Configuring the 4-Port Serial Interface SPA

This chapter provides information about configuring the 4-Port Serial Interface SPA on Cisco ASR 1000 Series Routers.

- Configuration Tasks, on page 1
- Verifying the Interface Configuration, on page 11
- Configuration Examples, on page 12

Configuration Tasks

This section describes how to configure the 4-Port Serial Interface SPA for the Cisco ASR 1000 Series Routers and verify the configuration. For information about managing your system images and configuration files, refer to the following:

- Cisco ASR 1000 Series Aggregation Services Routers Software Configuration Guide
- Cisco IOS Configuration Fundamentals Configuration Guide
- Cisco IOS Configuration Fundamentals Command Reference

Configuring the 4-Port Serial Interface SPA

To configure the 4-Port Serial Interface SPA, complete these steps:

SUMMARY STEPS

1. Router# **configure terminal**
2. Router(config)# **interface serial slot/subslot/port**
3. Router(config-if)# ip address address mask
4. Router(config-if)# clock rate bps

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	Router# **configure terminal**	Enters global configuration mode.
Step 2	Router(config)# **interface serial slot/subslot/port**	Selects the interface to configure and enters interface configuration mode.
Step 3	Router(config-if)# ip address address mask	Sets the IP address and subnet mask. • address—IP address. • mask—Subnet mask.
Step 4	Router(config-if)# clock rate bps	Configures the clock rate for the hardware to an acceptable bit rate per second (bps).

What to do next

Note Each port should first be connected with the appropriate cable before attempting full configuration. Some commands are enabled only based upon the cable type connected to the port.

Note The bandwidth of each interface is 2 MB by default; setting the clock rate does not change the interface bandwidth. We recommend that you configure the bandwidth value with the clock rate command at the DCE and DTE side.

Note A clock rate of 2016000 does not appear in the configuration because it is the default value.

Verifying the Configuration

After configuring the new interface, use the **show** commands to display the status of the new interface or all interfaces, and use the **ping** and **loopback** commands to check connectivity. This section includes the following subsections:

Using show Commands

The table below shows the **show** commands you can use to verify the operation of the 4-Port Serial Interface SPA. Sample displays of the output of selected **show** commands appear in the following section.

Note The outputs that appear in this document may not match the output you receive when running these commands. The outputs in this document are examples only.

Command	Purpose
`Router# show version`	Displays system hardware configuration, the number of each interface type installed, Cisco IOS software version, names and sources of configuration files, and boot images.
`Router# show controllers`	Displays all the current interface processors and their interfaces.
`Router# show controllers serial`	Displays serial line statistics.
`Router# show interfaces type port-adapter-slot-number/interface-port-number`	Displays status information about a specific type of interface (for example, serial).
`Router# show platform`	Displays types of hardware installed in your system and the firmware version.
`Router# show running-config`	Displays the running configuration file.
`Router# show startup-config`	Displays the configuration stored in NVRAM.

Verification Examples

The following is an example of the show version command with the 4-Port Serial Interface SPA:

```
Router# show version
Cisco IOS Software, IOS-XE Software (PPC_LINUX_IOSD-ADVIPSERVICESK9-M), Version 12.2(33)XN1,
 RELEASE SOFTWARE (fc1) Technical Support: http://www.cisco.com/techsupport Copyright (c)
1986-2008 by Cisco Systems, Inc.
Compiled Thu 14-Feb-08 08:59 by
ROM: IOS-XE ROMMON
 mcp-2ru-2 uptime is 4 minutes
Uptime for this control processor is 6 minutes System returned to ROM by reload System image
 file is ''tftp:/tftpboot/mcp-2ru-1/asr1000rp1-advipservicesk9.02.00.01.122-33.XN1.bin''
Last reload reason: Reload command
This product contains cryptographic features and is subject to United States and local
country laws governing import, export, transfer and use. Delivery of Cisco cryptographic
products does not imply third-party authority to import, export, distribute or use encryption.
Importers, exporters, distributors and users are responsible for compliance with U.S. and
local country laws. By using this product you agree to comply with applicable laws and
regulations. If you are unable to comply with U.S. and local laws, return this product
immediately.
A summary of U.S. laws governing Cisco cryptographic products may be found at:
http://www.cisco.com/wwl/export/crypto/tool/stqrg.html
If you require further assistance please contact us by sending email to export@cisco.com.
cisco ASR1002 (RP1) processor with 1647841K/6147K bytes of memory.
4 Gigabit Ethernet interfaces
```

```
6 Serial interfaces
2 Channelized T3 ports
32768K bytes of non-volatile configuration memory.
3728208K bytes of physical memory.
7798783K bytes of eUSB flash at bootflash:.
Configuration register is 0x0
```

