CHAPTER 4

Provisioning AXSM Communication Links

This chapter describes how to add ATM ports and connections to the physical lines defined in Chapter 3, "Preparing AXSM Cards and Lines for Communication." This chapter explains how to provision the following types of connections on the MGX 8850 switch:

- Switched Virtual Circuit (SVC)
- Soft Permanent Virtual Circuit (SPVC)
- Interim Inter-switch Protocol (IISP)
- Virtual trunk
- MGX 8850 Release 1 feeder

The configuration differences between these different types of connections are often as simple as an additional command or a different set of command options. To eliminate redundancy and help experienced users complete configuration procedures quickly, this chapter uses configuration quickstarts and task descriptions to explain how to configure these connections.

The first time you configure a connection type, use the quickstart procedure to see the order of tasks to complete, and then look to the task descriptions for detailed instructions on how to complete the tasks. As you get more experience configuring connections, you can look up fewer tasks.

Tips Remember that you can get information on most commands by entering the command without parameters. Experienced users can usually configure connections using just the quickstarts and the online help.

Note For all commands in this chapter, refer to the *Cisco MGX 8850 Command Reference* for detailed information.

Note Before you start configuring ATM connections, complete the general switch configuration as described in Chapter 2, "Configuring General Switch Features." Some of the procedures described in this chapter will not work if the switch has not been set up properly.

Quickstart Provisioning Procedures

The following sections present abbreviated procedures that you can use to provision the switch.

Trunk Configuration Quickstart

ATM trunks connect the switch to other ATM switches in the core ATM network. The quickstart procedure in this section provides a summary of the tasks required to configure ATM trunks on the MGX 8850 switch. This procedure is provided as an overview and as a quick reference for those who have previously configured these types of connections.

Note The trunk configuration is not complete until the following procedure has been completed on the switches at both ends of the trunk.

	Command	Purpose
Step 1	*username* *password*	Start a configuration session. Note: To perform all the procedures in this quickstart procedure, you must log in as a user with GROUP1 privileges or higher.
Step 2		Prepare AXSM cards and lines as described in Chapter 3, "Preparing AXSM Cards and Lines for Communication."
Step 3	**addport** <*options*> Related commands: **dspports**	Add and configure ATM ports. This step establishes ATM communications between two ATM devices. Specify **NNI** for interswitch trunks and **VNNI** for virtual trunks. See "Adding ATM Ports," which appears later in this chapter.
Step 4	**addpart** <*options*> Related commands: **dspparts** **dsppart** **cnfpart**	Add and configure partitions. This step configures one or more service partitions, which can be used to control how trunk resources are allocated to different services. See "Configuring Port Resources Using Partitions," which appears later in this chapter.
Step 5	**dnpnport** <*portid*> **cnfpnportsig** <*options*> **uppnport** <*portid*> Related commands: **dsppnports** **dsppnport** <*portid*> **dsppnportsig** <*portid*>	Define the signaling protocol used on the trunk. The default signaling protocol is UNI Version 3.1. Specify **pnni10** for PNNI trunks and either **iisp30** or **iisp31** for IISP trunks. The command for configuring a feeder trunk is: `pop20two.7.PXM.a > cnfpnportsig <portid> -cntlvc <ip>` See "Selecting the Port Signaling Protocol," which appears later in this chapter.

	Command	Purpose
Step 6	**dsppnni-link** **dsppnni-neighbor**	When both ends of the link are configured, verify PNNI communications between the two ends. In the dsppnni-link report, there should be an entry for the port for which you are verifying communications. The Hello state reported should be twoWayInside and the Remote note ID should display the remote node ATM address after the second colon. See "Verifying PNNI Trunk Communications," which appears later in this chapter.
Step 7	**upilmi** *<ifNum> <partId>* **cnfilmi** *<options>* Related commands: **dspports** **dspilmis**	This step is optional. Configure and start ILMI on trunks where you want to support Cisco WAN Manager or use ILMI features. See "Configuring ILMI on a Port," which appears later in this chapter.

Line Configuration Quickstart

ATM Lines connect the switch to ATM end devices, which serve as the boundary between the ATM network and other communications paths or networks. Typical end devices include ATM routers and multiservice concentrators.

The quickstart procedure in this section provides a summary of the tasks required to configure ATM lines on the MGX 8850 switch. This procedure is provided as an overview and as a quick reference for those who have previously configured these types of connections.

Note The line configuration is not complete until the equipment at both ends of the line has been configured with compatible configuration settings.

	Command	Purpose
Step 1	*username* *password*	Start a configuration session. Note: To perform all the procedures in this quickstart procedure, you must log in as a user with GROUP1 privileges or higher.
Step 2		Prepare AXSM cards and lines as described in Chapter 3, "Preparing AXSM Cards and Lines for Communication."
Step 3	**addport** *<options>* Related commands: **dspports**	Add and configure ATM ports. This step establishes ATM layer two communications between two ATM devices. Specify UNI for ATM lines. See "Adding ATM Ports," which appears later in this chapter.
Step 4	**addpart** *<options>* Related commands: **dspparts** **dsppart** **cnfpart**	Add and configure partitions. This step configures one or more service partitions, which can be used to control how line resources are allocated to different network controllers. See "Configuring Port Resources Using Partitions," which appears later in this chapter.

	Command	Purpose
Step 5	**dnpnport** *<portid>*	Bring down the port so it can be configured. The next three steps require this.
Step 6	**cnfpnportsig** *<options>* Related commands: **dsppnports** **dsppnport** *<portid>* **dsppnportsig** *<portid>*	Define the signaling protocol used on the line. The default signaling protocol for UNI lines is UNI Version 3.1. Specify **uni30**, **uni31**, or **uni40**. See "Selecting the Port Signaling Protocol," which appears later in this chapter.
Step 7	**cnfaddrreg** *<portid>* no **addaddr** *<options>* Related commands: **dsppnports** **dspaddr** *<portid>* **deladdr** *<options>*	Configure static ATM addresses for ports that require them. See "Assigning Static ATM Addresses to Destination Ports," which appears later in this chapter.
Step 8	**addprfx** *<portid> atm-prefix* Related commands: **cnfaddrreg** *<portid>* yes **dspprfx** *<portid>*	If dynamic addressing is to be used on a port, define an ATM address prefix that ILMI can use when assigning addresses. See "Configuring ILMI Dynamic Addressing," which appears later in this chapter.
Step 9	**uppnport** *<portid>*	Bring up port after configuration is complete.
Step 10	**upilmi** *<ifNum> <partId>* **cnfilmi** *<options>* Related commands: **dspports** **dspilmis**	Configure and start ILMI on the port. This step is required for dynamic addressing and the ILMI automatic configuration feature. Otherwise, it is optional. See "Configuring ILMI on a Port," which appears later in this chapter.

SVC Configuration Quickstart

Switched Virtual Circuits (SVCs) are the solution for on-demand connections. They are set up as needed and torn down when no longer needed. To enable this dynamic behavior, SVCs use signaling. End systems request connectivity to other end systems and, provided that the requested services are available, the connection is set up at the time of the request. When idle, an SVC is taken down to save network bandwidth.

MGX 8850 switches use the PNNI protocol to develop a logical map of the network. This logical map allows switches to work together to determine how to set up SVCs through the network. PNNI enables automatic SVC configuration, which greatly reduces the time required to set up new networks.

Because the switch automatically sets up SVCs, you do not have to configure SVC routes. However, the switch must be configured correctly before it can set up SVCs. The following quickstart procedure summarizes the tasks required to enable SVC communications. With the exception of CPE configuration, all these tasks are described in this chapter.

Note The tasks in the following procedure do not have to be completed in the order presented. However, all tasks must be completed before SVCs will operate.

	Command	Purpose
Step 1	See "Trunk Configuration Quickstart," which appears earlier in this chapter.	Configure the trunks that link the switches through which the ATM end stations connect.
Step 2	`dsppnni-reachable-addr network`	Verify PNNI connectivity between nodes pairs that will host SVCs. See "Verifying End-to-End PNNI Communications," which appears later in this chapter.
Step 3	See "Line Configuration Quickstart," which appears earlier in this chapter.	Configure lines for the ATM end stations at each end of the SVC, and assign either static or dynamic addressing to each line.
Step 4	See the CPE documentation.	Configure CPE devices for line communications with the MGX 8850 switch.
Step 5	`dsppncons`	This optional step displays the SVC connections that are operating. See "Displaying SVCs," in Chapter 6, "Switch Operating Procedures."

It is beyond the scope of this guide to describe how to configure each model of CPE to communicate with the MGX 8850 switch. To complete this configuration, you will need to learn the capabilities of the CPE and the MGX 8850 switch and define a set of communications parameters that are supported by both devices. For example, the MGX 8850 switch supports UNI 3.1 communications, but if the CPE does not, you'll need to select a signaling protocol (such as UNI 3.0) that is supported by both devices.

Once all the requirements have been met for SVC connections, CPE devices can establish SVC connections to other CPE devices on the same switched network.

SPVC and SPVP Configuration Quickstart

A Soft Permanent Virtual Circuit (SPVC) is a Permanent Virtual Circuit (PVC) that can be rerouted using the Private Network-to-Network Interface (PNNI) Version 1.0 protocol. As with PVCs, SPVCs are full-time connections. A PVC, however, uses a predefined circuit path and will fail if the path is interrupted. Using the PNNI protocol, SPVCs can be rerouted to avoid failed communication links or to use links that offer better bandwidth.

A Soft Permanent Virtual Path (SPVP) is a permanent virtual path that can be rerouted using the Private Network-to-Network Interface (PNNI) Version 1.0 protocol. The difference between an SPVC and an SPVP is that the SPVP supports multiple virtual circuits, whereas a SPVC is by definition a single virtual circuit. As with SPVCs, when an SPVP fails, PNNI can determine if an alternate route exists and reroute the connection.

The quickstart procedure in this section provides a summary of the tasks required to configure SPVCs and SPVPs on the MGX 8850 switch. This procedure is provided as an overview and as a quick reference for those who have previously configured these types of connections.

	Command	Purpose
Step 1	*username* *password*	Start a configuration session. Note: To perform all the procedures in this quickstart procedure, you must log in as a user with SUPER_GP privileges or higher.
Step 2	See "Trunk Configuration Quickstart," which appears earlier in this chapter.	Configure the trunks that link the switches to which the ATM end stations connect.
Step 3	**dsppnni-reachable-addr network**	Verify PNNI connectivity between the two nodes that will host the SPVC or SPVP end points. See "Verifying End-to-End PNNI Communications," which appears later in this chapter.
Step 4	See "Line Configuration Quickstart," which appears earlier in this chapter.	Configure lines for the ATM end stations at each end of the SPVC or SPVP, and assign either static or dynamic addressing to each line.
Step 5	**addcon** *<options>* Related commands: **dspcons** **dspcon** *<ifNum> <vpi> <vci>*	Configure the slave side of an SPVC. See "Configuring SPVCs and SPVPs," which appears later in this chapter.
Step 6	**addcon** *<options>* Related commands: **dspcons** **dspcon** *<ifNum> <vpi> <vci>*	Configure the master side of an SPVC. See "Configuring SPVCs and SPVPs," which appears later in this chapter.

