



## DOMOSAPIENS KEYPAD ENHANCER DRIVER

This driver enables you to significantly enhance the functionality of Control4 Keypads.

---

### OVERALL DESCRIPTION AND CAPABILITIES

When installed **between** the actual keypad driver and devices/functions such as Lights, Blinds, Advanced Lighting Scenes(ALS) and others, this driver allows you to track elaborate key presses not natively supported by the Control4 keypad driver, to set the LED colors on the keypad, to blink LEDs using programming commands, to disable the keypad altogether (for parties, etc.) and if Up and Down keys are used on the keypad, to select up to 6 different sets of functions to be performed by the Up/Down keys (see the **SELECTOR BUTTONS** section below). The driver is also able to simulate an 8<sup>th</sup> key, achieved when both Up and Down are pressed simultaneously.

The driver provides three sets of output connections (passthrough) which enable control of three sets of Lights/ALS, including full dimming capability. Simply use Single, Double and Triple taps on a keypad button. An additional method is provided: 'Press and Hold', including 'Single Press and Hold', 'Double Press and Hold' and 'Triple Press and Hold'.

The driver has a new set of Contact Sensor inputs which may be used instead of a real keypad button. When such a sensor is connected, properties allow the installer to select the contact type (NC or NO) as well as its effect (always Single Tap or Fully mirroring a real keypad button).. Debouncing is also available if desired. A connection to a real keypad button overrides a connection to a Contact sensor for a specific input.

If none of the outputs for a given keypad button are connected, the driver will force that keypad button's LED to Off state and the color associated with that state for the button. This may also be achieved via Programming.

A night mode is available via programming commands and Actions. When in Night Mode, the connected keypad button LEDs will be turned Off (black). Optionally, the backlighting may also be turned Off or to a darker Off color. If the user wishes to see the current status of the LEDs, simply press one of the connected buttons and the LEDs (and backlighting) will come on, and the buttons will become active for the duration specified in the Property 'Night Status Duration in Seconds'. For Night Mode to be effective (all LEDs turned Off), all keypad buttons must be connected to this driver, and then to other devices as required via the passthrough connections.

A rich set of programming events are provided as well as variables containing the status of up to 6 keypad LEDs.

Since many instances of this driver may be installed in a project (typically one per physical keypad), the driver provides Actions to broadcast selected colors to other instances as well as stopping blinking across all instances. This can simplify installation and programming.

---

### INSTRUCTIONS

- Use the free trial period or activate the driver at any time by assigning to this project the license you purchased from the driverCentral website (requires the driverCentral cloud driver). You may use multiple copies of this driver in your project.
- Connect up to seven Control Output BUTTON\_LINK connections to the real physical keypad's connections. Note that if your physical keypad is configured with double or triple buttons, there is no need to connect the unused bindings. The same is true when your keypad has a full length 6th button (not up/down arrows) – **IN THIS CASE, DO NOT CONNECT THE 'DOWN' BUTTON**. If you are using an older 3-button keypad, simply connect the first three bindings. Instead of connecting to a real physical keypad button, any input may be connected to a Contact Sensor to simulate a physical keypad button. In this case, review the properties and set accordingly.
- Connect Control Input BUTTON\_LINK connections to Lights, Blinds, Advanced Lighting Scenes and/or other devices as appropriate. Connecting to their Toggle connection is usually best.
- If required, use the Action 'Set Button Timings' to change the default MultiPress interval (defaults to 1500 msec) and/or the default Hold interval (defaults to 1000 msec). Generally, the defaults will be adequate. This action is also used to set the Hold Pulse intervals when holding Up or Down buttons for various selections. See the Selector Buttons section for more details.
- Once you have entered Colors A to J for one driver, you may use the Action 'Broadcast Colors to Other Instances' to simplify your installation tasks.