The following is an example of the show controllers serial command with the 4-Port Serial Interface SPA:

```
Router# show controllers serial 2/1/1

Serial2/1/1 - (SPA-4XT-SERIAL) is up
 Encapsulation: Frame Relay
 Cable type: RS-232 DTE
 mtu 1500, max_buffer_size 1524, max_pak_size 1608 enc 84
 loopback: Off,  crc: 16, invert_data: Off
 nrzi: Off, idle char: Flag
 tx_invert_clk: Off, ignore_dcd: Off
 rx_clockrate: 552216, rx_clock_threshold: 0
 serial_restartdelay:60000,  serial_restartdelay_def:60000

     RTS up, CTS up, DTR up, DCD up, DSR up
```

Note The acronyms are defined as follows: RTS (Request to Send); CTS (Clear To Send); DTR (Data Transmit Ready); DCD (Data Carrier Detect); DSR (Data Set Ready).

The following is an example of the show interfaces serial command with the 4-Port Serial Interface SPA:

```
Router# show interfaces serial2/0/0
Serial 2/0/0 is up, line protocol is up
Hardware is SPA-4XT-SERIAL
  Internet address is 192.168.33.1/29
  MTU 4470 bytes, BW 8000 Kbit, DLY 100 usec, rely 255/255, load 1/255
  Encapsulation HDLC, loopback not set, keepalive not set
         Clock Source Internal.
  Last input 00:00:01, output 00:00:00, output hang never
  Last clearing of "show interface" counters 1h
  Output queue 0/40, 0 drops; input queue 0/75, 0 drops
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     0 packets input, 0 bytes, 0 no buffer
     Received 0 broadcasts, 0 runts, 0 giants, 0 parity
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     0 packets output, 0 bytes, 0 underruns
     0 output errors, 0 applique, 0 interface resets
     0 output buffer failures, 0 output buffers swapped out
     0 carrier transitions
```

The following is an example of the show platform command:

```
Router# show platform

Chassis type: ASR1002
Slot      Type                State                 Insert time (ago)
--------- ------------------- --------------------- -----------------
0         ASR1002-SIP10       ok                    5d04h
 0/0      4XGE-BUILT-IN       ok                    5d04h
 0/1      SPA-4XT-SERIAL      ok                    4d05h
 0/2      SPA-4XT3/E3         ok                    4d04h
```

```
 0/3      SPA-8XCHT1/E1       ok                    4d04h
R0        ASR1002-RP1         ok, active            5d04h
F0        ASR1000-ESP5        ok, active            5d04h
P0        ASR1002-PWR-AC      ok                    5d04h
P1        ASR1002-PWR-AC      ok                    5d04h
Slot      CPLD Version        Firmware Version
--------- ------------------- ------------------------------------
0         07092701            12.2(33r)XN2
R0        07100216            12.2(33r)XN2
F0        07091401            12.2(33r)XN2
```

The following is an example of the show running-config command with the 4-Port Serial Interface SPA:

```
Router# show running-config serial
Building configuration...
Current configuration : 54 bytes
!
interface Serial2/0/0
 no ip address
 shutdown
end
```

