Adding a USB device to control cameras

Ross Video's DashBoard™ platform allows you to turn any USB device, such as a joystick or game controller, into a camera controller. The camera control panel allows you to configure and control various camera functions. These include: camera selection, camera motion (pan, tilt, zoom, and focus), and paintbox controls (lens iris and shutter speed).

You can configure a USB device to control all camera functions, or you can use multiple USB devices to each control a subset of camera functions. There are limitations to how camera functions can be divided up amongst USB devices. You can also configure unassigned buttons on the USB device to perform an action, such as opening a DashBoard device view or panel.

You must complete the following tasks:

- **Connecting the USB device to DashBoard**
- **Configuring Axis Controls and Button actions**
- **Configuring buttons to open device views or panels**

For these procedures you will need the following:

- DashBoard™ Version 8.3 and later
- A USB device
- A pre-configured **.joystick** file (recommended)

If you want to perform a quick setup, you can download the pre-configured **.joystick** file provided by Ross Video. The file is available on the Ross Video website on the DashBoard product page.

You can load the configuration file in the procedure ***"Configuring Axis Controls and Button actions" on page 1-3***, and it will configure the joystick settings as if you are using an APEM VM Desktop joystick. If you are using another brand or model of USB device, the configuration file will also work and it will only add in the information for buttons that are available on your USB device.

Connecting the USB device to DashBoard

Before you get started, ensure that DashBoard is downloaded, and follow the steps below to add as many USB devices as you want to DashBoard.

To add a USB device to control your cameras:

1. Close DashBoard, and then connect the USB Device to a USB port on the computer.
2. Start DashBoard.
3. Click the **File** menu, and select **New > Other**.

   The **New** dialog box appears.

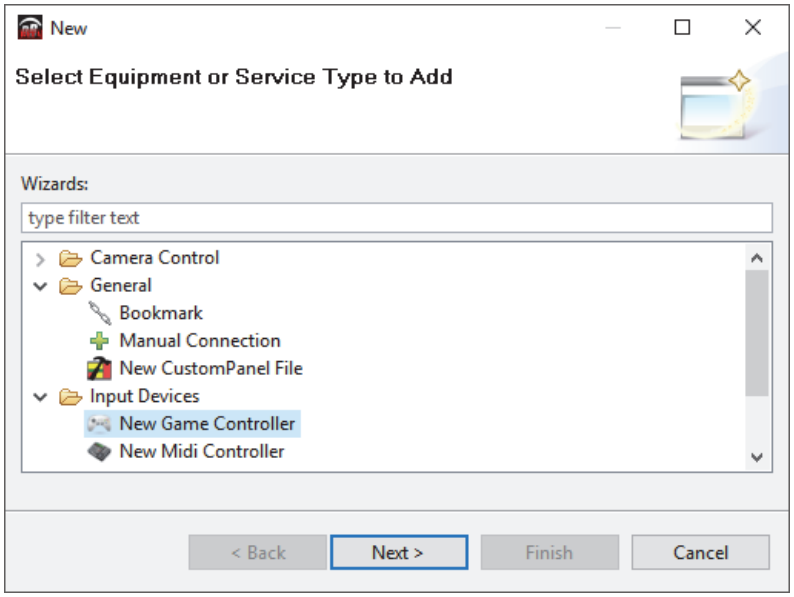


***Figure 1** - Adding a New Game Controller*

4. In the **Wizards** list, expand **Input Devices**, and double-click **New Game Controller**.

   The **New Game Controller Connection** dialog box appears.

