

# E&M Digital CAS 信令的 EM\_PARK 问题疑难解答

## 目录

[简介](#)

[先决条件](#)

[要求](#)

[使用的组件](#)

[背景理论](#)

[规则](#)

[问题](#)

[解决方案](#)

[冒充应答](#)

[相关信息](#)

## 简介

在Cisco 2600、3600和MC3810路由器平台上的数字E&M信令中，某些T1/E1时隙可能停滞在EM\_PARK状态。当您发出show voice call summary命令时，可见此处。本文解释如何发现并解决此问题。

此输出显示某些时隙处于EM\_PARK状态。EM\_PARK状态的时隙不用于语音呼叫。

```
Router#show voice call summary
PORT          CODEC      VAD      VTSP STATE      VPM STATE
=====
1/0:0.1      -         -         -               EM_ONHOOK
1/0:0.2      -         -         -               EM_PARK
1/0:0.3      -         -         -               EM_PARK
1/0:0.4      -         -         -               EM_ONHOOK
1/0:0.5      -         -         -               EM_ONHOOK
```

## 先决条件

### 要求

本文档没有任何特定的要求。

### 使用的组件

本文档中的信息基于以下软件和硬件版本：

- 硬件 — Cisco 2600、Cisco 3600、Cisco VG200和MC3810路由器

- 软件 — 全部

本文档中的信息都是基于特定实验室环境中的设备编写的。本文档中使用的所有设备最初均采用原始（默认）配置。如果您使用的是真实网络，请确保您已经了解所有命令的潜在影响。

## [背景理论](#)

在T1 CAS中，例如，闪烁启动信令，当PBX摘机时，路由器/网关时隙保持空闲(EM\_ONHOOK)状态，直到远程目标应答呼叫。当远程目标应答呼叫时，路由器时隙状态将更改为EM\_OFFHOOK。

如果呼叫未连接，路由器/网关将向主叫方播放带内重排音。由于路由器端的信道状态仍为EM\_ONHOOK，因此路由器无法挂断该信道。呼叫方挂断后，PBX需要将其信道状态从摘机更改为挂机。

在某些情况下，PBX在ABCD过渡的帮助下不发送挂机消息。路由器对此有一种解决方法，称为假应答。如果没有假的答案解决方法，信道将无限期地处于EM\_PARK状态。有关详细信息，[请参阅“假答案”部分](#)。

**注意：**如果语音网关路由器机箱未正确接地，则某些T1信道上的呼叫可能会停滞在EM\_PARK状态。有关电气接地的详细信息，[请参阅硬件安装指南](#)。

## [规则](#)

有关文档规则的详细信息，[请参阅 Cisco 技术提示规则。](#)

## [问题](#)

时隙停滞在EM\_PARK状态可能有两个主要原因：

- 数字信号处理器(DSP)不良，存在硬件或软件问题。
- PSTN交换机/PBX向路由器发送连续摘机信号，但不释放该信号。

## [解决方案](#)

以下是解决此问题的方法：

如果系统中的时隙停滞在EM\_PARK状态，请检查DSP。要检查[DSP，请参阅Cisco 2600/3600系列路由器的NM-HDV上的DSP故障排除](#)。

如果DSP处于活动状态，则问题可能出在PSTN交换机/PBX端或Cisco IOS®（路由器/网关不启动假应答过程）。有关详细信息，[请参阅“假答案”部分](#)。

## [冒充应答](#)

Cisco路由器/网关在播放重新排序音时知道需要从PBX将时隙设置为挂机后，会等待默认值30秒(使用[timeouts wait-release](#) 和[timeouts call-disconnect](#) 命令来更改这些值)。

如果不发生这种情况，路由器将时隙移至EM\_PARK状态，并启动另一个计时器，持续时间为10秒。如果PBX在10秒持续时间后仍不挂机，则路由器会欺骗PBX。路由器发送一秒的假应答，然后挂

机。

路由器发送假应答信号后，路由器将启动另一个计时器，时间为5分钟。如果PBX挂机，计时器停止，路由器将时隙转换到EM\_ONHOOK状态。否则，在5分钟后，它会发送另一个假应答信号，表示持续时间为1秒。路由器重复此过程，直到PBX挂机。路由器强制PBX清除呼叫。

