Configuring Digital T1 Packet Voice Trunk Network Modules on Cisco 2600 and 3600 Series Routers

This document describes how to configure digital T1 packet voice trunk network modules on Cisco 2600 and 3600 routers and includes the following sections:

Feature Overview

Digital T1 packet voice trunk network modules for Cisco 2600 and 3600 series routers allow enterprises or service providers, using the equipped routers as customer premises equipment, to deploy digital voice and fax relay. These modules receive constant bit-rate telephony information over T1 interfaces and can convert that information to a compressed format, so that it can be transmitted as voice over IP.

Cisco IOS software configuration allows you to set up a variety of applications. Here are a few examples:

- Compressed voice over WANs
- Routing of dialed variable-length digits collected from the public switched telephone network or PBX for VoIP calls
- Support for FRF.12 fragmentation and queuing in a VoIP over Frame-Relay network
- Setup of private-line auto-ringdown (PLAR) to allow a station or DS0 to go off hook and have the call completed without dialing (especially applicable to off-premises extensions)
- Transparent trunk connections among routers
- Drop and Insert of T1 channels on a T1 trunk to allow some PBX channels to be directed to the PSTN while others are used for compressed VoIP

For more information about these applications, see "Configuration Examples" on page 33.

T1 digital voice over IP includes the following functionality:

- T1 Channel Associated Signaling (CAS) for the following line-signaling types:
  - rEceive and transMit or Ear and Mouth (E&M) immediate start
  - E&M wink start
  - E&M delay start (also called "dial repeating")
  - Foreign Exchange Station (FXS) and Foreign Exchange Office (FXO) loop start
  - FXS and FXO ground start
- Dynamic bandwidth allocation using voice activity detection (VAD)
- Drop-and-Insert capability, allowing the interchange of time-division multiplexing (TDM) slots between the ports on a two-port T1 multiflex trunk voice/WAN interface card installed in a digital T1 packet voice trunk network module
- Support for a wide range of International Telecommunication Union (ITU-T) G-series compression specifications, including:
  - G.711 A Law at 64,000 bps
  - G.711 u Law at 64,000 bps
  - G.723.1 Annex A at 5,300 bps
  - G.723.1 Annex A at 6,300 bps
  - G.723.1 at 5,300 bps
  - G.723.1 at 6,300 bps
  - G.726 at 16,000 bps
  - G.726 at 24,000 bps
  - G.726 at 32,000 bps
  - G.728 at 16,000 bps
  - G.729 at 8,000 bps
  - G.729 Annex A at 8,000 bps
  - G.729 Annex B at 8,000 bps
  - G.729 Annex B with Annex A at 8,000 bps
- Depending on codec complexity, either 30 or 60 channels of compressed voice
- High-quality voice endpoint-standard features, such as high-quality echo cancellation, silence suppression, comfort noise generation, and DTMF relay
- Group 3 fax relay
- Support for the following framing formats and line coding:
  - Super Frame (SF)
  - Extended Super Frame (ESF)
  - Alternate mark inversion (AMI) line coding
  - Binary 8-zero substitution (B8ZS) line coding

Benefits

Digital T1 packet voice trunk network modules allow Cisco 2600 and 3600 series routers to provide T1 connectivity to PBXs or to a central office (CO). With digital T1 connectivity, Cisco 2600 and 3600 series routers can provide greater voice density for enterprise and service provider VoIP networks than they could before. A digital T1 packet voice trunk network module is a complete solution, made up of a network module with installed packet voice data modules (PVDMs), and one T1 multiflex trunk voice/WAN interface card with either one or two T1 ports.

VoIP: T1 Timing, Signaling, Framing, and Line Encoding

With the introduction of the digital T1 packet voice trunk network modules for the Cisco 2600 and 3600 series routers, you must set timing, signaling, framing, and line encoding. The digital T1 packet voice trunk network modules can connect to either a PBX (or similar telephony device) or to a Central Office (CO) in order provide PSTN connectivity.

The differences that set T1 digital configuration apart from analog configuration are as follows:

- **Timing.** Analog interfaces do not require specific timing configuration. Digital T1 interfaces require not only that you set timing but that you consider the source of the timers.
- **Framing.** Analog interfaces do not require specific framing configuration. Digital T1 interfaces require that you configure either SuperFrame (SF or D4 framing) or Extended SuperFrame (ESF) framing. Set the framing format to match that of the PBX or CO that connects to the digital T1 packet voice trunk network module.
- **Line Encoding.** Analog interfaces do not require that specific line encoding be configured. Digital T1 require that you configure either AMI (alternative mark inversion) or B8ZS (bipolar 8- zero substitution). Set the line encoding to match that of the PBX or CO that connects to the digital T1 packet voice trunk network module.

Timing

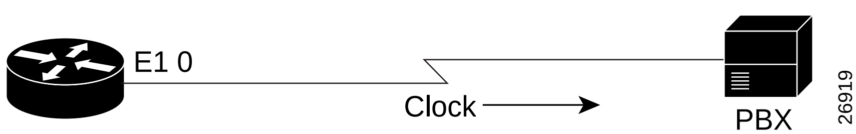
This section describes the five basic timing scenarios that can occur when a digital T1 packet voice trunk network module is connected to a PBX, CO, or both. In all of the examples below, the PSTN (or Central Office) and the PBX are interchangeable for the purposes of providing or receiving clocking.

The digital T1 module has an on-board PLL (Phase-Lock Loop) chip that can either provide a clock source to both T1s or receive clocking that can drive the second T1. All timing commands are T1 controller configuration commands.

Single T1 Port Provides Clocking

In this scenario, the digital T1 module is the clock source for the connected device. The PLL generates the clock internally and drives the clocking on the T1 line.

Figure 1 **Single T1 Port Providing Clock**



The following configuration sets up this clocking method:

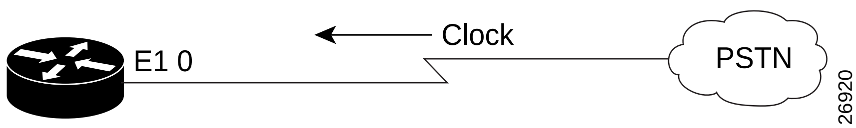
```
controller T1 1/0
framing esf
linecoding b8zs
clock source internal
ds0-group 1 timeslots 1-24 type e&m-wink
```

Note Generally this method is useful only when connecting to a PBX, key system or channel bank. A Cisco VoIP Gateway rarely provides clocking to the CO, because CO clocking provides a higher Stratum level.

Single T1 Port Receiving Clock from the Line

In this scenario, the digital T1 module receives clocking from the connected device (CO or PBX). The PLL clocking is driven by the clock reference on the receive (*Rx*) side of the T1 connection.

Figure 2 Single T1 Receiving Clock from Line



The following configuration sets up this clocking method:

```
controller T1 1/0
framing esf
linecoding b8zs
clock source line
ds0-group 1 timeslots 1-24 type e&m-wink
```

Dual T1s, Both Receive Clocking from the Line

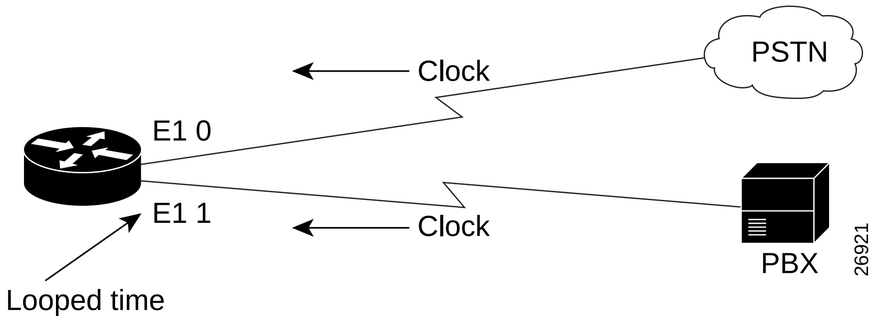
In this scenario, the digital T1 has two reference clocks, one from the PBX and another from the CO. Since the PLL can only derive clocking from one source, this case is more complex than the two preceding examples.

Before looking at the details, consider two important concepts that underlay the clocking method:

- **Looped-Time Clocking.** The T1 port takes the clock received on its *R*x (receive) pair and regenerates it on its *Tx* (transmit) pair. While the port receives clocking, the port is not driving the PLL on the card but is “spoofing” the T1 so that the connected device has a viable clock and does not see slips. PBXs are not designed to accept slips on a T1 line and such slips cause a PBX to drop the link into failure mode. While in looped-time mode, the router often sees slips, but because these are controlled slips, they usually do not force failures of the router’s T1 port.
- **Slips.** These messages indicate that the T1 port is receiving clock information that is out of phase, that is, out of synch. Because the router has only a single PLL, it can experience controlled slips while it receives clocking from two different time sources. The router can usually handle controlled slips because its single PLL architecture anticipates them.

Note Physical layer issues, such as bad cabling or faulty clocking references, can also cause slips. Eliminate these slips by addressing the physical layer or clock reference problems.

Figure 3 Dual T1s Receiving Line Clocking



In this scenario, the PLL derives clocking from the CO and puts the T1 port connected to the PBX into looped-time mode. This is usually the best method because the CO provides an excellent clock source (and usually requires that it provide that source) and a PBX usually must receive clocking from the other T1.

The following configuration sets up this clocking method:

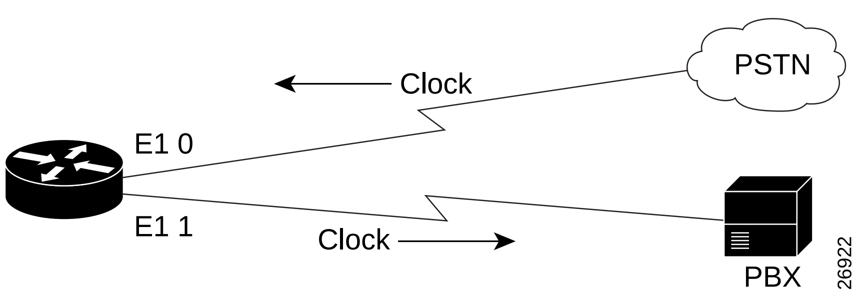
```
controller T1 1/0 << description - connected to the CO
framing esf
linecoding b8zs
clock source line primary
ds0-group 1 timeslots 1-24 type e&m-wink
!
controller T1 1/1 << description - connected to the PBX
framing esf
linecoding b8zs
clock source line
ds0-group 1 timeslots 1-24 type e&m-wink
```

The **clock source line primary** command tells the router to use this T1 port to drive the PLL. All other T1 ports configured as **clock source line** are then put into an implicit loop-timed mode. If the primary T1 port fails or goes down, the other T1 instead receives the clock that drives the PLL. In this configuration, T1 1/1 may see controlled slips, but these should not force it down. This method prevents the PBX from seeing slips.

Dual T1s, One Receives Clocking and One Provides Clocking

In this scenario, the digital T1 module receives clocking for the PLL from T1 0 and uses this clock as a reference to clock T1 1. If T1 0 fails, the PLL internally generates the clock reference to drive T1 1.

Figure 4 Dual T1s, One Receiving and One Providing Clocking



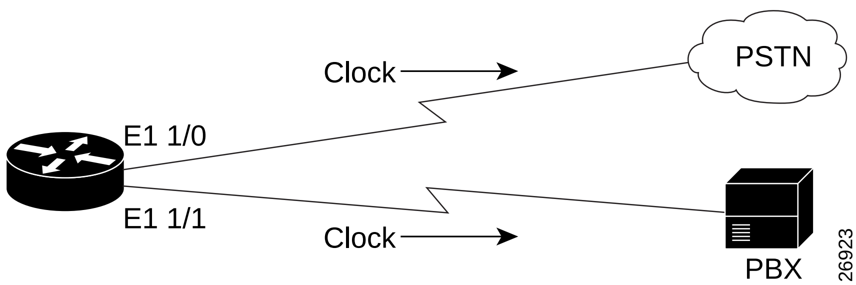
The following configuration sets up this clocking method:

```
controller T1 1/0
framing esf
linecoding b8zs
clock source line
ds0-group 1 timeslots 1-24 type e&m-wink
!
controller T1 1/1
framing esf
linecoding b8zs
clock source internal
ds0-group 1 timeslots 1-24 type e&m-wink
```

Dual T1s, Both Clocks from Router

In this scenario, the router is "Master of the Timing Universe," generating the clock for the PLL and therefore for both T1s.

Figure 5 Dual T1s, Both Clocks from Router



The following configuration sets up this clocking method:

```
controller T1 1/0
framing esf
linecoding b8sz
clock source internal
ds0-group 1 timeslots 1-24 type e&m-wink
!
controller T1 1/1
framing esf
linecoding b8zs
clock source internal
ds0-group 1 timeslots 1-24 type e&m-wink
```

Signaling

There are three types of signaling that you should consider for digital T1:

- **Channel-Associated Signaling (CAS).** CAS signaling means that instead of having a specific time slot (such as an ISDN D channel in PRI) designated to provide signaling only, signaling bits (on-hook and off-hook) are within the sixth, twelfth, eighteenth and twenty-fourth frames of each time slot. CAS signaling is often called robbed-bit signaling (RBS) because it takes bits from bearer channels and uses them for signaling. CAS signaling must be specified on both ends of the T1 link and is enabled by default on digital T1 packet voice trunk network modules.

  Note Digital T1 packet voice trunk network modules support T1 CAS at this time, but later plans are to support E1, Primary Rate Interface (PRI), R2, and Common-Channel (CCS) signaling. The digital T1 module can support E&M wink-start, immediate-start, and delay-start signaling, as well as FXS and FXO ground-start and loop-start signaling.

- **E&M Signaling.** E&M connections can use one of three different signaling types to acknowledge on-hook and off-hook states: wink-start, immediate-start and delay-start. E&M wink-start is usually preferred because it provides better Answer Supervision (knowledge that the connected device is ready to answer the call). However, not all COs and PBXs can handle wink-start signaling. The E&M connection between the router and switch (CO or PBX) must use matching E&M signaling types or calls are not be connected properly. E&M signaling is defined with the **ds0-group** controller configuration command, as in the following example:

  ```
  controller T1 1/0
  ds0-group 1 timeslots 1-24 type e&m-wink-start
  ```

  Note Currently, wink-start signaling provides only the functionality of Feature-Group B and not that of Feature-Group D, which will be supported in later releases.

- **FXO and FXS Signaling.** While most digital T1 connections used for switch-to-switch (or switch-to-router) trunks are E&M connections, a digital T1 module can also support FXS and FXO connections, which people normally use to provide emulated-OPX (Off-Premise eXtensions) from a PBX to remote stations. As a general rule, FXO ports connect to FXS ports. Either ground-start or loop-start signaling is appropriate for these connections. Ground-start provides better Disconnect Supervision to detect when a remote user has hung up the phone, but ground-start is not available on all PBXs. The FXO or FXS connection between the router and switch (CO or PBX) must use matching signaling or calls are not be connected properly. FXS and FXO signaling are defined with the **ds0-group** controller configuration command, as in the following example:

  ```
  controller T1 1/0
  ds0-group 1 timeslots 1-24 type fxo-ground-start
  ```

  or

  ```
  controller T1 1/0
  ds0-group 1 timeslots 1-24 type fxs-loop-start
  ```

Note While some switches (CO or PBX) can specify both an inbound and outbound signaling method, Cisco VoIP gateway routers can only specify one signaling type for both inbound and outbound calls. The switch inbound and outbound signaling types must match, or calls may only work in one direction.