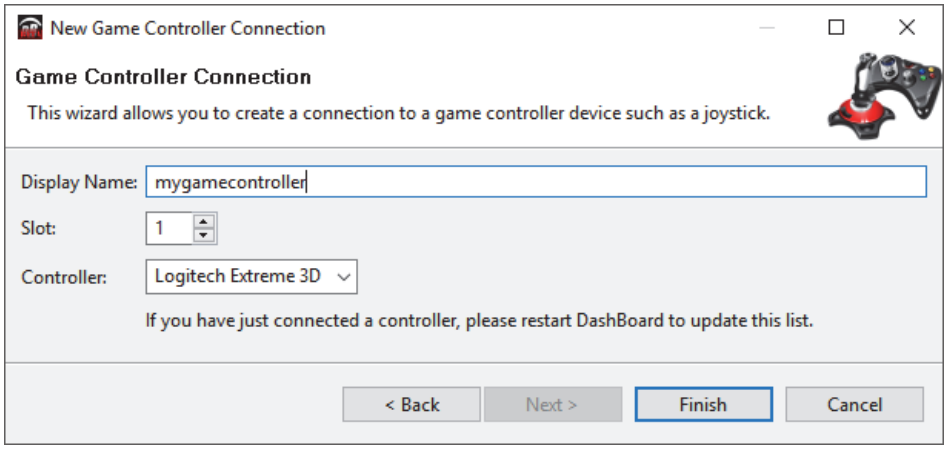


***Figure 2** - Connecting a USB Device*

5. In the **Display Name** field, enter a name for the controller.
6. In the **Slot** field, set a slot number.
7. In the **Controller** drop-down menu, select the USB device you connected earlier.

   Note: If the controller type is not listed, your USB device may not have been plugged into the computer before DashBoard was launched. Simply restart DashBoard to get it to recognize the connected USB device.

8. Click **Finish**.

   A node for the controller appears within the **Game Controllers** node in the **Basic Tree View**. The name of the node is the **Display name** you entered earlier.

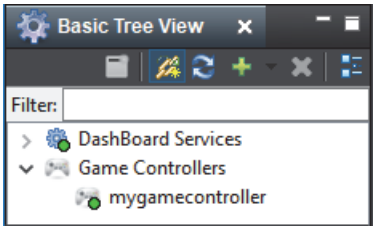


***Figure 3** - Basic Tree View*

9. In the **Basic Tree View**, expand the **Game Controllers** node, and then double-click the controller you added.

   The configuration interface for the controller appears.

