

Overview of Creating a Device Driver for the Particular Types of Motor and Lighting Control Products Manufactured by Converging Systems and Marketed Under Various Trade Names

Date: October 6, 2013Ov
Revision 3.1

This overview document describes the types of protocols utilized across a wide range of Converging Systems devices. Comprehensive device driver toolkits (DDKs) which often cover multiple hardware devices are currently available from the Converging Systems website. There are two types of DDKs representing the major supported protocols that Converging Systems utilizes for its hardware products. These types are listed below. It is useful to download both of these documents before beginning on a device driver development effort.

1. Resource Links

Protocol Type	URL for download	Shorthand Reference
BRIC I Protocol	http://convergingsystems.com/bin/doc/programmers_toolkit_bric.pdf	"BRIC I Protocol"
CS-Bus Protocol	http://convergingsystems.com/bin/doc/ddk_v2%203_g_cs-bus.pdf	"CS-Bus Protocol"

2. Types of Hardware Supported and Interface Adapters Required

Details on (i) how specific products that Converging Systems manufactures can be supported and (ii) the specific protocol(s) required are summarized below.

Targeted Product	Interface Type	
	RS-232C Serial	Ethernet (UDP)
Stewart Masking Products (aka BRIC I command set)	<p>Relevant Document: BRIC I Protocol</p> <p>Detail: RS-232c is built in to every BRIC I device. No additional hardware is required.</p>	<p>Relevant Document: BRIC I Protocol or CS-Bus Protocol depending upon what type of pre-existing driver may already exist. Specifically, if a CS-Bus Driver already exists for alternative CSI products, a Mode 1 driver may be preferable (less work). Alternatively, if a BRIC driver already exists over serial, a Model 2 driver might be the easier process for the developer.</p> <p>Detail: The CSI manufactured e-Node Ethernet device is required regardless of whether a Model 1 or Mode 2 driver is utilized. These Mode 1 and Mode 2 drivers are listed below:</p> <p>Mode 1 enables Ethernet commands customized for CS-Bus motor devices (see IMC-100 below) to be passed through the e-Node to the BRIC I using UDP and the built-in protocol converter resident within the e-Node (selected via software) for the BRIC 1. Since the number of presets in the BRIC I exceeds those generally available with the IMC-100, the</p>

		device driver developer only needs to expand the number of presets support by the CS-Bus driver if this Mode 1 process is desired. Mode 2 enables (RS-232c) commands customized for the BRIC 1 to be routed to the e-Node using UDP alternatively . The exact same command set specified under “RS-232c” in this row can be utilized.
Stewart Intelligent Motor Controller (CS-Bus) IMC-100	Relevant Document: CS-Bus Protocol-Motor Control section Detail: This requires the use of the IBT-100 Serial to CS-adapter available from Converging Systems	Relevant Document: CS-Bus Protocol-Motor Control section Detail: This requires the use of the e-Node Ethernet to CS-Bus adapter available from Converging Systems
Stewart LED lighting options and Converging Systems ILC-100 Intelligent Lighting Controllers	Relevant Document: CS-Bus Protocol-LED Control section Detail: This requires the use of the IBT-100 Serial to CS-adapter available from Converging Systems	Relevant Document: CS-Bus Protocol-LED Control section Detail: This requires the use of the e-Node Ethernet to CS-Bus adapter available from Converging Systems

3. Detail on Drivers and Products Supported

Overview. Before initiating a new device driver or set of device drivers, it is often useful to determine the scope of devices that could be supported by the creation of specific device driver. Also, in some cases alternative device drivers may have already been developed for specific products which in some cases could be easily modified without much work. See the table below for the range of devices, the uniqueness of the drivers required and other relevant information.

Background. Under the Type column are numbers which show the similarity of drivers. All those drivers labeled 2 with various suffixes refer to a single engineered device driver with just various (slight) alternations between the various suffixes (i.e. a, a’, a”). The driver with a “1” in front of it, is of a different structure but was initially created without much complexity. The engineering required to develop “1” type drivers is quite basic. All items in **Yellow** are very similar, and all items in **Green** are separately similar.