---

## PROPERTIES

- **Cloud Status** displays the status of the DriverCentral cloud driver.
- **Automatic Updates** enables automatic updates of this driver from DriverCentral.
- **Driver Version** displays the version of this driver.
- **Driver Information** displays various status messages about the driver.
- **Debug Mode** turns Debug Mode Off or On (with output to the Lua Output window).
- **Debug Duration in Minutes** sets the duration of Debug On.
- **MultiPress, Hold Intervals in msec** displays the values entered for MultiPress and Hold using the Action 'Set Button Timings'. MultiPress is the duration the driver will wait for an eventual second or third tap if these are used. Hold is the duration the driver will wait before 'declaring' a Button Held situation. The default values of 1500 and 1000 milliseconds should be adequate for most situations.
- **Hold Up/Down Pulsing in msec** displays the values entered using the Action 'Set Button Timings' for each Selection state, including Selector Off (NoSel). This indicates the interval between successive triggerings of the 'Single Tap Selection X' events when the Up or Down keys are held. This would be used, for example, to ramp the volume up or down. While you could use a starting

point of 300 milliseconds, you may need to experiment with the value to avoid overflowing your audio device driver with commands.

- **Virtual 8<sup>th</sup> Button Used:** if you are NOT planning to use the virtual 8<sup>th</sup> button (simultaneous pressing of Up and Down), set this property to 'No'. This will enable all events related to the Up and Down buttons. If the virtual 8<sup>th</sup> button is used (this property set to 'Yes'), some events are not functional for Up and Down, such as all Holds and Release.
- **\*\*NEW\*\* Button Types:** optionally designate one or more (up to 6, actually) buttons as Selector buttons ('S'); see the section below on **SELECTOR BUTTONS**. Normal buttons are simply designated as 'N' (Control4 paradigm) or 'H' (see below for Press/Press&Hold). For two or three functions on the same button, use 'D' (Double) or 'T' (Triple). This is selectable by button, so the following is acceptable: *N,D,D,T,H,N*. When used, Button 7 (the 'Down' button) uses the same type as button 6 (the 'Up' button).

Some users, however, prefer to have dual functions (on the same button) using Press/Press&Hold instead of Single/Double Tap, especially users coming from a different lighting control system such as Vantage, for example. In this case, use 'H' as button type. Then, when the user simply taps the button, the 'Button x Passthrough' connection will be activated and when the user presses and holds the button (for 1.5 second or more), the 'Button x Double Tap/Press&Hold Passthrough' connection will be activated instead. This allows for a different paradigm than Single/Double Tap. You may want to avoid using the two paradigms in the same installation as this might confuse users.

**IMPORTANT:** Remember that multiple-sized keypad buttons 'hide' button numbers. So, if your real keypad has two double-size buttons, a single-size button and Up/Down buttons, for example, the first double-size would be button 1, the second double-size would be button 3, the single-size would be button 5 and the Up/Down buttons would be buttons 6 and 7. To use the single-size button as a Selector Button, you would specify *N,N,N,N,S*. You do not need to specify Button Types for Up/Down buttons (buttons 6 and 7).

- **Selector Button(s) Off:** when using one or several Selector button(s), you may allow the 'Selected Off' state to be active. With multiple buttons, this occurs when you press the currently selected button. If allowed, the button would turn Off and the driver would be in the 'Selector Off' state. Similarly, with a Single Selector button where you cycle through the various states with each button press, the 'Off' state would become part of the cycle. This property is available only if one or more Selector buttons are specified in Button Types.
- **\*\*NEW\*\* Selector Button(s) Passthrough:** when using several Selector buttons, you may want to avoid activating the Single Tap Passthrough for a newly pressed Selector button when this changes the selection. Example: you have several Selector buttons, each selecting a different room and each having its Single Tap Passthrough connected to a light or Scene toggle for that room. When you switch to another room by hitting its Selector button, you do not necessarily want the toggle to be activated immediately, but rather on the next button press. This property allows you to do this.
- **Selector Timeout in Seconds** allows the selector to revert to a default value after the specified timeout. Set to 0 if you wish the selector to remain fixed.
- **Selector Value upon Timeout** specifies the default selector value to be set after the timeout expires. This is only available when a timeout is specified.