IISP Link Configuration Quickstart

The quickstart procedure in this section provides a summary of the tasks required to configure IISP links on the MGX 8850 switch. This procedure is provided as an overview and as a quick reference for those who have previously configured these types of connections.

	Command	Purpose
Step 1	*username* *password*	Start a configuration session. Note: To perform all the procedures in this quickstart procedure, you must log in as a user with SUPER_GP privileges or higher.
Step 2	See "Trunk Configuration Quickstart," which appears earlier in this chapter.	Configure the ports at each end of the IISP link. When setting the signaling protocol with the **cnfpnportsig** command, be sure to specify iisp30 or iisp31.
Step 3	**addaddr** *<options>*	Add destination addresses to each end of the trunk. See "Defining Destination Addresses for IISP Links," which appears later in this chapter.
Step 4	**addaddr** *<options>*	Add static addresses to destination ports. This step is required when addresses are not dynamically assigned to the CPE at the destination ports. See "Assigning Static ATM Addresses to Destination Ports," which appears later in this chapter.

Virtual Trunk Configuration Quickstart

Virtual trunks are introduced in the "Multiservice Edge Aggregation" section of Chapter 1, "Preparing for Configuration." Figure 4-1 shows illustrates how a virtual trunk is configured.

Figure 4-1 Virtual Trunk Configuration

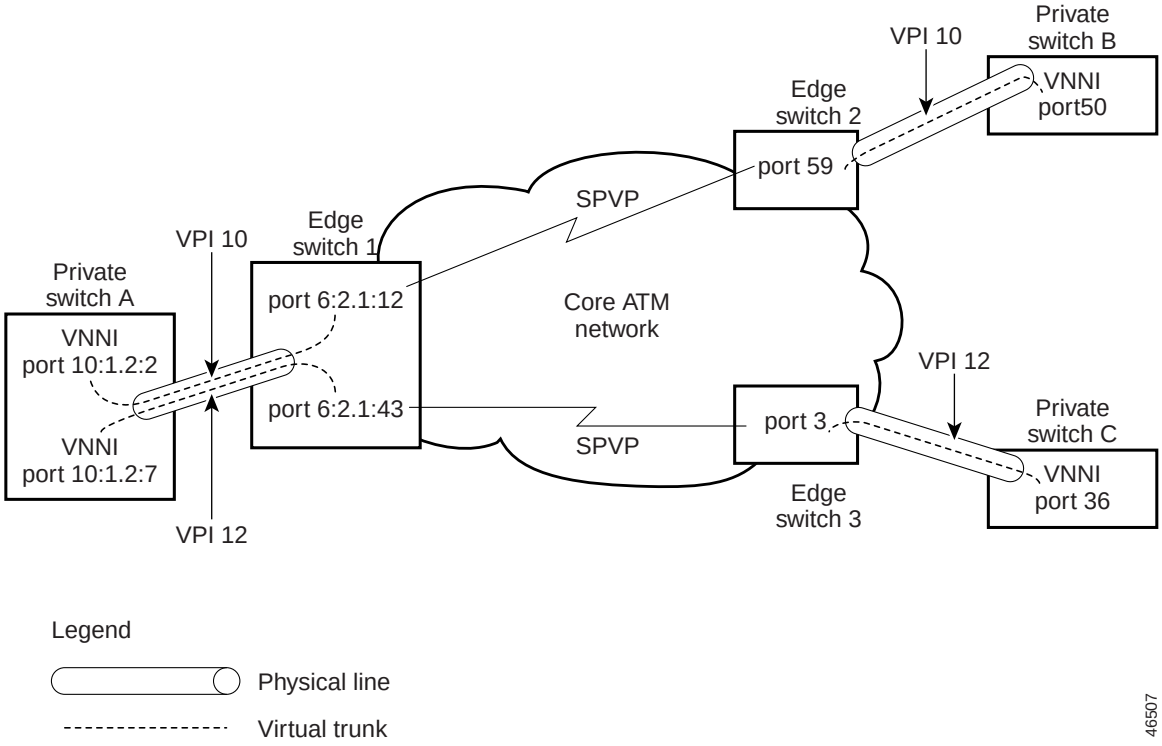


Figure 4-1 shows sample configuration data that you can use as an example when following the quickstart procedure below.

To set up a virtual trunk, the following tasks have to be completed:

- Virtual trunks must be defined between the private network nodes and the core edge nodes.
- The core network operators must define an SPVP for each virtual trunk that connects the core edge nodes on the virtual trunk path.

The MGX 8850 switch supports:

- Up to 256 SPVPs across an ATM core network (or ATM cloud). The range is 0 to 255.
- Up to 60 virtual trunks on a physical interface with a total of 60 per AXSM card and 100 ports per switch.

Note SPVPs are not allowed on virtual trunks.

The following quickstart procedure provides a summary of the tasks required to configure virtual trunks on the MGX 8850 switch. This procedure is provided as an overview and as a quick reference for those who have previously configured these types of connections.

	Command	Purpose
Step 1	*username* *password*	Start a configuration session. Note: To perform all the procedures in this quickstart procedure, you must log in as a user with SUPER_GP privileges or higher.
Step 2	See "Trunk Configuration Quickstart," which appears earlier in this chapter.	Configure the virtual trunk end ports at the private switches. When configuring the trunk, do the following: • Use the **addport** command to select interface type **3** for VNNI. • When adding the partition for the port, enter the same VPI number for the minVpi and maxVpi parameters. This number becomes the VPI number for the trunk. • Use the **cnfpnportsig** command to select PNNI signaling by by setting the **-nniver** option to **pnni10**.
Step 3	See "Trunk Configuration Quickstart, which appears earlier in this chapter.	Configure the virtual trunk end ports at each core edge node. When configuring these ports, do the following: • Use the **addport** command to select interface type **1** for UNI. • When adding the partition for the port, use a VPI range that includes all VPI numbers set for virtual trunks on this line at the private switch. • Use the **cnfpnportsig** command to select no trunk signaling by setting the **-univer** option to **none**.
Step 4	See "Configuring SPVCs and SPVPs," which appears later in this chapter.	Fore each virtual trunk, configure an SPVP between the virtual trunk UNI ports at each edge of the core network.

Feeder Configuration Quickstart

The quickstart procedure in this section provides a summary of the tasks required to configure a connection from an MGX 8850 Release 1 feeder, through one or more MGX 8850 Release 2 switches, and to a remote feeder or CPE. This procedure is provided as an overview and as a quick reference for those who have previously configured these types of connections.

Note The trunk configuration is not complete until the MGX 8850 Release 1 feeder is also configured.

	Command	Purpose
Step 1	*username* *password*	Start a configuration session on the MGX 8850 Release 2 switch. This will be the local routing switch that connects to the feeder. Note: To perform all the procedures in this quickstart procedure, you must log in as a user with GROUP1 privileges or higher.
Step 2	See "Trunk Configuration Quickstart" or "Line Configuration Quickstart," both of which appear earlier in this chapter.	Configure the local routing switch port that leads to the feeder. When configuring the line, do the following: • Use the **addport** command to select either interface type **1** (UNI) or **2** (NNI). Use the same interface type when defining the port on the feeder. • When adding the partition for the port, specify PNNI controller ID **2**. • If CWM will be used to manage the feeder, use the **cnfpnportsig** command to enable IP communications between the switch and the feeder: `pop20two.7.PXM.a > cnfpnportsig <portid> -cntlvc <ip>` Replace *ip* with the IP address assigned to ATM0.
Step 3	**addfdr** <*ifnum*> Related commands: **dspfdr** <*ifnum*>	Define the local routing switch port as a feeder port. See "Defining a Feeder Port," which appears later in this chapter.
Step 4	See the MGX 8850 Release 1 documentation.	At the MGX 8850 Release 1 feeder, use the **addcon** command to add a connection on the link to the MGX 8850 Release 2 switch.
Step 5		Configure the port on the remote routing switch that terminates calls in the core network. If the remote routing switch port connects to a feeder, repeat Steps 2 and 3 to configure the remote feeder trunk. If the remote routing switch port connects to CPE, configure the port for UNI communications.
Step 6	**addcon** <*options*> Related commands: **dspcons**	Create an SPVC from the local routing switch feeder port to the remote routing switch termination port. See "Configuring SPVCs and SPVPs."

Adding ATM Ports

In the previous chapter, you brought up physical lines by specifying the correct line port number. The line ports correspond to line connectors on the switch back cards. Bringing up a line establishes minimal connectivity between two nodes. When you add an ATM port to a line, you enable ATM communications over the line.

Each line can support UNI, NNI, or VNNI ports. UNI ports are used for lines that connect to PBXs, ATM routers, and other ATM devices that connect to the core ATM network through the MGX 8850 switch. NNI ports are used for trunks that connect to other core ATM network devices, such as the MGX 8850 switch. VNNI ports support virtual trunk connections between two ATM end stations.

You must configure one ATM port for each line or trunk to enable ATM communications over that link. You define the port type (UNI, NNI, or VNNI) when you add the ATM port to the line or trunk.

To add an ATM port to a line, use the following procedure.

Step 1 Establish a configuration session using a user name with GROUP1 privileges or higher.

Step 2 Get the line number on which you will add the port. To display a list of the lines and line numbers, enter the following command:

```
mgx8850a.10.AXSM.a > dsplns
```

Tips Remember that you cannot configure a line until you have brought it up as described in "Bringing Up Lines," in Chapter 3, "Preparing AXSM Cards and Lines for Communication."

Step 3 Verify that the line and port number you want to use is not configured. To display a list of the ports configured on the AXSM card, enter the following command:

```
mgx8850a.10.AXSM.a > dspports
```

This command displays all ports on the AXSM card in the ifNum (interface number) column. The interfaces listed include UNI, NNI, and VNNI ports. Pay attention to the port numbers already in use. When you add a port, you must specify a port number that is unique on the AXSM card. For example, if port number 2 is assigned to line 2.1 (bay 2, line 1), you cannot use port 2 on any other line on that AXSM card.

Note The Cisco MGX 8850 switch supports one port per line.

Step 4 To add an ATM port to a line, enter the following command:

```
mgx8850a.10.AXSM.a > addport <ifNum> <bay.line> <guaranteedRate> <maxRate> <sctID>
<ifType> [vpi]
```

Table 4-1 lists the parameter descriptions for adding ports. Figure 4-2 shows the relationship between logical interface numbers and physical lines.

