onto another node of the same type. The clipboard is accessed from the main Edit menu or by right-clicking on a tree node.

E.g. if a fader node is selected, its properties can be copied to clipboard (Edit → Copy), and then transferred onto another fader on the same or different ePlate (right-click on the destination fader and select Paste). Complete ePlates can easily be cloned using this method.

The editor will not allow pasting of incompatible properties. E.g. button copied to the clipboard cannot be pasted onto any other node type. 'Paste' function will be disabled unless information in the clipboard matches destination node.

# 5 Events, Actions, Operations

## 5.1 Events

When users press buttons or move faders on real ePlates, '**Events**' are generated. Events can be **Move** when a fader is moved; **Down** when a button is pressed; **Up** when a button is released, etc. Each button can be configured to trigger on 0 or more events. If a button is configured for 0 events, it will be inactive. Faders can only be configured for Move event.

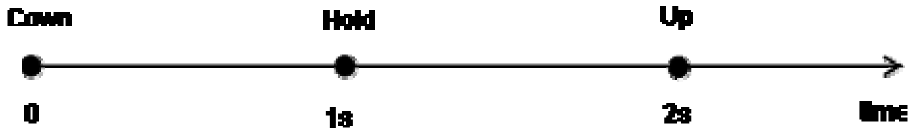
Different button Events are generated at various points in time from when a button is pressed to when it is released. The different types of events that can be configured are as follows:

- **Down** is generated when a button is pressed, regardless of when it is released
- **Up** is generated when a button is released, regardless of when it had been pressed
- **Click** is generated when a button is pressed and then released within half a second
- **Double click** is generated when a button is pressed down twice in quick succession.
- **Hold** is generated when the button is held down for one second, and every second thereafter until the button is released.
- **Click then Hold** is generated when a button is pressed and released within 0.5 sec, then pressed again and held for another second.
- **Click then Long Hold** is generated when a button is pressed and released within 0.5 sec, then pressed again and held for another three seconds.

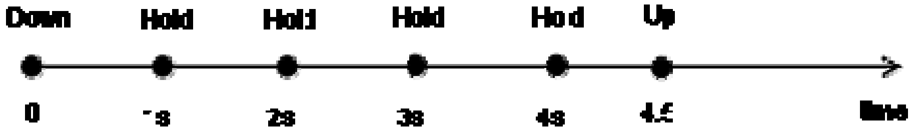
**NOTE:** Although there are many different options available, the most common button event is "Down", and this is the factory default for all buttons.

Some other examples of possible button events:

- if a button is quickly pressed and released, then Down, Click and Up Events would be generated.
- if a button is pressed, held down for two seconds, then released, the following events would be generated:



- if a button held down for 4.5 seconds, then released:



## 5.2 Actions

Buttons and faders can be configured to perform 1 or more '**Actions**' in response to each configured Event. There are three types of button actions:

- **Memory** - operates dimmer memories
- **Control Source Select** - selects whether dimmer outputs are controlled by dimmer memories or DMX. Also known as 'DMX Enable'
- **Lock Enable** - locks/unlocks other ePlates in the system

and three types of fader actions:

- **Memory** - operates dimmer memories
- **Net Master** - acts as master level control. Also known as 'Group Master'.
- **Time Master** - acts as master fade time control.

Normally, Actions would be set to 'Memory', however, multiple Actions are allowed. e.g. a button can be configured to activate Memory 1 when pressed (i.e. when 'Down' event occurs), and then to de-activate Memory 1, but activate Memory 2 when released (i.e. when 'Up' event occurs).

## 5.3 Operations

Each Action performs exactly one '**Operation**'. The available Operation choices are displayed via the drop down box and are context sensitive (i.e. depend on the Action being configured). Other action parameters are also context sensitive and will be displayed depending on the operation selected.

Possible Memory operations are:

- Set Level with Value.
- Set Level On - Turns Memory On.
- Set Level Off - Turns Memory Off.
- Toggle Level - Switches Memory On and Off
- Increment Level - Increases Memory Level by a fixed amount (typically 5%).
- Decrement Level - Decreases Memory Level by a fixed amount (typically 5%).

Possible Control Source Select operations are:

- DMX.
- Memory.
- Toggle - Switches between DMX and Memory

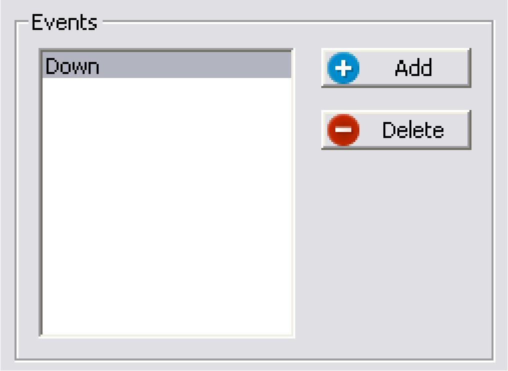
Possible Lock Enable operations are:

- On.
- Off.
- Toggle - Switches between Lock Enable On and Off.

# 6. ePlate Elements

ePlates can have a number of buttons and/or faders. Each ePlate button has an indicator located next to it. The number of buttons, indicators and faders depends on the ePlate hardware and cannot be changed.