Framing

Digital T1 packet voice trunk network modules support two types of framing for T1 CAS: ESF (Extended SuperFrame) and SF (SuperFrame), also called D4 framing. The framing type of the router and switch (CO or PBX) must match. The **framing** controller configuration command defines T1 framing, as in the following example:

```
controller T1 1/0
framing esf
```

or

```
controller T1 1/0
framing sf
```

Line Encoding

Digital T1 packet voice trunk network modules support two types of framing for T1 CAS: B8ZS (bipolar-8 zero substitution) and AMI (alternate mark inversion). The line encoding of the router and switch (CO or PBX) must match. The **linecoding** controller configuration command defines T1 framing, as in the following example:

```
controller T1 1/0
linecoding b8zs
```

or

```
controller T1 1/0
linecoding ami
```

Verifying Configuration

Use the **show controller** privileged EXEC command to verify the proper digital T1 configuration:

```
router# show controller T1 1/0
T1 1/0 is up.
  Applique type is Channelized T1
  Cablelength is short 133
  Description: Digital T1 WIC
  No alarms detected.
  Framing is ESF, Line Code is B8ZS, Clock Source is Line Primary.
  Data in current interval (2 seconds elapsed):
     0 Line Code Violations, 0 Path Code Violations
     0 Slip Secs, 0 Fr Loss Secs, 0 Line Err Secs, 0 Degraded Mins
     0 Errored Secs, 0 Bursty Err Secs, 0 Severely Err Secs, 0 Unavail Secs
```

Restrictions

The following restrictions apply to digital T1 packet voice trunk network module configuration:

- Group 4 fax is not supported.
- The high-density voice network module has one slot for a voice/WAN interface card (VWIC); VWICs supply one or two ports. Only the dual-mode (voice/WAN) multiflex trunk cards are supported in the digital T1 packet voice trunk network module, not older VICs. For more information, see the “Prerequisites” section on page 11.
- Drop-and-Insert capability is supported only between two ports on the same multiflex card.
- Voice over Frame Relay is not supported.
- Wink-start signaling Feature-Group D is not supported.

- Common-channel signaling (CCS) and Primary Rate Interface (PRI) are not supported.
- R2 signaling is not supported.
- Voice over ATM—including AAL5 encapsulation, circuit emulation service (CES), and AAL2—is not supported for VoATM.
- Digital T1 voice is not manageable through Simple Network Management Protocol (SNMP) using existing versions of Cisco Voice Manager. Release 2.0 of Cisco Voice Manager is planned to support the feature.

Related Features and Technologies

VoIP Quality of Service

This section explains the quality issues that you should consider when building Voice over IP (VoIP) networks and offers a few tips about configuring VoIP with the appropriate Quality of Service (QoS):

- **Delay.** Delay is the time it takes for VoIP packets to travel between two endpoints and you should design networks to minimize this delay. However, because of the speed of network links and the processing power of intermediate devices, some delay is expected. The human ear normally accepts up to about 150 milliseconds (ms) of delay without noticing problems (the ITU's G.114 standard recommends no more than 150 ms of one-way delay). Once delay exceeds 150 ms, a conversation becomes more and more like a walkie-talkie interchange, where one person must wait for the other to stop speaking before beginning to talk. This type of delay is often evident on international long-distance calls. You can measure delay fairly easily by using ping tests at various times of the day with different network traffic loads. If network delay is excessive, reduce it before deploying VoIP networks.
- **Jitter.** While delay can cause unnatural starting and stopping of conversations, variable-length delays (also known as *jitter*) can cause a conversation to break and become unintelligible. Jitter is not usually a problem with public switched telephone network (PSTN) calls, because the bandwidth of calls is fixed. However, in VoIP networks where existing data traffic might be bursty, jitter can become an issue. Cisco voice gateways have built-in de-jitter buffering to compensate for a certain amount of jitter, but if jitter is constant on a network, identify the source and control it before deploying a VoIP network.
- **Serialization.** Serialization is a term that describes what happens when a router attempts to send both voice and data packets out of an interface. In general, voice packets are very small (80 to 256 bytes), while data packets can be very large (1,500 to 18,000 bytes). On relatively slow links, such as WAN connections, large data packets can take a long time to transmit onto the wire. When these large packets are mixed with smaller voice packets, the excessive transmission time can lead to both delay and jitter. You can use fragmentation to reduce the size of the data packets so that the delay and jitter also decrease.
- **Bandwidth Consumption.** Traditional voice conversations consume 64 Kb of network bandwidth. When this voice traffic is run though a VoIP network, it can be compressed and digitized by digital signal processors (DSPs) built into the routers. This compression can reduce the calls to sizes as small as 5.3 Kb for voice samples. Once the packets go onto the IP network, the appropriate IP/UDP/RTP headers must be added, and this can add a significant amount of bandwidth to each call (about 40 bytes per packet). Technologies such as RTP header compression, however, can reduce the IP header overhead to about 2 bytes. In addition, VAD (voice activity detection) does not send any packets unless there is active speech.

Supported Platforms

This feature is supported on the following platforms:

- Cisco 2610
- Cisco 2611
- Cisco 2612
- Cisco 2613
- Cisco 2620
- Cisco 2621
- Cisco 3620
- Cisco 3640
- Cisco 3661
- Cisco 3662

Supported Standards, MIBs, and RFCs

RFCs

- RFC 1890
- RFC 1889

MIBs

- CISCO-ENTITY-VENDORTYPE-OID-MIB
- OLD-CISCO-CHASSIS-MIB
- CAS_INTF_MIB

International Telecommunication Union (ITU-T) G-Series Codec Compression Specifications

- G.711 A Law at 64,000 bps
- G.711 u Law at 64,000 bps
- G.723.1 Annex A at 5,300 bps
- G.723.1 Annex A at 6,300 bps
- G.723.1 at 5,300 bps
- G.723.1 at 6,300 bps
- G.726 at 16,000 bps
- G.726 at 24,000 bps
- G.726 at 32,000 bps
- G.728 at 16,000 bps
- G.729 at 8,000 bps

- G.729 Annex A at 8,000 bps
- G.729 Annex B at 8,000 bps
- G.729 Annex B with Annex A at 8,000 bps

Prerequisites

Digital T1 packet voice requires specific service, software, and hardware:

- Obtain T1 service from your service provider or PBX.
- Install Cisco IOS Software Release 12.0(5)XK, 12.0(7)T or a later release. The *minimum* DRAM memory requirements to support digital T1 packet voice trunk network modules are as follows:
  - 32 Mb with one or two T1s
  - 48 Mb with three or four T1s
  - 64 Mb with five to ten T1s
  - 128 Mb with more than ten T1s

  The memory required may be greater than listed above for high-volume applications.

  Support for digital T1 packet voice trunk network modules is included in Plus feature sets. The IP Plus feature set requires 8 Mb of flash memory; other Plus feature sets require 16 Mb.
- Install one of the following high-density T1 network modules in the router chassis:
  - Single-Port 24 Channel T1 High-Density Voice Network Module (NM-HDV-1T1-24)
  - Single-Port Enhanced 24 Channel T1 High-Density Voice Network Module (NM-HDV-1T1-24E)
  - Dual-Port 48 Channel High-Density Voice Network Module (NM-HDV-2T1-48)

Note You can install one module in a Cisco 2600 series router or a Cisco 3620 router. A Cisco 3640 router can support three modules, and you can install as many as six modules in a Cisco 3660 router.

- Install at least one packet voice data module (PVDM-12) in the high-density digital T1 network module if it is not already equipped. The digital T1 packet voice trunk network module contains five 72-pin SIMM sockets or banks, numbered 0 through 4, for PVDMs. Each socket can be filled with a single 72-pin PVDM. A digital T1 packet voice trunk network module can support the following numbers of channels:
  - When the digital T1 packet voice trunk network module is configured for high-complexity codec mode, up to six voice or fax calls can be completed per PVDM-12, using the following codecs: G.711, G.726, G.729, G.729 Annex B, G.723.1, G.723.1 Annex A, G.728, and fax relay.
  - When the digital T1 packet voice trunk network module is configured for medium-complexity codec mode, up to twelve voice or fax calls can be completed per PVDM-12, using the following codecs: G.711, G.726, G.729 Annex A, G.729 Annex B with Annex A, and fax relay.

Note Each PVDM holds three digital signal processors (DSPs). With five PVDM slots populated, a total of 15 DSPs are provided. High-complexity codecs support two simultaneous calls on each DSP, while medium-complexity codecs support four calls on each DSP.

- Install at least one dual-mode voice/WAN interface card (VWIC) for a voice connection if a VWIC was not included with the network module. You can install one VWIC (providing one or two line interfaces) in the digital T1 packet voice trunk network module. Only the one- and two-port T1 multiflex trunk interface cards (VWIC-1MFT-T1, VWIC-2MFT-T1, VWIC-2MFT-T1-DI) are supported with channel-associated signaling (CAS).

  For Drop-and-Insert capability, you must install a two-port Drop-and-Insert T1 multiflex trunk voice/WAN interface card (VWIC-2MFT-T1-DI). To install a VWIC in a network module, see *Cisco WAN Interface Cards Hardware Installation Guide*.
- Install at least one other network module or WAN interface card to provide the connection to the IP LAN or WAN.
- Establish a working IP network. For more information about configuring IP, see “IP Overview,” “Configuring IP Addressing,” and “Configuring IP Services” chapters in the Cisco IOS Release 12.0 *Network Protocols Configuration Guide, Part 1*.
- Complete your company’s dial plan.
- Establish a working telephony network based on your company's dial plan.

Voice, Video, and Home Applications Configuration Guide and *Voice, Video, and Home Applications Command Reference* for Cisco IOS Release 12.0 provide information about setting up voice networks.

Configuration Tasks

Perform the following tasks to configure a digital T1 packet voice trunk network module:

- Set up voice cards and T1 controllers.
- Configure serial and LAN interfaces.
- Set up voice ports.
- Configure voice dial peers.

Configuring Voice Card and T1 Controller Settings

The following steps specify codec settings for voice cards and set up T1 controllers for clocking and other T1 parameters, as well as for DS0 groups that define the channels for compressed voice and TDM groups for Drop-and-Insert capability.

Step	Command	Purpose
1	`Router#` **`configure terminal`**	Enter global configuration mode.
2	`Router(config)#` **`voice-card`** *`slot`*	Enter voice card interface configuration mode and specify the slot location by using a value from 0 to 5, depending upon your router.
3	`Router(config-voice-ca)#` **`codec complexity`** {**`high`** | **`medium`**}	Specify the codec complexity based on the codec standard you are using. High-complexity codecs support lower call density than do medium-complexity codecs. The number of channels supported is based on the number of PVDMs installed and the codec complexity. Here is a guideline: • When the digital T1 packet voice trunk network module is configured for high-complexity codec mode, up to six voice or fax calls can be completed per PVDM-12, using the following codecs: G.711, G.726, G.729, G.729 Annex B, G.723.1, G.723.1 Annex A, G.728, and fax relay. • When the digital T1 packet voice trunk network module is configured for medium-complexity codec mode, up to twelve voice or fax calls can be completed per PVDM-12, using the following codecs: G.711, G.726, G.729 Annex A, G.729 Annex B with Annex A, and fax relay All voice cards in a router must use the same codec complexity setting. The keyword that you specify for **codec complexity** affects the choice of codecs available using the **codec** dial-peer configuration command. See Step 7 in "Configuring Voice Dial Peers" on page 22. **Note** You cannot change codec complexity while DS0 groups are defined. If they are already set up, use the **no ds0-group** command before resetting the codec complexity. For more information about the **ds0-group** command, see Step 9.
4	`Router(config)#` **`controller T1`** *`slot/port`*	Enter controller configuration mode for the T1 controller at the specified *slot/port* location. Valid values for *slot* and *port* are 0 and 1.

Step	Command	Purpose
5	`Router(config-controller)#` **`clock source {line [primary] | internal}`**	Configure controller T1 1/0 to specify the clock source. The **line** keyword specifies that the clock source is derived from the active line—rather than from the free-running internal clock. This is the default setting and is generally more reliable. These rules apply to clock sourcing on the T1 controller ports: • When both ports are set to line clocking with no primary specification, port 0 is the default primary clock source and port 1 is the default secondary clock source. • When both ports are set to line and one port is set as the primary clock source, the other port is by default the backup or secondary source and is loop-timed. • If one port is set to **clock source line** or **clock source line primary** and the other is set to **clock source internal**, the internal port recovers clock from the clock source line port if the clock source line port is up. If it is down, then the internal port generates its own clock. • If both ports are set to **clock source internal**, there is only one clock source—internal.
6	`Router(config-controller)#` **`framing {sf | esf}`**	Set the framing according to your service provider's instructions. Choose Extended Superframe (ESF) format or Superframe (SF) format.
7	`Router(config-controller)#` **`linecode {b8zs | ami}`**	Set the line encoding according to your service provider's instructions. Bipolar-8 zero substitution (B8ZS) encodes a sequence of eight zeros in a unique binary sequence to detect line coding violations. Alternate mark inversion (AMI) represents zeros using a 01 during each bit cell, and ones are represented by 11 or 00, alternately, during each bit cell. AMI requires that the sending device maintain ones density. Ones density is not maintained independent of the data stream.

Step	Command	Purpose
8	`Router(config-controller)#` **`cablelength long {gain26	gain36} {-15db	-22.5db	-7.5db	0db}`**	(T1 interfaces only) The cable length setting must conform to the actual cable length you are using. For example, if you attempt to enter the **cablelength short** command on a long-haul T1 link, the command is rejected. To set a cable length longer than 655 feet for a T1 link, use the **cablelength long** command. The keywords are as follows: • **gain26** specifies the decibel pulse gain at 26. This is the default pulse gain. • **gain36** specifies the decibel pulse gain at 36. • **-15db** specifies the decibel pulse rate at -15 decibels. • **-22.5db** specifies the decibel pulse rate at -22.5 decibels. • **-7.5db** specifies the decibel pulse rate at -7.5 decibels. • **0db** specifies the decibel pulse rate at 0 decibels. This is the default pulse rate.
	or **`cablelength short {133	266	399	533	655}`**	To set a cable length 655 feet or less for a T1 link, use the **cablelength short** command. There is no default for **cablelength short**. The keywords are as follows: • **133** specifies a cable length from 0-133 feet. • **266** specifies a cable length from 134-266 feet. • **399** specifies a cable length from 267-399 feet. • **533** specifies a cable length from 400-533 feet. • **655** specifies a cable length from 534-655 feet. If you do not set the cable length, the system defaults to a setting of **cablelength long gain26 0db**.

Step	Command	Purpose
9	`Router(config-controller)#` **`ds0-group`** *`ds0-group-no`* **`timeslots`** *`timeslot-list`* **`type`** {**`e&m-immediate`** | **`e&m-delay`** | **`e&m-wink`** | **`fxs-ground-start`** | **`fxs-loop-start`** | **`fxo-ground-start`** | **`fxo-loop-start`**}	This command defines the T1 channels for use by compressed voice calls as well as the signaling method the router uses to connect to the PBX or CO. You should set up DS0 groups after you have specified codec complexity in voice-card configuration, as shown in Step 3. If you modify the **codec complexity** command parameters, you must first remove any existing DS0 groups, then reinstate them after the change to the codec complexity. *ds0-group-no* is a value from 0 to 23 that identifies the DS0 group. **Note** The **ds0-group** command automatically creates a logical voice port that is numbered as follows: *slot*/*port*:*ds0-group-no*. Although only one voice port is created, applicable calls are routed to any channel in the group. *timeslot-list* is a single number, numbers separated by commas, or a pair of numbers separated by a hyphen to indicate a range of timeslots. For T1, allowable values are from 1 to 24. To map individual DS0 timeslots, define additional groups. The system maps additional voice ports for each defined group. See Step 2 of "Configuring Voice Ports" on page 20. The signaling method selection for **type** depends on the connection that you are making: • The E&M interface allows connection for PBX trunk lines (tie lines) and telephone equipment. The wink and delay settings both specify confirming signals between the transmitting and receiving ends, whereas the immediate setting stipulates no special offhook/onhook signals. • The FXO interface is for connection of a central office (CO) to a standard PBX interface where permitted by local regulations; the interface is often used for off-premises extensions. • The FXS interface allows connection of basic telephone equipment and PBXs.

