CESM−8 Structured Data Transport

Document ID: 22237

Contents

Introduction

This document provides configuration examples for **structured** data transport on the Circuit Emulation Service Module−8 (CESM−8) card.

Prerequisites

Requirements

Before attempting this configuration, ensure that you are knowledgeable of:

- Cisco CESM−8
- Cisco MGX 8220
- Cisco MC3810

Components Used

The information in this document is based on these software and hardware versions:

- MGX 8220/8250 firmware 4.1.x and later supports the CESM−8T1E1 card set

All configuration examples in this document are based on a CESM−8 card with this firmware/bootcode/hardware revision:

```
wss−mgxb.1.10.CESM.a > dspcd
  ModuleSlotNumber:          10
  FunctionModuleState:       Active
  FunctionModuleType:        CESM−8T1
  FunctionModuleSerialNum:   754950
  FunctionModuleHWRev:       aa
  FunctionModuleFWRev:        4.1.01
  FunctionModuleResetReason: Local DRAM parity reset
```

```
  LineModuleType:            LM-RJ48-8T1
  LineModuleState:           Present
  mibVersionNumber:          20
  configChangeTypeBitMap:    CardCnfChng, LineCnfChng
  cardIntegratedAlarm:       Clear

  fab number:                28-2199-02
```

The MC3810 multiservice access concentrator units used in the configuration examples as the customer premises equipment (CPE) devices are based on this software/hardware release level:

```
wss-3810e# show ver
Cisco Internetwork Operating System Software
IOS (tm) MC3810 Software (MC3810-A2INR3V2-M), Version 11.3(1)MA62, EARLY DEPLOYMENT MAINTE
Copyright (c) 1986-1998 by cisco Systems, Inc.
Compiled Mon 26-Oct-98 19:35 by krunyan
Image text-base: 0x00023000, data-base: 0x0064BFDC

ROM: System Bootstrap, Version 11.3(1)MA1, MAINTENANCE INTERIM SOFTWARE
ROM: MC3810 Software (MC3810-WBOOT-M), Version 11.3(1)MA1,  MAINTENANCE INTERIM SOFTWARE

wss-3810e uptime is 1 minute
System restarted by reload
System image file is "flash:mc3810-a2inr3v2-mz.113-1.MA62", booted via flash:

Cisco MC3810 (MPC860) processor (revision 06.06) with 27648K/5120K bytes of memory.
Processor board ID 08465557
PPC860 PowerQUICC, partnum 0x0000, version A03(0x0013)
Bridging software.
MC3810 SCB board (v04.K0)
1 Multiflex T1(slot 3) RJ45 interface(v01.K0)
1 Six-Slot Analog Voice Module (v03.K0)
1    Analog FXS voice interface (v03.K0) port 1/1
1    Analog FXS voice interface (v03.K0) port 1/2
1    Analog FXO voice interface (v03.K0) port 1/6
1 6-DSP(slot2) Voice Compression Module(v01.K0)
1 Ethernet/IEEE 802.3 interface(s)
4 Serial network interface(s)
1 Channelized T1/PRI port(s)
256K bytes of non-volatile configuration memory.
8192K bytes of processor board System flash (AMD29F016)

Configuration register is 0x2102

wss-3810e#
```

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, make sure that you understand the potential impact of any command.

Conventions

For more information on document conventions, refer to the Cisco Technical Tips Conventions.

Background Information

The CESM−8 card offers a 100 percent port density increase over the CESM−4 card, as well as various clocking and framing functional enhancements. The CESM−4 card supports only T1/E1 basic unstructured service with synchronous clocking. The CESM−8 provides both basic unstructured service and basic/CAS structured service.

This document explores only the structured **structured** service features of the Circuit Emulation Service Module−8 (CESM−8) card n * digital signal level 0s (DS0s) transport, channel associated signaling (CAS), and on−hook detection.

- The ATM Forum

  af−vtoa−0078.000 Circuit Emulation Service Interoperability Specification Version 2.0 (January 1997)
- International Telecommunication Union (ITU)

  ITU−T I.363.1 BISDN ATM Adaptation Layer specification: Type 1 AAL (August 1996)

Configure

The CESM−8 supports both T1 and E1 lines. Both line types are configured using the same set of commands.

- Lines are the first entity to be configured on the card.
- Next, logical ports are configured and associated with the active lines. These ports define a range of digital signal level 0s (DS0s) to use on the card and the type of CES to use (structured or unstructured).
- Finally, channels are created for each port to route the data through the ATM network.

This section describes these configurations:

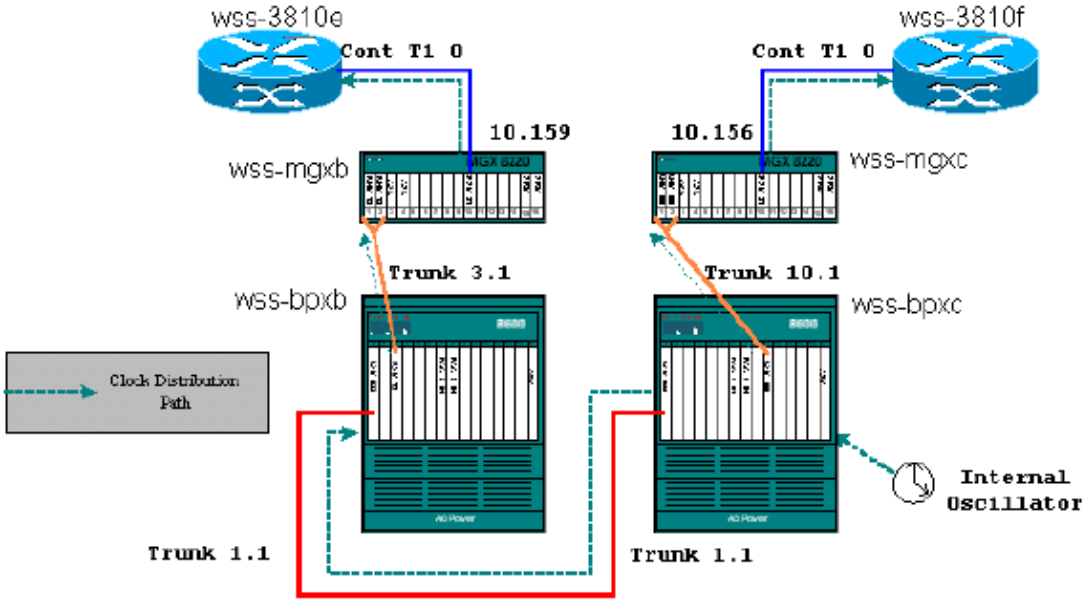
- Basic
- Channel Associated Signaling
- On−Hook Detection

Note: To find additional information on the commands used in this document, use the Command Lookup Tool (registered customers only) .

Basic

In this configuration example, we connect the MC3810 units WSS−3810E and WSS−3810F via the MultiFlex Trunk (MFT) modules (Controller T1 0/Serial 2) using HDLC as the Layer 2 protocol. Each MFT connects into line 5 on their respective CESM cards. However, we dedicate only three DS0s from the T1 to the HDLC connection. All devices in the test network derive their timing from the internal oscillator on WSS−BPXC.

This example uses this network diagram:



1. CESM Build Lines

   The first step in building the connection is to bring up the lines on the two CESM cards. The line configuration covers the physical layer parameters for the associated T1/E1 line.

