



KRAMER ELECTRONICS, Ltd.

KRAMER STEP-IN Requirements 2.0

Table of Contents

Functional description.....	3
STEP-IN Master: Polling.....	4
Asynchronous commands and notifications.....	6
WEB to Device interface.....	7
STEP-IN capability.....	7
Supported commands list.....	8
Simple #.....	8
#MODEL.....	8
#BTN.....	9
#VID.....	10
#ROUTE.....	11
Layer Enumeration.....	11
#STEP-IN CP.....	12
Video Port Type.....	12
#REMOTE-INFO.....	13
Stages.....	13
Video Port Type.....	13
#PROG-ACTION.....	14
#TUNNEL-CTRL.....	15
Stages.....	15



KRAMER ELECTRONICS, Ltd.

3 Am VeOlamo St. Jerusalem, 95463, Israel
Tel: + 972 2 6544000 Fax: + 972 2 6535369
E-mail: info@kramerel.com
Web: www.kramerelectronics.com

Functional description

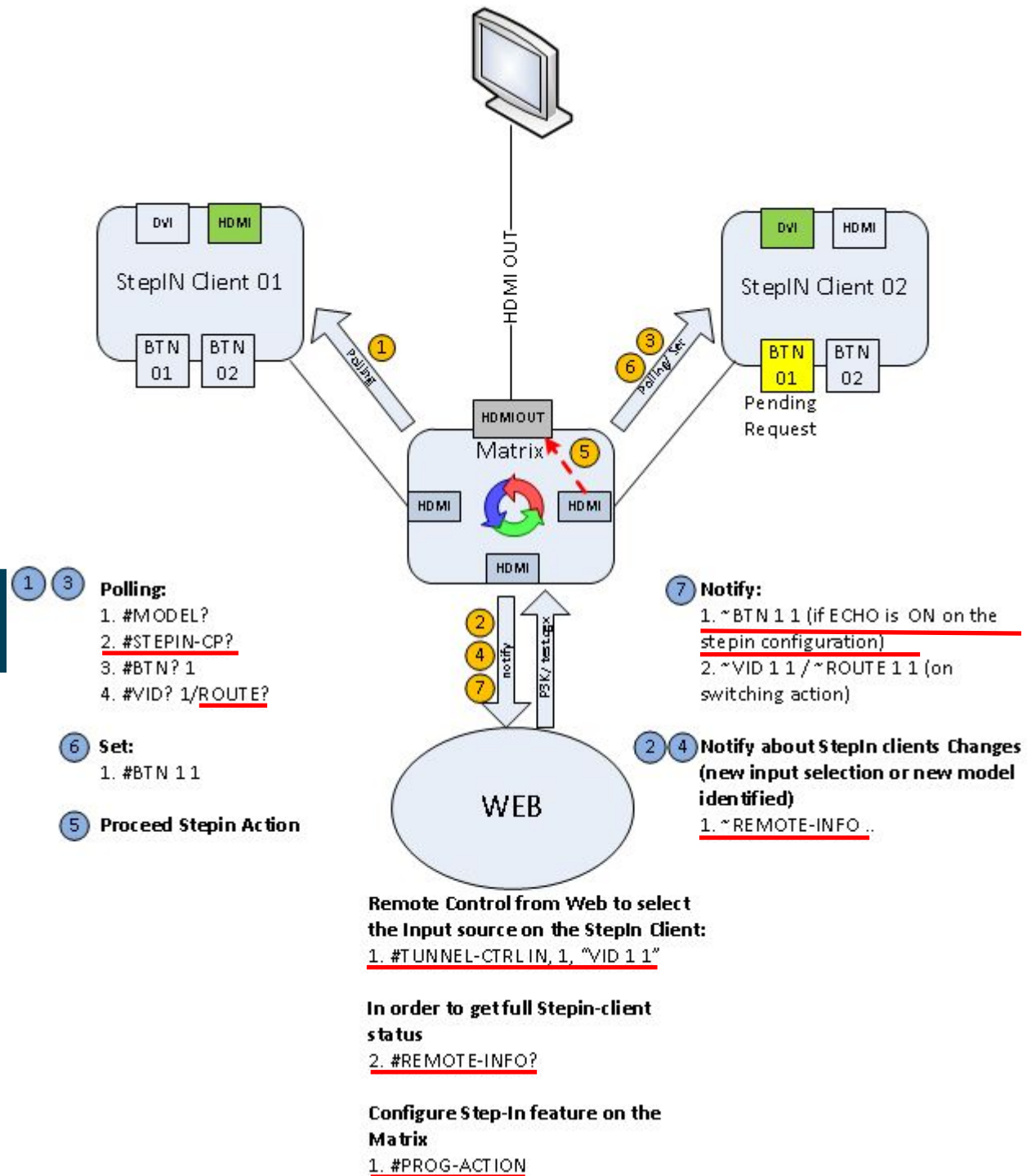


Figure 1: Step-In common use case

The STEP-IN feature is a feature designed by Kramer and exclusively part of Kramer IP. The architecture of the STEP-IN feature defined a STEP-IN clients and ONE STEP-IN master.

The STEP-IN Client is a product which is responsible to raise a request of STEP-IN and give indication to STEP-IN Master which button have been pressed on the remote unit.

The STEP-IN master is responsible to scan all the equipment connected to it over Video RS232 control line (over HDMI or over HDBT), to identify their capabilities and detect STEP-IN requests. The baud rate of communication on those RS232 lines is fixed to **9600** bps.

The Figure 1: Step-In common use case , describes clearly all the process.

STEP-IN Master: Polling

Command **#STEPIN-CP?** (Detailed in chap "Supported commands list") is used by the STEP-IN master, commonly a matrix/switcher (i.e. VS-62H) in order to know if the connected module supports STEP-IN capability or not. Timeout or error "Command not supported" means that module is not STEP-IN compatible. In the contrary, the number of inputs and buttons supported will be returned.

Backward compatibility requirement: if the MODEL is **SID-XnN**, the STEPIN-CP won't return Video Type information. STEP-IN master should be able to handle this kind of response and bypass the missing information. In that case, the WEB UI SHALL handle static maps for each model and be able to show the Video Input type for each STEP-IN client.

