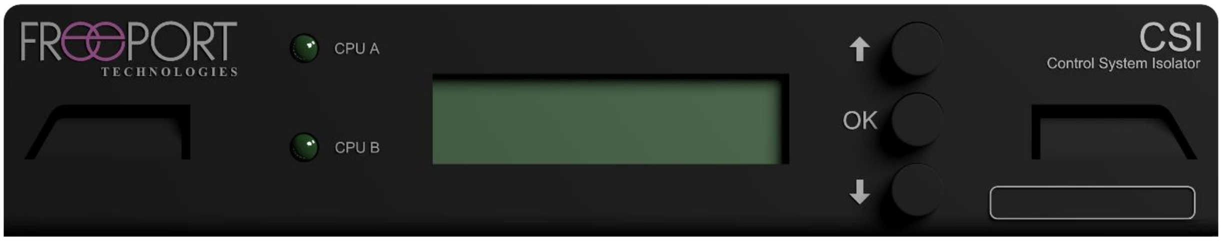
Control System Isolator

Application Programming Interface
v1.3.0

(Revised December 2023)



Point of Contact:
Freeport Technologies Help Desk
571-262-0422
866-226-4125
TechSupport@freeporttech.com

1 Application Programming Interface

1.1 Message Format

All data sent to the CSI from the control system must be encapsulated into a data packet. Packets have the following format:

	Start of Packet	Command Type	Start of Data	Command Data	End of Data	End of Packet
Char	SOH	*Command*	STX	*Data*	ETX	EOT
Hex	0x01	*Command*	0x02	*Data*	0x03	0x04

Message packets begin with an SOH (start of heading: 0x01) marker and end with an EOT (end of transmission: 0x04) marker. Command data fields begin with an STX (start of text: 0x02) marker and end with an ETX (end of text: 0x03) marker. Command data fields can contain any characters or symbols except SOH, STX, ETX, EOT (i.e., 0x01, 0x02, 0x03, 0x04). A message can have multiple command types and command data fields.

Command Types that may be used are:

Type	Function
A	Set CPU B Enable
B	Set Baud Rate
C	Send Device Command
D	Set Timeout Delay
F	Set Device Failure String
L	Send Login Command
O	Send Logout Command
Q	Get CPU B Enable
R	Set Custom Response String
S	Set Device Success String

The statuses that can be returned to the room controller are:

OK	Command accepted, operation done, or success
FAIL	Command failed, operation failed, or error
LOGIN	API user login required using command L
BOOT	CSI CPU A has reset and is ready for commands
R01 to R1000	Customizable responses indicating a device event
RESET	CSI CPU B has reset and is ready for commands

1.2 Multiple Commands in One Message

Multiple commands can be packaged in one message. The most typical grouping is to send the maximum delay “D”, the expected good response “S”, possibly the failure response “F” and the command to send “C”. Each command data field must be preceded by a command type specifying what the content of that command data field represents.

1.3 Status Responses

Certain commands will generate an OK, FAIL, or LOGIN response. The status feedbacks that are possible are shown in the following table.

Type	Function	OK	FAIL	LOGIN	
A	Set CPU B Enable	√	√	√	
B	Set Baud Rate	√	√	√	
C	Send Device Command	√	√	√	Must be last command in a sequence
D	Set Timeout Delay	√	√	√	Usually combined with “C” command
F	Set Device Failure String	√		√	Usually combined with “C” command
L	Send Login Command	√		√	
O	Send Logout Command			√	
Q	Get CPU B Enable	√	√	√	
R	Set Custom Response String	√	√	√	
S	Set Device Success String	√		√	Usually combined with “C” command

1.4 API User Login Required

Starting with CSI v1.3 an API user must be created in the CSI web UI. The API user credentials must be used to login to the CSI on the API communication channel. All API command types will return a LOGIN string prompting for the API user credentials to be sent using the “L” command type. Once the credentials have been validated then the remaining API command types will work the same as they did in CSI v1.2 and previous versions.

1.5 Command Types

1.5.1 *A – Set CPU B Enable*

The CSI is internally divided into two electrically and logically separated circuits. There is a CPU A side and a CPU B side. Each has its own isolated power supply. The CPU B Enable command turns on or off the power to the CPU B side of the unit. When turned off, no control of the isolated device is possible. All CPU B side components are powered off when disabled. All volatile memory is cleared.

