# CAS Protocols Reference Manual

**P/N 6675-10**


**Natural MicroSystems Corporation**
**100 Crossing Blvd.**
**Framingham, MA 01702**

## Revision History

| Revision | Release Date | Notes |
|---|---|---|
| 1.0 | June, 1999 | SJC |
| This manual printed: June 23, 1999 | | |

# Table of Contents

# About This Manual

The *CAS Protocols Reference Manual* presents reference information about NMS Trunk Control Programs (TCPs), scripts that implement telephony protocols in NMS' Alliance Generation architecture. This version of the manual contains information about two TCPs (among several) in the NMS AG CAS product. (More TCPs will be added in future versions.) This manual is for use by NMS support and sales engineers, and by external organizations wishing to use the TCP development kit for implementation of their own TCPs in NMS architecture. Each individual TCP is described within a chapter, outlining its capabilities and requirements.

Within each chapter, a TCP's line and register signaling protocols are discussed through the use of text (for description) and tables (for raw data organization). Also within these chapters is a comprehensive list of the parameters used in programming each TCP. Then, to illustrate the TCP's reaction to various scenarios, diagrams are given showing the state of the TCP at significant points in the cycle. Finally, each chapter ends with a set of test data (shown through charts and log files) to analyze the performance of the TCP and its influences on MIPS and DSP resources.

This manual is organized as follows:

| **For...** | **See...** |
|---|---|
| Information on implementing the MFC-R2 TCP | Chapter 1 - MFC-R2 |
| Information on implementing the R1.5 TCP | Chapter 2 - R1.5 |
| A glossary of terms | Appendix A - Glossary |

## Related Documents

For more information, refer to the following manuals:

| This manual... | Provides... |
|---|---|
| *CT Access Developer's Reference Manual* | An overview of CT Access and a reference of functions, events, reason codes, and errors. |
| *AG CAS Installation and Developer's Manual* | Describes how to install and run NMS Channel Associated Signaling (CAS) protocol software. |
| *AG Runtime Configuration and Developer's Manual* | Information about the AG configuration file, `agmon`, AG board and driver error messages, and AG task processors. |
| *TCP Developer's Kit Manuals* | Consists of manuals and tools used to create Trunk Control Programs for use on Alliance Generation Natural MicroSystems telephony boards. |
| *CASmon User's Manual* | Information about installing and using CASmon, a tool for monitoring and testing digital trunks and protocols. |
| *PROTtest User's Manual* | Information on installing and using PROTtest, a tool that provides an environment for testing telephony protocols. |

The following manuals provide an overview of the capabilities and functions of the services that use CT Access:

- *ADI Service Developer's Manual*
- *ADI Service Function Reference Manual*
- *T1/E1 Digital Trunk Monitor Service Developer's Reference Manual*

# Developer Support

Natural MicroSystems provides quality assistance to users of its products. If you have any technical questions or problems, please contact us using any of the following addresses and telephone numbers:


**Natural MicroSystems Corporation**
100 Crossing Blvd.
Framingham, MA 01702 USA

| | |
|---|---|
| Developer Support | (508) 271-1333 (8:30-8:00 eastern time) |
| Toll-Free within USA | (800) 533-6120 (8:30-5:30 eastern time) |
| Corporate Headquarters | (508) 620-9300 (8:30-5:30 eastern time) |
| Fax | (508) 271-1123 |
| Internet | tech_support@nmss.com |
| World Wide Web | http://www.nmss.com |

**Natural MicroSystems Europe S.A.**
Immeuble Copernic
Parc Technologique
18-22, Rue Edouard-Herriot - 92356
LE PLESSIS ROBINSON Cedex
France

| | |
|---|---|
| Developer Support | 33 (0)1 46 01 40 50 (9:00-6:00 Paris time) |
| Fax | 33 (0)1 46 01 40 99 |
| Internet | europe_support@nmss.com |

**Natural MicroSystems Asia**
Room 1815-16 Concordia Plaza
#1 Science Museum Road
Tsim Sha Tsui East
Kowloon, Hong Kong

| | |
|---|---|
| NMS Asia | +852.2926.1820 |
| Fax | +852.2620.5600 |
| Internet | asia_support@nmss.com |

Developer Support engineers are available Monday through Friday. To assist the engineer who answers your call, please make sure to do the following:

- Write down what you were doing when the problem occurred, listing the steps you followed before the problem occurred and the exact text of any error messages displayed on your screen.
- Know which NMS telephony boards are installed in your system including the AG board model number, part number, revision number, and serial number (i.e., model AG-8, P/N 2450, rev. D5, S/N 19564). The last three items are on a single sticker on the back of the board.
- Have some knowledge of the computer you are using including the type of the computer, the speed of the processor, the amount of memory (RAM), and the operating system and its version number.
- Know which version of NMS drivers and software you have installed and the names, sizes, and creation dates of any files that were downloaded from the NMS web site.
- Be prepared to submit by fax or e-mail, copies of configuration files and diagnostic files such as *`ag.cfg`*, *`ag.rpt`*, *`agerror.log`*, *`qx.cfg`*, *`qx.rpt`*, *`qxerror.log`*, *`cta.cfg`*, and applicable code fragments from your application.

Chapter 1

# MFC-R2

## 1.1 Introduction

This chapter characterizes the Natural MicroSystems MFC-R2 protocol package, consisting of:

- Two Trunk Control Programs (TCPs) that run on Natural MicroSystems AG Boards, performing call setup functions with the MFC-R2 line protocol. The TCPs interact with an application running on the host through the NMS standard API (CT Access), by receiving commands and sending events according to CT Access's specifications.
- A set of parameter files, each programming the TCP to execute the MFC-R2 variation used in a particular country or on a particular network.

This chapter is divided into four sections dealing with the TCPs:

| Section | Description |
|---|---|
| Line Signaling | Set of signals that represent fixed events in the course of call setup. |
| Register Signaling | The way in which the protocol transmits information as it changes for each call; as addresses and category specifications for the call. |
| Parameters | A description of all the parameters the host can send the TCP to program its behavior. |
| Country-specific Implementation | A list of variations from the CCITT standard for each country which MFC-R2 can be implemented. |

## 1.2 MFC-R2 Line Signaling

This section summarizes the MFC-R2 line signaling protocol implemented by NMS's MFC-R2 TCP, in the normal case where no errors occur for the duration of the call. The TCP follows the CCITT recommendations Q.421 and Q.422 (CCITT Blue Book Volume VI, Fascicle VI.4, Geneva 1989).

Although E1 Channel Associated Signaling (CAS) framing supports 4 signaling bits, only 2 of them (per direction) are used for R2 line signaling. Thus the signaling channels supporting the R2 line signaling protocol are referred to as $A_f$ and $B_f$ in the forward direction, and $A_b$ and $B_b$, in the backward direction. The forward channel indicates the condition of the outgoing switch equipment and reflects the condition of the *calling* party's line. The backward channel indicates

the condition of the *called* party's line. The following table summarizes the signaling states of a typical call:

| **State** | **Outbound $A_fB_f$** | **Direction** | **Inbound $A_bB_b$** |
|---|---|---|---|
| Idle | 10 | ← → | 10 |
| Seizure | 00 | → | 01 |
| Seizure Acknowledged | 00 | ← | 11 |
| The outbound side starts to send the address information, using in-band compelled MF tones. The inbound side completes the compelled sequence by accepting or rejecting the call, using the last backward compelled tone. If the call has been *accepted*, the inbound side plays a ring tone on the line, and then signals that the call has been answered by setting the $A_b$ bit to 0. | | | |
| Ringing | 00 | ← | 11 |
| Answer - conversation state | 00 | ← | 01 |
| If the inbound side *rejects* the call, the outbound side clears forward by setting the $A_f$ bit to 1. The inbound side goes back to idle, by setting the $B_b$ bit to 0. | | | |
| Clear forward | 10 | → | 11 |
| Idle | 10 | ← | 10 |
| During conversation, the outbound protocol can receive *billing pulses*, to signal that a unit of cost has been billed to the call. The bit used to carry a billing pulse depends upon national specifications. | | | |
| Answer - conversation state | 00 | ← | 01 |
| Billing pulses | 00 | ← | 11 or 00 |
| Answer - conversation state | 00 | ← | 01 |
| Depending upon which of the sides hangs up the call first, a *clear back* signal, or a *clear forward* signal occurs. Depending on national specifications, there might be a period of time in which the inbound side holds a *release guard* state. This state is the same as *clear back* but happens when the outbound side is already in the idle state. Idle follows. | | | |
| Inbound side hangs up first: Clear back | 00 | ← | 11 |
| Clear forward | 10 | → | 11 |

| State | Outbound $A_fB_f$ | Direction | Inbound $A_bB_b$ |
|---|---|---|---|
| Idle | 10 | ← → | 10 |
| Outbound side hangs up first: Clear forward | 10 | → | 01 |
| Release guard | 10 | ← | 11 |
| Idle | 10 | ← → | 10 |

## 1.3 MFC-R2 Register Signaling

This section defines the register signaling protocol implemented by the NMS MFC-R2 TCP. The protocol is defined by the CCITT recommendations Q.440 to Q.442 (CCITT Blue Book Volume VI, Fascicle VI.4, Geneva 1989).

The protocol consists in the exchange of multi-frequency tones, following a compelling sequence.

The tones used are composed of two single-frequencies each, according to the following table:

| | **Single Tone #** | **1** | **2** | **3** | **4** | **5** | **6** |
|---|---|---|---|---|---|---|---|
| **Digits** | **Forward (Hz)** | 1380 | 1500 | 1620 | 1740 | 1860 | 1980 |
| | **Backward (Hz)** | 1140 | 1020 | 900 | 780 | 660 | 540 |
| 1 | | **X** | **X** | | | | |
| 2 | | **X** | | **X** | | | |
| 3 | | | **X** | **X** | | | |
| 4 | | **X** | | | **X** | | |
| 5 | | | **X** | | **X** | | |
| 6 | | | | **X** | **X** | | |
| 7 | | **X** | | | | **X** | |
| 8 | | | **X** | | | **X** | |
| 9 | | | | **X** | | **X** | |
| 10 | | | | | **X** | **X** | |
| 11 | | **X** | | | | | **X** |
| 12 | | | **X** | | | | **X** |
| 13 | | | | **X** | | | **X** |
| 14 | | | | | **X** | | **X** |
| 15 | | | | | | **X** | **X** |

This table is interpreted as follows: each tone is composed of the two single frequency tones marked with an **X**. There are two kinds of tones: forward and backward. Forward tones are defined as the combination of the frequencies in the *forward* row. Backward tones are defined as the combination of the frequencies in

the *backward* row. For example, a forward 15 is composed of (1860+1980) Hz; a backward 1 is composed of (1140+1020) Hz.

Tones have specific meanings in the context of the call set-up. Forward tones are divided into groups of meaning, used to convey both address information (the number being called), and information about the call (the caller's number, and the call's category). This grouping is controlled by the receiving party, using backward tones to request specific information from the caller. The backward tones are divided into groups, so a different meaning is attached to them depending upon the logical group to which they belong.

Under the CCITT standard, there are two classes of forward tones, Group I and II (some countries have a Group III), and two classes of backward tones, Group A and B, (some countries also have a Group C).

The following tables list the standard CCITT meanings of the different tones in the various groups. A blank and shaded cell signifies a tone that, although specified by CCITT, is not supported by the current release of NMS's MFC-R2 protocol package.

| **Forward Group I** | | **Forward Group II** | |
|---|---|---|---|
| Designation of the Signal | Meaning of the Signal | Designation of the Signal | Meaning of the Signal |
| I-1 | digit 1 | II-1 | Subscriber without priority |
| I-2 | digit 2 | II-2 | Subscriber with priority |
| I-3 | digit 3 | II-3 | Maintenance equipment |
| I-4 | digit 4 | II-4 | Spare |
| I-5 | digit 5 | II-5 | Operator |
| I-6 | digit 6 | II-6 | Data transmission |
| I-7 | digit 7 | II-7 | Subscriber (international) |
| I-8 | digit 8 | II-8 | Data transmission (international) |
| I-9 | digit 9 | II-9 | Subscriber with priority (international) |
| I-10 | digit 0 | II-10 | Operator with forward transfer facility (international) |

| **Forward Group I** | | **Forward Group II** | |
|---|---|---|---|
| Designation of the Signal | Meaning of the Signal | Designation of the Signal | Meaning of the Signal |
| I-11 | | II-11 | Spare for national use |
| I-12 | Request not accepted | II-12 | Spare for national use |
| I-13 | Satellite link not included (1) | II-13 | Spare for national use |
| I-14 | Incoming half-echo suppression required (2)<br>Satellite link included (1) | II-14 | Spare for national use |
| I-15 | End of information | II-15 | Spare for national use |

*1. This tone meaning arises in response to a A-13 request from the inbound side (send nature of circuit).*

*2. This tone meaning arises in response to a A-14 request from the inbound side (request information on half-echo suppression).*

| **Backward Group A** | | **Backward Group B** | |
|---|---|---|---|
| Designation of the Signal | Meaning of the Signal | Designation of the Signal | Meaning of the Signal |
| A-1 | Send next digit (N+1) | B-1 | Spare for national use |
| A-2 | Send last digit (N-1) | B-2 | Send special information tone |
| A-3 | Address completed, change to reception of Group B | B-3 | Subscriber line busy |
| A-4 | Congestion in the national network | B-4 | Congestion |
| A-5 | Send calling party’s category | B-5 | Unallocated number |
| A-6 | Address complete, charge, setup speech conditions | B-6 | Subscriber line free, charge |
| A-7 | Send second-to-last digit (N-2) | B-7 | Subscriber line free, no charge |
| A-8 | Send third-from-last digit (N-3) | B-8 | Subscriber line out of order |

| **Backward Group A** | | **Backward Group B** | |
|---|---|---|---|
| Designation of the Signal | Meaning of the Signal | Designation of the Signal | Meaning of the Signal |
| A-9 | Spare for national use | B-9 | Spare for national use |
| A-10 | Spare for national use | B-10 | Spare for national use |
| A-11 | | B-11 | Spare for national use |
| A-12 | | B-12 | Spare for national use |
| A-13 | Send nature of circuit | B-13 | Spare for national use |
| A-14 | Request info on use of half-echo suppression | B-14 | Spare for national use |
| A-15 | | B-15 | Spare for national use |

The meaning of both forward and backward tones vary widely for different countries. See Section 1.5 for an interpretation of each target country's MFC-R2 tones meaning.

The tones specified in the previous table are exchanged between the two ends of the link using a *compelling* procedure. During the procedure, the address information associated with the calling and called party is transferred from the outbound register to the inbound register.

For example, the outbound protocol will seize the line as the first step in placing a call. After receiving a seizure acknowledgment, the outbound protocol will play the first DID digit (using the multi-frequency codes specified in the Digits table at the beginning of this section). This digit will be acknowledged by means of an A-1 tone (the convention "A-1" signifies that the digit 1 (1140 Hz + 1020 Hz ) is played in the backward direction while in the Group A context). The A-1 tone also indicates that the outbound protocol should send the next DID digit. This process continues until the outbound protocol indicates (using an appropriate forward tone) that there are no more digits to send. Upon receipt of the backward signals A-3 or A-5, the outbound protocol switches its forward tones to Group II meaning. Note that the meaning of the tones belonging to the various groups varies by country.

Figure 1 illustrates the compelling sequence and related timing:

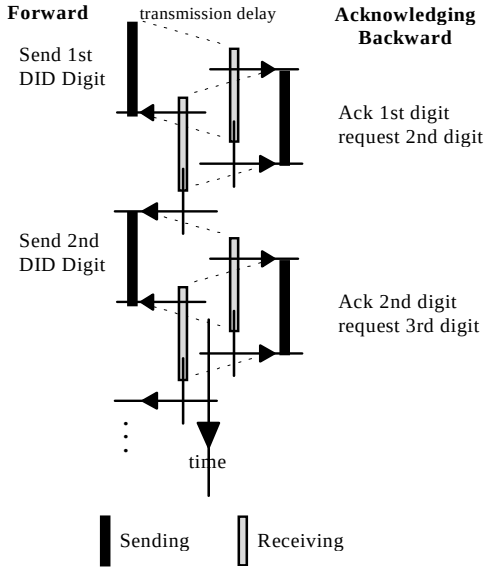



*Figure 1. MFC-R2 Compelling Cycles*

Figure 1 illustrates the first two cycles of a compelled sequence. The outbound register starts dialing a digit, by means of a forward tone. This is detected by the inbound register, which starts a backward tone. This tone is both an acknowledgment that the digit has been received and a request for a second digit. The inbound side can choose what kind of digit it is requesting (DID, ANI or Category) by transmitting different backward tones to the outbound side.

The timing of the cycle is as follows:

- As soon as the inbound side detects the forward tone, it starts playing its backward tone.
- As soon as the outbound side detects the backward tone, it stops playing its forward tone.
- As soon as the inbound side detects silence, it stops playing the backward tone it was playing.

- As soon as the outbound side detects silence, it starts playing a new digit, and another cycle begins.

There is no fixed timing. The whole cycle proceeds at the maximum speed allowed by the tone detection and generation equipment.

However, there is a timing constraint on how slow the cycles can be. The CCITT specifications define three timers in the compelled sequence:

| Timer | Description |
|---|---|
| T1 | The maximum time a forward tone can be on, from the outbound perspective. |
| T2 | The maximum time a forward tone can be off, from the outbound perspective. |
| T3 | The maximum time a whole compelled cycle can take, from the inbound perspective. |

If any of these timers expire, the compelled sequence is interrupted. In certain cases, the compelled sequence can resume, in others, the call setup fails.

## 1.4 TCP Parameters

This section describes the parameters that program the TCP. The parameters are used to adapt the TCP to the national variations of MFC-R2 used in the different countries of operation, as well as to give the host application more flexibility in its interaction with the TCP. The parameters are of different types. Some of them are application-dependent, and can be changed at will by the user. Others (the majority) are rigidly fixed for a given country and/or a given network. The following table explains the parameters' meaning.

### Flow Control Parameters

The following four parameters are public; they can be freely modified by the user:

| Parameter name (category) | Unit | Comments | Default | # |
|---|---|---|---|---|
| ADI.MFC.DIDnumber | number | *Inbound side:* number of DID digits to expect. | 7 | 0 |
| ADI.MFC.ANInumber | number | *Inbound side:* number of ANI digits to expect. | 8 | 1 |
| ADI.MFC.DIDBeforeANI | number | *Inbound side:* DID number to receive before asking for category. | 2 | 2 |

| Parameter name (category) | Unit | Comments | Default | # |
|---|---|---|---|---|
| ADI.MFC.optionflags | mask | Flags controlling TCP behavior (inbound side):<br>• bit 0 (& 0x1): Does the TCP play busy when rejecting a call? If not, the switch does it instead.<br>1=do not play, 0=play (default)<br>• bit 1 (& 0x2): Set this bit if the trunk you are using is *inbound* only (no calls can be placed on it).<br>• bit 2 (& 0x4): Set this bit if the trunk you are using is *outbound* only (no calls can be received on it).<br>• bit 3 (&0x8): Set this bit if a cleardown tone (busy tone) is necessary when the inbound side hangs up a call first.<br>• bit 4(& 0x10): Set this bit if you need to play a message while rejecting a call. If the bit is set, the TCP will send to the network the backward Group B tone to accept the call, thus causing the network to establish the voice path. The TCP will then wait for the far end to hang up.<br>• bit 5 (& 0x20): If this bit is set, the TCP will answer an incoming call from Group A, without asking the network for the caller's category (Group II forward tone). The caller's category will therefore be missing from the incoming call-related information delivered to the host. | 0x0 | 3 |
| ADI.MFC.debugmask | mask | Trace level specification, as follows:<br>• Show states: 0x01<br>• Show tones: 0x02<br>• Show line code: 0x04<br>• Show tone ends: 0x08<br>• Show compel errors: 0x10<br>• Show tone validity errors: 0x20<br>These values can be ORed for cumulative effect. The debug printouts have effect if running AGTRACE 1000 or higher. | 0x0 | 4 |

## Compelled Timers

The following parameters are *reserved*. Do not modify these parameters, as this would affect regulatory compliance in the target country.

| Parameter name (category) | Unit | Comments | Default | # |
|---|---|---|---|---|
| ADI.MFC.compelledtimers | internal | All compelled timers, both for the inbound side and the outbound side, expressed in seconds and packed in a 16-bits word. The timers are the following:<br>1. T1 (forward tones max on-time): lowest 5 bits.<br>Example 01111 = 15 s<br>2. T2 (forward tones max off-time): middle 5 bits.<br>Example 11011 = 27 s<br>3. T3 (inbound side compelled cycle timer): highest 6 bits.<br>Example 001111 = 15 s | 0x3F6F | 5 |

## Flow Control Timers

The following parameters are *reserved*. Do not modify these parameters, as this would affect the regulatory compliance of the AG-E1 board in the target country.

| Parameter name (category) | Unit | Comments | Default | # |
|---|---|---|---|---|
| ADI.MFC.releaseguardtime | time (ms) | *Inbound side*: time to wait in release guard, if set (regulated time for the inbound protocol to remain in blocking state after mutual hang-up) | 0 | 6 |

| Parameter name (category) | Unit | Comments | Default | # |
|---|---|---|---|---|
| ADI.MFC.inboundqualtimers | internal | Group of three inbound line qualification timers (which determine how long the inbound protocol will wait to recognize a line signaling change), expressed in units of 10 ms as follows:<br>1. Qualification time during the idle state: lowest hexade, typical value 20 ms.<br>2. Qualification time during the compelled sequence: middle hexade, typical value 20 ms.<br>3. Qualification time during the connected state: middle hexade, typical value 20 ms. | 0x222 | 7 |
| ADI.MFC.outboundqualtimers | internal | Group of three outbound qualification timers (which determine how long the protocol will wait to recognize a line signaling change), expressed in units of 10 ms, as follows:<br>1. Qualification time during idle: lowest hexade, typical value 20 ms.<br>2. Qualification time during the compelled sequence: middle hexade, typical value 20 ms.<br>3. Qualification time during connected: middle hexade, typical value 20 ms. | 0x222 | 8 |
| ADI.MFC.seizureacktime | time (ms) | *Outbound side*: time to wait for seizure acknowledge after seizure, before clearing forward. The CCITT Blue Book value is 200 ms, but the TCP must account for internal transmission time. | 240 | 9 |

| Parameter name (category) | Unit | Comments | Default | # |
|---|---|---|---|---|
| ADI.MFC.answertime | internal | *Outbound side*: group of two timers used during the answering phase of call setup. Both timers are expressed in seconds.<br>1. Lower byte: maximum time for the protocol to wait between receipt of the Group B backward tone until the phone is answered (line signaling event). After this timer expires, the TCP clears the line.<br>2. Higher byte: maximum time for the protocol to wait between the generation of the Group II forward tone and the detection of the Group B backward tone. This is usually the same as timer T1 from the parameter ADI.MFC.compelledtimers, but it can be different in some countries. | 0x0F3C | 10 |
| ADI.MFC.inboundreleasetime | time (ms) | *Outbound side*: the time after which a clearback seen by the outbound side is treated as a remote hangup signal. If the signal lasts less than this value, it could be a billing pulse. (Only applicable during conversation.) | 0 | 11 |

## Compelled Tones

The following parameters specify all the tones needed by the protocols to implement the country-specific variation of R2. Each parameter holds more than one tone. Each tone uses 4 bits (one hexade) of the 16 bit word. Tones are listed from least to most significant inside each parameter.