MGX−B

```
wss−mgxb.1.10.CESM.a > addln 5

wss−mgxb.1.10.CESM.a > dsplns
  Line   Conn       Type      Status/Coding      Length      XmtClock Alarm Stats
         Type                                               Source         Alarm
  ----   -----  ------------ ------ -------- ------------- -------- ----- -----
  10.1   RJ−48 dsx1ESF          Dis/dsx1B8ZS 0−131 ft      LocalTim
  10.2   RJ−48 dsx1ESF          Dis/dsx1B8ZS 0−131 ft      LocalTim
  10.3   RJ−48 dsx1ESF          Dis/dsx1B8ZS 0−131 ft      LocalTim
  10.4   RJ−48 dsx1ESF          Dis/dsx1B8ZS 0−131 ft      LocalTim
  10.5   RJ−48 dsx1ESF          Ena/dsx1B8ZS 0−131 ft      LocalTim  No    No
  10.6   RJ−48 dsx1ESF          Dis/dsx1B8ZS 0−131 ft      LocalTim
  10.7   RJ−48 dsx1ESF          Dis/dsx1B8ZS 0−131 ft      LocalTim
  10.8   RJ−48 dsx1ESF          Dis/dsx1B8ZS 0−131 ft      LocalTim

wss−mgxb.1.10.CESM.a > dspln 5
  LineNum:                    5
  LineConnectorType:          RJ−48
  LineEnable:                 Enabled
  LineType:                   dsx1ESF
  LineCoding:                 dsx1B8ZS
  LineLength:                 0−131 ft
  LineXmtClockSource:         LocalTiming
  LineLoopbackCommand:        NoLoop
  LineSendCode:               NoCode
  LineUsedTimeslotsBitMap:    0x0
  LineLoopbackCodeDetection: codeDetectDisabled
```

MGX−C

```
wss−mgxc.1.10.CESM.a > addln 5
```

2. CESM Configure Ports

   Next, we add the logical ports. Ports provide a logical grouping for a series of DS0s on a particular

line and define the CES mode. Structured service allows encompassing any sequential series of DS0s on the line. You are not limited to one logical port associated with each line, but two logical ports cannot both use the same DS0s on a line.

MGX−B

```
wss−mgxb.1.10.CESM.a > addport 59 5 9 3 1

wss−mgxb.1.10.CESM.a > dspports
  Port     Ena/Speed   Type
  ------   --- ------ --------
  10.5.59  Add/ 192k structur

  Number of ports:    1

  PortDs0UsedLine1:              0x00000000
  PortDs0UsedLine2:              0x00000000
  PortDs0UsedLine3:              0x00000000
  PortDs0UsedLine4:              0x00000000
  PortDs0UsedLine5:              0x00000700
  PortDs0UsedLine6:              0x00000000
  PortDs0UsedLine7:              0x00000000
  PortDs0UsedLine8:              0x00000000
  PortNumNextAvailable:          10

wss−mgxb.1.10.CESM.a > dspport 59
  SlotNum:                       10
  PortLineNum:                   5
  PortNum:                       59
  PortRowStatus:                 Add
  PortNumOfSlots:                3
  PortDs0ConfigBitMap(1stDS0):   0x700(9)
  PortSpeed:                     192kbps
  PortType:                      structured
  PortState:                     Active
```

MGX−C

```
wss−mgxc.1.10.CESM.a > addport 56 5 6 3 1
```

3. CESM Add Channels

The logical channel controls the ATM side parameters for the connection. You link one logical channel to one logical port. Below, we configure the ports for basic service. The HDLC frames traversing the link do not require CAS signaling. As we will see later, configuring the line for CAS reduces bandwidth on the link. We also set the conditioning criteria to send all 1s if the channel experiences cell loss. We then configure the channel buffer size, CDVT, and clocking values.

MGX−B

```
wss−mgxb.1.10.CESM.a > addchan 159 59 1 0 255 15

wss−mgxb.1.10.CESM.a > cnfchan 159 1000 1000 0 1 1 1

wss−mgxb.1.10.CESM.a > dspchans
    Channel     ChanNum   Status   CDV   MaxBufSize   CLIP    CBRservice
  ------------  -------   ------  -----  ----------  ------  ------------
  10.5.59.159     159      Mod     1000      384      1000   structured

wss−mgxb.1.10.CESM.a > dspchan 159
  ChanNum:                  159
  ChanRowStatus:            Mod
  ChanLineNum:              5
```

```
  ChanMapVpi:               10
  ChanMapVci:               159
  ChanCBRService:           structured
  ChanClockMode:            Synchronous
  ChanCAS:                  Basic
  ChanPartialFill:          47
  ChanMaxBufSize:           384 bytes
  ChanCDV:                  1000 micro seconds
  C L I P:                  1000 milliseconds
  ChanLocalRemoteLpbkState: Disabled
  ChanTestType:             TestOff
  ChanTestState:            NotInProgress
  ChanRTDresult:            65535 ms
  ChanPortNum                  59
  ChanConnType              PVC
  ISDetType                 DetectionDisabled
  CondData                  255
  CondSignalling            15
  ExtISTrig                 DisableIdleSupression
  ISIntgnPeriod             4095 seconds
  ISSignallingCode          0
  OnHookCode                1
```

MGX−C

```
wss−mgxc.1.10.CESM.a > addchan 156 56 1 0 255 15

wss−mgxc.1.10.CESM.a > cnfchan 156 1000 1000 0 1 1 1
```

4. CESM Examine Channel Statistics

Looking at the channel statistics at this stage of the configuration, we see some definite problems. The associated channel on each card appears to be sending cells into the network, but neither channel reports receiving any cells, which results in the alarm condition for the channel state. This is happening because we have not built the PVC to transport the AAL1 cells from the local CESM card to the remote CESM card.

MGX−B

```
wss−mgxb.1.10.CESM.a > clrchancnts

wss−mgxb.1.10.CESM.a > clrsarcnts

wss−mgxb.1.10.CESM.a > dspchancnt 159
ChanNum:                  159
Chan State:               alarm
Chan RCV ATM State:       Normal
Chan XMT ATM State:       Normal
Cell Loss Status:         Cell Loss
Reassembled Cells:        0
Generated Cells:          4802
Header Errors:            0
Seqence Mismatches :      0
Lost Cells:               0
Channel Uptime (secs.)    46
Signalling Status         Offhook

wss−mgxb.1.10.CESM.a > dspsarcnt 159
                         SarShelfNum:         1
                         SarSlotNum:          10
                         SarChanNum:          159
                               Tx                    Rx
                         ---------------       ---------------
  Total Cells:           9717                   0
```

```
  Total CellsCLP:         0                    0
  Total CellsAIS:         0                    0
  Total CellsFERF:        0                    0
  Total CellsEnd2EndLpBk: 0                    0
  Total CellsSegmentLpBk: 0                    0
  RcvCellsDiscOAM:                             0
```

5. BPX Add CBR Connection

In order to carry the cells from one CESM to the other, there needs to be a CBR connection from the BXM trunk cards off of which each MGX shelf hangs. Because this PVC will carry an n * DS0 structured stream without partial fill, we configure the cell rate as 512 cells per second.

`{ [8000 * N] / 46.875} = 512 cells/sec`, where N = 3 and {x} means smallest integer >= to x.