Command **#MODEL?** (Detailed in chap "Supported commands list") returns the Model name of the connected remote module. This info will be useful in order to report the name of the equipment to the Web interface and identify if equipment has been changed. In some older boxes, the model name was used to identify if the Box is supporting STEP-IN capabilities.

Backward compatibility requirement: if the MODEL is **SID-XnN**, the STEP-IN master should consider it as a STEP-IN client even if the **STEPIN-CP** command returns an error.

The STEP-IN Master will poll all the inputs at a frequency of max 150 milliseconds per input.

Polling will include commands:

- **#STEPIN-CP?** - detect Step-In support
- **#MODEL?** - to detect the name of the module connected
- **#BTN?** - to know if module is pending for STEP-IN and to know its actual state



KRAMER ELECTRONICS, Ltd.

3 Am VeOlamo St. Jerusalem, 95463, Israel
Tel: + 972 2 6544000 Fax: + 972 2 6535369
E-mail: info@kramerelectronics.com
Web: www.kramerelectronics.com



KRAMER ELECTRONICS, Ltd.

- **#VID?** (legacy command for old modules – **SID-XnN**) – to know currently active input on module
- **#ROUTE?** (coming to replace **#VID?** In all new devices)– to know currently active input on module

If during a polling cycle **#STEPIN-CP?**, **#MODEL?**, **#ROUTE?** (or **#VID?**) return values different than the previous polling cycle, STEP-IN Master (i.e VS-62H) SHALL reports it to all control ports by sending a **~REMOTE-INFO** notification. (Control ports mean all local RS232 control, UART and USB ctrl and UDP and TCP connections).

If **#BTN?** returns status “pending” (waiting for step-in) the STEP-IN Master will proceed for the following:

1. Makes corresponding input ACTIVE and all others INACTIVE (perform an internal video switch)
2. Reports the switching action to all control ports by sending a **~ROUTE** or **~VID** notification.

Backward compatibility requirement: if the MODEL is **SID-XnN**, VID should be sent otherwise **ROUTE** notification SHALL be sent.