The following is an example of the show running interface command with the 4-Port Serial Interface SPA:

```
Router# show running interface ser2/0/1
Building configuration...
Current configuration : 54 bytes
!
interface Serial2/0/1
 no ip address
 shutdown
end
```

The following is an example of the show startup-config command with the 4-Port Serial Interface SPA:

```
Router# show startup-config | b Serial2/0/0
interface Serial2/0/0
 no ip address
 shutdown
!
interface Serial2/0/1
 no ip address
 shutdown
!
interface Serial2/0/2
 no ip address
 shutdown
!
interface Serial2/0/3
 no ip address
 shutdown
!
```

Using the ping Command to Verify Network Connectivity

Use the **ping** command to verify if an interface port is functioning properly. This command sends echo request packets out to a remote device at an IP address that you specify. After sending an echo request, the system waits a specified time for the remote device to reply. Each echo reply is displayed as an exclamation point (!) on the console terminal; each request that is not returned before the specified timeout is displayed as a period (.). A series of exclamation points (!!!!!) indicates a good connection; a series of periods (.....) or the messages [timed out] or [failed] indicate a bad connection.

Following is an example of a successful **ping** command to a remote server with the address 10.0.0.10:

```
Router# ping 10.0.0.10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echoes to 10.0.0.10, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/15/64 ms
Router#
```

If the connection fails, verify that you have the correct IP address for the destination and that the device is active (powered on), and repeat the **ping** command.

Proceed to the next section to check the network connectivity.

Using loopback Commands

With the loopback test, you can detect and isolate equipment malfunctions by testing the connection between the 4-Port Serial Interface SPA and a remote device such as a modem or a channel service unit (CSU) or a data service unit (DSU). The **loopback** command places an interface in loopback mode, which enables test packets that are generated from the **ping** command to loop through a remote device or compact serial cable. If the packets complete the loop, the connection is good. If not, you can isolate a fault to the remote device or compact serial cable in the path of the loopback test.

Note You must configure a clock rate on the port *before* performing a loopback test. However, if no cable is attached to the port, the port is administratively up, and the port is in loopback mode; you do not have to configure a clock rate on the port *before* performing a loopback test.

Depending on the mode of the port, issuing the **loopback** command checks the following path:

- When no compact serial cable is attached to the 4-Port Serial Interface SPA port, or if a data communication equipment (DCE) cable is attached to a port that is configured as line protocol up, the **loopback** command tests the path between the network processing engine and the interface port only (without leaving the network processing engine and port adapter).
- When a data terminal equipment (DTE) cable is attached to the port, the **loopback** command tests the path between the network processing engine and the near (network processing engine) side of the DSU or modem to test the 4-Port Serial Interface SPA and compact serial cable. (The X.21 DTE interface cable does not support this loopback test.)

Note The X.21 interface definition does not include a loopback definition. On the 4-Port Serial Interface SPA, the X.21 DTE interface does not support the loopback function. Because of the internal clock signal present on the 4-Port Serial Interface SPAs, loopback will function on an X.21 DCE interface.

Specifying the Interface Address on a SPA

SPA interface ports begin numbering with “0” from left to right. Single-port SPAs use only the port number 0. To configure or monitor SPA interfaces, you need to specify the physical location of the SIP, SPA, and interface in the CLI. The interface address format is *slot*/*subslot*/*port*, where:

- *slot*—Specifies the chassis slot number in the Cisco ASR 1000 Series Router where the SIP is installed.

- *subslot*—Specifies the secondary slot of the SIP where the SPA is installed.
- *port*—Specifies the number of the individual interface port on a SPA.

The following example shows how to specify the first interface (0) on a SPA installed in the first subslot of a SIP (0) installed in chassis slot 2:

```
Router(config)# interface serial 2/0/0
```

For more information about identifying slots and subslots, see the "Identifying Slots and Subslots for the SIPs and SPAs" section in the *Configuring the SIP* chapter.