BPX–B

```
wss-bpxb       TN    StrataCom       BPX 8600  9.1.04    Nov. 18 1998 13:53 EST
                   BPX 8600 Interface Shelf Information

Trunk    Name      Type          Part ICtrl Id   Alarm
 3.1     wss-mgxb  AXIS             -     -      MIN

Last Command: dspnode
```

BPX–C

```
wss-bpxc       TN    StrataCom       BPX 8600  9.1.04    Nov. 18 1998 13:54 EST

                   BPX 8600 Interface Shelf Information

Trunk    Name      Type          Part ICtrl Id   Alarm
10.1     wss-mgxc  AXIS             -     -      MIN

Last Command: dspnode
```

BPX–B

```
wss-bpxb       TN    StrataCom       BPX 8600  9.1.04    Nov. 18 1998 13:59 EST

 Local           Remote      Remote                            Route
 Channel         NodeName    Channel         State  Type       Avoid COS O
 3.1.10.159      wss-bpxc    10.1.10.156     Ok     cbr              0  L

Last Command: addcon 3.1.10.159 wss-bpxc 10.1.10.156 cbr 512 * 1000 * Y
------------------------------------------------------------------------

wss-bpxb       TN    StrataCom       BPX 8600  9.1.04    Nov. 18 1998 13:59 EST

Conn:  3.1.10.159       wss-bpxc    10.1.10.156       cbr       Status:OK
   PCR(0+1)     % Util    CDVT(0+1)     Policing
    512/512    100/100    1000/1000      4/4
Owner: LOCAL  Restriction: NONE  COS: 0
Trunk Cell Routing Restrict: Y
Path:   wss-bpxb 1.1-- 1.1wss-bpxc
Pref:   Not Configured

wss-bpxb      BXM       : OK           wss-bpxc  BXM       : OK
              Line  3.1 : OK                     Line 10.1 : OK
              OAM Cell RX: Clear                 NNI       : OK
              NNI       : OK
```

```
Conn:  3.1.10.159       wss-bpxc    10.1.10.156       cbr       Status:OK
   PCR(0+1)     % Util       CDVT(0+1)         Policing
  512/512      100/100       1000/1000             4

Trunk Cell Routing Restrict: Y

This Command: dspcon 3.1.10.159
```

6. CESM Monitor Connection Statistics

Now the channel counters on the CESM card indicate much improvement. Notice that the connections are not in alarm, and that the cells in and out are, for all intents and purposes, the same.

MGX-B

```
wss-mgxb.1.10.CESM.a > clrchancnts

wss-mgxb.1.10.CESM.a > clrsarcnts

wss-mgxb.1.10.CESM.a > dspchancnt 159
ChanNum:                    159
Chan State:                    okay
Chan RCV ATM State:         Normal
Chan XMT ATM State:         Normal
Cell Loss Status:           No Cell Loss
Reassembled Cells:          5094
Generated Cells:            5094
Header Errors:              0
Seqence Mismatches :        0
Lost Cells:                 0
Channel Uptime (secs.)      1105
Signalling Status           Offhook

wss-mgxb.1.10.CESM.a > dspsarcnt 159
                          SarShelfNum:          1
                          SarSlotNum:           10
                          SarChanNum:           159
                                Tx                      Rx
                          ---------------       ---------------
  Total Cells:            8524                  8524
  Total CellsCLP:         0                     0
  Total CellsAIS:         0                     0
  Total CellsFERF:        0                     0
  Total CellsEnd2EndLpBk: 0                     0
  Total CellsSegmentLpBk: 0                     0
  RcvCellsDiscOAM:                        0
```

7. BPX Monitor Connection Statistics

Again, the key point to notice in this output is that the cells port-to-net and net-to-port are the same. If you ever see the "Avg CPS" above the PCR for the connection, the "%util" above 100, or any of the "dscd" counters clocking up, you probably have not calculated the correct PCR for the data stream.

BPX-B

```
wss-bpxb       TN    StrataCom       BPX 8600  9.1.04    Nov. 18 1998 14:00 EST

Channel Statistics for 3.1.10.159  Cleared: Nov. 18 1998 14:00  (/)  Snapshot
PCR: 512/512 cps        Collection Time: 0 day(s) 00:00:15      Corrupted: NO
   Traffic      Cells      CLP      Avg CPS   %util  Chan Stat Addr: 31132770
From Port   :      7818          0      511      99
To Network  :      7818          ---    511      99
From Network:      7818          0      511      99
To Port     :      7818          0      511      99
```

```
Rx Frames Rcv :         0  NonCmplnt Dscd:         0  Rx Q Depth    :         0
Tx Q Depth    :         0  Rx CLP0       :      7818  Rx Nw CLP0    :      7818
Igr VSVD ACR  :         0  Egr VSVD ACR  :         0  Tx Clp 0      :      7818
Rx Clp 0+1    :      7818  Tx Clp 0 Dscd :         0  Tx Clp 1 Dscd :         0
Tx Clp0+1 Dscd:         0

Last Command: dspchstats 3.1.10.159 1
```

8. Verify Clocking

With a structured service, synchronous clocking is our only clocking scheme option. Because we possibly have individual DS0s from various sources muxed into one T1/E1 stream at the far end, it is not possible rely on the assumptions driving the implementations of the SRTS or Adaptive clocking found with the unstructured service.

The clocking arrangement in this example has all units tracing their clock to the internal oscillator on WSS−BPXC. Since BPXC is the highest routing node in the network, BPXB automatically takes its clock from BPXC. Each MGX shelf is configured to time its bus inband from its BNM card. The lines on both CESM cards are locally timed. And, each MC3810 is configured to time its internal bus using the recovered clock from the MFT. The MFTs do not show any clock slips, so timing looks good end−to−end (and we chose the MC3810s because they are very particular about timing).

BPX−B

```
wss−bpxb       TN    StrataCom       BPX 8600  9.1.04    Nov. 18 1998 14:03 EST

                              Current Clock Source

Source Node:     wss−bpxc
Source Line:     Internal (CC)

Clock Type:
Clock Frequency: 1544000

Path to Source:
   wss−bpxb 1.1−−wss−bpxc

Last Command: dspcurclk
```

MGX−B

```
wss−mgxb.1.3.ASC.a > dspclksrc

PrimaryClockSource:   Inband from BNM
  SecondaryClockSource: Internal Oscillator
  CurrentClockSource:   Primary
  ClockSwitchState:     NoChange
  ExtClkPresent:        Not Present
  ExtClkSrcImpedance:   100 ohms
  ExtClkConnectorType:  DB−15
```

MGX−C

```
wss−mgxc.1.4.ASC.a > dspclksrc

  PrimaryClockSource:   Inband from BNM
  SecondaryClockSource: Internal Oscillator
  CurrentClockSource:   Primary
  ClockSwitchState:     NoChange
  ExtClkPresent:        Not Present
  ExtClkSrcImpedance:   75 ohms
  ExtClkConnectorType:  BNC
```

3810−E

```
wss−3810e# show network−clocks
Priority 1 clock source(desired config): T1 0
Priority 1 clock source(run−time config): T1 0
Clock switch delay: 10
Clock restore delay: 10
T1 0 is clocking system bus for 117 seconds.
Run Priority Queue:  controller0,

wss−3810e# show cont t1 0
T1 0 is up.
  No alarms detected.
  Slot 3 CSU Serial #08464190 Model TEB HWVersion 4.70
  Framing is ESF, Line Code is B8ZS, Clock Source is Line.
  Data in current interval (8 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
```

3810−F

```
wss−3810f# show network−clocks
Priority 1 clock source(desired config): T1 0
Priority 1 clock source(run−time config): T1 0
Clock switch delay: 10
Clock restore delay: 10
T1 0 is clocking system bus for 132 seconds.
Run Priority Queue:  controller0,

wss−3810f# show cont t1 0
T1 0 is up.
  No alarms detected.
  Slot 3 CSU Serial #08104198 Model TEB HWVersion 4.70
  Framing is ESF, Line Code is B8ZS, Clock Source is Line.
  Data in current interval (19 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
```

9. MC3810 Configurations

The relevant portions of the 3810 configuration files are included below.

Note: Controller T1 0 is the MFT and ties into the Serial 2 interface on the box. The default clock source on the T1 controllers is Line.