3. Send **#BTN** “active” to the STEP-IN Client to notify about its successful selection and **#BTN** echo notification will be sent if “echo” feature is configured and if **#BTN** command was successful (look below at the format of the Notification in this case).
4. Send **#BTN** “mute” to the other STEP-IN Clients to notify about their un-selection and NO **#BTN** echo notification will be sent.

If **#BTN?** returns status “MUTE” on a STEP-IN client which was previously active, this means that the STEP-IN client ask for MUTE pending and the STEP-IN Master has to decide if to mute this input based on Matrix configuration (stop video switching or choose another input).

Asynchronous commands and notifications

As described in the previous chapter, the STEP-IN master is responsible to notify STEP-Clients about the status of the Matrix.

The following events will trigger some actions:

- **INPUT SWITCH** (front panel, P3K commands , etc)

#BTN ,<btn num>,1 (make it active) will be sent to the STEP-IN Client connected to the new chosen input. It also sends **#BTN <btn num>,0** (make it muted) to the input which was active until the switch

The BTN command is detailed in chap "Supported commands list".

- **REMOTE SWITCH REQUEST** (via **#TUNNEL-CTRL** command)

This command is tunneling commands to STEP-IN client modules. This is used by the WEB interface in order to control the STEP-IN client equipment remotely, **i.e** sends

#ROUTE 1,inp,out (for legacy : **#VID inp>out**).

#TUNNEL-CTRL stage, stage_id,"command string to send"<cr>.

For backward compatibility on "**REMOTE SWITCH REQUEST**", **#VID N>M** will be sent only for **SID-XnN modules** For newer modules **#ROUTE** command will be sent. (This is the responsibility of the Web to use the proper command based on the Model name). Those commands are not returning any values and are sent asynchronously via the mailbox (queue) of the STEP-IN master application.



KRAMER ELECTRONICS, Ltd.

3 Am VeOlamo St. Jerusalem, 95463, Israel
Tel: + 972 2 6544000 Fax: + 972 2 6535369
E-mail: info@kramerelectronics.com
Web: www.kramerelectronics.com



KRAMER ELECTRONICS, Ltd.

WEB to Device interface

STEP-IN capability

#PROG-ACTION – command used to configure step-in buttons action list on the STEP-IN Master (Matrix/Switcher)

#TUNNEL-CTRL – send tunneling command to remote module connected (essentially will use it to send VID/ROUTE commands)

#REMOTE-INFO – retrieving remote module info at the first time the Web will connect

Supported commands list

Simple #

Command - #		Command Type - System-mandatory	
Command Name		Permission	Transparency
Set:	#	End User	Public
Get:	-	-	-
Description		Syntax	
Set:	Protocol handshaking	# <div>CR</div>	
Get:	-	-	
Response			
~nn@ <div>SP</div> OK <div>CR LF</div>			
Parameters			
Response Triggers			
Notes			
Use to validate the Protocol 3000 connection and get the machine number STEP-IN MASTER products are using this command to identify the availability of a box.			

#MODEL

Command - MODEL?		Command Type - System-mandatory	
Command Name		Permission	Transparency
Set:	-	-	-
Get:	MODEL?	End User	Public
Description		Syntax	
Set:	-	-	
Get:	Get device model	# MODEL? CR	
Response			
~ nn @ MODEL? SP model_name CR LF			
Parameters			
model_name - String of up to 19 printable ASCII chars			
Response Triggers			
Notes			



KRAMER ELECTRONICS, Ltd.

3 Am VeOlamo St. Jerusalem, 95463, Israel
 Tel: + 972 2 6544000 Fax: + 972 2 6535369
 E-mail: info@kramerelectronics.com
 Web: www.kramerelectronics.com



KRAMER ELECTRONICS, Ltd.

STEP-IN MASTER products are using this command in order to identify equipment connected to it and notify about changes about the identity of the equipment connected to it. The Matrix will keep this data in memory in order to be able to answer REMOTE-INFO requests.