- Button 1-6 Off/On and Backlight Colors:** specify the Off and On colors for each LED of the keypad, as well as Backlighting (7<sup>th</sup> entry). This is a pair of letters for each LED, except for Single Selector buttons. The letters refer to the color selectors below (A to J) and may also include 'K' for Black and 'N' for the connected Light/ALS's native Off and/or On Colors. The 7<sup>th</sup> entry is used for Backlighting, especially related to Night Mode.
  - The defaults are: *AB,AB,AB,AB,AB,AB,KA* - this means the Off color for each button key is Color A and the On color is Color B. For Backlighting, the default Off color is Black and the On color is Color A.
  - Example: *AB,GH,KC,KD,KN,NN,KG* - key 1 has colors AB, key 2 uses GH, key 3 uses color C for On and Black for Off, key 4 uses Color D for On and Black for Off, key 5 uses the Light's native On color with Black for Off and key 6 uses the Light's native colors for both Off and On. Backlighting is Black when Off and uses Color G when On.
  - If you wish the driver to turn Off the LED of any connected real keypad button which is unused (none of its outputs are connected), specify K (Black) as the Off color. If necessary, Force the state to Off using Actions. Another 'Off' color may also be used, if desired.
  - If you have defined a Single Selector Button, you may specify up to 5 On colors for that button (example: *KABCDE*), corresponding to the number of different selector states you wish to use. See the section below on **SELECTOR BUTTONS**.
- Keypad Disabled Color:** Specify the color to be used for each connected button when the keypad is disabled. **IMPORTANT:** if you plan to use both Night Mode and Keypad Disabling, set this color to something other than black, such as blue, for example. Otherwise, this would create confusion for the user.
- Enabling Button Sequence:** optionally specify a 1-, 2- or 3-button sequence which may be used locally to enable a disabled keypad. You may leave empty or use 'None' if this feature is not desired. If one button number (1 to 7 with 7 being the 'Down' button) is specified, then the keypad is enabled when this button is pressed. If two or three buttons are specified (separated by a comma), then these buttons must be pressed in the same sequence on the keypad and within a 2-second interval for the keypad to be enabled.
- Night Status Duration in Seconds:** allows you to specify for how long a keypad in Night Mode (set via a programming command) will 'wake up' to allow the user to see the current status of LEDs and to optionally use some keys. A keypad is 'woken up' by pressing any key while in Night Mode. At the end of this duration, they keypad returns to Night Mode with all LEDs Off.
- This Keypad Part of Group:** you may assign this keypad to one of 10 groups, arbitrarily. This is used when you broadcast colors or enable/disable keypad in other ("peer") instances of this driver. This allows you to limit which keypads will respond to the broadcasted commands.
- Contact Sensors Are** specifies the default state of the connected contact sensors. This property is available only when one or more contact sensors are connected.
- Contact Sensors Button Simulation** specifies the action performed when connected contact sensors are switched to their active (non-default) state. Depending on the nature of the contact

sensor, you may limit this to single taps only, or allow double/triple taps (fully mirror, including holding the contact sensor). This property is available only when one or more contact sensors are connected.

- **Debounce Timer in msec or 0 to disable** is used if your contacts sensors are subject to bouncing. Specify a value in milliseconds (such as 50 or 100) or 0 to disable this function. This property is available only when one or more contact sensors are connected.
- **Select Color A to J** allows you to assign a specific color to each letter. Most popular (and reliable) colors are provided by name, but you may further customize the color. Color K does not need to be specified and corresponds to 'Black' (LED turned Off). Be aware that the colors shown in Composer may not exactly match what will be displayed by the actual keypad. Experimentation may be necessary.