3810−E

```
wss−3810e# wr t
Building configuration...

Current configuration:
!
version 11.3
!
hostname wss−3810e
!
network−clock−select 1 T1 0
!
controller T1 0
 framing esf
 linecode b8zs
 channel−group 1 timeslots 9−11 speed 64
!
```

```
interface Serial2
 ip address 10.98.199.1 255.255.255.0
!
voice-port 1/1
voice-port 1/2
voice-port 1/6
!
end
```

3810-F

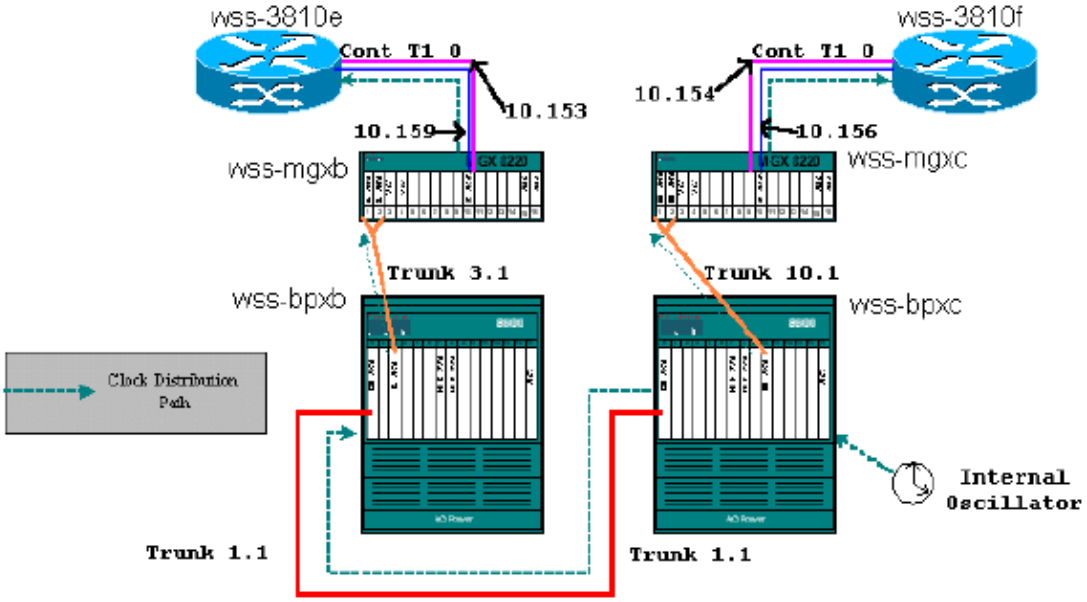
```
wss-3810f# wr t
Building configuration...

Current configuration:
!
version 11.3
!
hostname wss-3810f
!
network-clock-select 1 T1 0
!
controller T1 0
 framing esf
 linecode b8zs
 channel-group 1 timeslots 6-8 speed 64
!
interface Serial2
 ip address 10.98.199.2 255.255.255.0
!
voice-port 1/1
voice-port 1/2
voice-port 1/6
!
end
```

Channel Associated Signaling

One of the major concerns with an N*DS0 service is how to pass channel associated signaling (CAS). With the unstructured service, this concern is not an issue. The CAS bits are identified based on offsets from the line framing structure. Because all the bits are sent in order with the unstructured service, the relative positions do not change. With a structured service, we have the possibility of muxing DS0s from various inputs to one output. The framing on a T1/E1 stream is controlled by the transmitter. Thus, without special provisions, it is very unlikely that the CAS bits on ingress to the CES interface would make it out in the correct position on egress.

Structured circuit emulation gets around this problem by removing the CAS bits on ingress, passing them along with the data bits in the AAL1 frame and re-inserting them into the T1/E1 stream on egress. Obviously, this requires slightly more overhead than does the Structured-Basic service.

In this example, we configure the 3810 MFTs to put one timeslot in a voice group. We will carry this DSO and its associated signaling across the network. With some other slight modifications to the configuration used in the Structured-Basic example, analog phones connected to the FXS ports on the 3810 units will be able to place calls across this CES connection.



1. CESM Add CAS Port/Channel for Voice DS0

MGX−B
wss−mgxb.1.10.CESM.a > **addport 53 5 3 1 1** wss−mgxb.1.10.CESM.a > **addchan 153 53 4 0 0 0** **ERR: cesCAS mismatch with existing channel on the line.** **requested channel = 153** **existing channel = 159 Set failed due to illegal parameter(s)**

The **addchan** *chan_num port_num cesCAS_sig_type partial_fill cond_data cond_signaling* command syntax is:

- *chan_num* In the range from 32 to 279.
- *port_num* In the range from 1 to 192 for T1, and from 1 to 248 for E1.
- *cesCAS_sig_type*¡ = basic, 2 = e1Cas, 3 = ds1SfCas, or 4 = ds1EsfCas.
- *partial_fill* = fully filled, in the range from 20 to 47 = E1 structured, in the range from 25 to 47 = T1 structured, or in the range from 33 to 47 = T1/E1 unstructured.
- *cond_data*¢55 = UDT, in the range from 0 to 255 = SDT.
- *cond_signaling* In the range from 0 to 15.

The error can be caused by:

- Illegal/invalid parameters
- Channel already exists
- Port may not be up

What happened? It appears that we can not define one channel on a line for Structured−CAS and one channel for Structured−Basic. This means that we have to delete the existing Basic channels and re−add them in as CAS along with the new CAS channels:

```
wss−mgxb.1.10.CESM.a > delchan 159
wss−mgxb.1.10.CESM.a > addchan 153 53 4 4 0 0 0
wss−mgxb.1.10.CESM.a > addchan 159 59 4 0 255 15
wss−mgxb.1.10.CESM.a > cnfchan 153 1000 1000 0 1 1 1
wss−mgxb.1.10.CESM.a > cnfchan 159 1000 1000 0 1 1 1
```

MGX−C

```
wss-mgxc.1.10.CESM.a > addport 54 5 4 1 1
wss-mgxc.1.10.CESM.a > delchan 156
wss-mgxc.1.10.CESM.a > addchan 154 54 4 0 0 0
wss-mgxc.1.10.CESM.a > addchan 156 56 4 0 255 15
wss-mgxc.1.10.CESM.a > cnfchan 154 1000 1000 0 1 1 1
wss-mgxc.1.10.CESM.a > cnfchan 156 1000 1000 0 1 1 1
```

2. CESM Verify Current Configuration

MGX-B

```
wss-mgxb.1.10.CESM.a > dspports
  Port     Ena/Speed   Type
  ------   --- ------ --------
  10.5.53  Add/  64k structur
  10.5.59  Add/ 192k structur

  Number of ports:    2

  PortDs0UsedLine1:             0x00000000
  PortDs0UsedLine2:             0x00000000
  PortDs0UsedLine3:             0x00000000
  PortDs0UsedLine4:             0x00000000
  PortDs0UsedLine5:             0x00000704
  PortDs0UsedLine6:             0x00000000
  PortDs0UsedLine7:             0x00000000
  PortDs0UsedLine8:             0x00000000
  PortNumNextAvailable:         19

wss-mgxb.1.10.CESM.a > dspchans
    Channel     ChanNum   Status   CDV   MaxBufSize   CLIP    CBRservice
  ------------  -------   ------  -----  ----------  ------  ------------
  10.5.53.153     153      Mod     1000      384      1000   structured
  10.5.59.159     159      Mod     1000      384      1000   structured

wss-mgxb.1.10.CESM.a > dspchan 153
  ChanNum:                  153
  ChanRowStatus:            Mod
  ChanLineNum:              5
  ChanMapVpi:               10
  ChanMapVci:               153
  ChanCBRService:           structured
  ChanClockMode:            Synchronous
  ChanCAS:                  DS1 ESF
  ChanPartialFill:          47
  ChanMaxBufSize:           384 bytes
  ChanCDV:                  1000 micro seconds
  C L I P:                  1000 milliseconds
  ChanLocalRemoteLpbkState: Disabled
  ChanTestType:             TestOff
  ChanTestState:            NotInProgress
  ChanRTDresult:            65535 ms
  ChanPortNum                  53
  ChanConnType              PVC
  ISDetType                 DetectionDisabled
  CondData                  0
  CondSignalling            0
  ExtISTrig                 DisableIdleSupression
  ISIntgnPeriod             4095 seconds
  ISSignallingCode          0
  OnHookCode                1

  ChanNumNextAvailable:  33
```

3. MC3810 Add Voice Group and Dial Peers

We had to make some adjustments to the channel−group on the 3810 units in order to support the new CAS arrangement. Because the CESM card will not allow us to mix Structured−Basic and Structured−CAS channels on the same line, we had to make all channels Structured−CAS. This change implies that we can no longer reliably pass 64kbps through each DS0, so must reconfigure the channel−group lines to 56kbps.