1.5.1.1 *Command*

	Start of Packet	Command Type	Start of Data	Command Data	End of Data	End of Packet
Char	SOH	A	STX	*Enable*	ETC	EOT
Hex	0x01	0x41	0x02	*Enable*	0x03	0x04

1.5.1.2 Command Data

- 0 (0x30) False, turn off the CPU B side
- 1 (0x31) True, turn on the CPU B side

1.5.1.3 Response

If a valid Enable command is received, then the CSI will respond with OK:

	Response	New Line	Carriage Return
Char	OK	LF	CR
Hex	0x4F, 0x4B	0x0A	0x0D

If an invalid Enable command is received or the CPU B fails to enable/disable, then the CSI will respond with FAIL:

	Response	New Line	Carriage Return
Char	FAIL	LF	CR
Hex	0x46, 0x41, 0x49, 0x4C	0x0A	0x0D

If a successful "L" command as not already been sent then the CSI will respond with LOGIN:

	Response	New Line	Carriage Return
Char	LOGIN	LF	CR
Hex	0x4C, 0x4F, 0x47, 0x49, 0x4E	0x0A	0x0D

1.5.1.4 Example

To enable the CPU B side send the following packet:

	Start of Packet	Command Type	Start of Data	Command Data	End of Data	End of Packet
Char	SOH	A	STX	1	ETC	EOT
Hex	0x01	0x41	0x02	0x31	0x03	0x04

The CSI will respond with OK:

	Response	New Line	Carriage Return
Char	OK	LF	CR
Hex	0x4F, 0x4B	0x0A	0x0D

1.5.2 B – Set Baud Rate

A "B" command type indicates the next command data field will contain the Baud rate for the isolated device. This Baud rate will be specified in ASCII characters in the data field.

1.5.2.1 *Command*

	Start of Packet	Command Type	Start of Data	Command Data	End of Data	End of Packet
Char	SOH	B	STX	*Baud rate*	ETC	EOT
Hex	0x01	0x42	0x02	*Baud rate*	0x03	0x04

1.5.2.2 *Command Data*

- 9600 (0x39, 0x36, 0x30, 0x30)
- 19200 (0x31, 0x39, 0x32, 0x30, 0x30)
- 38400 (0x33, 0x38, 0x34, 0x30, 0x30)
- 57600 (0x35, 0x37, 0x36, 0x30, 0x30)
- 115200 (0x31, 0x31, 0x35, 0x32, 0x30, 0x30)

1.5.2.3 *Response*

If a valid Baud rate command is received, then the CSI will respond with OK:

	Response	New Line	Carriage Return
Char	OK	LF	CR
Hex	0x4F, 0x4B	0x0A	0x0D

If an invalid Baud rate is received, then the default Baud rate of 115200 will be used and the CSI will respond with FAIL:

	Response	New Line	Carriage Return
Char	FAIL	LF	CR
Hex	0x46, 0x41, 0x49, 0x4C	0x0A	0x0D

If a successful “L” command as not already been sent then the CSI will respond with LOGIN:

	Response	New Line	Carriage Return
Char	LOGIN	LF	CR
Hex	0x4C, 0x4F, 0x47, 0x49, 0x4E	0x0A	0x0D

1.5.2.4 *Example*

To set the isolated device baud rate to 115200 send the following packet:

	Start of Packet	Command Type	Start of Data	Command Data	End of Data	End of Packet
Char	SOH	B	STX	115200	ETC	EOT
Hex	0x01	0x42	0x02	0x31, 0x31, 0x35, 0x32, 0x30, 0x30	0x03	0x04

The CSI will respond with OK:

	Response	New Line	Carriage Return
Char	OK	LF	CR
Hex	0x4F, 0x4B	0x0A	0x0D

1.5.3 *C – Send Device Command*

A "C" command type indicates the next command data field will contain a device specific command. The entire command data field (the bytes between 0x02 and 0x03) is sent to the device exactly as received.

1.5.3.1 *Command*

	Start of Packet	Command Type	Start of Data	Command Data	End of Data	End of Packet
Char	SOH	C	STX	*device command*	ETC	EOT
Hex	0x01	0x43	0x02	*device command*	0x03	0x04

1.5.3.2 *Command Data*

Any valid device command.

1.5.3.3 *Response*

If an expected response is not set with either the "S" or "F" command, then the CSI will respond immediately with an OK.