The following colors have been found to display adequately on the Control4 Configurable Keypad:

- (Default for Color A) **Blue**: 0, 0, 255
- (Default for Color B) **Green**: 0, 120, 0 (too bright otherwise)
- (Default for Color C) **Red**: 255, 0, 0
- (Default for Color D) **Yellow**: 255, 120, 0
- (Default for Color E) **Purple/Magenta**: 255, 0, 255
- (Default for Color F) **Turquoise/Cyan**: 0, 102, 188
- (Default for Color G) **Orange**: 255, 86, 0
- (Default for Color H) **White**: 255, 120, 255 (too greenish otherwise)
- (Default for Color I) **FullGreen**: 0, 255,0 (brighter than other colors)
- (Default for Color J) **FullWhite**: 255, 255, 255 (greenish White)
- (Pre-defined Color K) **Black**: 0, 0, 0

---

## SELECTOR BUTTONS

This driver is able to assign multiple 'personalities' to the Up/Down buttons using one or more Selector buttons. There are two ways to accomplish this, each with its own benefits.

1. You may decide to assign some keypad buttons (say buttons 3, 4 and 5 in this example) to allow the user to select a specific assignment for the Up/Down buttons. Button 3 could assign the Up/Down buttons to a blind driver, for example, Button 4 could assign them to the TV volume and Button 5 could assign them to a Music source. Each physical button could be labeled accordingly. In this scenario, the user would press the appropriate selector button before using the Up/Down buttons. Events are available to activate the source if required, when a specific selection is made. Buttons specified for multiple selector buttons may also be used for normal duties (Single Tap, etc.).

To specify several selector buttons, as in the example above, set the 'Button Types' property with several 'S' indicators. For example, *N,N,S,S,S* will set buttons 3, 4 and 5 as selectors. You may then set the specific color for each selector button using the 'Button 1-6 Off/On Colors' property. It is recommended that you use Black as the Off color in this case. Using our example, you could specify *AC,AC,KB,KB,KB*. You could use different On colors for your Selector buttons if it better suits your installation (such as *AC,AC,KB,KD,KE*). **IMPORTANT:** with multiple selector buttons, the selector state will match the selected button number. In this example: 3, 4 or 5.

2. You may decide to have only one selector button with several states. This is called the 'Single Selector Button'. In this case, each state would have a separate LED color which would tell the user which function will be assigned to the Up/Down buttons. This requires fewer physical buttons, but in turn requires the user to remember which color corresponds to which function. Events are available to activate the source if required, when a specific selection is made. A button specified as a Single Selector buttons will only trigger the 'Pressed' and 'Released' events and may not be used for other duties.

To specify a Single Selector button, simply use only one 'S' in the 'Button Types' property (example: *N,N,N,N,S*). You may then specify up to 6 colors (Off color and up to 5 On colors) for that button under the 'Button 1-6 Off/On Colors' property. For example, if you wish to have 3 selector states for button 5 being the Single Selector button, you could specify: *AC,AC,AC,AC,KBDE*. **IMPORTANT:** with a single selector button, the selector state will match the color selected in the sequence. In this example: 1, 2 or 3 for *B, D* or *E*.

When Selector buttons are used, you may also request that the driver activate its events 'When Keypad Button 6/Up Single Tap Selector x Active' or 'When Keypad Button Down Single Tap Selector x Active' repeatedly if the user holds down the Up or Down button. While the button is held, the event will be triggered xxx milliseconds, as you specify using the 'Set Button Timings' Action. You may specify a different value for each Selector state. **IMPORTANT:** for this function to work, you **MUST** specify a timing for the appropriate Selection(s) using the Action 'Set Button Timings'.

#### **An Example:**

- You want 3 personalities (states) for your Up/Down buttons: Blind A (Up/Down/Stop), Blind B (Up/Down/Stop) and Music (On, Off, Vol Up, Vol Down, including ramping if held).
- For the blinds, the Up and Down buttons would move the blind as expected. Pressing both Up and Down simultaneously would stop the blind.
- For Music, if the room is Off or our preferred source is not selected, then our preferred source is selected when first pressing the Up button. Otherwise, pressing (Single Tap) the Up or Down buttons adjusts the volume by 3 increments (you may modify). Holding the Up or Down buttons ramps the volume up or down. Pressing both Up and Down simultaneously turns the Room Off. This is a simple but fully functional 2-button audio control solution requiring minimum programming.