The following parameters are *reserved*. Do not modify these parameters, as this would affect regulatory compliance in the target country.

| Parameter name (category) | Unit | Comments | Default | # |
|---|---|---|---|---|
| ADI.MFC.tonesgroupA | internal | Backward Group A tones. The TCP uses these tones to send requests to the calling party during the compelled sequence. Each hexade of the parameter contains one request. Low to high hexade:<br>1. Send next DID (A-1).<br>2. Send Group I category (A-5).<br>3. Send next ANI (A-5).<br>4. Send Group II tone (and switch to group B tone reception)(A-3). | 0x3551 | 12 |
| ADI.MFC.tonesgroupB | internal | Some backward Group B tones. The TCP uses these tones to send the final indication of the compelled sequence to the calling party. Each hexade of the parameter contains one Group B indication. Low to high hexade:<br>1. Indicate congestion (B-4). This is also applicable during Group A transmission.<br>2. Indicate unallocated number (B-5).<br>3. Indicate busy (B-3).<br>4. Indicate line out of order (B-8). | 0x8354 | 13 |

| Parameter name (category) | Unit | Comments | Default | # |
|---|---|---|---|---|
| ADI.MFC.tonesendofinfo | internal | Forward tones that indicate the end or the non-availability of certain types of information. The types of information are:<br>1. In some countries, a tone that signals the end of the DID digits does not exist. In this case, the first hexade will be 0.<br>2. Caller's category. In some countries, the category must be available to the caller, so the second hexade will be 0.<br>3. End of ANI - caller ID available.<br>4. End of ANI - called ID restricted. In most countries, there is no distinction for MFC-R2 between restricted and non-restricted caller ID. In this case, the fourth hexade is 0. | 0x0FCF | 14 |
| ADI.MFC.tonesanswer | internal | Backward tones indicating acceptance of the call. Each hexade of the parameter contains one distinct type of acceptance indication. Low to high hexade:<br>1. Call accepted in Group B - charge (B-6).<br>2. Call accepted in Group B - free call (B-7).<br>3. Call accepted in Group A (A-6).<br>4. Alternative tone for call accepted in Group B (not in CCITT specs, but necessary in some countries). | 0x0667 | 15 |

| **Parameter name (category)** | **Unit** | **Comments** | **Default** | **#** |
|---|---|---|---|---|
| ADI.MFC.tonesmiscellaneous | internal | Request or indication tones used in different contexts by the TCP . They are (low to high hexade):<br>1. The tone the outbound part of the TCP plays in Group II (toll category in some countries) (II-1, normal subscriber).<br>2. Default user category (Group I category) to be used if the application does not provide it (in some countries the outbound side must play it in all cases) (I-1, normal subscriber).<br>3. Tone meaning that after the user category no ANIs are available (I-12, or 0xC).<br>4. Backward Group B tone used in some countries to reject an incoming call while requesting that the voice path be open for a special announcement. | 0x0C11 | 16 |
| ADI.MFC.tonesrepeatrequest | internal | Backward Group A tones the inbound side plays to request a digit repetition from the outbound side. Consider the DID that the outbound side played last to be DID ***n***. Low to high hexade:<br>1. Repeat DID ***n***-1 (A-2)<br>2. Repeat DID ***n***-2 (A-7)<br>3. Repeat DID ***n***-3 (A-8)<br>4. Repeat all DIDs (restart dialing) (Not specified by the CCITT Blue Book, but used in many countries.) | 0x0872 | 17 |

| Parameter name (category) | Unit | Comments | Default | # |
|---|---|---|---|---|
| ADI.MFC.tonesgroupC | internal | Backward tones that the inbound side plays when it is collecting ANIs (the specifications of some countries identify a Group C in this case). Low to high hexades:<br>1. Request the outbound side to go back to sending DIDs, and send the next DID. This is typically the same as the normal send DID tone, hexade 1 of ADI.MFC.tonesgroupA, but it can be different in some countries.<br>2. Request the outbound side to go back to sending DIDs, and repeat the last DID transmitted (Not supported by the CCITT Blue Book).<br>In this parameter there are two more hexades, not related with the Group C tones. These are:<br>1. Tone that the inbound side sends to request that the outbound side specifies the nature of the circuit, either land-based or through a satellite link (A-13)<br>2. Tone that the inbound side sends to ask if a half-echo suppressor is needed (A-14) | 0xED01 | 18 |

## Ring / Busy Control

These parameters do not have a CCITT specification describing them, but rather they describe the target country's call progress tones. Their default values are set to a reasonable international standard.

The following parameters are *reserved*. Do not modify these parameters, as this would affect regulatory compliance in the target country.

| Parameter name (category) | Unit | Comments | Default | # |
|---|---|---|---|---|
| ADI.MFC.ringfreq1 | Hz | *Inbound side*, ring frequency #1. | 425 | 19 |
| ADI.MFC.ringfreq2 | Hz | *Inbound side*, ring frequency #2 (if the ringing tone is composed of 2 frequencies: otherwise, 0). | 0 | 20 |
| ADI.MFC.ringontime | time (ms) | *Inbound side*, time the ring tone is on in a ring cycle. | 1000 | 21 |
| ADI.MFC.ringofftime1 | time (ms) | *Inbound side*, time the ring tone is off in a ring cycle. | 4000 | 22 |
| ADI.MFC.ringofftime2 | time (ms) | *Inbound side*, ring off-time two (for UK-style ringing cycles, e.g., 400 on, 200 off, 400 on, 2000 off). | 0 | 23 |
| ADI.MFC.busyfreq1 | Hz | *Inbound side*, busy frequency #1 | 425 | 24 |
| ADI.MFC.busyfreq2 | Hz | *Inbound side*, busy frequency #2 (if the busy tone is composed of 2 frequencies: otherwise, 0). | 0 | 25 |
| ADI.MFC.busyontime | time (ms) | *Inbound side*, time the busy tone is on in a busy cycle. | 500 | 26 |
| ADI.MFC.busyofftime | time (ms) | *Inbound side*, time the busy tone is off in a busy cycle. | 500 | 27 |
| ADI.MFC.ptoneslevel | IDU | *Inbound side*, amplitude of call progress tones (ring and busy). | 350 | 28 |

## Miscellaneous

The following parameters are *reserved*. Do not modify these parameters, as this would affect regulatory compliance in the target country.

| Parameter name (category) | Unit | Comments | Default | # |
|---|---|---|---|---|
| ADI.MFC.compelledtoneslevel | IDU | R2 tones amplitude (forward and backward) | 330 | 29 |
| ADI.MFC.signalingflags | mask | Parameter containing flags describing signaling variations. These are:<br>• bits 0, 1 (& 0x3): Value of the C and D bits (usually, C=0, D=1)<br>• bit 2 (& 0x4): What to do if next DID does not come: pulse congestion (0) or pulse request for Group II tone (1). This addresses the problem of those protocol variations without a tone to signal the end of DID digits. In this case, the bit is set to 1, so the compelled sequence continues with the request to send the Group II tone and switch to reception of Group B tones.<br>• bit 3 (& 0x8): Ignore bit faults during ring (0), or detect and abort the call (1).<br>• bit 4 (& 0x10): Expect metering pulses on the A bit (1=yes, 0=no).<br>• bit 5 (& 0x20): Expect metering pulses on the B bit (1=yes, 0=no) (if no metering pulses, both these bits are set to zero). | 0x1021 | 30 |

| Parameter name (category) | Unit | Comments | Default | # |
|---|---|---|---|---|
| ADI.MFC.signalingflags (cont.) | mask | • bits 6 and 7 (& 0xC0): Controls the length of delay after the outbound protocol hangs up before the application is able to place a new call. This is needed by some switches to clear the line and be seized again. Values:<br>0x0 = no delay<br>0x1 = 400 ms delay<br>0x2 = 700 ms delay (default)<br>0x3 = 1000 ms delay<br>• bit 8 (& 0x100): Flag to determine if an inbound side clearing back is signaled by a release guard (AB=11), value 1, or by a forced release (AB=00), value 0.<br>• bit 9 (& 0x200): Flag to determine if the bit detector should be set with a simultaneity window of zero (if clear), or of 5 ms (if set). Having two bit transitions falling in the same 5 ms period is the definition of simultaneous bit transitions on the line. Double bit transitions are illegal in some countries.<br>• bit 10 (& 0x400): Expect metering pulses on the C bit (1=yes, 0=no).<br>• bit 11 (& 0x800): If set, a half-echo suppression is needed in international implementation (the outbound side answers with a I-14 to a A-14 request, in CCITT speech). | 0x1021 | 30 |

| Parameter name (category) | Unit | Comments | Default | # |
|---|---|---|---|---|
| ADI.MFC.validDIDmask | internal | The valid DID tones in the target country. If a DID received by the inbound side is not valid, the compelled sequence is aborted with a congestion indication.<br>The mask is the following:<br>`FEDC\|BA98\|7654\|321-`<br>`1000\|0111\|1111\|1110` | 0x87FE | 31 |
| ADI.MFC.validANImask | mask | The valid ANI tones in the target country. If an ANI received by the inbound side is not valid, the compelled sequence is aborted with a congestion indication.<br>The mask is the following:<br>`FEDC\|BA98\|7654\|321-`<br>`1001\|0111\|1111\|1110` | 0x97FE | 32 |
| ADI.MFC.validcategorymask | mask | The valid category tones in the target country. If an invalid category is received by the inbound side, the compelled sequence is aborted with a congestion indication.<br>The mask is the following:<br>`FEDC\|BA98\|7654\|321-`<br>`0000\|0111\|1111\|1110` | 0x07FE | 33 |
| ADI.MFC.validGroupIImask | mask | The valid Group II tones in the country. If an invalid Group II tone is received by the inbound side, the compelled sequence is aborted with a congestion indication.<br>The mask is the following:<br>`FEDC\|BA98\|7654\|321-`<br>`0000\|0111\|1111\|1110` | 0x07FE | 34 |
| ADI.MFC.catnoANImask | mask | The category tones that imply that no ANI follows. The mask is the following:<br>`FEDC\|BA98\|7654\|321-`<br>`0000\|0000\|0000\|0000` | 0x0 | 35 |

| Parameter name (category) | Unit | Comments | Default | # |
|---|---|---|---|---|
| ADI.MFC.tonesinternational | internal | Forward Group A tones that the outbound side plays to answer the inbound side's request for information about the presence or absence of a satellite link in the circuit, and the need for half-echo suppression. Low to high hexade:<br>• No satellite link in the circuit (I-13)<br>• Satellite link in the circuit (I-14), or half-echo needed for other circumstances<br>*Note:* If half-echo suppression is *not* needed, the outbound side ignores the request and plays any other tone. | 0xEED | 36 |
| ADI.MFC.reanswerdelay | time (ms) | Time to wait for re-answer pulse. This might be needed for special switches in certain countries. Set to zero to disable. | 0 | 37 |
| ADI.MFC.reanswerpulsetime | time (ms) | Length of re-answer pulse. Disabled if ADI.MFC.reanswerdelay is set to zero. | 0 | 38 |
| ADI.MFC.resourcegettimes | internal | This parameter addresses a special need of protocols running on an AG Quad board with resource management enabled. In this case, for certain operations it is necessary to acquire a resource from a resource pool on the board. The parameter defines two timeouts after which the operation is aborted if a resource is not available. (This is a very unlikely occurrence.)<br>• Low byte: time to wait for resource before placing a call (15 s).<br>• High byte: time to wait for resource when a resource is needed by the inbound side to release a call (e.g., to play a cleardown tone) (10 s). | 0x0a0f | 39 |

## 1.5 Country-Specific Implementation

Variations of the MFC-R2 protocol used in different countries are captured by the TCP parameters, as previously discussed. This section details the values of the parameters most important for implementing the various national specifications.

The parameters that tend to differ the most from country to country generally describe tones that request or transmit specific information in the protocol:

| Parameter Type | Description |
|---|---|
| Request | The backward tones used in register signaling. Both Group A tones and Group B tones change in their meaning for different countries and networks. |
| Transmit | The forward tones used in register signaling. Most of them represent address information, however, there are forward tones to signal the absence or end of information, and these tones differ among networks. |

*Note:* In some cases, parameters other than tones are unique to particular countries.

Each supported country/network is treated separately in this section. Deviations from the standard CCITT specification are highlighted, and the relevant parameter values are discussed.

### 1.5.1 Argentina

The Argentine MFC-R2 specifications use the same tones as the CCITT standard specifications for register signaling. However, they introduce meanings for some of the backward tones that CCITT leaves to the country-specific specification.

The following table gives the meaning of the backward tones. The tones with a special national meaning in Argentina are in **bold**.

| **Backward Group A** | | **Backward Group B** | |
|---|---|---|---|
| Designation of the Signal | Meaning of the Signal | Designation of the Signal | Meaning of the Signal |
| A-1 | Send next digit (N+1) | B-1 | **Not Used** |
| A-2 | Send last digit (N-1) | B-2 | Send special information tone, **and setup speech conditions** |
| A-3 | Address completed, change to reception of Group B | B-3 | Subscriber line busy |
| A-4 | Congestion in the national network | B-4 | Congestion |
| A-5 | Send calling party's category | B-5 | Unallocated number |
| A-6 | Address complete, charge, setup speech conditions | B-6 | Subscriber line free, charge |
| A-7 | Send second-to-last digit (N-2) | B-7 | Subscriber line free, no charge |
| A-8 | Send third-from-last digit (N-3) | B-8 | Subscriber line out of order |
| A-9 | **Send the last digit again** | B-9 | **Not Used** |
| A-10 | **Repeat the DID digits from the beginning** | B-10 | **Not Used** |
| A-11 | | B-11 | **Not Used** |
| A-12 | | B-12 | **Not Used** |
| A-13 | Send nature of circuit | B-13 | **Not Used** |
| A-14 | Request info on use of half-echo suppression | B-14 | **Not Used** |
| A-15 | | B-15 | **Not Used** |

The following is a list of Argentine forward absence or end of information tones:

| Meaning | Tone | Comments |
|---|---|---|
| End of DID digits | none | Expiration of timer T3 and pulsing of A-3 from the inbound side is used |
| End of ANI digits | I-15, I-12 | I-15: end of ANI, presentation allowed<br>I-12: end of ANI, presentation restricted |
| Category not available | none | Compulsory category digit - the outbound side must provide the information |
| ANI digits not available | I-12 | If no ANI received |

Other variations:

- The forward I-15 tone signals the end of ANI digits, but not the end of DID digits. If the inbound register expects a variable number of inbound digits, and it requests one DID more than the outbound register has to offer, the outbound side remains silent until the inbound side times out. When this happens, the inbound side pulses an A-3 tone to resume the compelled sequence and switch to the Group II / Group B compelled cycle.
- Metering pulses, if provided by the switch, come on the A-bit. Since in this case the line code that the outbound equipment sees during a metering pulse is the same as the line code that signals a clear back condition, this line code must subsist for at least 400 ms to be qualified as a clear back, as set by the parameter ADI.MFC.inboundreleasetime.

### 1.5.2 Bahrain

Same as CCITT specifications.

### 1.5.3 Bolivia

The Bolivian MFC-R2 specification differs from the CCITT standard specifications for register signaling only in the definitions of a few backward tones.

The following table gives the meaning of the backward tones. The tones that differ from the CCITT specification are in **bold**.

| **Backward Group A** | | **Backward Group B** | |
|---|---|---|---|
| Designation of the Signal | Meaning of the Signal | Designation of the Signal | Meaning of the Signal |
| A-1 | Send next digit (N+1) | B-1 | Not Used |
| A-2 | Send last digit (N-1) | B-2 | Announcement follows |
| A-3 | Address completed, change to reception of Group B | B-3 | Subscriber line busy |
| A-4 | Congestion in the national network | B-4 | Congestion |
| A-5 | Send calling party's category | B-5 | Unallocated number |
| A-6 | Answer from Group A | B-6 | Subscriber line free, charge |
| A-7 | Send second-to-last digit (N-2) | B-7 | Subscriber line free, no charge |
| A-8 | Send third-from-last digit (N-3) | B-8 | **Not Used** |
| A-9 | **Repeat the DID digits from the beginning** | B-9 | **Not Used** |
| A-10 | **Not Used** | B-10 | **Not Used** |
| A-11 | | B-11 | **Not Used** |
| A-12 | | B-12 | **Not Used** |
| A-13 | Send nature of circuit | B-13 | **Not Used** |
| A-14 | Request information on use of half-echo suppression | B-14 | **Not Used** |
| A-15 | | B-15 | **Not Used** |

The following is a list of Bolivian forward absence or end of information tones:

| **Meaning** | **Tone** |
|---|---|
| End of DID digits | I-15 |

| Meaning | Tone |
|---|---|
| End of ANI digits | I-15 |
| Category not available | I-12 |
| ANI digits not available | I-12 |

### 1.5.4 Brazil

The Brazilian MFC-R2 specification differs from the CCITT standard specifications for register signaling only in the definition of some meanings for backward tones.

The following table gives the meaning of the backward tones. The tones that differ from the CCITT specification are in **bold**.

| **Backward Group A** | | **Backward Group B** | |
|---|---|---|---|
| Designation of the Signal | Meaning of the Signal | Designation of the Signal | Meaning of the Signal |
| A-1 | Send next digit (N+1) | B-1 | **Subscriber line free, charge** |
| A-2 | **Repeat the DID digits from the beginning** | B-2 | **Subscriber line busy** |
| A-3 | Address completed, change to reception of Group B | B-3 | Not Used |
| A-4 | Congestion in the national network | B-4 | Congestion |
| A-5 | Send calling party's category | B-5 | **Subscriber line free, no charge** |
| A-6 | **Not Used** | B-6 | **Not Used** |
| A-7 | Send second-to-last digit (N-2) | B-7 | **Unallocated number** |
| A-8 | Send third-from-last digit (N-3) | B-8 | Not Used |
| A-9 | **Send last digit (N-1)** | B-9 | Not Used |
| A-10 | Not Used | B-10 | Not Used |
| A-11 | | B-11 | Not Used |

| **Backward Group A** | | **Backward Group B** | |
|---|---|---|---|
| Designation of the Signal | Meaning of the Signal | Designation of the Signal | Meaning of the Signal |
| A-12 | | B-12 | Not Used |
| A-13 | Send nature of circuit | B-13 | Not Used |
| A-14 | Request info on use of half-echo suppression | B-14 | Not Used |
| A-15 | | B-15 | **Not Used** |

The following is a list of Brazilian forward absence or end of information tones:

| **Meaning** | **Tone** | **Comments** |
|---|---|---|
| End of DID digits | none | Expiration of timer T3 and pulsing of A-3 from the inbound side is used |
| End of ANI digits | I-15 | |
| Category not available | I-12 | |
| ANI digits not available | I-12 | |

Other variations:

- The forward I-15 tone signals the end of ANI digits, but not the end of DID digits. If the inbound register expects a variable number of inbound digits, and it requests one DID more than the outbound register has to offer, the outbound side remains silent until the inbound side times out its request. When this happens, the inbound side pulses an A-3 tone to resume the compelled sequence and switch to the Group II / Group B compelled cycle.
- Some switches have a provision for a “reanswer” signal with which an inbound terminal equipment can automatically reject all collect calls. To do this, the equipment pulses a *clear back* line code, immediately after signaling the *answer* line code to connect a call. If this pulse is sent to the switch, and the call is a collect call, the switch clears the call back and then clears the line forward.

The relevant parameters are:

| Parameter | Description |
|---|---|
| ADI.MFC.reanswerdelay | The time between when the answer signal is put on the line, and when the clear back pulse starts. |
| ADI.MFC.reanswerpulsetime | The duration of the clear back pulse. |

### 1.5.5 Chile

The Chilean MFC-R2 specification is very similar to the CCITT standard. All backward and forward tones have the same meaning.

Only one difference exists: billing pulses are conveyed to a terminal placing a call by means of bit variations on the C-bit.

### 1.5.6 China

The Chinese MFC-R2 specifications are rather different from the CCITT standard specifications, especially for register signaling.

The following table reports the meaning of the backward tones. The tones that differ from the CCITT specification are in **bold**.

| **Backward Group A** | | **Backward Group B** | |
|---|---|---|---|
| Designation of the Signal | Meaning of ths Signal | Designation of the Signal | Meaning of the Signal |
| A-1 | Send next digit (N+1) - **no matter if DID or ANI** | B-1 | **Subscriber line free, charge** |
| A-2 | **Repeat the DID digits from the beginning** | B-2 | **Subscriber line busy** |
| A-3 | Address completed, change to reception of Group B | B-3 | **Not Used** |
| A-4 | Congestion in the national network | B-4 | Congestion |
| A-5 | **Not Used** | B-5 | Unallocated number |

| **Backward Group A** | | **Backward Group B** | |
|---|---|---|---|
| Designation of the Signal | Meaning of ths Signal | Designation of the Signal | Meaning of the Signal |
| A-6 | **Send calling party's category** | B-6 | Subscriber line free, no charge |
| A-7 | **Not Used** | B-7 | **Not Used** |
| A-8 | **Not Used** | B-8 | Subscriber line out of order |
| A-9 | **Not Used** | B-9 | **Not Used** |
| A-10 | **Not Used** | B-10 | **Not Used** |
| A-11 | | B-11 | **Not Used** |
| A-12 | | B-12 | **Not Used** |
| A-13 | Send nature of circuit | B-13 | **Not Used** |
| A-14 | Request information on use of half-echo suppression | B-14 | **Not Used** |
| A-15 | | B-15 | **Not Used** |

The following is a list of Chinese forward absence or end of information tones:

| **Meaning** | **Tone** | **Comments** |
|---|---|---|
| End of DID digits | none | Expiration of timer T3 and pulsing of A-3 from the inbound side is used |
| End of ANI digits | I-15 | Necessary to decode the end of ANI |
| Category not available | I-15 | |
| ANI digits not available | I-15 | If no ANI are received |

Other variations:

- The forward I-15 tone signals the end of ANI digits, but not the end of DID digits. If the inbound register expects a variable number of inbound digits, and it requests one DID more than the outbound register has to offer, the outbound side remains silent until the inbound side times out. When this

happens, the inbound side pulses an A-3 tone to resume the compelled sequence and switch to the Group II / Group B compelled cycle.

- The fact that the backward tone to request ANI digits is the same as the one to request DID has wide implications. The inbound register is no longer in control of the type of digits it receives, so to distinguish between ANI and DID it must wait for an I-15 (end of ANI) from the outbound register. In turn, this means that the parameter ADI.MFC.ANInumber (which is used by the inbound side to determine when to switch from requesting ANI back to requesting DID) has no meaning. In the Chinese variation of the MFC-R2 package this parameter is set to 30, a number higher than any possible caller ID.
- In the CCITT line signaling specifications, the C and D bits are set to 0 and 1 respectively, in China they are both set to 1. However, they are never used, so their value never changes during a call.

## 1.5.7 Colombia

Natural MicroSystems supports two network-specific variations in Colombia.

### Telecom Colombia

Telecom Colombia has an MFC-R2 variation similar to the Chinese variation. However, it is closer to the CCITT specifications.

The following table reports the meaning of the backward tones. The tones that differ from the CCITT specification are in **bold**.

| **Backward Group A** | | **Backward Group B** | |
|---|---|---|---|
| Designation of the Signal | Meaning of the Signal | Designation of the Signal | Meaning of the Signal |
| A-1 | Send next digit (N+1) - **no matter if DID or ANI** | B-1 | **Subscriber line free, charge** |
| A-2 | Send last digit (N-1) | B-2 | **Subscriber line busy** |
| A-3 | Address completed, change to reception of Group B | B-3 | **Not Used** |
| A-4 | Congestion in the national network | B-4 | Congestion |

| **Backward Group A** | | **Backward Group B** | |
|---|---|---|---|
| Designation of the Signal | Meaning of the Signal | Designation of the Signal | Meaning of the Signal |
| A-5 | **Not Used** | B-5 | **Subscriber line free, no charge** |
| A-6 | **Send calling party's category** | B-6 | **Unallocated number** |
| A-7 | Send second-to-last digit (N-2) | B-7 | **Not Used** |
| A-8 | Send third-from-last digit (N-3) | B-8 | **Not Used** |
| A-9 | **Not Used** | B-9 | **Not Used** |
| A-10 | **Not Used** | B-10 | **Not Used** |
| A-11 | | B-11 | **Not Used** |
| A-12 | | B-12 | **Not Used** |
| A-13 | Send nature of circuit | B-13 | **Not Used** |
| A-14 | Request information on use of half-echo suppression | B-14 | **Not Used** |
| A-15 | | B-15 | **Not Used** |

The following table lists forward absence or end of information tones for Telecom Colombia:

| **Meaning** | **Tone** | **Comments** |
|---|---|---|
| End of DID digits | I-15 | |
| End of ANI digits | I-15 | Necessary to decode the end of ANI |
| Category not available | I-12 | |
| ANI digits not available | I-12 | If no ANI received |

Other variations:

- The fact that the backward tone to request ANI digits is the same as the one to request DID has wide implications. The inbound register is no longer in control of the type of digits it receives. To distinguish between ANI and

DID, it must wait for an I-15 (end of ANI) from the outbound register. In turn, this means that the parameter ADI.MFC.ANInumber (which is used by the inbound side to determine when to switch from requesting ANI back to requesting DID) has no meaning.

- Metering pulses, if provided by the switch, come on the A-bit. Since in this case the line code that the outbound equipment receives during a metering pulse is the same as the line code that signals a clear back condition, this line code must subsist for at least 800 ms to be qualified as a clear back, as set by the parameter ADI.MFC.inboundreleasetime.

## ComSel (Colombian Cellular provider)

ComSel provides a MFC-R2 variation very similar to the Chinese MFC-R2.