	Response	New Line	Carriage Return
Char	OK	LF	CR
Hex	0x4F, 0x4B	0x0A	0x0D

If an expected success response is set with the "S" command, then the CSI will wait until a matching response is received from the isolated device. Once a match is received then the CSI will respond with an OK.

If an expected failure response is set with the "F" command, then the CSI will wait until a matching response is received from the isolated device. Once a match is received then the CSI will respond with a FAIL.

	Response	New Line	Carriage Return
Char	FAIL	LF	CR
Hex	0x46, 0x41, 0x49, 0x4C	0x0A	0x0D

If a match is never received, then the "C" command will timeout based on the "D" command delay setting. In this case the CSI will respond with a FAIL.

If a successful "L" command as not already been sent then the CSI will respond with LOGIN:

	Response	New Line	Carriage Return
Char	LOGIN	LF	CR
Hex	0x4C, 0x4F, 0x47, 0x49, 0x4E	0x0A	0x0D

1.5.3.4 Example

An example of a "C" command is:

	Start of Packet	Command Type	Start of Data	Command Data	End of Data	End of Packet
Char	SOH	C	STX	vidin LF CR	ETC	EOT
Hex	0x01	0x43	0x02	0x76, 0x69, 0x64, 0x69, 0x6E, 0x0A, 0x0D	0x03	0x04

1.5.4 D – Set Timeout Delay

A "D" command type indicates the next command data field will contain the ASCII value for the maximum delay that will be allowed in waiting for a good response. This delay is in 10ths of seconds (i.e. deciseconds). If the expected response (specified in an "S" command data field) is not received within the specified time period, the CSI will respond with FAIL.

1.5.4.1 Command

	Start of Packet	Command Type	Start of Data	Command Data	End of Data	End of Packet
Char	SOH	D	STX	*Timeout delay*	ETC	EOT
Hex	0x01	0x44	0x02	*Timeout delay*	0x03	0x04

1.5.4.2 Command Data

ASCII value representing deciseconds.

- Default value: 10 ds
- Minimum value: 1 ds
- Maximum value: 600 ds

If the command data passed is outside the allowable range, then the default value, 10 ds, will be used.

1.5.4.3 Response

If a valid Delay command is received, then the CSI will respond with OK:

	Response	New Line	Carriage Return
Char	OK	LF	CR
Hex	0x4F, 0x4B	0x0A	0x0D

If an invalid Delay command is received, then the delay value will be reset to the default, 10 ds, and the CSI will respond with FAIL:

	Response	New Line	Carriage Return
Char	FAIL	LF	CR
Hex	0x46, 0x41, 0x49, 0x4C	0x0A	0x0D

If a successful "L" command as not already been sent then the CSI will respond with LOGIN:

	Response	New Line	Carriage Return
Char	LOGIN	LF	CR
Hex	0x4C, 0x4F, 0x47, 0x49, 0x4E	0x0A	0x0D

1.5.4.4 Example

To set a delay of 1.5 seconds send the following packet:

	Start of Packet	Command Type	Start of Data	Command Data	End of Data	End of Packet
Char	SOH	D	STX	15	ETC	EOT
Hex	0x01	0x44	0x02	0x31, 0x35	0x03	0x04

The CSI will respond with OK:

	Response	New Line	Carriage Return
Char	OK	LF	CR
Hex	0x4F, 0x4B	0x0A	0x0D

1.5.5 F – Set Device Failure String

An "F" command type indicates the next command data field will contain a string that will be sent from the isolated device indicating a failure. The FAIL response is sent when a match is detected.

1.5.5.1 Command

	Start of Packet	Command Type	Start of Data	Command Data	End of Data	End of Packet
Char	SOH	F	STX	*Failure string*	ETC	EOT
Hex	0x01	0x46	0x02	*Failure string*	0x03	0x04

1.5.5.2 Command Data

Expected device response indicating a command failure.

1.5.5.3 Response

The CSI will normally respond with OK:

	Response	New Line	Carriage Return
Char	OK	LF	CR
Hex	0x4F, 0x4B	0x0A	0x0D

If a successful “L” command as not already been sent then the CSI will respond with LOGIN:

	Response	New Line	Carriage Return
Char	LOGIN	LF	CR
Hex	0x4C, 0x4F, 0x47, 0x49, 0x4E	0x0A	0x0D

1.5.5.4 Example

To set a failure string to “error” send the following packet:

	Start of Packet	Command Type	Start of Data	Command Data	End of Data	End of Packet
Char	SOH	F	STX	error	ETC	EOT
Hex	0x01	0x46	0x02	0x65, 0x72, 0x72, 0x6F, 0x72	0x03	0x04

The CSI will respond with OK:

	Response	New Line	Carriage Return
Char	OK	LF	CR
Hex	0x4F, 0x4B	0x0A	0x0D

1.5.6 L – Send Login Command

An “L” command type indicates the next command data field will contain a username and password. Whenever the LOGIN response is received the “L” command must be sent to authenticate the user credentials. Without a successful “L” command the CSI will continue to transmit a LOGIN response to any other command.

