Configuring Interfaces

This chapter describes the basic interface configuration for the ML-Series card to help you get your ML-Series card up and running. For more information about the Cisco IOS commands used in this chapter, refer to the *Cisco IOS Command Reference* publication.

This chapter contains the following major sections:

Note Complete the initial configuration of your ML-Series card before proceeding with configuring interfaces.

Interface Configuration

The main function of the ML-Series card is to relay packets from one data link to another. Consequently, you must configure the characteristics of the interfaces, which receive and send packets. Interface characteristics include, but are not limited to, IP address, address of the port, data encapsulation method, and media type.

Many features are enabled on a per-interface basis. Interface configuration mode contains commands that modify the interface operation (for example, of an Ethernet port). When you enter the **interface** command, you must specify the interface type and number.

The following general guidelines apply to all physical and virtual interface configuration processes:

- All interfaces have a name which is comprised of an interface type (word) and a Port ID (number). For example, FastEthernet 2.
- Configure each interface with a bridge-group or IP address and IP subnet mask.
- VLANs are supported through the use of subinterfaces. The subinterface is a logical interface configured separately from the associated physical interface.
- Each physical interface, and the internal Packet-over-SONET/SDH (POS) interfaces, have an assigned MAC address.

MAC Addresses

Every port or device that connects to an Ethernet network needs a MAC address. Other devices in the network use MAC addresses to locate specific ports in the network and to create and update routing tables and data structures.

To find MAC addresses for a device, use the **show interfaces** command, as follows:

```
Router# sh interfaces fastEthernet 0
FastEthernet0 is up, line protocol is up
  Hardware is epif_port, address is 0005.9a39.6634 (bia 0005.9a39.6634)
  MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  Full-duplex, Auto Speed, 100BaseTX
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input 00:00:01, output 00:00:18, 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
     11 packets input, 704 bytes
     Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
     0 watchdog, 11 multicast
     0 input packets with dribble condition detected
     3 packets output, 1056 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets
     0 babbles, 0 late collision, 0 deferred
     0 lost carrier, 0 no carrier
     0 output buffer failures, 0 output buffers swapped out
```

Interface Port ID

The interface port ID designates the physical location of the interface within the ML-Series card. It is the name that you use to identify the interface you are configuring. The system software uses interface port IDs to control activity within the ML-Series card and to display status information. Interface port IDs are not used by other devices in the network; they are specific to the individual ML-Series card and its internal components and software.

The ML100T-12 port IDs for the 12 Fast Ethernet interfaces are Fast Ethernet 0 through 11. The ML1000-2 port IDs for the two Gigabit Ethernet interfaces are Gigabit Ethernet 0 and 1. Both ML-Series cards feature two POS ports, and the ML-Series port IDs for the two POS interfaces are POS 0 and 1. You can use user-defined abbreviations such as f0 through f11 to configure the 12 Fast Ethernet interfaces, gi0 or gi1 to configure the two Gigabit Ethernet interfaces, and POS0 and POS1 to configure the two POS ports.

You can use Cisco IOS **show** commands to display information about any or all the interfaces of the ML-Series card.

Caution Do not use the g0 or g1 for a Gigabit Ethernet user-defined abbreviation. This will create an unsupported group asynchronous interface.

Instructions for Configuring Interfaces

The following general configuration instructions apply to all interfaces. Before you configure interfaces, develop a plan for a bridge or routed network.

To configure an interface, do the following:

Note Router or Switch is used as a generic prompt in documentation. Your specific prompt will vary.

Step 1 Enter the **configure** EXEC command at the privileged EXEC prompt to enter global configuration mode.

```
Router> enable
Password:
Router# configure terminal
Router(config)#
```

Step 2 Enter the **interface** command, followed by the interface type (for example, fastethernet, gigabitethernet, or pos), and its interface port ID (see the "Interface Port ID" section on page 4-2).

For example, to configure a Gigabit Ethernet port, enter this command:

```
Router(config)# interface gigabit-ethernet-number
```

Step 3 Follow each **interface** command with the interface configuration commands required for your particular interface.

The commands you enter define the protocols and applications that will run on the interface. The ML-Series card collects and applies commands to the **interface** command until you enter another **interface** command or a command that is not an interface configuration command. You can also enter **end** to return to privileged EXEC mode.

Step 4 Check the status of the configured interface by entering the EXEC **show interface** command.

```
Router# sh interface fastEthernet 0
FastEthernet0 is up, line protocol is up
 Hardware is epif_port, address is 0005.9a39.6634 (bia 0005.9a39.6634)
 MTU 1500 bytes, BW 100000 Bit, DLY 100 use,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  Full-duplex, Auto Speed, 100BaseTX
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input 00:00:01, output 00:00:18, 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
     11 packets input, 704 bytes
     Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
     0 watchdog, 11 multicast
     0 input packets with dribble condition detected
     3 packets output, 1056 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets
     0 babbles, 0 late collision, 0 deferred
     0 lost carrier, 0 no carrier
     0 output buffer failures, 0 output buffers swapped out
```

Understanding Interfaces

ML-Series cards support Fast Ethernet, Gigabit Ethernet and POS interfaces. This section provides some examples of configurations for all interface types.