**注意：**由于实际呼叫被清除，此应答转换不会更新到任何记帐记录。但是，PBX将其理解为应答，用户可能会在一秒的持续时间呼叫中付费。

如果与处于EM\_PARK状态的时隙关联的DSP处于活动状态且运行正常，并且问题仍然存在，请运行[debug vpm all](#)和[debug vtsp all](#)命令，以查看Cisco IOS是否尝试发送假应答。

**注意：**您需要运行调试超过五分钟。

**注意：**在大多数情况下，如果DSP损坏，路由器不执行假应答解决方法。有关详细[信息，请参阅Cisco 2600/3600系列路由器NM-HDV上的DSP故障排除](#)。

此调试输出显示了时隙在EM\_PARK中如何卡住，以及假应答解决方法如何工作。

```
Jan 11 17:19:00.767: htsp_dsp_message: SEND/RESP_SIG_STATUS: state=0xC timestamp
=44262 systime=31305235
Jan 11 17:19:00.767: htsp_process_event:
[4/1:1(10), EM_ONHOOK, E_DSP_SIG_1100]em_onhook_offhook htsp_setup_ind
!--- Offhook signal is received from the switch. Jan 11 17:19:00.767: [4/1:1(10)]
get_local_station_id calling num= calling name= calling time=01/11 17:19 Jan 11 17:19:00.767:
vtsp_tsp_call_setup_ind (sdb=0x62BB7B14, tdm_info=0x0, tsp_info=0x62BB4050, calling_number=
calling_oct3 = 0x0, called_number= called_oct3 = 0x81, oct3a=0x0): peer_tag=0 Jan 11
17:19:00.767: : ev.clg.clir is 0 ev.clg.clid_transparent is 0 ev.clg.null_orig_clg is 1
ev.clg.calling_translated is false Jan 11 17:19:00.767: htsp_timer - 3000 msec Jan 11
17:19:00.767: vtsp_do_call_setup_ind Jan 11 17:19:00.767: vtsp_allocate_cdb,cdb 0x62DCEA70 Jan
11 17:19:00.767: vtsp_do_call_setup_ind: Call ID=112722, guid=62DC4230 Jan 11 17:19:00.767:
vtsp_do_call_setup_ind: type=0, under_spec=1640890368, name=, id0=10, id1=1, id2=25038,
calling=, called= subscriber=RegularLine Jan 11 17:19:00.767: vtsp_do_normal_call_setup_ind Jan
11 17:19:00.771: cc_api_call_setup_ind (vdbPtr=0x62BB7FA0, callInfo={called=
,called_oct3=0x81,calling=,calling_oct3=0x0,calling_oct3a=0x0,calling_xlated=fal
se,subscriber_type_str=RegularLine,fdest=0,peer_tag=0, prog_ind=3},callID=0x62DC 40DC) Jan 11
17:19:00.771: cc_api_call_setup_ind type 1 , prot 0 Jan 11 17:19:00.771: vtsp_insert_cdb,cdb
0x62DCEA70 Jan 11 17:19:00.771: vtsp_open_voice_and_set_params Jan 11 17:19:00.771:
dsp_close_voice_channel: [4/1:1:32995] packet_len=8 channel_id=3 packet_id=75 Jan 11
17:19:00.771: dsp_open_voice_channel_20: [4/1:1:32995] packet_len=16 channel_id=3 packet_id=74
alaw_ulaw_select=0 associated_signaling_channel=130 time_slot=2 serial_port=0 Jan 11
17:19:00.771: vtsp_modem_proto_from_cdb: cap_modem_proto 1073741824 Jan 11 17:19:00.771:
vtsp_modem_proto_from_cdb: cap_modem_proto 1073741824 Jan 11 17:19:00.771: dsp_encap_config:
[4/1:1:32995] packet_len=30 channel_id=3 packet_id=92 TransportProtocol 2 t_ssrc=0x0 r_ssrc=0x0
t_vpxcc=0x0 r_vpxcc=0x0 sid_support=1, tse_payload=65535, seq_num=0x0, redundancy=0 Jan 11
17:19:00.771: dsp_set_playout_delay Jan 11 17:19:00.771: dsp_set_playout: [4/1:1:32995]
packet_len=18 channel_id=3 packet_id=76 mode=1 initial=60 min=40 max=200 fax_nom=300
dsp_set_playout_delay_config Jan 11 17:19:00.771: dsp_set_playout_config Jan 11 17:19:00.771:
mode 0, init 60, min 40, max 200 playout default Jan 11 17:19:00.771:
dsp_set_playout_config:mode 0, init 60, min 40, max 200 Jan 11 17:19:00.771:
dsp_set_playout_config: [4/1:1:32995] packet_len=18 channel_id=3 packet_id=76 mode=1 initial=60
min=40 max=200 fax_nom=300 Jan 11 17:19:00.771: dsp_echo_canceler_control: echo_cancel: 1 Jan 11
17:19:00.771: dsp_echo_canceler_control: [4/1:1:32995] echo_cancel 1, disable_hpf 0, flags=0x0,
threshold=-21 Jan 11 17:19:00.771: dsp_echo_canceler_control: [4/1:1:32995] packet_len=12
channel_id=3 packet_id=66 flags=0x0, threshold=-21 Jan 11 17:19:00.771: set_gains: FXX/E&M: msg-
>message.set_codec_gains.out_gain=0 Jan 11 17:19:00.771: dsp_set_gains: [4/1:1:32995]
packet_len=12 channel_id=3 packet_id=91 in_gain=0 out_gain=0 Jan 11 17:19:00.771:
dsp_vad_enable: [4/1:1:32995] enable: packet_len=12 channel_id=3 packet_id=78 thresh=-38 Jan 11
```