Optional Configurations

There are several standard, but optional, configurations that might be necessary to complete the configuration of your serial SPA.

Configuring Timing Signals

All interfaces support both DTE and DCE mode, depending on the mode of the compact serial cable attached to the port. To use a port as a DTE interface, you need only connect a DTE compact serial cable to the port. When the system detects the DTE mode cable, it automatically uses the external timing signal. To use a port in DCE mode, you must connect a DCE compact serial cable and set the clock speed with the clock rate configuration command. You must also set the clock rate to perform a loopback test. This section describes how to set the clock rate on a DCE port and, if necessary, how to invert the clock to correct a phase shift between the data and clock signals.

Use the following commands when configuring timing signals:

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **interface serial slot/subslot/port**	Selects the interface to configure and enters interface configuration mode.
Router(config-if)# **invert txclock**	Inverts the transmit clock signal. When the EIA/TIA-232 interface is a DTE, the invert txclock command inverts the TxC signal the DTE receives from the remote DCE. When the EIA/TIA-232 interface is a DCE, the invert txclock command inverts the clock signal to the remote DTE port. The no form of this command changes the clock signal back to its original phase.
Router(config-if)# clock rate bps	Set standard clock rate, in bits per second: 1200, 2400, 4800, 9600, 19200, 38400, 56000, 64000, 72000, 125000, 148000, 250000, 500000, 800000, 1000000, 1300000, 2000000, 4000000, or 8000000. Any nonstandard clock rates that are entered are rounded off to the nearest hardware-supported clock rate. The actual clock rate is then displayed on the console. The no form of this command removes a clock rate that has been set.

Command	Purpose
Router(config-if)# **invert data**	Inverts the data signal. The no form of this command disables the inversion of the data signal.

Note Clock rates supported for EIA/TIA-232: 1.2K, 2.4K, 4.8K, 9.6K, 14.4K, 19.2K, 28.8K, 32K, 38.4K, 56K, 64K, 128K.

Note Clock rates supported for EIA-530, EIA-530A, EIA-449, V.35(bps): 1.2K, 2.4K, 4.8K, 9.6K, 14.4K, 19.2K, 28.8K, 32K, 38.4K, 56K, 64K, 72K, 115.2K, 2.048M, 2.341M, 2.731M, 3.277M, 4.09M, 5.461M, 8.064K.

Note Clock rates supported for X.21: 1.2K, 2.4K, 4.8K, 9.6K, 14.4K, 19.2K, 28.8K, 32K, 38.4K, 56K, 64K, 72K, 115.2K, 2.016M.

Inverting the Clock Signal

Systems that use long cables or cables that are not transmitting the TxC (clock) signal might experience high error rates when operating at higher transmission speeds. If a 4-Port Serial Interface SPA port is reporting a high number of error packets, a phase shift might be the problem; inverting the clock might correct this phase shift.

When the EIA/TIA-232 interface is a DTE, the invert-txclock command inverts the TxC signal the DTE receives from the remote DCE. When the EIA/TIA-232 interface is a DCE, the invert-txclock command inverts the clock signal to the remote DTE port. Use the no invert-txclock command to change the clock signal back to its original phase.

Use the following commands when inverting the clock signal:

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **interface serial slot/subslot/port**	Selects the interface to configure and enters interface configuration mode.
Router(config-if)# **invert txclock**	Inverts the transmit clock signal. When the EIA/TIA-232 interface is a DTE, the invert txclock command inverts the TxC signal the DTE receives from the remote DCE. When the EIA/TIA-232 interface is a DCE, the invert txclock command inverts the clock signal to the remote DTE port. The no form of this command changes the clock signal back to its original phase.

Command	Purpose
Router(config-if)# **invert data**	Invert the data signal. The no form of this command disables the inversion of the data stream.