To configure an IP address or bridge-group number on a Fast Ethernet, Gigabit Ethernet, or POS interface, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	`Router(config)# interface type number`	Activates interface configuration mode to configure either the Gigabit Ethernet interface, the Fast Ethernet interface or the POS interface.
Step 2	`Router(config-if)# {ip address ip-address subnet-mask | bridge-group bridge-group-number}`	Sets the IP address and IP subnet mask to be assigned to the interface. or Assigns a network interface to a bridge group.
Step 3	`Router(config-if)# no shutdown`	Enables the interface by preventing it from shutting down.
Step 4	`Router(config)# end`	Returns to privileged EXEC mode.
Step 5	`Router# copy running-config startup-config`	(Optional) Saves configuration changes to timing and control card (TCC+/TCC2) Flash database.

Note Repeat Steps 1 through 3 to configure the other interfaces on the ML-Series card.

Configuring the Fast Ethernet Interfaces (ML100T-12)

To configure the IP address or bridge-group number, autonegotiation, and flow control on a Fast Ethernet interface, perform the following procedure, beginning in global configuration mode:

	Command	Purpose
Step 1	`Router(config)# interface fastethernet number`	Activates interface configuration mode to configure the Fast Ethernet interface.
Step 2	`Router(config-if)# {ip address ip-address subnet-mask | bridge-group bridge-group-number}`	Sets the IP address and IP subnet mask to be assigned to the interface. or Assigns a network interface to a bridge group.

	Command	Purpose
Step 3	`Router(config-if)# {no} speed [10 | 100 | auto}`	Configures the transmission speed for 10 or 100 Mbps. If you set the speed or duplex for **auto**, you enable autonegotiation on the system—the ML-Series card matches the speed and duplex mode of the partner node.
Step 4	`Router(config-if)# [no] duplex {full | half | auto}`	for full duplex, half duplex, or autonegotiate.
Step 5	`Router(config-if)# flowcontrol send {on | off | desired}`	(Optional) Sets the send flow control value for an interface. Flow control works only with port-level policing.
Step 6	`Router(config-if)# no shutdown`	Enables the interface by preventing it from shutting down.
Step 7	`Router(config)# end`	Returns to privileged EXEC mode.
Step 8	`Router# copy running-config startup-config`	(Optional) Saves your configuration changes to TCC+/TCC2 Flash database.

The following example shows how to do the initial configuration of a Fast Ethernet interface with an IP address, autonegotiated speed, and autonegotiated duplex:

Example 4-1 Initial Configuration of a Fast Ethernet Interface

```
Router(config)# interface fastethernet 1
Router(config-if)# ip address 10.1.2.4 255.0.0.0
Router(config-if)# speed auto
Router(config-if)# duplex auto
Router(config-if)# no shutdown
Router(config-if)# end
Router# copy running-config startup-config
```

Configuring the Gigabit Ethernet Interface (ML1000-2)

To configure IP address or bridge-group number, autonegotiation, and flow control on a Gigabit Ethernet interface, perform the following procedure, beginning in global configuration mode:

Note The default setting for the negotiation mode is **auto** for the Gigabit Ethernet and Fast Ethernet interfaces. The Gigabit Ethernet port always operates at 1000 Mbps in full-duplex mode.

	Command	Purpose
Step 1	`Router# interface gigabitethernet` *number*	Activates interface configuration mode to configure the Gigabit Ethernet interface.
Step 2	`Router#(config-if)# {ip address` *ip-address subnet-mask* `| bridge-group` *bridge-group-number}*	Sets the IP address and subnet mask. or Assigns a network interface to a bridge group.

	Command	Purpose
Step 3	`Router#(config-if)# [no] negotiation auto`	Sets negotiation mode to **auto**. The Gigabit Ethernet port attempts to negotiate the link with the partner port. If you want the port to force the link up no matter what the partner port setting is, set the Gigabit Ethernet interface to **no negotiation auto**.
Step 4	`Router(config-if)# flowcontrol {send | receive} {on | off | desired}`	(Optional) Sets the send or receive flow control value for an interface. Flow control works only with port-level policing.
Step 5	`Router#(config-if)# no shutdown`	Enables the interface by preventing it from shutting down.
Step 6	`Router#(config)# end`	Returns to privileged EXEC mode.
Step 7	`Router# copy running-config startup-config`	(Optional) Saves configuration changes to TCC+/TCC2 Flash database.

Note Repeat Steps 1 to 4 to configure the other Gigabit Ethernet interfaces.