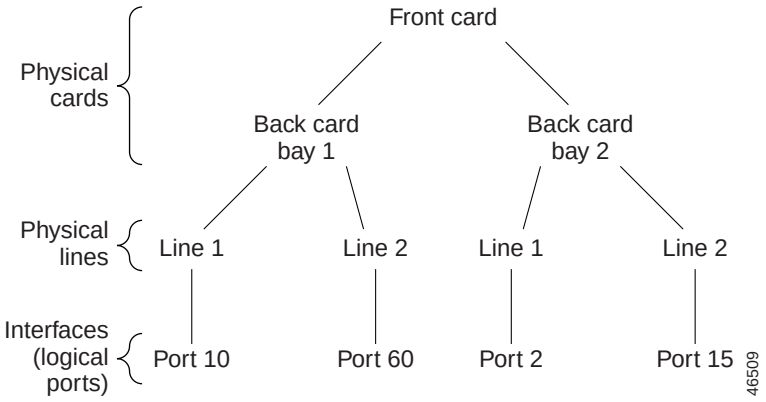
Table 4-1 Parameters for addport Command

Parameter	Description
ifNum	An ATM port is also called an interface. An ATM port is defined by its slot, bay, line, and interface numbers. You do not have to enter a slot number during port configuration because you identify the slot number when you select the card. Enter a number from 1 to 60 to identify this interface. The interface number must be unique on the card to which it is assigned. For UNI and NNI ports, you can assign one logical interface per line. For VNNI ports (for virtual trunks), you can assign multiple logical interfaces per line.
bay	Replace *bay* with **1** if the line is connected to a back card in the upper bay, or replace it with **2** if the line is connected to a back card in the lower bay. Remember that the bay number is always 1 for an AXSM-1-2488.
line	Replace *line* with the number that corresponds to the back card port to which the line is connected. Table 4-2 lists the valid line numbers for each AXSM card.
guaranteedRate	Enter the minimum rate for the port in cells per second (cps). Note that in this release, the *guaranteedRate* value should equal the *maxRate* value. The rate ranges are as follows: OC48: 50 to 5651320 OC12: 50 to 1412830 OC3: 50 to 353207 T3: 50 to 96000 (PLCP) or 104268 (ADM) E3: 50 to 80000
maxRate	Enter the maximum rate for the port in cps. Note that in this release, the *maxRate* value should equal the *guaranteedRate* value. The rate ranges are as follows: OC48: 50 to 5651320 OC12: 50 to 1412830 OC3: 50 to 353207 T3: 50 to 96000 (PLCP) or 104268 (ADM) E3: 50 to 80000
sctID	Enter the port SCT number (0–255). The Cisco policing SCT ID is **2**, and the non-policing ID is **3**.
ifType	Enter a number that indicates the interface type. Enter **1** for UNI, **2** for NNI, and **3** for VNNI, which defines a virtual trunk port.
vpi	When the *ifType* parameter is set to 3 for VNN, enter a VPI number for the virtual trunk in the range of 1 to 4095. This parameter is not required for UNI and NNI ports.

Table 4-2 AXSM Card Types

Front Card	Valid Line Numbers	Valid Bay Numbers
AXSM-16-155	1-8	1, 2
AXSM-4-622	1-2	1, 2
AXSM-1-2488	1	1
AXSM-16-T3E3	1-8	1, 2

Figure 4-2 Relationship Between Cards, Bays, Lines, and Logical Interface Numbers



The following example command defines a line port as a UNI T3 line:

```
mgx8850a.10.AXSM.a > addport 1 1.1 96000 96000 1 1
```

The following example command defines a line port as an OC48 NNI trunk:

```
mgx8850a.10.AXSM.a > addport 2 2.1 5651328 5651328 2 2
```

Step 5 To display a list of the ports configured on the AXSM card, enter the following command:

```
mgx8850a.10.AXSM.a > dspports
```

This command displays all configured ports on the AXSM card. Port numbers are listed in the ifNum (interface number) column. If you want to view information on a particular port, note the number of that port.

Step 6 To display the port configuration, enter the following command:

```
mgx8850a.10.AXSM.a > dspport <ifNum>
```

Replace *ifNum* with the number assigned to the port during configuration. The following example shows the report for this command:

```
pop20two.1.AXSM.a > dspport 1
  Interface Number                : 1
  Line Number                     : 1.1
  Admin State                     : Up        Operational State    : Up
  Guaranteed bandwidth(cells/sec): 1412830   Number of partitions: 1
  Maximum bandwidth(cells/sec)    : 1412830   Number of SPVC       : 0
  ifType                          : NNI       Number of SPVP       : 0
  Port SCT Id                     : 2
  VPI number(VNNI only)           : 0         Number of SVC        : 2
```

Tips To change the port configuration, enter the **cnfport** command, or enter the **delport** command to delete a port configuration. You can also activate and deactivate ports using the **upport** and **dnport** commands. For more information on these commands, refer to the *Cisco MGX 8850 Command Reference*.

Configuring Port Resources Using Partitions

After you add a line or trunk port, you need to configure additional parameters that define how the port resources are used by the switch network controllers. These parameters include the following:

- Range of VPI values
- Range of VCI values
- Guaranteed percent of bandwidth for ingress and egress directions
- Minimum and maximum number of connections

In this release, the MGX 8850 switch supports only the PNNI controller (see Chapter 2, “Configuring General Switch Features”). Some parameters, such as the ranges for VPI and VCI values, are provided so that the port resources can be divided between two or more controllers. Because the current software supports one controller, you can assign all port VPI, VCI, and bandwidth resources to the PNNI controller. However, since the switch supports a maximum number of connections, you may choose to limit the number of connections allowed on a single port. For example, if you allowed the maximum number of connections on all ports, two or three very busy ports could use all available connections and disable communications on all other ports.

The port resources are defined as a group in a controller partition, which is dedicated to a single port controller. You must define one controller partition for the PNNI controller, and you must configure one resource partition for each port that uses the PNNI controller. Figure 4-3 presents a simplified view of the relationship between the port controller, controller partition, and resource partitions.

Figure 4-3 Relationship of Port Controller, Controller Partition, and Resource Partitions

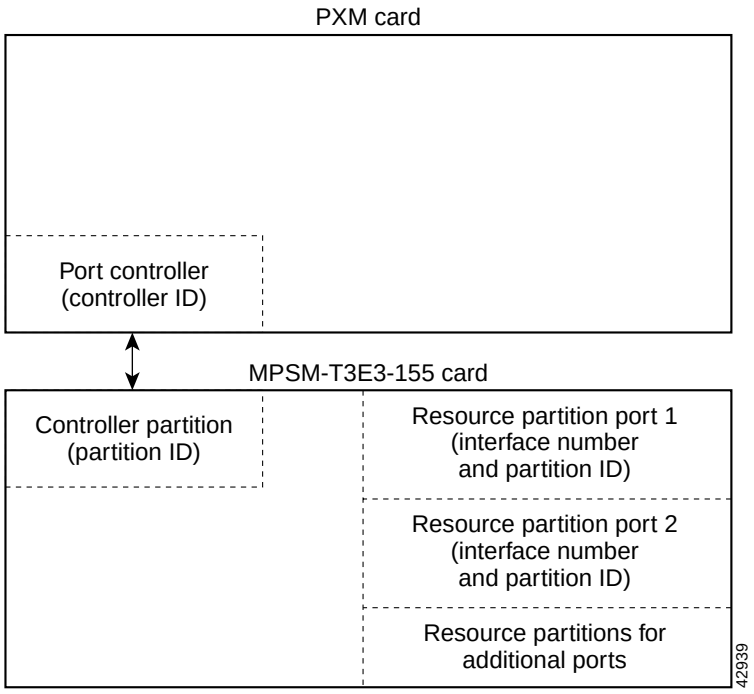


Figure 4-3 shows that the single controller partition connects to the PNNI port controller and to the resource partitions. After you create a port, you must create a resource partition for that port to select the PNNI controller and define which ATM resources the port will use. You do not have to create the controller partition, as it is automatically created when you create the first resource partition. The important concept is that the same controller partition, and therefore the same partition ID, is used for all resource partitions on the same AXSM card. The PNNI controller is identified by the controller ID and the controller partition is identified by the partition ID. The resource partitions are identified by specifying the partition ID in combination with the port ID (interface number).

To create a resource partition for a port, use the following procedure.

Step 1 Establish a configuration session using a user name with GROUP1 privileges or higher.

Note You must add the PNNI controller and add a port before you create a resource partition for a port. For instructions on adding the controller, see "Configuring the PNNI Controller," in Chapter 2, "Configuring General Switch Features." For instructions on adding ports, see "Adding ATM Ports," which appears earlier in this chapter.

Step 2 Determine the port number to which you want to assign the resource partition. To display a list of the ports, enter the following command:

```
mgx8850a.10.AXSM.a > dspports
```

This command displays all ports on the AXSM card in the ifNum (interface number) column.

Step 3 To create a resource partition, enter the following command:

```
mgx8850a.10.AXSM.a > addpart <ifNum> <partId> <ctrlrId> <egrminbw> <egrmaxbw> <ingminbw>
<ingmaxbw> <minVpi> <maxVpi> <minVci> <maxVci> <minConns> <maxConns>
```

Table 4-3 describes the parameters for this command.

Table 4-3 Parameters for the addpart Command

Parameter	Description
ifNum	Interface number or port number. This number identifies the port this resource partition configures. Enter the interface number that was assigned to the port when it was configured (see "Adding ATM Ports").
partId	Partition identification number. Enter a number in the range of 1 to 20. On an AXSM card, this number must be the same for all ports that use the PNNI controller.
ctrlrId	Controller identification number. Enter the number **2** to specify the PNNI controller (see "Configuring the PNNI Controller," in Chapter 2, "Configuring General Switch Features").
egrminbw	Egress minimum bandwidth. Enter the minimum percentage of the outgoing port bandwidth that you want assigned to the PNNI controller. One percent is equal to 0.00001 units. For example, an *egrminbw* of 1000000 = 100%.
egrmaxbw	Egress maximum bandwidth. Enter the maximum percentage of the outgoing port bandwidth that you want assigned to the PNNI controller. One percent is equal to 0.00001 units. For example, an *egrminbw* of 250000 = 25%.
ingminbw	Ingress minimum bandwidth. Enter the minimum percentage of the incoming port bandwidth that you want assigned to the PNNI controller. One percent is equal to 0.00001 units. For example, an *egrminbw* of 500000 = 50%.
ingmaxbw	Ingress maximum bandwidth. Enter the maximum percentage of the incoming port bandwidth that you want assigned to the PNNI controller. One percent is equal to 0.00001 units. For example, an *egrminbw* of 750000 = 75%.
minVpi	Minimum VPI number for this port. For UNI ports, enter a value in the range from 0 to 255. For NNI ports, enter a value in the range from 0 to 4095.
maxVpi	Maximum VPI number for this port. For UNI ports, enter a value in the range from 0 to 255. For NNI ports, enter a value in the range from 0 to 4095. The value for *maxVpi* cannot be less than for *minVpi*.
minVci	Minimum VCI number for this port. For OC-48 AXSM cards, enter a number in the range from 32 to 131072. For all other cards, enter a number in the range from 32 to 65535. To support features planned for the future, Cisco recommends setting the minimum VCI to 35 or higher.
maxVci	Maximum VCI number for this port. For OC-48 AXSM cards, enter a number in the range from 32 to 131072. For all other cards, enter a number in the range from 32 to 65535.

Table 4-3 Parameters for the addpart Command (continued)

Parameter	Description
minConns	Minimum number of simultaneous connections allowed on this port. The minimum number of connections is 0. The type of back card and line determine the maximum number of connections as follows: T3/E3 lines: 65535 per line to a total of 65535 per back card OC3 lines: 32767 per line to a total of 65535 per back card OC12 lines: 32767 per line to a total of 65535 per back card OC48 lines: 131071 per line to a total of 131071 per back card Note that the maximum number of connections is 128K (131,071) for the AXSM front card and the OC48 back card. For the other AXSM back cards, which are used in pairs (upper and lower bays), the maximum number of connections is 64K (65535), which totals 128K for the front card.
maxConns	Maximum number of simultaneous connections allowed on this port. The range is the same as described for the *minConns* parameter, and this parameter must be set to number that is greater than the number defined for *minConns*.