17:19:00.771: cc\_process\_call\_setup\_ind (event=0x62E63ACC) Jan 11 17:19:00.771: >>>>CCAPI handed  
cid 32995 with tag 0 to app "DEFAULT" Jan 11 17:19:00.771: sess\_appl:  
ev(24=CC\_EV\_CALL\_SETUP\_IND), cid(32995), disp(0) Jan 11 17:19:00.771: sess\_appl:  
ev(SSA\_EV\_CALL\_SETUP\_IND), cid(32995), disp(0) Jan 11 17:19:00.771: ssaCallSetupInd Jan 11  
17:19:00.771: ccCallSetContext (callID=0x80E3, context=0x62DFBCF0) Jan 11 17:19:00.771:  
ssaCallSetupInd cid(32995), st(SSA\_CS\_MAPPING),oldst(0), ev (24)ev-  
>e.evCallSetupInd.nCallInfo.finalDestFlag = 0 Jan 11 17:19:00.771: ccCallSetupAck  
(callID=0x80E3) Jan 11 17:19:00.771: ccGenerateTone (callID=0x80E3 tone=8) Jan 11 17:19:00.771:  
ccCallReportDigits (callID=0x80E3, enable=0x1) Jan 11 17:19:00.771: vtsp\_report\_digit\_control:  
enable=1: digit reporting enabled Jan 11 17:19:00.771: cc\_api\_call\_report\_digits\_done  
(vdbPtr=0x62BB7FA0, callID=0x80E3, disp=0) Jan 11 17:19:00.771: : vtsp\_get\_digit\_timeouts Jan 11  
17:19:00.771: sess\_appl: ev(52=CC\_EV\_CALL\_REPORT\_DIGITS\_DONE), cid(32995), disp(0) Jan 11  
17:19:00.771: cid(32995)st(SSA\_CS\_MAPPING)ev (SSA\_EV\_CALL\_REPORT\_DIGITS\_DONE)  
oldst(SSA\_CS\_MAPPING)cfid(-1)csz(0)in(1)fDest(0) Jan 11 17:19:00.771: ssaReportDigitsDone  
cid(32995) peer list: (empty) Jan 11 17:19:00.771: ssaReportDigitsDone callid=32995 Enable  
succeeded Jan 11 17:19:00.771: ccGenerateTone (callID=0x80E3 tone=8) Jan 11 17:19:00.771:  
vtsp:[4/1:1:32995, S\_SETUP\_INDICATED, E\_CC\_SETUP\_ACK] Jan 11 17:19:00.775: act\_setup\_ind\_ack Jan  
11 17:19:00.775: vtsp\_modem\_proto\_from\_cdb: cap\_modem\_proto 0 Jan 11 17:19:00.775:  
vtsp\_modem\_proto\_from\_cdb: cap\_modem\_proto 0 Jan 11 17:19:00.775: dsp\_encap\_config:  
[4/1:1:32995] packet\_len=30 channel\_id=3 packet\_id=92 TransportProtocol 2 t\_ssrc=0x0 r\_ssrc=0x0  
t\_vpxcc=0x0 r\_vpxcc=0x0 sid\_support=1, tse\_payload=65535, seq\_num=0x0, redundancy=0 Jan 11  
17:19:00.775: dsp\_voice\_mode: [4/1:1:32995] cdb 62DCEA70, cdb->codec\_params.modem 2,  