#BTN

Command – BTN		Command Type – System	
Command Name		Permission	Transparency
Set:	BTN		
Get:	BTN?		
Description		Syntax	
Set:	set button state	# BTN _{SP} <i>button_num</i> , mode _{CR}	
Get :	Get button state	#BTN? <i>button_num</i> _{CR}	
Response			
~ _{nn} @ BTN _{SP} <i>button_num</i> , mode _{CR LF}			
Parameters			
button_num – button number (0...N) mode – 0 – muted 1 – active 255 (0xFF) – pending (asking to step in)			
In Case of ECHO notification, the mode will be replaced by the INPUT # of the STEPIN client and won't mean the STATUS of the button. This notification will be ECHO-ED only when a Button become ACTIVE.			
Response triggers			
Notes			
As a result of a SET command this button MUST reflect its status using LEDs.			
Mute – means button LED is turned off Active – Button LED is ON In Pending mode – button LED is blinking			
STEP-IN master is using this command in order to Get the actual status and identify if the box is in PENDING STEP-IN REQUEST. In reply of the STEP-IN request, the STEP-IN MASTER will update the button status by sending SET to active and will proceed to STEP-IN action configuration. Others STEP-IN client will be SET to MUTE.			

Value	Description	comment
0	Module state – MUTED	Set/Get
1	Model state - ACTIVE	SET/Get
255	Pending : ask to StepIn	Get only

#VID

Command - VID		Command Type - Switch	
Command Name		Permission	Transparency
Set:	VID	End User	Public
Get:	VID?	End User	Public
Description		Syntax	
Set:	Set video switch state	#VID _{SP} <i>in>out</i> _{CR}	
Get:	Get video switch state	#VID? _{SP} <i>out</i> _{CR}	
Response			
Set:	~nn@VID _{SP} <i>in>out</i> _{CR LF}		
Get:	~nn@VID _{SP} <i>in>out</i> _{CR LF}		
Parameters			
<i>in</i> - input number or '0' to disconnect output			
> - connection character between in and out parameters			
<i>out</i> - output number or '*' for all outputs			
Response Triggers			
Notes			
The GET command will be used to identify input switching on “StepIn” clients			
The SET command will be used for remote input switching on “StepIn” clients (essentially used by the WEB)			
This command is a legacy command. New “StepIn” modules will support ROUTE command			



KRAMER ELECTRONICS, Ltd.

3 Am VeOlamo St. Jerusalem, 95463, Israel
 Tel: + 972 2 6544000 Fax: + 972 2 6535369
 E-mail: info@kramerelectronics.com
 Web: www.kramerelectronics.com



#ROUTE

Command - ROUTE		Command Type - Routing	
Command Name		Permission	Transparency
Set:	ROUTE	End User	Public
Get:	ROUTE?	End User	Public
Description		Syntax	
Set:	Set layer routing	#ROUTE _{SP} /layer, dest, src _{CR}	
Get:	Get layer routing	#ROUTE? _{SP} /layer, dest _{CR}	
Response			
~ _{CR} @ ROUTE _{SP} /layer, dest, src _{CR LF}			
Parameters			
layer - see Layer enumeration table			
dest - * - ALL			
x - disconnect, otherwise destination id			
src - source id			
Response Triggers			
Notes			
This command replaces all other routing commands.			
The GET command will be used to identify input switching on "StepIn" clients			
The SET command will be used for remote input switching on "StepIn" clients (essentially used by the WEB)			

Layer Enumeration

Number	Value
1	Video
2	Audio
3	Data

#STEP-IN CP

Command – STEPIN-CP		Command Type – (System)	
Command Name		Permission	Transparency
Set:	None		
Get:	STEPIN-CP?	End User	Public
Description		Syntax	
Set:	None		
Get:	Get module STEP-IN capabilities	# STEPIN-CP? CR	
Response			
~nn@ STEPIN-CP SP capabilities, num_of_inputs, num_of_cntl_btn, type1, type2... typeN CR LF			
Parameters			
capabilities – 1- module support STEP-IN 0 – module doesn't support STEP-IN num_of_inputs – number of video inputs for remote switching num_of_cntl_btn – number of control buttons, to be programmed in Master device type1, type2... typeN – input type according to num_of_inputs (See Video Port Type table)			
Response Triggers			
Notes			
If module doesn't support STEP-IN it might answer with error "command not supported" This command is used by STEP-IN Master products in order to know if the equipment support STEP-IN capabilities.			