The following table reports the meaning of the backward tones. The tones that differ from the CCITT specification are in **bold**.

| **Backward Group A** | | **Backward Group B** | |
|---|---|---|---|
| Designation of the Signal | Meaning of the Signal | Designation of the Signal | Meaning of the Signal |
| A-1 | Send next digit (N+1) **- no matter if DID or ANI** | B-1 | **Subscriber line free, charge** |
| A-2 | **Repeat the DID digits from the beginning** | B-2 | **Call rejected, no indication of cause** |
| A-3 | Address completed, change to reception of Group B | B-3 | **Not Used** |
| A-4 | Congestion in the national network | B-4 | Congestion |
| A-5 | **Not Used** | B-5 | **Subscriber line free, no charge** |
| A-6 | **Send calling party's category** | B-6 | **Not Used** |
| A-7 | **Not Used** | B-7 | **Not Used** |
| A-8 | **Not Used** | B-8 | **Not Used** |
| A-9 | **Not Used** | B-9 | **Not Used** |
| A-10 | **Not Used** | B-10 | **Not Used** |
| A-11 | | B-11 | **Not Used** |

| **Backward Group A** | | **Backward Group B** | |
|---|---|---|---|
| Designation of the Signal | Meaning of the Signal | Designation of the Signal | Meaning of the Signal |
| A-12 | | B-12 | **Not Used** |
| A-13 | Send nature of circuit | B-13 | **Not Used** |
| A-14 | Request information on use of half-echo suppression | B-14 | **Not Used** |
| A-15 | | B-15 | **Not Used** |

The following is a list of forward absence or end of information tones for ComSel:

| **Meaning** | **Tone** | **Comments** |
|---|---|---|
| End of DID digits | none | Expiration of timer T3 and pulsing of A-3 from the inbound side is used |
| End of ANI digits | I-15 | Necessary to decode the end of ANI |
| Category not available | I-15 | |
| ANI digits not available | I-15 | If no ANI received |

Other variations:

- The forward I-15 tone signals the end of ANI digits, but not the end of DID digits. If the inbound register expects a variable number of inbound digits, and it requests one DID more than the outbound register has to offer, the outbound side remains silent until the inbound side times out. When this happens, the inbound side pulses an A-3 tone, to resume the compelled sequence and switch to the Group II / Group B compelled cycle.
- The fact that the backward tone to request ANI digits is the same as the one to request DID has wide implications. The inbound register is no longer in control of the type of digits it receives. To distinguish between ANI and DID it must wait for an I-15 (end of ANI) from the outbound register. In turn, this means that the parameter ADI.MFC.ANInumber (which is used by the inbound side to determine when to switch from requesting ANI back to requesting DID) has no meaning.
- Metering pulses, if provided by the switch, come on the B-bit.

## 1.5.8 Czech Republic

The specifications for the MFC-R2 variation in the Czech Republic are similar to the CCITT specifications.

The following table gives the meaning of the backward tones. The tones that differ from the CCITT specification are in **bold**.

| **Backward Group A** | | **Backward Group B** | |
|---|---|---|---|
| Designation of the Signal | Meaning of the Signal | Designation of the Signal | Meaning of the Signal |
| A-1 | Send next digit (N+1) | B-1 | **alternative answer tone** |
| A-2 | Send last digit (N-1) | B-2 | **Not Used** |
| A-3 | Address completed, change to reception of Group B | B-3 | Subscriber line busy |
| A-4 | Congestion in the national network | B-4 | Congestion |
| A-5 | Send calling party's category | B-5 | Unallocated number |
| A-6 | Address complete, charge, setup speech conditions | B-6 | Subscriber line free, charge |
| A-7 | Send second-to-last digit (N-2) | B-7 | Subscriber line free, no charge |
| A-8 | Send third-from-last digit (N-3) | B-8 | Subscriber line out of order |
| A-9 | **Not Used** | B-9 | **Not Used** |
| A-10 | **Not Used** | B-10 | **Not Used** |
| A-11 | | B-11 | **Not Used** |
| A-12 | | B-12 | **Not Used** |
| A-13 | Send nature of circuit | B-13 | **Not Used** |
| A-14 | Request information on use of half-echo suppression | B-14 | **Not Used** |
| A-15 | | B-15 | **Not Used** |

The following is a list of Czech forward absence or end of information tones:

| Meaning | Tone | Comments |
|---|---|---|
| End of DID digits | I-15 | |
| End of ANI digits | I-15 | |
| Category not available | none | Compulsory category digit - the outbound side must provide the information. |
| ANI digits not available | I-12 | If no ANI received. |

Other variations:

- The compelled timer T3 (inbound compelled whole-cycle timer) is set to 5 seconds (the CCITT value is 15 sec.). This means that in the case of a slow compelled sequence, the inbound equipment will time out before the outbound one, initiating a clearing of the connection.
- When an outgoing call from a CPE reaches conversation state, if the inbound switch clears back first, the line signal used is a *forced release* rather than a *release guard*. The CPE will see the B-bit going to 0, instead of the A-bit going to 1. However, the protocol is not symmetrical in this point: an inbound CPE releases the call normally by setting the A-bit to 1. This behavior is programmed in the TCP by the parameter ADI.MFC.signalingflags & 0x100 being 0.
- Metering pulses, if provided by the switch, come on the B-bit.

### 1.5.9 Honduras

Same as CCITT specifications.

### 1.5.10 India

MFC-R2 specifications in India define three line signaling standards: Digital Types 1, 2 and 3. Of these, only Digital Type 1 follows the CCITT Recommendation Q.421, and is thus supported by NMS standard MFC-R2 product.

Indian specifications define only 10 possible tones of the 15 standard CCITT forward and backward tones. CCITT Recommendation Q.441 describes this

arrangement. It is cheaper in terms of MF tone detection equipment, because the first 10 backward and forward tones are generated using 2 frequencies out of a pool of 5, instead of 6. This eliminates the need for one backward and one forward frequency.

This model places some restrictions on the possible kinds of register signaling. These are described later in this section.

The following table reports the meaning of the backward tones. The tones that differ from the CCITT specification are in **bold**.

| **Backward Group A** | | **Backward Group B** | |
|---|---|---|---|
| Designation of the Signal | Meaning of the Signal | Designation of the Signal | Meaning of the Signal |
| A-1 | Send next digit (N+1) | B-1 | **Not Used** |
| A-2 | **Repeat the DID digits from the beginning** | B-2 | **Changed number (announcement on the line)** |
| A-3 | Address completed, change to reception of Group B | B-3 | Called line busy |
| A-4 | **Send next ANI digit** | B-4 | Congestion |
| A-5 | Send calling party's category | B-5 | Unallocated number |
| A-6 | Call accepted (from Group A) – switch through the voice path | B-6 | Call accepted - charge |
| A-7 | Send second-to-last digit (N-2) | B-7 | **Unallocated number** |
| A-8 | **Send third-from-last digit (N-3)** | B-8 | **Not Used** |
| A-9 | **Send last digit (N-1)** | B-9 | **Not Used** |
| A-10 | **Not Used** | B-10 | **Not Used** |
| A-11 | **not defined** | B-11 | **not defined** |
| A-12 | **not defined** | B-12 | **not defined** |
| A-13 | **not defined** | B-13 | **not defined** |
| A-14 | **not defined** | B-14 | **not defined** |
| A-15 | **not defined** | B-15 | **not defined** |

*Note:* This is the only case we know of an A-4 tone with a different meaning from a B-4 tone. Also, a Group A congestion signal is not defined.

The fact that only 10 tones are defined in each direction puts some constraints in how the outbound side communicates end of digits or no availability of digit types to the inbound side:

- DID: the terminating (inbound side) exchange is supposed to use its internal knowledge of the numbering plan, to determine the number of expected DID from the initial digits. However, the timeout – pulsed A-3 method can be used.
- ANI: a fixed number of ANI (10) is transmitted. If less than 10 digits are available, the outbound exchange fills the remaining places in the digit string with ‘0’s (I-10). Also, an additional field is defined at the beginning of the ANI digits. As a response to the first A-4, the outbound exchange sends a ‘0’ (I-10) if ANI digits are not available, or a ‘1’ (I-1) if ANI digits are available.

Sending the category digit as a response to an A-5 is obligatory. Categories are defined as follows:

**Forward Group II**

| Designation of the Signal | Meaning of the Signal |
|---|---|
| II-1 | Ordinary subscriber |
| II-2 | Priority subscriber |
| II-3 | spare |
| II-4 | spare |
| II-5 | Operator |
| II-6 | Pay phone |
| II-7 | spare |
| II-8 | spare |
| II-9 | spare |
| II-10 | spare |

## 1.5.11 Indonesia

The Indonesian MFC-R2 specification defines some different meanings for backward tones that appear in the CCITT standard specifications for register signaling.

The following table gives the meaning of the backward tones. The tones that differ from the CCITT specification are in **bold**.

| **Backward Group A** | | **Backward Group B** | |
|---|---|---|---|
| Designation of the signal | Meaning of the signal | Designation of the signal | Meaning of the signal |
| A-1 | Send next digit (N+1) | B-1 | **Subscriber line free, charge** |
| A-2 | **Repeat the DID digits from the beginning** | B-2 | **Subscriber line busy** |
| A-3 | Address completed, change to reception of Group B | B-3 | **Not Used** |
| A-4 | Congestion in the national network | B-4 | Congestion |
| A-5 | **Address complete, charge, setup speech conditions** | B-5 | **Subscriber line free, no charge** |
| A-6 | **Send calling party's category - and ANI digits** | B-6 | **Not Used** |
| A-7 | **Not Used** | B-7 | **Unallocated number** |
| A-8 | **Send last digit (N-1)** | B-8 | **Not Used** |
| A-9 | **Send second-to-last digit (N-2)** | B-9 | **Not Used** |
| A-10 | **Not Used** | B-10 | **Not Used** |
| A-11 | | B-11 | **Not Used** |
| A-12 | | B-12 | **Not Used** |
| A-13 | Send nature of circuit | B-13 | **Not Used** |
| A-14 | Request info on use of half-echo suppression | B-14 | **Not Used** |
| A-15 | | B-15 | **Not Used** |

The following is a list of Indonesian forward absence or end of information tones:

| **Meaning** | **Tone** |
|---|---|
| End of DID digits | I-15 |
| End of ANI digits | I-15 |
| Category not available | I-15 |
| ANI digits not available | I-15 |

There are no other variations.

### 1.5.12 Korea

The Korean MFC-R2 specification is similar to CCITT. The only difference is that the backward tone A-9, that CCITT leaves free for the national specifications to use, means “Repeat the DID digits from the beginning”.

### 1.5.13 Malaysia

The Malaysian MFC-R2 specifications are very similar to the Indonesian specifications (see Section 1.5.11), with several variations from the CCITT specifications in the backward register signaling tones.

The following table reports the meaning of the backward tones. The tones that differ from the CCITT specification are in **bold**.

| **Backward Group A** | | **Backward Group B** | |
|---|---|---|---|
| Designation of the Signal | Meaning of the Signal | Designation of the Signal | Meaning of the Signal |
| A-1 | Send next digit (N+1) | B-1 | **Subscriber line free, charge** |
| A-2 | **Repeat the DID digits from the beginning** | B-2 | **Subscriber line busy** |
| A-3 | Address completed, change to reception of Group B | B-3 | **Unallocated number** |
| A-4 | Congestion in the national network | B-4 | Congestion |

| **Backward Group A** | | **Backward Group B** | |
|---|---|---|---|
| Designation of the Signal | Meaning of the Signal | Designation of the Signal | Meaning of the Signal |
| A-5 | **Not Used** | B-5 | **Subscriber line free, no charge** |
| A-6 | **Send calling party's category - and ANI digits** | B-6 | **Not Used** |
| A-7 | **Not Used** | B-7 | **Not Used** |
| A-8 | **Send last digit (N-1)** | B-8 | **Not Used** |
| A-9 | **Send second-to-last digit (N-2)** | B-9 | **Not Used** |
| A-10 | **Not Used** | B-10 | **Not Used** |
| A-11 | | B-11 | **Not Used** |
| A-12 | | B-12 | **Not Used** |
| A-13 | Send nature of circuit | B-13 | **Not Used** |
| A-14 | Request information on use of half-echo suppression | B-14 | **Not Used** |
| A-15 | | B-15 | **Not Used** |

The following is a list of Malaysian forward absence or end of information tones:

| **Meaning** | **Tone** | **Comments** |
|---|---|---|
| End of DID digits | I-15 | |
| End of ANI digits | I-15 | |
| Category not available | none | Compulsory category digit - the outbound side must provide the information |
| ANI digits not available | I-12 | If no ANI received |

There are no other variations.

## 1.5.14 Mexico

Mexican MFC-R2 specifications are considerably different from the CCITT standard. Not only do most backward tones have different meanings, but a backward Group C is defined, together with a forward Group III. Group C tones are those that the inbound register sends while receiving the Category and ANI digits. Forward Group III tones are the Category and ANI digits themselves.

The following table gives the meaning of the backward tones. The tones that differ from the CCITT specification are in **bold**.

| **Backward Group A** | | **Backward Group B** | | **Backward Group C** | |
|---|---|---|---|---|---|
| Designation of the Signal | Meaning of the Signal | Designation of the Signal | Meaning of the Signal | Designation of the Signal | Meaning of the Signal |
| A-1 | Send next DID digit (N+1) | B-1 | **Subscriber line free, charge** | C-1 | **Send next ANI digit** |
| A-2 | **Repeat the DID digits from the beginning** | B-2 | **Subscriber line busy** | C-2 | **Same as A-2: request transition to Group A, restart from first DID** |
| A-3 | Address completed, change to reception of Group B | B-3 | Not Used | C-3 | **Same as A-3** |
| A-4 | Congestion in the national network | B-4 | Congestion | C-4 | Congestion |
| A-5 | Not Used | B-5 | **Subscriber line free, no charge** | C-5 | **Request transition back to Group A, and send next DID** |

| **Backward Group A** | | **Backward Group B** | | **Backward Group C** | |
|---|---|---|---|---|---|
| Designation of the Signal | Meaning of the Signal | Designation of the Signal | Meaning of the Signal | Designation of the Signal | Meaning of the Signal |
| A-6 | **Send calling party's category and change to reception of Group C tones** | B-6 | **Not Used** | C-6 | **Request transition back to Group A, and repeat last DID** |
| A-7 | **Not Used** | B-7 | **Not Used** | C-7 | **Not Used** |
| A-8 | **Not Used** | B-8 | **Not Used** | C-8 | **Not Used** |
| A-9 | **Not Used** | B-9 | **Not Used** | C-9 | **Not Used** |
| A-10 | **Not Used** | B-10 | **Not Used** | C-10 | **Not Used** |
| A-11 | **Not Used** | B-11 | **Not Used** | C-11 | **Not Used** |
| A-12 | **Not Used** | B-12 | **Not Used** | C-12 | **Not Used** |
| A-13 | **Not Used** | B-13 | **Not Used** | C-13 | **Not Used** |
| A-14 | **Not Used** | B-14 | **Not Used** | C-14 | **Not Used** |
| A-15 | **Not Used** | B-15 | **Not Used** | C-15 | **Not Used** |

The following is a list of Mexican forward absence or end of information tones:

| **Meaning** | **Tone** |
|---|---|
| End of DID digits | I-15 |
| End of ANI digits | I-15 |
| Category not available | I-15 |
| ANI digits not available | I-15 |

Other variations from CCITT specifications:

- The forward I-15 tone signals the end of ANI digits, but not the end of DID digits. If the inbound register expects a variable number of inbound digits, and it requests one DID more than the outbound register has to offer, the outbound side remains silent until the inbound side times out. When this happens, the inbound side pulses an A-3 tone to resume the compelled sequence and switch to the Group II / Group B compelled cycle.
- Even though the tone to request DID digits is the same as the one to request ANI digits, if the request C-5 can make the outbound register stop sending ANI digits and send DID digits, then the inbound register can still be in control of ANI / DID requesting.
- The time the inbound register has to accept or reject the call (by sending the Group B tone) is longer than the normal compelled sequence timer. In this case it is 90 seconds. This time is set in the high byte of the parameter ADI.MFC.answertime.

### 1.5.15 Panama

The Panamanian MFC-R2 specification is very similar to the CCITT. The only difference is that the backward tone A-9, that CCITT leaves free for the national specifications to use, means "Repeat the DID digits from the beginning".

### 1.5.16 Singapore

Singapore's MFC-R2 specifications are very similar to the Indonesian MFC-R2 specifications (see Section 1.5.11), with several variations from the CCITT specifications in the backward register signaling tones.

The following table gives the meaning of the backward tones. The tones that differ from the CCITT specification are in **bold**.

| **Backward Group A** | | **Backward Group B** | |
|---|---|---|---|
| Designation of the Signal | Meaning of the Signal | Designation of the Signal | Meaning of the Signal |
| A-1 | Send next digit (N+1) | B-1 | **Subscriber line free, charge** |

| **Backward Group A** | | **Backward Group B** | |
|---|---|---|---|
| Designation of the Signal | Meaning of the Signal | Designation of the Signal | Meaning of the Signal |
| A-2 | **Repeat the DID digits from the beginning** | B-2 | **Subscriber line busy** |
| A-3 | Address completed, change to reception of Group B | B-3 | **Unallocated number** |
| A-4 | Congestion in the national network | B-4 | Congestion |
| A-5 | **Not Used** | B-5 | **Subscriber line free, no charge** |
| A-6 | **Send calling party's category - and ANI digits** | B-6 | **Not Used** |
| A-7 | **Not Used** | B-7 | **Not Used** |
| A-8 | **Send last digit (N-1)** | B-8 | **Not Used** |
| A-9 | **Send second-to-last digit (N-2)** | B-9 | **Not Used** |
| A-10 | **Not Used** | B-10 | **Not Used** |
| A-11 | | B-11 | **Not Used** |
| A-12 | | B-12 | **Not Used** |
| A-13 | Send nature of circuit | B-13 | **Not Used** |
| A-14 | Request information on use of half-echo suppression | B-14 | **Not Used** |
| A-15 | | B-15 | **Not Used** |

The following is a list of forward absence or end of information tones for Singapore:

| Meaning | Tone | Comments |
|---|---|---|
| End of DID digits | I-15 | |
| End of ANI digits | I-15 | |
| Category not available | none | Compulsory category digit - the outbound side must provide the information |
| ANI digits not available | I-12 | If no ANI received |

The only other difference from CCITT specifications is that a number of values of the caller's category digit are defined, for which it is known that the ANI digits are not available. This is controlled by the parameter ADI.MFC.catnoANImask, that in this case has a value of 0x382. This means that the following categories are not followed by ANI information, and the TCP will not ask for ANI after receiving them:

- II-1: operator with trunk offering
- II-7: line test desk
- II-8: interception operator
- II-9: call from transit exchange without the calling subscriber number information

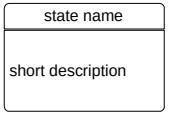
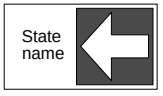
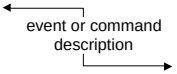
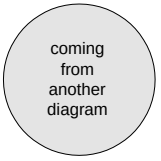
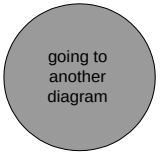
### 1.5.17 Thailand

Same as China.

## 1.6 State Diagrams

This section contains the state diagrams of the MFC-R2 TCP. States that interface with the line or the host are shown, grouped logically (if appropriate). For instance, fault states that wait for the line to return to idle, but can transition into one another if a new line fault code is received, are contracted into one. Pass-through states (states in the TCP script that simply execute instructions without receiving input from the external world) are omitted.

The following symbols are used in the state diagrams:

| Description | Icon |
|---|---|
| State block |   |
| Reference to a state within same machine |   |
| Link betweeen states |   |
| Reference *to* a state in another machine |   |
| Reference *from* a state in another machine |   |

Each of the following state diagrams describes a phase of the TCP execution flow.

### 1.6.1 Startup and Idle States

Figure 2 depicts the TCP states that implement the startup phase of the TCP, and its idle condition:

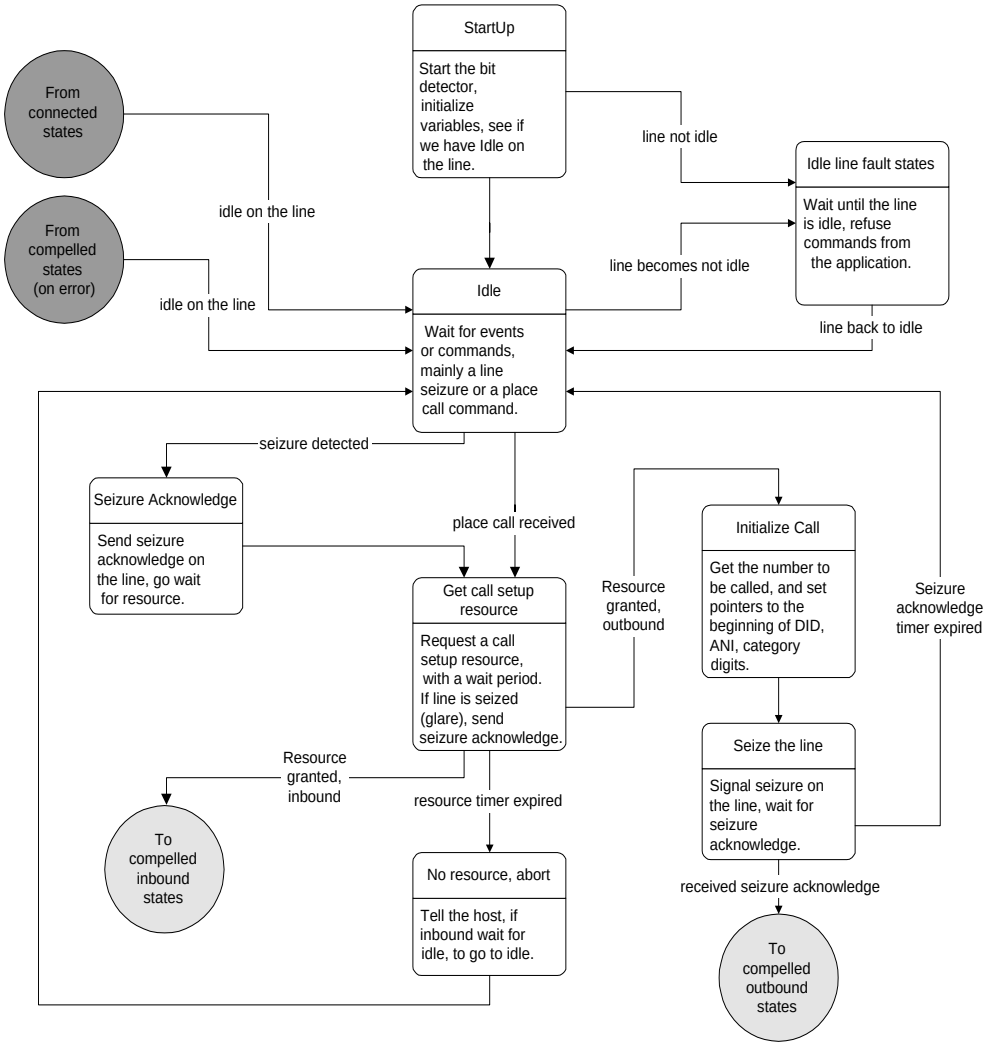



*Figure 2. State Diagram: Startup and Idle*

When in idle, the TCP waits for either a command to place a call, or a seizure on the line. In both cases, the resource management state is entered next. If resource management is not necessary for the type of board being used, the TCP immediately transitions the following state. If resource management is necessary, a call setup resource is requested and the state waits for it to be granted before continuing.

## 1.6.2 Inbound Compelled States

Figure 3 depicts the states going from the reception of the first DID digits to the reception of the Group II category digit, the last forward digits of the compelled sequence.

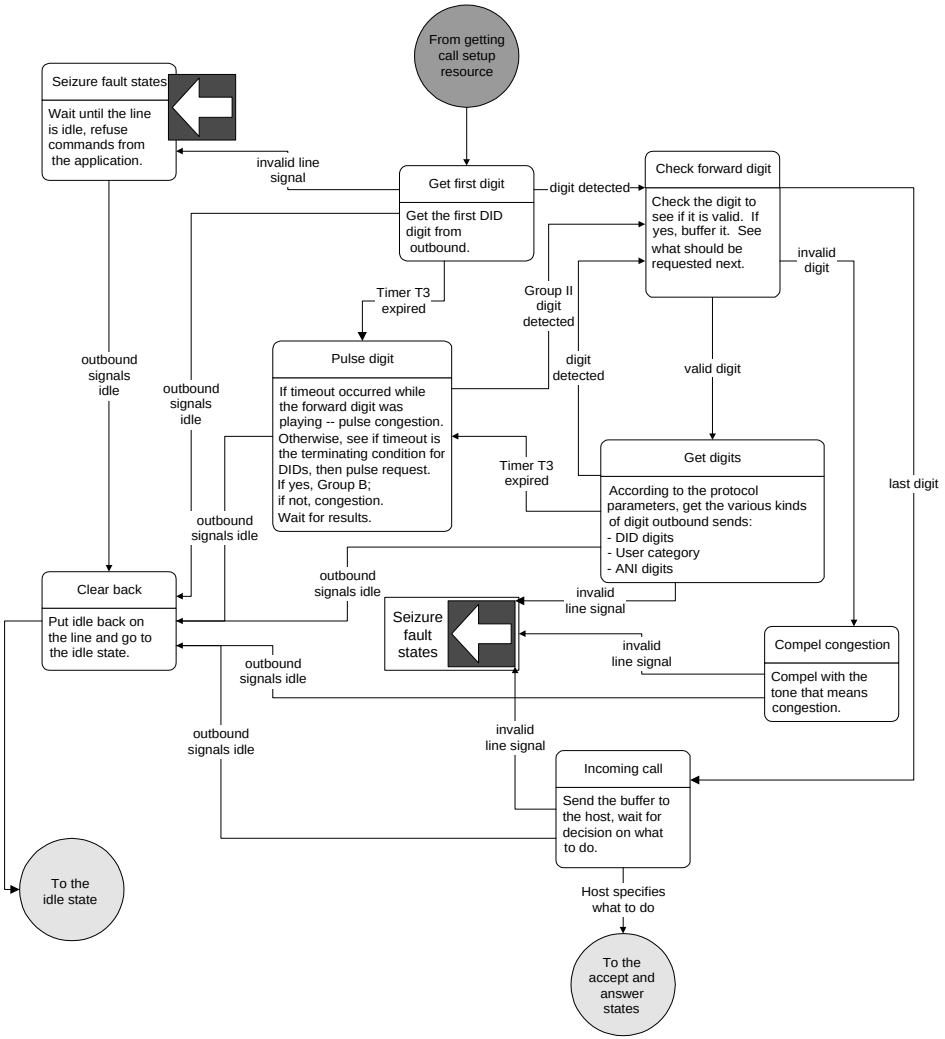



*Figure 3. State Diagram: Inbound Compelled States*

In the Incoming Call state, the TCP sends to the host an INCOMING_CALL event, with the received digits attached. The host must then decide if the call is to be:

- Accepted (without committing to answer it): **adiAcceptCall**, seen by the TCP as an ACCEPT_CALL command
- Answered: **adiAnswerCall**, seen by the TCP as an ANSWER_CALL command
- Rejected: **adiRejectCall**, seen by the TCP as a REJECT_CALL command

The TCP translates this command from the host into a backward Group B tone. See Section 1.5 for the meaning of the various Group B tones in each supported country. The backward Group B tone is the last of the compelled sequence. Its detection makes the outbound side stop the forward Group II tone, and ends the compelled sequence.