Step 4 To display a list showing the resource partition you have created, enter the following command:

```
mgx8850a.10.AXSM.a > dspparts
```

Step 5 To display the configuration of a specific resource partition, note the interface and partition numbers and enter the following command:

```
mgx8850a.10.AXSM.a > dsppart <ifNum> <partId>
```

Table 4-3 describes the parameters for this command. The following example shows the report provided by the **dsppart** command.

```
pop20two.1.AXSM.a > dsppart 1 2
  Interface Number               : 1
  Partition Id                   : 2        Number of SPVC: 0
  Controller Id                  : 2        Number of SPVP: 0
  egr Guaranteed bw(.0001percent): 1000000  Number of SVC : 2
  egr Maximum bw(.0001percent)   : 1000000
  ing Guaranteed bw(.0001percent): 1000000
  ing Maximum bw(.0001percent)   : 1000000
  min vpi                        : 0
  max vpi                        : 4095
  min vci                        : 32
  max vci                        : 65535
  guaranteed connections         : 0
  maximum connections            : 5000
```

Note For more information on working with partitions, see "Managing Partitions" in Chapter 6, "Switch Operating Procedures."

Selecting the Port Signaling Protocol

The default signaling protocol for all new ports is UNI Version 3.1. If you plan to use this protocol on a line, you can accept this default and skip this section. However, if you plan to use a different protocol on the line, or if you are setting up a PNNI trunk, you must select the correct protocol using the following procedure.

Step 1 Establish a configuration session using a user name with GROUP1 privileges or higher.

Step 2 Use the following command to display a list of the ports you can configure:

popeye2.7.PXM.a > **dsppnports** *<portid>*

Step 3 Use the following command to bring down the port you want to configure:

popeye2.7.PXM.a > **dnpnport** *<portid>*

A port is automatically brought up when you add it. You must down the port before you can change the port signaling protocol. Replace *portid* using the format *slot*[:*bay*].*line*[:*ifNum*]. Table 4-4 describes these parameters.

Step 4 To confirm the port is down, enter the **dsppnports** command. The following example shows the report that appears.

```
pop20one.7.PXM.a > dsppnports
Summary of total connections
(p2p=point to point,p2mp=point to multipoint,SpvcD=DAX spvc,SpvcR=Routed spvc)
Type   #Svcc:   #Svpc:   #SpvcD:  #SpvpD:  #SpvcR:  #SpvpR:  #Total:
p2p:   0        0        0        0        0        0        0
p2mp:  0        0        0        0        0        0        0
                                                      Total=0
Summary of total configured SPVC endpoints
Type   #SpvcCfg: #SpvpCfg:
p2p:   0         0
p2mp:  0         0

Per-port status summary

PortId          IF status         Admin status       ILMI state       #Conns

7.35            up                up                 Undefined        0

7.36            up                up                 Undefined        0

7.37            up                up                 Undefined        0

7.38            up                up                 Undefined        0

Type <CR> to continue, Q<CR> to stop:

10:1.1:1        up                up                 UpAndNormal      0

10:1.2:2        up                up                 Disable          0
```

Step 5 To select the port signaling protocol, enter the following command:

```
pop20two.7.PXM.a > cnfpnportsig <portid> [-univer {uni30|uni31|uni40|none}] [-nniver
{iisp30|iisp31|pnni10}] [-unitype {public|private}] [-addrplan {both|aesa|e164}] [-side
{user|network}] [-vpi <vpi>] [-sigvci <signalling-vci>] [-rccvci <routing-vci>] [-cntlvc
<ip>]
```

The only required parameter for this command is the portid parameter, but the command serves no purpose if you don't enter at least one option with it. If you include some options with the command and omit others, the omitted option remains set to the last configured value.

Table 4-4 shows the components required in the portid parameter, which is used with many commands. Table 4-5 lists and describes the options and parameters for the **cnfpnportsig** command.

Tips

With some commands, you can refer to a port using only the interface number, while other commands require you to enter a complete port identification number, which includes the slot, bay, line, and interface numbers. When entering commands at the PXM45 switch prompt, you always need to specify the complete port identification number. When entering commands at the AXSM switch prompt, you can enter only the interface number, because the interface number is unique on the AXSM card and identifies the slot, bay, and line for the port.

Table 4-4 Port Identification Parameters

Parameter	Description
slot	Enter the slot number for the card that hosts the port you are configuring.
bay	Replace *bay* with **1** if the line is connected to a back card in the upper bay, or replace it with **2** if the line is connected to a back card in the lower bay. Remember that the bay number is always **1** for an AXSM-1-2488.
line	Replace *line* with the number that corresponds to the back card port to which the line is connected. Table 4-2 lists the valid line numbers for each AXSM card.
ifNum	An ATM port is also called an interface. Enter a number from 1 to 60 to identify this interface. The interface number must be unique on the card to which it is assigned. An ATM port is defined by its slot, bay, line, and interface numbers. You do not have to enter a slot number during port configuration because you identify the slot number when you select the card.

Table 4-5 PNNI Port Signaling Configuration Parameters

Parameter	Description
portid	Port identifier in the format *slot*:*bay.line*:*ifnum*. These parameters are described in Table 4-4.
-univer	When configuring PNNI signaling for a UNI port, you can use this option to specify which version of UNI signaling you want the port to use. You can select UNI version 3.0 (**uni30**), UNI version 3.1 (**uni31**), UNI version 4.0 (**uni40**) or no signaling (**none**). The default value is **uni31**. For lines that will support ABR SVCs, select **uni40**. The UNI ports at each end of a virtual trunk SPVP must be set to **none**. SPVCs and SPVPs can use UNI 3.x or 4.0 signaling.
-nniver	When configuring PNNI signaling for an NNI port, you can use this option to specify which signaling protocol you want the port to use. You can select IISP version 3.0 (**iisp30**), IISP version 3.1 (**iisp31**), or PNNI version 1.0 (**pnni10**).
-unitype	When configuring PNNI signaling for a UNI port, you can use this option to specify the UNI type. You can define the port as a private UNI port (**private**) or as a public UNI port (**public**). The default value is **private**.

Table 4-5 PNNI Port Signaling Configuration Parameters (continued)

Parameter	Description
-addrplan	When configuring PNNI signaling for a UNI port, this parameter specifies the ATM address plan used on this port. You can select AESA (**aesa**), E.164 (**e164**), or both (**both**). The default value is **aesa**.
-side	Defines the role of the signaling service used on the port. This parameter applies to IISP ports when static addressing is used (address registration is disabled). If this is a UNI connection or an NNI connection within the network, select **network**. For connections to other networks, you might need to select **user** (this is negotiated with the administrators of the other network). The default value is **network**.
-vpi	Defines the VPI for signaling services on this port. Enter a value in the range from 0 to 4095. The default value is 0.
-cntlvc	This option defines a feeder trunk. The syntax for the feeder trunk definition is: `pop20two.7.PXM.a > cnfpnportsig <portid> -cntlvc <ip>` Replace *ip* with the IP address defined for ATM0. For information on setting the ATM0 address, refer to "Setting Up ATM WAN Connections" in Appendix C, "Supporting and Using Additional CLI Access Options."
-sigvci	Defines the VCI for signaling services on this port. The default value is **5**, which is the well-known, reserved VCI for signaling services on VPI 0. If you choose another VCI for signaling, choose a VCI value in the range from 32 to 65535. Otherwise, the VCI can conflict with other VCIs in the reserved range from 0 to 31 on VPI 0.
-rccvci	Defines the VCI for the PNNI routing control connection (RCC) on this port. The default value is **18**, which is the well-known, reserved VCI for this services on VPI 0. If you choose another VCI for signaling, choose a VCI value in the range of 32 to 65535. Otherwise, the VCI can conflict with other VCIs in the reserved range from 0 to 31 on VPI 0.

Note The selection of UNI or NNI is made when the port is added with the **addport** command. You cannot use the -univer and -nniver options to change the port type.

The following example illustrates how to configure an NNI port to use PNNI Version 1.0 signaling.

```
popeye2.7.PXM.a > cnfpnportsig 1:1.1:1 -nniver pnni10
```

Step 6 Use the following command to bring up the port you just configured:

popeye2.7.PXM.a > **uppnport** *<portid>*

Replace *portid* using the format *slot*:*bay.line*:*ifNum*. Table 4-4 describes these parameters.

Step 7 To verify the status of the port, enter the **dsppnports** command.

Step 8 To display the configuration of the PNNI port, enter the following command:

popeye2.7.PXM.a > **dsppnport** *<portid>*

Replace *portid* using the format *slot*:*bay.line*:*ifNum*. Table 4-4 describes these parameters. The following example shows the report for this command.

```
pop20one.7.PXM.a > dsppnport 10:1.1:1
Port:           10:1.1:1              Logical Id:     17438721
IF status:      up                    Admin Status:     up
UCSM:           enable
Auto-config:    enable                Addrs-reg:        enable
IF-side:        network               IF-type:          nni
UniType:        private               version:          pnni10
Input filter:   0                     Output filter:    0
minSvccVpi:             0             maxSvccVpi:         4095
minSvccVci:            35             maxSvccVci:        65535
minSvpcVpi:             1             maxSvpcVpi:         4095
       #SpvcCfg:  #SpvcActive:      #SpvpCfg:  #SpvpActive:
p2p :       0             0              0            0
p2mp:       0             0              0            0
       #Svcc:     #Svpc:            Total:
p2p :       0          0               0
p2mp:       0          0               0
                                          Total :            0
```

Verifying PNNI Communications

After setting up trunks or when problems occur, use the procedures in this section to determine if PNNI is operating. The next section describes how to verify PNNI communications on a single trunk. The following section describes how to verify PNNI communications between two nodes, which can be separated by multiple PNNI links.

Verifying PNNI Trunk Communications

After you configure both ends of a PNNI trunk, it should be ready to support SVCs and any SPVCs or SPVPs that are configured. To verify that the trunk is functioning, use the following procedure.

Step 1 Establish a CLI session using a user name at any access level. When both ends of the trunk are connected to MGX 8850 switches, you can start the CLI session at either end.

Step 2 If you don't know the line number you are validating, you can view the port and line numbers by entering the **dsppnports** command. The first three numbers identify the slot, bay, and line. For example, port 10:2.1:3 represents slot 10, bay 2, line 1. The remaining number is the interface number assigned with the addport command.