The default speed for a channel−group DS0 is 56kbps. Thus, that parameter will not display in the configuration. We include it here for display purposes.

Changes are in **boldface** type in the output:

3810−E

```
wss−3810e# wr t
Building configuration...

Current configuration:
!
version 11.3
!
hostname wss−3810e
!
network−clock−select 1 T1 0
!
controller T1 0
 framing esf
 linecode b8zs
 channel−group 1 timeslots 9−11 speed 56
 mode cas
 voice−group 0 timeslots 3 type e&m−immediate
!
interface Serial2
 ip address 10.98.199.1 255.255.255.0
!
voice−port 0/3
!
voice−port 1/1
voice−port 1/2
voice−port 1/6
!
dial−peer voice 60000 pots
 destination−pattern 6
 port 0/3
!
dial−peer voice 7777 pots
 destination−pattern 7777
 port 1/2
!
end
```

3810−F

```
wss−3810f# wr t
Building configuration...

Current configuration:
!
version 11.3
!
hostname wss−3810f
!
network−clock−select 1 T1 0
!
```

```
controller T1 0
 framing esf
 linecode b8zs
 channel-group 1 timeslots 6-8 speed 56
 mode cas
 voice-group 0 timeslots 4 type e&m-immediate
!
interface Serial2
 ip address 10.98.199.2 255.255.255.0
!
voice-port 0/4
!
voice-port 1/1
voice-port 1/2
voice-port 1/6
!
dial-peer voice 50000 pots
 destination-pattern 5
 port 0/4
!
dial-peer voice 8888 pots
 destination-pattern 8888
 port 1/2
!
end
```

4. BPX Verify HDLC PVC Operation

Before adding the CBR PVC for the voice channel, let's look at the existing PVC for the HDLC connection. Notice the impact caused by changing the channel type from Structured−Basic to Structured−CAS.

BPX−B

```
wss-bpxb       TN    StrataCom       BPX 8600  9.1.04    Nov. 18 1998 14:21 EST

Channel Statistics for 3.1.10.159  Cleared: Nov. 18 1998 14:21  (|)  Snapshot
PCR: 512/512 cps        Collection Time: 0 day(s) 00:00:10      Corrupted: NO
   Traffic      Cells      CLP      Avg CPS   %util  Chan Stat Addr: 31132770
From Port   :      5370          0      526     102
To Network  :      5369     ---          526     102
From Network:      5370          0      526     102
To Port     :      5370          0      526     102

Rx Frames Rcv :         0  NonCmplnt Dscd:         0  Rx Q Depth    :         0
Tx Q Depth    :         0  Rx CLP0       :      5370  Rx Nw CLP0    :      5370
Igr VSVD ACR  :         0  Egr VSVD ACR  :         0  Tx Clp 0      :      5370
Rx Clp 0+1    :      5370  Tx Clp 0 Dscd :         0  Tx Clp 1 Dscd :         0
Tx Clp0+1 Dscd:         0

Last Command: dspchstats 3.1.10.159 1
```

Use this formula to compute the PCR for a Strcutured−CAS connection with no partial fill where the number of DS0s (N) is odd:

```
{8000 * [ (1 + N * 49) / 48 ] / 46.875} = {526.22} = 527
```

So, now we need to change the parameters for this connection.

BPX−B

```
Last Command: cnfcon 3.1.10.159 527 * * * Y
------------------------------------------------------------------------
```

```
wss-bpxb       TN    StrataCom       BPX 8600  9.1.04    Nov. 18 1998 14:26 EST

Conn:  3.1.10.159       wss-bpxc    10.1.10.156       cbr       Status:OK
   PCR(0+1)     % Util    CDVT(0+1)     Policing
    527/527    100/100    1000/1000      4/4
Owner: LOCAL  Restriction: NONE  COS: 0
Trunk Cell Routing Restrict: Y
Path:   wss-bpxb 1.1-- 1.1wss-bpxc
Pref:   Not Configured

wss-bpxb      BXM       : OK           wss-bpxc  BXM       : OK
              Line  3.1 : OK                     Line 10.1 : OK
              OAM Cell RX: Clear                 NNI       : OK
              NNI       : OK
Conn:  3.1.10.159       wss-bpxc    10.1.10.156       cbr       Status:OK
   PCR(0+1)     % Util       CDVT(0+1)         Policing
  527/527      100/100       1000/1000            4

Trunk Cell Routing Restrict: Y

Last Command: dspcon 3.1.10.159
------------------------------------------------------------------------
wss-bpxb       TN    StrataCom       BPX 8600  9.1.04    Nov. 18 1998 14:30 EST

Channel Statistics for 3.1.10.159  Cleared: Nov. 18 1998 14:29  (|)  Snapshot
PCR: 527/527 cps        Collection Time: 0 day(s) 00:00:20      Corrupted: NO
   Traffic      Cells      CLP      Avg CPS   %util  Chan Stat Addr: 31132770
From Port   :     10729          0      526      99
To Network  :     10728    ---           526      99
From Network:     10729          0      526      99
To Port     :     10729          0      526      99

Rx Frames Rcv :         0  NonCmplnt Dscd:         0  Rx Q Depth    :         0
Tx Q Depth    :         0  Rx CLP0       :     10729  Rx Nw CLP0    :     10729
Igr VSVD ACR  :         0  Egr VSVD ACR  :         0  Tx Clp 0      :     10729
Rx Clp 0+1    :     10729  Tx Clp 0 Dscd :         0  Tx Clp 1 Dscd :         0
Tx Clp0+1 Dscd:         0

Last Command: dspchstats 3.1.10.159 1
```

5. BPX Add and Verify Voice PVC

For the single DS0 voice connection, use the same PCR calculation formula used previously. In this case, N = 1:

{8000 * [(1+ N*49)/48]/46.875} = {177.78} = 178

BPX−B

```
Last Command: addcon 3.1.10.153 wss-bpxc 10.1.10.154 cbr 178 * 100 * Y
------------------------------------------------------------------------
wss-bpxb       TN    StrataCom       BPX 8600  9.1.04    Nov. 18 1998 16:06 EST

Conn:  3.1.10.153       wss-bpxc    10.1.10.154       cbr       Status:OK
   PCR(0+1)     % Util    CDVT(0+1)     Policing
    178/178    100/100     100/100       4/4
Owner: LOCAL  Restriction: NONE  COS: 0
Trunk Cell Routing Restrict: Y
Path:   wss-bpxb 1.1-- 1.1wss-bpxc
Pref:   Not Configured

wss-bpxb      BXM       : OK           wss-bpxc  BXM       : OK
              Line  3.1 : OK                     Line 10.1 : OK
              OAM Cell RX: Clear                 NNI       : OK
              NNI       : OK
```

```
Conn:  3.1.10.153       wss-bpxc    10.1.10.154       cbr       Status:OK
   PCR(0+1)     % Util       CDVT(0+1)         Policing
  178/178      100/100        100/100             4

Trunk Cell Routing Restrict: Y

Last Command: dspcon 3.1.10.153
------------------------------------------------------------------------
wss-bpxb       TN    StrataCom       BPX 8600  9.1.04    Nov. 18 1998 16:08 EST

Channel Statistics for 3.1.10.153  Cleared: Nov. 18 1998 16:07  (|)  Snapshot
PCR: 178/178 cps        Collection Time: 0 day(s) 00:01:36      Corrupted: NO
   Traffic      Cells      CLP      Avg CPS   %util  Chan Stat Addr: 31132878
From Port   :     17080          0      177      99
To Network  :     17080    ---           177      99
From Network:     17080          0      177      99
To Port     :     17080          0      177      99

Rx Frames Rcv :         0  NonCmplnt Dscd:         0  Rx Q Depth    :         0
Tx Q Depth    :         0  Rx CLP0       :     17080  Rx Nw CLP0    :     17080
Igr VSVD ACR  :         0  Egr VSVD ACR  :         0  Tx Clp 0      :     17080
Rx Clp 0+1    :     17080  Tx Clp 0 Dscd :         0  Tx Clp 1 Dscd :         0
Tx Clp0+1 Dscd:         0

Last Command: dspchstats 3.1.10.153 1
```