### 1.6.3 Inbound Accept and Answer Call States

Once the host has told the TCP what to do with the call, the TCP plays the Group B tone that accepts or rejects the call with different reasons and modes. Depending on the host's command, the TCP then does one of the following:

- If the call has been rejected, the TCP plays busy or lets the host play a reject message until the call is cleared by the calling party
- If the host committed to answer the call or commanded to accept the call and played ring indefinitely, the TCP plays ring tone and then answers
- If the call has been provisionally accepted, the TCP stays silent or allows the host to play a custom message

*Note:* If the TCP rejects the call, it is unlikely that it will stay in the Play Busy state for long, or that it will even reach this state, since the outbound side is supposed to clear the call as soon it detects the Group B rejection tone.

Figure 4 depicts behavior during the Inbound Accept and Answer Call States.

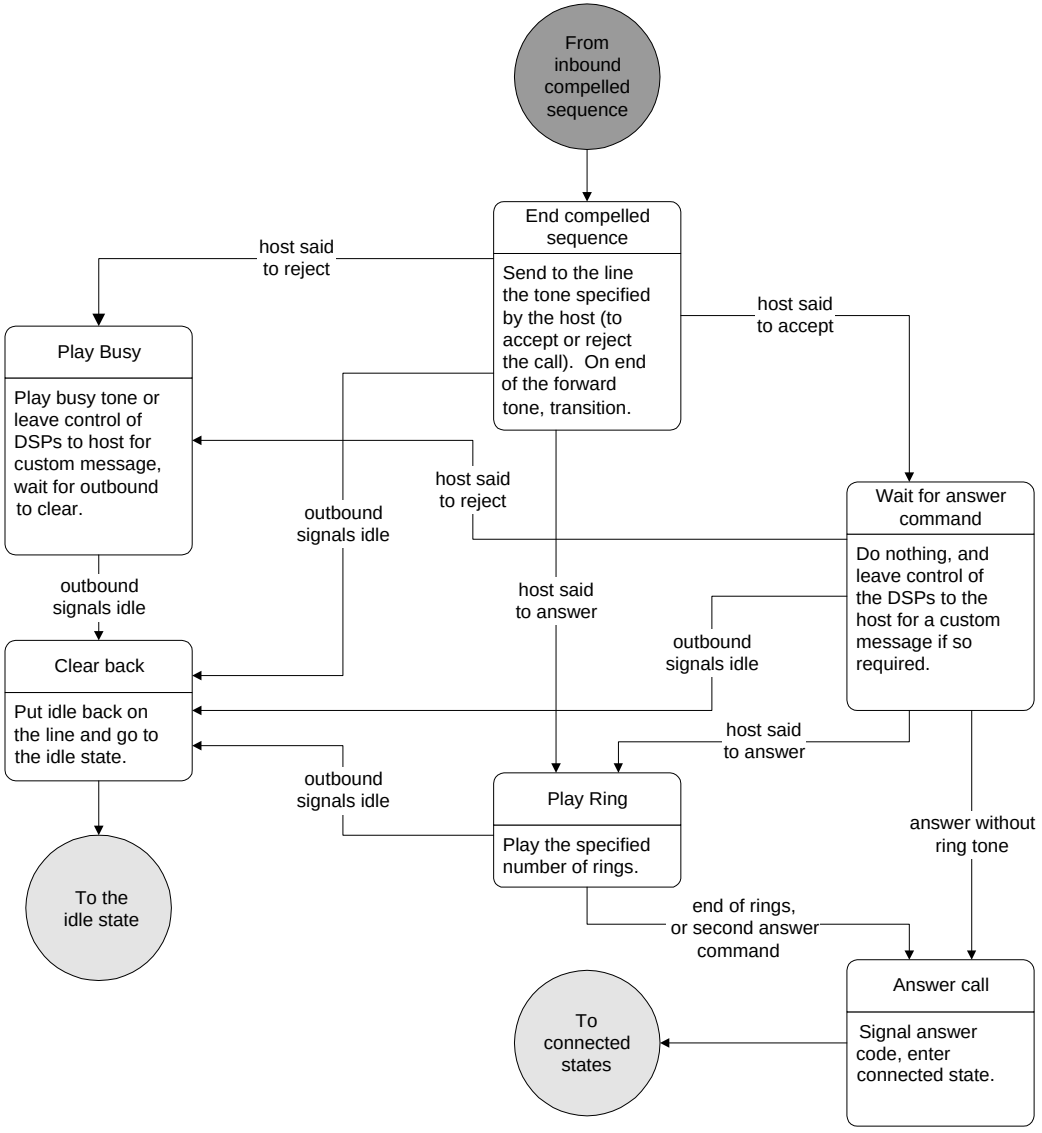



*Figure 4. State Diagram: Inbound Accept and Answer Call States*

### 1.6.4 Outbound Call Setup States

The outbound compelled sequence proceeds smoothly from one digit to the next. Backward tones from the inbound side are analyzed and a pointer to the forward digit to play next is set accordingly, or the compelled sequence aborts if a backward congestion indication is detected.

Figure 5 illustrates the cycle in a somewhat abstract manner. The main Send Digit state actually consists of three states:

- One that sends DID digits
- One that sends the category digit
- One that sends ANI digits

These states are accessed from the check backward digit state, which checks the received digit and sets the pointer to the next forward digit to transmit. In reality, this state is also a virtual one, composed of a number of states, to take into account the different backward tones that can come in the different phases of register signaling.

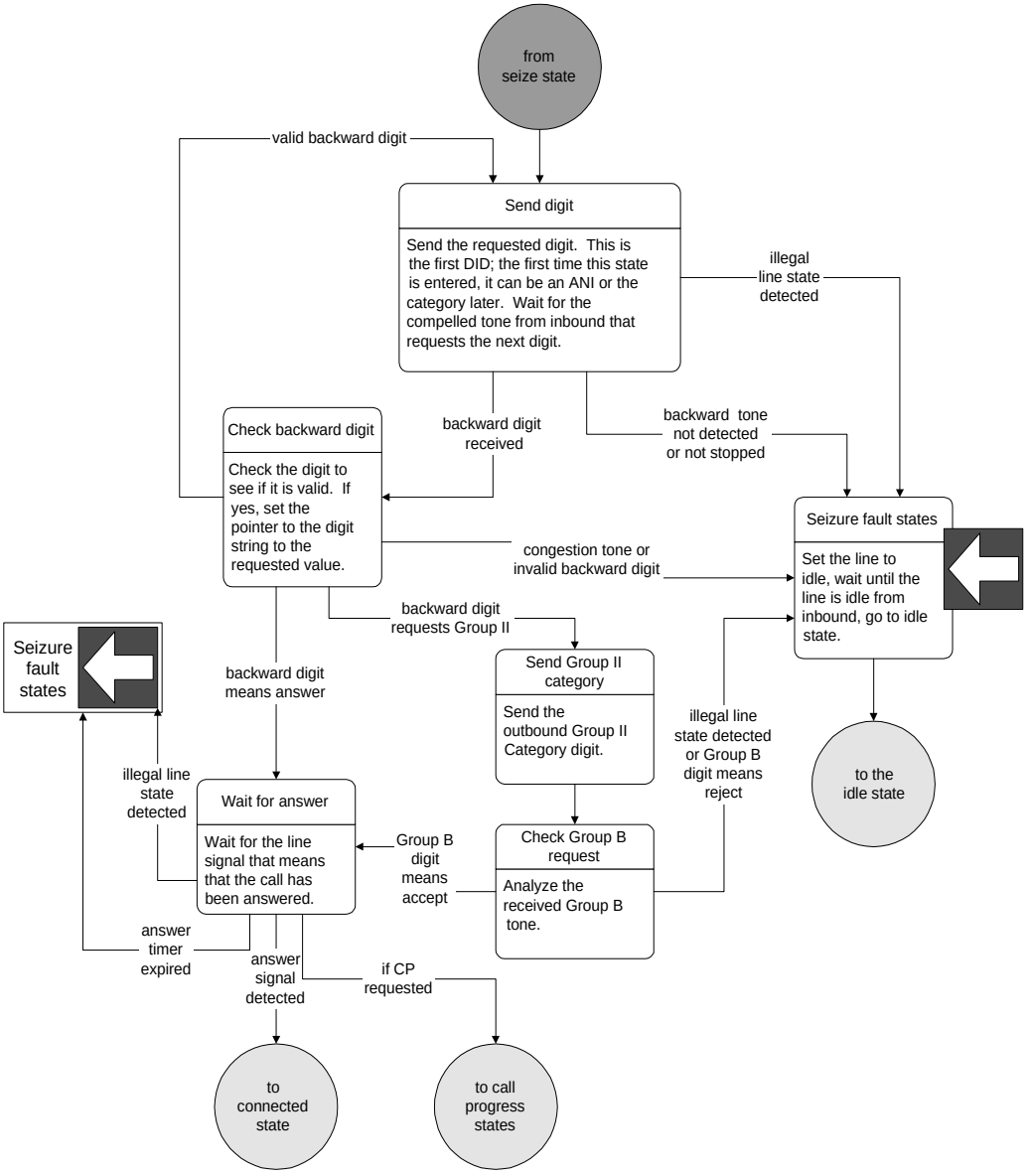



*Figure 5. State Diagram: Outbound Call Setup States*

## 1.6.5 Connected States

The host can request to perform call progress detection before declaring the call connected. Call progress is only used to connect on a particular condition (i.e., voice detected, SIT tone detected, etc.). It is not used to distinguish an accepted call from a rejected call, since the Group B backward tone contains this information.

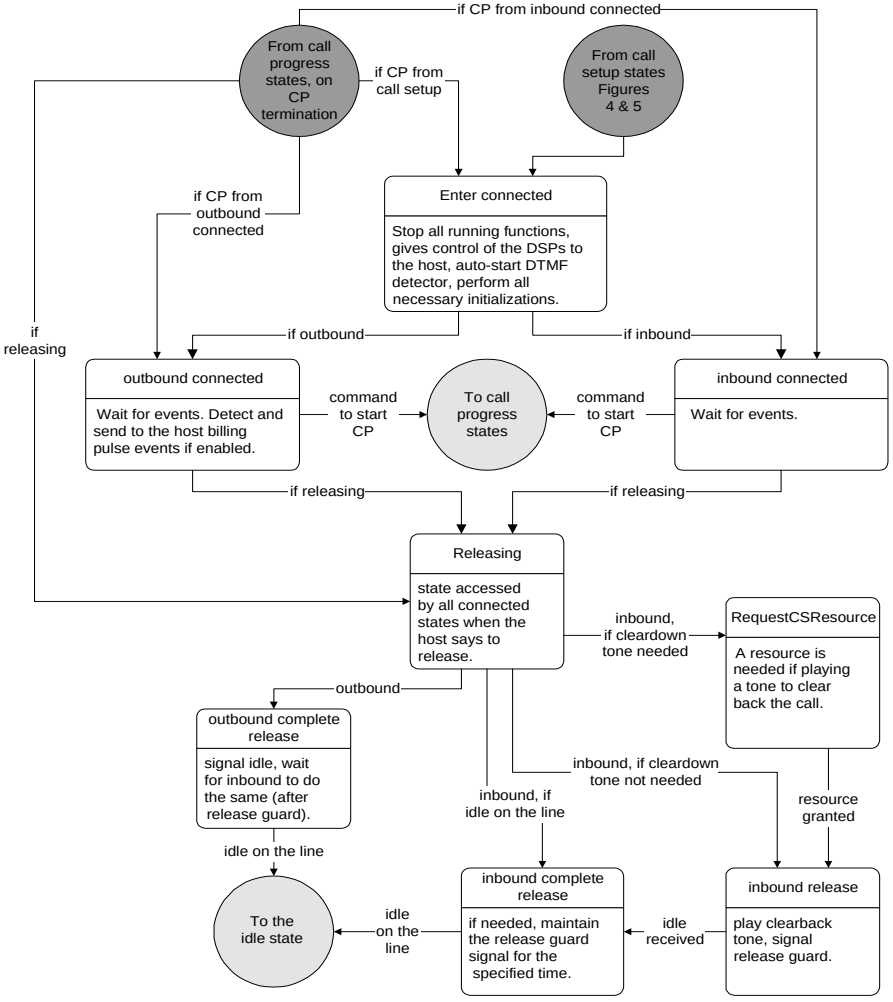



*Figure 6. State Diagram: Connected States*

## 1.7 TCP Performance

TCP performance is the measure of:

- How fast a call can be set up, either inbound or outbound
- The DSP resources necessary to set up a call
- The load limits, i.e., the number of calls that can be set up simultaneously for each board

To analyze the call setup timing, the signal on the line is captured and recorded. The spectral distribution of the energy on the line is then analyzed, and the timing is deduced. A model of the timing is also produced, to explain the observations in the light of the AG architecture. Note that in general the timing results will be different if the TCP is run “back-to-back” or is talking to a switch. In the case of MFC-R2, switches are usually faster than the TCP, so the call setup timing will generally be shorter if one of the sides is a network switch.

All results in this section come from tests executed on AG boards. AG 2000 and AG 4000 boards are faster. A full battery of tests has not been completed at the time of writing using AG 4000 boards.

*Note:* The end of this chapter contains some preliminary results of these tests.

The resources necessary to setup the call differ among protocols. The kinds of resources depend on the signaling method in use. Diagrams are included, plotting MIPS consumption against time, per channel. The timing of the diagram is derived from the previous time analysis and the TCP code, identifying the point in the TCP where DSP functions are started and stopped.

The number of calls that can be set up simultaneously depends on the TCP MIPS consumption, the type of board the TCP runs on, and (possibly) any task processors optimization carried out in the AG configuration file. Thus the matter is very complex, and only some general calculations and guidelines are reported in the following sections.

### 1.7.1 Call Setup Timing Analysis

Figure 7 represents the line energy distribution against time for a complete MFC-R2 compelled sequence, captured in back-to-back mode on an AG-E1 board.

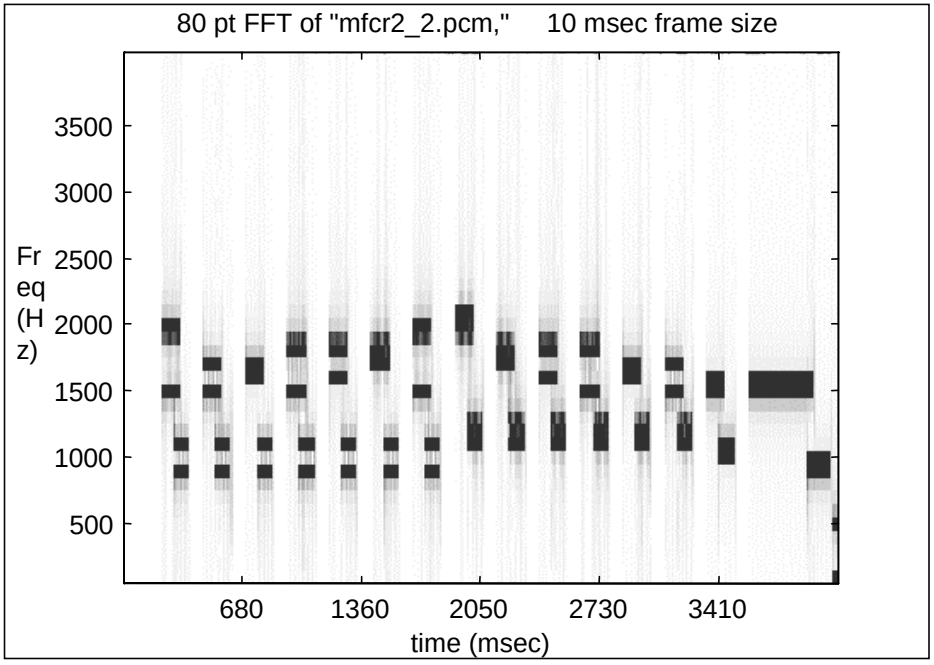



*Figure 7. Spectral Analysis of Line Energy Distribution*

Figure 7 was obtained by first adding the signals of the inbound and outbound voice channels (with the use of a conference card), recording the signal, and then performing a spectral analysis on it.

The compelled sequence represents the whole of register signaling for the MFC-R2 protocol. A compelled sequence is composed of compelled cycles, that are the unit of digit transfer. A compelled cycle is shown in Figure 8.

The time granularity is apparent here, each pixel representing a 10 ms time period. Figure 8 shows that a whole compelled cycle performed by two instances of the MFC-R2 TCP connected back-to-back takes exactly 240 ms to complete, from the moment in which a forward tone appears on the line to the moment the next forward tone is seen.

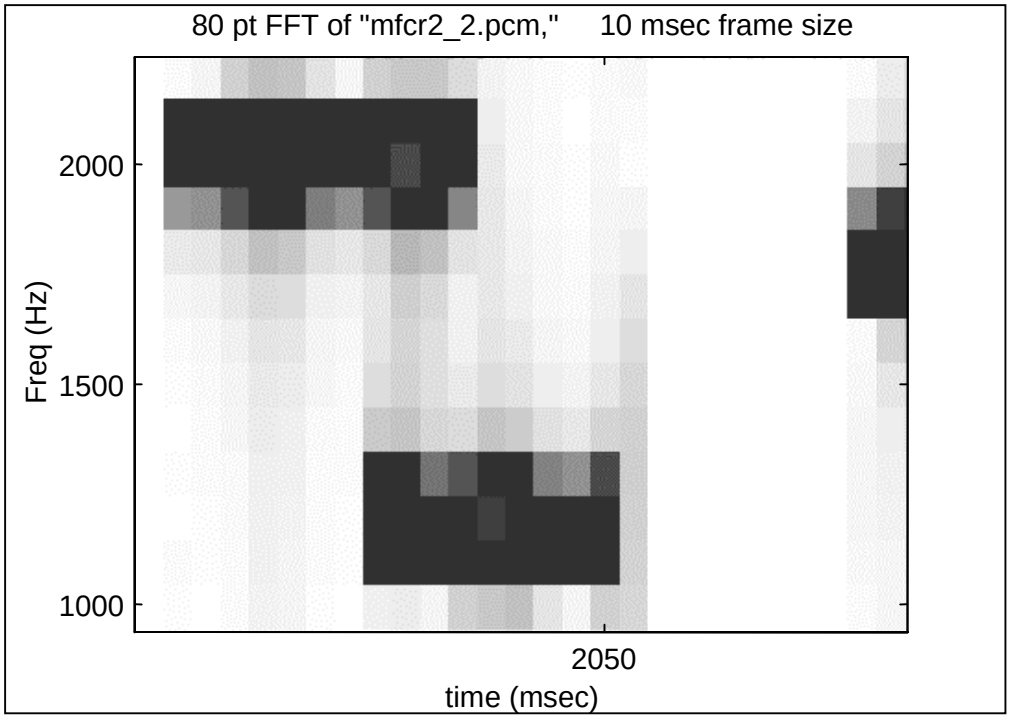



*Figure 8. Spectral Analysis of a Compelled Cycle*

In the spectral analysis in Figure 8:

- The forward tone is alone on the line for 70 ms.
- The two tones are both on the line for 40 ms.
- The backward tone is alone on the line for 50 ms.
- There is silence on the line for 80 ms.

This can be explained using the model contained in the following table (remembering that the clock cycle in an AG board is 10 ms long).

The table starts from the moment the outbound part of the TCP calls the service function that starts the compelled sequence cycle. However, we do not see the tone on the line until roughly 30 ms later, and this is taken as time 0 for this analysis.

Three entities play a role in this model: the TCP, the DSP overlay, and the MVIP processor. The line transmission time is negligible. Each event has an associated delay that follows it, and determines a step in the time count in the rightmost column. Refer to Figure 8 to identify the correct spot on the compelled cycle; this is marked by the *What is on the Line* column in the table. The qualification time for an incoming tone and an incoming silence varies between 20 and 30 ms, depending on when during the clock cycle the tone actually arrives to the MVIP / H.100 processor.

| Event | Associated Delay | What is on the Line | Time |
|---|---|---|---|
| Outbound TCP calls Compel_R2forward | | silence | -30 ms |
| | One clock cycle to get to the DSP | silence | |
| Outbound DSP starts the tone | | silence | -20 ms |
| | Double buffering, so 2 clock cycles to get to the MVIP bus / external line | silence | |
| Forward tone on the line | | forward tone | 0 ms |
| Inbound DSP starts receiving the tone | | forward tone | 0 ms |
| | Qualification time | forward tone | |
| Tone is detected by the inbound DSP | | forward tone | 30 ms |
| | Event is transmitted to the TCP | forward tone | |
| Inbound TCP receives the event | | forward tone | 40 ms |
| | TCP processing (negligible) | forward tone | |

| Event | Associated Delay | What is on the Line | Time |
|---|---|---|---|
| Inbound TCP calls Compel_R2backward | | forward tone | 40 ms |
| | One clock cycle to get to the DSP | forward tone | |
| Outbound DSP starts the tone | | forward tone | 50 ms |
| | Double buffering, so two clock cycles to get to the MVIP bus / external line | forward tone | |
| Backward tone on the line | | forward & back | 70 ms |
| Outbound DSP starts receiving the tone | | forward & back | 70 ms |
| | Qualification time | forward & back | |
| Outbound DSP stops the forward tone | | forward & back | 100 ms |
| | Double buffering, so two clock cycles to get to the MVIP bus / external line | forward & back | |
| Forward silence on the line | | back tone | 120 ms |
| Inbound DSP starts receiving silence | | back tone | 120 ms |
| | Qualification time | back tone | |
| Inbound DSP stops the backward tone | | back tone | 150 ms |
| | Double buffering, so two clock cycles to get to the MVIP bus / external line | back tone | |
| Backward silence on the line | | silence | 170 ms |
| Outbound DSP starts receiving silence | | silence | 170 ms |
| | Qualification time | silence | |

| Event | Associated Delay | What is on the Line | Time |
|---|---|---|---|
| Silence is detected by the outbound DSP | | silence | 200 ms |
| | Event is transmitted to the TCP | silence | |
| TCP receives the event | | silence | 210 ms |
| | TCP processing (negligible) | silence | |
| Outbound TCP calls Compel_R2forward | | silence | 210 ms |
| | One clock cycle to get to the DSP | silence | |
| Outbound DSP starts the tone | | silence | 220 ms |
| | Double buffering, so two clock cycles to get to the MVIP bus / external line | silence | |
| Forward tone on the line | | forward tone | 240 ms |

The same time analysis can be performed from the TCP point of view. The following table is a TCP trace with time stamps that mark the tens of milliseconds when the:

- Outbound TCP calls the compelled function
- Inbound TCP detects the forward digit
- Inbound TCP calls the compelled function
- Outbound TCP detects the backward tone
- Inbound TCP detects silence
- Outbound TCP detects silence

The phase of this period is the same as the one of the preceding table, starting from the compelled R2 forward cell.

For each event, the following table presents a:

- Time stamp (down to the 10 ms interval)
- TCP custom printout
- Comment

| Timestamp | TCP Printout | Comments |
|---|---|---|
| 111.68 | Outbound side, about to call compel | Sends first digit. The tone will be on the line 30 ms later. |
| 111.75 | Inbound side, digit1 detected: 2 | First digit received by the inbound side. |
| 111.75 | Inbound side, about to call compel | It takes virtually no time for the TCP to analyze the digit and decide what to compel with; all states are pass-through. The tone will be on the line 30 ms later. |
| 111.82 | Outbound side, backward tone 5 detected | The DSP stops the forward tone 10 ms later, since it takes 20 ms for the tone to get to the line. Tone A5 means send Category digit. The TCP waits for end of tone to do it. |
| 111.87 | Inbound side, end of tone detected | The DSP stops the backward tone 10 ms later. The TCP does nothing. |
| 111.92 | Outbound side, end of tone 5 detected | The DSP detects the end of tone. The TCP checks what the received digit means and starts a new compelled cycle with the requested new digit; in this case, the Category digit. The tone will be on the line 30 ms later. |
| **end of cycle - 240 ms** | | |

The overall call setup timing depends on the compelled cycle timing, as well as these other factors:

- The number of digits exchanged. There may be optimizations in the number of dialed or expected digits that eliminate one or more compelled cycles from a compelled sequence.
- The system load (for inbound calls). This is because after the caller's category has been collected, the TCP sends an ADIEVN_INCOMING_CALL event to the application. The application must then decide if the call is to be

accepted or rejected, and call the appropriate ADI function. This takes more time on loaded systems.