The following example shows how to do an initial configuration of a Gigabit Ethernet interface with autonegotiation and an IP address:

Example 4-2 Initial Configuration of a Gigabit Ethernet Interface

```
Router(config)# interface gigabitethernet 0
Router(config-if)# ip address 10.1.2.3 255.0.0.0
Router(config-if)# negotiation auto
Router(config-if)# no shutdown
Router(config-if)# end
Router# copy running-config startup-config
```

Monitoring Operations on the Fast Ethernet and Gigabit Ethernet Interfaces

To verify the settings after you have configured Fast Ethernet interfaces, enter the **show interface** command.

The following output from the **show interface** command displays the status of the Fast Ethernet interface including port speed and duplex operation:

Example 4-3 show interface command output

```
Router# show interface fastEthernet 0
FastEthernet0 is up, line protocol is up
  Hardware is epif_port, address is 0005.9a39.6634 (bia 0005.9a39.6634)
  MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  Full-duplex, 100Mb/s, 100BaseTX
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input never, output 00:00:23, 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
     Received 0 broadcasts (0 IP multicast)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
     0 watchdog, 0 multicast
     0 input packets with dribble condition detected
     4 packets output, 1488 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets
     0 babbles, 0 late collision, 0 deferred
     0 lost carrier, 0 no carrier
     0 output buffer failures, 0 output buffers swapped out
```

Enter the **show controller** command to display information about the Fast Ethernet controller chip.

The following output from the **show controller** command shows statistics, including information about initialization block information, transmit ring, receive ring, and errors:

Example 4-4 show controller command output

```
Router#show controller fastEthernet 0
IF Name: FastEthernet0
Port Status DOWN
Send Flow Control    : Disabled
Receive Flow Control : Enabled
MAC registers
CMCR : 0x0000042D (Tx Enabled, Rx Disabled)
CMPR : 0x150B0A80 (Long Frame Disabled)
FCR  : 0x0000A00B (Rx Pause detection Enabled)
MII registers:
Control Register              (0x0): 0x4000 (Auto negotiation disabled)
Status Register               (0x1): 0x7809 (Link status Down)
PHY Identification Register 1 (0x2): 0x40
PHY Identification Register 2 (0x3): 0x61D4
Auto Neg. Advertisement Reg   (0x4): 0x1E1  (Speed 100, Duplex Full)
Auto Neg. Partner Ability Reg (0x5): 0x0    (Speed 10, Duplex Half)
Auto Neg. Expansion Register  (0x6): 0x4
100Base-X Aux Control Reg    (0x10): 0x2000
100Base-X Aux Status Register(0x11): 0x0
100Base-X Rcv Error Counter  (0x12): 0x0
100Base-X False Carr. Counter(0x13): 0x0
```

Enter the **show run interfaces fastEthernet 0** command to display information about the configuration of the Fast Ethernet interface. The command is useful when there are multiple interfaces and you want to look at the configuration of a specific interface.

The following output from the **show controller** command includes the IP or lack of IP address and the state of the interface:

Example 4-5 show controller command output

```
daytona#show run interface fastEthernet 0
Building configuration...

Current configuration : 56 bytes
!
interface FastEthernet0
```

```
no ip address
shutdown

end
```

POS on the ML-Series Card

Packet over SONET/SDH (POS) is a high-speed method of transporting IP traffic between two points. This technology combines the Point-to-Point Protocol (PPP) with SONET and Synchronous Digital Hierarchy (SDH) interfaces. SONET is an octet-synchronous multiplex scheme defined by the American National Standards Institute (ANSI) standard (T1.105.1988) for optical digital transmission, and SDH is the European Telecommunications Standards Institute (ETSI) equivalent.

ML-Series SONET/SDH Transmission Rates

SONET transmission rates are integral multiples of 51.840 Mbps. Table 4-1 shows supported transmission multiples.

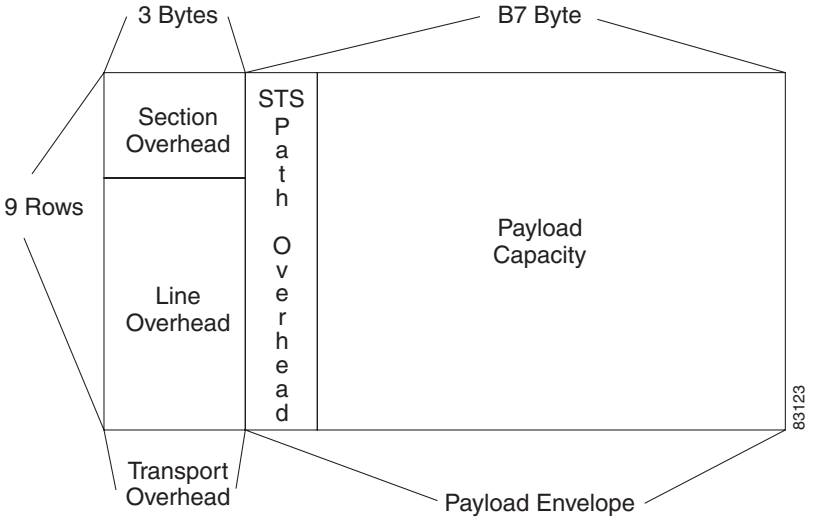
Table 4-1 Transmission Multiples Supported by ML-Series Cards