Step	Command	Purpose
10	`Router(config-controller)#` **`tdm-group`** *`tdm-group-no`* **`timeslots`** *`timeslot-list`* **`type`** `[`**`e&m`** `|` **`fxs`** `[`**`loop-start`** `|` **`ground-start`**`]` **`fxo`** `[`**`loop-start`** `|` **`ground-start`**`]]`	(Optional) Use this command only when you need TDM channel groups for the Drop-and-Insert (also called TDM Cross-Connect) function with a two-port T1 multiflex trunk interface card. *tdm-group-no* is a value from 0 to 23 that identifies the channel group. *timeslot-list* is a single number, numbers separated by commas, or a pair of numbers separated by a hyphen to indicate a range of timeslots. For T1, allowable values are from 1 to 24. The signaling method selection for **type** depends on the connection that you are making. The **fxs** and **fxo** options allow you to specify a ground-start or loop-start line. Choose a type based on the criteria described above in Step 9. **Note** The group numbers for controller groups must be unique. For example, a TDM group should not have the same ID number as a DS0 group.
11	`Router(config-controller)#` **`no shutdown`**	Activate the controller.
12	`Router(config-controller)#` **`exit`**	Exit controller configuration mode. Skip the next step if you are not setting up Drop and Insert.
13	`Router(config)#` **`connect`** *`id`* **`T1`** *`slot/port tdm-group-no-1`* **`T1`** *`slot/port tdm-group-no-2`*	(Optional) This global configuration command sets up the connection between two T1 TDM groups of timeslots on the trunk interfaces—for Drop and Insert. *id* is a name for the connection. Identify each T1 controller by its *slot/port* location. Valid values for *slot* and *port* are 0 and 1. *tdm-group-no-1* and *tdm-group-no-2* identify the TDM group numbers (from 0 to 23) on the specified controller. The groups were set up in Step 10. See the “Configuration Examples” section on page 33 for sample Drop and Insert configurations.

Repeat Steps 2 and 3 for each voice card.

Repeat Steps 4 through 12 for each controller.

Verifying Voice Card and Controller Settings

To verify the configuration of voice card and controller settings, follow these steps:

Step 1 Enter the **show running-config** command to display the current voice-card setting. If no codec complexity is shown, the default of medium complexity is set. The following example shows an excerpt from the command output:

```
Router# show running-config
.
.
.
hostname router-alpha

voice-card 1
 codec complexity high
.
.
.
```

Step 2 The privileged EXEC **show controllers t1** command displays the status of T1 controllers and displays information about clock sources and other settings for the T1 ports:

```
Router# show controller T1 1/0

T1 1/0 is up.
  Applique type is Channelized T1
  Cablelength is short 133
  Description: T1 WIC card Alpha
  No alarms detected.
  Framing is ESF, Line Code is B8ZS, Clock Source is Line Primary.
  Data in current interval (1 seconds elapsed):
     0 Line Code Violations, 0 Path Code Violations
     0 Slip Secs, 0 Fr Loss Secs, 0 Line Err Secs, 0 Degraded Mins
     0 Errored Secs, 0 Bursty Err Secs, 0 Severely Err Secs, 0 Unavail Secs
```

Step 3 The privileged EXEC **show connection all** command displays the status of T1 or E1 TDM controller groups and how they are set up:

```
Router# show connection all

ID   Name               Segment 1            Segment 2           State
=========================================================================
1    Test              -T1 1/0 01           -T1 1/1 02           ADMIN UP
```

Configuring Serial Interfaces

The way you set up serial and LAN interfaces depends on your application. To configure VoIP, you must at least set up IP addresses for serial interfaces. When a user dials enough digits to match a configured destination pattern, the telephone number is mapped to an IP host through the dial plan mapper. The IP host has a direct connection to either the destination telephone number or a PBX that completes the call to the configured destination pattern.

This document does not explain all possible serial interface configuration options, nor does it show LAN interface configuration. For complete information, see the Cisco IOS Release 12.0 *Cisco IOS Interface Configuration Guide* and the *Cisco IOS Interface Command Reference*.

The "Configuration Examples" section on page 33 shows a sample configuration that sets up VoIP over Frame Relay. For more information about setting up voice networks, see *Voice, Video, and Home Applications Configuration Guide* for Cisco IOS Release 12.0.

Note For information about monitoring serial interfaces in order to trigger a busyout condition on a voice port when an interface is down, see “Configuring Voice Ports” on page 20.

Step	Command	Purpose
1	`Router#` **`configure terminal`**	Enter global configuration mode.
2	`Router(config)#` **`interface serial`** *`slot/port:channel-group`*	Enter interface configuration mode for a serial interface that you specify by slot and port. The *:channel-group* portion of the command is only required for channelized T1 interfaces. (For setting up channelized T1 interfaces, see *Dial Solutions Configuration Guide* for Cisco IOS Release 12.0.)
3	`Router(config-if)#` **`ip address`** *`ip-address mask`*	Assign the IP address and subnet mask to the interface.

Verifying Serial Interface Configuration

To verify serial interface configuration, enter the privileged EXEC command **show interfaces serial**, which displays the status of all serial interfaces or of a specific serial interface, as shown in the following example. You can use this command to check the encapsulation, IP addressing, and other settings:

```
Router #show interface serial0/0:0
Serial0/0:0 is up, line protocol is up
  Hardware is QUICC Serial
  Internet address is 1.156.1.1/24
  MTU 1500 bytes, BW 1536 Kbit, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation HDLC, loopback not set
  Keepalive not set
  Last input 00:00:00, output 00:00:00, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0 (size/max/drops); Total output drops: 0
  Queueing strategy: weighted fair
  Output queue: 0/1000/64/0 (size/max total/threshold/drops)
     Conversations  0/1/256 (active/max active/max total)
     Reserved Conversations 0/0 (allocated/max allocated)
  5 minute input rate 1000 bits/sec, 1 packets/sec
  5 minute output rate 1000 bits/sec, 1 packets/sec
     637 packets input, 64736 bytes, 0 no buffer
     Received 181 broadcasts, 0 runts, 5 giants, 0 throttles
     3617 input errors, 1506 CRC, 1646 frame, 0 overrun, 0 ignored, 0 abort
     682 packets output, 67213 bytes, 0 underruns
     0 output errors, 0 collisions, 1070 interface resets
     0 output buffer failures, 0 output buffers swapped out
     13 carrier transitions
     Timeslot(s) Used:1-24, Transmitter delay is 0 flags
```

Configuring Voice Ports

Follow these steps to set up voice ports to support the local and remote stations. Not all possible commands are shown here. To learn more, see *Voice, Video, and Home Applications Configuration Guide* and *Voice, Video, and Home Applications Command Reference* for Cisco IOS Release 12.0.

Step	Command	Purpose
1	`Router#` **`configure terminal`**	Enter global configuration mode.
2	`Router(config)#` **`voice-port`** *`slot/port:ds0-group-no`*	Enter voice-port configuration mode. *slot* is the router location where the voice module is installed. Valid entries are from 0 to 3. *port* indicates the voice interface card location. Valid entries are 0 or 1. Each defined DS0 group number is represented on a separate voice port. This allows you to define individual DS0s on the digital T1 card. For more information about DS0 groups, see Step 12 of "Configuring Voice Card and T1 Controller Settings" on page 13. **Note** This **voice-port** command syntax does not apply to analog voice network modules and voice interface cards. The latter are specified using *slot*/*subunit*/*port*, designating the router slot for the voice network module, the location of the voice interface card in the network module, and the port on the voice interface card.
3	`Router(config-voice-port)#` **`busyout monitor interface`** *`interface number`*	(Optional) This command allows you to specify a LAN or WAN interface that will be monitored, and, when it is down, trigger a busyout (offhook) state on the voice port. This allows rerouting of calls. For example, if you specify *Serial 1/0* as the interface and number, the voice port sends a busyout signal when that interface is down. You can issue the command repeatedly to specify as many interfaces, virtual interfaces, and subinterfaces as are required for a voice port. For example, if you issue the command three times so that three interfaces are monitored, the voice port only goes into busyout state when all three interfaces are down. When any one of the interfaces is operational, the busyout state is removed.
4	`Router(config-voice-port)#` **`comfort-noise`**	(Optional) This parameter is enabled by default. It creates subtle background noise to fill silent gaps during calls when VAD is enabled on voice dial peers. If comfort noise is not generated, the silence can be unsettling to callers.
5	`Router(config-voice-port)#` **`echo-cancel enable`**	(Optional) This setting is enabled by default. Echo cancellation adds to the quality of voice transmissions by adjusting the echo that occurs on the interface due to impedance mismatches. Some echo is reassuring; echo over 25 milliseconds can cause problems.
6	`Router(config-voice-port)#` **`echo-cancel coverage`** {**16** | **24** |**32** | **8**}	(Optional) This command adjusts the echo canceller by the specified number of milliseconds; the default is 16.

Step	Command	Purpose
7	Router(config-voice-port)# **connection** {**plar** |**trunk**} *string*	(Optional) This command sets up a connection mode for the voice port. **plar** specifies a private line auto ring down (PLAR) connection, which rings a remote telephone when the dial peer goes off hook. **trunk** specifies a straight tie-line connection to a PBX. *string* specifies the remote telephone number or significant start digits of the number. See the "Configuration Examples" section on page 33 for sample PLAR and trunk configurations.
8	Router(config-voice-port)# **timeouts interdigit** *seconds*	(Optional) This command sets the number of seconds the system waits—after the caller has input the initial digit—for a subsequent digit of the dialed string. If the timeout ends before the destination is identified, a tone sounds and the call ends. The default value is 10 seconds, and the timeout can be set from 0 to 120 seconds. **Note** Changes to the default for this command normally are not required. Other timing settings may also be needed. For more information, see the Cisco IOS Release 12.0 *Voice, Video, and Home Applications Configuration Guide*.
9	Router(config-voice-port)# **exit**	Exit voice-port configuration mode. Repeat Steps 2 through 9 for each DS0 group you create.

Verifying Voice Ports

Follow the procedure below to verify voice-port configuration. To learn more about these commands, see *Voice, Video, and Home Applications Command Reference* for Cisco IOS Release 12.0.

Important command output is shown in bold.

To verify the voice-port configuration, enter the privileged EXEC **show voice port** *slot*/*port*:*ds0-group* command. The following sample output from the command shows explanatory information after the "<<" characters:

```
cisco-router# show voice port 1/0:1

receEive and transMit Slot is 1, Sub-unit is 0, Port is 1  << voice-port 1/0:1
 Type of VoicePort is E&M
 Operation State is DORMANT
 Administrative State is UP
 No Interface Down Failure
 Description is not set
 Noise Regeneration is enabled
 Non Linear Processing is enabled
 Music On Hold Threshold is Set to -38 dBm
 In Gain is Set to 0 dB
 Out Attenuation is Set to 0 dB
 Echo Cancellation is enabled
 Echo Cancel Coverage is set to 8 ms
 Connection Mode is normal
 Connection Number is not set
 Initial Time Out is set to 10 s
 Interdigit Time Out is set to 10 s
 Region Tone is set for US
```

Configuring Voice Dial Peers

Follow these steps to set up voice dial peers to support the local and remote stations. Not all possible commands are shown here. To learn more, see *Voice, Video, and Home Applications Configuration Guide* and *Voice, Video, and Home Applications Command Reference* for Cisco IOS Release 12.0.

Step	Command	Purpose
1	`Router#` **`configure terminal`**	Enter global configuration mode.
2	`Router(config)#` **`dial-peer voice`** *`number`* **`pots`**	Enter dial-peer configuration mode and define a local dial peer that will connect to the plain old telephone service (POTS) network. *number* is one or more digits identifying the dial peer. Valid entries are from 1 to 2147483647. **pots** indicates a peer using basic telephone service.
3	`Router(config-dialpeer)#` **`destination-pattern`** *`string`* `[`**`T`**`]`	Configure the dial peer's destination pattern so that the system can reconcile dialed digits with a telephone number. *string* is a series of digits that specify the E.164 or private dialing plan phone number. Valid entries are the digits 0 through 9 and the letters A through D. The plus symbol (+) is not valid. The following special characters can be entered: • The star character (*) that appears on standard touch-tone dial pads can be in any dial string but not as a leading character (for example, *650). • The period (.) acts as a wildcard character. • The comma (,) can be used only in prefixes and inserts a one-second pause. When the timer (T) character is included at the end of the destination pattern, the system collects dialed digits as they are entered—until the interdigit timer expires (10 seconds, by default)—or the user dials the termination of end-of-dialing key (default is #). **Note** The timer character must be a capital T.
4	`Router(config-dialpeer)#` **`prefix`** *`string`*	(Optional) Include a dial-out prefix that the system enters automatically instead of people dialing it. *string* is a value from 0 to 9, and you can use a comma (,) to indicate a pause. **Note** There are other digit manipulation commands available to handle such situations as prefixes for special services, ignoring some digits, and dialing into remote PBXs as though they are local.
5	`Router(config-dialpeer)#` **`port`** *`slot`*`/`*`port`*`:`*`ds0-group-no`*	This command associates the dial peer with a specific logical interface. *slot* is the router location where the voice module is installed. Valid entries are from 0 to 3. *port* indicates the voice interface card location. Valid entries are 0 or 1. Each defined DS0 group number is represented on a separate voice port. This allows you to define individual DS0s on the digital T1 card.

Step	Command	Purpose
6	`Router(config)#` **`dial-peer voice`** *`number`* **`voip`**	Enter dial-peer configuration mode and define a remote VoIP dial peer. *number* is one or more digits identifying the dial peer. Valid entries are from 1 to 2147483647. **voip** indicates a VoIP peer using voice encapsulation on the IP network.
7	`Router(config-dialpeer)#` **`codec`** {**`g711alaw`** | **`g711ulaw`** | **`g723ar53`** | **`g723ar63`** | **`g723r53`** | **`g723r63`** | **`g726r16`** | **`g726r24`** | **`g726r32`** | **`g728`** | **`g729r8`** [**`pre-ietf`**] | **`g729br8`** } [*`bytes`*]	The voice-card configuration **codec complexity** command sets the codec options that are available when you execute this command. See Step 3 of the "Configuring Voice Card and T1 Controller Settings" section on page 13. If you do not set codec complexity, **g729r8** with IETF bit-ordering is used. If you set codec complexity to **high**, the following options are available: • **g711alaw**—G.711 A Law 64,000 bps • **g711ulaw**—G.711 u Law 64,000 bps • **g723ar53**—G.723.1 Annex A 5,300 bps • **g723ar63**—G.723.1 Annex A 6,300 bps • **g723r53**—G.723.1 5,300 bps • **g723r63**—G.723.1 6,300 bps • **g726r16**—G.726 16,000 bps • **g726r24**—G.726 24,000 bps • **g726r32**—G.726 32,000 bps • **g728**—G.728 16,000 bps • **g729r8**---G.729 8,000 bps (default) • **g729br8**—G.729 Annex B 8,000 bps If you set codec complexity to **medium**, the following options are valid: • **g711alaw**—G.711 A Law 64,000 bps • **g711ulaw**—G.711 u Law 64,000 bps • **g726r16**—G.726 16,000 bps • **g726r24**—G.726 24,000 bps • **g726r32**—G.726 32,000 bps • **g729r8**—G.729 Annex A 8,000 bps • **g729br8**—G.729 Annex B with Annex A 8,000 bps The optional *bytes* parameter sets the number of voice data bytes per frame. Acceptable values are from 10 to 240 in increments of 10 (for example, 10, 20, 30, and so on). Any other value is rounded down (for example, from 236 to 230). If you specify **g729r8**, then the IETF (Internet Engineering Task Force) bit-ordering is used. For interoperability with a Cisco 2600, 3600, or AS5300 router running a Cisco IOS release prior to Release 12.0(5)T or12.0(4)XH, you *must* specify the additional key word **pre-ietf** after **g729r8**.