- The speed of the switch, generally faster than the TCP's.

## 1.7.2 Call Setup Resource Diagrams

Four call setup cases are presented that illustrate the MIPS consumption and the kind of DSP resources needed by the MFC-R2 TCPs to place and receive a call, plotted against time.

- Figure 9 shows seven DID digits, category digit, inbound, AG-E1. In this case, the DSP overlay to receive / transmit bits signaling is on. The compelled sequence starts with the TCP only detecting R2 forward tones. After the compelled sequence, the tone generation service is activated, to play ring tone.

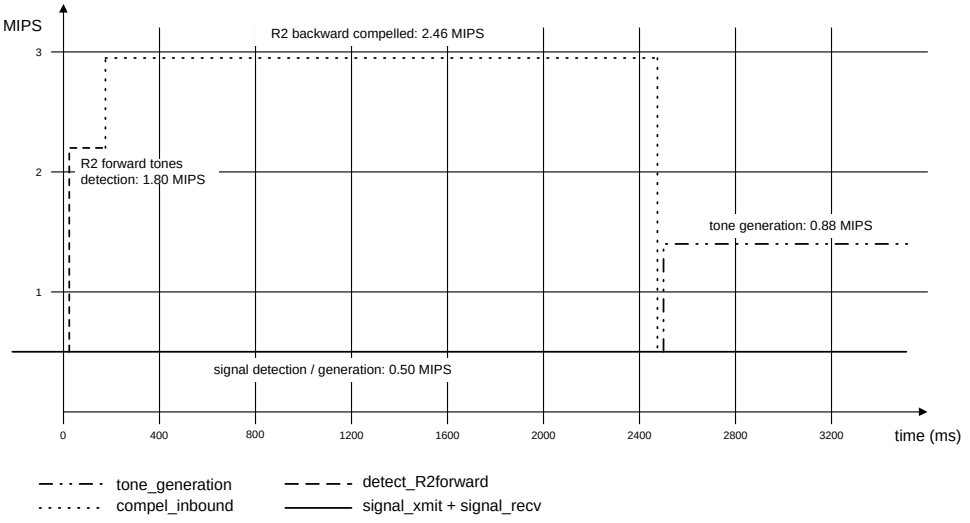



*Figure 9. MIPS Consumption and DSP Resources for Inbound AG-E1*

- Figure 10 shows seven DID digits, category digit, outbound, AG-E1. The DSP overlay to receive / transmit bits signaling is on, the compelled sequence starts with the TCP compelling forward tones. After the compelled sequence, call progress detection is activated (not compulsory).

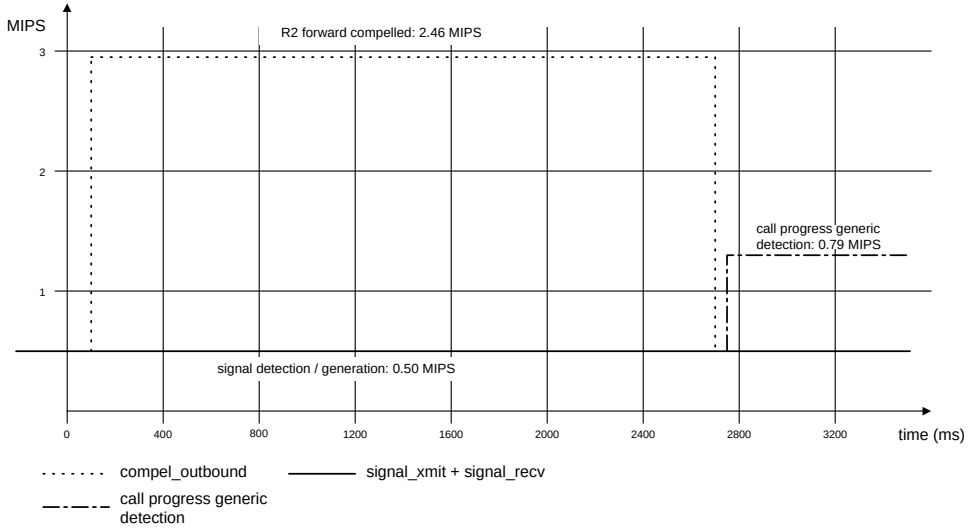



*Figure 10. MIPS Consumption and DSP Resources for Outbound AG-E1*

- Figure 11 shows seven DID digits, category digit, inbound, Quad E1. In this case, the signaling bits detection / generation function is carried out by the 486 coprocessor directly, and no DSP overlay is required for the task. This reduces the DSP load by 0.5 MIPS.

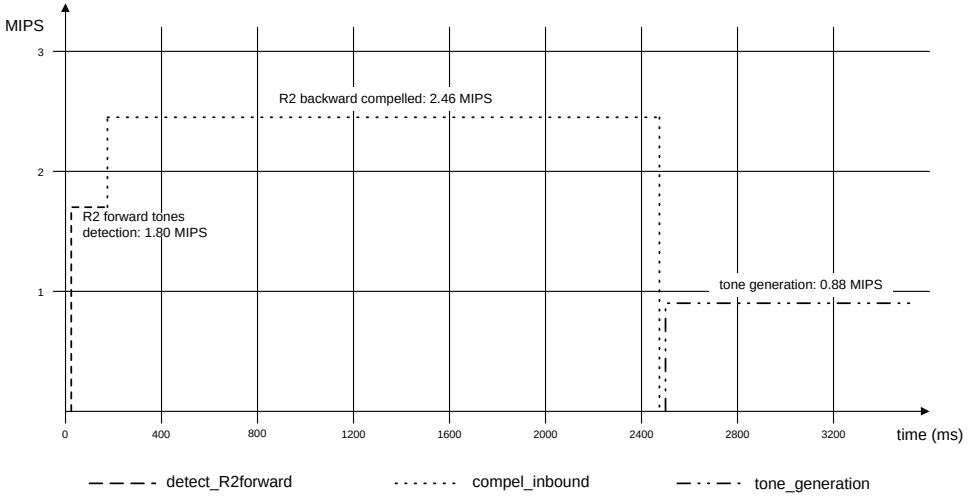



*Figure 11. MIPS Consumption and DSP Resources for Inbound Quad E1*

- Figure 12 shows seven DID digits, category digit, outbound, Quad E1. (Similar to Figure 11.)

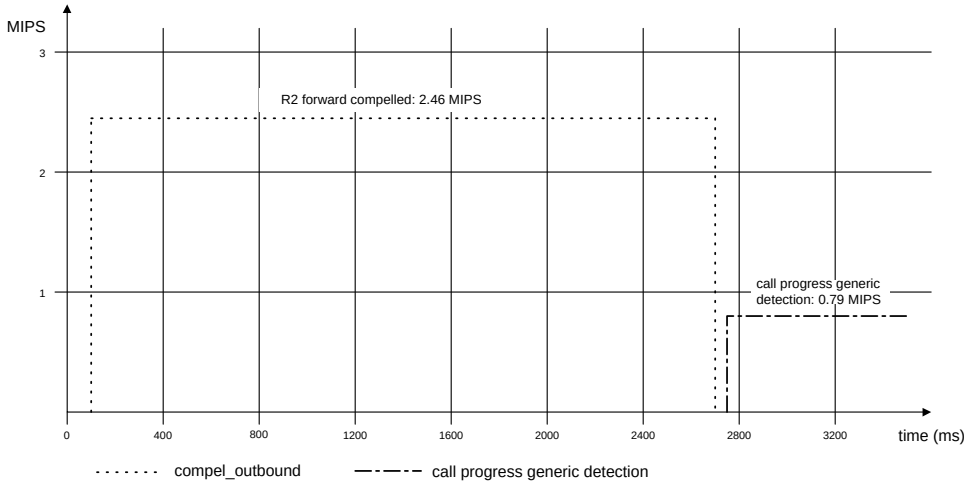



*Figure 12. MIPS Consumption and DSP Resources for Outbound Quad E1*

### 1.7.3 Behavior Under Load

The MFC-R2 TCP does not present problems when running under load on the AG-E1 ISA board series or on the AG Quad and Dual E PCI boards, when resource management is not needed. In this case, the TCP can run on all available channels simultaneously.

If all line interface channels are to be used on AG Quad E boards:

- You need to run the protocol on more channels than the available DSP power would support. Therefore, the resource management feature must be used. Calls are queued and the system performance depends on the kind of telephony application that runs on the host. In this case, it might happen that the application runs out of resources in the conversation phase of the call, but call setup proceeds normally regardless of the load.
- An explicit allocation of task processors on the board is necessary, to avoid running out of MIPS on some task processors, while under-utilizing others. See the AG configuration files for AG Quad E boards for details.

The CAStest utility can be used to test the protocol's behavior under load. It spawns a number of threads, each either placing or receiving calls. Then it collects statistics and errors from the threads. (For more information, see the CASmon User's Manual.) Examples of CAStest log files for AG-E1 and AG Quad E boards running at full capacity are presented.

## Load Test Results: Two AG-E1, One AG Quad E

In this test, two AG-E1 boards and one AG Quad E board are connected back to back. The AG Quad E board places calls to the two AG-E1 boards on all their available channels (60 channels in total). Statistics are gathered every 10 minutes.

```
Overnight test with 2 AG-E1 against 60 ports of a Quad E, MFC-R2
Quad E with CCMODE=NONE
================================================================================
CAStest - CTA Digital CAS protocols tester V.1 (Feb  3 1998) - inbound side
test start time = Tue Feb 03 18:28:02 1998
arguments = -i -b1 -t30 -T10 -v1
================================================================================
=
================================================================================
=
CAStest - CTA Digital CAS protocols tester V.1 (Feb  3 1998) - inbound side
test start time = Tue Feb 03 18:28:16 1998
arguments = -i -b2 -t30 -T10 -v1
================================================================================
=
================================================================================
=
CAStest - CTA Digital CAS protocols tester V.1 (Feb  3 1998) - outbound side
test start time = Tue Feb 03 18:28:37 1998
arguments = -t60 -T10 -v1 -n2345678
================================================================================
=
stat:       INBOUND time = 010 minutes
        attempted inbound calls..................... 542
        successful inbound calls.................... 528
        calls with no resource available for call setup.. 0
        calls with no resource available for application. 0
        calls with an out-of-service event.............. 0
        calls with an in-service event.................. 0
stat:       INBOUND time = 010 minutes
        attempted inbound calls..................... 570
        successful inbound calls.................... 540
        calls with no resource available for call setup.. 0
        calls with no resource available for application. 0
        calls with an out-of-service event.............. 0
        calls with an in-service event.................. 0
stat:       OUTBOUND time = 010 minutes
        attempted outbound calls.................... 1200
        successful outbound calls................... 1140
        calls with no resource available for call setup.. 0
        calls failed because inbound had no CS resource.. 0
        calls with no resource available for application. 0
        calls with no resource available for CP......... 0
        calls with an out-of-service event.............. 0
```

```
        calls with an in-service event.................. 0
stat:       INBOUND time = 020 minutes
        attempted inbound calls........................ 1140
        successful inbound calls....................... 1110
        calls with no resource available for call setup.. 0
        calls with no resource available for application. 0
        calls with an out-of-service event.............. 0
        calls with an in-service event.................. 0
stat:       INBOUND time = 020 minutes
        attempted inbound calls........................ 1140
        successful inbound calls....................... 1110
        calls with no resource available for call setup.. 0
        calls with no resource available for application. 0
        calls with an out-of-service event.............. 0
        calls with an in-service event.................. 0
stat:       OUTBOUND time = 020 minutes
        attempted outbound calls....................... 2340
        successful outbound calls...................... 2280
        calls with no resource available for call setup.. 0
        calls failed because inbound had no CS resource.. 0
        calls with no resource available for application. 0
        calls with no resource available for CP......... 0
        calls with an out-of-service event.............. 0
        calls with an in-service event.................. 0
----------------------------
printouts omitted - no errors
----------------------------
stat:       INBOUND time = 830 minutes
        attempted inbound calls........................ 47820
        successful inbound calls....................... 47790
        calls with no resource available for call setup.. 0
        calls with no resource available for application. 0
        calls with an out-of-service event.............. 0
        calls with an in-service event.................. 0
stat:       INBOUND time = 830 minutes
        attempted inbound calls........................ 47790
        successful inbound calls....................... 47760
        calls with no resource available for call setup.. 0
        calls with no resource available for application. 0
        calls with an out-of-service event.............. 0
        calls with an in-service event.................. 0
stat:       OUTBOUND time = 830 minutes
        attempted outbound calls....................... 95670
        successful outbound calls...................... 95610
        calls with no resource available for call setup.. 0
        calls failed because inbound had no CS resource.. 0
        calls with no resource available for application. 0
        calls with no resource available for CP......... 0
        calls with an out-of-service event.............. 0
        calls with an in-service event.................. 0
```

## Load Test Results: One AG Quad E Back to Back

In this test, one AG Quad E board places calls to itself. The first two trunks of the board receive calls. The second two trunks place calls. Resource management is used in this case, but the application is inert in the connected phase. This simulates a gateway application, that only switches calls through other boards / trunks, without playing or recording voice files, or detecting DTMFs. Statistics are gathered every minute.

```
Test with one Quad E with 2 trunks connected with the other 2, MFC-R2
Quad E AG config file with
   CallSetupPorts = 60      # 1/2 of total, on a Quad E
   CallSetupPool 0..6 = ALL
   taskprocessor 0..6 = mf_a.dsp,callp_a.dsp,tone_a.dsp
   taskprocessor 7 = callp_a.dsp,tone_a.dsp,dtmf_a.dsp,voice_a.dsp

================================================================================
=
CAStest - CTA Digital CAS protocols tester V.1 (Feb  2 1998) - inbound side
test start time = Mon Feb 02 11:37:37 1998
arguments = -i -v1 -t60 -N -T1 -C10
================================================================================
=
================================================================================
=
CAStest - CTA Digital CAS protocols tester V.1 (Feb  2 1998) - outbound side
test start time = Mon Feb 02 11:37:41 1998
arguments = -v1 -t60 -N -n#2345678#2345678 -s16:60 -T1 -C10
================================================================================
=
stat:       INBOUND time = 001 minutes
       attempted inbound calls..... 180
       successful inbound calls..... 120
       calls with no resource available for call setup.. 0
       calls with an out-of-service event..... 0
       calls with an in-service event..... 0
stat:       OUTBOUND time = 001 minutes
       attempted outbound calls..... 180
       successful outbound calls..... 120
       calls with no resource available for call setup.. 0
       calls failed because inbound had no CS resource.. 0
       calls with an out-of-service event..... 0
       calls with an in-service event..... 0
stat:       INBOUND time = 002 minutes
       attempted inbound calls..... 360
       successful inbound calls..... 300
       calls with no resource available for call setup.. 0
       calls with an out-of-service event..... 0
       calls with an in-service event..... 0
```

```
stat:      INBOUND time = 001 minutes
       attempted inbound calls..................... 180
       successful inbound calls.................... 120
       calls with no resource available for call setup.. 0
       calls with an out-of-service event.............. 0
       calls with an in-service event.................. 0
stat:      OUTBOUND time = 001 minutes
       attempted outbound calls.................... 180
       successful outbound calls................... 120
       calls with no resource available for call setup.. 0
       calls failed because inbound had no CS resource.. 0
       calls with an out-of-service event.............. 0
       calls with an in-service event.................. 0
stat:      INBOUND time = 002 minutes
       attempted inbound calls..................... 360
       successful inbound calls.................... 300
       calls with no resource available for call setup.. 0
       calls with an out-of-service event.............. 0
       calls with an in-service event.................. 0
stat:      OUTBOUND time = 002 minutes
       attempted outbound calls.................... 360
       successful outbound calls................... 300
       calls with no resource available for call setup.. 0
       calls failed because inbound had no CS resource.. 0
       calls with an out-of-service event.............. 0
       calls with an in-service event.................. 0
----------------------------
printouts omitted - no errors
----------------------------
stat:      OUTBOUND time = 046 minutes
       attempted outbound calls.................... 8156
       successful outbound calls................... 8100
       calls with no resource available for call setup.. 0
       calls failed because inbound had no CS resource.. 0
       calls with an out-of-service event.............. 0
       calls with an in-service event.................. 0
stat:      INBOUND time = 047 minutes
       attempted inbound calls..................... 8313
       successful inbound calls.................... 8261
       calls with no resource available for call setup.. 0
       calls with an out-of-service event.............. 0
       calls with an in-service event.................. 0
```

```
stat:       OUTBOUND time = 047 minutes
        attempted outbound calls..................... 8336
        successful outbound calls.................... 8276
        calls with no resource available for call setup.. 0
        calls failed because inbound had no CS resource.. 0
        calls with an out-of-service event............... 0
        calls with an in-service event................... 0
stat:       INBOUND time = 048 minutes
        attempted inbound calls...................... 8489
        successful inbound calls..................... 8434
        calls with no resource available for call setup.. 0
        calls with an out-of-service event............... 0
        calls with an in-service event................... 0
```

## AG 4000 Tests *Preliminary* Results

Tests on AG 4000 boards were carried out using the CASmon utility. CASmon is included in the TCP Developers Kit. CASmon can monitor any number of DSP or trunk timeslots, detecting tones and signaling bit changes. It saves the monitor results in a log file.

CASmon has an option to connect to two different ends of a connection, by monitoring a DSP resource and the corresponding trunk timeslot. This is useful to track both parties of a call setup. This option has been used to produce the following table.

The test setup is an AG 4000 board with two trunks connected with a loopback cable. Channel 60 places calls to channel 0 using MFC-R2.

```
------------------------------------------------------------
CASmon - CTA Digital CAS protocols tester V.1(Apr 28 1999)
test start time = Wed Apr 28 10:44:01 1999
------------------------------------------------------------
00:trunk 1:000  10:44:21.200  Event: 0x20002065     0001
00:trunk 1:000  10:44:21.200  Event: 0x2000206f     Seizure detected
00:dsp:000      10:44:21.250  Event: 0x20002065     1101
00:dsp:000      10:44:21.250  Event: 0x20002070     Seizure ack detected
00:trunk 1:000  10:44:21.320  Event: 0x20002066     R2F-1
00:dsp:000      10:44:21.360  Event: 0x20002067     R2B-5
00:trunk 1:000  10:44:21.400  Event: 0x2000206a     R2F-END
00:dsp:000      10:44:21.440  Event: 0x2000206b     R2B-END
00:trunk 1:000  10:44:21.480  Event: 0x20002066     R2F-1
00:dsp:000      10:44:21.520  Event: 0x20002067     R2B-5
00:trunk 1:000  10:44:21.560  Event: 0x2000206a     R2F-END
00:dsp:000      10:44:21.600  Event: 0x2000206b     R2B-END
00:trunk 1:000  10:44:21.640  Event: 0x20002066     R2F-C
00:dsp:000      10:44:21.680  Event: 0x20002067     R2B-1
00:trunk 1:000  10:44:21.720  Event: 0x2000206a     R2F-END
00:dsp:000      10:44:21.760  Event: 0x2000206b     R2B-END
00:trunk 1:000  10:44:21.800  Event: 0x20002066     R2F-2
00:dsp:000      10:44:21.840  Event: 0x20002067     R2B-1
00:trunk 1:000  10:44:21.880  Event: 0x2000206a     R2F-END
00:dsp:000      10:44:21.920  Event: 0x2000206b     R2B-END
00:trunk 1:000  10:44:21.960  Event: 0x20002066     R2F-3
00:dsp:000      10:44:22.000  Event: 0x20002067     R2B-1
00:trunk 1:000  10:44:22.040  Event: 0x2000206a     R2F-END
00:dsp:000      10:44:22.080  Event: 0x2000206b     R2B-END
00:trunk 1:000  10:44:22.120  Event: 0x20002066     R2F-4
00:dsp:000      10:44:22.160  Event: 0x20002067     R2B-1
00:trunk 1:000  10:44:22.200  Event: 0x2000206a     R2F-END
00:dsp:000      10:44:22.240  Event: 0x2000206b     R2B-END
00:trunk 1:000  10:44:22.280  Event: 0x20002066     R2F-5
00:dsp:000      10:44:22.320  Event: 0x20002067     R2B-1
```

```
00:trunk 1:000  10:44:22.360  Event: 0x2000206a     R2F-END
00:dsp:000      10:44:22.400  Event: 0x2000206b     R2B-END
00:trunk 1:000  10:44:22.440  Event: 0x20002066     R2F-6
00:dsp:000      10:44:22.480  Event: 0x20002067     R2B-1
00:trunk 1:000  10:44:22.520  Event: 0x2000206a     R2F-END
00:dsp:000      10:44:22.560  Event: 0x2000206b     R2B-END
00:trunk 1:000  10:44:22.600  Event: 0x20002066     R2F-7
00:dsp:000      10:44:22.640  Event: 0x20002067     R2B-3
00:trunk 1:000  10:44:22.680  Event: 0x2000206a     R2F-END
00:dsp:000      10:44:22.720  Event: 0x2000206b     R2B-END
00:trunk 1:000  10:44:22.760  Event: 0x20002066     R2F-1
00:dsp:000      10:44:22.810  Event: 0x20002067     R2B-1
00:trunk 1:000  10:44:22.850  Event: 0x2000206a     R2F-END
00:dsp:000      10:44:22.880  Event: 0x2000206b     R2B-END
00:dsp:000      10:44:23.880  Event: 0x20002065     0101
00:dsp:000      10:44:23.880  Event: 0x20002071     Answer detected
00:trunk 1:000  10:44:43.240  Event: 0x20002065     1001
00:trunk 1:000  10:44:43.240  Event: 0x20002072     Clear Forward detected
00:dsp:000      10:44:43.260  Event: 0x20002065     1101
00:dsp:000      10:44:43.260  Event: 0x20002075     Release_Guard detected
00:dsp:000      10:44:43.290  Event: 0x20002065     1001
00:dsp:000      10:44:43.290  Event: 0x2000206e     Idle detected
```

Results are interpreted in the following table, showing the first three compelled cycles (shaded differently):

| Event | Timing (ms) | Comments |
|---|---|---|
| Seizure | 0 | Taken as base of timing |
| Seizure Acknowledge | 50 | |
| R2 forward 1 | 120 | Beginning of first DID |
| R2 backward 5 | 160 | Beginning of backward tone (request category) |
| R2 forward end | 200 | End of first DID |
| R2 backward end | 240 | End of backward tone |
| R2 forward 1 | 280 | Beginning of category digit |
| R2 backward 5 | 320 | Beginning of backward tone (request first ANI) |
| R2 forward end | 360 | End of category digit |
| R2 backward end | 400 | End of backward tone |

| Event | Timing (ms) | Comments |
|---|---|---|
| R2 forward C | 440 | ANI not available (I-12) |
| R2 backward 1 | 480 | Beginning of backward tone (request next DID) |
| R2 forward end | 520 | End of category digit |
| R2 backward end | 560 | End of backward tone |
| … | … | … |

Based on these results, we can conclude that AG 4000 boards are about 33% faster than AG boards in executing the MFC-R2 compelled sequence, with a compelled cycle time of about 160 ms instead of 240 ms in back-to-back operation.

Chapter 2

# R1.5

## 2.1 Introduction

This chapter characterizes the Natural MicroSystems System R1.5 protocol package. The package consists of:

- Two Trunk Control Programs (TCPs) that run on Natural MicroSystems AG Boards, performing call setup functions with the R1.5 line protocol. The TCPs interact with an application running on the host through the NMS standard API (CT Access), by receiving commands and sending events according to CT Access's specifications.
- A set of parameter files, each programming the TCP to execute the R1.5 variation used in a particular country or on a particular network.

This chapter is divided into three sections, dealing with the TCPs:

| Section | Description |
|---|---|
| Line Signaling | Set of signals that represent fixed events in the course of call setup. |
| Register Signaling | The way the protocol transmits information as it changes for each call; as addresses and category specifications for the call. |
| Parameters | A description of all the parameters the host can send the TCP to program its behavior. |

## 2.2 System R1.5 Line Signaling

System R1.5 is an asymmetrical protocol that is widely used in Russia over E1 trunks. It has provisions to interface with analog and digital switches.