inband\_detect flags 0x21 Jan 11 17:19:00.775: map\_dtmf\_relay\_type--digit relay mode: 2 Jan 11  
17:19:00.775: dsp\_voice\_mode: [4/1:1:32995] packet\_len=24 channel\_id=3 packet\_id=73  
coding\_type=1 voice\_field\_size=160 VAD\_flag=0 echo\_length=256 comfort\_noise=1 inband\_detect=33  
digit\_relay\_mode=2 AGC\_flag=0act\_setup\_ind\_ack: modem\_mode = 0, fax\_relay\_on = 1 Jan 11  
17:19:00.775: act\_setup\_ind\_ack(): dsp\_dtmf\_mode() dsp\_dtmf\_mode(VTSP\_TONE\_DTMF\_MODE) Jan 11  
17:19:00.775: dsp\_dtmf\_mode: [4/1:1:32995] packet\_len=10 channel\_id=3 packet\_id=65 dtmf\_or\_mf=0  
Jan 11 17:19:00.775: vtsp\_timer: 31305236 Jan 11 17:19:00.775: vtsp:[4/1:1:32995,  
S\_DIGIT\_COLLECT, E\_CC\_GEN\_TONE] Jan 11 17:19:00.775: act\_gen\_tone Jan 11 17:19:00.775:  
dsp\_cp\_tone\_off: [4/1:1:32995] packet\_len=8 channel\_id=3 packet\_id=71 Jan 11 17:19:00.775:  
dsp\_cp\_tone\_on: [4/1:1:32995] packet\_len=38 channel\_id=3 packet\_id=72 tone\_id=4 n\_freq=2  
freq\_of\_first=350 freq\_of\_second=440 amp\_of\_first=5514 amp\_of\_second=5514 direction=1  
on\_time\_first=65535 off\_time\_first=0 on\_time\_second=0 off\_time\_second=0 Jan 11 17:19:00.775:  
vtsp:[4/1:1:32995, S\_DIGIT\_COLLECT, E\_CC\_GEN\_TONE] Jan 11 17:19:00.775: act\_gen\_tone Jan 11  
17:19:00.775: dsp\_cp\_tone\_off: [4/1:1:32995] packet\_len=8 channel\_id=3 packet\_id=71 Jan 11  
17:19:00.775: dsp\_cp\_tone\_on: [4/1:1:32995] packet\_len=38 channel\_id=3 packet\_id=72 tone\_id=4  
n\_freq=2 freq\_of\_first=350 freq\_of\_second=440 amp\_of\_first= 5514 amp\_of\_second=5514 direction=1  
on\_time\_first=65535 off\_time\_first=0 on\_time4\_second=0 off\_time\_second=0 Jan 11 17:19:00.775:  
htsp\_process\_event: [4/1:1(10), EM\_WAIT\_SETUP\_ACK, E\_HTSP\_SETUP\_ACK]em\_wait\_setup\_ack\_get\_ack  
Jan 11 17:19:00.775: htsp\_timer\_stop Jan 11 17:19:00.775: htsp\_timer2 - 172 msec Jan 11  
17:19:00.947: htsp\_process\_event: [4/1:1(10), EM\_WAIT\_SETUP\_ACK,  