#### **Solution A (Multiple Selector Buttons):**

- You define Buttons 3, 4 and 5 as Selector buttons (Button Types: *N,N,S,S,S*). Pressing Button 3 sets the UP/Down buttons to control Blind A, Button 4 for Blind B and Button 5 for Music.
- Using the Action 'Set Button Timings', you set the 'Hold Pulse for Selection 5' to 300 milliseconds (or another value which works in your installation).
- Make sure the property 'Virtual 8<sup>th</sup> Button Used' is set to 'Yes' and 'Selector Button(s) Off' is set to 'Not Allowed'.
- In Connections, make the following connections:

- 'Button 6/Up Selector 3' to 'Blind A' driver's 'Open Button Link'
- 'Button Down Selector 3' to 'Blind A' driver's 'Close Button Link'
- 'Button Up/Down Selector 3' to 'Blind A' driver's 'Stop Button Link'
- 'Button 6/Up Selector 4' to 'Blind B' driver's 'Open Button Link'
- 'Button Down Selector 4' to 'Blind B' driver's 'Close Button Link'
- 'Button Up/Down Selector 4' to 'Blind B' driver's 'Stop Button Link'

**NOTE:** instead of using Connections, you may implement the above using Programming.

- In Programming, you specify:
  - When Keypad Button 6/Up Single Tap Selector 5 Active:
    - If Room Power is Off OR Audio Device Selected IS NOT 'my Device'  
Select the 'my Device' as the audio source in Room
    - Else  
Send the command 'Pulse Volume Up' to Room  
Send the command 'Pulse Volume Up' to Room  
Send the command 'Pulse Volume Up' to Room
  - When Keypad Button Down Single Tap Selector 5 Active:
    - Send the command 'Pulse Volume Down' to Room
    - Send the command 'Pulse Volume Down' to Room
    - Send the command 'Pulse Volume Down' to Room
  - When Keypad Button Up/Down Single Tap Selector 5 Active:
    - Turn the Room Off

**Solution B (Single Selector Button):**

- You define Button 5 as a Single Selector button (Button Types: *N,N,N,N,S*) with the colors for button 5 specified as *KABC* (or whichever you prefer, as long as there are 4 colors: Off and 3 On states). Pressing Button 5 repeatedly will set the Up/Down buttons to control Blind A (when Button 5 color is 'A' or the first 'On' color specified), Blind B (when Button 5 color is 'B' or the second 'On' color specified) or Music (when Button 5 color is 'C' or the third 'On' color specified).
- Using the Action 'Set Button Timings', you set the 'Hold Pulse for Selection 3' to 300 milliseconds (or another value which works in your installation).
- Make sure the property 'Virtual 8<sup>th</sup> Button Used' is set to 'Yes' and 'Selector Button(s) Off' is set to 'Not Allowed'.
- In Connections, make the following connections:
  - 'Button 6/Up Selector 1' to 'Blind A' driver's 'Open Button Link'
  - 'Button Down Selector 1' to 'Blind A' driver's 'Close Button Link'
  - 'Button Up/Down Selector 1' to 'Blind A' driver's 'Stop Button Link'
  - 'Button 6/Up Selector 2' to 'Blind B' driver's 'Open Button Link'
  - 'Button Down Selector 2' to 'Blind B' driver's 'Close Button Link'
  - 'Button Up/Down Selector 2' to 'Blind B' driver's 'Stop Button Link'

**NOTE:** instead of using Connections, you may implement the above using Programming.

- In Programming, you specify:
  - When Keypad Button 6/Up Single Tap Selector 3 Active:
    - If Room Power is Off OR Audio Device Selected IS NOT 'my Device'  
Select the 'my Device' as the audio source in Room
    - Else
      - Send the command 'Pulse Volume Up' to Room
      - Send the command 'Pulse Volume Up' to Room
      - Send the command 'Pulse Volume Up' to Room
  - When Keypad Button Down Single Tap Selector 3 Active:
    - Send the command 'Pulse Volume Down' to Room
    - Send the command 'Pulse Volume Down' to Room
    - Send the command 'Pulse Volume Down' to Room
  - When Keypad Button Up/Down Single Tap Selector 3 Active:
    - Turn the Room Off

**IMPORTANT:** please note that the selection numbers are different if you choose a Single Selector button (Solution B) or if you choose Multiple Selector buttons (Solution A). Also, Solution A leaves you with only two free buttons to use for other functions, while Solution B leaves you with four buttons for other uses.