6. 3810 Verify Voice Call Operation

For the sake of completeness, the output below shows the operation of the voice call on the 3810 when someone picks up the phone connected to port 1/2 on 3810E and dials 6−8888. With this configuration, there is not much to see on the CESM cards or on the BPX. We do not have any visibility into the signaling bits on the CESM card or the data contained therein. So, all we see is a constant 177 cells/sec running through the PVC.

3810−E

```
wss-3810e# deb voice cp

Voice Call Processing State Machine debugging is on
wss-3810e# deb voice ee

Voice end-to-end call manager debugging is on
wss-3810e#
```

Access the phone on port 1/2 and dial 68888:

```
Nov 20 13:03:22:  1/2: CPD( ), idle gets event seize_ind
Nov 20 13:03:22:  1/2: EECM(out), ST_NULL            EV_ALLOC_DSP
Nov 20 13:03:22:  1/2: CPD( ), idle gets event dsp_ready
Nov 20 13:03:22:  1/2: CPD( ), idle ==> collect
Nov 20 13:03:27:  1/2: CPD(in), collect gets event digit
Nov 20 13:03:27:  1/2: EECM(in), ST_DIGIT_COLLECT   EV_PARSE_DIGIT    6
Nov 20 13:03:27:  1/2: CPD(in), collect gets event addr_done
Nov 20 13:03:27:  1/2: CPD(in), collect ==> request
Nov 20 13:03:27:  1/2: EECM(in), ST_ADDRESS_DONE    EV_OUT_SETUP
Nov 20 13:03:27:  1/2: CPD(in), request gets event call_proceeding
Nov 20 13:03:27:  1/2: CPD(in), request ==> in_wait_answer
Nov 20 13:03:27:  -1/-1: EECM(out), ST_NULL            EV_IN_SETUP
Nov 20 13:03:27:  1/2: EECM(in), ST_OUT_REQUEST     EV_IN_PROCEED
Nov 20 13:03:27:  0/3: CPD( ), idle gets event call_ind
Nov 20 13:03:27:  0/3: EECM(out), ST_SEIZE           EV_ALLOC_DSP
Nov 20 13:03:27:  0/3: CPD( ), idle gets event dsp_ready
Nov 20 13:03:27:  0/3: CPD( ), idle ==> out_wait_ready
```

```
Nov 20 13:03:27:  0/3: CPD(out), out_wait_ready gets event ready_ind
Nov 20 13:03:27:  0/3: EECM(out), ST_SEIZE           EV_OUT_ALERT
Nov 20 13:03:27:  1/2: EECM(in), ST_OUT_REQUEST     EV_IN_ALERT
Nov 20 13:03:27:  1/2: CPD(in), in_wait_answer gets event call_accept
Nov 20 13:03:27:  1/2: EECM(in), ST_OUT_REQUEST     EV_OUT_ALERT_ACK
Nov 20 13:03:27:  0/3: CPD(out), out_wait_ready ==> addr_forward
Nov 20 13:03:27:  0/3: CPD(out), addr_forward gets event dial_done
Nov 20 13:03:27:  0/3: CPD(out), addr_forward ==> out_wait_answer
```

Other side picks up the phone:

```
Nov 20 13:03:34:  0/3: CPD(out), out_wait_answer gets event answer_ind
Nov 20 13:03:34:  0/3: CPD(out), out_wait_answer ==> connected
Nov 20 13:03:34:  0/3: EECM(out), ST_IN_PENDING      EV_OUT_CONNECT
Nov 20 13:03:34:  1/2: EECM(in), ST_WAIT_FOR_ANSWER EV_IN_CONNECT
Nov 20 13:03:34:  1/2: CPD(in), in_wait_answer gets event call_answered
Nov 20 13:03:34:  1/2: CPD(in), in_wait_answer ==> connected
```

Other side hangs up the phone:

```
Nov 20 13:03:42:  0/3: CPD(out), connected gets event disconnect_ind
Nov 20 13:03:42:  0/3: CPD(out), connected ==> tx_disc
Nov 20 13:03:42:  0/3: EECM(out), ST_ACTIVE          EV_OUT_REL
Nov 20 13:03:42:  0/3: CPD(out), tx_disc gets event peer_msg_done
Nov 20 13:03:42:  0/3: CPD(out), tx_disc ==> disconnect_wait
Nov 20 13:03:42:  1/2: EECM(in), ST_ACTIVE          EV_IN_REL
Nov 20 13:03:42:  1/2: CPD(in), connected gets event peer_onhook
Nov 20 13:03:42:  1/2: CPD(in), connected ==> disconnect_wait
Nov 20 13:03:42:  1/2: EECM(in), ST_DISCONN_PENDING EV_OUT_REL_ACK
Nov 20 13:03:43:  0/3: CPD(out), disconnect_wait gets event idle_ind
Nov 20 13:03:43:  0/3: CPD(out), disconnect_wait ==> idle_pending
Nov 20 13:03:43:  0/3: EECM(out), ST_DISCONN_PENDING EV_FREE_DSP
Nov 20 13:03:43:  0/3: CPD( ), idle_pending gets event timer1
Nov 20 13:03:43:  0/3: CPD( ), idle_pending ==> idle
Nov 20 13:03:43:  1/2: CPD(in), disconnect_wait gets event idle_ind
Nov 20 13:03:43:  1/2: CPD(in), disconnect_wait ==> idle_pending
Nov 20 13:03:43:  1/2: EECM(in), ST_DISCONN_PENDING EV_FREE_DSP
Nov 20 13:03:43:  1/2: CPD( ), idle_pending gets event timer1
Nov 20 13:03:43:  1/2: CPD( ), idle_pending ==> idle
wss-3810e#
```

On−Hook Detection

The CESM card has the capability to apply some intelligence to the signaling bits captured on Structured−CAS connections. We can configure the channel to recognize when the call is off−hook and on−hook. If the signaling bits indicate an on−hook condition, the channel generates minimal information to the other side of the connection. When the signaling bits change to an off−hook condition, traffic flows at the 177 cells−per−second rate.

Note: You can use the on−hook detection feature on 1x64 kbps connections only. Thus, if you want to configure this feature for each DS0 on a T1, you need to configure 24 logical ports and assign one channel for each port.