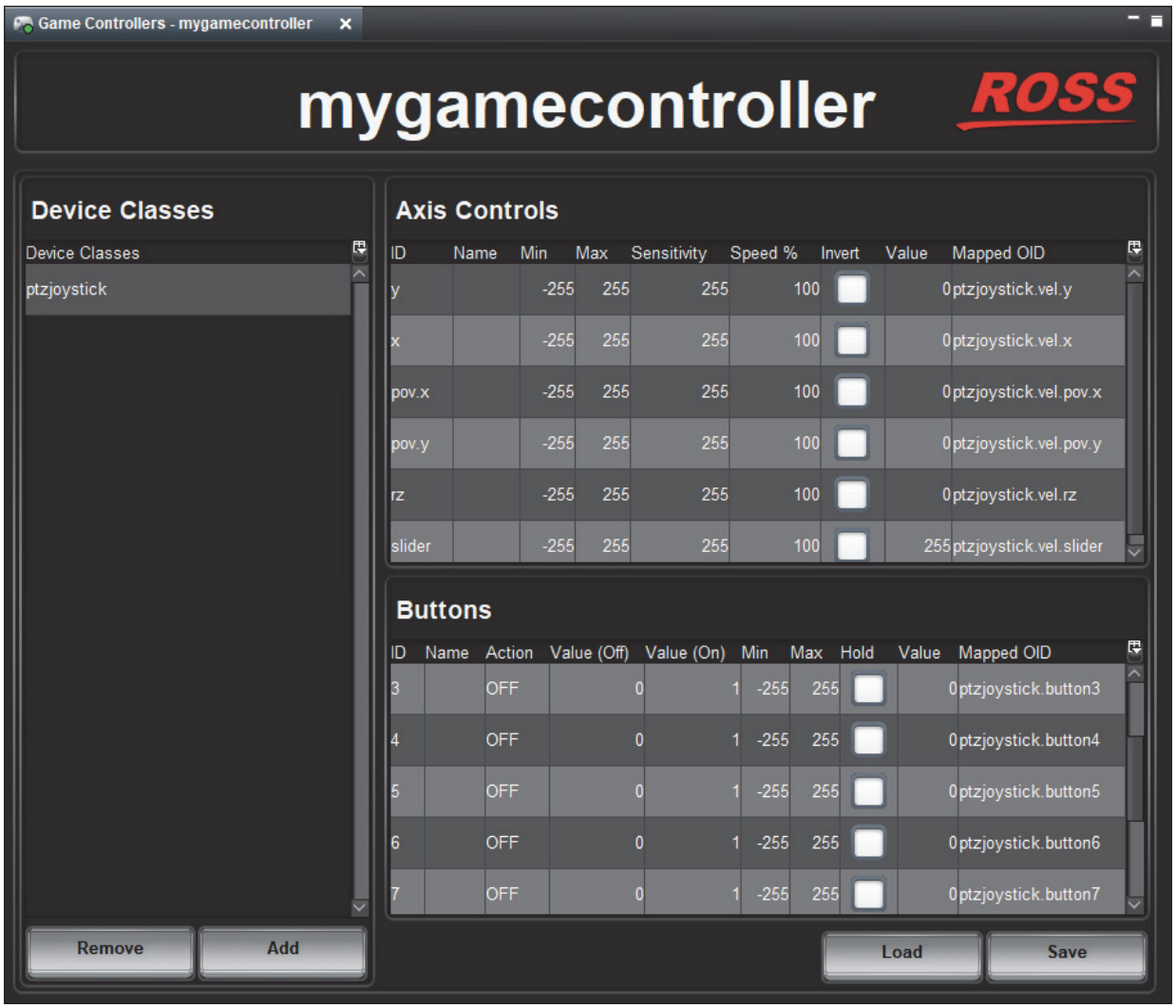


***Figure 4** - The Controller Interface*

Configuring Axis Controls and Button actions

First, you'll add device classes, and then you'll configure axis controls and button actions. Each device class defines controls for a specific subset of camera functions, such as **ptzjoystick** for joystick controls or **paintbox** for lens shutter and iris control. The following device classes are available: **ptzjoystick**, **paintbox**, **selector** and **accessories** (for outdoor cameras).

If you want to speed up your configuration, you can load the pre-configured joystick settings file that will automatically populate the Axis Controls table and Buttons table. Note that you must still follow the procedure “To add device classes:” on page 1-4. You may still wish to read **Step 1** to **Step 4** of the procedure “To create data mappings for the Axis Controls and Buttons” below, but when your data mappings are complete you can skip ahead to **Step 5**.

To add device classes:

1. In the **Device Classes** list, make sure the default device class, **ptzjoystick,** is included.
2. In the **Device Classes** list, add a device class named **paintbox.**
3. In the **Device Classes** list, add a device class named **selector**.

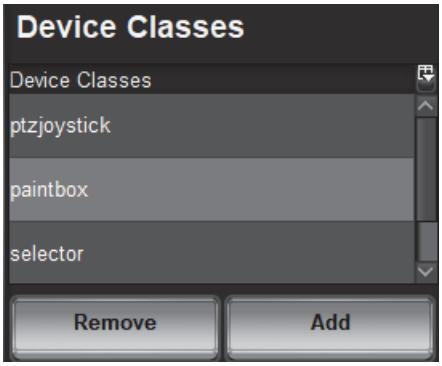


***Figure 5** - Adding device classes*

To create data mappings for the Axis Controls and Buttons

Create data mappings for the parameters you want to control with the controller. When you create data mappings, you link items reported by the controller to parameters (OIDs) in the DashBoard Camera Panel.

Note: If you are using multiple controllers and dividing the control functions among them, the controls within each device class must be assigned to a single controller. For example, you can’t share **paintbox** controls between two controllers.

1. Determine which **Axis Controls** and **Buttons** correspond to physical controls on the controller, by moving the controller’s joystick and pushing its buttons while observing changes in the **Value** column.