E\_HTSP\_EVENT\_TIMER2]em\_wait\_prewink\_timer **Jan 11 17:19:00.947: em\_offhook (0)[recEive and  
transMit4/1:1(10)] set signal st  
ate = 0x8em\_onhook (200)[recEive and transMit4/1:1(10)] set signal state = 0x0  
!--- A wink of duration 200 msec is sent out to the switch.** Jan 11 17:19:01.471:  
vtsp\_process\_dsp\_message: MSG\_TX\_DTMF\_DIGIT\_BEGIN: digit=9, rtp\_timestamp=0xED31C493 Jan 11  
17:19:01.471: vtsp:[4/1:1:32995, S\_DIGIT\_COLLECT, E\_DSP\_DTMF\_DIGIT\_BEGIN] Jan 11 17:19:01.471:  
act\_report\_digit\_begin Jan 11 17:19:01.471: cc\_api\_call\_digit\_begin (dstVdbPtr=0x0,  
dstCallId=0xFFFFFFFF F, srcCallId=0x80E3, digit=9, digit\_begin\_flags=0x1,  
rtp\_timestamp=0xED31C493 rtp\_expiration=0x0, dest\_mask=0x1) Jan 11 17:19:01.471: sess\_appl:  
ev(10=CC\_EV\_CALL\_DIGIT\_BEGIN), cid(32995), disp(0) Jan 11 17:19:01.471:  
cid(32995)st(SSA\_CS\_MAPPING)ev(SSA\_EV\_DIGIT\_BEGIN) oldst(SSA\_CS\_MAPPING)cfid(-  
1)csz(0)in(1)fDest(0) Jan 11 17:19:01.471: ssaIgnore cid(32995), st(SSA\_CS\_MAPPING),oldst(0),  
ev(10) Jan 11 17:19:01.503: vtsp\_process\_dsp\_message: MSG\_TX\_DTMF\_DIGIT\_OFF: digit=9,  
duration=65 Jan 11 17:19:01.503: vtsp:[4/1:1:32995, S\_DIGIT\_COLLECT, E\_DSP\_DTMF\_DIGIT] Jan 11  
17:19:01.503: act\_report\_digit\_end Jan 11 17:19:01.503: vtsp\_timer\_stop: 31305308 Jan 11  
17:19:01.503: dsp\_cp\_tone\_off: [4/1:1:32995] packet\_len=8 channel\_id=3 pa cket\_id=71 Jan 11  
17:19:01.503: cc\_api\_call\_digit\_end (dstVdbPtr=0x0, dstCallId=0xFFFFFFFF, srcCallId=0x80E3,  
digit=9,duration=65,xruleCallingTag=0,xruleCalledTag=0, dest\_mask=0x1), digi t\_tone\_mode=0 Jan  
11 17:19:01.503: htsp\_digit\_ready: digit = 39 Jan 11 17:19:01.503: vtsp\_timer: 31305308 Jan 11  
17:19:01.503: htsp\_process\_event: [4/1:1(10), EM\_OFFHOOK, E\_VTSP\_DIGIT]em\_offhook\_digit\_collect  
Jan 11 17:19:01.503: sess\_appl: ev(9=CC\_EV\_CALL\_DIGIT\_END), cid(32995), disp(0) Jan 11  
17:19:01.503: cid(32995)st(SSA\_CS\_MAPPING)ev(SSA\_EV\_CALL\_DIGIT) oldst(SSA\_CS\_MAPPING)cfid(-