1. 3810 Identify On−hook/Off−hook Signaling Code

   The first step in implementing on−hook detection is to understand the ABCD bits states for the signaling plan employed by the end equipment. In this example, we have configured the Cisco MC3810 digital voice port for E&M Immediate signaling. We will use the debug facilities on the 3810 to read the signaling states in an on−hook and off−hook condition. The debug serial interface command enables the robbed bit signaling display in the show controller t1 output, as shown below:

MGX−B

```
wss−3810f# deb serial int
Serial network interface debugging is on
wss−3810f# clear count
Clear "show interface" counters on all interfaces [confirm]y
%CLEAR−5−COUNTERS: Clear counter on all interfaces by console
```

3810−F On−Hook

```
wss−3810f# show cont t1 0
T1 0 is up.
  No alarms detected.
  Framing is ESF, Line Code is B8ZS, Clock Source is Line.
  Data in current interval (11 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
 Robbed bit signals state:
        timeslots       rxA rxB rxC rxD         txA txB txC txD

        4               0   0   0   0           0   0   0   0
```

3810−F Off−Hook

```
wss−3810f# show cont t1 0
T1 0 is up.
  No alarms detected.
  Slot 3 CSU Serial #08104198 Model TEB HWVersion 4.70
  Framing is ESF, Line Code is B8ZS, Clock Source is Line.
  Data in current interval (74 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
 Robbed bit signals state:
        timeslots       rxA rxB rxC rxD         txA txB txC txD

        4               1   1   1   1           1   1   1   1
```

2. CESM Modify Channels for On−Hook Detection

To do this, we need to make only two modifications to the previous Structured−CAS configuration:

- ♦ Enable Signaling Detection for the channel.
- ♦ Modify the on−hook code to reflect the ABCD bit pattern for an on−hook condition.

Here, the on−hook ABCD bit pattern is 0000 in binary notation, which is 0 in decimal notation.

MGX−B

```
wss−mgxb.1.10.CESM.a > cnfchan 153 1000 1000 0 1 2 1
wss−mgxb.1.10.CESM.a > xcnfchan −chn 153 −onhkcd 0
```

MGX−C

```
wss−mgxc.1.10.CESM.a > cnfchan 154 1000 1000 0 1 2 1
wss−mgxc.1.10.CESM.a > xcnfchan −chn 154 −onhkcd 0
```

3. CESM Monitor Channel Statistics

Once again we review the channel and SAR counts for the CESM card. When the channel is in an on−hook condition, no cell traffic is recorded on the CESM counters (though we will see the keepalive cells on the BPX channel statistics). When either channel detects that the signaling bits have changed to an off−hook condition, that channel will begin to send cells toward the network.

Note: Just because one side goes off−hook and starts sending cells does not imply that the other side will start sending cells back. If the other side is configured for on−hook detection, it follows the same rules and will not send cells until it detects signaling bits indicating an off−hook condition.

MGX−B On−Hook

```
wss−mgxb.1.10.CESM a > clrchancnts
wss−mgxb.1.10.CESM.a > clrsarcnts

wss−mgxb.1.10.CESM.a > dspchancnt 153
ChanNum:                    153
Chan State:                 okay
Chan RCV ATM State:         Normal
Chan XMT ATM State:         Normal
Cell Loss Status:           No Cell Loss
Reassembled Cells:          0
Generated Cells:            0
Header Errors:              0
Seqence Mismatches :        0
Lost Cells:                 0
Channel Uptime (secs.)      7051
Signalling Status           Onhook

wss−mgxb.1.10.CESM.a > dspsarcnt 153
                          SarShelfNum:          1
                          SarSlotNum:           10
                          SarChanNum:           153
                                Tx                        Rx
                          ----------------      ----------------
  Total Cells:            0                       0
  Total CellsCLP:         0                       0
  Total CellsAIS:         0                       0
  Total CellsFERF:        0                       0
  Total CellsEnd2EndLpBk: 0                       0
  Total CellsSegmentLpBk: 0                       0
  RcvCellsDiscOAM:                                0
```

MGX−B Off−Hook

```
wss−mgxb.1.10.CESM.a > dspchancnt 153
ChanNum:                    153
Chan State:                 okay
Chan RCV ATM State:         Normal
Chan XMT ATM State:         Normal
Cell Loss Status:           No Cell Loss
Reassembled Cells:          3151
Generated Cells:            5199
Header Errors:              0
Seqence Mismatches :        0
Lost Cells:                 3
Channel Uptime (secs.)      7122
Signalling Status           Offhook

wss−mgxb.1.10.CESM.a > dspsarcnt 153
                          SarShelfNum:          1
                          SarSlotNum:           10
                          SarChanNum:           153
                                Tx                        Rx
                          ----------------      ----------------
  Total Cells:            8150                    6102
  Total CellsCLP:         0                       0
  Total CellsAIS:         0                       0
  Total CellsFERF:        0                       0
  Total CellsEnd2EndLpBk: 0                       0
  Total CellsSegmentLpBk: 0                       0
```

```
  RcvCellsDiscOAM:                                0
```

4. BPX Monitor Channel Statistics

Looking at the channel statistics on the BPX nodes, we can see the three phases of call. Because the counters are averages, it would take a longer time than we allowed here for the "Avg. CPS" to reach the PCR.

BPX−B On−Hook

```
wss−bpxb       TN    StrataCom       BPX 8600  9.1.04    Nov. 18 1998 16:13 EST

Channel Statistics for 3.1.10.153  Cleared: Nov. 18 1998 16:12  (\)  Snapshot
PCR: 178/178 cps        Collection Time: 0 day(s) 00:00:14      Corrupted: NO
   Traffic      Cells      CLP      Avg CPS   %util  Chan Stat Addr: 31132878
From Port   :        16          0        1       0
To Network  :        16    ---            1       0
From Network:        16          0        1       0
To Port     :        16          0        1       0

Rx Frames Rcv :        16  NonCmplnt Dscd:         0  Rx Q Depth    :         0
Tx Q Depth    :         0  Rx CLP0       :        16  Rx Nw CLP0    :        16
Igr VSVD ACR  :         0  Egr VSVD ACR  :         0  Tx Clp 0      :        16
Rx Clp 0+1    :        16  Tx Clp 0 Dscd :         0  Tx Clp 1 Dscd :         0
Tx Clp0+1 Dscd:         0

Last Command: dspchstats 3.1.10.153 1
```

BPX−B£810E Off−Hook

```
wss−bpxb       TN    StrataCom       BPX 8600  9.1.04    Nov. 18 1998 16:15 EST

Channel Statistics for 3.1.10.153  Cleared: Nov. 18 1998 16:14  (−)
PCR: 178/178 cps        Collection Time: 0 day(s) 00:00:18      Corrupted: NO
   Traffic      Cells      CLP      Avg CPS   %util  Chan Stat Addr: 31132878
From Port   :      1232          0       68      38
To Network  :      1232    ---           68      38
From Network:        20          0        1       0
To Port     :        20          0        1       0

Rx Frames Rcv :        12  NonCmplnt Dscd:         0  Rx Q Depth    :         0
Tx Q Depth    :         0  Rx CLP0       :      1232  Rx Nw CLP0    :        20
Igr VSVD ACR  :         0  Egr VSVD ACR  :         0  Tx Clp 0      :        20
Rx Clp 0+1    :      1232  Tx Clp 0 Dscd :         0  Tx Clp 1 Dscd :         0
Tx Clp0+1 Dscd:         0

This Command: dspchstats 3.1.10.153 1
```

BPX−B£810F Off−Hook

```
wss−bpxb       TN    StrataCom       BPX 8600  9.1.04    Nov. 18 1998 16:15 EST

Channel Statistics for 3.1.10.153  Cleared: Nov. 18 1998 16:14  (|)
PCR: 178/178 cps        Collection Time: 0 day(s) 00:00:30      Corrupted: NO
   Traffic      Cells      CLP      Avg CPS   %util  Chan Stat Addr: 31132878
From Port   :      3499          0      113      63
To Network  :      3498    ---           113      63
From Network:      1479          0       48      26
To Port     :      1479          0       48      26

Rx Frames Rcv :        12  NonCmplnt Dscd:         0  Rx Q Depth    :         0
Tx Q Depth    :         0  Rx CLP0       :      3499  Rx Nw CLP0    :      1479
Igr VSVD ACR  :         0  Egr VSVD ACR  :         0  Tx Clp 0      :      1479
Rx Clp 0+1    :      3499  Tx Clp 0 Dscd :         0  Tx Clp 1 Dscd :         0
```

```
Tx Clp0+1 Dscd:         0

This Command: dspchstats 3.1.10.153 1
```