1.5.6.1 Command

	Start of Packet	Command Type	Start of Data	Command Data	End of Data	End of Packet
Char	SOH	L	STX	*username,password string*	ETC	EOT
Hex	0x01	0x4C	0x02	*username,password string*	0x03	0x04

1.5.6.2 Command Data

A comma separated string containing *username,password*. This username and password must be created in the CSI web UI. The associated user must have the API role to allow it to successfully login and use the other API commands.

1.5.6.3 Response

If the *username,password* supplied is valid and associated with a user having the API role then CSI will respond with OK:

	Response	New Line	Carriage Return
Char	OK	LF	CR
Hex	0x4F, 0x4B	0x0A	0x0D

If the *username,password* supplied is invalid or the user does not have the API role then CSI will respond with LOGIN:

	Response	New Line	Carriage Return
Char	LOGIN	LF	CR
Hex	0x4C, 0x4F, 0x47, 0x49, 0x4E	0x0A	0x0D

1.5.6.4 Example

To login with an API user called RC and a password set to E@sy send the following packet:

	Start of Packet	Command Type	Start of Data	Command Data	End of Data	End of Packet
Char	SOH	L	STX	RC,E@sy	ETC	EOT
Hex	0x01	0x4C	0x02	0x52, 0x43, 0x2C, 0x45, 0x40,0x73,0x79	0x03	0x04

The CSI will respond with OK:

	Response	New Line	Carriage Return
Char	OK	LF	CR
Hex	0x4F, 0x4B	0x0A	0x0D

1.5.7 O – Send Logout Command

An “O” command type will logout the current API user. All subsequent responses from the CSI will be LOGIN until an “L” command is sent with the correct API username and password.

1.5.7.1 Command

	Start of Packet	Command Type	Start of Data	Command Data	End of Data	End of Packet
Char	SOH	O	STX	*Empty*	ETC	EOT
Hex	0x01	0x4F	0x02	*Empty*	0x03	0x04

1.5.7.2 Command Data

The command data field is left empty. Note the Start of Data and End of Data characters still need to be included in order to have a properly constructed packet.

1.5.7.3 Response

The CSI will always respond with LOGIN:

	Response	New Line	Carriage Return
Char	LOGIN	LF	CR
Hex	0x4C, 0x4F, 0x47, 0x49, 0x4E	0x0A	0x0D

1.5.7.4 *Example*

To logout from the current API user session send the following packet:

	Start of Packet	Command Type	Start of Data	Command Data	End of Data	End of Packet
Char	SOH	O	STX		ETC	EOT
Hex	0x01	0x4F	0x02		0x03	0x04

The CSI will respond with LOGIN:

	Response	New Line	Carriage Return
Char	LOGIN	LF	CR
Hex	0x4C, 0x4F, 0x47, 0x49, 0x4E	0x0A	0x0D

1.5.8 *Q – Get CPU B Enable*

The “Q” command provides a read-only query of the current CPU B power-on status.

1.5.8.1 *Command*

	Start of Packet	Command Type	Start of Data	Command Data	End of Data	End of Packet
Char	SOH	Q	STX	*Empty*	ETC	EOT
Hex	0x01	0x51	0x02	*Empty*	0x03	0x04

1.5.8.2 *Command Data*

The command data field is left empty. Note the Start of Data and End of Data characters still need to be included in order to have a properly constructed packet.