Topology	Supported Sizes
Circuits terminated by two ML-Series cards	STS-1, STS-3c, STS-6c, STS-9c, STS-12c, and STS-24c (SONET) or VC4, VC4-2c, VC4-3c, VC4-4c, and VC4-8c (SDH)
Circuits terminated by G-Series card and ML-Series card	STS-1, STS-3c, STS-6c, STS-9c, STS-12c (SONET) or VC4, VC4-2c, VC4-3c, VC4-4c, and VC4-8c (SDH)
Circuits terminated by ML-Series card and External POS device	STS-3c and STS-12c (SONET) or VC4 and VC4-4c (SDH)

SONET Frame Fundamentals

SONET is a Layer 1 protocol that uses a layered architecture. Figure 4-1 shows SONET's three layers: section, line, and path. The section overhead (SOH) and line overhead (LOH) form the transport overhead (TOH), while the path overhead (POH) and actual payload (referred to as payload capacity) form the synchronous payload envelope (SPE). Each layer adds a number of overhead bytes to the SONET frame.

Figure 4-1 Three SONET Layers



C2 Byte

One of the overhead bytes in the SONET frame is the C2 Byte. The SONET standard defines the C2 byte as the path signal label. The purpose of this byte is to communicate the payload type being encapsulated by the SONET framing overhead (FOH). The C2 byte functions similarly to EtherType and Logical Link Control (LLC)/Subnetwork Access Protocol (SNAP) header fields on an Ethernet network; it allows a single interface to transport multiple payload types simultaneously. Table 4-2 provides C2 byte hex values.

Table 4-2 C2 Byte Common Values

Hex Value	SONET Payload Contents
00	Unequipped
01	Equipped non specific payload
02	Virtual Tributaries (VTs) inside (default)
03	VTs in locked mode (no longer supported)
04	Asynchronous DS3 mapping
12	Asynchronous DS-4NA mapping
13	Asynchronous Transfer Mode (ATM) cell mapping
14	Distributed Queue Dual Bus (DQDB) protocol cell mapping
15	Asynchronous Fiber Distributed Data Interface (FDDI) mapping
16	IP inside PPP with scrambling
CF	IP inside PPP without scrambling
FE	Test signal mapping (see ITU-T G.707)

C2 Byte and Scrambling

As listed in Table 4-2, POS interfaces use a value of 0x16 or 0xCF in the C2 byte depending on whether ATM-style scrambling is enabled or not. RFC 2615, which defines PPP over SONET, mandates the use of these values based on the scrambling setting. The RFC defines the C2 byte values as follows: "the value of 22 (16 hex) is used to indicate PPP with X^43+ 1 scrambling [4]. For compatibility with RFC 1619 (STS-3c-SPE/VC-4 only), if scrambling has been configured to be off, then the value 207 (CF hex) is used for the Path Signal Label to indicate PPP without scrambling."

In other words:

- If scrambling is enabled, POS interfaces use a C2 value of 0x16 (PPP and HDLC encapsulation).
- If scrambling is disabled, POS interfaces use a C2 value of 0xCF (PPP and HDLC encapsulation).
- LEX encapsulation uses a C2 value of 0x01 regardless of the scrambling setting.

Most POS interfaces that use a default C2 value of 0x16 (22 decimal) insert the **pos flag c2 22** command in the configuration, although this line does not appear in the running configuration since it is the default. Use the **pos flag c2** command to change the value from its default, as follows:

Example 4-6 pos fag c2 command

```
Router(config-if)# pos flag c2 ?
  <0-255>  byte value, default 0x16
```

Note Changing the C2 value from the default value does not affect POS scrambling settings.

Use the **show run** command to confirm your change. The **show controller pos** command outputs the receive and transmit values and the C2 value. Thus, changing the value on the local end will not change the value in the **show controller** command output.