## 6.1 Buttons



As described in 5.1. each type of button press can trigger a number of different Events. The most common one is the 'Down' Event, i.e. the button acts as soon as it is pressed. Several events can be configured for every button.

Events can be added or deleted from the list by pressing 'Add' and 'Delete' buttons. Remember that adding/deleting events to/from the list is not applied and not displayed in the tree until 'Apply' button is clicked (or a different tree node is selected).

Multiple combinations of Events are allowed (e.g. Down, Hold, Up).

### 6.1.1 Button Actions

Each button Event can trigger one or more 'Actions'. Button Action types can be 'Memory', 'Control Source Select' and 'Lock Enable'. Multiple Actions of the same type as well as mixing of different Action types is allowed.

Actions can be added or deleted from the list by pressing 'Add' and 'Delete' buttons. Remember that adding/deleting actions to/from the list is not applied and will not display in the tree until 'Apply' button is clicked (or a different tree node is selected).

### 6.1.2 Lock Enable

If an ePlate is required to be lockable by other ePlates in the system, it should be assigned a 'Lock Group Number' (on the ePlate's property page), and at least one button on another ePlate should be configured for 'Lock Enable' function and with the same 'Group Number'.

## 6.2. Faders

Faders can only produce and act on a 'Move' event. However, like buttons, each fader movement can trigger one or more 'Actions'. Actions can be 'Memory', 'Net Master' or 'Time Master'. Typically, a fader will only ever need to trigger one action.

## 6.3. Indicators

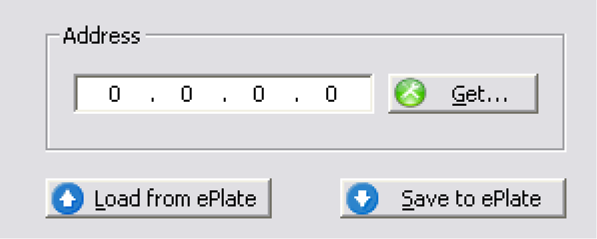
Indicators are usually configured to mimic the intensity level of the memory associated with the button (if only one Memory is associated with the button). In most cases, configuring the indicator is not necessary (and indicators therefore do not appear in the tree).The editor will by default configure each indicator to mimic its button.

However, if a button is configured for multiple events or multiple memory actions, the Editor may not assume the most logical Indicator behaviour. In such cases, the Indicator configuration can be altered by selecting the ePlate button whose Indicator needs to be edited and then clicking . This will allow you to set the Indicator to be associated with a specific memory number.

## 7. Uploading and downloading of configuration files

Downloading and uploading configurations directly to ePlates is achieved via an LSC program called CodeNet. The editor automatically starts CodeNet if required, and commands it to connect to LNet using default (or last used) set of parameters. If for any reason, starting CodeNet or connecting to LNet fails, CodeNet must be connected to LNet manually. For all operations described in this section, it is assumed that a connection to LNet is active through CodeNet.

Whilst CodeNet is running, the ePlate Editor can communicate directly with ePlates and perform both uploading and downloading of configuration files.



Each physical ePlate has its own unique address (similar to TCP/IP). Addresses for all configured ePlates can be filled in prior to connecting to LNet (e.g. off site). To upload/download a configuration, select an ePlate node in the editor, and fill in the address field if known. Once the address field is populated:

- clicking 'Load from ePlate' will get the configuration from the physical ePlate through CodeNet, and re-configure the current ePlate in the editor.
- clicking 'Save to ePlate' will send current configuration from the editor to the selected physical ePlate.

If unsure what the address of an ePlate is, clicking Get... button will display a list of addresses of all detected ePlates of the same type as the ePlate being configured in the editor (e.g. if a 4-button ePlate is selected in the editor tree, then only 4-button ePlate addresses are displayed). Select one to populate the address field.

### 7.1. Loading from ePlate

Loading from ePlates can be useful to quickly re-configure an existing ePlate in the building. To do so, create a new ePlate in the editor (e.g. 4-button ePlate), select the newly created ePlate node, populate the address field either by typing the address manually or by selecting one from the list obtained from LNet (using 'Get' button). Click 'Load from ePlate' button to retrieve current configuration from the  device, and apply it to the selected ePlate in the editor.

Hit 'Apply' button to apply changes.

### 7.2. Saving to ePlate

Saving to ePlate should be the last operation performed - when the configuration is complete and needs to be applied to the actual ePlate.

Select the ePlate in the editor's tree, populate address field and click 'Save to ePlate' button to start download process. The editor will send the configuration to the ePlate. This process can be observed in CodeNet.

### 7.3. Remote Control

CodeNet may reside on the same computer (http://localhost), or on a remote computer. CodeNet's host computer address can be changed in editor's Preferences. If using CodeNet on a remote computer, a TCP/IP link with port 18083 must be available, and the port open for access through any internet firewalls.

If CodeNet's host computer is remote, then CodeNet must be started manually. Once CodeNet is running, and editor can communicate with it, connection to LNet will be attempted. If that is unsuccessful, CodeNet must be connected to LNet manually.

## 8. Report

Once Project configuration is complete, the editor can produce a report in an easy to understand and view format. This report can be used to verify that the system is configured as expected and then printed or saved for future reference. When 'Report' button is clicked, the default web browser opens and the report is displayed. Please use your browser to save or print the report if required.

END.