2. In the **Axis Controls** table, edit the configurable fields for each control you wish to configure:

ID	The ID of the control, as reported by the controller. Note: This is not configurable.
Name	Enter a display name for the axis control. (optional)
Sensitivity	Adjust the number value to change the responsiveness of the input.
Speed %	Adjust the number value from 1 to 100 percent to adjust the speed, where 100% is the fastest speed and 1% is the slowest speed.
Invert	Choose the **Invert** checkbox to reverse the direction of the joystick motion required to move the axis. Tip: If the camera is ceiling mounted, you would check **Invert** for pan and tilt so that when you move the joystick the camera moves as desired.

Value	Displays the current data value reported by the controller. Note: This is not configurable. Tip: This is useful for testing, because you can move the controller's joystick and press controller buttons to see which data items they correspond to in the mapping tables.
Mapped OID	Specifies the parameter OID from DashBoard Camera Panel to be mapped to the control. For example, for the **y** axis control you must enter **ptzjoystick.vel.tilt** as the parameter OID. For a complete list see, ***Table 1.1, "Mapped OID Parameters," on page 6***.

3. In the **Buttons** table, edit the configurable fields for each button you wish to configure:

ID	The ID of the control, as reported by the controller. Note: This is not configurable.
Name	Enter a display name for the button. (optional)
Action	You can set what action will occur when the button is pressed or released from the following options: **Off**, **Stateless**, **GPI**, **Set Value**, **Toggle** or **Increment**. Determine which option suits your requirements from the descriptions below. • **Off:** Choose **Off** if you do not want the value of the button to change when it's pressed. • **Stateless:** Choose **Stateless** if you want the value **on** when the button is pressed, and **off** when the button is released. • **GPI:** Choose **GPI** if you want to trigger a GPI when the button is pressed. This is useful for controlling external devices or executing a button on a DashBoard panel that is configured to accept GPI commands. • **Set Value:** Choose **Set Value** if you want the o provide a specific value to set the state of the DashBoard panel when a button is pressed, such as selecting a specific camera. • **Toggle:** Choose **Toggle** if you want the value to switch **on** or **off** when the button is pressed. • **Increment:** Choose **Increment** if you want the value to increase, or decrease, when the button is pressed.
Value (Off)	This is the value of the parameter when the button is not pressed.
Value (On)	This is the value of the parameter when you press the button.
Min	The minimum value. By default it is set to match the minimum value of the item being controlled.
Max	The maximum value. By default it is set to match the maximum value of the item being controlled.
Hold	Choose the **Hold** checkbox if you want the button to respond when you press and hold, rather than requiring you to press the button several times in rapid succession.
Value	Displays the current data value reported by the controller. Note: This is not configurable. Tip: This is useful for testing, because you can move the controller's joystick and press controller buttons to see which data items they correspond to in the mapping tables.
Mapped OID	Specifies the parameter OID from DashBoard Camera Panel to be mapped to the control. For example, for buttons that control camera selection you must enter **selector.selection** as the parameter OID. For a complete list see, ***Table 1.1, "Mapped OID Parameters," on page 6***.

The name of each parameter OID starts with the name of its device class, such as **selector**, **ptzjoystick**, or **paintbox**.

You can map the following parameters:

Table 1.1 Mapped OID Parameters

Parameter (Mapped OID)	Description
selector.selection	Camera selection. Map this parameter to the set of buttons you'll use to select the camera that's controlled. Each row in the **Buttons** table corresponds to a button on the controller. For each button you want to assign as the selector for a camera, do the following: • In the **Name** column, assign a camera name for future reference (optional). • Set **Action** to **Set Value**. • Set **Value Off** to **-1** • Set **Value (On)** to a camera number (**0**, **1**, **2**, **3**, etc.)
ptzjoystick.vel.tilt	Tilt axis
ptzjoystick.vel.pan	Pan axis
ptzjoystick.vel.zoom	Zoom axis (lens zoom)
ptzjoystick.vel.focus	Focus axis (lens focus)
paintbox.vel.shutter.preset	Shutter preset
paintbox.vel.iris.control	Iris control
paintbox.dnoise.reduction	Digital noise reduction

You can see an example in Figure 6:

***Figure 6** - Mapping USB Device Data to Camera Control Panel Parameter OIDs*

4. If you want to save the mappings in a file that can be loaded onto other DashBoard computers, click **Save > Save** and specify a file name and path. Click **Done**.