Example 4-7 show controller pos command

```
Router# sh controllers pos 0
Interface POS0
Hardware is Packet/Ethernet over Sonet
PATH
  PAIS      = 0         PLOP      = 0         PRDI     = 0         PTIM = 0
  PPLM      = 0         PUNEQ     = 0         PPDI     = 0
  BER_SF_B3 = 0         BER_SD_B3 = 0         BIP(B3)  = 14         REI = 155
  NEWPTR    = 0         PSE       = 0         NSE      = 0

Active Alarms : None
Demoted Alarms: None
Active Defects: None
Alarms reportable to TCC/CLI: PAIS PRDI PLOP PUNEQ PPLM PTIM PPDI BER_SF_B3 BER_
SD_B3
Link state change defects: PAIS PLOP PRDI PPDI BER_SF_B3
Link state change time   : 200 (msec)

DOS FPGA channel number: 0
Starting STS (0 based) : 0
Circuit size           : STS-24c
RDI Mode               : 1 bit
C2 (tx / rx)           : 0x01 / 0x01
Framing                : SONET

Path Trace
```

```
 Mode            : off
 Buffer          : Unstable
 Remote hostname :
 Remote interface:
 Remote IP addr  :

B3 BER thresholds:
SFBER = 1e-5,   SDBER = 1e-7


     1106 total input packets,  80059 post-HDLC bytes
     0 input short packets,  80714 pre-HDLC bytes
     0  input long packets , 205  input runt packets
     17  input CRCerror packets , 0  input drop packets
     0 input abort packets
     1107 input packets dropped by ucode


     0  total output packets, 0  output pre-HDLC bytes
     0 output post-HDLC bytes

  Carrier delay is 200 msec
```

The **show interface pos0** command shows scrambling.

```
daytona#show interface pos0
POS0 is up, line protocol is up
Hardware is Packet/Ethernet over Sonet, address is 0005.9a3b.bf90 (bia 0005.9a3b.bf90)
MTU 1500 bytes, BW 1244160 Kbit, DLY 100 usec,
    reliability 243/255, txload 1/255, rxload 166/255
    Encapsulation ONS15454-G1000, crc 32, loopback not set
    Keepalive set (10 sec)
    Scramble enabled
    ARP type: ARPA, ARP Timeout 04:00:00
    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, 2385314109 bytes
        Received 0 broadcasts (0 IP multicast)
        0 runts, 0 giants, 0 throttles
            0 parity
        2839625 input errors, 2839625 CRC, 0 frame, 0 overrun, 0 ignored
        0 input packets with dribble condition detected
        9 packets output, 3393 bytes, 0 underruns
        0 output errors, 0 applique, 0 interface resets
        0 babbles, 0 late collision, 0 deferred
        0 lost carrier, 0 no carrier
        0 output buffer failures, 0 output buffers swapped out
        0 carrier transitions
```

Third-Party POS Interfaces

If a Cisco POS interface fails to come up when connected to a third-party device, confirm the scrambling and cyclic redundancy check (CRC) settings as well as the advertised value in the C2 byte. On routers from Juniper Networks, configuring RFC 2615 mode sets the following three parameters:

- Scrambling enabled

- C2 value of 0x16
- CRC-32

Previously, when scrambling was enabled, these third-party devices continued to use a C2 value of 0xCF, which did not properly reflect the scrambled payload.

Configuring the ML-Series POS Interfaces

To configure the POS interface, perform the following procedure, beginning in global configuration mode. Encapsulation changes on POS ports are allowed only when the interface is in a manual shutdown (ADMIN_DOWN):

	Command	Purpose
Step 1	`Router(config)# interface pos` *number*	Activates interface configuration mode to configure the POS interface. The POS interface is created upon the creation of a SONET/SDH circuit.
Step 2	`Router#(config-if)# {ip address` *ip-address subnet-mask* `| bridge-group` *bridge-group-number}*	Sets the IP address and subnet mask. or Assigns a network interface to a bridge group.
Step 3	`Router#(config-if)#shutdown`	Manually shuts down the interface. Encapsulation changes on POS ports are allowed only when the interface is shutdown (ADMIN_DOWN).
Step 4	`Router#(config-if)# encapsulation` *type*	Sets the encapsulation type. Valid values are: • **hdlc**—Cisco HDLC • **lex**—LAN extension, special encapsulation for use with Cisco ONS G-Series Ethernet line cards • **ppp**—Point-to-Point Protocol
Step 5	`Router#(config-if)# pos flag c2` *byte value*	(Optional) Sets the C2 byte value. Valid choices are 0 to 255 (decimal). The default value is 0x01 (hex) for LEX.
Step 6	`Router#(config-if)#no shutdown`	Restarts the shutdown interface.
Step 7	`Router#(config)# end`	Returns to privileged EXEC mode.
Step 8	`Router# copy running-config startup-config`	(Optional) Saves configuration changes to NVRAM.

Note The POS interface is not present until a SONET STS or SDH STM circuit is created.