Step	Command	Purpose
8	`Router(config-dialpeer)#` **`vad`**	(Optional) This setting is enabled by default. It activates voice activity detection (VAD). VAD allows the system to reduce unnecessary voice transmissions caused by unfiltered background noise.
9	`Router(config-dialpeer)#` **`dtmf-relay`** `[`**`cisco-rtp`**`] [`**`h245-signal`**`] [`**`h245-alphanumeric`**`]`	(Optional) Dual-tone multifrequency (DTMF) describes the tone that sounds in response to a keypress on a touch-tone phone. DTMF tones are compressed at one end of a call and decompressed at the other end. If a low-bandwidth codec, such as a G.729 or G.723, is used, the tones can sound distorted. The **dtmf-relay** command transports DTMF tones generated after call establishment out-of-band by using a method that transmits with greater fidelity than is possible in-band for most low-bandwidth codecs. Without DTMF relay, calls established with low-bandwidth codecs may have trouble accessing automated phone menu systems, such as voice mail and interactive voice response (IVR) systems. A signaling method is supplied only if the remote end supports it, and the options are: Cisco proprietary (**cisco-rtp**), standard H.323 (**h245-alphanumeric**), and H.323 standard with signal duration (**h245-signal**).
10	`Router(config-dialpeer)#` **`fax-rate`** `{`**`2400`** `|` **`4800`** `|` **`7200`** `|` **`9600`** `|` **`12000`** `|` **`14400`** `|` **`disable`** `|` **`voice`**`}`	(Optional) Specify the transmission speed of a fax to be sent to this dial peer. **disable** turns off fax transmission capability, and **voice** specifies the highest possible fax speed supported by the voice rate.
11	`Router(config-dialpeer)#` **`destination-pattern`** *`string`* `[`**`T`**`]`	See Step 3 in this procedure.
12	`Router(config-dialpeer)#` **`session target`** `{`**`ipv4:`***`destination-address`* `|` **`dns:`**`[`**`$s$.`** `|` **`$d$.`** `|` **`$e$.`** `|` **`$u$.`**`]` *`host-name`*`}`	Configure the IP session target for the dial peer. **ipv4:***destination-address* indicates IP address of the dial peer. **dns:***host-name* indicates that the domain name server will resolve the name of the IP address. Valid entries for this parameter are characters representing the name of the host device. There are also wildcards available for defining domain names with the keyword by using source, destination, and dialed information in the host name. For complete command syntax information, see *Voice, Video, and Home Applications Command Reference* for Cisco IOS Release 12.0.

Verifying Voice Dial Peers

Follow the procedure below to verify dial-peer configuration. To learn more about these commands, see *Voice, Video, and Home Applications Command Reference* for Cisco IOS Release 12.0.

Important command output is shown in bold.

Enter the privileged EXEC **show dial-peer voice** command. The following text is sample output from the command for a POTS dial peer:

```
router# show dial-peer voice 1
VoiceEncapPeer1
        tag = 1, dest-pat = \Q+14085551000',
        answer-address = \Q',
```

```
        group = 0, Admin state is up, Operation state is down
        Permission is Both,
        type = pots, prefix = \Q',
        session-target = \Q', voice-port =
        Connect Time = 0, Charged Units = 0
        Successful Calls = 0, Failed Calls = 0
        Accepted Calls = 0, Refused Calls = 0
        Last Disconnect Cause is "10"
        Last Disconnect Text is ""
        Last Setup Time = 0
```

The following text is sample output from the **show dial-peer voice** command for a VoIP dial peer:

```
Router# show dial-peer voice 10
VoiceOverIpPeer10
        tag = 10, dest-pat = \Q',
        incall-number = \Q+14087',
        group = 0, Admin state is up, Operation state is down
        Permission is Answer,
        type = voip, session-target = \Q',
        sess-proto = cisco, req-qos = bestEffort,
        acc-qos = bestEffort,
        fax-rate = voice, codec = g729r8,
        Expect factor = 10,Icpif = 30, VAD = disabled, Poor QOV Trap = disabled,
        Connect Time = 0, Charged Units = 0
        Successful Calls = 0, Failed Calls = 0
        Accepted Calls = 0, Refused Calls = 0
        Last Disconnect Cause is "10"
        Last Disconnect Text is ""
        Last Setup Time = 0
```

Monitoring and Maintaining T1 Digital Packet Voice Configuration

This section presents some useful show and debugging commands for understanding, maintaining, and troubleshooting your configuration.

Table 1 Debug and Show Commands for Maintaining and Troubleshooting Your Configuration

Command	Purpose
Router# **show dialplan number** *number*	Shows which dial-peer is matched by a called number.
Router# **show call active voice**	Shows statistics for currently active voice calls.
Router# **show call active fax**	Shows statistics for currently active fax calls.
Router# **show call history voice**	Shows statistics on previous voice calls.
Router# **show call history fax**	Shows statistics on previous fax calls.
Router# **show connect** {**all** | **elements** | **name** | **id** | **port** { **T1** | **E1** } *slot/port* }}	Shows the status of connections. See “Verifying Voice Card and Controller Settings” on page 18.
Router# **show voice port**	Shows the status of voice ports. See “Verifying Voice Ports” on page 21.
Router# **show controller t1** *slot/port*	Shows the status of the T1 controller. See “Verifying Voice Card and Controller Settings” on page 18.
Router# **debug vpm all**	Debugs the T1 signaling.
Router# **debug vtsp all**	Debugs the digits received and sent.
Router# **debug voip ccapi inout**	Debugs the call setup process.

The balance of this section shows the output of the commands listed in Table 1.

Show Commands

This section illustrates some of the privileged EXEC show commands that are useful for analyzing your system. Note that important information appears in bold, and bold text preceded by the “<<“ characters explains the process.

The **show dialplan number** command provides information about the dial peer associated with a specified dial-plan number. Notice that the dial peer is operational and that IP Precedence has been configured to the preferred setting of 5.

Note To pair different voice ports and telephone numbers together for troubleshooting, enter the **show dialplan incall** *number* privileged EXEC command.

```
cisco-router# show dialplan number 75435
Macro Exp.: ##75435
VoiceOverIpPeer70000
        information type = voice,
        tag = 70000, destination-pattern = `##7....',
        answer-address = `', preference=0,
        group = 70000, Admin state is up, Operation state is up,
        incoming called-number = `', connections/maximum = 0/unlimited,
        DTMF Relay = disabled,
        application associated:
        type = voip, session-target = `ipv4:171.68.253.18',
        technology prefix:
        settlement: disabled
        ip precedence = 5, UDP checksum = disabled,
        session-protocol = cisco, req-qos = best-effort,
        acc-qos = best-effort,
        fax-rate = 14400,   payload size =  20 bytes
        codec = g729r8,   payload size =  20 bytes,
        Expect factor = 10, Icpif = 30,signaling-type = cas,
        VAD = disabled, Poor QOV Trap = disabled,
        Connect Time = 0, Charged Units = 0,
        Successful Calls = 3, Failed Calls = 0,
        Accepted Calls = 3, Refused Calls = 0,
        Last Disconnect Cause is "10  ",
        Last Disconnect Text is "normal call clearing.",
        Last Setup Time = 344813.
Matched: ##75435   Digits: 3
Target: ipv4:171.68.253.18
```

The **show call active voice** command displays information about a current call:

```
cisco-router# show call active voice

GENERIC:
SetupTime=94523746 ms
Index=448
PeerAddress=##73072
PeerSubAddress=
PeerId=70000
PeerIfIndex=37
LogicalIfIndex=0
ConnectTime=94524043
DisconectTime=94546241
CallOrigin=1
```

```
ChargedUnits=0
InfoType=2
TransmitPackets=6251
TransmitBytes=125020
ReceivePackets=3300
ReceiveBytes=66000
VOIP:
ConnectionId[0x142E62FB 0x5C6705AF 0x0 0x385722B0]
RemoteIPAddress=171.68.235.18
RemoteUDPPort=16580
RoundTripDelay=29 ms
SelectedQoS=best-effort
tx_DtmfRelay=inband-voice
SessionProtocol=cisco
SessionTarget=ipv4:171.68.235.18
OnTimeRvPlayout=63690
GapFillWithSilence=0 ms
GapFillWithPrediction=180 ms
GapFillWithInterpolation=0 ms
GapFillWithRedundancy=0 ms
HiWaterPlayoutDelay=70 ms
LoWaterPlayoutDelay=30 ms
ReceiveDelay=40 ms
LostPackets=0 ms
EarlyPackets=1 ms
LatePackets=18 ms
VAD = disabled
CoderTypeRate=g729r8
CodecBytes=20
cvVoIPCallHistoryIcpif=0
SignalingType=cas
```

The **show call history voice** command shows statistics about previous calls:

```
sb1pbx-voip# show call history voice

GENERIC:
SetupTime=94893250 ms
Index=450
PeerAddress=##52258
PeerSubAddress=
PeerId=50000
PeerIfIndex=35
LogicalIfIndex=0
DisconnectCause=10
DisconnectText=normal call clearing.
ConnectTime=94893780
DisconectTime=95015500
CallOrigin=1
ChargedUnits=0
InfoType=2
TransmitPackets=32258
TransmitBytes=645160
ReceivePackets=20061
ReceiveBytes=401220
VOIP:
ConnectionId[0x142E62FB 0x5C6705B3 0x0 0x388F851C]
RemoteIPAddress=171.68.235.18
RemoteUDPPort=16552
RoundTripDelay=23 ms
SelectedQoS=best-effort
tx_DtmfRelay=inband-voice
SessionProtocol=cisco
SessionTarget=ipv4:171.68.235.18
OnTimeRvPlayout=398000
```

```
GapFillWithSilence=0 ms
GapFillWithPrediction=1440 ms
GapFillWithInterpolation=0 ms
GapFillWithRedundancy=0 ms
HiWaterPlayoutDelay=97 ms
LoWaterPlayoutDelay=30 ms
ReceiveDelay=49 ms
LostPackets=1 ms
EarlyPackets=1 ms
LatePackets=132 ms
VAD = disabled
CoderTypeRate=g729r8
CodecBytes=20
cvVoIPCallHistoryIcpif=0
SignalingType=cas
```

Debug Commands

This section illustrates some of the EXEC mode debug commands that are useful when analyzing and troubleshooting your system. Note that important information appears in bold, and bold text preceded by the “<<“ characters explains the process.

The **debug vpm all** command displays information that allows you to troubleshoot T1 signaling:

```
cisco-router# debug vpm all
Apr 19 19:18:54 PDT: htsp_process_event: [1/0/16, 1.4 , 34]
em_onhook_offhookem_offhookem_onhookhtsp_setup_ind  << port goes offhook
Apr 19 19:18:54 PDT: htsp_process_event: [1/0/16, 1.5 , 8]
Apr 19 19:19:01 PDT: htsp_process_event: [1/0/16, 1.5 , 10] htsp_alert_notify
Apr 19 19:19:01 PDT: htsp_process_event: [1/0/16, 1.5 , 11]
Apr 19 19:19:02 PDT: htsp_process_event: [1/0/16, 1.5 , 11]
Apr 19 19:19:15 PDT: htsp_process_event: [1/0/16, 1.5 , 22]
em_offhook_onhookem_stop_timers em_onhook            << port goes onhook
Apr 19 19:19:15 PDT: htsp_process_event: [1/0/16, 1.4 , 7] em_onhook_releaseem_onhook
```

The **debug vtsp all** command displays information that allows you to troubleshoot digits received and sent on a call:

```
cisco-router# debug vtsp all
Apr 19 19:21:55 PDT: dsp_cp_tone_on: [1/0:1 (9502)] packet_len=30 channel_id=1
packet_id=72 tone_id=3 n_freq=2 freq_of_first=350 freq_of_second=440 amp_of_first=4000
amp_of_second=4000 direction=1 on_time_first=65535 off_time_first=0
on_time_second=65535 off_time_second=0  << providing dialtone

Apr 19 19:21:59 PDT: vtsp_process_dsp_message: MSG_TX_DTMF_DIGIT_BEGIN:
digit=2,rtp_timestamp=0xF2D37240
act_report_digit_begin
Apr 19 19:22:00 PDT: vtsp_process_dsp_message: MSG_TX_DTMF_DIGIT_OFF: digit=2,
duration=102act_report_digit_end
Apr 19 19:22:00 PDT: dsp_cp_tone_off: [1/0:1 (9502)] packet_len=8 channel_id=1
packet_id=71
Apr 19 19:22:00 PDT: vtsp_timer: 34838705
Apr 19 19:22:00 PDT: vtsp_process_dsp_message: MSG_TX_DTMF_DIGIT_BEGIN:
digit=3,rtp_timestamp=0xF2D37240
act_report_digit_begin
Apr 19 19:22:00 PDT: vtsp_process_dsp_message: MSG_TX_DTMF_DIGIT_OFF: digit=3,
duration=92act_report_digit_end
Apr 19 19:22:00 PDT: dsp_cp_tone_off: [1/0:1 (9502)] packet_len=8 channel_id=1
packet_id=71
Apr 19 19:22:00 PDT: vtsp_timer: 34838724
Apr 19 19:22:00 PDT: vtsp_process_dsp_message: MSG_TX_DTMF_DIGIT_BEGIN:
digit=1,rtp_timestamp=0xF2D37240 act_report_digit_begin
```

```
Apr 19 19:22:00 PDT: vtsp_process_dsp_message: MSG_TX_DTMF_DIGIT_OFF: digit=1,
duration=92act_report_digit_end
Apr 19 19:22:00 PDT: dsp_cp_tone_off: [1/0:1 (9502)] packet_len=8 channel_id=1
packet_id=71
Apr 19 19:22:00 PDT: vtsp_timer: 34838744
Apr 19 19:22:00 PDT: vtsp_process_dsp_message: MSG_TX_DTMF_DIGIT_BEGIN:
digit=9,rtp_timestamp=0xF2D37240
act_report_digit_begin
Apr 19 19:22:00 PDT: vtsp_process_dsp_message: MSG_TX_DTMF_DIGIT_OFF: digit=9,
duration=102act_report_digit_end
Apr 19 19:22:00 PDT: dsp_cp_tone_off: [1/0:1 (9502)] packet_len=8 channel_id=1
packet_id=71
Apr 19 19:22:00 PDT: vtsp_timer: 34838768
Apr 19 19:22:00 PDT: vtsp_process_dsp_message: MSG_TX_DTMF_DIGIT_BEGIN:
digit=8,rtp_timestamp=0xF2D37218
act_report_digit_begin
Apr 19 19:22:00 PDT: vtsp_process_dsp_message: MSG_TX_DTMF_DIGIT_OFF: digit=8,
duration=107act_report_digit_end

*** The Caller dialed the digits 23198 ***
```

The **debug voip ccapi inout** EXEC command traces the execution path through the call control API, which serves as the interface between the call-session application and the underlying network-specific software.

During the capabilities exchange shown in the command output, both sides agree on what compression to use, and the **debug voip ccapi inout** output helps you determine what each side is negotiating.