Type*	Customer (Brand Name(s) for your Driver)	Product Name	RS-232 Interface Via <specific adapter >	Ethernet Interface Via <specific CSI Ethernet adapter)	Applicable Manual	Bi-Directional Functionality
2a	Mechoshade	IQ485 Network Interface	✓ Via <on-board serial port>	✓ (Mechoshade calls our Ethernet Adapter the Mechoshade IPI)	CS-Bus Manual *	✓ (enhanced capabilities)
1	Stewart Filmscreen	BRIC Masking System (Motor Control)	✓ Via <integrated serial adapter on-board>	Special Case. Our e-Node has a special on-board translator that takes CS-Bus commands and translates them into BRIC commands and communicates	BRIC Manual	✓ (basic)

				those to BRIC using the e-Node's RS-232 optional output port. Note: see note below under **		
2a'	a)Stewart Filmscreen b) Converging Systems Inc.	IMC-100 Intelligent Motor Controller (Motor Control)	✓ Via <separate IBT-100 serial adapter>	✓ Via <e-Node>	CS-Bus Manual	✓(enhanced capabilities)
2a''	a)Stewart Filmscreen b) Converging Systems Inc.	ILC-100 Intelligent LED controller (LED Lighting Control)	✓ Via <separate IBT-100 serial adapter>	✓ Via <e-Node>	CS-Bus Manual	✓(enhanced capabilities)

*Minor text difference in a few commands, but syntax, structure and methodology all the same with the standard CS-Bus Manual. A simple addendum will be provided separately.

Provided STORE and RECALL functions available in CS-Bus manual are implement, a user simply selects the Ethernet Interface and picks an IMC Controller and our e-Node does the translations so the **control developer does not have to create a separate IP solution for this platform. A user just selects the "BRIC" driver for Serial, and the "IMC-100" for IP. Some developers who are interested can create a separate IP driver for BRIC using our e-Node and in this case, e-Node is put into "direct" mode rather than "translate" mode.

4. Summary on Device Drivers

Motor Control. It is generally recommended if motor control is desired, a CS-Bus Motor Control driver for both serial (using the IBT-100) and Ethernet (using the e-Node) be created. The command set supported is identical between the two with the **only** exception that the commands are either directed over a RS-232c channel (using an IBT-100) or a UDP channel (using the e-Node).

LED Control. It is generally recommended if lighting control is desired, a CS-Bus Lighting Control driver for both serial (using the IBT-100) and Ethernet (using the e-Node) be created. The command set supported is identical between the two with the **only** exception that the commands are either directed over a RS-232c channel (using an IBT-100) or a UDP channel (using the e-Node).

Special Note: If no current support for the Stewart BRIC I exists, and support is required only for serial, then a BRIC serial driver should be created. If additional support for Ethernet is required, then the e-Node can be used provided the software developer re-purposes the serial driver to be routed over UDP.

5. Summary on Bi-Directional Drivers

Converging Systems enables various types of bi-directional communication across its various platforms. Refer to the appropriate section below for more information.

Motor Control. The BRIC I command set is basically a one-directional command set. Although there is the capability of deriving *some* controller information, it is preferable to utilize the CS-Bus command set to derive controller information. Specifically, the IMC-100 controller can provide backchannel information as to if a motor has reached its top location (home location) or its fully deployed location (down location generally). As new motor products are supported by CSI, **exact positioning information** rather than just 0% and 100% will be made available. Therefore for new device drivers, it is recommend

that the #Z.G.N.MOTOR.POSITION=?;<cr> command is implemented. A control system's graphical user interface could show a .gif image of a motor or window covering for current systems in either the home position or the deployed position—this certainly raises the level of user convenience and attractiveness for an advance control system! Then later, when other hardware products are released from Converging Systems that have digital positioning information available, the exact location between 0% and 100% could be displayed by the developer with no additional support effort required.

Lighting Control. All lighting products are only supported by BS-Bus commands. There is a full range of color, status, and other information related on LED controllers using this protocol (regardless if the software commands are routed over serial or UDP (Ethernet)). It is highly recommended when supporting CS-Bus LED products, that at least some bi-directional inquiry commands should be supported. Specifically, it is becoming quite popular to display on GUIs or keypads the exact color that is being driven on a CSI lighting device. This information can be requested in either of the two-color models supported by CSI, these being RGB as well as HSB (Hue, Saturation and Brightness). An equally valuable utilization of this feature within the CS-Bus language is for the developer to enable the end user to see exactly the GOTO color after a prior action such as a Fade, or Saturation (change) is initiated. This quite popular feature also enables the inquiry to occur in either of the two supported color models.

6. Support

Developers are encouraged to contact Converging System prior to any software development to obtain the best guidance possible from Converging Systems. Please contact us at 310-544-2628 and ask to speak with the Device Driver Department.