   Note: The mappings are specific to the controller(s) you configured. You can use the mappings on other computers running DashBoard only if the controllers are the same model, or report the exact same controls.

5. In the **Basic Tree View**, within the **DashBoard Services** node, double-click **Device Class Mappings**.

   The **Device Class Mappings** interface appears.

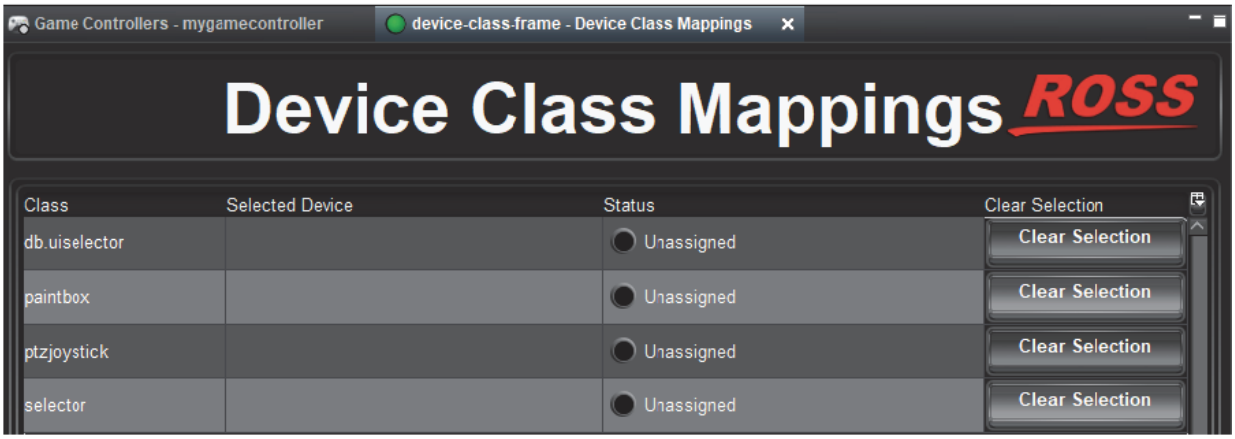


***Figure 7** - Device Class Mappings Interface*

6. If you plan to use the controller to adjust lens iris and lens shutter speed, in the **paintbox** row set **Selected Device** to the controller you are configuring.

   Tip: If a class you want to map is already mapped, click **Clear Selection** and then map it to the controller.

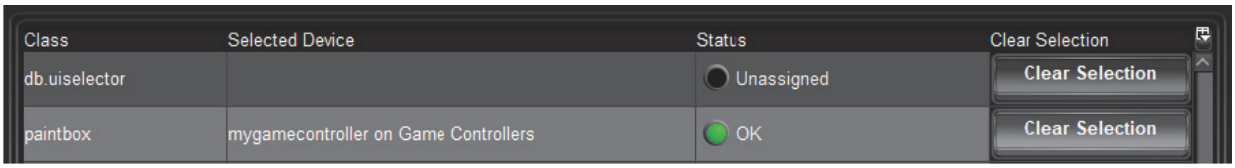


***Figure 8** - Adding Paintbox Mappings*

7. If you plan to use the controller to move cameras (pan, tilt, zoom, focus), in the **ptzjoystick** row set **Selected Device** to the controller you are configuring.
8. If you plan to use buttons on the controller to select cameras in the DashBoard Camera Panel, in the **selector** row set **Selected Device** to the controller you are configuring.
9. In the **Basic Tree View**, expand the **DashBoard Services** node, and then double-click **Selector UI Mappings**.

   The **Selection Mapping** interface appears.