1.5.8.3 *Response*

If the CPU B side is powered on (true), then the CSI will respond with OK:

	Response	New Line	Carriage Return
Char	OK	LF	CR
Hex	0x4F, 0x4B	0x0A	0x0D

If the CPU B side is powered off (false), then the CSI will respond with FAIL:

	Response	New Line	Carriage Return
Char	FAIL	LF	CR
Hex	0x46, 0x41, 0x49, 0x4C	0x0A	0x0D

If a successful “L” command as not already been sent then the CSI will respond with LOGIN:

	Response	New Line	Carriage Return
Char	LOGIN	LF	CR
Hex	0x4C, 0x4F, 0x47, 0x49, 0x4E	0x0A	0x0D

1.5.8.4 Example

To get the current CPU B side Enabled status send the following packet:

	Start of Packet	Command Type	Start of Data	Command Data	End of Data	End of Packet
Char	SOH	Q	STX		ETC	EOT
Hex	0x01	0x51	0x02		0x03	0x04

The CSI will respond with OK if the CPU B side Enabled is already set:

	Response	New Line	Carriage Return
Char	OK	LF	CR
Hex	0x4F, 0x4B	0x0A	0x0D

1.5.9 R01 – R1000 – Set Device Custom Response Strings

A “R” command type indicates the next command data field will contain an event string. This string may be sent from the isolated device as an event message or as a response to the issuance of a “C” command.

1.5.9.1 Command

	Start of Packet	Command Type	Start of Data	Command Data	End of Data	End of Packet
Char	SOH	R####	STX	*Event Message*	ETC	EOT
Hex	0x01	0x52, 0x3#, 0x3#, 0x3#, 0x3#	0x02	*Event Message*	0x03	0x04

1.5.9.2 Command Type Index

A “R” command type must include an index indicating one of 1000 customizable responses. If a “R14” command is issued with the command data “Hello World” then each time the isolated device responds with “Hello World” the CSI will respond with “R14” back to the room controller.

- 01 (0x30, 0x31) for custom response “R01”

- 02 (0x30, 0x32) for custom response “R02”

  •

  •

  •

- 1000 (0x31, 0x30, 0x30, 0x30) for custom response “R1000”

1.5.9.3 Command Data

Any valid device event or response string that needs to be monitored. The string must be at least 6 characters long.

1.5.9.4 Response

If a valid Custom Response command is received, then the CSI will respond with OK:

	Response	New Line	Carriage Return
Char	OK	LF	CR
Hex	0x4F, 0x4B	0x0A	0x0D

If an invalid Custom Response command is received, then the CSI will respond with FAIL:

	Response	New Line	Carriage Return
Char	FAIL	LF	CR
Hex	0x46, 0x41, 0x49, 0x4C	0x0A	0x0D

If a successful “L” command as not already been sent then the CSI will respond with LOGIN:

	Response	New Line	Carriage Return
Char	LOGIN	LF	CR
Hex	0x4C, 0x4F, 0x47, 0x49, 0x4E	0x0A	0x0D

When a matching event is received from the isolated device, then the CSI will respond with R01 through R1000 that corresponds with the matched event:

	Response	New Line	Carriage Return
Char	R01	LF	CR
Hex	0x52, 0x30, 0x31	0x0A	0x0D

	Response	New Line	Carriage Return
Char	R02	LF	CR
Hex	0x52, 0x30, 0x32	0x0A	0x0D

•

•

•

	Response	New Line	Carriage Return
Char	R1000	LF	CR
Hex	0x52, 0x31, 0x30, 0x30, 0x30	0x0A	0x0D

1.5.9.5 *Example: Ring event*

To monitor a Ring event string and assign it to custom response R01 send the following packet:

	Start of Packet	Command Type	Start of Data	Command Data	End of Data	End of Packet
Char	SOH	R01	STX	*e IncomingCallIndication	ETC	EOT
Hex	0x01	0x52, 0x30, 0x31	0x02	0x2A 0x65 0x20 0x49 0x6E 0x63 0x6F 0x6D 0x69 0x6E 0x67 0x43 0x61 0x6C 0x6C 0x49 0x6E 0x64 0x69 0x63 0x61 0x74 0x69 0x6F 0x6E	0x03	0x04

The CSI will respond with OK:

	Response	New Line	Carriage Return
Char	OK	LF	CR
Hex	0x4F, 0x4B	0x0A	0x0D

1.5.9.6 *Example: Hangup event*

To monitor a Call Hangup event string and assign it to custom response R02 send the following packet:

	Start of Packet	Command Type	Start of Data	Command Data	End of Data	End of Packet
Char	SOH	R02	STX	*e CallDisconnect	ETC	EOT
Hex	0x01	0x52, 0x30, 0x32	0x02	0x2A 0x65 0x20 0x43 0x61 0x6C 0x6C 0x44 0x69 0x73 0x63 0x6F 0x6E 0x6E 0x65 0x63 0x74	0x03	0x04