There are two different line signaling schemes:

| Line Signaling Scheme | Description |
|---|---|
| Group 1 | Direct connection lines and request-connection lines (local inbound side and local and long distance outbound side). |
| Group 2 | Incoming inter-city direct connection lines (long distance inbound side only). |

Because of the protocol's asymmetry, there are two TCPs in the Natural MicroSystems implementation: one to receive inbound calls and one to place outbound calls. This is needed because the protocol asymmetry prevents the possibility of two-way trunks with R1.5.

| TCP | Description |
|---|---|
| *R150.TCP* (inbound side only) | • The local inbound protocol, with the Multi-Frequency Pulse Shuttle (MF-PS) or Decadic Pulses (DEK) register signaling, using signaling Group 1.<br>• The long-distance inbound protocol, with the Multi-Frequency Pulse Shuttle (MF-PS) or Decadic Pulses (DEK) register signaling, using signaling Group 2. |
| *R151.TCP* (outbound side only) | • The local outbound protocol, with Multi-Frequency Pulse Shuttle or Decadic Pulses register signaling, using signaling Group 1.<br>• The long distance outbound protocol, with register signaling being a combination of Pulse Shuttle and Decadic Pulses, using signaling Group 1. |

Symmetry occurs only between the local inbound side in the *R150.TCP* and the local outbound side in the *R151.TCP*. In this case, the CPE inbound and the CPE outbound expect to see complementary line and register signaling, enabling the two protocols to run in back-to-back mode. In all other cases, an R1.5 switch is required to establish a connection.

In the following tables, the signaling channels supporting the R1.5 trunk protocol are referred to as $A_f$ and $B_f$, in the forward direction and $A_b$ and $B_b$, in the backward direction. The forward channel indicates the condition of the outgoing equipment and reflects the condition of the line of the *calling* party. The backward channel indicates the condition of the line of the *called* party.

### 2.2.1 R1.5 Line Signaling – Group 1

The following table summarizes the signaling states of a typical call using line signaling Group 1:

| State | Outbound $A_fB_f$ | Direction | Inbound $A_bB_b$ |
|---|---|---|---|
| Idle | 11 | ← → | 01 |
| Seizure | 10 | → | 01 |
| Seizure Acknowledged | 10 | ← | 11 |
| Depending on the protocol, either the inbound side starts to send the in-band tones requesting the address information, or the outbound side starts decadic dialing. | | | |
| Decadic dialing | (1/0/1)0 | → | 11 |
| The inbound side completes the sequence by accepting or rejecting the call, using a specific tone. If the call has been accepted, the inbound side may request ANI information. This is accomplished by an answer signal, accompanied by a 500 Hz tone. If no tone had been played within a certain time limit, the outbound side assumes that the signal meant that the call is connected. | | | |
| Answer (or ANI request) | 10 | ← | 10 |
| After the outbound side transmits the ANI information, the inbound side may remove the answer signal and either play a ring tone on the line and then signal that the call has been answered by resetting the $B_b$ bit to 0, or repeat the ANI request.<br>If the inbound side rejects the call, the inbound side clears backward by setting both the $A_b$ and the $B_b$ bit to 0 and, depending on the protocol, may also play a backward tone indicating reject. The outbound side goes back to idle, by setting the $B_f$ bit to 1, then the inbound side goes to idle too, by setting the $B_b$ to 1. | | | |
| Clear back | 10 | ← | 00 |
| Clear forward | 11 | → | 00 |
| Idle (backward) | 11 | ← | 01 |
| After the connected state has been reached, depending on which side hangs up the call first, we have a *clear back* signal, or a *clear forward* signal. Idle follows. | | | |
| Inbound hangs up first: Clear back | 10 | ← | 00 |
| Clear forward | 11 | → | 00 |

| State | Outbound $A_fB_f$ | Direction | Inbound $A_bB_b$ |
|---|---|---|---|
| Idle | 11 | ← | 01 |
| Outbound side hangs up first: Clear forward | 11 | → | 1X |
| Idle (backward) | 11 | ← | 01 |
| There is a different hang up signal, depending on whether the inbound side or the outbound side hangs up first in the after-answer state: | | | |
| Clear forward, after answer | 00 | → | 10 |
| Idle (inbound side) | 00 | ← | 01 |
| Idle (outbound side) | 11 | → | 01 |

## 2.2.2 R1.5 Line Signaling – Group 2

The following table summarizes the signaling states of a typical call using signaling Group 2. (Long distance inbound protocol only):

| State | Outbound $A_fB_f$ | Direction | Inbound $A_bB_b$ |
|---|---|---|---|
| Idle | 11 | ← → | 01 |
| Seizure | 10 | → | 01 |
| Seizure Acknowledged | 10 | ← | 11 |
| Depending on the protocol, either the inbound side starts to send the in-band tones requesting the address information, or the outbound side starts decadic dialing. | | | |
| Decadic dialing | (0/1/0)0 | → | 11 |
| The inbound side completes the sequence by accepting or rejecting the call. If the call has been accepted, the inbound side sends "subscriber free" on the line. | | | |
| Subscriber free | 10 | ← | 10 |
| In the case of a long distance inbound call, the inbound side does not play ring. Ring is played by the remote switch and is triggered by “sending a call” signal from the local switch. | | | |

| State | Outbound $A_fB_f$ | Direction | Inbound $A_bB_b$ |
|---|---|---|---|
| Sending a call | 00 | → | 10 |
| Answer signal may come either before or during the sending of a call. No ANI request is allowed in this protocol. | | | |
| Answer | 1X | ← | 11 |
| If the inbound side rejects the call, the inbound side clears backward, by setting both the $A_b$ and the $B_b$ bit to 0. The outbound side goes back to idle, by setting the $B_f$ bit to 1, and then the inbound side goes to idle, but setting the $B_b$ to 1. | | | |
| Clear back | 10 | ← | 00 |
| Clear forward | 11 | → | 00 |
| Idle (backward) | 11 | ← | 01 |
| Depending on which side hangs up the call first, we have a *clear back* signal, or a *clear forward* signal. Idle follows. | | | |
| Inbound hangs up first: Clear back | 10 | ← | 00 |
| Clear forward | 11 | → | 00 |
| Idle | 11 | ← | 01 |
| Outbound side hangs up first: Clear forward | 11 | → | 1X |
| Idle (backward) | 11 | ← | 01 |
| If the inbound side hangs up first in the connected state, it is possible for the operator to restore the connection (a recall). To accomplish this, inbound (instead of signaling the release sequence) signals "subscriber free". Then, if the outbound side signals idle, inbound also signals idle. However, if the outbound side signals “sending a call”, the inbound returns to the pre-answered state and sends “incoming call” again. | | | |
| Inbound first: Clear back after answer | 10 | ← | 10 |
| Outbound side (recall) | 00 | → | 10 |

## 2.3 System R1.5 Register Signaling

This section defines the register signaling protocol implemented by the NMS R1.5 TCP. Two different register signaling types are considered:

- Decadic(DEK)
- Multi-Frequency Pulse Shuttle (MF-PS)

In addition, Multi-Frequency Continuous Packet (MF-CP) is implemented specifically for the transmission of ANI information.

The following table shows the type of register signaling implemented by each protocol, and the information carried by it:

| **Protocol** | **Type of Call** | **Supported Register Signaling** |
|---|---|---|
| Protocol | type of call | Supported Register Signaling |
| DEK | all | DID information |
| MF-PS | local + long distance inbound | DID information |
| MF-IP1 | long distance outbound side | DID + ANI information |
| MF-IP2 | long distance outbound side | DID + ANI information |
| MF-CP (continuous packet) | local + long distance outbound side* | ANI information by request |

* Used during ANI transmission by outbound side when it receives the request for ANI (AB-10 + 500 Hz tone)

This protocol consists of the exchange of multi-frequency tones. The tones used in both directions are CCITT R1 MF (US MF) and are composed of two single-frequencies each, according to the following table:

| **Digit** | **Frequency (Hz)** | | | | | |
|---|---|---|---|---|---|---|
| **#** | 700 | 900 | 1100 | 1300 | 1500 | 1700 |
| 1 | X | X | | | | |
| 2 | X | | X | | | |
| 3 | | X | X | | | |
| 4 | X | | | X | | |
| 5 | | X | | X | | |
| 6 | | | X | X | | |
| 7 | X | | | | X | |
| 8 | | X | | | X | |
| 9 | | | X | | X | |
| 10 | | | | X | X | |
| 11 | X | | | | | X |
| 12 | | X | | | | X |
| 13 | | | X | | | X |
| 14 | | | | X | | X |
| 15 | | | | | X | X |

*Note:* This table is interpreted as follows: each tone is composed by the two single frequency tones marked with an **X**. For example, tone 15 is composed of (1500+1700) Hz.

Register signaling in the R1.5 is not necessarily accomplished by using only one of the protocols described in the following sections. Depending on the type of call and on the type of switches encountered along the call path, the protocol may be required to switch from one type of signaling protocol to another.

### 2.3.1 Register Signaling – MF-PS

Register signaling can be carried out with decadic pulses, exchange of MF tones, or a combination of the two. The inbound side is the master and requests information from the outbound side.

The basic scheme of in-band signaling in the MF-PS protocol is illustrated in the following table:

| **Outbound** | **Direction** | **Inbound** |
|---|---|---|
| (seizes the line) | | (seizure acknowledge) |
| | ← | Request for first DID (B-1) |
| DID #1 | → | |
| | ← | Request for next DID (B-2) |
| DID #2 | → | |
| | ← | Request for next DID (B-2) |
| DID #3 | → | |
| | ← | Request to repeat DID (B-3) |
| DID #3 | → | |
| | … | |
| | ← | Digits received (B-4) |
| request confirmation (A-12) | → | |
| | | (answer or reject) |

*Note:* B-8, B-9 or B-10 can come as a signal to switch to Decadic dialing (DEK). Also, B-1 is not necessarily the first command. Valid first commands include: B-1, B-2, and B-3.

Tones have specific meanings in the context of the call setup. Forward tones are used to convey both address information (the number being called) and information about the call (the caller's number and the call's category). Also, some tones are used specifically as control tones. Backward tones are used only for transmission of control signals. The following table shows the meaning of all the tones:

| **Forward** | | **Backward** | |
|---|---|---|---|
| Designation of the Signal | Meaning of the Signal | Designation of the Signal | Meaning of the Signal |
| A-1 | digit 1 | B-1 | Transmit the first digit |
| A-2 | digit 2 | B-2 | Transmit the next digit |
| A-3 | digit 3 | B-3 | Transmit the previous digit |
| A-4 | digit 4 | B-4 | End of call setup |
| A-5 | digit 5 | B-5 | Reject |
| A-6 | digit 6 | B-6 | Repeat the last forward tone (if not intelligible). |
| A-7 | digit 7 | B-7 | Congestion |
| A-8 | digit 8 | B-8 | Transmit all DID as decadic pulses (switch to DEK register signaling). |
| A-9 | digit 9 | B-9 | Transmit the next DID, and all the rest of the DIDs as decadic pulses (switch to DEK register signaling). |
| A-10 | digit 0 | B-10 | Transmit the previous DID and then all the rest of the DIDs as decadic pulses (switch to DEK register signaling). |
| A-11 | Automatic call from subscriber with category III of priority service. | B-11 | Request the information on the type of call and the category of priority service. |
| A-12 | Confirmation of the reception of (special) backward signals. Sent in response to B-4, B-5, B-8, B-9, B-10 | B-12 | Spare |

| **Forward** | | **Backward** | |
|---|---|---|---|
| Designation of the Signal | Meaning of the Signal | Designation of the Signal | Meaning of the Signal |
| A-13 | Repeat the last backward tone (if not intelligible) | B-13 | Spare |
| A-14 | Automatic call from subscriber with category IV of service (without priority) | B-14 | Spare |
| A-15 | Semi-automatic call | B-15 | Information not accepted (followed by a backward release) |

These tones are timed, not compelled, each lasting 45 ms.

The exchange of information is carried out as follows:

- The inbound side sends a request by playing an MF tone.
- The outbound side detects the request from the inbound side, and waits for the tone to end.
- Once the outbound register detects the end of the tone, it waits for a few tenths of a millisecond. This is to allow the other side to disqualify the echo deriving from the last tone played.
- The outbound side dials a digit, according to the inbound side’s request.
- The inbound detects the tone from the outbound side, and waits for the tone to end.
- Once the inbound register detects the end of the tone, it waits for a few tenths of a millisecond. This is to allow the other side to disqualify the echo deriving from the last tone played.
- The cycle continues.

### MF-PS Register Signaling Timers

The following table represents the default value of the timers and the events for which they are used:

| Timer # | Description | Duration (ms) |
|---|---|---|
| T1 | Time to recieve "seizure acknowledged". | 400 |
| T2 | Time between seizure acknowledge and the reception of the first backward request. | 4000 |
| T3 | Time between end of forward signal and the reception of the backward request. | 4000 |
| T4 | Time between end of backward request and the reception of the forward signal. | 350 |
| T5 | Duration of the frequency transmission (both backward and forward tones; for example: A-1 or DID). | 45 |
| T6 | Time from end of reception of tone signal to beginning of transmission (both for inbound and outbound, pause between requests and DIDs). | 60 |
| T7 | Time from end of transmission of tone signal to activating the MF detector (guaranteed no tone on the line after end of transmission - echo cancellation). | 40 |

## 2.3.2 Register Signaling – DEK (in Combination With MF-PS)

| Outbound | Direction | Inbound |
|---|---|---|
| (seizes the line) | | (seizure acknowledge) |
| | ← | Request for first DID (B-1) |
| DID #1 | → | |
| | ← | Request for next DID (B-2) |
| DID #2 | → | |
| | ← | Request for next DID (B-2)<br>Switching from MF-PS to decadic dialing can happen at any time during register signaling, with one of the following tones:<br>• B-8: request for all DID digits in decadic form<br>• B-9: request for DID digit N+1 (if the last DID dialed is N) and all following DID digits<br>• B-10: request for DID digit N-1 and all following DID digits |
| A-12: request acknowledgement | → | |
| if B-8: DID #1...n<br>if B-9: DID #N...n<br>if B-10: DID #N-1...n | → | |
| | | (answer or reject) |

## 2.3.3 Register Signaling – ANIs

The outbound side transmits the ANI digits (the caller's identification number) as a continuous stream of MF tones (without silence between consecutive tones) after the inbound side requests them.

The request consists of the establishment of the Answer line code (AB = 10) together with a pure 500 Hz tone, for more than 90 ms. The tone must begin no later than 275 ms after the answer line code is established, otherwise, the outbound side will consider the call connected.

## Timing

Figure 13 shows the delays and necessary timers for best ANI detection:

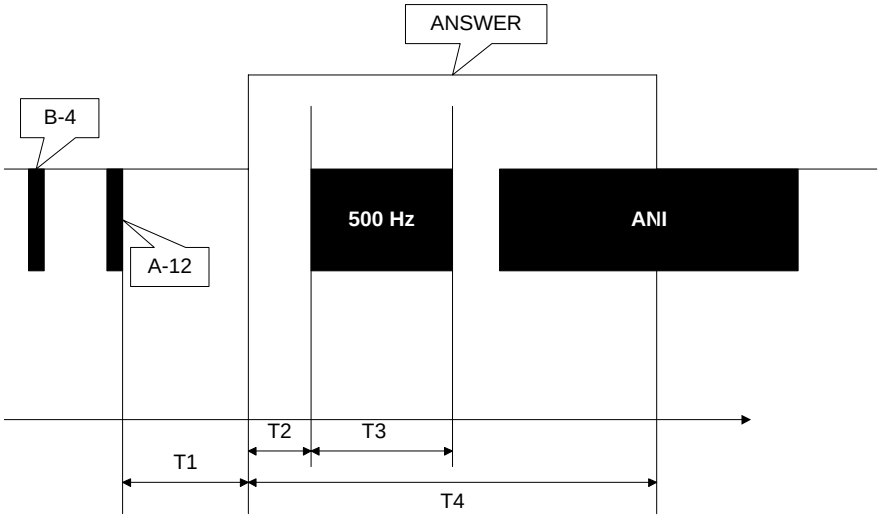



*Figure 13. Delays and Timers*

The following table represents the default values of the timers and the R1.5 parameter associated with each timer:

| Timer Number | Time (ms) | R 1.5 Parameters | Description |
|---|---|---|---|
| T1 | 2000 | adi.r15.requestdelaytime | Specifies the delay between A-12 and the ANI request line signal AB-10. |
| T2 | 230 | adi.r15.tonedelaytime | Specifies the delay between signaling answer during the ANI request and sending the 500 Hz tone. |
| T3 | 110 | adi.r15.anirequestontime | Specifies the duration of 500 Hz tone during an ANI request. |
| T4 | 500 | adi.r15.answersignalontime | Specifies the duration of line signal AB-10 (answer) when requesting ANIs. |

### 2.3.4 Algorithm Description

The calling party's number is usually 7 digits long. These digits are divided into two logical groups: the subscriber's number (4 digits, called ***'defg'*** respectively), and the switch index (3 digits, called ***'abc'*** respectively). Also, the category (*Ca*) of the calling party is transmitted. Thus, 8 informational digits are transmitted. However, some digits are repeated in a standard way in order to increase detectability, and the ANI digits are transmitted in "reverse" order. The sequence goes as follows:

| Digit | Meaning |
|---|---|
| MF-13 | Flag to signal the beginning of the sequence |
| Ca (cat) | Category of the calling party |
| E | Fourth and last digit of the calling party's number |
| D | Third digit of the calling party's number |
| C | Second digit of the calling party's number |
| T | First digit of the calling party's number |
| c | Third digit of the switch index |
| b | Second digit of the switch index |
| a | First digit of the switch index |
| MF-13 | Flag to signify the beginning of the repeat |
| Ca | Repeat of the category |
| E | Repeat of the last digit |

Since this sequence of MF tones is continuous, there is no way to detect two consecutive digits with the same value (same tone). Therefore, the second of these digits is represented with the tone MF-14 (repeated digit signal). However, when the stream arrives in reverse order, the '14' is actually in front of the digit itself.

The sequence may look as follows:

1. If the number of the calling party is '*abcdefg*' :

   {d c b a [ '13' Ca g f e d c b} a ] '13' Ca g f e …

   where the square brackets show the valid sequence and the rounded brackets show how much information is needed to decode the sequence. The minimum number of MF tones is twelve.

2. If the two first digits in the calling party's number are the same, i.e., a = b, so the number is aacdefg:

   ['13' Ca g f e d c '14' a].

3. If the category is the same as the last digit of the calling party's number, i.e., Ca = g:

   ['13' '14' g f e d c b a].

*Note:* The correct sequence does not necessarily have to start with '13'. The '13' may be anywhere within the packet.

The ANI detection algorithm implemented in the R1.5 TCP is as follows:

- Buffer all ANI digits that are sent by the switch without checking their validity.
- Check the first digit (n) in the buffer and see if it matches the tenth digit (n+9), then match second (n+1) to the ((n+1) + 9)th digit, and so on until 3 matches in a row are found.
- Search through the sequence composed of 9 digits, starting from where the first match was found and find the "13", which indicates where the category is located.
- Delete all irrelevant information from the buffer while putting all relevant information into variables.
- Reverse the order of ANIs, by putting them into the buffer from the variables in reversed order.
- Send STATUS_INFO ANI_INFO to the application, indicating the arrival of ANIs.

For example, consider a calling party's number:

457-8054, with category = 1

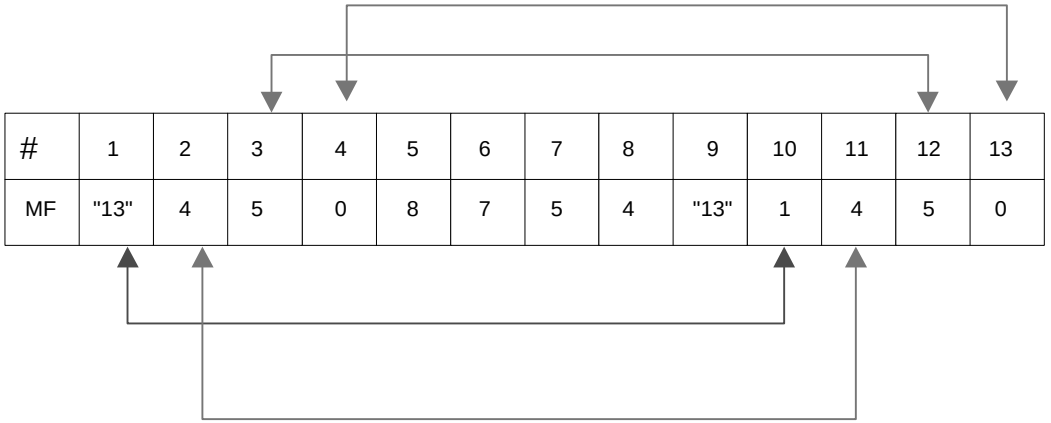



*Figure 14. Example Packet*

The two top and the lower bottom arrows show the correct ANI sequence within the packet. The first digits that will match will be the 4, then 5, then 0, so the correct packet would be extracted, starting with the 4.

### 2.3.5 Category Definitions

The category of the subscriber may have one of the following meanings:

| Category | Definition |
|---|---|
| 1 | Residential and institutional telephone with access to the automatic zone, inter-city, and international network. |
| 2 | Hotel telephone with access to the automatic zone, inter-city and international network. |
| 3 | Residential, institutional and hotel telephone with access to the local networks, without the access to the automatic inter-zone, inter-city and international network and toll services. |
| 4 | Institutional telephone with access tot the automatic zone, inter-city and international network, and toll services; priority is given to connections on inter-zone and inter-city networks. |
| 5 | Institutional telephone for Ministry of Telecommunications with access to automatic zone, inter-city and international network, and toll services; the conversations are logged but not charged. |
| 6 | Payphone and inter-city public telephone located at the telecommunications center with access to the automatic zone and inter-city network; also universal payphone with access to inter-city and local network; conversations are charged immediately; payphone for the access to toll services. |
| 7 | Residential and institutional telephone with access to the automatic zone, inter-city and international network, and toll services. |
| 8 | Institutional telephone with possibility of data transmissions, facsimiles, and electronic mail, with access to the automatic zone, inter-city and international network. |
| 9 | Local payphone. |
| 10 | Reserved. |

## 2.4 TCP Parameters

This section describes the parameters that program the TCP. The parameters are used to adapt the TCP to particular variations of R1.5, as well as to give the host application more flexibility in its interaction with the TCP. Some of these parameters are application-dependent, and can be changed at will by the user. Others (the majority) are rigidly fixed for a given network. The following table explains the parameters' meaning:

### 2.4.1 Flow Control Parameters

The following four parameters are *public*, they can be freely modified by the user:

| Parameter Name | Unit | Comments | Default Value | # |
|---|---|---|---|---|
| ADI.R15.DIDnumber | number | *Inbound side*. Specifies the number of incoming DID digits to expect. | 7 | 0 |
| ADI.R15.ANInumber | number | *Inbound side*. Number of ANI digits to expect. If this parameter is set to 0, the inbound protocol does not attempt to request the ANIs. If this parameter is set to any other number, the inbound protocol will request the ANIs, and will receive as many as the outbound side will provide. | 8 | 1 |

| Parameter Name | Unit | Comments | Default Value | # |
|---|---|---|---|---|
| ADI.R15.optionflags | mask | *Inbound and outbound sides.*<br>Flags controlling optional TCP behavior:<br>(MF-Pulse Shuttle protocol is the default register signaling for both inbound and outbound protocols and can be combined or replaced with some other types of register signaling)<br>• `0x1`: Set this bit to do decadic dialing (for both inbound and outbound protocols).<br>• `0x2`: Set this bit for long distance inbound protocol.<br>• `0x4`: Set this bit to make the transmission of ANIs mandatory (relevant to the outbound protocol).<br>• `0x8`: Set this bit to make the reception of ANIs mandatory (relevant to the inbound protocol).<br>• `0x10`: (default) Play busy on timeout.<br>• `0x20`: Play fast busy on timeout.<br>• `0x40`: Play silence on timeout.<br>These values can be Ored for cumulative effect. | 0x20 | 2 |
| ADI.R15.debugmask | mask | Specifies what trace messages are generated (run *`agtrace 1000`* or higher to see them):<br>• `0x01`: Show all states as they are entered.<br>• `0x02`: Show sent and received digits.<br>• `0x04`: Show signaling bits.<br>• `0x08`: Print a description of timeout related errors.<br>These values can be Ored for cumulative effect. | 0x0 | 3 |

| Parameter Name | Unit | Comments | Default Value | # |
|---|---|---|---|---|
| ADI.R15.firstcommand | count | *Inbound side*. First command for inbound protocol.<br>Acceptable values: 1, 2, or 3.<br>If the value is set to 2 or 3, make sure that the ADI.R15.DIDnumber is set to a correct value. | 1 | 4 |
| ADI.R15.anirequestattempts | count | *Inbound side*. Specifies the number of attempts to obtain ANIs. If the parameter is set to 0, no attempts will be made.<br>When the reception of ANIs is mandatory (bit 0x8 in ADI.R15.optionflags is set) and no ANIs are received after the first attempt, this parameter can be used to perform additional ANI requests before clearing back. | 1 | 5 |
| ADI.R15.dialstartposition | count | *Outbound side*. Used to define the starting position within a dial string; i.e. when the first request from the inbound side, defined by ADI.R15.firstcommand, is B2 (send next digit) or B3 (send the previous digit) the first DID that is presented by the outbound side is located in the ADI.R15.dialstartposition position. | 4 | 6 |
| ADI.R15.anionebyone | mask | Determines whether or not the TCP will send ANIs one at a time:<br>• `0x1`: enables the detection of ANIs one by one. | 0x0 | 7 |