Monitoring Operations on the POS Interface and POS Controller

The following output from the **show interface** command displays the POS interface's status and global parameters:

Example 4-8 show interface command

```
Router# show interface pos 0
POS0 is up, line protocol is up
  Hardware is Packet/Ethernet over Sonet, address is 0005.9a39.6630 (bia 0005.9a
39.6630)
  MTU 1500 bytes, BW 311040 Kbit, DLY 100 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ONS15454-G1000, crc 32, loopback not set
  Keepalive set (10 sec)
  Scramble enabled
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input 00:02:34, 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
     1107 packets input, 11267427 bytes
     Received 0 broadcasts (0 IP multicast)
     0 runs, 0 giants, 0 throttles
              0 parity
     1 input errors, 1 CRC, 0 frame, 0 overrun, 0 ignored
     0 input packets with dribble condition detected
     0 packets output, 0 bytes, 0 underruns
     0 output errors, 0 applique, 0 interface resets
     0 babbles, 0 late collision, 0 deferred
     0 lost carrier, 0 no carrier
     0 output buffer failures, 0 output buffers swapped out
     0 carrier transitions
```

The following output from the **show controllers** command displays the POS controllers:

Example 4-9 show controllers command

```
Router# show controllers pos 0
Interface POS0
Hardware is Packet/Ethernet over Sonet
PATH
  PAIS      = 1         PLOP      = 0         PRDI    = 0         PTIM = 0
  PPLM      = 0         PUNEQ     = 0         PPDI    = 0
  BER_SF_B3 = 0         BER_SD_B3 = 0         BIP(B3) = 2975       REI = 7
  NEWPTR    = 1         PSE       = 0         NSE     = 0

Active Alarms : None
Demoted Alarms: None
Active Defects: None
Alarms reportable to CLI: PAIS PRDI PLOP PUNEQ PPLM PTIM PPDI BER_SF_B3 BER_
3
Link state change defects: PAIS PLOP PRDI PPDI BER_SF_B3
Link state change time   : 200 (msec)

DOS FPGA channel number: 0
Starting STS (0 based) : 0
Circuit size           : STS-6c
```

```
RDI Mode                : 1 bit
C2 (tx / rx)            : 0x01 / 0x01
Framing                 : SONET

Path Trace
 Mode             : off
 Buffer           : Unstable
 Remote hostname :
 Remote interface:
 Remote IP addr  :

B3 BER thresholds:
SFBER = 1e-5,   SDBER = 1e-7


     1107 total input packets,  11267259 post-HDLC bytes
     0 input short packets,  11267427 pre-HDLC bytes
     0  input long packets , 0  input runt packets
     1  input CRCerror packets , 0  input drop packets
     0 input abort packets
     945 input packets dropped by ucode


     0  total output packets, 0  output pre-HDLC bytes
     0 output post-HDLC bytes

  Carrier delay is 200 msec
```

Additional Configurations

To configure additional properties to match those of the interface at the far end, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	`Router(config-if)# no keepalive`	Turns off keep alive messages. Keep alive messages, though not required, are recommended.
Step 2	`Router(config-if)# crc {16 | 32}`	Sets the CRC value. If the device to which the POS module is connected does not support the default CRC value of 32, set both devices to use a value of 16.

Setting the MTU Size

To set the maximum transmission unit (MTU) size, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	`Router(config)# interface pos` *number*	Enters interface configuration mode and specifies the POS interface to configure.
Step 2	`Router(config-if)# mtu` *bytes*	Configures the MTU size up to a maximum of 9000 bytes. See Table 4-3 on page 4-15.

Table 4-3 shows the default MTU sizes.

Table 4-3 Default MTU Size

Encapsulation Type	Default Size
LEX (default)	1500
HDLC	4470
PPP	4470

Configuring Framing

No Cisco IOS configuration is necessary. Framing type is determined during circuit configuration.

Configuring POS SPE Scrambling

To configure POS SPE scrambling, perform the following steps, beginning in global configuration mode:

	Command	Purpose
Step 1	`Router(config)# interface pos` *number*	Enters interface configuration mode and specifies the POS interface to configure.
Step 2	`Router(config-if)# no pos scramble-spe`	Disables payload scrambling on the interface. Payload scrambling is on by default.
Step 3	`Router(config-if)# no shutdown`	Enables the interface with the previous configuration.

SONET/SDH Alarms

The ML-Series cards report SONET/SDH alarms under both Cisco IOS and CTC/TL1. A number of path alarms are reported in the Cisco IOS console. Configuring Cisco IOS console alarm reporting has no effect on CTC alarm reporting. The "Configuring SONET/SDH Alarms" procedure specifies the alarms reported to the Cisco IOS console.

CTC/TL1 has sophisticated SONET/SDH alarm reporting capabilities. As a card in the ONS node, the ML-Series card reports alarms to CTC/TL-1 like any other ONS card. On the ONS 15454 SONET, the ML-Series card reports Telcordia GR-253 SONET alarms in the Alarms panel of CTC. For more information on alarms and alarm definitions, refer to the "Alarm Troubleshooting" chapter of the *Cisco ONS 15454 Troubleshooting Guide*, or the *Cisco ONS 15454 SDH Troubleshooting Guide*.