You can use the output from this command to understand how calls are being handled by the router. This command shows how a call flows through the system. By using this debug level, you can see the call setup and teardown operations performed on both the telephony and network call legs:

```
cisco-router# debug voip ccapi inout
 Apr 19 19:23:11 PDT: sess_appl: ev(19=CC_EV_CALL_SETUP_IND), cid(9504), disp(0)  << a
new call is originating
Apr 19 19:23:11 PDT: ccCallSetContext (callID=0x2520, context=0x61C0806C)
Apr 19 19:23:11 PDT: ccCallSetupAck (callID=0x2520)
Apr 19 19:23:11 PDT: ccGenerateTone (callID=0x2520 tone=8)  << dialtone
Apr 19 19:23:18 PDT: cc_api_call_digit_begin (vdbPtr=0x61A1B1B4, callID=0x2520,
digit=2, flags=0x1, timestamp=0xCE2796D1, expiration=0x0)  << digit 2 received
Apr 19 19:23:18 PDT: sess_appl: ev(10=CC_EV_CALL_DIGIT_BEGIN), cid(9504), disp(0)
Apr 19 19:23:18 PDT: ssa: cid(9504)st(0)oldst(0)cfid(-1)csize(0)in(1)fDest(0)
Apr 19 19:23:18 PDT: ssaIgnore cid(9504), st(0),oldst(0), ev(10)
Apr 19 19:23:18 PDT: cc_api_call_digit (vdbPtr=0x61A1B1B4, callID=0x2520, digit=2,
duration=102)
Apr 19 19:23:18 PDT: sess_appl: ev(9=CC_EV_CALL_DIGIT), cid(9504), disp(0)
Apr 19 19:23:18 PDT: ssa: cid(9504)st(0)oldst(0)cfid(-1)csize(0)in(1)fDest(0)
Apr 19 19:23:18 PDT: cc_api_call_digit_begin (vdbPtr=0x61A1B1B4, callID=0x2520,
digit=3, flags=0x1, timestamp=0xCE2796D1, expiration=0x0)
Apr 19 19:23:18 PDT: sess_appl: ev(10=CC_EV_CALL_DIGIT_BEGIN), cid(9504), disp(0)
Apr 19 19:23:18 PDT: ssa: cid(9504)st(0)oldst(0)cfid(-1)csize(0)in(1)fDest(0)
Apr 19 19:23:18 PDT: ssaIgnore cid(9504), st(0),oldst(0), ev(10)
Apr 19 19:23:18 PDT: cc_api_call_digit (vdbPtr=0x61A1B1B4, callID=0x2520, digit=3,
duration=102)  << digit 3 received
Apr 19 19:23:18 PDT: sess_appl: ev(9=CC_EV_CALL_DIGIT), cid(9504), disp(0)
Apr 19 19:23:18 PDT: ssa: cid(9504)st(0)oldst(0)cfid(-1)csize(0)in(1)fDest(0)
Apr 19 19:23:18 PDT: cc_api_call_digit_begin (vdbPtr=0x61A1B1B4, callID=0x2520,
digit=1, flags=0x1, timestamp=0xCE2796D1, expiration=0x0)
Apr 19 19:23:18 PDT: sess_appl: ev(10=CC_EV_CALL_DIGIT_BEGIN), cid(9504), disp(0)
Apr 19 19:23:18 PDT: ssa: cid(9504)st(0)oldst(0)cfid(-1)csize(0)in(1)fDest(0)
Apr 19 19:23:18 PDT: ssaIgnore cid(9504), st(0),oldst(0), ev(10)
Apr 19 19:23:18 PDT: cc_api_call_digit (vdbPtr=0x61A1B1B4, callID=0x2520, digit=1,
duration=92)    << digit 1 received
```

```
Apr 19 19:23:18 PDT: sess_appl: ev(9=CC_EV_CALL_DIGIT), cid(9504), disp(0)
Apr 19 19:23:18 PDT: ssa: cid(9504)st(0)oldst(0)cfid(-1)csize(0)in(1)fDest(0)
Apr 19 19:23:18 PDT: cc_api_call_digit_begin (vdbPtr=0x61A1B1B4, callID=0x2520,
digit=9, flags=0x1, timestamp=0xCE2796B9, expiration=0x0)
Apr 19 19:23:18 PDT: sess_appl: ev(10=CC_EV_CALL_DIGIT_BEGIN), cid(9504), disp(0)
Apr 19 19:23:18 PDT: ssa: cid(9504)st(0)oldst(0)cfid(-1)csize(0)in(1)fDest(0)
Apr 19 19:23:18 PDT: ssaIgnore cid(9504), st(0),oldst(0), ev(10)
Apr 19 19:23:18 PDT: cc_api_call_digit (vdbPtr=0x61A1B1B4, callID=0x2520, digit=9,
duration=105)    << digit 9 received
Apr 19 19:23:18 PDT: sess_appl: ev(9=CC_EV_CALL_DIGIT), cid(9504), disp(0)
Apr 19 19:23:18 PDT: ssa: cid(9504)st(0)oldst(0)cfid(-1)csize(0)in(1)fDest(0)
Apr 19 19:23:18 PDT: cc_api_call_digit_begin (vdbPtr=0x61A1B1B4, callID=0x2520,
digit=8, flags=0x1, timestamp=0xCE279691, expiration=0x0)
Apr 19 19:23:18 PDT: sess_appl: ev(10=CC_EV_CALL_DIGIT_BEGIN), cid(9504), disp(0)
Apr 19 19:23:18 PDT: ssa: cid(9504)st(0)oldst(0)cfid(-1)csize(0)in(1)fDest(0)
Apr 19 19:23:18 PDT: ssaIgnore cid(9504), st(0),oldst(0), ev(10)
Apr 19 19:23:18 PDT: cc_api_call_digit (vdbPtr=0x61A1B1B4, callID=0x2520, digit=8,
duration=100)    << digit 8 received
Apr 19 19:23:18 PDT: sess_appl: ev(9=CC_EV_CALL_DIGIT), cid(9504), disp(0)
Apr 19 19:23:18 PDT: ssa: cid(9504)st(0)oldst(0)cfid(-1)csize(0)in(1)fDest(0)
Apr 19 19:23:18 PDT: ssaSetupPeer cid(9504) peer list:  tag(20000)
Apr 19 19:23:18 PDT: ssaSetupPeer cid(9504), destPat(23198), matched(1), prefix(),
peer(61C04464)   << matched dial-peer 20000 voip

Apr 19 19:23:18 PDT: peer_tag=20000    << matched dial-peer voip 20000
Apr 19 19:23:18 PDT: ccIFCallSetupRequest: (vdbPtr=0x61A25524, dest=, callParams
<< voip call setup
={called=23198, calling=+9.......T, fdest=0, voice_peer_tag=20000}, mode=0x0)
Apr 19 19:23:18 PDT: ccCallSetContext (callID=0x2521, context=0x61C12E18)
Apr 19 19:23:18 PDT: ccCallProceeding (callID=0x2520, prog_ind=0x0)
Apr 19 19:23:19 PDT: cc_api_call_alert(vdbPtr=0x61A25524, callID=0x2521, prog_ind=0x88,
sig_ind=0x1)
Apr 19 19:23:19 PDT: sess_appl: ev(7=CC_EV_CALL_ALERT), cid(9505), disp(0)
Apr 19 19:23:19 PDT: ssa:
cid(9505)st(1)oldst(0)cfid(-1)csize(0)in(0)fDest(0)-cid2(9504)st2(1)oldst2(0)
Apr 19 19:23:19 PDT: ccCallAlert (callID=0x2520, prog_ind=0x88, sig_ind=0x1)
Apr 19 19:23:19 PDT: ccConferenceCreate (confID=0x61A21670, callID1=0x2520,
callID2=0x2521, tag=0x0)
Apr 19 19:23:19 PDT: cc_api_bridge_done (confID=0x33, srcIF=0x61A25524,
srcCallID=0x2521, dstCallID=0x2520, disposition=0, tag=0x0)
Apr 19 19:23:19 PDT: cc_api_bridge_done (confID=0x33, srcIF=0x61A1B1B4,
srcCallID=0x2520, dstCallID=0x2521, disposition=0, tag=0x0)
Apr 19 19:23:19 PDT: cc_api_caps_ind (dstVdbPtr=0x61A25524, dstCallId=0x2521, sr
<< negotiating capabilities with the remote VoIP gateway

Apr 19 19:23:36 PDT: sess_appl: ev(8=CC_EV_CALL_CONNECTED), cid(9505), disp(0)
Apr 19 19:23:36 PDT: ssa:
cid(9505)st(4)oldst(1)cfid(51)csize(0)in(0)fDest(0)-cid2(9504)st2(4)oldst2(4)
<< the VoIP call is connected

 Apr 19 19:23:54 PDT: sess_appl: ev(12=CC_EV_CALL_DISCONNECTED), cid(9505),disp(0)
<< the VoIP call is disconnected
 Apr 19 19:23:54 PDT: ccCallDisconnect (callID=0x2520, cause=0x10 tag=0x0)
<< the VoIP call is disconnected by cause_code 0x10
```

Table 2 explains the codec negotiation values that appear—in hexadecimal format— during the capabilities exchange portion of the command output.

Table 2 Codec Negotiation Values in debug voip ccapi inout

Negotiation Value in Decimal	Meaning
1	U-law PCM (g711ulaw)
2	A-law PCM (g711alaw)

Table 2 Codec Negotiation Values in debug voip ccapi inout (continued)

Negotiation Value in Decimal	Meaning
3	32k ADPCM (g726r32)
4	24k ADPCM (g726r24)
5	16k ADPCM (g726r16)
6	CS-ACELP - pre-IETF (g729r8 pre-ietf)
7	low complexity CS-ACELP - pre-IETF (g729ar8 pre-ietf)
8	CS-ACELP with VAD (g729br8)
9	low complexity CS-ACELP with VAD (G.729abr8)
10	16K LD-CELP (g728)
11	G.723.1 High Rate - 6300 bps (g723r63)
12	G.723.1 High Rate with VAD - 6300 bps (g723ar63)
13	G.723.1 Low Rate - 5300 bps (g723r53)
14	G.723.1 Low Rate with VAD - 5300 bps (g723ar53)
19	CS-ACELP - IETF standard (g729r8)
20	low complexity CS-ACELP - IETF standard (g729ar8)

Reference Information

The information in this section helps you interpret the output from **debug** and **show** commands.

Table 3 shows Q.931 call disconnection causes. In the examples that follow, the disconnects are caused by normal call clearing.

Table 3 Q.931 Call Disconnection Causes

Call Disconnection Cause Value	Meaning and Number
CC_CAUSE_UANUM = 0x1	/* unassigned number. (1) */
CC_CAUSE_NO_ROUTE = 0x3	/* no route to destination. (3) */
CC_CAUSE_NORM = 0x10	/* normal call clearing. (16) */
CC_CAUSE_BUSY = 0x11	/* user busy. (17) */
CC_CAUSE_NORS = 0x12	/* no user response. (18) */
CC_CAUSE_NOAN = 0x13	/* no user answer. (19) */
CC_CAUSE_REJECT = 0x15	/* call rejected. (21) */
CC_CAUSE_INVALID_NUMBER = 0x1C	/* invalid number. (28) */
CC_CAUSE_UNSP = 0x1F	/* normal, unspecified. (31) */
CC_CAUSE_NO_CIRCUIT = 0x22	/* no circuit. (34) */

Call Disconnection Cause Value	Meaning and Number
CC_CAUSE_NO_REQ_CIRCUIT = 0x2C	/* no requested circuit. (44) */
CC_CAUSE_NO_RESOURCE = 0x2F	/* no resource. (47) */
CC_CAUSE_NOSV = 0x3F	/* service or option not available,
	Unspecified. (63) */

Table 4 Tone Types and Their Meanings

Tone Type	Meaning
CC_TONE_RINGBACK	0x1 - Ring Tone
CC_TONE_FAX	0x2 - Fax Tone
CC_TONE_BUSY	0x4 - Busy Tone
CC_TONE_DIALTONE	0x8 - Dial Tone
CC_TONE_OOS	0x10 - Out of Service Tone
CC_TONE_ADDR_ACK	0x20 - Address Acknowledgement Tone
CC_TONE_DISCONNECT	0x40 - Disconnect Tone
CC_TONE_OFF_HOOK_NOTICE	0x80 - Tone indicating the phone was left off hook
CC_TONE_OFF_HOOK_ALERT	0x100 /* A more urgent version of CC_TONE_OFF_HOOK_NOTICE*/
CC_TONE_CUSTOM	0x200 - Custom Tone - used when specifying a custom tone
CC_TONE_NULL	0x0 - Null Tone

These are codec capabilities bits that can appear in command output:

- CC_CAP_CODEC_G711U 0x1
- CC_CAP_CODEC_G711A 0x2
- CC_CAP_CODEC_G723ar63 0x2000
- CC_CAP_CODEC_G723ar53 0x4000
- CC_CAP_CODEC_G723r63 0x100
- CC_CAP_CODEC_G723r53 0x200
- CC_CAP_CODEC_G726r16 0x10
- CC_CAP_CODEC_G729 0x4
- CC_CAP_CODEC_G729 0x8000
- CC_CAP_CODEC_G729a 0x8
- CC_CAP_CODEC_G729b 0x800
- CC_CAP_CODEC_G729ab 0x1000

These are fax capabilities bits that can appear in command output. The numbers following “FAX_” refer to the fax speed (for example, “144” means 14,400 bps):

- CC_CAP_FAX_NONE 0x1
- CC_CAP_FAX_VOICE 0x2
- CC_CAP_FAX_144 0x4

- CC_CAP_FAX_96 0x8
- CC_CAP_FAX_72 0x10
- CC_CAP_FAX_48 0x20
- CC_CAP_FAX_24 0x40
- CC_CAP_FAX_120 0x80

These are the VAD on and off capability bits:

- CC_CAP_VAD_OFF 0x1
- CC_CAP_VAD_ON 0x2

Configuration Examples

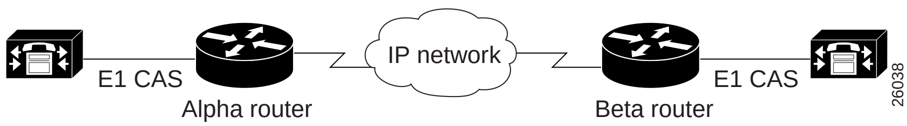
This section includes the following configuration examples:

- **Routed Digits.** Shows how to set up a router to collect digits from the PBX/PSTN or from a phone and route the VoIP call based on the digits received.
- **FRF.12.** Shows how to configure a Cisco 2600 or 3600 router to support FRF.12 fragmentation and queuing in a VoIP over Frame-Relay network.
- **Gatekeeper.** Shows how to configure a Cisco 2600 or 3600 series router to route VoIP calls by using an H.323 Gatekeeper.
- **Private-Line Auto-Ringdown (PLAR).** Shows how to set up a Cisco 2600 or 2600 series router for PLAR.
- **Trunk Connection.** Shows how to configure a Cisco 2600 or 3600 router for a transparent trunk connection.
- **Variable-Length Digits.** Shows how to configure a Cisco 2600 or 3600 router to collect variable-length strings of digits PBX/PSTN or phone and route the VoIP call based on the digits received.
- **Drop and Insert.** Shows how to configure a Cisco 2600 or 3600 router with a 2-port Drop-and-Insert T1 multiflex trunk voice/WAN interface card (VWIC-2MFT-T1-DI) and a digital T1 packet voice network module so that individual DS0 channels are transparently passed between T1 ports without going through a DSP. For example, this allows the directing of some PBX channels to the PSTN for long-distance service, while other channels are compressed for VoIP calls between interoffice sites.

These examples are not necessarily complete configurations. They are designed to illustrate specific tips and techniques, and only the relevant portions of the configurations are shown. Each configuration includes a brief introduction, side-by-side configurations for routers at either end, and explanations of key points.

Routed Digits - Switched VoIP Calls

Figure 6 Sample Configuration: Routed Digits



This example shows how to set up a Cisco 2600 or 3600 router to collect digits from either a PBX/PSTN or a phone and route a VoIP call based on the digits received. The commands used in the configurations are explained inline. Only relevant sections of the configuration are shown. The example assumes that the IP portion of the network is already in place.

```
hostname router-alpha
!
voice-card 1
 codec complexity high
!
dial-peer voice 1 voip
 codec g723r53
 fax-rate 14400
 destination-pattern 5....
 session target  ipv4:192.168.100.1
!
dial-peer voice 2 pots
 destination-pattern 4....
 prefix 4
 port 1/0:1
!
controller T1 1/0
 framing esf
 linecode b8zs
 clock source line
 ds0-group 1 timeslots 1-24 type e&m-wink
!
interface serial 0/0
 ip address 192.168.100.2 255.255.255.0
```

```
hostname router-beta
!
voice-card 1
 codec complexity high
!
dial-peer voice 1 voip
 codec g723r53
 fax-rate 14400
 destination-pattern 4....
 session-target ipv4:192.168.100.2
!
dial-peer voice 2 pots
 destination-pattern 5....
 prefix 5
 port 1/0:1
!
controller T1 1/0
 framing esf
 linecode b8zs
 clock source internal
 ds0-group 1 timeslot 1-24 type e&m-wink
!
interface s0/0
 ip address 192.168.100.1 255.255.255.0
```

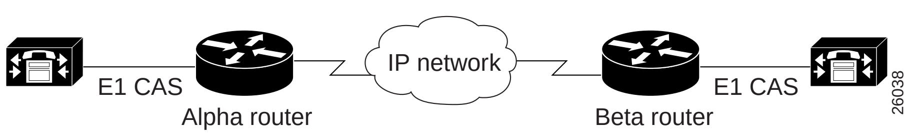
In this configuration, the PBX seizes the T1 to the router, which expects to collect digits from the PBX. Upon collecting those digits, the router tries to match a dial peer to route the call. If the router receives the correct digits, it routes the call according to the configuration of the dial peer.