1)csz(0)in(1)fDest(0) Jan 11 17:19:01.503: ssaDigit Jan 11 17:19:01.503: ssaDigit, 0. sct->digit , sct->digit len 0, usrDigit 9, digit\_tone\_mode=0 Jan 11 17:19:01.503: ssaDigit,1. callinfo.called , digit 9, callinfo.calling , xrulecallingtag 0, xrulecalledtag 0 Jan 11 17:19:01.503: ssaDigit, 7. callinfo.calling , sct->digit 9, result 1 Jan 11 17:19:01.603: vtsp\_process\_dsp\_message: MSG\_TX\_DTMF\_DIGIT\_BEGIN: digit=1, rtp\_timestamp=0xED31C493 Jan 11 17:19:01.603: vtsp:[4/1:1:32995, S\_DIGIT\_COLLECT, E\_DSP\_DTMF\_DIGIT\_BEGIN] Jan 11 17:19:01.603: act\_report\_digit\_begin Jan 11 17:19:01.603: cc\_api\_call\_digit\_begin (dstVdbPtr=0x0, dstCallId=0xFFFFFFFF F, srcCallId=0x80E3, digit=1, digit\_begin\_flags=0x1, rtp\_timestamp=0xED31C493 rtp\_expiration=0x0, dest\_mask=0x1) Jan 11 17:19:01.603: sess\_appl: ev(10=CC\_EV\_CALL\_DIGIT\_BEGIN), cid(32995), disp(0) Jan 11 17:19:01.603: cid(32995)st(SSA\_CS\_MAPPING)ev(SSA\_EV\_DIGIT\_BEGIN) oldst(SSA\_CS\_MAPPING)cfid(-1)csz(0)in(1)fDest(0) Jan 11 17:19:01.603: ssaIgnore cid(32995), st(SSA\_CS\_MAPPING),oldst(0), ev(10) Jan 11 17:19:01.643: vtsp\_process\_dsp\_message: MSG\_TX\_DTMF\_DIGIT\_OFF: digit=1, duration=75 Jan 11 17:19:01.643: vtsp:[4/1:1:32995, S\_DIGIT\_COLLECT, E\_DSP\_DTMF\_DIGIT] Jan 11 17:19:01.643: act\_report\_digit\_end Jan 11 17:19:01.643: vtsp\_timer\_stop: 31305322 Jan 11 17:19:01.643: cc\_api\_call\_digit\_end (dstVdbPtr=0x0, dstCallId=0xFFFFFFFF, srcCallId=0x80E3, digit=1,duration=75,xruleCallingTag=0,xruleCalledTag=0, dest\_mask=0x1), digit\_tone\_mode=0 Jan 11 17:19:01.643: htsp\_digit\_ready: digit = 31 Jan 11 17:19:01.643: vtsp\_timer: 31305322 Jan 11 17:19:01.643: htsp\_process\_event: [4/1:1(10), EM\_OFFHOOK, E\_VTSP\_DIGIT]em\_offhook\_digit\_collect Jan 11 17:19:01.643: sess\_appl: ev(9=CC\_EV\_CALL\_DIGIT\_END), cid(32995), disp(0) Jan 11 17:19:01.643: cid(32995)st(SSA\_CS\_MAPPING)ev(SSA\_EV\_CALL\_DIGIT) oldst(SSA\_CS\_MAPPING)cfid(-1)csz(0)in(1)fDest(0) Jan 11 17:19:01.643: ssaDigit Jan 11 17:19:01.643: ssaDigit, 0. sct->digit 9, sct->digit len 1, usrDigit 1, digit\_tone\_mode=0 Jan 11 17:19:01.643: ssaDigit,1. callinfo.called , digit 91, callinfo.calling , xrulecallingtag 0, xrulecalledtag 0 Jan 11 17:19:01.643: ssaDigit, 7. callinfo.calling , sct->digit 91, result 1 Jan 11 17:19:01.743: vtsp\_process\_dsp\_message: MSG\_TX\_DTMF\_DIGIT\_BEGIN: digit=8, rtp\_timestamp=0xED31C493 Jan 11 17:19:01.743: vtsp:[4/1:1:32995, S\_DIGIT\_COLLECT, E\_DSP\_DTMF\_DIGIT\_BEGIN] Jan 11 17:19:01.743: act\_report\_digit\_begin Jan 11 17:19:01.743: cc\_api\_call\_digit\_begin (dstVdbPtr=0x0, dstCallId=0xFFFFFFFF F, srcCallId=0x80E3, digit=8, digit\_begin\_flags=0x1, rtp\_timestamp=0xED31C493 rtp\_expiration=0x0, dest\_mask=0x1) Jan 11 17:19:01.743: sess\_appl: ev(10=CC\_EV\_CALL\_DIGIT\_BEGIN), cid(32995), disp(0) Jan 11 17:19:01.743: cid(32995)st(SSA\_CS\_MAPPING)ev(SSA\_EV\_DIGIT\_BEGIN) oldst(SSA\_CS\_MAPPING)cfid(-1)csz(0)in(1)fDest(0) Jan 11 17:19:01.743: ssaIgnore cid(32995), st(SSA\_CS\_MAPPING),oldst(0), ev(10) radius\_decrypt: null length Jan 11 17:19:01.843: vtsp\_process\_dsp\_message: MSG\_TX\_DTMF\_DIGIT\_OFF: digit=8, duration=75 Jan 11 17:19:01.843: vtsp:[4/1:1:32995, S\_DIGIT\_COLLECT, E\_DSP\_DTMF\_DIGIT] Jan 11 17:19:01.843: act\_report\_digit\_end Jan 11 17:19:01.843: vtsp\_timer\_stop: 31305342 Jan 11 17:19:01.843: cc\_api\_call\_digit\_end (dstVdbPtr=0x0, dstCallId=0xFFFFFFFF, srcCallId=0x80E3, digit=8,duration=75,xruleCallingTag=0,xruleCalledTag=0, dest\_mask=0x1), digit\_t\_tone\_mode=0 Jan 11 17:19:01.843: htsp\_digit\_ready: digit = 38 Jan 11 17:19:01.843: vtsp\_timer: 31305342 Jan 11 17:19:01.843: htsp\_process\_event: [4/1:1(10), EM\_OFFHOOK, E\_VTSP\_DIGIT]em\_offhook\_digit\_collect Jan 11 17:19:01.843: sess\_appl: ev(9=CC\_EV\_CALL\_DIGIT\_END), cid(32995), disp(0) Jan 11 17:19:01.843: cid(32995)st(SSA\_CS\_MAPPING)ev(SSA\_EV\_CALL\_DIGIT) oldst(SSA\_CS\_MAPPING)cfid(-1)csz(0)in(1)fDest(0) Jan 11 17:19:01.843: ssaDigit Jan 11 17:19:01.843: ssaDigit, 0. sct->digit 91, sct->digit len 2, usrDigit 8, digit\_tone\_mode=0 Jan 11 17:19:01.843: ssaDigit,1. callinfo.called , digit 918, callinfo.calling , xrulecallingtag 0, xrulecalledtag 0 Jan 11 17:19:01.843: ssaDigit, 7. callinfo.calling , sct->digit 918, result -1 Jan 11 17:19:01.843: ccCallDisconnect (callID=0x80E3, cause=0x1C tag=0x0) Jan 11 17:19:01.843: vtsp:[4/1:1:32995, S\_DIGIT\_COLLECT, E\_CC\_DISCONNECT] Jan 11 17:19:01.843: act\_pre\_con\_disconnect Jan 11 17:19:01.843: vtsp\_ring\_noan\_timer\_stop: 31305342 Jan 11 17:19:01.843: dsp\_cp\_tone\_off: [4/1:1:32995] packet\_len=8 channel\_id=3 packet\_id=71 Jan 11 17:19:01.843: dsp\_voice\_mode: [4/1:1:32995] cdb 62DCEA70, cdb->codec\_para ms.modem 2, inband\_detect flags 0x21 Jan 11 17:19:01.843: map\_dtmf\_relay\_type--digit relay mode: 2 Jan 11 17:19:01.843: dsp\_voice\_mode: [4/1:1:32995] packet\_len=24 channel\_id=3 packet\_id=73 coding\_type=1 voice\_field\_size=160 VAD\_flag=0 echo\_length=256 comfort\_noise=1 inband\_detect=33 digit\_relay\_mode=2 AGC\_flag=0 Jan 11 17:19:01.843: **dsp\_cp\_tone\_on: [4/1:1:32995] packet\_len=38 channel\_id=3 packet\_id=72 tone\_id=3 n\_freq=2 freq\_of\_first=480 freq\_of\_second=620amp\_of\_first=5206 amp\_of\_second=2928 direction=1 on\_time\_first=250 off\_time\_first=250 on\_time\_second=0 off\_time\_second=0** Jan 11 17:19:01.843: vtsp\_timer: 31305342 Jan 11 17:19:01.843: htsp\_pre\_connect\_disconnect, cdb = 62DCEA70 cause = 1C !--- Since the call is disconnected because the number received is "unassigned" !--- or "invalid" the router starts to play the reorder !--- tone and a timer, which is the wait-release !--- timeout timer, starts with default 30 seconds. !--- This call is disconnected !--- prior to