Configuring NRZI Format

All EIA/TIA-232 interfaces on the 4-Port Serial Interface SPA support non-return-to-zero (NRZ) and non-return-to-zero inverted (NRZI) formats. Both formats use two different voltage levels for transmission. NRZ signals maintain constant voltage levels with no signal transitions—no return to a zero voltage level—during a bit interval and are decoded using absolute values: 0 and 1. NRZI uses the same constant signal levels but interprets the absence of data—a space—at the beginning of a bit interval as a signal transition and the presence of data—a mark—as no signal transition. NRZI uses relational encoding to interpret signals rather than determining absolute values.

Use the following commands when configuring NRZI format:

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **interface serial slot/subslot/port**	Selects the interface to configure and enters interface configuration mode.
Router(config-if)# **nrzi-encoding**	Enables NRZI encoding.
Router(config-if)# **no nrzi-encoding**	Disables NRZI encoding.

Configuring Cyclic Redundancy Checks

Cyclic redundancy checking (CRC) is an error-checking technique that uses a calculated numeric value to detect errors in transmitted data. All interfaces use a 16-bit CRC (CRC-CITT) by default but also support a 32-bit CRC. The designators 16 and 32 indicate the length (in bits) of the frame check sequence (FCS). The sender of a data frame calculates the frame check sequence (FCS). Before it sends a frame, the sender appends the FCS value to the message. The receiver recalculates the FCS and compares its calculation to the FCS from the sender. If there is a difference between the two calculations, the receiver assumes that a transmission error occurred and sends a request to the sender to resend the frame.

To set the length of the cyclic redundancy checks (CRC), use the following commands:

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **interface serial slot/subslot/port**	Selects the interface to configure and enters interface configuration mode.
Router(config-if)# **crc** {**16** | **32**}	Specifies the length of the CRC, where: • **16**—Specifies a 16-bit length CRC. This is the default. • **32**—Specifies a 32-bit length CRC. To set the CRC length to the default value, use the no form of this command.

Configuring Encapsulation

When traffic crosses a WAN link, the connection needs a Layer 2 protocol to encapsulate traffic. To set the encapsulation method, use the following commands:

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **interface serial** *slot /subslot /port*	Selects the interface to configure and enters interface configuration mode.
Router(config-if)# *encapsulation encapsulation-type* {*hdlc* | *ppp* | *frame-relay*}	Sets the encapsulation type on the interface. • hdlc—High-Level Data Link Control (HDLC) protocol for serial interface. This is the default. • ppp—Point-to-Point Protocol (PPP) (for serial interface). • frame-relay—Frame Relay (for serial interface).

Verifying Encapsulation

Use the show interfaces serial command to display the encapsulation method:

```
Router# show interfaces serial3/1/1
Serial3/1/1 is up, line protocol is down
  Hardware is SPA-4XT-SERIAL
  MTU 1500 bytes, BW 2016 Kbit, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation FRAME-RELAY, crc 16, loopback not set
  Keepalive set (10 sec)
  LMI enq sent  13698, LMI stat recvd 0, LMI upd recvd 0, DTE LMI down
  LMI enq recvd 0, LMI stat sent  0, LMI upd sent  0
  LMI DLCI 1023  LMI type is CISCO  frame relay DTE
  FR SVC disabled, LAPF state down
  Broadcast queue 0/64, broadcasts sent/dropped 0/0, interface broadcasts 0
  Last input never, output 00:00:05, output hang never
  Last clearing of show interface counters 1d14h
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 3
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     0 packets input, 0 bytes, 0 no buffer
     Received 0 broadcasts (0 IP multicasts)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     19344 packets output, 254168 bytes, 0 underruns
     0 output errors, 0 collisions, 2283 interface resets
     0 output buffer failures, 0 output buffers swapped out
     4566 carrier transitions
     RTS up, CTS up, DTR up, DCD up, DSR up
```

Saving the Configuration

To save your running configuration to nonvolatile random-access memory (NVRAM), use the following command in privileged EXEC configuration mode:

Command	Purpose
Router# **copy running-config startup-config**	Writes the new configuration to NVRAM.