Here are some key points for consideration:

- The **codec complexity high** command tells the router what types of codecs that can be used on this voice card—either high or medium. High-complexity permits only two calls for each DSP (6 for each PVDM-12). The codecs supported under high complexity are G.711, G.726, G.729, G.729 Annex B, G.728, G.723.1, G.723.1 Annex A, and fax relay. The default is medium complexity, which allows G.711, G.726, G.729 Annex A, G.729 Annex A with Annex B, and fax relay. Medium-complexity codecs permit four calls for each DSP—a total of twelve for each PVDM-12. All T1 cards in a router must have the same complexity. To change the codec complexity, first remove any configured DS0 group from the T1 controller and then reapply them after the change is complete.
- The **ds0-group 1 timeslots 1-24 type e&m-wink** command performs the following functions:
  - Defines the T1 channels for compressed voice calls.
  - Defines the signaling method that the router uses to connect to the PBX or PSTN.
  - Automatically creates a voice-port 1/0:1. The numbering for this voice-port is *slot*/*port*:*ds0-group no*. In this configuration, all calls to "4...." or "5...." are routed to any DS0 timeslot, although only 1/0:1 is shown. To map individual DS0s, define additional DS0 groups under the T1 controller. This creates individual DS0 voice ports.
- The **dial-peer voice** commands define the dialing plan within the router. They specify both the remote phone numbers (**voip** or **vofr**) and the locally connected phone numbers (**pots**). The digits in the destination pattern can either be complete numbers or partial numbers with wildcard digits, represented by ".". Each "." represents an individual digit for collection.

FRF.12 - Switched VoIP Calls

Figure 7 Sample Configuration: FRF.12 Switched VoIP Calls



This example shows how to configure a Cisco 2600 or 3600 router to support FRF.12 fragmentation and queuing in a voice over IP over Frame-Relay network. FRF.12 is a Frame Relay Forum standard mechanism for fragmenting data packets. This fragmentation helps eliminate the delays that occur when transmitting voice and data over the same network—large data packets can delay smaller voice packets from being transmitted into the IP network. FRF.12 is also supported on the MC3810 and 7200 routers, which can be used as tandem-nodes for VoIP networks.

Note This example shows VoIP over Frame Relay, which is not the same as voice over Frame Relay (VoFR). For more information about VoFR, see the Cisco IOS Release12.0(4)T feature module *Voice over Frame Relay Using FRF.11 and FRF.12.*

This configuration fragments both the IP and IPX data traffic to 80 bytes, allowing the VoIP traffic to be only minimally delayed on the network. The FRF.12 setup also traffic-shapes the output traffic rate to match the provisioned CIR from the Frame Relay carrier. This ensures that traffic is not dropped or delayed within the Frame Relay network.

Here are some key points for consideration:

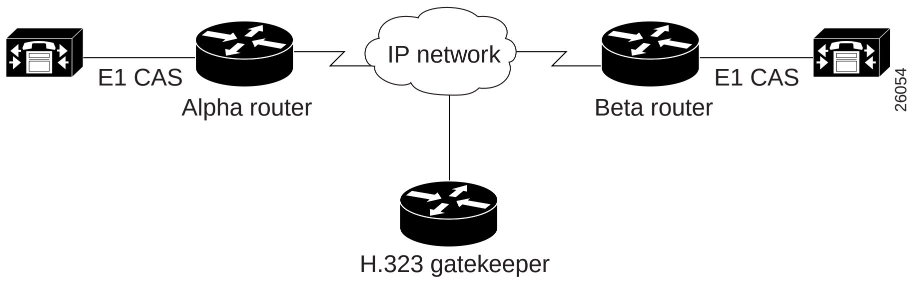
- The **frame-relay traffic-shaping** command enables Frame-Relay traffic-shaping (FRTS) on the main interface. Enable it if FRTS will be used on subinterfaces.
- The **class cisco_frf12** command tells the interface to use the parameters for FRTS defined in the map-class called "cisco_frf12."
- The **map-class cisco_frf12** grouping of commands defines the rules for FRTS. If per-interface/subinterface parameters must differ, define multiple map-classes per router.
- The **frame-relay fragment 80** command defines the size of the data or voice packets that FRF.12 fragments. Set the size to about the size of the voice packets or slightly larger. A good rule of thumb is 80 bytes for each DS0 of WAN bandwidth. With large quantities of bandwidth and small data frames, the fragment size may need to remain small.
- The **frame-relay fair-queue** command enables Weighted-Fair Queuing (WFQ) on a per-PVC basis to ensure that voice traffic gets priority over data traffic.

```
hostname router-alpha
!
ipx routing
!
voice-card 1
 codec complexity high
!
controller T1 1/0
 framing esf
 linecode b8zs
 clock source line
 ds0-group 1 timeslot 1-24 type e&m-wink
!
dial-peer voice 1 voip
 dtmf-relay  h245-alpha
 codec g723r53
 destination-pattern 5....
 session target ipv4:192.168.100.2
!
dial-peer voice 2 pots
 destination-pattern 4....
 prefix 4
 port 1/0:1
!
interface serial 0/0
 encapsulation frame-relay
 frame-relay traffic-shaping
!
interface serial 0/0.1 point-to-point
 ip address 192.168.100.1 255.255.255.0
 ipx network ABCD
 frame-relay interface-dlci 100
 class cisco_frf12
!
map-class frame-relay cisco_frf12
frame-relay voice bandwidth 42000
frame-relay fragment 80
no frame-relay adaptive-shaping
frame-relay cir 32000
frame-relay bc 1000
frame-relay mincir 64000
frame-relay fair-queue
```

```
hostname router-beta
!
ipx routing
!
voice-card 1
 codec complexity high
!
controller T1 1/0
 framing esf
 linecode b8zs
 clock source line
 ds0-group 1 timeslot 1-24 type e&m-wink
!
dial-peer voice 1 voip
 dtmf-relay h245-alpha
 codec g723r53
 destination-pattern 4....
 session target ipv4:192.168.100.2
!
dial-peer voice 2 pots
 destination-pattern 5....
 prefix 5
 port 1/0:1
!
interface serial 0/0
 encapsulation frame-relay
 frame-relay traffic-shaping
!
interface serial 0/0.1 point-to-point
 ip address 192.168.100.2 255.255.255.0
 ipx network ABCD
 frame-relay interface-dlci 101
 class cisco_frf12
!
map-class frame-relay cisco_frf12
frame-relay voice bandwidth 42000
frame-relay fragment 80
no frame-relay adaptive-shaping
frame-relay cir 64000
frame-relay bc 1000
frame-relay mincir 64000
frame-relay fair-queue
```

Routing Calls through an H.323 Gatekeeper

Figure 8 Sample Configuration: Routing Calls through an H.323 Gatekeeper



Note With the introduction of Cisco IOS Release 12.0(5)T and subsequent releases, Cisco VoIP gateways support H.323v2 (H.323 Version 2), which is backwards-compatible with systems running H.323v1. However, H.323 Version 2 features do not interoperate with H.323 Version 1 features in Cisco IOS releases prior to 11.3(9)NA or 12.0(3)T. Earlier Cisco IOS versions contain H.323 Version 1 software that does not support protocol messages with an H.323 Version 2 protocol identifier. All systems must be running either Cisco IOS version 11.3(9)NA and later or releases Cisco IOS version 12.0(3)T and later releases to interoperate with H.323 Version 2. Gateway Resource Availability Indication (RAI) messages are currently not supported on the Cisco 2600 and 3600 series. (These are messages that are sent to the Gatekeeper to inform it about the status of a Gateway's DSP or DS0 availability.)

The example in this section shows how to configure a Cisco 2600 or 3600 series router to route VoIP calls through an H.323 Gatekeeper. This setup shows calls being routed from a Gateway in Zone-Alpha, through the Gatekeeper, to a Gateway in Zone-Beta.

Alpha Router

```
hostname router-alpha
!
voice-card 1
!
controller T1 1/0
 framing esf
 linecode b8zs
 clock source internal
 ds0-group 1 timeslot 1-24 type e&m-wink
!
voice-port 1/0:1
!
dial-peer voice 1 voip
 dtmf-relay h245-alpha
 destination-pattern 5....
 tech-prefix 1#
 session target ras
!
dial-peer voice 2 pots
 destination-pattern 4....
 prefix 4
 port 1/0:1
!
gateway
!
interface ethernet 0/0
 ip address 10.1.1.1 255.255.255.0
 h323-gateway voip interface
 h323-gateway voip id alpha ipaddr 10.1.1.3
1719
 h323-gateway voip h323-id
router-alpha@alpha.com
 h323-gateway voip tech-prefix 1#
```

Beta Router

```
hostname router-beta
!
voice-card 1
!
controller T1 1/0
 framing esf
 linecode b8zs
 clock source line
 ds0-group 1 timeslot 10-24 type e&m-wink
!
voice-port 1/0:1
!
dial-peer voice 1 voip
 dtmf-relay h245-alpha
 destination-pattern 4....
 tech-prefix 1#
 session target ras
!
dial-peer voice 2 pots
 destination-pattern 5....
 prefix 5
 port 1/0:1
!
gateway
!
interface ethernet 0/0
 ip address 10.1.1.2 255.255.255.0
 h323-gateway voip interface
 h323-gateway voip id beta ipaddr 10.1.1.3
1719
 h323-gateway voip h323-id
router-beta@beta.com
 h323-gateway voip tech-prefix 1#
```

Gatekeeper

```
hostname router-gatekeeper
!
gatekeeper
zone local alpha alpha.com
zone local beta beta.com
no use-proxy alpha.com remote-zone beta.com
no use-proxy beta.com remote-zone alpha.com
zone prefix router-alpha 4....
zone prefix router-beta 5....
no shutdown
!
interface ethernet 0/0
ip address 10.1.1.3 255.255.255.0
```

For complete documentation of H.323 gatekeeper functionality, refer to the IOS documentation on CCO at these URLs:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios120/120newft/120t/120t3/mcmtcfg.htm

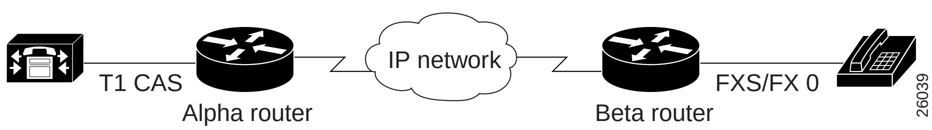
http://www.cisco.com/univercd/cc/td/doc/product/software/ios120/120newft/120t/120t3/mcmtcmd.htm

Here are some key points for consideration:

- The **session target ras** command tells the router to route through the Gatekeeper. RAS (Registration, Admission, Status) is the communication that occurs between an H.323 Gateway and the Gatekeeper.
- The **gateway** command tells the router to use RAS to register with the Gatekeeper.
- The **gatekeeper** command tells the router to act as a Gatekeeper and respond to calls made through RAS from H.323 Gateways and H.323 clients.

Private-Line Auto-Ringdown Configuration - Switched VoIP Calls

Figure 9 Sample Configuration: PLAR



This example shows how to set up a Cisco 2600 or 3600 series router for a private-line auto-ringdown (PLAR). PLAR is used to allow a station or DS0 to go off hook, and—without the user dialing digits—have a call completed to the far end. PLAR can also provide dial tone from a remote PBX for off-premises applications.

In this configuration, the phones off router Beta go off hook and receive dial tone from the PBX connected to router Alpha. From there, users can dial any digits in to the PBX as if their stations are directly connected to it.

Here are some key points for consideration:

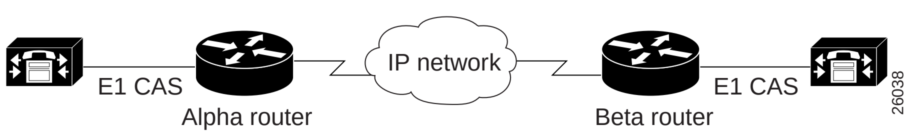
- The configuration includes the **dtmf-relay** command because the users will send DTMF digits to the PBX over the VoIP network, and the router must not compress these digits. The command ensures that the router sends the digits out-of-band, so that they are not distorted.
- The **connection plar** command configures the PLAR connection. The router uses the digits that follow the command internally to send the call to a dial peer—the user does not dial these digits.
- **voice-port 1/0:2** is created by DS0 group 2, as shown in the last digit of the specification. Each DS0 group creates a separate voice port, which allows the definition of individual DS0s on the digital T1 card.

```
hostname router-alpha
!
voice-card 1
!
!
controller T1 1/0
 framing esf
 linecode b8zs
 ds0-group 1 timeslot 1 type fxo-loop
 ds0-group 2 timeslot 2 type fxo-loop
!
dial-peer voice 1 voip
 dtmf-relay  h245-alpha
 codec g729a
 destination-pattern 2..
 session target ipv4:192.168.100.2
!
dial-peer voice 2 pots
 destination-pattern 101
 port 1/0:1
!
dial-peer voice 3 pots
 destination-pattern 102
 port 1/0:2
!
voice-port 1/0:1
 connection plar 201
!
voice-port 1/0:2
 connection plar 202
!
interface s0/0
 ip address 192.168.100.1 255.255.255.0
```

```
hostname router-beta
!
dial-peer voice 1 voip
 destination-pattern 1..
 dtmf-relay h245-alpha
 codec g729a
 session target ipv4:192.168.100.1
!
dial-peer voice 2 pots
 destination-pattern 201
 port 1/1
!
!
dial-peer voice 3 pots
 destination-pattern 202
 port 1/2
!
voice-port 1/1
!
!
voice-port 1 / 2
!
!
interface serial 0/0
 ip address 192.168.100.2 255.255.255.0
```

Connection Trunk Configuration - Permanent VoIP Calls

Figure 10 Sample Configuration: Connection Trunk Permanent VoIP Calls



This example shows how to configure a Cisco 2600 or 3600 router for a trunk connection. A trunk connection is like a “wire” between the two routers. It is a transparent connection, so it allows features such as hookflash (also called *switchhook flash*) or hoot ‘n’ holler (point-to-point) to pass. This type of trunk configuration can also be used for OPXs (Off-Premise Extensions) that require rollover to a centralized voice mail system when the user does not answer.

A trunk connection can only be used between E&M ports or with FXO-to-FXS connections.

```
hostname router-alpha
!
voice-card 1
!
controller T1 1/0
 framing esf
 linecode b8zs
 ds0-group 1 timeslot 1 type e&m-wink
 ds0-group 2 timeslot 2 type e&m-wink
 clock source line
!
voice-port 1/0:1
 connection trunk 1111
!
voice-port 1/0:2
 connection trunk 1112
!
dial-peer voice 1 voip
 dtmf-relay h245-alpha
 codec g729a
 destination-pattern 111.
 session target ipv4:192.168.100.2
!
dial-peer voice 2 pots
 destination-pattern 2221
port 1/0:1
!
dial-peer voice 3 pots
 destination-pattern 2222
 port 1/0:2
!
interface serial 0/0
 ip address 192.168.100.1 255.255.255.0
```

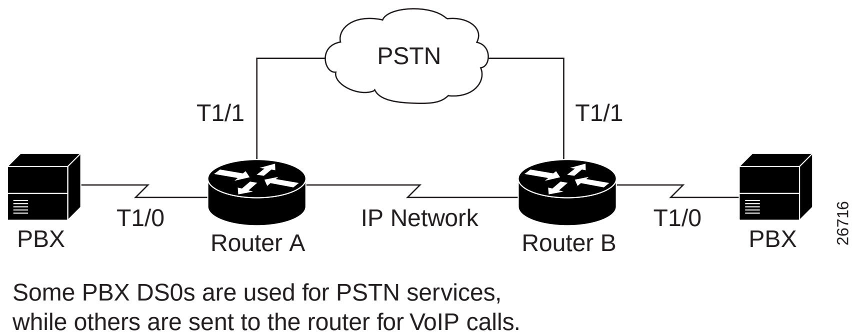
```
hostname router-beta
!
voice-card 1
!
controller T1 1/0
 framing esf
 linecode b8zs
 ds0-group 1 timeslot 1 type e&m-wink
 ds0-group 2 timeslot 2 type e&m-wink
 clock source line
!
voice-port 1/0:1
 connection trunk 2221
!
voice-port 1/0:2
 connection trunk 2222
!
dial-peer voice 1 voip
 dtmf-relay h245-alpha
 codec g729a
 destination-pattern 222.
 session target ipv4:192.168.100.1
!
dial-peer voice 2 pots
 destination-pattern 1111
 port 1/0:1
!
dial-peer voice 3 pots
 destination-pattern 1112
 port 1/0:2
!
interface serial 0/0
 ip address 192.168.100.2 255.255.255.0
```

In this configuration, a permanent and transparent path is set up between individual DS0s on each router. It passes dial tone from the remote PBX and passes DTMF digits out of band.