### 2.4.2 Flow Control Timers

The following parameters are *reserved*, they should not be modified by the user, since this would affect regulatory compliance in the target country.

| Parameter Name | Unit | Comments | Default Value | # |
|---|---|---|---|---|
| ADI.R15.qualtime | ms | Specifies the qualification time for bit changes. | 20 | 8 |
| ADI.R15.waitbegintime | ms | Specifies the delay before starting to play the forward and backward tones. | 60 | 9 |

### 2.4.3 Outbound Timers

| Parameter name | Unit | Comments | Default value | # |
|---|---|---|---|---|
| ADI.R15.seizetime | ms | Specifies the maximum time to receive the seizure acknowledge. | 400 | 10 |
| ADI.R15.waitbacksignaltime | ms | Specifies the maximum time to receive a backward request. | 4000 | 11 |
| ADI.R15.pausefrstdigittime | ms | Specifies the initial wait before dialing the first decadic pulse. | 300 | 12 |
| ADI.R15.interdigittime | ms | Specifies the time between two trains of pulses while dialing with decadic pulses. | 750 | 13 |
| ADI.R15.waitechotime | ms | Specifies the echo cancellation time. – NOT USED (see adi.r15.toneofftime). | 40 | 14 |
| ADI.R15.confirmanswertime | ms | Specifies the timeout for differentiating answer vs. ANI request which is accompanied by the 500 Hz tone. | 275 | 15 |

### 2.4.4 Inbound Timers

| Parameter Name | Unit | Comments | Default Value | # |
|---|---|---|---|---|
| ADI.R15.waitfrwdsignaltime | ms | Specifies the maximum time to receive a forward signal. | 350 | 16 |
| ADI.R15.waitfrwddeksignaltime | ms | Specifies the maximum time to receive a decadic digit. | 4000 | 17 |
| ADI.R15.inbinterdigittime | ms | Specifies the time to wait before deciding that a train of decadic pulses is over. | 200 | 18 |
| ADI.R15.waitforPCtime | ms | Specifies the time for the TCP to wait for the host to answer or reject the call. | 10000 | 19 |
| ADI.R15.anitonetimeout | ms | Specifies the time to wait before deciding that the transmission of ANIs is over. | 100 | 20 |
| ADI.R15.anirequesttimeout | ms | Specifies the timeout to get the first ANI. | 1000 | 21 |

### 2.4.5 Dialing Control

| Parameter Name | Unit | Comments | Default Value | # |
|---|---|---|---|---|
| ADI.R15.toneontime | ms | Specifies the duration of the MF tone used to convey DID information (backward and forward). | 45 | 22 |
| ADI.R15.toneofftime | ms | Specifies the silence between the MF tones used to convey DID information (backward and forward) – also used for echo cancellation. | 45 | 23 |
| ADI.R15.pulseontime | ms | Relevant to the outbound protocol. Specifies the time a pulse should be ON while dialing with decadic pulses. | 50 | 24 |

| Parameter Name | Unit | Comments | Default Value | # |
|---|---|---|---|---|
| ADI.R15.pulseofftime | ms | Relevant to the outbound protocols. Specifies the time a pulse should be OFF while dialing with decadic pulses. | 50 | 25 |
| ADI.R15.anitoneontime | ms | Relevant to outbound protocols. Specifies the duration of the MF tone used to convey ANI information. | 40 | 26 |
| ADI.R15.toneslevel | IDU | Specifies the amplitude of MF (call setup) tones (forward and backward). | 315 | 27 |

## 2.4.6 Ring / Busy Control

| Parameter Name | Unit | Comments | Default Value | # |
|---|---|---|---|---|
| ADI.R15.ringfreq | Hz | Relevant to inbound side protocols. Specifies the ring tone frequency. | 425 | 28 |
| ADI.R15.ringontime | ms | Relevant to inbound protocols. Specifies the time the ring tone is on in a ring cycle. | 1000 | 29 |
| ADI.R15.ringofftime | ms | Relevant to inbound protocols. Specifies the time the ring tone is off in a ring cycle. | 3000 | 30 |
| ADI.R15.busyfreq | Hz | Relevant to inbound protocols. Specifies the busy tone frequency. | 425 | 31 |
| ADI.R15.busyontime | ms | Relevant to inbound protocols. Specifies the time the busy tone is on in a busy cycle. | 300 | 32 |
| ADI.R15.busyofftime | ms | Relevant to inbound protocols. Specifies the time the busy tone is off in a busy cycle. | 300 | 33 |
| ADI.R15.fastbusyontime | ms | Relevant to inbound protocols. Specifies the time the congestion tone is on. | 150 | 34 |
| ADI.R15.fastbusyofftime | ms | Relevant to inbound protocols. Specifies the duration of the first period of time the congestion tone is off. | 150 | 35 |
| ADI.R15.Cptoneslevel | IDU | Specifies the amplitude of call progress tones. | 315 | 36 |

## 2.4.7 Miscellaneous

| Parameter Name | Unit | Comments | Default Value | # |
|---|---|---|---|---|
| ADI.R15.resourcerequesttime | mask | Specifies the time to wait for resources. | 0x0a0f | 37 |

## 2.5 Call Setup Diagrams

This scction presents diagrams that illustrate the functions of several R1.5 protocols in local and long distance circumstances.

### 2.5.1 Local Inbound and Outbound Protocols

As shown in Figure 15, the local inbound protocol is symmetrical to the local outbound protocol, so they are depicted in the same diagram. The line signaling used is Group 1.

Refer to the table in Section 2.2.1, *R1.5 Line Signaling – Group 1*.

The register signaling is Pulse Shuttle, but the TCP can also switch to decadic dialing.

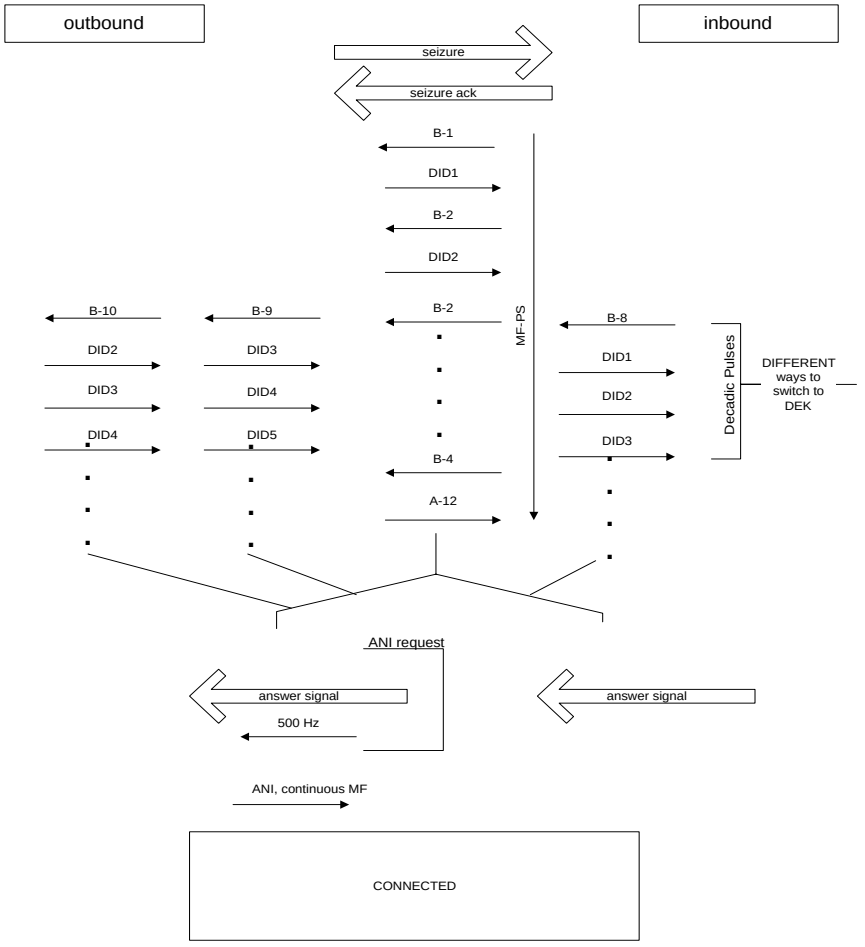



*Figure 15. Local Inbound and Outbound Protocols*

## 2.5.2 Long Distance Outbound Protocol

### Switching from MF-PS to DEK

As shown in Figure 16, the register signaling starts out as Pulse Shuttle, but when the first calling address digit is detected to be digit “8” (which is known to be the

access to the long distance switches), the inbound side sends the end of call setup tones and requests ANIs. Only after ANIs have been received, the DID phase returns, but in the form of Decadic Dialing.

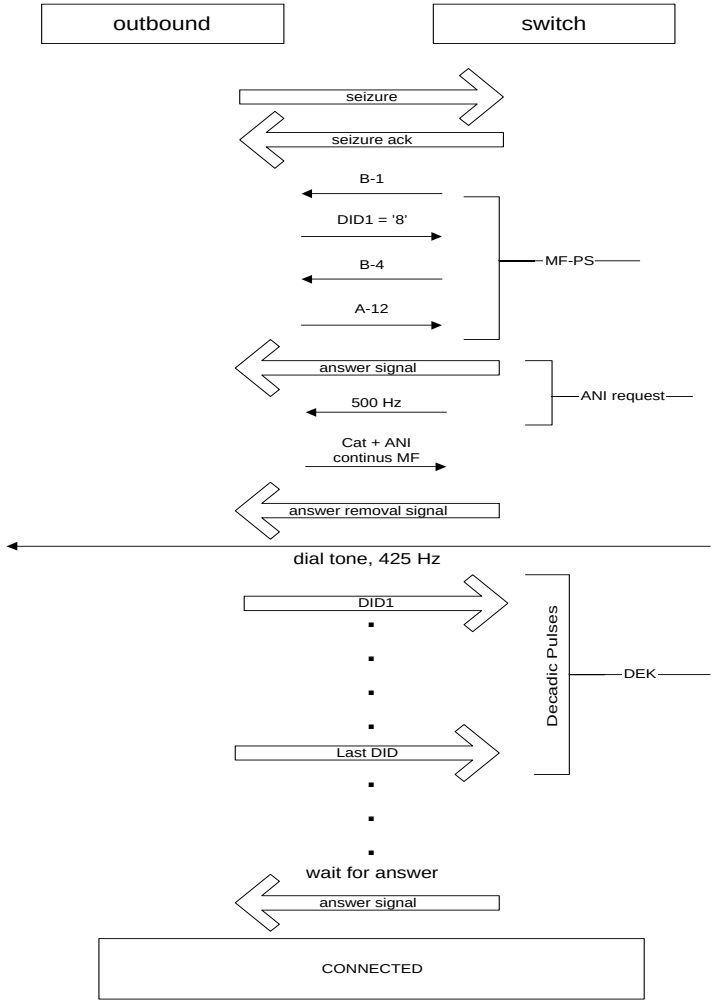



*Figure 16. Switching From MF-PS to DEK*

## Decadic Register Signaling for Call Setup

As shown in Figure 17, the first expected command from the switch is an ANI request. Only after the ANIs have been received, the call proceeds with the DIDs as decadic pulses.

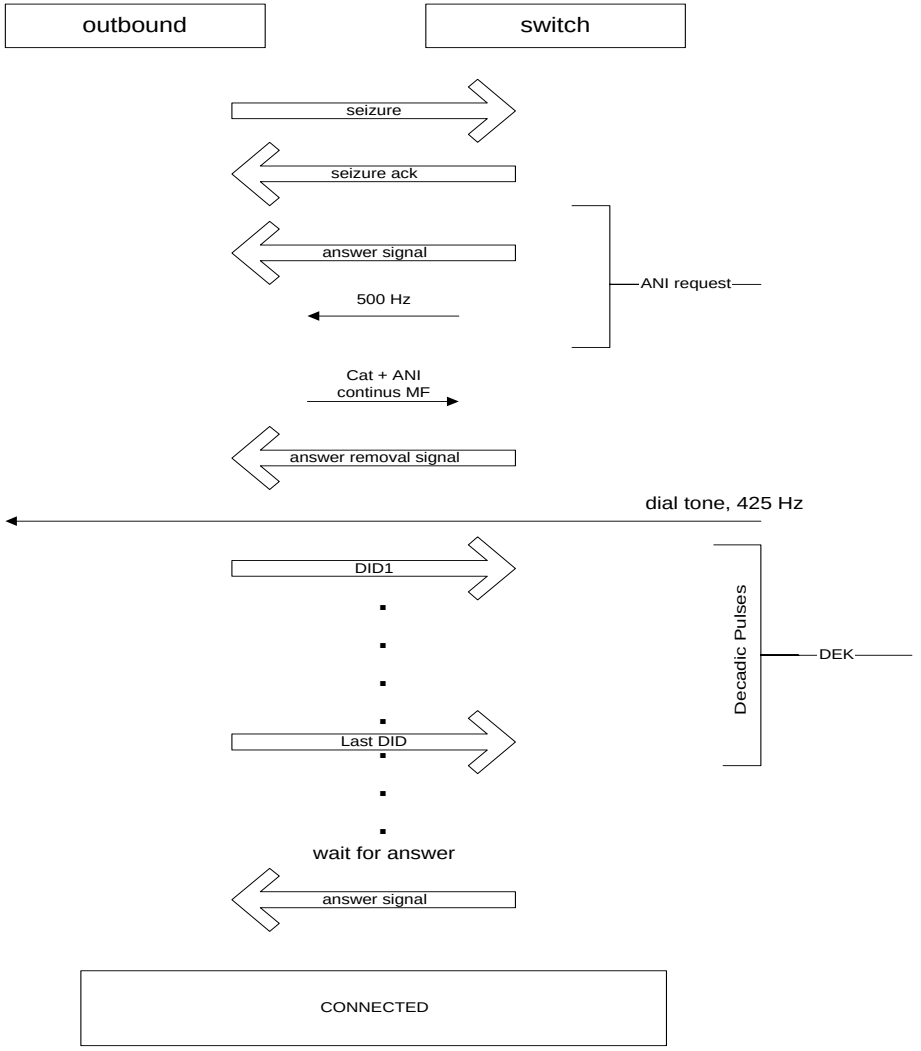



*Figure 17. Register Signaling for Call Setup*

## 2.5.3 Long Distance Inbound Protocol

Long distance inbound protocol differs only slightly from the inbound local protocol. One difference is line signaling. Long distance protocol uses line signaling group 2 (see the table in Section 2.2.2, *R1.5 Line Signaling – Group 2*). The second difference is that the long distance protocol does not have the ability to ask for ANIs.

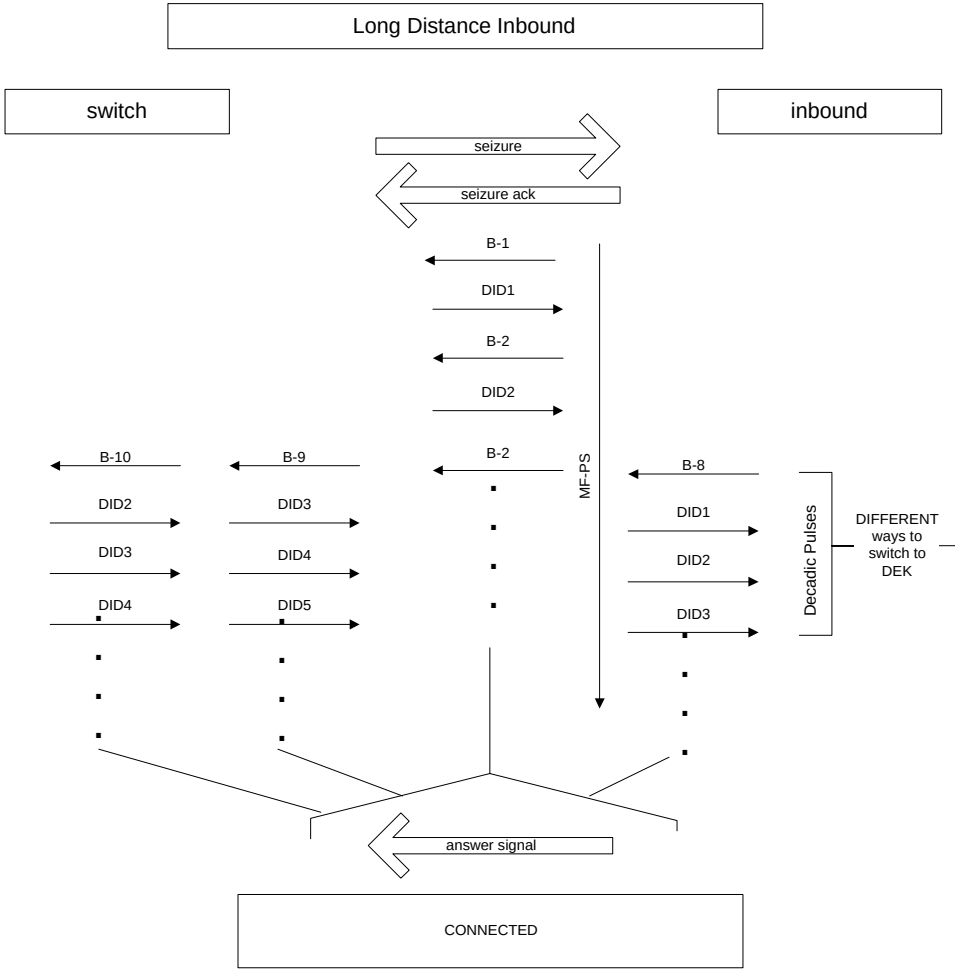


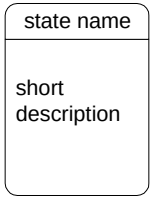
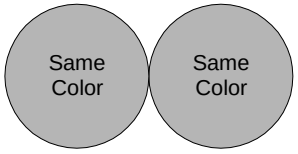

*Figure 18. Long Distance Inbound Protocol*

## 2.6 State diagrams

This section contains the state diagrams of the System R1.5 TCP. States that interface with the line or the host are shown, grouped in logical groups if appropriate. States in the TCP script that simply execute instructions without receiving input from the external world (pass-through states) are omitted.

*Note:* Although it is not shown in the diagrams, every state handles error conditions.

The following symbols are used in the diagrams:

| Description | Icon |
|---|---|
| TCP State |  |
| Reference to a state on another diagram |  |
| Reference to a state *from* another diagram |  |
| Link between states |  |
| Reference to a state within same page |  |
| Grouping of states of the same protocol |  |

### 2.6.1 Inbound and Outbound TCP States

Natural MicroSystems's R1.5 protocol package consists of two TCPs, one performing the functions of the inbound protocol (*r150.tcp*) and the second performing the functions of the outbound protocol (*r151.tcp*). Only some of the initial, resource management and connected states are shared by the two protocols. Some of the following diagrams will relate to both TCPs at the same time, and others purely to inbound or outbound protocols.

#### Startup and Idle States

Figure 19 depicts the common states of the R150 and R151 TCPs that implement the startup phase for the inbound and the outbound TCPs, and the idle condition. The TCP determines whether it is an inbound or an outbound trunk and sets the particular idle code depending on the direction of the trunk. When in idle, the outbound TCP waits for a command to place a call, and the inbound TCP waits for a seizure on the line. In both cases, the resource management state is entered next. If resource management is not necessary for the type of board being used, the TCP immediately transitions the following state. If resource management is necessary, a call setup resource is requested and the state waits for it to be granted before continuing.

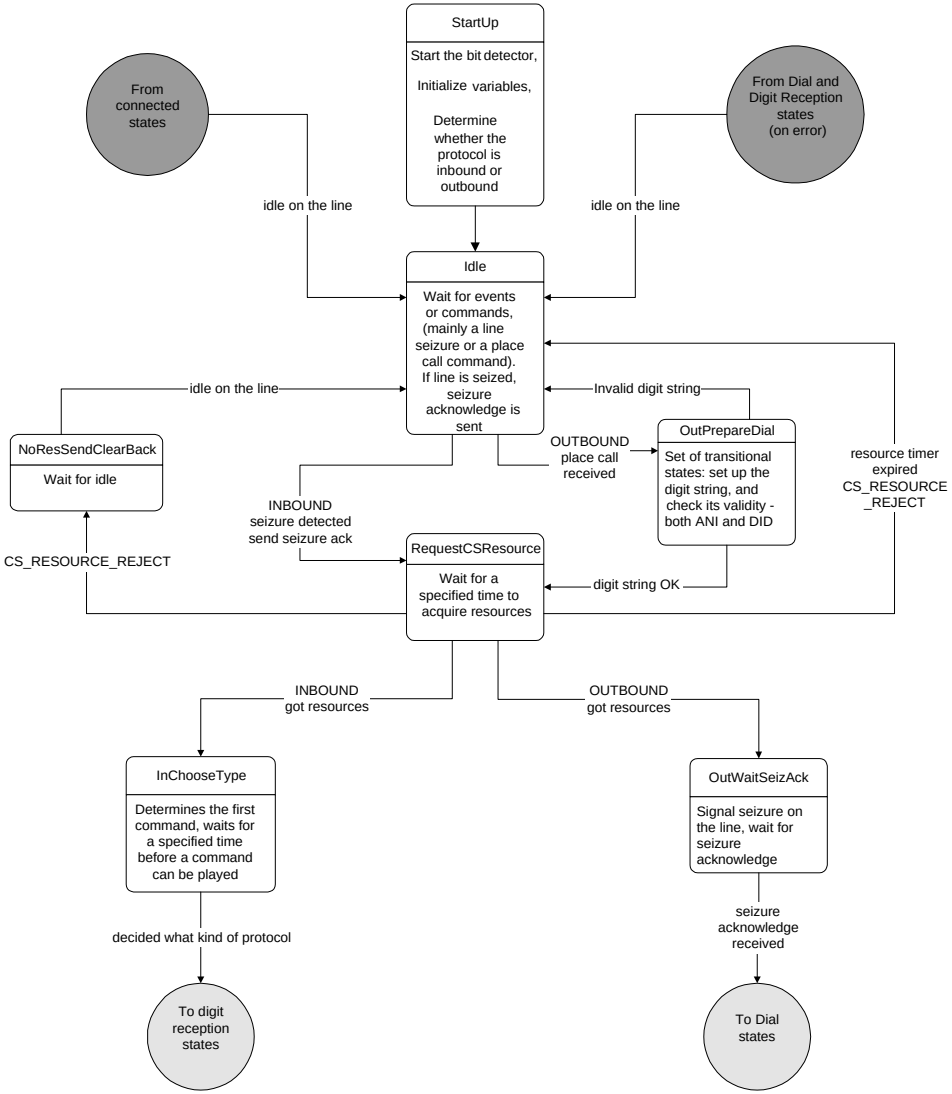



*Figure 19. Startup and Idle*

## Overview of the Outbound Dial States

Figure 20 relates only to the outbound protocol. The DID portion of call setup is performed either by decadic pulses or with Multi-Frequency Pulse Shuttle. An ANI request may come at any time during the DID sequence. In this case, the outbound side must present ANIs to the inbound protocol.

The host can request to perform call progress detection before declaring the call connected. Call progress is used to connect on a particular condition (i.e., voice detected, SIT tone detected…); it may also be used to distinguish an accepted from a rejected call, if the answer signal was never removed after the ANI sequence.