Video Port Type

Number	Value
0	undefined
1	DVI
2	HDMI
3	DisplayPort
4	HDBaseT
5	SDI
6	VGA
7	DGKat



KRAMER ELECTRONICS, Ltd.

3 Am VeOlamo St. Jerusalem, 95463, Israel
Tel: + 972 2 6544000 Fax: + 972 2 6535369
E-mail: info@kramerelectronics.com
Web: www.kramerelectronics.com



#REMOTE-INFO

Command – REMOTE-INFO?		Command Type - Switching	
Command Name		Permission	Transparency
Set:			
Get:	REMOTE-INFO?	End User	Public
Description		Syntax	
Set:			
Get:	Get connected Step-in module information	#REMOTE-INFO? _{SP} stage, stage_id _{CR}	
Response			
~nn@REMOTE-INFO _{SP} stage, stage_id, connected, model_name, curr_input, capabilities, num_of_inputs, num_of_ctl_btn, type1, type2... typeN _{CR LF}			
Parameters			
stage - 0 for input, 1 for output (see Stages table) stage_id – # of chosen stage (1.. max number of Inputs/Outputs) connected – 0/1 (if module connected) model_name – model name string curr_input – input, currently chosen on module capabilities – 1-module supports STEPIN, 0-module doesn't support STEPIN num_of_inputs – number of inputs on module num_of_ctl_btn – number of control buttons on module type1, type2... typeN – input type according to num_of_inputs (See Video Port Type table)			
Response Triggers			
Any change in module's parameters			
Notes			
This command is used by the Matrix in order to notify about StepIN client changes.			

Stages

Number	Value
0	Input
1	Output
2	Reserved
3	Reserved

Video Port Type

Number	Value
0	undefined
1	DVI
2	HDMI
3	DisplayPort
4	HDBaseT
5	SDI
6	VGA
7	DGKat

#PROG-ACTION

Command - PROG-ACTION		Command Type - Step-in	
Command Name		Permission	Transparency
Set:	PROG-ACTION	End user	Public
Get:	PROG-ACTION?	End user	Public
Description		Syntax	
Set:	Set step-in button action bitmap	# PROG-ACTION _{SP} <i>port_type, port_id, button_id, actions_bitmap</i> _{CR}	
Get:	Get step-in button action bitmap	# PROG-ACTION? _{SP} <i>port_type, port_id, button_id</i> _{CR}	
Response			
Get / Set~ nn @ PROG-ACTION _{SP} <i>port_type, port_id, button_id, actions_bitmap</i> _{CR LF}			
Parameters			
<i>port_type</i> - input/output			
<i>port_id</i> - port id			
<i>button_id</i> - external programmable button ID			
<i>actions_bitmap</i> – <i>bitmap representing actions to perform after receiving button_id. Format: XXXX...X, where X hex digit. The binary form of every hex digit represents actions from the table (see table Software Programmed). Setting '1' says that the corresponding action has to be executed.</i>			
Response Triggers			
Notes			
Programs matrix action as a response for external event (programmable button pressed)			

1.2 Software programmed:

Bitmap Number	Action
0	Echo to controller
1	Step-in out 1
2	Step-in out 2
...	...
N	Step-in out N



KRAMER ELECTRONICS, Ltd.

3 Am VeOlamo St. Jerusalem, 95463, Israel
 Tel: + 972 2 6544000 Fax: + 972 2 6535369
 E-mail: info@kramerelectronics.com
 Web: www.kramerelectronics.com



#TUNNEL-CTRL

Command – TUNNEL-CTRL		Command Type – Switching	
Command Name		Permission	Transparency
Set:	#TUNNEL-CTRL		
Get:		End User	Public
Description		Syntax	
Set:	Send an asynchronous command to a remote STEP-IN equipment	#TUNNEL-CTRL _{SP} stage, stage_id,"command" _{CR}	
Get:	N/A		
Response			
None			
Parameters			
stage - 0 for input, 1 for output (see Stages table)			
stage_id – # of chosen stage (1.. max number of Inputs/Outputs)			
command – the command to send to the STEP-IN client			
Response Triggers			
Notes			

Stages

Number	Value
0	Input
1	Output
2	Reserved
3	Reserved