The **connection trunk** command establishes the permanent trunk connection between the routers. The digits after the command are passed internally within the router to match a dial-peer so that the call can be set up.

Drop-and-Insert Sample Configuration

Figure 11 Sample Configuration: Drop and Insert



Drop-and-Insert technology is one way to integrate old PBX technologies with VoIP. It allows you to take 64Kb DS0 channels from one T1 and digitally cross-connect them to 64Kb DS0 channels on another T1. Drop and Insert is sometimes called TDM Cross-Connect.

Drop and Insert allows individual 64Kb DS0 channels to be transparently passed, uncompressed, between T1 ports without passing through a DSP. Using this method, the channel traffic is sent between a PBX and Central Office switch (PSTN) or other telephony device, allowing the use, for example, of some PBX channels for long-distance service through the PSTN while the router compresses others for interoffice VoIP calls. In addition, Drop and Insert can cross-connect a telephony switch (from the CO or PSTN) to a channel bank to provide external analog connectivity.

Note the following design requirements:

- On the 2600, 3620, and 3640 platforms, Drop and Insert is only permitted between T1 ports on the same multiflex trunk module (MFT). The MFT module can either be in a standalone WIC slot or integrated into a digital T1 packet voice trunk network module VWIC slot.
- When the MFT module is installed in the VWIC slot of a digital T1 packet voice trunk network module, it does not allow the T1 ports to provide WAN connectivity (for example. Frame-Relay, PPP, and so on) in addition to voice and Drop and Insert.
- WAN and Drop-and-Insert capabilities are supported when the MFT is in a standalone WIC slot.

```
hostname RTR-A
!
voice-card 1
  codec complexity high
!
controller T1 1/0
 clock source line
 framing esf
 linecoding b8zs
 ds0-group 1 timeslots 1-12 type e&m-wink
 tdm-group 2 timeslots 13-24 type e&m
!
controller T1 1/1
 clock source line primary
 framing esf
 linecoding b8zs
 tdm-group 3 timeslots 13-24 type e&m
!
voice-port 1/0:1
!
dial-peer voice 1 voip
 destination-pattern 4....
 codec g723r63
 dtmf-relay h245-alpha
 session target ipv4:192.168.100.2
!
dial-peer voice 2 pots
 destination-pattern 5....
 prefix 5
 port 1/0:1
!
interface serial 0/0
 encapsulation ppp
 ip address 192.168.100.1 255.255.255.0
!
connect tdm1 T1 1/0 2 T1 1/1 3
```

```
hostname RTR-B
!
voice-card 1
 codec complexity high
!
controller T1 1/0
 clock source line
 framing esf
 linecoding b8zs
 ds0-group 1 timeslots 1-12 type e&m-wink
 tdm-group 2 timeslots 13-24 type e&m
!
controller T1 1/1
 clock source line primary
 framing esf
 linecoding b8zs
 tdm-group 3 timeslots 13-24 type e&m
!
voice-port 1/0:1
!
dial-peer voice 1 voip
 destination-pattern 5....
 codec g723r63
 dtmf-relay h245-alpha
 session target ipv4:192.168.100.1
!
dial-peer voice 2 pots
 destination-pattern 4....
 prefix 4
 port 1/0:1
!
interface serial 0/0
 encapsulation ppp
 ip address 192.168.100.2 255.255.255.0
!
connect tdm1 T1 1/0 2 T1 1/1 3
```

Here are some key points for consideration:

- The **tdm-group 2 timeslots 13-24 type e&m** command defines Drop and Insert by setting up the timeslots from each T1 that will be used in the digital cross-connect. The **type** keyword is optional, but its use is specific to the Drop and Insert feature.
  - If you include the **type** keyword with a signaling type, the Drop-and-Insert cross-connect ensures that the specified signaling (on-hook and off-hook) is passed between the DS0s. It also uses the signaling bits to signal busy-out if one of the T1s goes down.
  - If you do not use the **type** keyword, the Drop-and-Insert cross-connect is clear-channel and does not interpret any signaling.
- The **connect tdm1 T1 1/0 2 T1 1/1 3** command activates the Drop-and-Insert digital cross-connect between the T1s. The **tdm1** portion of the command is just a name for the cross-connect, and the name can be any word, number, or series of letters.
- You can verify Drop-and-Insert connections by using the **show connect** command.

Command Reference

This section documents new or modified commands. All other commands used with this feature are documented in the Cisco IOS Release 12.0 command references.

- **busyout monitor interface**
- **codec (dial-peer)**
- **codec complexity**
- **connect**
- **ds0-group**
- **echo-cancel coverage**
- **loopback (T1 controller)**
- **show connect**
- **show voice port**
- **voice-card**

busyout monitor interface

To place a voice port into busyout monitor state, enter the **busyout-monitor interface** voice-port configuration command. To remove the busyout monitor state on the voice port, use the **no** form of this command.

busyout-monitor interface *interface number*
no busyout-monitor interface *interface number*

Syntax Description

interface	The name of the associated interface or subinterface that will be monitored to trigger a voice-port busyout, for example **serial**, **atm**, or **ethernet**.
number	The slot and port position of the interface or subinterface, for example, *0/1*, *1/1.0*, and so on.

Default

The voice port is not in busyout monitor state.

Command Mode

Voice-port configuration

Command History

Release	Modification
12.0(3)T	This command was introduced for the Cisco MC3810.
12.0(5)XK and 12.0(7)T	The command was modified for the Cisco 2600 and 3600 series.

Usage Guidelines

When you place a voice port in busyout monitor state, the voice port monitors the specified interface and enters the busyout state when the interface is down. This forces rerouting of calls when an interface is down.

If you specify more than one monitored interface for a voice port, all the monitored interfaces must be down in order to trigger busyout on the voice port.

The command monitors only the up or down status of an interface—not end-to-end TCP/IP connectivity.

When an interface is operational, a busied-out voice port returns to its normal state.

This feature can monitor LAN, WAN, and virtual interfaces, as well as subinterfaces.

Example

The following example configures the voice port to monitor two serial interfaces and an Ethernet interface. When all these interfaces are down, the voice port is busied out. When at least one interface is operating, the voice port is put back into a normal state.

```
voice-port 3/0:0
  busyout monitor interface Ethernet0/0
  busyout monitor interface Serial1/0
  busyout monitor interface Serial2/0
```

codec (dial-peer)

To specify the voice coder rate of speech for a VoIP dial peer, enter the **codec** dial-peer configuration command. Use the **no** form of this command to restore the default value.

codec {**g711alaw** | **g711ulaw** | **g723ar53** | **g723ar63** | **g723r53** | **g723r63** | **g726r16** | **g726r24** | **g726r32** | **g728** | **g729r8** [**pre-ietf**] | **g729br8** } [*bytes*]

no codec

Syntax Description

codec	The voice-card configuration **codec complexity** command sets the codec options that you can use when you execute this command. If you set codec complexity to **high**, the following options are available: • **g711alaw**—G.711 A Law 64,000 bps • **g711ulaw**—G.711 u Law 64,000 bps • **g723ar53**—G.723.1 Annex A 5,300 bps • **g723ar63**—G.723.1 Annex A 6,300 bps • **g723r53**—G.723.1 5,300 bps • **g723r63**—G.723.1 6,300 bps • **g726r16**—G.726 16,000 bps • **g726r24**—G.726 24,000 bps • **g726r32**—G.726 32,000 bps • **g728**—G.728 16,000 bps • **g729r8**---G.729 8,000 bps (default) • **g729br8**—G.729 Annex B 8,000 bps If you set codec complexity to **medium**, the following options are valid: • **g711alaw**—G.711 A Law 64,000 bps • **g711ulaw**—G.711 u Law 64,000 bps • **g726r16**—G.726 16,000 bps • **g726r24**—G.726 24,000 bps • **g726r32**—G.726 32,000 bps • **g729r8**—G.729 Annex A 8,000 bps • **g729br8**—G.729 Annex B with Annex A 8,000 bps
bytes	(Optional) Specifies the voice data bytes per frame. Acceptable values are from 10 to 240 in increments of 10 (10, 20, 30 ... 220, 230, 240). Any other value is rounded down.
pre-ietf	Specifies pre-IETF (Internet Engineering Task Force) bit-ordering. This keyword is valid only on the Cisco 2600, 3600, or AS5300 routers when the **g729r8** codec is specified. You *must* specify this keyword for connection to a Cisco 2600, 3600, or AS5300 router running a Cisco IOS release prior to 12.0(5)T or 12.0(4)XH.

Default

The default is **g729r8**.

Command Mode

Dial-peer configuration

Command History

Release	Modification
11.3(1)T	This command was introduced as a Cisco 3600 VoIP dial-peer configuration command.
12.0(4)T	This command was modified for VoFR dial peers. On the Cisco MC3810, this command was first supported as a dial-peer command.
12.0(5)XK and 12.0(7)T	Additional codec choice and other options were added.

Usage Guidelines

This command applies only to VoIP dial peers.

A specific codec type can be configured on the dial-peer as long as it is supported by the setting used with the **codec complexity** voice-card configuration command.

The dial-peer configuration command is particularly useful when you must change to a small-bandwidth codec. Large-bandwidth codecs, such as G.711, do not fit in a small-bandwidth link. However, **g711alaw** and **g711ulaw** provide higher-quality voice transmission than other codecs. **g729r8**, which provides near-toll quality with considerable bandwidth savings.

If codec values for the VoIP peers of a connection do not match, the call fails.

You can change the payload of each VoIP frame by using the *byte* setting. However, increasing the payload size can add processing delay for each voice packet.

Example

The following example configures a dial peer to use the g723r53 (G.723.1 at 5,300 bps) codec type:

```
dial-peer voice 1 voip
 codec g723r53
```

Related Commands

Command	Description
codec complexity	This voice-card configuration command sets codec complexity and call density. **high** supports the following services: G.711, G.726, G.729, G.729 Annex B, G.723.1, G.723.1 Annex B, G.728, and fax relay. **medium** supports G.711, G.726, G.729 Annex A, G.729 Annex B with Annex A, and fax relay.
show dial-peer voice	Displays the codec setting for dial peers.

codec complexity

Based on the codec standard you are using, enter the **codec complexity** voice-card configuration command to specify call density and codec complexity. High-complexity codecs support lower call density than do medium-complexity codecs. The **no** form of the command resets the voice card to the default.

codec complexity {**high** | **medium**}

no codec complexity

Syntax Description

high	High-complexity codecs support the following services: G.711, G.726, G.729, G.729 Annex B, G.723.1, G.723.1 Annex A, G.728, and fax relay.
medium	Medium-complexity codecs support the following services: G.711, G.726, G.729 Annex A, G.729 Annex B with Annex A, and fax relay.

Default

The default is **medium**.

Command Mode

Voice-card configuration

Command History

Release	Modification
12.0(5)XK and 12.0(7)T	The command was introduced for the Cisco 2600 and 3600 series.

Usage Guidelines

Codec complexity refers to the amount of processing required in order to perform compression. Codec complexity affects the number of calls that can take place on a voice card's digital signal processors (DSPs), referred to as call density. The greater the codec complexity, the fewer calls are handled. For example, G.711 requires less DSP processing than G.728, so that as long as the bandwidth is available, more calls can be handled simultaneously by using the G.711 standard than using G.728.

All voice cards in a router must use the same codec complexity. The voice-card **codec complexity** setting affects the options available for the **codec** dial-peer configuration command.

To change codec complexity, you must first remove any configured CAS or DS0 groups, then reinstate them after the change.

Example

The following example configures a voice card for high-complexity codecs:

```
voice-card 1
 codec complexity high
```

Related Command

Command	Description
ds0-group	Controller configuration command that defines the T1 channels for compressed voice calls and the signaling method by which the router connects to the PBX or PSTN. Before you can change codec complexity, you must remove any DS0 groups that are already configured; then, re-create them after making the change.

connect

To define connections between T1 or E1 controller ports for Drop and Insert (also called *TDM Cross-Connect*), enter the **connect** global configuration command.

connect *id* {**t1** | **e1**} *slot/port-1 tdm-group-no-1* {**t1** | **e1**} *slot/port-2 tdm-group-no-2*

no connect *id* {**t1** | **e1**} *slot/port-1 tdm-group-no-1* {**t1** | **e1**} *slot/port-2 tdm-group-no-2*

Syntax Description

id	A name for this connection
T1	Specifies a T1 port.
E1	Specifies an E1 port.
slot/port-1	The location of the first T1 or E1 controller to be connected. Valid values for *slot* and *port* are 0 and 1.
tdm-group-no-1	The number identifier of the time-division multiplexing (TDM) group associated with the first T1 or E1 controller port and created by using the **tdm-group** command. Valid values are from 0 to 23 for T1 and from 0 to 30 for E1.
slot/port-2	The location of the second T1 or E1 controller port to be connected. Valid values for *slot* are from 0 to 5 depending on the platform. Valid values for *port* are 0 to 3 depending on the platform and the presence of a network module.
tdm-group-no-2	The number identifier of the time-division multiplexing (TDM) group associated with the second T1 or E1 controller and created by using the **tdm-group** command. Valid values are from 0 to 23 for T1 and from 0 to 30 for E1.

Default

There is no Drop-and-Insert connection between the ports.

Command Mode

Global configuration

Command History

Release	Modification
12.0(5)XK and 12.0(7)T	The command was introduced.

Usage Guidelines

The **connect** command creates a named connect between two TDM groups associated with Drop-and-Insert ports on T1 or E1 interfaces where the user has already defined the groups by using the **tdm-group** command.

Example

The following example shows how two T1 TDM groups are set up and then connected:

```
Router(config)# controller T1 1/0
Router(config-controller)tdm-group 2 timeslots 13-24 type e&m
Router(config-controller)# controller T1 1/1
Router(config-controller)tdm-group 3 timeslots 13-24 type e&m
Router(config-controller)exit
Router(config)connect tdm1 T1 1/0 2 T1 1/1 3
```

Related Command

Command	Description
show connect	This command shows the status of current Drop-and-Insert connections that have been set up by using the **connect** command.
tdm-group	This controller configuration command creates TDM groups that can be connected for Drop-and-Insert functionality.

ds0-group

To define T1 channels for compressed voice calls and the channel-associated signaling (CAS) method by which the router connects to the PBX or PSTN, enter the **ds0-group** controller configuration command. The **no** form of the command removes the group and signaling setting.

ds0-group *ds0-group-no* **timeslots** *timeslot-list* **type** {**e&m-immediate** | **e&m-delay** | **e&m-wink** | **fxs-ground-start** | **fxs-loop-start** | **fxo-ground-start** | **fxo-loop-start**}

no ds0-group *ds0-group-no*

Syntax Description

ds0-group-no	A value from 0 to 23 that identifies the DS0 group
timeslot-list	*timeslot-list* is a single timeslot number, a single range of numbers, or multiple ranges of numbers separated by commas. For T1, allowable values are from 1 to 24. Examples are: • 2 • 1-15, 17-24 • 1-23 • 2, 4, 6-12
type	The signaling method selection for **type** depends on the connection that you are making. The E&M interface allows connection for PBX trunk lines (tie lines) and telephone equipment. The FXS interface allows connection of basic telephone equipment and PBXes. The FXO interface is for connecting the central office (CO) to a standard PBX interface where permitted by local regulations; it is often used for off-premises extensions. The options are as follows: • **e&m-immediate** specifies no specific offhook and onhook signaling. • **e&m-delay** specifies that the originating endpoint sends an offhook signal and then and waits for an offhook signal followed by an onhook signal from the destination. • **e&m-wink** specifies that the originating endpoint sends an offhook signal and waits for a wink signal from the destination. • **fxs-ground-start** specifies Foreign Exchange Station ground-start signaling support. • **fxs-loop-start** specifies Foreign Exchange Station loop-start signaling support. • **fxo-ground-start** specifies Foreign Exchange Office ground-start signaling support. • **fxo-loop-start s**pecifies Foreign Exchange Office loop-start signaling support.