---

## PROGRAMMING

- A programming command allows you to change the Off and On colors associated with each keypad LED. This overrides what you have specified in the 'Button 1-6 Off/On Colors' Property.
  - Some output drivers/devices do not return their state so the driver can properly set the LED state of connected real keypad buttons. A programming command allows you to force the state of the keypad LED to one of the two specified colors (Off or On). An Action is also available to do this.
  - A programming command allows you to blink a keypad LED. Normal blinking (once every second) and slow blinking (once every two seconds) are available. Blinking for a button stays on until stopped via programming from this or another instance of the driver.
  - Programming commands allow you to disable/enable this keypad and/or, optionally, keypads connected to other instances of this driver in your project. It is important to note that the driver will disable only the real keypad buttons it is connected to.
  - The UP\_DOWN\_SELECTOR variable contains the current selector for Up/Down buttons (1 to 5) or 0 when not selected.
  - A very comprehensive set of events are available. Setting Debug On will show you which events are triggered. Experiment to see which ones match your requirements. The event '1P-Hold in Progress' is triggered as soon as the driver detects a hold (the button has been pressed beyond the hold delay) and the event 'Press and Hold' is triggered when the button is actually released. This is also valid for '2P-Hold In Progress' vs. 'Double Press and Hold' as well as for '3P-Hold in Progress' vs. 'Triple Press and Hold'.
-



## SUPPORT

For support on this driver please go to <https://help.drivercentral.io/>. Give a detailed description of the problem and also include the version number of the driver and the version of Control4 OS that you are using.

---

## AUTO UPDATE

This driver is updated with fixes and new features from time to time. To ensure your project uses the latest version, set the Automatic Updates property of the driver to On.

---

## CHANGELOG

10.0.0	January 5, 2018	Initial Release
10.1.0	January 18, 2018	Added Keypad Disable, 2 more colors, Keypad Groups, enhanced Actions and Commands
11.0.0	February 8, 2018	Added 8 <sup>th</sup> button (Up/Down simultaneously) and Selector capability for Up and Down buttons
11.1.0	October 14, 2018	Ability to exclude Up/Down from MultiPress, added Hold in Progress events, issue fixed with some events not always triggered
11.2.0	November 5, 2018	New command to force LED state
12.0.0	January 3, 2019	Added Contact Sensor inputs, ability to specify blink color, improved resilience to Cloud Driver updates
12.1.0	March 18, 2019	All taps available for selector buttons and new property to disable Virtual (8 <sup>th</sup> ) button
12.2.0	April 2, 2019	Added property to specify keypad enabling button sequence
13.0.0	November 14, 2019	Added Single Selector button, hold event pulsing for Up/Down, several issues fixed, documentation reworked
13.2.0	March 8, 2021	Licensing update, commands to blink LED, improved passthroughs, issues fixed
13.3.0	September 24, 2021	Added Press/Press&Hold model for dual commands, issues fixed
14.0.0	February 14, 2022	Added Night Mode with backlighting support
14.0.1	February 23, 2022	Minor issue fixed with backlighting in Night mode
15.0.0	January 18, 2023	New simpler specification for colors, additional events for button presses and new Selector timeout option
15.0.1	July 5, 2023	Fixed issue with Set Backlight programming command
15.1.0	February 14, 2023	New Button Types 'D' and 'T' to replace 'MultiPress Used' property, improvements when using Native colors including for Selector buttons, new Selector Button(s) Passthrough property, issues fixed