5. 3810 Verify Voice Call Operation

   Voice call operation, from the 3810 perspective, remains unchanged. If you want to monitor the ABCD bit changes on the 3810 in real time, use the following debug command:

3810−E

```
wss−3810e# deb dsx1 sig
DSX1 Signaling debugging is on
wss−3810e#
Slot 3 Receive DS0 (3) ABCD= 0 Timestamp 7908877

Slot 3 Receive DS0 (3) ABCD= 0 Timestamp 7909979
```

Command List

This section lists the commands, with available options and values, used in the configuration examples throughout this document.

addln *line_num* , where&

- *line_num* can be a numeric value in the range from 1 to 8.

cnfln *line_num line_code line_len clk_src [E1−signaling]* , where&

- *line_num* can be a numeric value in the range from 1 to 8.
- *line_code* can be:
  - ♦ 2 = B8ZS (T1)
  - ♦ 3 = HDB3 (E1)
  - ♦ 4 = AMI (T1/E1)
- *line_len* (line length) can be:
  - ♦ 8 = E1 with SMB line module
  - ♦ 9 = E1 with RJ48 line module
  - ♦ 10 = T1 0 to 131 ft. line build out
  - ♦ 11 = T1 131 to 262 ft.
  - ♦ 12 = T1 262 to 393 ft.
  - ♦ 13 = T1 393 to 524 ft.
  - ♦ 14 = T1 524 to 655 ft.
  - ♦ 15 = T1 > 655 ft.
- *clk_src* (clock source) can be:
  - ♦ 1 = Loop clock; the transmit clock on the interface is locked to the receive clock from the attached device.
  - ♦ 2 = Local clock; the CESM−8 cards uses clock derived from the backplane to drive the transmit clock.
- *E1−signaling* can be:
  - ♦ CAS = Channel Associated Signaling; signaling information is contained in timeslot 16 and framing is carried in timeslot 0.
  - ♦ CAS_CRC = CAS with Cyclic Redundancy Check (CRC).

- CCS = Common Channel Signaling; signaling information is not tied to a particular timeslot. Framing is still carried in the first timeslot.
- CCS_CRC = CCS with CRC.
- CLEAR = No attempt is made to identify framing or signaling on the incoming stream. The entire data stream is considered data.

addport *port_num line_num begin_slot num_slot port_type* , where&

- *port_num* can be:
  - In the range from 1 to 192 = T1 (8 lines * 24 DS0s/line)
  - In the range from 1 to 248 = E1 (8 lines * 31 DS0s / line)
- *line_num* can be a numeric value from 1 to 8.
- *begin_slot* is the beginning time slot in line to start port.
- *num_slot* is the number of DS0 time slots assigned to the port.
- *port_type* can be:
  - 1 = Structured

    For T1, you can set up a structured port type for bandwidths ranging from 1 to 24 DS0s.

    For E1, structured ports can not include the framing timeslot (CCS or CAS), or the signaling timeslot (CAS).
  - 2 = Unstructured

    For T1, unstructured port = 24 DS0s.

    For E1, unstructured port can only be configured when the E1 signaling is set to CLEAR.
  - 3 = framingOnVcDisconnect

    This port type is basically the same as unstructured with one key difference when an unstructured port experiences a network−side cell loss, the CESM−8 transmits conditional data down the line. With framingOnVcDisconnect, cell loss from the network side results in the CESM−8 looping back data received from the CPE back out the port so that the CPE does not lose framing. The same restrictions on DS0 count and signaling type present for an unstructured port apply here.

addchan *chan_num port_num CesCas partial_fill cond_data cond_signaling* , where&

- *chan_num* can be a numeric value in the range from 32 to 279.
- *port_num* can be:
  - In the range from 1 to 192 = T1 (8 lines * 24 DS0s/line)
  - In the range from 1 to 248 = E1 (8 lines * 31 DS0s / line)
- *CesCas* can be:
  - 1 = Basic; the CES interworking function does not recognize signaling information for special transport across the network.
  - 2 = e1Cas; recover E1 Channel Associated Signaling (CAS) for transport.
  - 3 = ds1SfCas; recover T1 Channel Associated Signaling from a SuperFrame structure (ABAB).
  - 4 = ds1EsfCas; recover T1 CAS from an Extended SuperFrame structure (ABCD).
- *partial_fill* can be 0 (zero) or a value from 20 to 47:

- ♦ 0 or 47 = Fully filled
- ♦ In the range from 20 to 47 = E1 structured
- ♦ In the range from 25 to 47 = T1 structured
- ♦ In the range from 33 to 47 = T1/E1 unstructured

- *cond_data* can be:
  - ♦ 0 to 255 = Structured Data Transport (SDT)
  - ♦ 255 = Unstructured Data Transport (UDT)
- *cond_signaling* is a decimal representation of the the 4−bit ABCD bit pattern, in the range from 0 to 15, where&
  - ♦ 0 = 0000 binary
  - ♦ 1 decimal = 0001 binary
  - ♦ 8 decimal = 1000 binary
  - ♦ 15 decimal = 1111 binary

cnfchan *chan_num CDV CLIP bufsize clockmode IdleDetEnable ExtIStrig* , where&

- *chan_num* can be a numeric value in the range from 32 to 279.
- *CDV* (Cell Delay Variation) can be:
  - ♦ In the range from 1000 to 24000 micro seconds, for T1 (in increments of 125).
  - ♦ In the range from 1000 to32000 micro seconds, for E1 (in increments of 125).
- *CLIP* (Cell Loss Integration Period) can be a value in the range from 1000 to 65535 milliseconds.
- *bufsize* (the egress buffer size, in bytes) can be:
  - ♦ 0 = Autocompute buffer size (must be large enough to hold 8 SAR−PDUs).
  - ♦ Minimum buffer size = 384 bytes (8 cell payloads to one complete seq. num cycle).
  - ♦ Maximum buffer size = 9216 for T1 structured; 16384 for others.
- *clockmode* can be:
  - ♦ 1 = Synchronous (UDT/SDT)
  - ♦ 2 = SRTS (UDT)
  - ♦ 3 = Adaptive (UDT)
- *IdleDetEnable* can be:
  - ♦ 1 = Disable
  - ♦ 2 = Enable
- *ExtIStrig* can be:
  - ♦ 1 = Disable idle suppression
  - ♦ 2 = Enable idle suppression

xcnfchan −chn *chan_num* **−en** *chan_status* **[−cbrserv** *cbr_service* **−clkmode** *mode* **−cdv** *CDV* **−clip** *CLIP* **−maxbuf** *max_buf_size***]**, where&

- *chan_num* can be a numeric value in the range from 16 to 264.
- *chan_status* can be:
  - ♦ 1 = Add
  - ♦ 2 = Delete
  - ♦ 3 = Modify
- *cbr_service* can be:

    - 1 = Unstructured
    - 2 = Structured
- *mode* can be:
    - 1 = Synchronous
    - 2 = SRTS
    - 3 = Adaptive
- *CDV* can be a numeric value in the range from 1 to 65535.
- *CLIP* can be a numeric value in the range from 1000 to 65535.
- *max_buf_size* can be a numeric value in the range from 1 to 35565.

Verify

There is currently no verification procedure available for this configuration.

Troubleshoot

There is currently no specific troubleshooting information available for this configuration.

Related Information

- **Cisco WAN Switching Solutions − Cisco Documentation**
- **Guide to New Names and Colors for WAN Switching Products**
- **Downloads − WAN Switching Software**
- **Technical Support − Cisco Systems**

Updated: Apr 30, 2009 Document ID: 22237