Configuring SONET/SDH Alarms

All SONET/SDH alarms are logged on the Cisco IOS CLI by default. But to provision or disable the reporting of specific SONET/SDH alarms on the Cisco IOS CLI, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	`Router(config)# interface pos` *number*	Enters interface configuration mode and specifies the POS interface to configure.
Step 2	`Router(config-if)# pos report {all | encap | pais | plop | ppdi | pplm | prdi | ptim | puneq | sd-ber-b3 | sf-ber-b3}`	Permits logging of selected SONET/SDH alarms. Use the **no** form of the command to disable reporting of a specific alarm. The alarms are as follows: • **all**—All alarms/signals • **encap**—Path encapsulation mismatch • **pais**—Path alarm indication signal • **plop**—Path loss of pointer • **ppdi**—Path payload defect indication • **pplm**—Payload label, C2 mismatch • **prdi**—Path remote defect indication • **ptim**—Path trace identifier mismatch • **puneq**—Path label equivalent to zero • **sd-ber-b3**—PBIP BER in excess of SD threshold • **sf-ber-b3**—PBIP BER in excess of SF threshold
Step 3	`Router(config-if)# end`	Returns to the privileged EXEC mode.
Step 4	`Router# copy running-config startup-config`	(Optional) Saves configuration changes to NVRAM.

To determine which alarms are reported on the POS interface and to display the bit error rate (BER) thresholds, use the **show controllers pos** command.

Note Cisco IOS alarm reporting commands apply only to the Cisco IOS CLI. SONET/SDH alarms reported to the CTC are not affected.

To configure path alarms as triggers and specify a delay, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	`Router(config)# interface pos` *number*	Enters interface configuration mode and specifies the POS interface to configure.
Step 2	`Router(config-if)# pos trigger defect {all | ber_sf_b3 | encap | pais | plop | ppdi | pplm | prdi | ptim | puneq}`	Configures certain path defects as triggers to bring down the POS interface. The configurable triggers are as follows: • **all**—All link down alarm failures • **ber_sd_b3**—PBIP BER in excess of SD threshold failure • **ber_sf_b3**—PBIP BER in excess of SD threshold failure (default) • **encap**—Path Signal Label Encapsulation Mismatch failure (default) • **pais**—Path Alarm Indication Signal failure (default) • **plop**—Path Loss of Pointer failure (default) • **ppdi**—Path Payload Defect Indication failure (default) • **pplm**—Payload label mismatch path (default) • **prdi**—Path Remote Defect Indication failure (default) • **ptim**—Path Trace Indicator Mismatch failure (default) • **puneq**—Path Label Equivalent to Zero failure (default)
Step 3	`Router(config-if)# pos trigger delay` *millisecond*	Sets waiting period before the line protocol of the interface goes down. Delay can be set from 200 to 2000 ms. If no time intervals are specified, the default delay is set to 200 ms.
Step 4	`Router(config-if)# end`	Returns to the privileged EXEC mode.
Step 5	`Router# copy running-config startup-config`	(Optional) Saves configuration changes to NVRAM.

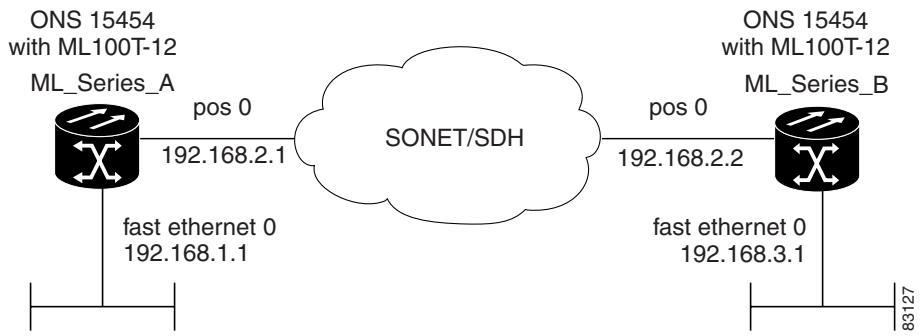
Common ML-Series POS Configurations

The following sections describe common ML-Series card POS configurations.

ML-Series Card to ML-Series Card

Figure 4-2 illustrates a POS configuration between two ML-Series cards.

Figure 4-2 ML-Series Card to ML-Series Card POS Configuration



Example 4-10 Router A Configuration

```
hostname Router_A
!
interface FastEthernet0
 ip address 192.168.1.1 255.255.255.0
!
interface POS0
 ip address 192.168.2.1 255.255.255.0
 crc 32
pos flag c2 1
!
router ospf 1
 log-adjacency-changes
 network 192.168.1.0 0.0.0.255 area 0
 network 192.168.2.0 0.0.0.255 area 0
```

