NetLinx Control Cards and NetModules

NetLinx Control Cards can be installed in either the NXF CardFrame, NI-4000, or NetModules. This document provides basic specifications and wiring information for the NetLinx Control cards. For detailed information on the cards, refer to the NetLinx CardFrame, Control Cards, and NetModules Instruction Manual.

NXC-IRS4 4-Port IR/S Control Card

The NXC-IRS4 4-Port IR/S Control Card provides four IR/Serial output control ports with LED status feedback. Each port in the NXC-IRS4 stores programmed commands for IR- or serial-controlled devices.

Specifications

NXC-IRS4 (FG2023) Specifications	
Power	110 mA @ +12 VDC power for sensors
Operation:	• IR 1-4: 4 IR/Serial control ports • Input 1-4: 4 input ports for closure or 0-5 VDC sensing, 200 mA
Memory	32K of IR memory shared between four ports.
IR Frequency range	Support of high-frequency carriers up to 1.14 MHz.
Status LEDs:	Red LED shows IR transmission activity / yellow LED shows input status activity (2 LEDs per channel). LEDs light to indicate ON status: • LED 1: Channel 1 Input (yellow) • LED 2: Channel 1 IR Out (red) • LED 3: Channel 2 Input (yellow) • LED 4: Channel 2 IR Out (red) • LED 5: Channel 3 Input (yellow) • LED 6: Channel 3 IR Out (red) • LED 7: Channel 4 Input (yellow) • LED 8: Channel 4 IR Out (red)
Connections/wiring:	• Two 2-pin 3.5 mm captive-screw terminals • Two CC-NIRC IR Emitters
Models:	• NXC-IRS4 4-Port IR/S Card • NXM-IRS4 4-Port IR/S NetModule
DEVICE_ID:	$0108

Pinouts, Signals, and Functions

NXC-IRS4 Pinouts, Signals, and Functions					
Pin	Signal	Function	Pin	Signal	Function
1	GND	Signal ground	10	Input #1	Logic input
2	Output #1	IR data	11	Input #2	Logic input
3	GND	Signal ground	12	Input #3	Logic input
4	Output #2	IR data	13	Input #4	Logic input
5	GND	Signal ground	14	Power	+12 VDC
6	Output #3	IR data	15	---------------	no connection
7	GND	Signal ground	16	---------------	no connection
8	Output #4	IR data	17	---------------	no connection
9	GND	Signal ground	18	---------------	no connection
			19	---------------	no connection
			20	---------------	no connection

NXC-IRS4 Channel Assignments

The IRS4 channel settings listed in the following table set the IR output channels. The NXC-IRS4 can process up to two IR or serial device channel setting commands simultaneously. If more than two device commands are sent simultaneously, only the first two devices receive the commands.

NXC-IRS4 Channel Assignments	
Channel	Description
1-255	Generates the IR or serial command assigned to that channel.
1-199	Provides intelligent feedback; if a channel with no IR command is turned on, the card will turn that channel off.
Ports 1-4	Generates PUSH and RELEASE statements corresponding to the state of inputs 1 - 4. A contact closure to GND is reported as a PUSH. • The PUSH and RELEASE channel is 255. • Channel reporting status is 255. • Channel 255 changes are disabled after receipt of the 'PON' command.

Programming Information

NXC-IRS4 SEND_COMMANDS	

CAROFF
Disable the IR carrier signal until a 'CARON' command is received.
	Syntax:
	`SEND_COMMAND <DEV>,"'CAROFF'"`
	Example:
	`SEND_COMMAND IR_1,"'CAROFF'"`
	Stops transmitting IR carrier signals to the IR_1 port.

CARON
Enable the IR carrier signals (default).
	Syntax:
	`SEND_COMMAND <DEV>,"'CARON'"`
	Example:
	`SEND_COMMAND IR_1,"'CARON'"`
	Starts transmitting IR carrier signals to the IR_1 port.

CH
Send IR pulses for the selected a channel.
	All channels below 100 are transmitted as two digits. If the IR code for ENTER (function #21) is loaded, an Enter will follow the number. If the channel is greater than or equal to (>=) 100, then IR function 127 or 20 (whichever exists) is generated for the one hundred digit. Uses 'CTON' and 'CTOF' times for pulse times.
	Syntax:
	`SEND_COMMAND <DEV>,"'CH',<Number>"`
	Channel number = 0 - 199.
	Example:
	`SEND_COMMAND IR_1,"'CH',18"`
	The NXC-IRS4 performs the following:
	• Transmits IR signals for 1 (IR code 11). The transmit time is set with the CTON command.
	• Waits until the time set with the CTOF command elapses.
	• Transmits IR signals for 8 (IR code 18).
	• Waits for the time set with the CTOF command elapses. If the IR code for Enter (IR code 21) is programmed, the IRS4 performs the following steps.
	• Transmits IR signals for Enter (IR code 21).
	• Waits for the time set with the CTOF command elapses.