Step 3 Enter the **dsppnni-link** command as follows:

```
pop20two.7.PXM.a > dsppnni-link
```

The **dsppnni-link** command displays a report for every PNNI link on the switch. The following example shows the report for a switch with a single PNNI link.

```
pop20two.7.PXM.a > dsppnni-link

node index   : 1
Local port id:   16848897          Remote port id:   17438721
Local Phy Port Id: 1:1.1:1
   Type. lowestLevelHorizontalLink     Hello state....... twoWayInside
   Derive agg...........          0     Intf index...........  16848897
   SVC RCC index........          0     Hello pkt RX.........         10
                                        Hello pkt TX.........          9
   Remote node name.......pop20one
   Remote node id.........56:160:47.00918100000000107b65f33c.00107b65f33c.01
   Upnode id..............0:0:00.000000000000000000000000.000000000000.00
   Upnode ATM addr........00.000000000000000000000000.000000000000.00
   Common peer group id...00:00.00.0000.0000.0000.0000.0000.00
```

In the **dsppnni-link** command report, there should be an entry for the port for which you are verifying communications. The Local Phy Port Id field in this entry displays the port id in the same format shown in the **dsppnports** command report. The Hello state reported for the port should be twoWayInside and the Remote note ID should display the remote node ATM address after the second colon.

In the example above, the report shown is for port 1:1.1:1. The Hello state is **twoWayInside**, and the ATM address of the node at the other end of the link is **47.00918100000000107b65f33c.00107b65f33c.01**. This link is ready to support connections between the two switches.

Tips

If the Hello state for the link is oneWayInside, that side is trying to communicate. Check the status at the other end. Remember that the configuration at each end of the trunk must be compatible with that on the other end. For example, if ILMI auto configuration is configured at one end and not at the other, the Hello state cannot change to twoWayInside.

Verifying End-to-End PNNI Communications

When connections between two nodes travel over multiple trunks, use the following procedure to verify that the PNNI communications path is operational.

Step 1 Establish a CLI session using a user name at any access level. When both ends of the communications path are connected to MGX 8850 switches, you can start the CLI session at either end.

Step 2 Write down the ATM address of the switches at each end of the PNNI communications path. To display the ATM address on an MGX 8850 switch, use the **dsppnni-node** command. The ATM address appears in the dsppnni-node command report as shown in the following example.

```
pop20two.7.PXM.a > dsppnni-node

node index: 1                       node name: pop20two
   Level..............        56     Lowest..............      true
   Restricted transit..       off     Complex node........       off
   Branching restricted        on
   Admin status........        up     Operational status..        up
   Non-transit for PGL election..       off
   Node id..............56:160:47.00918100000000001a531c2a.00001a531c2a.01
   ATM address...........47.00918100000000001a531c2a.00001a531c2a.01
   Peer group id.........56:47.00.9181.0000.0000.0000.0000.00
```

Step 3 Enter the **dsppnni-reachable-addr** command as follows:

```
pop20two.7.PXM.a > dsppnni-reachable-addr network
```

The **dsppnni-reachable-addr** command lists all the PNNI nodes with which the node you are managing can communicate. The following example shows the report for a switch with a single PNNI link.

```
pop20two.7.PXM.a > dsppnni-reachable-addr network

scope...............          0     Advertising node number         2
Exterior............     false
ATM addr prefix.....47.0091.8100.0000.0010.7b65.f33c/104
Advertising nodeid..56:160:47.00918100000000107b65f33c.00107b65f33c.01
Node name...........pop20one
```

The remote node is identified in the Advertising nodeid row. The information before the first colon (56) is the PNNI level, the information between the first and second colons (160) is the ATM address length, and the remainder of the node ID is the ATM address for the remote node.

When the **dsppnni-reachable-addr** command lists the ATM address of a remote node, you have verified PNNI communications with that node.

Tips If you cannot verify communications with a remote node, try verifying communications across each of the links between the nodes as described in the previous section, "Verifying PNNI Trunk Communications."

Defining Destination Addresses for IISP Links

Typically, an IISP link joins two independent networks. IISP is used instead of PNNI so that the topologies of the two networks remain unknown to the each other.

When you create an IISP link, you must identify destination addresses for the link. These addresses identify which ATM nodes are accessible through this link. After you define these addresses, all requests for these addresses are routed over the IISP link to the other network. To enable bidirectional call initiation, destination addresses must also be defined at the remote end of the link.

To add destination addresses to an IISP link, do the following.

Step 1 Establish a configuration session using a user name with GROUP1 privileges or higher.

Step 2 To locate the port to which you want to add an address, enter the **dsppnports** command.

Step 3 Specify an ATM address using the following command:

```
popeye2.7.PXM.a > addaddr <portid> <atm-address> <length> -type ext -proto static [-plan
{e164 | nsap}] [-scope scope] [-redistribute {yes | no}]
```

Note The **addaddr** command is used to define destination addresses for IISP links and to specify static addresses for links to CPE. The command format above shows the options as they apply when defining destination addresses for IISP links.

Table 4-7 describes the parameters used with the **addaddr** command.

Table 4-6 ATM Address Configuration Parameters

Parameter	Description
portid	Port identifier in the format *slot*:*bay.line*:*ifnum*. These parameters are described in Table 4-4.
atm-address	Enter the ATM address using up to 40 nibbles. The ATM address can include up to 20 bytes, which is 40 nibbles or 160 bits. To summarize a group of destination addresses, enter an ATM address that is less than 20 bytes and includes the common bytes in the group of destination addresses.
length	Enter the length, in bits, of the address you specified with the *atm-address* parameter. Each nibble is equal to 4 bits. The acceptable range for the parameter is from 0 to 160 bits. When you enter a complete 20-byte ATM address, the length is 160. When you summarize a group of destination addresses, the length is equal to the number of bytes entered multiplied by 8.
-type	Enter the address type, which is **ext** (external) for IISP destination addresses. The **int** (internal) value is used when creating static addresses for links to CPE. Default = **int**.
-proto	For IISP destination addresses, specify the **-proto** option with the **static** value. The **local** value applies to CPE links. Default = **local**.

Table 4-6 ATM Address Configuration Parameters (continued)

Parameter	Description
-plan	Enter the address plan, which is either **e164** (E.164) or **nsap** (NSAP). For an NSAP address, the first byte of the address automatically implies one of the three NSAP address plans: NSAP E.164, NSAP DCC, or NSAP ICD. Default = **nsap**.
scope	PNNI scope of advertisement. The scope defines the level of the PNNI hierarchy at which this address is advertised. Enter 0 to advertise the destination address to all nodes in the node's peer group. Range: 0 through 104. Default = **0**.
-redistribute	Specifies whether or not the ATM address should be distributed or advertised to PNNI neighbor nodes. Enter **yes** to enable distribution and enter **no** to disable. When this option is set to **yes**, the node distributes the address to the PNNI neighbors defined with the scope option. When set to **no**, the address is not advertised to any other nodes. Default = **no**.

Step 4 To verify that the new address has been assigned, enter the following command:

popeye2.7.PXM.a > **dspaddr** <*portid*>

Replace *portid* with the port address using the format *slot*:*bay.line*:*ifnum*. These parameters are described in Table 4-4.

Assigning Static ATM Addresses to Destination Ports

When a CPE does not support ILMI, the switch cannot automatically determine the CPE address. To enable communications with the CPE, you must assign a static ATM address to the port leading to the CPE. The static address must match the address used by the CPE. When assigning the static address, you can use command options to define how widely the static address is advertised within the switch network. Use the following procedure to define a static address for a UNI port.

Step 1 Establish a configuration session using a user name with GROUP1 privileges or higher.

Step 2 To locate the port to which you want to add an address, enter the **dsppnports** command.

Step 3 Use the following command to turn off automatic address registration (it is enabled by default) on the port that will use the static address:

```
popeye2.7.PXM.a > cnfaddrreg <portid> no
```

Replace *portid* using the format *slot*:*bay.line*:*ifNum*. Table 4-4 describes these parameters.

Step 4 Specify an ATM address for the port using the following command:

```
popeye2.7.PXM.a > addaddr <portid> <atm-address> <length> [-type int] [-proto local]
[-plan {e164 | nsap}] [-scope scope] [-redistribute {yes | no}]
```

Note The **addaddr** command is used to specify static addresses for UNI links to CPE and to define destination addresses for IISP links. The command format above shows the options that apply when defining static addresses for CPE.

Replace *portid* with the ID you used with the **cnfaddreg** command described earlier. Table 4-7 describes the other parameters used with the **addaddr** command.

Note The static ATM address you choose should conform to the address plan for your network. For more information on address planning, see "Guidelines for Creating an ATM Address Plan" in Chapter 1, "Preparing for Configuration."

Table 4-7 ATM Address Configuration Parameters

Parameter	Description
portid	Port identifier in the format *slot*:*bay.line*:*ifnum*. These parameters are described in Table 4-4.
atm-address	Enter the ATM address using up to 40 nibbles. The ATM address can include up to 20 bytes, which is 40 nibbles or 160 bits.
length	Enter the length, in bits, of the address you specified with the *atm-address* parameter. Each nibble is equal to 4 bits. The acceptable range for the parameter is from 0 to 160 bits.
-type	Enter the address type, which is **int** (internal) for CPE static addresses. The **ext** (external) value is used when creating destination addresses for IISP links. Note that because the default value is **int**, you do not have to specify this option when defining static CPE addresses. Default = **int**.
-proto	For CPE static addresses, specify the **-proto** option with the **local** value. The **static** value applies to IISP links. Note that because the default value is **local**, you do not have to specify this option when defining static CPE addresses. Default = **local**.
-plan	Enter the address plan, which is either **e164** (E.164) or **nsap** (NSAP). For an NSAP address, the first byte of the address automatically implies one of the three NSAP address plans: NSAP E.164, NSAP DCC, or NSAP ICD. Default = **nsap**.

Table 4-7 ATM Address Configuration Parameters (continued)

Parameter	Description
scope	PNNI scope of advertisement. The scope defines the level of the PNNI hierarchy at which this address is advertised. Enter 0 to advertise the destination address to all nodes in the node's peer group. Range: 0 through 104. Default = **0**.
-redistribute	Specifies whether or not the ATM address should be distributed or advertised to PNNI neighbor nodes. Enter **yes** to enable distribution and enter **no** to disable. When this option is set to **yes**, the node distributes the address to the PNNI neighbors defined with the scope option. When set to **no**, the address is not advertised to any other nodes. Default = **no**.

The following example assigns an ATM address to port 9:1.2:2:

```
popeye2.7.PXM.a > addaddr 1:2.1:3 47.1111.1111.1111.1111.1111.1111.1111.1111.1111.11 160
```

Step 5 To verify that the new address has been assigned, enter the **dspaddr** command as shown in the following example:

```
pop20two.7.PXM.a > dspaddr 1:2.1:3
47.1111.1111.1111.1111.1111.1111.1111.1111.1111.11
length: 160    type: internal      proto: local
scope: 0       plan: nsap_icd      redistribute: false
```

Configuring ILMI on a Port

ILMI is optional on all ports. Use ILMI on a port when you want to do any of the following:

- Use ILMI automatic configuration, which negotiates ATM communication parameters.
- Use ILMI address registration, which negotiates an ATM address for an attached CPE using an ILMI prefix assigned to the port.
- Enable CWM auto-discovery on a link, which allows CWM to search for and discover MGX 8850 switches that it can manage.

ILMI is enabled by default on all ports and remains in a down state until ILMI is started. There are two ways to start ILMI on a port. To configure and start ILMI with a single command, use the **cnfilmi** command. To start ILMI using the default values, use the **upilmi** command. The following sections describe how to:

- Configure ILMI traps and signaling and start ILMI
- Configure ILMI automatic configuration
- Configure ILMI dynamic addressing
- Start ILMI with the default trap and signaling parameters

Note For information on additional ILMI management procedures, see "Managing ILMI" in Chapter 6, "Switch Operating Procedures."