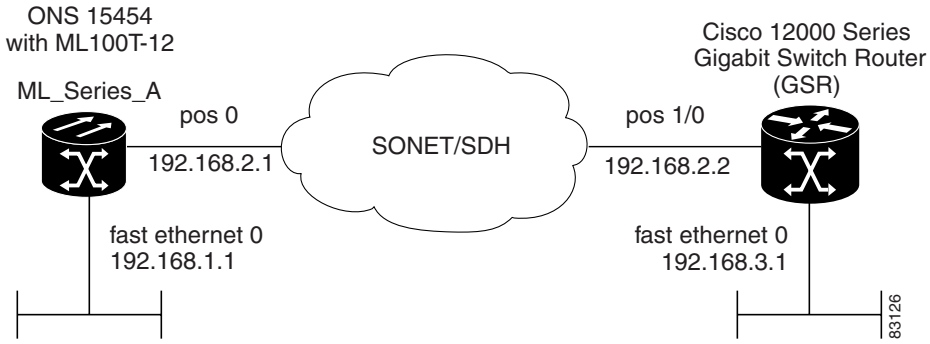
Example 4-11 Router B Configuration

```
hostname Router_B
!
interface FastEthernet0
 ip address 192.168.3.1 255.255.255.0
!
interface POS0
 ip address 192.168.2.2 255.255.255.0
 crc 32
pos flag c2 1
!
router ospf 1
 log-adjacency-changes
 network 192.168.2.0 0.0.0.255 area 0
 network 192.168.3.0 0.0.0.255 area 0
!
```

ML-Series Card to Cisco 12000 GSR-Series Router

Figure 4-3 illustrates a POS configuration between an ML-Series card and a Cisco 12000 GSR-Series router.

Figure 4-3 ML-Series Card to Cisco 12000 Series Gigabit Switch Router (GSR) POS Configuration



Example 4-12 Router A Configuration

```
hostname Router_A
!
interface FastEthernet0
 ip address 192.168.1.1 255.255.255.0
!
!
interface POS0
 ip address 192.168.2.1 255.255.255.0
 encapsulation ppp
 crc 32
!
router ospf 1
 log-adjacency-changes
 network 192.168.1.0 0.0.0.255 area 0
 network 192.168.2.0 0.0.0.255 area 0
```

Example 4-13 GSR-12000 Configuration

```
hostname GSR
!
interface FastEthernet1/0
 ip address 192.168.3.1 255.255.255.0
!
interface POS2/0
 ip address 192.168.2.2 255.255.255.0
 crc 32
encapsulation PPP
pos scramble-atm
!
router ospf 1
 log-adjacency-changes
 network 192.168.2.0 0.0.0.255 area 0
 network 192.168.3.0 0.0.0.255 area 0
!
```

Note The default encapsulation for the ML-Series card is LEX and the corresponding default MTU is 1500 bytes. When connecting to an external POS device, it is important to ensure that both the ML-Series switch and the external device uses the same configuration for the parameters listed in Table 4-4.

Table 4-4 ML-Series Parameter Configuration for Connection to a Cisco 12000 GSR-Series Router

Command	Parameter

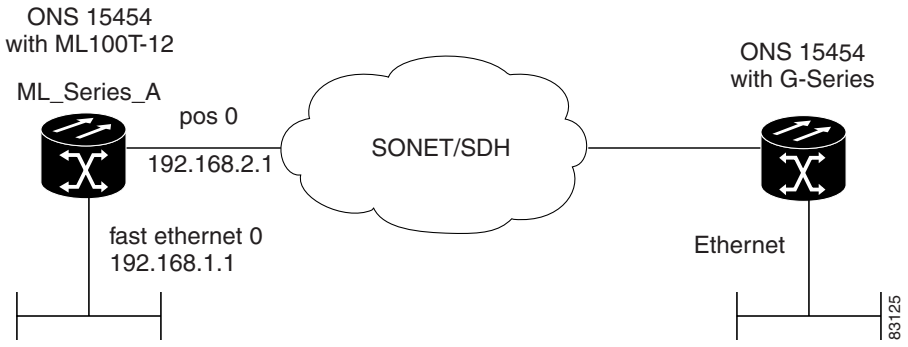
`Router(config-if)# encapsulation ppp`
or
`Router(config-if)# encapsulation hdlc`
	Encapsulation—Default encapsulation is HDLC on GSR. Default encapsulation on ML-Series card is LEX.

`Router(config-if)# show controller pos`	C2 Byte—Use the **show controller pos** command to verify that the transmit and receive C2 values are the same.
`Router(config-if)# pos flag c2 value`	Sets the C2 byte value. Valid choices are 0 to 255 (decimal). The default value is 0x01 (hex) for LEX.

ML-Series Card to G-Series Card

Figure 4-4 illustrates a POS configuration between an ML-Series card and a G-Series card.

Figure 4-4 ML-Series Card to G-Series Card POS Configuration



Example 4-14 Router A Configuration

```
hostname Router_A
!
interface FastEthernet0
 ip address 192.168.1.1 255.255.255.0
!
interface POS0
 ip address 192.168.2.1 255.255.255.0
 crc 32
!
router ospf 1
 log-adjacency-changes
 network 192.168.1.0 0.0.0.255 area 0
 network 192.168.2.0 0.0.0.255 area 0
```