Default

There is no DS0 group.

Command Mode

Controller configuration

Command History

Release	Modification
11.3 MA	The command was introduced as the **voice-group** command for the Cisco MC3810 multiservice access concentrator.
12.0(5)XK and 12.0(7)T	The command was introduced for the Cisco 2600 and 3600 series with a different name and some keyword modifications.

Usage Guidelines

The **ds0-group** command automatically creates a logical voice port that is numbered as follows on Cisco 2600 and 3600 series routers: *slot*/*port*:*ds0-group-no*. Although only one voice port is created for each group, applicable calls are routed to any channel in the group.

Example

The following example configures ranges of T1 controller timeslots for FXS ground-start and FXO loop-start signaling:

```
controller T1 1/0
 framing esf
 linecode b8zs
 ds0-group 1 timeslot 1-10 type fxs-ground-start
 ds0-group 2 timeslot 11-24 type fxo-loop-start
```

Related Command

Command	Description
codec complexity	To change codec complexity by using this voice-card configuration command, you must first remove any configured CAS or DS0 groups; then, reinstate them after the change.

echo-cancel coverage

To adjust the size of the echo canceller, use the **echo-cancel coverage** voice-port configuration command. Use the **no** form of this command to reset this command to the default value.

echo-cancel coverage {**8** | **16** | **24** | **32**}

no echo-cancel coverage

Syntax Description

8	8 milliseconds
16	16 milliseconds
24	24 milliseconds
32	24 milliseconds

Default

16 milliseconds

Command Mode

Voice-port configuration

Command History

Release	Modification
11.3(1)T	The command was introduced.
12.0(5)XK and 12.0(7)T	The command was modified to add the 8-millisecond option.

Usage Guidelines

Use the **echo-cancel coverage** command to adjust the coverage size of the echo canceller. This command enables cancellation of voice that is sent out the interface and received back on the same interface within the configured amount of time. If the local loop (the distance from the interface to the connected equipment producing the echo) is longer, the configured value of this command should be extended.

If you configure a longer value for this command, it takes the echo canceller longer to converge; in this case, the user might hear slight echo when the connection is initially set up. If the configured value for this command is too short, the user may hear some echo for the duration of the call because the echo canceller is not cancelling the longer-delay echoes.

There is no echo or echo cancellation on the network (for example, non-POTS) side of the connection.

Note This command is valid only if the echo cancel feature has been enabled. For more information, see the **echo-cancel enable** command.

Example

The following example adjusts the size of the echo canceller to 8 milliseconds on the Cisco 3600 series:

```
voice-port 1/0:0
 echo-cancel enable
 echo-cancel coverage 8
```

Related Command

Command	Description
echo-cancel enable	Activates the echo canceller.

loopback (T1 controller)

To set the loopback method for testing the T1 interface, enter the **loopback** controller configuration command. Use the **no** form of this command to restore the default value.

loopback {diagnostic | local {payload | line} | remote {iboc | esf {payload | line}}

no loopback

Syntax Description

diagnostic	Loops the outgoing transmit signal back to the receive signal
line	Places the interface into external loopback mode at the line.
local	Places the interface into local loopback mode.
payload	Places the interface into external loopback mode at the payload level.
remote	Keeps the local end of the connection in remote loopback mode.
iboc	Sends an in-band bit-oriented code to the far-end to cause it to go into line loopback.
esf	Specifies extended super frame as the T1 or E1 frame type.

Default

No loopback is configured.

Command Mode

Controller configuration

Command History

Release	Modification
11.3 MA	This command was introduced as a controller configuration command for the Cisco MC3810.
12.0(5)T and 12.0(5)XK	The command was introduced as an ATM interface configuration command for the Cisco 2600 and 3600 series.
12.0(5)XE	The command was introduced as an ATM interface configuration command for the Cisco 7200 and 7500 series.
12.0(7)T	The command was introduced as a controller configuration command for the Cisco 2600 and 3600 series.

Usage Guidelines

You can use a loopback test on lines to detect and distinguish equipment malfunctions caused either by line and Channel Service Unit/Digital Service Unit (CSU/DSU) or by the interface. If correct data transmission is not possible when an interface is in loopback mode, the interface is the source of the problem.

Example

The following example shows how to set the diagnostic loopback method on controller T1 0/0:

```
Router(config)# controller t1 0/0
loopback diagnostic
```

show connect

To display configuration information about Drop-and-Insert connections that have been configured on a router, enter the **show connect** privileged EXEC command.

show connect {**all** | **elements** | **name** | **id** | **port** { **T1** | **E1** } *slot/port* }}

Syntax Description

all	Shows a table of all configured connections.
elements	Shows registered hardware or software interworking elements.
name	Displays a connection that has been named by using the **connect** global configuration command. The name you enter is case-sensitive and must match the configured name exactly.
id	Displays the status of a connection that you specify by an identification number or range of identification numbers. The router assigns these IDs automatically in the order that they were created, beginning with 1. The **show connect all** command displays these IDs.
port	Displays the status of a connection that you specify by indicating the type of controller (T1 or E1) and location of the interface.
T1	Specifies a T1 controller.
E1	Specifies an E1 controller.
slot/port	The location of the T1 or E1 controller port whose connection status you want to see. Valid values for *slot* and *port* are 0 and 1.

Default

There is no default.

Command History

Release	Modification
12.0(5)XK and 12.0(7)T	This command was introduced.

Usage Guidelines

This command shows Drop and Insert connections on the Cisco 2600 and 3600 series. The **elements** keyword is not supported in Cisco IOS Releases 12.0(5)XK and 12.0(7)T.

The command displays different information in different formats depending on the keyword that you use.

Examples

The following examples show how different keywords affect the display of information.

These example commands show how the same tabular information appears when you enter different keywords:

```
Router# show connect all
ID   Name               Segment 1           Segment 2           State
=========================================================================
1    Test              -T1 1/0 01          -T1 1/1 02           ADMIN UP
2    Test2             -T1 1/0 03          -T1 1/1 04           ADMIN UP

Router# show connect id 1-2
ID   Name               Segment 1           Segment 2           State
=========================================================================
1    Test              -T1 1/0 01          -T1 1/1 02           ADMIN UP
2    Test2             -T1 1/0 03          -T1 1/1 04           ADMIN UP

Router# show connect port t1 1/1
ID   Name               Segment 1           Segment 2           State
=========================================================================
1    Test              -T1 1/0 01          -T1 1/1 02           ADMIN UP
2    Test2             -T1 1/0 03          -T1 1/1 04           ADMIN UP
```

These example commands show details about specific connections, including the number of timeslots in use and the switching elements.

```
Router# show connect id 2
Connection: 2 - Test2
 Current State: ADMIN UP
 Segment 1: -T1 1/0 03
  TDM timeslots in use: 14-18 (5 total)
 Segment 2: -T1 1/1 04
  TDM timeslots in use: 14-18
Internal Switching Elements: VIC TDM Switch

Router# show connect name Test
 Connection: 1 - Test
 Current State: ADMIN UP
 Segment 1: -T1 1/0 01
  TDM timeslots in use: 1-13 (13 total)
 Segment 2: -T1 1/1 02
  TDM timeslots in use: 1-13
Internal Switching Elements: VIC TDM Switch
```

Table 5 shows descriptions of the command output fields.

Table 5 show connect Fields

Field	Description
ID	ID automatically assigned to this connection.
Internal Switching Elements	Hardware component that enables the switched connection.
Name	Name for the connection, specified using the connect command.
Segment 1	T1 or E1 controller location and time-division multiplexing (TDM) group number for the first segment of the connection.
Segment 2	T1 or E1 controller location and time-division multiplexing (TDM) group number for the second segment of the connection.
State	Operational status of the connection.

Related Commands

Command	Description
connect	Configures Drop-and-Insert connections.
tdm-group	This controller configuration command creates TDM groups that can be connected for Drop-and-Insert functionality by using the **connect** command.

show voice port

To display configuration information about a specific digital voice port, enter the **show voice port** privileged EXEC command.

show voice port *slot*/*port***:***ds0-group*

Syntax Description

slot	Slot number in the Cisco router where the voice interface card is installed. Valid entries are from 0 to 3, depending on the slot where it has been installed.
port	Indicates the voice interface card location. Valid entries are 0 or 1.
ds0-group-no	A value from 0 to 23 that identifies the DS0 group for the voice port.

Default

There is no default.

Command History

Release	Modification
11.3(1)T	The command was introduced.
12.0(5)XK and 12.0(7)T	Additional syntax was created for digital voice on the Cisco 2600 and 3600 series to allow specification of the DS0 group.

Usage Guidelines

This command applies to VoIP on the Cisco 2600 and 3600 series.

The **ds0-group** command automatically creates a logical voice port that is numbered as follows on Cisco 2600 and 3600 series routers: *slot*/*port*:*ds0-group-no*. Although only one voice port is created for each group, applicable calls are routed to any channel in the group.

Example

The following displays voice port configuration information for the digital voice port 0 located in slot 1, DS0 group 1:

```
cisco-router# show voice port 1/0:1

receEive and transMit Slot is 1, Sub-unit is 0, Port is 1
 Type of VoicePort is E&M
 Operation State is DORMANT
 Administrative State is UP
 No Interface Down Failure
 Description is not set
 Noise Regeneration is enabled
 Non Linear Processing is enabled
 Music On Hold Threshold is Set to -38 dBm
 In Gain is Set to 0 dB
 Out Attenuation is Set to 0 dB
 Echo Cancellation is enabled
```

```
Echo Cancel Coverage is set to 8 ms
Connection Mode is normal
Connection Number is not set
Initial Time Out is set to 10 s
Interdigit Time Out is set to 10 s
Region Tone is set for US
```

Related Command

Command	Description
ds0-group	Defines T1 channels for compressed voice calls and the channel-associated signaling (CAS) method by which the router connects to the PBX or PSTN.

voice-card

To configure a voice card and enter voice-card configuration mode, enter the **voice-card** command.

voice-card *slot*

Syntax Description

slot	A value from 0 to 3 that describes the card location in the module

Command Mode

Global configuration

Command History

Release	Modification
12.0(5)XK and 12.0(7)T	The command was introduced for the Cisco 2600 and 3600 series.

Usage Guidelines

The command is used to enter voice-card configuration mode and set codec complexity.

Example

The following example enters voice-card configuration mode for the card in Slot 1:

```
voice-card 1
```

Related Command

Command	Description
codec complexity	Change codec complexity using by this voice-card configuration command.

Glossary

AAL—ATM Adaptation Layer. Service-dependent sublayer of the data link layer. The AAL accepts data from different applications and presents it to the ATM layer in the form of 48-byte ATM payload segments. AALs consist of two sublayers: convergence sublayer (CS) and segmentation and reassembly (SAR). AALs differ on the basis of the source-destination timing used, whether they use constant bit rate (CBR) or variable bit rate (VBR), and whether they are used for connection-oriented or connectionless mode data transfer. At present, the four types of AAL recommended by the ITU-T are AAL1, AAL2, AAL3/4, and AAL5.

AAL1—ATM adaptation layer 1. One of four AALs recommended by the ITU-T. AAL1 is used for connection-oriented, delay-sensitive services requiring constant bit rates, such as uncompressed video and other isochronous traffic.

AMI—alternate mark inversion. Line-code type used on T1 and E1 circuits. In AMI, zeros are represented by 01 during each bit cell, and ones are represented by 11 or 00, alternately, during each bit cell. AMI requires that the sending device maintain ones density. Ones density is not maintained independent of the data stream. Sometimes called *binary coded alternate mark inversion*.

ATM—Asynchronous Transfer Mode. International standard for cell relay in which multiple service types (such as voice, video, or data) are conveyed in fixed-length (53-byte) cells. Fixed-length cells allow cell processing to occur in hardware, thereby reducing transit delays. ATM is designed to take advantage of high-speed transmission media such as E3, SONET, and T3.

B8ZS—binary 8-zero substitution. Line-code type, used on T1 and E1 circuits, in which a special code is substituted whenever 8 consecutive zeros are sent over the link. This code is then interpreted at the remote end of the connection. This technique guarantees ones density independent of the data stream.

CAS—channel-associated signaling. Trunk signaling (for example, in a T1 line) in which control signals, such as those for synchronizing and bounding frames, are carried in the same channel along with voice and data signals.

CBR—constant bit rate. QoS class defined by the ATM Forum for ATM networks. CBR is used for connections that depend on precise clocking to ensure undistorted delivery.

CCS—common channel signaling. Trunk signaling (for example, using Primary Rate Interface) in which a control channel carries signaling for separate voice and data channels.

CES—circuit emulation service. Enables users to multiplex or concentrate multiple circuit emulation streams for voice and video with packet data on a single high-speed ATM link without a separate ATM access multiplexer.

CO—central office. Local telephone company office to which all local loops in a given area connect and in which circuit switching of subscriber lines occurs.

codec—Coder-decoder. Device that typically uses pulse code modulation to transform analog signals into a digital bit stream and digital signals back into analog.

DTMF—Dual-tone multifrequency. Use of two simultaneous voice-band tones for dialing (such as touch tone).

Drop and Insert—(also called TDM Cross-Connect) Allows DSO channels from one T1 or E1 facility to be digitally cross-connected to DS0 channels on another T1 or E1. Using this method, channel traffic is sent between a PBX and CO PSTN switch or other telephony device, so that some PBX channels are directed for long-distance service through the PSTN while the router compresses others for interoffice VoIP calls. In addition, Drop and Insert can cross-connect a telephony switch (from the CO or PSTN) to a channel bank for external analog connectivity.

DSP—digital signal processor, same as PVDM

E1—European digital carrier facility used for transmitting data through the telephone hierarchy. The transmission rate for E1 is 2.048 megabits per second (Mbps).

E&M—rEceive and transMit, or Ear and Mouth. Type of signaling originally developed for analog two-state voltage telephony using the ear and mouth leads; in digital telephony, uses two bits.

ESF—Extended Superframe. Framing type used on T1 circuits that consists of 24 frames of 192 bits each, with the 193rd bit providing timing and other functions. ESF is an enhanced version of SF format.

FXO—Foreign Exchange Office. A voice interface emulating a PBX trunk line to a switch or telephone equipment to a PBX extension interface.

FXS—Foreign Exchange Station. A voice interface for connecting telephone equipment, emulates the extension interface of a PBX or the subscriber interface for a switch.

IETF—Internet Engineering Task Force

ISDN—Integrated Services Digital Network. Communication protocol, offered by telephone companies, that permits telephone networks to carry data, voice, and other source traffic.

IVR—interactive voice response. Term used to describe systems that provide information in the form of recorded messages over telephone lines in response to user input in the form of spoken words or more commonly DTMF signaling. Examples include banks that allow you to check your balance from any telephone and automated stock quote systems.

packet—Logical grouping of information that includes a header containing control information and (usually) user data. Packets are most often used to refer to network layer units of data.

POTS—plain old telephone service

PDVM—packet data voice module

PSTN—Public Switched Telephone Network. General term referring to the variety of telephone networks and services in place worldwide.

QoS—quality of service. Measure of performance for a transmission system that reflects its transmission quality and service availability.

SF—Super Frame. Common framing type used on T1 circuits. SF consists of 12 frames of 192 bits each, with the 193rd bit providing error checking and other functions. SF is superseded by ESF, but is still widely used. Also called D4 framing.

SNMP—Simple Network Management Protocol. Network management protocol used almost exclusively in TCP/IP networks. SNMP provides a means to monitor and control network devices, and to manage configurations, statistics collection, performance, and security.

T1—Digital WAN carrier facility. T1 transmits DS 1-formatted data at 1.544 Mbps through the telephone switching network, using alternate mark inversion or B8ZS coding.

T1 trunk—Digital WAN carrier facility. See T1.

TDM—time-division multiplexing

Trunk—Physical and logical connection between two switches across which network traffic travels. A backbone is composed of a number of trunks.

UNI—User-Network Interface. ATM Forum specification that defines an interoperability standard for the interface between ATM-based products (a router or an ATM switch) located in a private network and the ATM switches located within the public carrier networks. Also used to describe similar connections in Frame Relay networks.

VAD—voice activity detection