CP
Halt and Clear all active or buffered IR commands, and then send a single IR pulse. Set the Pulse and Wait times with the 'CTON' and 'CTOF' commands.
	Syntax:
	`SEND_COMMAND <DEV>,"'CP',<code>"`
	Code = IR port's channel value 0 - 252 (253 - 255 reserved).
	Example:
	`SEND_COMMAND IR_1,"'CP',2"`
	Clears the active/buffered commands and pulses IR_1 port's channel 2.

CTOF
Set the duration of the Off time (no signal) between IR pulses for channel and IR function transmissions. Off time settings are stored in non-volatile memory.
	This command sets the delay time between pulses generated by the 'CH' or 'XCH' send commands in tenths of seconds.
	Syntax:
	`SEND_COMMAND <DEV>,"'CTOF',<time>"`
	Time = 0 - 255. Given in 1/10ths of a second. Default is 5 (0.5 sec).
	Example:
	`SEND_COMMAND IR_1,"'CTOF',10"`
	Sets the Off time between each IR pulse to 1 second.

CTON
Set the total time of IR pulses transmitted and is stored in non-volatile memory.
	This command sets the pulse length for each pulse generated by the 'CH' or 'XCH' send commands in tenths of seconds.
	Syntax:
	`SEND_COMMAND <DEV>,"'CTON',<time>"`
	Time= 0 - 255. Given in 1/10ths of a second. Default is 5 (0.5 sec).
	Example:
	`SEND_COMMAND IR_1,"'CTON',20"`
	Sets the IR pulse duration to 2 seconds.

GET MODE
Poll the IR/Serial port's configuration parameters and report the active mode settings to the device requesting the information.
	The port responds with: <port #> <mode>,<carrier>,<io link channel>.
	Syntax:
	`SEND_COMMAND <DEV>,"'GET MODE'"`
	Example:
	`SEND_COMMAND IR_1,"'GET MODE"`
	The system could respond with:
	`PORT 4 IR,CARRIER,IO LINK 0`

IROFF
Halt and Clear all active or buffered IR commands being output on the designated port.
	Syntax:
	`SEND_COMMAND <DEV>,"'IROFF'"`
	Example:
	`SEND_COMMAND IR_1,"'IROFF"`
	Immediately halts and clears all IR output signals on the IR_1 port.

POD

Disable previously active 'PON' (power on) or 'POF' (power off) command settings.
This command is used in conjunction with the I/O Link command.

Channel 255 changes are enabled.

Syntax:
```
SEND_COMMAND <DEV>,"'POD'"
```
Example:
```
SEND_COMMAND IR_1,"'POD'"
```
Disables the 'PON' and 'POF' command settings on the IR_1 device.

POF

Turn OFF a device connected to an IR port based on the status of the corresponding I/O Link input.

If at any time the IR sensor input reads that the device is ON (such as if someone turned it on manually at the front panel), IR function 28 (if available) or IR function 9 is automatically generated in an attempt to turn the device back OFF. If three attempts fail, the IR port will continue executing commands in the buffer. If there are no commands in the buffer, the IR port will continue executing commands in the buffer and trying to turn the device OFF until a 'PON' or 'POD' command is received.

If the IR port fails to turn the device OFF, a PUSH and RELEASE is made on channel 254 to indicate a power failure error. You can only use the 'PON' and 'POF' commands when an IR device has a linked I/O channel. Channel 255 changes are disabled after receipt of this command. You can only use the PON and POF commands when an IR device has a linked I/O channel.

Syntax:
```
SEND_COMMAND <DEV>,"'POF'"
```
Example:
```
SEND_COMMAND IR_1,"'POF'"
```
Sends power down IR commands 28 (if present) or 9 to the IR_1 device.

PON

Turn ON a device connected to an IR port based on the status of the corresponding I/O Link input.

If at any time the IR sensor input reads that the device is OFF (such as if one turned it off manually at the front panel), IR function 27 (if available) or IR function 9 is automatically generated in an attempt to turn the device back ON. If three attempts fail, the IR port will continue executing commands in the buffer and trying to turn the device On.

If there are no commands in the buffer, the IR port will continue trying to turn the device ON until a 'POF' or 'POD' command is received. If the IR port fails to turn the device ON, a PUSH and RELEASE is made on channel 254 to indicate a power failure error.

You can only use the 'PON' and 'POF' commands when an IR device has a linked I/O channel. Channel 255 changes are disabled after receipt of this command.

Syntax:
```
SEND_COMMAND <DEV>,"'PON'"
```
Example:
```
SEND_COMMAND IR_1,"'PON'"
```
Sends power up IR commands 27 or 9 to the IR_1 port.

PTOF

Set the time duration between power pulses in .10-second increments. Time is given in 1/10ths of a second.
This time increment is stored in permanent memory.

This command also sets the delay between pulses generated by the 'PON' or 'POF' send commands in tenths of seconds. It also sets the delay required after a power ON command before a new IR function can be generated. This gives the device time to power up and get ready for future IR commands. It also sets the delay required after a power ON command before a new IR function can be generated. This gives the device time to power up and get ready for future IR commands.

Syntax:
```
SEND_COMMAND <DEV>,"'PTOF',<time>"
```
Time = 0 - 255. Default is 15 (1.5 seconds).
Example:
```
SEND_COMMAND IR_1,"'PTOF',15"
```
Sets the time between power pulses to 1.5 seconds for the IR_1 device.