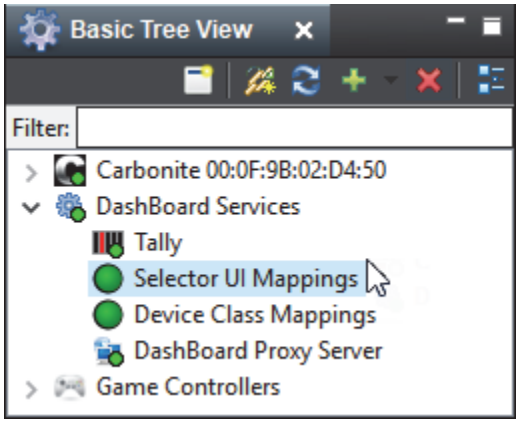


***Figure 9** - Basic Tree View*

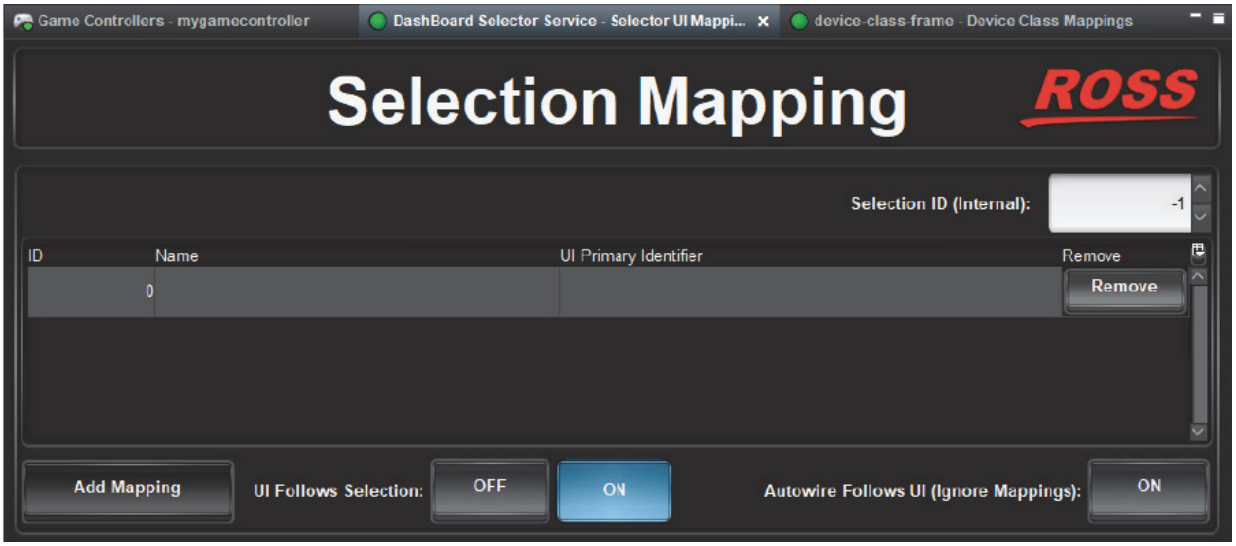


***Figure 10** - UI Follows and Autowire Settings*

10. In the **Selection Mapping** interface, select one of the following three options:

Option	Description
UI Follows Selection - OFF	If you set **UI Follows Selection** to **OFF**, when you push a configured button on the controller, the button's UI selection is activated, but the User Interface (UI) does not become the active window in the DashBoard UI. UI selection options can be configured to activate selections on any camera control panel, custom panel, or device view. Choose this option to use the controller to select and control cameras, while performing other tasks using the DashBoard Camera Panel, custom panels, or device views. This option is suitable for controlling cameras as part of the Lightning Control System (LCS), provided that you want the LCS panel to stay active while you perform other tasks with custom panels or device views. **Note:** Opening other active panels will not impact your configured controller actions.