the connect state. Jan 11 17:19:01.843: htsp\_process\_event: [4/1:1(10), EM\_OFFHOOK, E\_HTSP\_PRE\_CONN\_DISC] Jan 11 17:19:31.844: vtsp\_main: timer: 31308342  
!--- The wait-release timer expires after 30 seconds. Jan 11 17:19:31.844: vtsp:[4/1:1:32995, S\_WAIT\_RELEASE\_NC, E\_TIMER]  
!--- The VTSP module is in a wait release state for that call. It also receives !--- event timer, which means that the timer expires so that it !--- goes into another state. Jan 11 17:19:31.844: act\_pre\_con\_disc\_rel htsp\_release\_req: cause 28, no\_onhook 0 Jan 11 17:19:31.844: htsp\_process\_event: [4/1:1(10), EM\_OFFHOOK, E\_HTSP\_RELEASE\_REQ]em\_offhook\_release  
Jan 11 17:19:31.844: htsp\_timer\_stop2 em\_onhook (0)[recEive and transMit4/1:1(10)] set signal state = 0x0  
Jan 11 17:19:31.844: htsp\_timer\_stop  
Jan 11 17:19:31.844: em\_start\_timer: 400 ms  
Jan 11 17:19:31.844: htsp\_timer - 400 msec  
!--- HTSP receives an event that requests the release of !--- the time slot and it goes into EM wait !--- onhook state. But, it cannot do anything since it says I am onhook already. !--- Also, the router starts a timer of 400 msec. Jan 11 17:19:32.296: htsp\_process\_event: [4/1:1(10), EM\_WAIT\_ONHOOK, E\_HTSP\_EVENT\_TIMER]em\_wait\_timeout  
Jan 11 17:19:32.296: em\_stop\_timers  
Jan 11 17:19:32.296: htsp\_timer\_stop  
Jan 11 17:19:32.296: em\_start\_timer: 400 ms  
Jan 11 17:19:32.296: htsp\_timer - 400 msec  
!--- When the 400 msec timer expires, HTSP gets into EM clear pending state. !--- It also starts another timer of 400 msec. Jan 11 17:19:32.696: htsp\_process\_event: [4/1:1(10), EM\_CLR\_PENDING, E\_HTSP\_EVENT\_TIMER]em\_clr\_timeout Jan 11 17:19:32.696: em\_stop\_timers Jan 11 17:19:32.696: htsp\_timer\_stop Jan 11 17:19:32.696: em\_start\_timer: 10000 ms Jan 11 17:19:32.696: htsp\_timer - 10000 msec Jan 11 17:19:32.700: htsp\_dsp\_message: SEND/RESP\_SIG\_STATUS: state=0xC timestamp=1533 systime=31308428 Jan 11 17:19:32.700: htsp\_process\_event: [4/1:1(10), EM\_PARK, E\_DSP\_SIG\_1100]em\_park\_offhook !--- When the 400 msec timer expires, the router puts the time slot into !--- the EM\_PARK state, and it starts another timer of 10 seconds. !--- The router still sees the ABCD=1100 from the switch. Jan 11 17:19:42.760: htsp\_process\_event: [4/1:1(10), EM\_PARK, E\_HTSP\_EVENT\_TIMER]em\_park\_timerhtsp\_report\_onhook\_sig  
Jan 11 17:19:42.760: em\_offhook (0)[recEive and transMit4/1:1(10)] set signal state = 0x8em\_onhook (1000)[recEive and transMit4/1:1(10)] set signal state = 0x0  
Jan 11 17:19:42.760: htsp\_timer2 - 300000 msec  
Jan 11 17:19:42.760: htsp\_process\_event: [4/1:1(10), EM\_PARK, E\_HTSP\_EVENT\_TIMER]em\_park\_timerhtsp\_report\_onhook\_sig  
Jan 11 17:19:42.760: em\_offhook (0)[recEive and transMit4/1:1(10)] set signal state = 0x8em\_onhook (1000)[recEive and transMit4/1:1(10)] set signal state = 0x0  
Jan 11 17:19:42.760: htsp\_timer2 - 300000 msec  
!--- As seen from the timestamps, when the timer expires in ten seconds, !--- the router goes offhook for one second (1000 msec) and then onhook. !--- It also starts another timer of 300000 msec (5 minutes).

## 相关信息

- [语音技术支持](#)
- [语音和统一通信产品支持](#)
- [Cisco IP 电话故障排除](#)
- [技术支持和文档 - Cisco Systems](#)