The CSI will respond with OK:

	Response	New Line	Carriage Return
Char	OK	LF	CR
Hex	0x4F, 0x4B	0x0A	0x0D

1.5.9.7 *Example: Call up event*

To monitor a Call Up event string and assign it to custom response R03 send the following packet:

	Start of Packet	Command Type	Start of Data	Command Data	End of Data	End of Packet
Char	SOH	R03	STX	*e CallSuccessful	ETC	EOT
Hex	0x01	0x52, 0x30, 0x33	0x02	0x2A 0x65 0x20 0x43 0x61 0x6C 0x6C 0x53 0x75 0x63 0x63 0x65 0x73 0x73 0x66 0x75 0x6C	0x03	0x04

The CSI will respond with OK:

	Response	New Line	Carriage Return
Char	OK	LF	CR
Hex	0x4F, 0x4B	0x0A	0x0D

1.5.10 S – Set Device Success String

A "S" command type indicates the next command data field will contain a string that will be sent from the isolated device indicating the success of a command. The OK response is sent when a match is detected.

1.5.10.1 Command

	Start of Packet	Command Type	Start of Data	Command Data	End of Data	End of Packet
Char	SOH	S	STX	*Success string*	ETC	EOT
Hex	0x01	0x53	0x02	*Success string*	0x03	0x04

1.5.10.2 Command Data

Expected device response indicating success.

1.5.10.3 Response

The CSI will normally respond with OK:

	Response	New Line	Carriage Return
Char	OK	LF	CR
Hex	0x4F, 0x4B	0x0A	0x0D

If a successful "L" command as not already been sent then the CSI will respond with LOGIN:

	Response	New Line	Carriage Return
Char	LOGIN	LF	CR
Hex	0x4C, 0x4F, 0x47, 0x49, 0x4E	0x0A	0x0D

1.5.10.4 Example

To set a success string to "OK" send the following packet:

	Start of Packet	Command Type	Start of Data	Command Data	End of Data	End of Packet
Char	SOH	F	STX	OK	ETC	EOT
Hex	0x01	0x46	0x02	0x4F, 0x4B	0x03	0x04

The CSI will respond with OK:

	Response	New Line	Carriage Return
Char	OK	LF	CR
Hex	0x4F, 0x4B	0x0A	0x0D

2 Error Handling

The CSI unit sits in the middle of the serial communication path between the controller and the isolated device. This blocks serial communication errors from reaching the controller causing the controller to not know what, if any errors might be occurring. Some general defensive coding techniques in the controller code can be deployed to alleviate the most common serial communication errors.

2.1 Delay Serial Commands

Select a reasonable delay for sending commands and waiting for responses. If errors are being returned by the CSI, try increasing the delay value until correct responses are occurring. This delay value will depend on the speed of the isolated device. Some devices are slower or faster than others in handling serial commands.

2.2 Stabilize Communication after Reboot

After the isolated device is initially powered (or rebooted) the controller code should wait until the isolated device has stabilized and is ready for serial communication. This can be determined by sending a simple serial command to the isolated device and receiving a successful response multiple times (e.g. three or four successful responses to the same command). This reasonably indicates that the serial communication path is stable and ready for full communication.

2.3 Resend Commands

Even with a proper command delay and a stabilized communication path serial errors can still occur. Proper error handling code in the controller should reattempt to send a command a second or even third time before employing a more aggressive error handling technique.

2.4 Reset Serial Connection

If continued errors occur the controller can reset the isolated device serial connection by sending a CSI Set Baud Rate command. This command will cause the CSI to disable/enable the serial port and flush the serial transmit line (by writing carriage return and line feed characters) and serial receive line (by reading several characters). If necessary, the controller can send one or more initial commands that are specific to the device for flushing the serial line and ensuring the device is ready for serial commands.

2.5 Controller Responsible for Error Handling

Serial line errors may be introduced simply by plugging or unplugging the serial cable between the CSI and the isolated device. This typically does not occur during normal operation, but any erroneous or corrupted characters inserted into the serial communication path need to be handled by the controller code.

In all cases the controller knows how the isolated device should behave and therefore the controller code will know how to handle errors. The CSI is purposefully ignorant and agnostic regarding the functionality of the isolated device. The CSI will homogenize both proper and improper serial communication responses to OK, FAIL, R01 to R1000, RESET, or BOOT. It is up to the controller to determine what those homogenized responses really mean in context of the commands sent and the functionality of the isolated device.