PTON

Set the time duration between power pulses in .10-second increments. Time is given in 1/10ths of a second.
This time increment is stored in permanent memory.

This command also sets the pulse length for each pulse generated by the 'PON' or 'POF' send commands in tenths of seconds.

Syntax:
```
SEND_COMMAND <DEV>,"'PTON',<time>"
```
Time = 0 - 255. Default is 5 (0.5 seconds).
Example:
```
SEND_COMMAND IR_1,"'PTON',15"
```
Sets the duration of the power pulse to 1.5 seconds for the IR_1 device.

SET INPUT LINK

Sets an IR device to link to an input channel for use with 'PON' and 'POF' commands.
The input channel is used for power sensing (via a PCS).

Syntax:
```
SEND_COMMAND <DEV>,"'SET INPUT LINK <channel>'"
```
Channel = 1 - 4. Setting the channel to 0 disables the link.
Examples:
```
SEND_COMMAND IR_1,"'SET INPUT LINK 0'"
SEND_COMMAND IR_1,"'SET INPUT LINK 1'"
```
Sets the IR_1 port link to I/O channel 1.

SET IO LINK

Link an IR or Serial port to a selected I/O channel for use with the 'DE', 'POD', 'PON', and 'POF' commands.

Syntax:
```
SEND_COMMAND <DEV>,"'SET IO LINK <I/O number>'"
```
I/O number = 1 - 4. An I/O channel to 0 disables the I/O link.
Example:
```
SEND_COMMAND IR_1,"'SET INPUT LINK 1'"
```
Sets the IR_1 port link to I/O channel 1. The IR port uses the specified input as power status for processing PON and POF commands.

Note: This command is included for older programs only. The Input port is not an I/O. No output functions are available.'SET IO LINK' applies the same configuration as the 'SET INPUT LINK' command.

SET MODE

Set the IR/Serial ports for IR or Serial-controlled devices connected to a CardFrame or NetModule.
Sets an IR port to either IR or Serial mode.

Syntax:
```
SEND_COMMAND <DEV>, 'SET MODE <mode>'"
```
Mode = IR or Serial.
Examples:
```
SEND_COMMAND IR_1,"'SET MODE IR'"
```
Sets the IR_1 port to IR mode for IR control.

SP

Generate a single IR pulse.

You can use the 'CTON' to set pulse lengths and the 'CTOF' for time off between pulses.

Syntax:
```
SEND_COMMAND <DEV>,"'SP',<code>"
```
Code = IR code value 1 - 252 (253-255 reserved).
Example:
```
SEND_COMMAND IR_1, "'SP',25"
```
Pulses IR code 25 on IR_1 device.

XCH

Transmit the selected channel IR codes in the format/pattern set by the 'XCHM' send command.

Syntax:
```
SEND_COMMAND <DEV>,"'XCH <channel>'"
```
Channel = 0 - 999.
Example:
For detailed usage examples, refer to the 'XCHM' command.

XCHM

Changes the IR output pattern for the 'XCH' send command.

Syntax:
```
SEND_COMMAND <DEV>,"'XCHM <extended channel mode>'"
```
Extended channel mode = 0 - 4.
Example:
```
SEND_COMMAND IR_1,"'XCHM 3'"
```
Sets the IR_1 device's extended channel command to mode 3.
Mode 0 Example (default): [x][x]<x><enter>
```
SEND_COMMAND IR_1,"'XCH 3'"
```
Transmits the IR code as 3-enter.
```
SEND_COMMAND IR_1,"'XCH 34'"
```
Transmits the IR code as 3-4-enter.
```
SEND_COMMAND IR_1,"'XCH 343'"
```
Transmits the IR code as 3-4-3-enter.
Mode 1 Example: <x> <x> <x> <enter>
```
SEND_COMMAND IR_1,"'XCH 3'"
```
Transmits the IR code as 0-0-3-enter.
```
SEND_COMMAND IR_1,"'XCH 34'"
```
Transmits the IR code as 0-3-4-enter.
```
SEND_COMMAND IR_1,"'XCH 343'"
```
Transmits the IR code as 3-4-3-enter.
Mode 2 Example: <x> <x> <x>
```
SEND_COMMAND IR_1,"'XCH 3'"
```
Transmits the IR code as 0-0-3.
```
SEND_COMMAND IR_1,"'XCH 34'"
```
Transmits the IR code as 0-3-4.
```
SEND_COMMAND IR_1,"'XCH 343'"
```
Transmits the IR code as 3-4-3.
Mode 3 Example: [[100][100]…] <x> <x>
```
SEND_COMMAND IR_1,"'XCH 3'"
```
Transmits the IR code as 0-3.
```
SEND_COMMAND IR_1,"'XCH 34'"
```
Transmits the IR code as 3-4.
```
SEND_COMMAND IR_1,"'XCH 343'"
```
Transmits the IR code as 100-100-100-4-3.
Mode 4: Mode 4 sends the same sequences as the 'CH' command. Only use Mode 4 with channels 0 - 199.

93-2023   REV: E