Configuring ILMI Traps and Signaling

The default ILMI configuration uses the standard ILMI signaling VPI and VCI, sets three ILMI signaling timers, and enables the distribution of ILMI management messages (traps) to SNMP managers such as CWM. If the defaults are acceptable, you can start ILMI on the port using the **upilmi** command. To change the defaults and start ILMI, use the following procedure.

Note When ILMI is configured and started at one end of a link, it must be started and properly configured at the other end of the link before the link will operate properly.

Step 1 Establish a configuration session using a user name with GROUP1 privileges or higher.

Step 2 If you want to preview the current ILMI configuration for a port, enter the **dspilmis** command. The following example shows the **dspilmis** command report:

```
pop20two.1.AXSM.a > dspilmis

    Sig.  rsrc   Ilmi   Sig   Sig Ilmi   S:Keepalive T:conPoll K:conPoll
    Port  Part  State   Vpi   Vci Trap   Interval    Interval  InactiveFactor
    ----  ----   ----  ----  ---- ---  ------------  ---------- ----------
    1     2      On     0     16    On           1           5          4
    2     2     Off     0     16    On           1           5          4
    3     2     Off     0     16    On           1           5          4
```

The example above shows that ILMI is enabled on port 1 (ILMI State = On) and is disabled on ports 2 and 3 (ILMI State = Off). All other ILMI parameters are set to the default values.

Note The ILMI state displayed by the **dspilmis** command is the configuration state, not the operational state, which appears when you enter the **dsppnports** or **dsppnilmi** commands.

Step 3 Enter the **cnfilmi** command as follows:

```
pop20one.10.AXSM.a > cnfilmi -if <ifNum> -id <partitionID> [-ilmi <ilmiEnable>] [-vpi
<vpi>] [-vci <vci>] [-trap <ilmiTrapEnable>] [-s <keepAliveInt>] [-t
<pollingIntervalT491>] [-k <pollInctFact>]
```

Table 4-8 describes the parameters for the **cnfilmi** command.

Table 4-8 cnfilmi Command Configuration Parameters

Parameter	Description
ifNum	Interface number or port number. This number identifies the port on which you are configuring ILMI. Enter the interface number that was assigned with the **addport** command (see "Adding ATM Ports").
partitionID	Partition ID number. This number identifies the PNNI partition assigned to the port. Enter the partition number that was assigned to the port with the **addpart** command (see "Configuring Port Resources Using Partitions").

Table 4-8 cnfilmi Command Configuration Parameters (continued)

Parameter	Description
ilmiEnable	ILMI enable parameter. To change the current state of ILMI, enter **1** to enable or start ILMI or **2** to disable ILMI. Note that the default value is **1**, which causes ILMI to start whenever the **cnfilmi** command is entered, unless you enter this parameter with value **2**. Default = **1**, enable.
vpi	ILMI signaling VPI. If you need to change the default, enter a VPI number in the range of 0 to 255. Note that changing this value disables ILMI communications until the device at the remote end of the line has been configured for the same ILMI VPI. Default = **0**.
vci	ILMI signaling VCI. If you need to change the default, enter a VCI number in the range of 0 to 65535. Note that changing this value disables ILMI communications until the device at the remote end of the line has been configured for the same ILMI VCI. Default = **16**.
ilmiTrapEnable	ILMI trap distribution. When ILMI is started on a port, ILMI traps are sent to SNMP managers such as CWM. To enable or disable the distribution of ILMI traps, enter **1** to enable ILMI traps or **2** to disable ILMI traps. Default = **1**, enable.
keepAliveInt	ILMI keep alive timer. Range: 1 through 65535. Default = **1**.
pollingIntervalT491	ILMI polling interval T491 timer. Range: 0 through 65535. Default = **5**.
pollInctFact	ILMI polling factor K timer. Range: 0 through 65535. Default = **4**.

Step 4 To confirm your configuration changes, enter the **dspilmis** command.

Configuring ILMI Automatic Configuration

The MGX 8850 supports the automatic configuration feature of ILMI 4.0, which allows two devices that share a link to share their configurations and negotiate a common set of communication parameters. For example, if two network devices share a link and are configured for different maximum VCIs on a partition, the automatic configuration feature can determine and select the highest VCI supported by both nodes. To use ILMI automatic configuration, the devices at each end of the link must support this ILMI 4.0 feature.

To enable or disable automatic configuration on a port, use the **cnfautocnf** command as described in the following procedure.

Note A link between two nodes will not operate correctly if the ILMI automatic configuration feature is enabled at one end and disabled at the other.

Step 1 Establish a configuration session using a user name with GROUP1 privileges or higher.

Step 2 To display the automatic configuration status of a port, use the **dsppnport** command. For example:

```
pop20two.7.PXM.a > dsppnport 1:1.1:1
Port:           1:1.1:1               Logical Id:     16848897
IF status:      up                    Admin Status:     up
UCSM:           enable
Auto-config:    enable                Addrs-reg:        enable
IF-side:        network               IF-type:          nni
UniType:        private               version:          pnni10
Input filter:   0                     Output filter:    0
minSvccVpi:            0              maxSvccVpi:         4095
minSvccVci:           35              maxSvccVci:        65535
minSvpcVpi:            1              maxSvpcVpi:         4095
       #SpvcCfg:  #SpvcActive:      #SpvpCfg:  #SpvpActive:
p2p :       0             0              0            0
p2mp:       0             0              0            0
       #Svcc:     #Svpc:            Total:
p2p :       0          0               0
p2mp:       0          0               0
                                          Total :            0
```

The Auto-config field shows whether the automatic configuration feature is enabled or disabled.

Step 3 If you want to enable or disable automatic configuration, down the port to be configured with the **dnpnport** command. For example:

```
pop20one.7.PXM.a > dnpnport 1:1.1:1
```

Step 4 To enable or disable the automatic configuration feature, enter the **cnfautocnf** command as follows:

```
pop20one.7.PXM.a > cnfautocnf <portid> <yes | no>
```

Replace *portid* with the port address using the format *slot*:*bay.line*:*ifnum*. These parameters are described in Table 4-4.

Enter **yes** to enable automatic configuration or enter **no** to disable automatic configuration. The default is **yes**.

Step 5 Up the port you configured with the **uppnport** command. For example:

```
pop20one.7.PXM.a > uppnport 1:1.1:1
```

Step 6 To verify the change, re-enter the **dsppnport** command.

Configuring ILMI Dynamic Addressing

Dynamic ATM addressing is enabled by default on all MGX 8850 ports. Once ILMI is started, ILMI can negotiate ATM addresses for CPE connected to the port. To determine the ATM address for the CPE, the switch uses a 13-byte ILMI prefix that is assigned to the port and the MAC address from the CPE. For dynamic addressing to work, the remote device must support it. ILMI versions 3.x and 4.0 support dynamic address registration.

The default ILMI prefix matches the PNNI node prefix and the SPVC prefix, both of which are described in "Guidelines for Creating an ATM Address Plan" in Chapter 1, "Preparing for Configuration." If you change the PNNI node prefix, the SPVC prefix and the ILMI prefix remain unchanged. If you change the SPVC prefix, the ILMI prefix will change with it, as long as no ILMI prefix is assigned directly to the port. To eliminate the possibility of having a future SPVC prefix change effect dynamic addressing on a port, assign one or more ILMI prefixes to the port.

The following procedure describes how to enable or disable dynamic addressing and how to assign an ILMI address prefix to a port.

Note The MGX 8850 switch supports up to 255 ILMI prefixes per AXSM card, and these prefixes can be assigned to one port or distributed among the ports.

Step 1 Establish a configuration session using a user name with GROUP1 privileges or higher.

Step 2 To display the dynamic addressing status of a port, use the **dsppnport** command. For example:

```
pop20two.7.PXM.a > dsppnport 1:1.1:1
Port:           1:1.1:1              Logical Id:     16848897
IF status:      up                   Admin Status:     up
UCSM:           enable
Auto-config:    enable               Addrs-reg:        enable
IF-side:        network              IF-type:          nni
UniType:        private              version:          pnni10
Input filter:   0                    Output filter:    0
minSvccVpi:             0            maxSvccVpi:         4095
minSvccVci:            35            maxSvccVci:        65535
minSvpcVpi:             1            maxSvpcVpi:         4095
       #SpvcCfg:  #SpvcActive:      #SpvpCfg:  #SpvpActive:
p2p :       0             0              0            0
p2mp:       0             0              0            0
       #Svcc:     #Svpc:            Total:
p2p :       0          0               0
p2mp:       0          0               0
                                          Total :            0
```

The Auto-reg field shows whether the dynamic addressing feature is enabled or disabled.

Step 3 To view the ILMI prefixes assigned to a port, enter the **dspprfx** command as follows:

```
pop20one.7.PXM.a > dspprfx <portid>
```

Replace *portid* with the port address using the format *slot*:*bay.line*:*ifnum*. These parameters are described in Table 4-4. For example:

```
pop20one.7.PXM.a > dspprfx 1:1.1:1

INFO:  No Prefix registered
```

In the example above, no ILMI prefixes have been assigned to the port, so the port will use the prefix configured for the SPVC prefix.

Step 4 If you want to change the dynamic addressing configuration, bring down the port to be configured with the **dnpnport** command. For example:

```
pop20one.7.PXM.a > dnpnport 1:1.1:1
```

Step 5 To enable or disable dynamic address registration, enter the following command:

```
popeye2.7.PXM.a > cnfaddrreg <portid> <yes | no>
```

Enter **yes** to enable dynamic address configuration or enter **no** to disable it. The default is **yes**.

Step 6 Use the following command to define an ATM prefix for a port:

```
popeye2.7.PXM.a > addprfx <portid> <atm-prefix>
```

Replace *portid* using the format *slot*:*bay.line*:*ifNum*. Table 4-4 describes these parameters.

Replace *atm-prefix* with the 13-byte ATM address prefix that you want the dynamically assigned address to use. Specify the address prefix using 26 hexadecimal digits. The range for each digit is 0 through F (0 through 9, A, B, C, D, E, and F).

Note The address prefix you choose should conform to the address plan for your network. For more information on address planning, see "Guidelines for Creating an ATM Address Plan" in Chapter 1, "Preparing for Configuration."

Tips Each hexadecimal digit represents 1 nibble (four bits), and each pair of hexadecimal digits represents a byte. There are 13 pairs of hexadecimal digits in the prefix, or 26 total digits.

Step 7 Up the port you configured with the **uppnport** command. For example:

```
pop20one.7.PXM.a > uppnport 1:1.1:1
```

Step 8 To verify the proper ATM prefix configuration for a port, re-enter the **dspprfx** command.

Starting ILMI with the Default or Existing Values

The **upilmi** command starts ILMI on a port with the existing ILMI configuration, which is the default configuration when ILMI has never been configured on that port. Although ILMI starts automatically when you configure it with the **cnfilmi** command, you might have to bring down ILMI with the **dnilmi** command to make a configuration change such as adding an ILMI prefix. To start or restart ILMI with the **upilmi** command, use the following procedure.