Option	Description
UI Follows Selection - ON	If you set **UI Follows Selection** to **ON**, when you push a configured button on the controller, DashBoard opens the UI selection and it becomes the new active window. Choose this option to configure unassigned controller buttons to open DashBoard panels and device views on demand, such as a Carbonite control or a camera control panel. **Note:** If the DashBoard Camera Panel is active and you press a controller button to open a different panel or a device view, the DashBoard Camera Panel will no longer be your active window. This is important, because you cannot use the controller to select and control cameras unless the DashBoard Camera Panel is the active window.
Autowire Follows UI - ON	If you set **Autowire Follows UI** to **ON**, the **Selection Mapping** table is ignored. Choose this option to prevent accidental movement of cameras while the operator is using other DashBoard panels or device views. **Note:** You cannot use the controller to select and control cameras unless the DashBoard Camera Panel is the active window.

Configuring buttons to open device views or panels

If there are unassigned controller buttons and you want to configure them to open device views or panels, follow the procedure below.

Note: This procedure requires that you have already completed the data mappings for Axis Controls and Buttons. Refer to **Step 3** of the procedure **To create data mappings for the Axis Controls and Buttons** to verify that you have completed your button entries. You must create an entry in the Buttons table with a name, action, value (off), value (on) and mapped OID for each button you wish to open a device view or panel.

To configure unassigned buttons to open device views or panels

1. Click **Add Mapping**.

   A new mapping row appears in the Selection Mapping table.

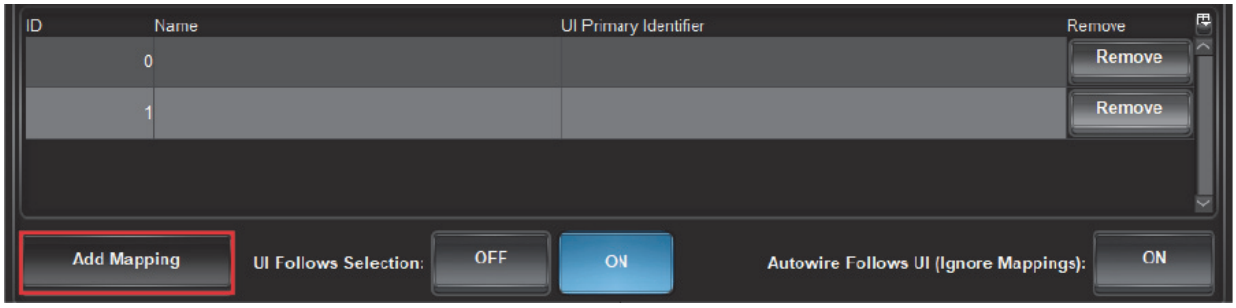
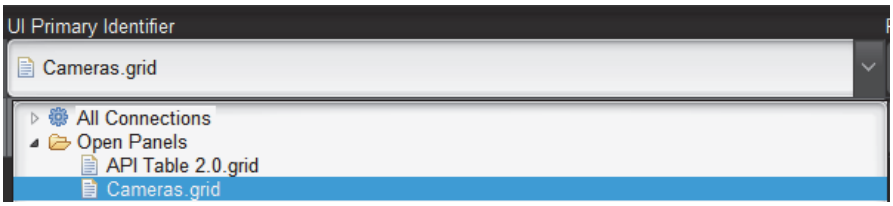


***Figure 11** - Adding New Mapping Rows*

2. In the Selection Mapping table, set the following values:

Value	Description
ID	The ID correlates to the **Value On** field in the button mapping table. **Note:** You must enter a unique number for each ID.

Value	Description
Name	Enter a name for the mapping. (optional)
UI Primary Identifier	Click the appropriate row, and use the drop-down menu to navigate to the panel or device view that you want the button to correspond to when it is pressed.  **Note:** For panels to appear in the **Open Panels** drop-down menu, you must first open the panel in DashBoard. To open a panel, double-click on its **.grid** file. **Tip**: Make a note of the **ID** and which panel or device view it is configured to open.

3. Repeat the steps above for each button you want to configure.
4. Click **Save**, and provide a file name and path.

   Note: The mappings are specific to the controller(s) you configured. You can use the mappings on other DashBoard computers only if their controllers are the same model, or report the exact same controls.