For more information about managing your system image and configuration files, refer to the *Cisco IOS Configuration Fundamentals Configuration Guide* and the *Cisco IOS Configuration Fundamentals Command Reference* publications for your Cisco IOS software release.

Verifying the Interface Configuration

Besides using the **show running-configuration** command to display your Cisco ASR 1000 Series Routers configuration settings, you can use the **show interfaces serial** and the **show controllers serial** commands to get detailed information on a per-port basis for your 2-Port and 4-Port Channelized T3 SPA.

Verifying the Per-Port Interface Status

To find detailed interface information on a per-port basis for the 2-Port and 4-Port Channelized T3 SPA, use the **show interfaces serial** command.

The following example provides sample output for the serial interface:

```
Router# show interface serial4/0/0

Serial4/0/0 is down, line protocol is down
Hardware is SPA-4XT-SERIAL
  MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec,
     Reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation HDLC, crc 16, loopback not set
  Keepalive set (10 sec)
  Restart-Delay is 0 secs
  Last input never, output never, output hang never
  Last clearing of show interface counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     0 packets input, 0 bytes, 0 no buffer
     Received 0 broadcasts (0 IP multicast)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     0 packets output, 0 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets
     0 output buffer failures, 0 output buffers swapped out
     0 carrier transitions
       RTS down, CTS down, DTR down, DCD down, DSR down
```

To find detailed status and statistical information on a per-port basis for the 4-Port Serial Interface SPA, use the **show controllers serial** command:

```
Router# show controllers serial 2/0/0

Serial2/0/0 - (SPA-4XT-SERIAL) is down
 Encapsulation : HDLC
 Cable type: RS-232 DTE
 mtu 1500, max_buffer_size 1524, max_pak_size 1656 enc 132
```

```
loopback: Off,  crc: 16, invert_data: Off
nrzi: Off, idle char: Flag
tx_invert_clk: Off, ignore_dcd: Off
rx_clockrate: 0, rx_clock_threshold: 0
serial_restartdelay:60000,  serial_restartdelay_def:60000

    RTS up, CTS down, DTR up, DCD down, DSR down
```

Configuration Examples

This section includes the following configuration examples:

Inverting the Clock Signal Configuration Example

The following example configures the interface to invert the clock signal:

```
Router(config-if)# interface serial3/0/0
Router(config-if)# invert txclock ?
  <cr>
Router(config-if)# invert txclock
Router(config-if)# invert ?
  data     Invert data stream
  txclock  Invert transmit clock
Router(config-if)# invert data
```

NRZI Format Configuration Example

The following example configures the interface for NRZI format:

```
Router(config-if)# nrzi-encoding ?
  <cr>
```

Cyclic Redundancy Checks Configuration Example

The following example configures the interface for 32-bit CRC error-checking:

```
Router(config-if)# crc ?
  16  crc word-size
  32  crc word-size
Router(config-if)# crc 32
```

Encapsulation Configuration Example

The following example configures Point-to-Point Protocol encapsulation method on the Serial SPA:

```
Router(config-if)# encapsulation ppp
```

Distributed Multilink PPP Configuration Example

The following example configures multilink Point-to-Point Protocol encapsulation method for port 0 of the Serial SPA in subslot 2 of the SIP in slot 1 of the router and sets the IP address and subnet mask:

```
Router(config)# interface multilink1
Router(config-if)# ip addr 10.0.0.1 255.255.255.0
Router(config)# interface serial3/2/0
Router(config-if)# encapsulation ppp
Router(config-if)# ppp chap hostname X1
Router(config-if)# ppp multilink gr 1
```