Step 1 Establish a configuration session using a user name with GROUP1 privileges or higher.

Step 2 If you don't know the interface number and partition ID for the port on which you are starting ILMI, use the **dspparts** command as shown in the following example.

```
pop20two.1.AXSM.a > dspparts
if  part Ctlr egr     egr     ingr    ingr    min max   min   max  min   max
Num ID   ID   GuarBw  MaxBw   GuarBw  MaxBw   vpi vpi   vci   vci  conn  conn
             (.0001%)(.0001%)(.0001%)(.0001%)
-------------------------------------------------------------------------------
 1   2    2 1000000 1000000 1000000 1000000    0 4095    32 65535      0   5000
 2   2    2 1000000 1000000 1000000 1000000    0 4095    32 65535      0   5000
 3   2    2 1000000 1000000 1000000 1000000    0  255    32 65535      0   1000
```

Tips To see the relationship between interface numbers and lines, use the **dspports** command.

Step 3 To start ILMI on a port, use the **upilmi** command as follows:

```
pop20one.10.AXSM.a > upilmi <ifNum> <partId>
```

Replace *ifNum* with the interface number for the port, and replace *partId* with the partition number assigned to the port. For example:

```
pop20one.10.AXSM.a > upilmi 2 1
```

Step 4 To display the ILMI status of all the ports on an AXSM card, use the **dspilmis** command. For example:

```
pop20two.1.AXSM.a > dspilmis

    Sig. rsrc  Ilmi  Sig  Sig Ilmi  S:Keepalive T:conPoll K:conPoll
    Port Part State  Vpi  Vci Trap  Interval    Interval  InactiveFactor
    ---- ----  ---- ---- ---- --- ------------  ---------- ----------
    1    2    On     0    16    On          1          5          4
    2    2   Off     0    16    On          1          5          4
    3    2   Off     0    16    On          1          5          4
```

The ILMI State column displays the configured state for ILMI, which is On if ILMI is enabled and Off if ILMI is disabled (use **dsppnports** or **dsppnilmi** to see the operational state). The other columns display ILMI configuration parameters described in Table 4-8.

Configuring SPVCs and SPVPs

SPVCs and SPVPs are created between two nodes and must be configured on each node. The master node is responsible for routing and rerouting. The slave node is responsible for responding to requests from the master during connection setup and rerouting.

The master and slave relationships exist for each SPVC or SPVP and apply only to the SPVC or SPVP connection. For example, you can have one SPVC with a master on Node A and a slave on Node B, and then create another with the Master on Node B and the slave on Node A. It is good practice to distribute the master side of SPVCs and SPVPs among the network nodes so that the route processing task is distributed.

The following sections describe how to configure slave and master SPVC and SPVP connections.

Tips The configuration of SPVCs and SPVPs is very similar. The difference is that SPVPs are assigned VCI 0 and do not use non-zero VCI numbers. An SPVC requires a non-zero VCI.

Configuring the Slave Side of SPVCs and SPVPs

To configure the slave side of an SPVC or SPVP, use the following procedure.

Step 1 Establish a configuration session using a user name with GROUP1 privileges or higher.

Step 2 Define the slave side of the SPVC by entering the following command:

```
mgx8850a.10.AXSM.a > addcon <ifNum> <vpi> <vci> <serviceType> <mastership>
[-slave atmAddr.vpi.vci] [-lpcr <cellrate>] [-rpcr <cellrate>] [-lscr <cellrate>]
[-rscr <cellrate>] [-lmbs <cells>] [-rmbs <cells>] [-lcdv <time>] [-rcdv <time>]
[-lctd <time>] [-rctd <time>] [-lmcr <cellrate>] [-rmcr <cellrate>] [-cdvt <time>]
[-cc <1|0>] [-stat <1|0>] [-frame <1|0>] [-mc <maxCost>]
```

Caution Once you create an SPVC connection, you cannot change the SPVC prefix until all SPVC connections have been deleted. The procedure for changing the SPVC prefix is described in "Setting and Viewing the SPVC Prefix," in Chapter 2, "Configuring General Switch Features."

Table 4-9 lists and defines the parameters and options for the **addcon** command. The local and remote terms used in Table 4-9 refer to settings for the local port you are configuring and the remote port at the other end of the connection. If you omit an option, the SPVC uses the default value.

Note The options you set with the **addcon** command on the slave side must correspond to the options set with the **addcon** command on the master side, and they must be valid and appropriate for the lines at each end. For example, if you configure the local peak cell rate (lpcr) on the slave side, the value you enter must match the remote peak cell rate (rpcr) you enter when you configure the master side of the connection.

Table 4-9 Parameters for the addcon Command

Parameter	Description
ifNum	Enter the interface number (which is define with the **addport** command) for the port to which this SPVC will connect. The range is from 1 to 60.
vpi	Enter the VPI for the slave side of the SPVC. UNI Range: 0 to 255 NNI Range: 0 to 4095

Table 4-9 Parameters for the addcon Command (continued)

Parameter	Description
vci	Enter the VCI for the slave side of the SPVC or SPVP. SPVC Range: 32 to 65535 SPVP Range: 0 Note: Cisco Systems, Inc., recommends setting the minimum VCI to 35 or higher. Future products will use VCI 32 through 34 for other services.
serviceType	Replace with the number that corresponds to the requested service type for this SPVC (this value must be identical on master and slave sides): cbr1 = **1** cbr2 = **11** cbr3 = **12** vbr1rt = **2** vbr2rt = **3** vbr3rt = **4** vbr1nrt = **5** vbr2nrt = **6** vbr3nrt = **7** ubr1 = **8** ubr2 = **9** abrstd = **10**
mastership	Enter **2** or **s** if this port will serve as the slave side of the connection. Enter **1** or **m** if the port serves as the master side of the connection.
atmAddr.vpi.vci	This parameter is used only when defining the master side of a connection. The value entered here should match the NSAP displayed after the slave side of the connection is defined. The *atmAddr* portion of the address corresponds to the remote ATM address and the *vpi* and *vci* parameters correspond to the VPI and VCI settings for the slave. The periods between *atmAddrp* and *vpi* and between *vpi* and *vci* are required.
-lpcr> **-rpcr**	These options specify the local-to-remote (**-lpcr**) and remote-to-local (**-rpcr**) Peak Cell Rate (PCR) for the SPVC. Range cells per second: OC12: 7 to 1412830 OC3: 7 to 353207 T3: 7 to 96000(PLCP) or 104268(ADM) E3:b 7 to 80000
-lscr **-rscr**	These options specify the local-to-remote (**-lscr**) and remote-to-local (**-rscr**) Sustained Cell Rate (SCR) for the SPVC. Range cells per second: OC12: 7 to 1412830 OC3: 7 to 353207 T3: 7 to 96000(PLCP) or 104268(ADM) E3:b 7 to 80000
-lmbs **-rmbs**	These options specify the local-to-remote (**-lmbs**) and remote-to-local (**-rmbs**) Maximum Burst Size (MBS) for the SPVC. Range: 1 to 5000000 cells

Table 4-9 Parameters for the addcon Command (continued)

Parameter	Description
-lcdv **-rcdv**	These options specify the local-to-remote (**-lcdv**) and remote-to-local (**-rcdv**) Cell Delay Variation (CDV) for the SPVC. Range: 1 to 16777215 microseconds
-lctd **-rctd**	These options specify the local-to-remote (**-lctd**) and remote-to-local (**-rctd**) Cell Transfer Delay (CTD) for the SPVC. Range: 1 to 65535 milliseconds
-lmcr> **-rmcr**	These options specify the local-to-remote (**-lmcr**) and remote-to-local (**-rmcr**) Minimum Cell Rate (MCR) for the SPVC. Range cells per second: OC12: 7 to 1412830 OC3: 7 to 353207 T3: 7 to 96000(PLCP) or 104268(ADM) E3:b 7 to 80000
-cdvt	This option specifies the local Cell Delay Variation Tolerance (CDVT) for the SPVC. Range: 1 to 5000000 microseconds
-cc	This option enables or disables the flow of Operation, Administration, and Maintenance Continuity Check (OAMCC) traffic on the SVPC. Enter **1** to enable OAM traffic flow, or enter **0** to disable traffic flow. Note that when this option is enabled on only one side of a connection, a transient alarm is reported until this option is set to the same value at both ends.
-stat	This option enables or disables statistics collection for the SPVC. Enter **1** to enable OAM statistics collection, or enter **0** to disable it.
-frame	This option enables or disables frame discard. Enter **1** to enable frame discard, or enter **0** to disable it.
-mc	This option assigns a maximum cost value to the connection. When multiple connections are available during SPVC setup, the connection with the lowest cost value is selected. Range: 0 to 2147483647

Tips The PCR, MBS, CDVT, CDV, MCR, and CTD configuration options are optional. Use the *serviceType* parameter to select the SCT parameters for the type of traffic this connection will support. If all of the SCT file parameters are acceptable, you can omit the PCR, MBS, CDVT, CDV, and CTD configuration options. To override the SCT file settings for any option, enter the option with a new value. For example, if the PCR values in the SCT file are not acceptable, use the **-rpcr** and **-lpcr** options to override the SCT file values.

Note You can configure additional ABR parameters with the **cnfabr** and **cnfabrtparmdft** commands. For more information, refer to the *Cisco MGX 8850 Command Reference*.

The following command example defines a port as the slave side of an SPVC. Note the slave id shown in the command response.

```
pop20two.1.AXSM.a > addcon 3 101 101 1 2
slave endpoint added successfully
slave endpoint id : 4700918100000000001A531C2A00000101180300.101.101
```

Step 3 Write down the NSAP address the switch displays when the **addcon** command is complete. You will need this to configure the master side of the SPVC.

Step 4 Verify the slave-side SPVC addition by entering the following command:

```
pop2two.1.AXSM.a > dspcons
```

The switch displays a report similar to the following:

```
pop20two.1.AXSM.a > dspcons
record    Identifier   Type   SrvcType   M/S    Upld    Admn   Alarm
------    ----------   ----   --------   ---    ----    ----   -----
    0  03 0101 00101   VCC        cbr1   S   02022a26     UP     Condn
```

Configuring the Master Side of SPVCs and SPVPs

To configure the master side of an SPVC, use the following procedure.

Step 1 Establish a configuration session using a user name with GROUP1 privileges or higher.

Step 2 Use the following command to select the AXSM card that hosts the master side of the SPVC:

```
mgx8850a.7.PXM.a > cc <slotnumber>
```

Step 3 Define the master side of the SPVC by entering the following command:

```
mgx8850a.10.AXSM.a > addcon <ifNum> <vpi> <vci> <serviceType> <mastership>
[-slave atmAddr.vpi.vci] [-lpcr <cellrate>] [-rpcr <cellrate>] [-lscr <cellrate>]
[-rscr <cellrate>] [-lmbs <cells>] [-rmbs <cells>] [-cdvt <time>]
[-lcdv <time>] [-rcdv <time>] [-lctd <time>] [-rctd <time>]
[-cc <1|0>] [-stat <1|0>] [-frame <1|0>] [-mc <maxCost>]
```

Table 4-9 lists and defines the parameters and options for this command. If you omit an option, the SPVC uses the default value.

Tips The PCR, MBS, CDVT, CDV, and CTD configuration options are optional. Use the *serviceType* parameter to select the SCT parameters for the type of traffic this connection will support. If all of the SCT file parameters are acceptable, you can omit the PCR, MBS, CDVT, CDV, and CTD configuration options. To override the SCT file settings for any option, enter the option with a new value. For example, if the PCR values in the SCT file are not acceptable, use the **-rpcr** and **-lpcr** options to override the SCT file values.

The following command example defines a port as the master side of an SPVC. Note the master id shown in the command response.

```
pop20one.10.AXSM.a > addcon 3 101 101 1 1 -slave
4700918100000000001A531C2A00000101180300.101.101
master endpoint added successfully
master endpoint id : 4700918100000000107B65F33C0000010A180300.101.101
```

Step 4 Verify the master-side SPVC addition by entering the following command:

```
pop2one.10.AXSM.a > dspcons
```

The switch displays a report showing all connections. The following example show a report for a switch with one connection:

```
pop20one.10.AXSM.a > dspcons
record    Identifier   Type   SrvcType   M/S    Upld    Admn   Alarm
------    ----------   ----   --------   ---    ----    ----   -----
    0  03 0101 00101   VCC        cbr1   M   02022c36     UP      none
```

Step 5 To display the configuration for a single connection, enter the following command:

```
pop20two.9.AXSM.a > dspcon ifNum vpi vci
```

Replace the *ifNum* parameter with the interface or port number. The *vpi* and *vci* parameters are described in Table 4-9. The following example shows a **dspcon** command report.

```
pop20one.10.AXSM.a > dspcon 3 101 101
------------------------------------------------------------------------------
Local   :                NSAP  Address                     vpi       vci
(M)       47009181000000000107B65F33C0000010A180300        101       101
Remote  :                NSAP  Address                     vpi       vci
(S)       47009181000000000001A531C2A00000101180300        101       101
------------------------------------------------------------------------------
Conn. Type    :      VCC                        Admn Status  :  ADMN-UP
Service Type  :     cbr1                        Oper Status  :       OK
Controller    :        2                        Record #     :        0
------------------------------------------------------------------------------
Local PCR     :       50                        Remote PCR   :       50
Local SCR     :      N/A                        Remote SCR   :      N/A
Local CDV     :       -1                        Remote CDV   :       -1
Local CTD     :       -1                        Remote CTD   :       -1
Local MBS     :      N/A                        Remote MBS   :      N/A
Max Cost      :       -1                        Frame discard:        N
Local CDVT    :   250000
------------------------------------------------------------------------------
OAM CC Config : DISABLED                        Statistics   : DISABLED
------------------------------------------------------------------------------
Loopback Type : No  Lpbk | Dir: N/A     | Status: No Lpbk | RTD:      0us
------------------------------------------------------------------------------

Type <CR> to continue, Q<CR> to stop:

------------------------------------------------------------------------------
Port side Tx  :  normal                         Swth side Tx :  normal
Port side Rx  :  normal                         Swth side Rx :  normal
------------------------------------------------------------------------------
I-AIS/RDI   E-AIS/RDI   CONDITIONED   CCFAIL   IfFail   Mismatch  LMI-ABIT
   NO          NO           NO          NO       NO        NO         NO
------------------------------------------------------------------------------
```

The -1 entries in the example above indicate that a value was not specified with the **addcon** command. The N/A entries indicate that a value is not applicable to connections with this service type.

Step 6 To display connections from the PXM45 card, use the **cc** command to select the active PXM45, then enter the following command:

```
pop20two.7.PXM.a > dspcons
```

The following example shows the report for the connection shown in the preceding examples.

```
pop20two.7.PXM.a > dspcons

Local Port          Vpi.Vci    Remote Port         Vpi.Vci     State    Owner
--------------------------+------------------------------+-------+------
1:2.1:3            101 101     Routed             101 101     OK           SLAVE
Local  Addr: 47.00918100000000001a531c2a.000001011803.00
Remote Addr: 47.00918100000000107b65f33c.0000010a1803.00
```

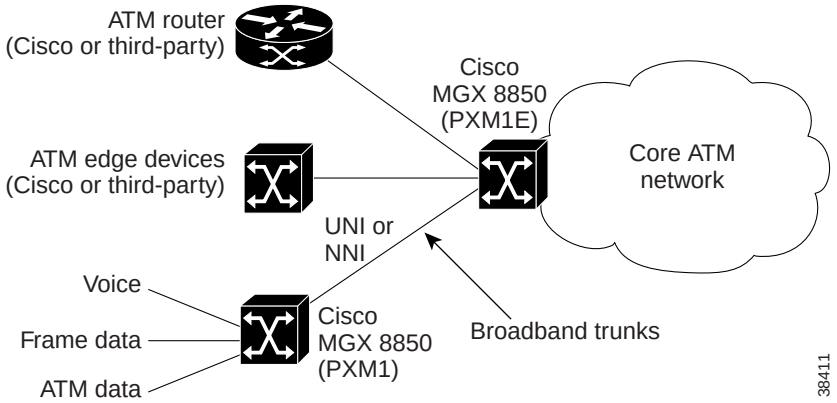
Defining a Feeder Port

An ATM feeder node provides a connection between multiple relatively slow lines (such as T1 lines) and a relatively faster uplink (such as an OC-3 line) to an ATM core network. Feeders such as the MGX 8850 Release 1 switch can concatenate traffic from Frame Relay, ATM, circuit emulation, and voice circuits for transmission over the core to other feeders or to Customer Premise Equipment (CPE).

Note In this guide, the term MGX 8850 switch refers to an MGX 8850, Release 2 switch. Whenever the discussion applies to the MGX 8850 Release 1 switch, the release number is included.

Figure 4-4 shows a topology that includes an MGX 8850 Release 1 feeder node.

Figure 4-4 Feeder Node Topology



In the configuration shown in Figure 4-4, the MGX 8850 switch supports up to 16 feeders. When using the MGX 8850 Release 1 switch as a feeder, you can route traffic to the core from the following MGX 8850 Release 1 service modules:

- AUSM
- CESM
- FRSM
- RPM
- VISM

The lower speed communication lines that connect to the feeder must exit the core network on lines that lead to another feeder or CPE. To enable communications between a feeder and a remote feeder or CPE, you need to configure an SPVC as described in "Configuring SPVCs and SPVPs," which appears earlier in this chapter. Table 4-10 identifies the supported interoperability between MGX 8850 Release 1 service modules over these AXSM SPVCs.

Table 4-10 Service Module Compatibility Between Feeders

Feeder A Service Module Type	MGX 8850 Service Module Type	Feeder B Service Module Type
FRSM	AXSM	FRSM
FRSM		AUSM
FRSM		RPM
AUSM		AUSM
AUSM		CESM
AUSM		VISM
AUSM		RPM
CESM		CESM
VISM		VISM
RPM		RPM

Note To operate properly, the MGX 8850 Release 1 feeder must be running compatible software. For information on the compatible feeder software for this release, refer to the *2.0.12 Version Software Release Notes, Cisco WAN MGX 8850 Software.*

The MGX 8850 switch uses the LMI Annex G protocol to communicate with the MGX 8850 Release 1 feeder node. When you define a feeder port, you instruct the switch to use this protocol to communicate with a feeder. The following procedure describes how to define a feeder port on the MGX 8850 switch.

Step 1 Establish a configuration session using a user name at any user level.

Step 2 To identify a port as a feeder port, enter the **addfdr** command as follows:

```
pop20one.10.AXSM.a > addfdr <ifNum>
```

Replace *ifNum* with the interface number for the port. For example:

```
pop20one.10.AXSM.a > addfdr 1
```

Tips The interface number is displayed in the **dspports** command report.

Note The **addfdr** command is blocked if other connections have been defined on the interface.

Step 3 To display the feeder ports configured on the AXSM card, use the **dspfdrs** command.

Step 4 To display information on a specific feeder port, enter the **dspfdr** <*ifnum*> command and replace *ifnum* with the interface number.

Note For more information on managing feeder node connections, see "Managing Feeder Connections" in Chapter 6, "Switch Operating Procedures."

After you configure a feeder connection, you can use the **dspcons** command to check for alarms on the feeder line. In the example below, the Abitfail alarm on connections 3 and 4 indicate a communication problem between the routing switch and the feeder node.

```
rtnode3.13.AXSM.a > dspcons
record    Identifier    Type    SrvcType    M/S    Upld    Admn    Alarm
------    ----------    ----    --------    ---    ----    ----    -----
    0   01.0001.00032   VCC         ubr1    M    00dfdfe9     UP    multiple
    1   01.0001.00033   VCC         ubr1    M    00de8ad8     UP    multiple
    2   01.0001.00041   VCC         cbr1    S    00dfb0d8     UP    Condn
    3   01.0001.00042   VCC         cbr1    S    00dfe281     UP    Abitfail
    4   01.0001.00043   VCC         cbr1    S    00dfe28a     UP    Abitfail
    5   01.0001.00052   VCC         ubr1    S    00e1244f     UP    multiple
```

Possible causes for the alarms shown above include:

- Disconnected or damaged line
- Feeder port not configured to communicate with routing switch
- Service module failure in feeder

Configuring AXSM Line Clock Sources

To configure the switch to receive a clock source on an AXSM line, you must do the following:

- Connect a line between the AXSM and the node with the clock source.
- Activate the line.
- Create a logical port (subport) for the clock signal.
- Create a resource partition.

Chapter 3, "Preparing AXSM Cards and Lines for Communication," describes how to activate a line. The procedures for creating ports and resource partitions appear earlier in this chapter. The following procedure describes how to configure an AXSM clock source after the line and port have been configured.

Step 1 Establish a configuration session using a user name with GROUP1 privileges or higher.

Step 2 To set a primary or secondary AXSM clock source, enter the following command:

```
mgx8850a.7.PXM.a > cnfclksrc <priority> [shelf.]slot:bay.line:ifnum
```

Table 4-11 describes the parameters for this command.

Tips To get the correct *slot*:*bay*.*line*:*ifnum* specification, use the port ID displayed by the **dsppnports** command.

Table 4-11 Parameter Descriptions for cnfclksrc Command when Used for AXSM

Parameter	Values	Descriptions
priority	primary or secondary	Replace *priority* with the type of clock source, which is either **primary** or **secondary**. The default is **primary**.
shelf	1	The *shelf* value is always 1, and it is optional.
slot	1-6, 9-14	Enter the slot number of the AXSM card that is receiving the clock signal.
bay	1 or 2	The *bay* identifies the bay in which the back card is installed. If the clock source line is connected to upper card, enter **1**. If it is connected to the lower card, enter **2**. The default is **1**.
line	1-8	The *line* number corresponds to the line number on the back card. The line must already be active (using **upln**).
ifnum	1-60	The *ifnum* number corresponds to the interface number or logical port number, which is between 1 and 60. The interface number must have been previously defined using the **addport** command.

Step 3 To configure an additional clock source, repeat Step 2 using the correct parameters for the additional source.

The following command example shows how to configure a secondary clock source for subport (logical port) 10 on line 1 of the AXSM card in the upper bay of slot 3. Note the placement of the periods and colons.

```
mgx8850a.7.PXM.a > cnfclksrc secondary 3:1.1:10
```