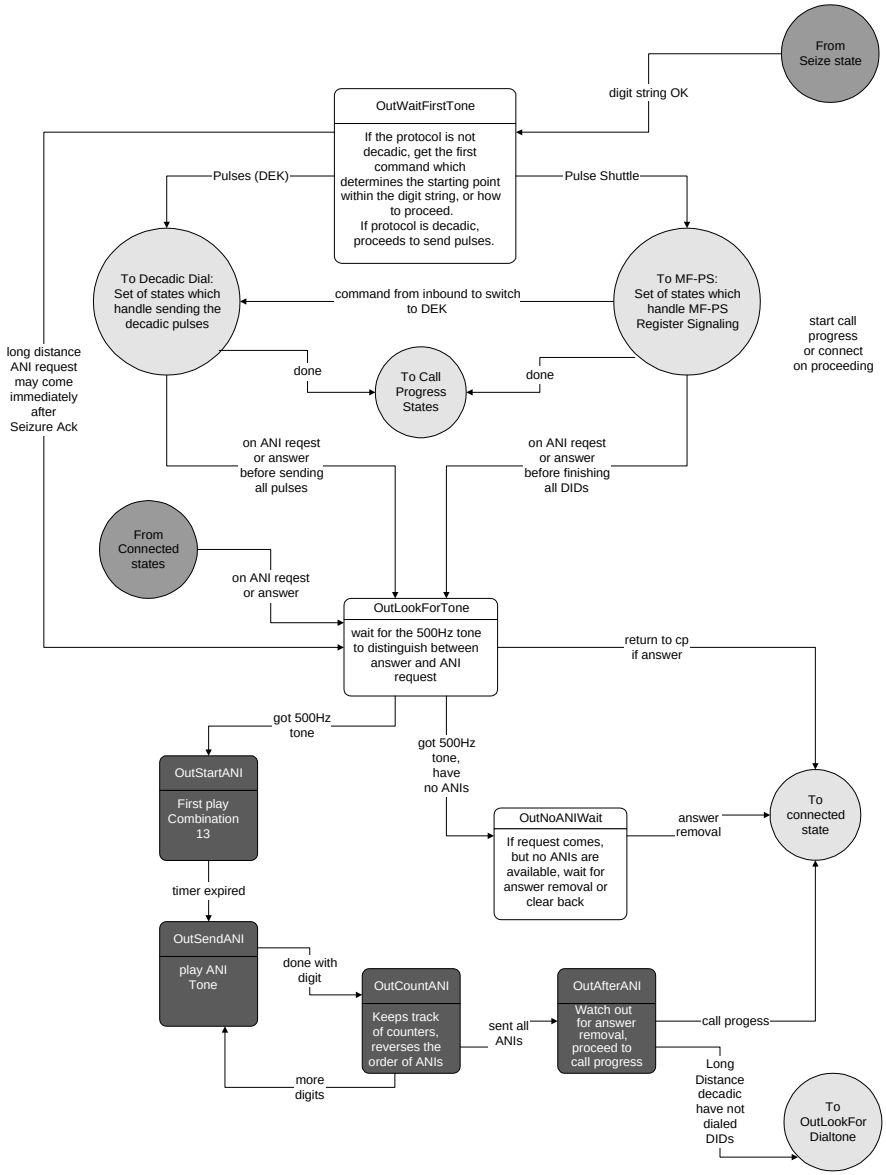



*Figure 20. Outbound Dial States*

## Outbound Decadic Dial States

Decadic Dialing is often used as a method of register signaling. It is used with switches which do not have the MF capability.

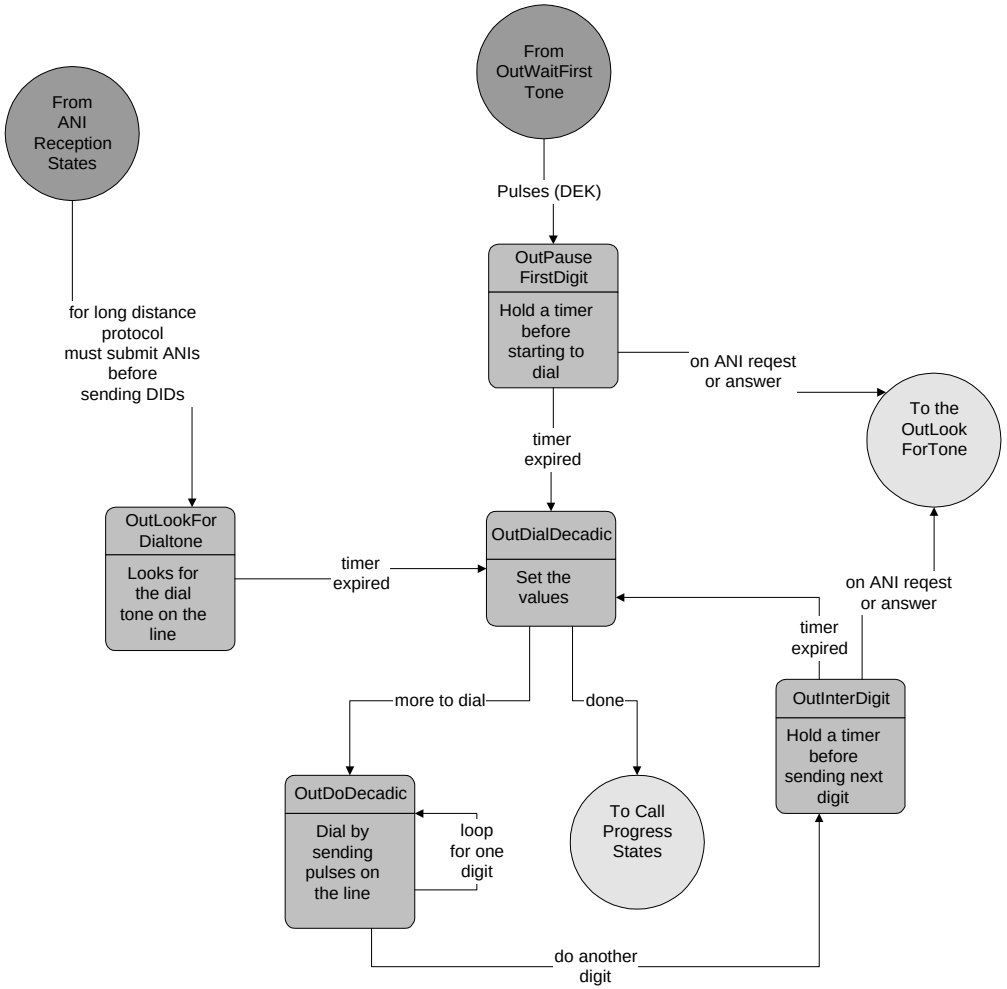



*Figure 21. Outbound Decadic Dial States*

## Outbound Pulse Shuttle Dial States

The outbound side called number presentation sequence proceeds from one digit to the next. Backward requests from the inbound side are analyzed, and depending on their meanings, a particular forward digit is played next, or the DID sequence is aborted if an error or a backward congestion indication is detected.

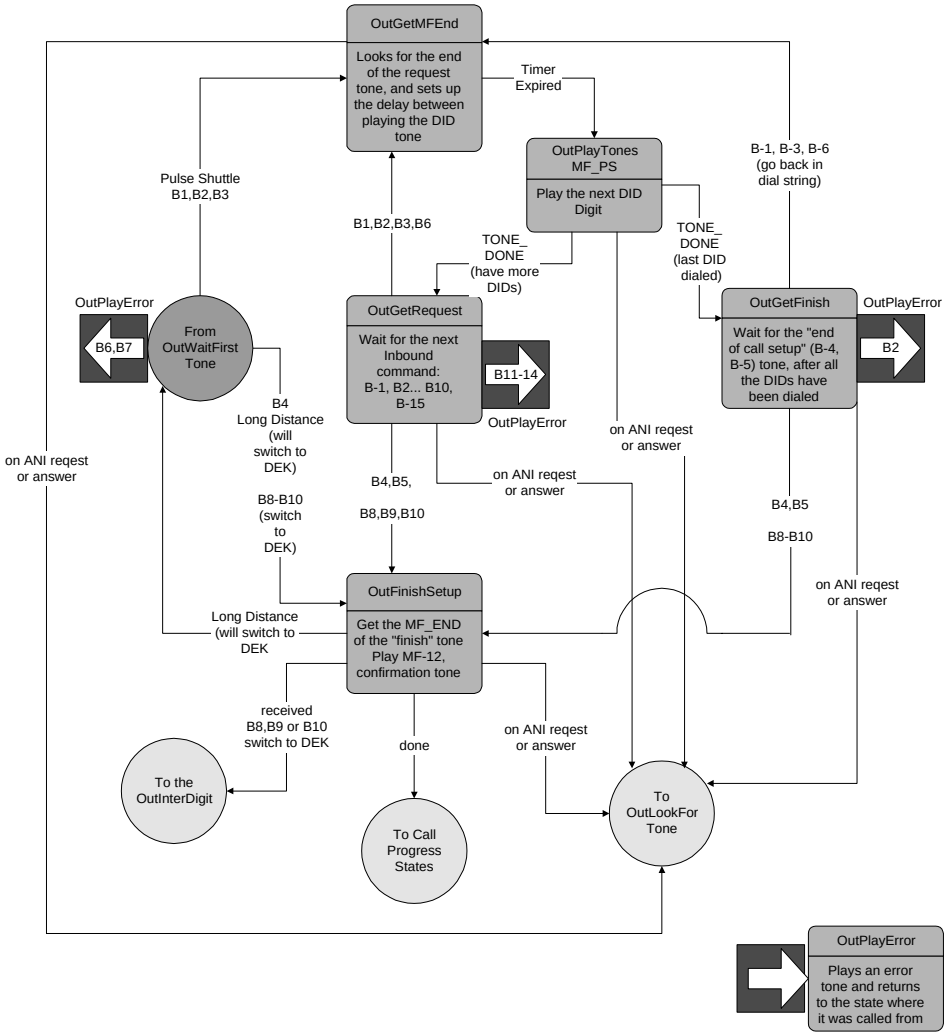



*Figure 22. Outbound Pulse Shuttle Dial States*

## Inbound Digit Reception States

In the protocol where the inbound side is the master, it is very important to parameterize the commands that the inbound protocol must send to the outbound side. The following states show the two possible ways to accept the called party's address information: The Multi-Frequency Pulse Shuttle, or Decadic Pulses. Which one of the methods is more appropriate depends on the local switch.

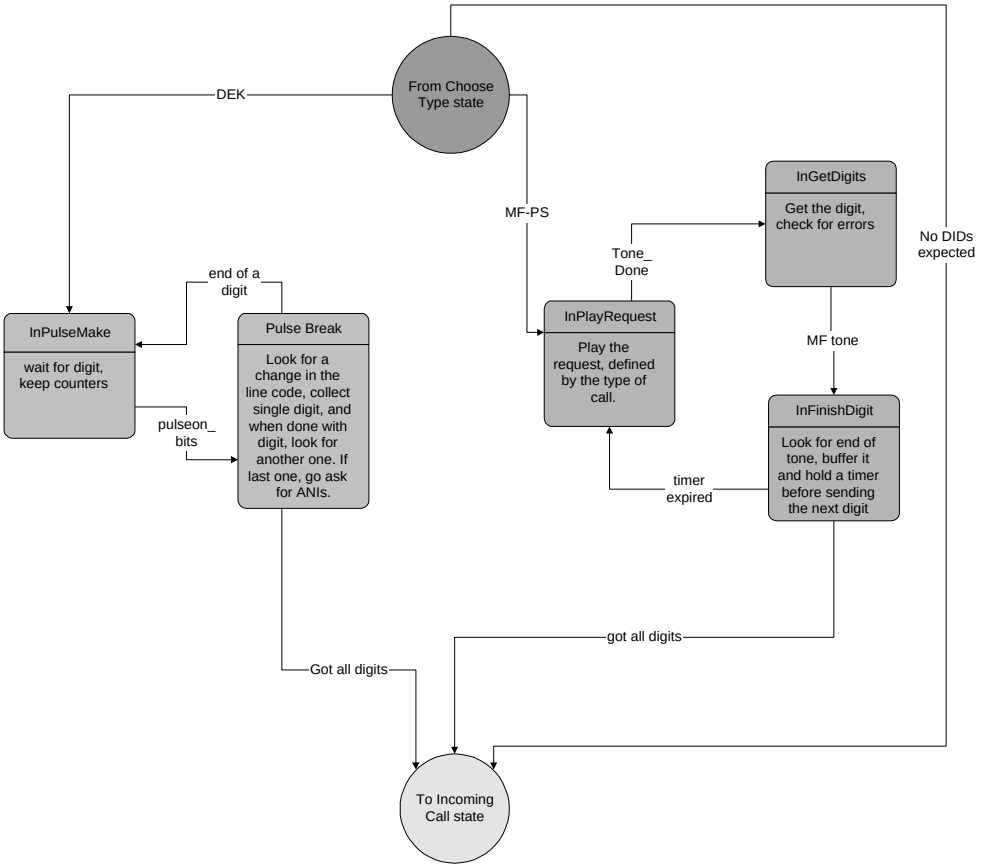



*Figure 23. Inbound Digit Reception States*

## Inbound Accept, Answer and Reject States

In the *Incoming Call* state, the TCP sends to the host an INCOMING_CALL event, with the received DID digits attached. The host must then decide if the call is to be:

- Accepted (without committing to answer it): **adiAcceptCall**, seen by the TCP as an ACCEPT_CALL command
- Answered: **adiAnswerCall**, seen by the TCP as an ANSWER_CALL command, or
- Rejected: **adiRejectCall**, seen by the TCP as a REJECT_CALL command

If the register signaling is done by MF-PS, this tells the TCP which backward tone to play to terminate this phase of call setup.

Once the host has told the TCP what to do with the call, the TCP plays an "end of call setup" tone (B-4, B-5, or B-7), that accepts or rejects the call with different reasons and modes. Depending on the host's command, the TCP then does one of the following:

| If the call needs to be: | Then the: |
|---|---|
| Rejected | TCP plays busy or lets the host play a reject message, until the call is cleared by the calling party, if the call has been rejected. |
| Accepted or answered | Local inbound protocol makes an ANI request. |
| Accepted or answered (long distance inbound protocol) | Long distance inbound protocol sends a selection free line signal. This tells the local switch to play ring tone and then answer if the host committed to answer the call (or commanded to accept the call) and play ring indefinitely. |

After the ANI detection phase for the local protocol, the answer line code is removed. Depending upon whether the call needs to be accepted or answered, the local inbound TCP then reacts in the following ways:

| If the... | Then the TCP... |
|---|---|
| Host committed to answer the call or commanded to accept the call and play ring indefinitely | Plays ring tone and then answers. |
| Call has been provisionally accepted | Stays silent or allows the host to play a custom message. |

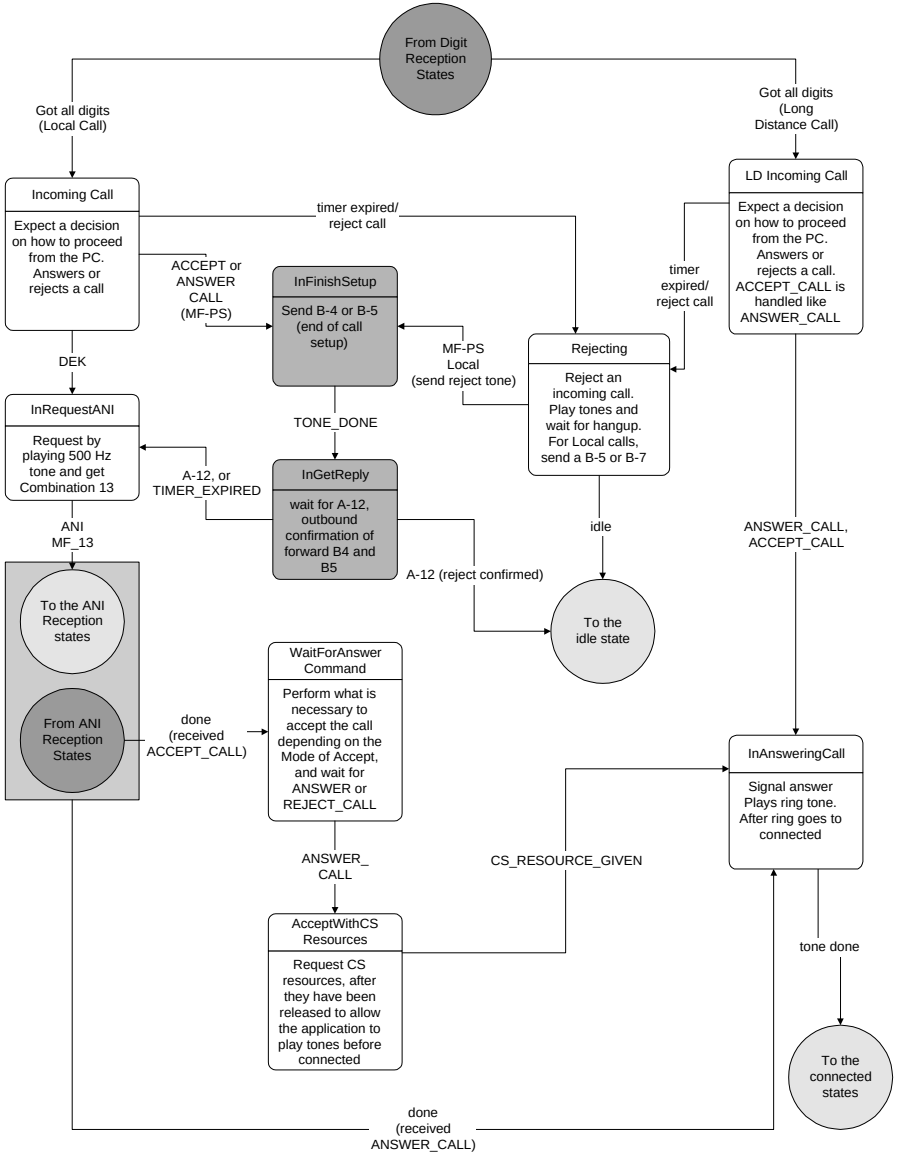



*Figure 24. Inbound Accept, Answer, and Reject States*

## Inbound ANI Reception States

After the call had been either answered or accepted, the local inbound protocol makes an ANI request, by setting the answer line signaling code and sending a pure 500 Hz tone.

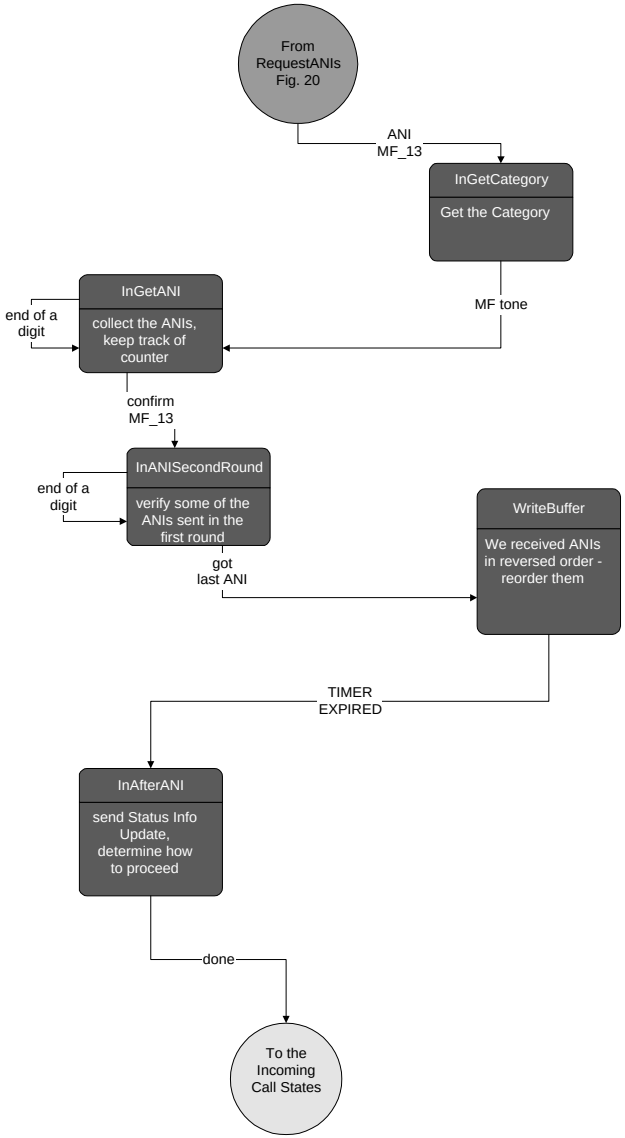



*Figure 25. Inbound ANI Reception States*

## Connected States

Figure 26 shows the common inbound and outbound connected and call progress states. After the call had been answered, the application may request to perform call progress. From this point on, the TCP expects a hang-up. An unusual situation arises for long distance inbound calls when the called party hangs up first. In this case, there is a chance of a recall, when the calling party (the operator) reestablishes the connection. In the case of a clear back, the inbound side waits not only for the hang-up line code from the outbound side but also for another bit pattern indicating recall. If this happens, the inbound side reestablishes the connection, while in a regular case it would clear the line.

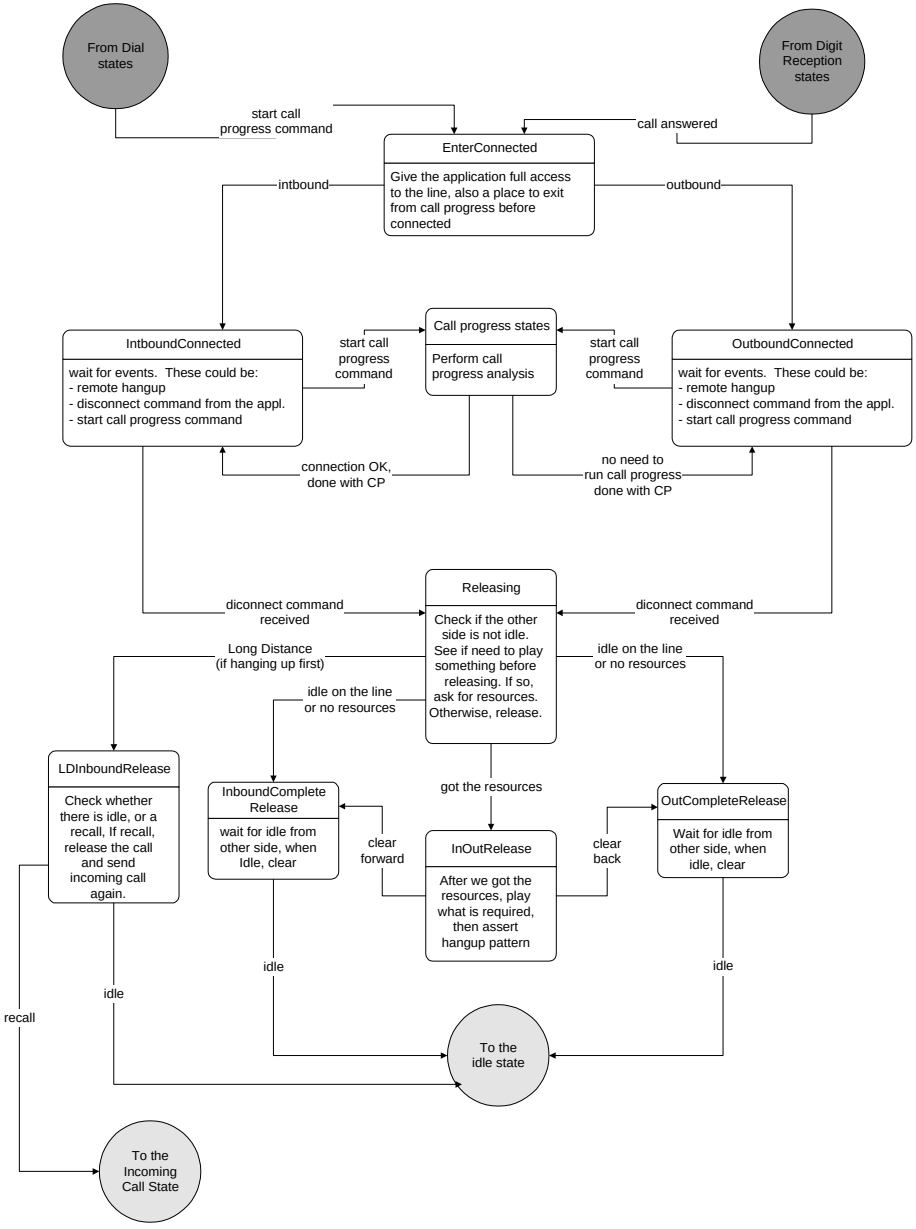



*Figure 26. Connected States*

## 2.7 TCP performance

TCP performance is the measure of:

- How fast a call can be set up, either inbound or outbound
- The DSP resources necessary to set up a call
- The load limits, i.e., the number of calls that can be setup simultaneously for each board

To analyze the call setup timing, the signal on the line is captured and recorded. The spectral distribution of the energy on the line is then analyzed, and the timing is deduced. A model of the timing is also produced, to explain the observations in the light of the AG architecture. In general, the timing results will be different if the TCP is run “back-to-back” or is talking to a switch. With MFC-R2, switches are usually faster than the TCP, so the call setup timing will generally be shorter if one of the sides is a network switch.

The resources necessary to set up the call differ for different protocols. They depend on the signaling method in use. Diagrams are included, plotting MIPS consumption against time, per channel. The timing of the diagram is derived from the previous time analysis and the TCP code, identifying the point in the TCP where DSP functions are started and stopped.

The number of calls that can be set up simultaneously depends on:

- The TCP MIPS consumption
- The type of board the TCP runs on
- Any task processor optimization carried out in the AG configuration file

Thus, the matter is very complex, and only some general calculations and guidelines are reported in the following sections.

### 2.7.1 Optimal Task Processor Configuration

The R1.5 protocol is highly unusual, partly because it uses A-law US-MF tones to send tones in forward and backward directions, thus requiring both the MF Generator to be active all the time as well as the MF Detector. Both the detector and the generator have high MIPS consumption. This is not presumed by the default assignment of DSP files to task processors on the board. Therefore, the default configuration allows only 23 ports to be active with this protocol on an AG

E1 board (57 MHz DSPs). To have better performance, task processors must be explicitly specified.

The following table shows the DSP files that are most likely to be necessary to run R1.5 protocol:

| DSP file | Usage |
|---|---|
| *tone_a.dsp* | Beep, tone generation, dial |
| *callp_a.dsp* | Call progress detection |
| *signal_a.dsp* | Out-of-band channel associated signaling (Not used on Dual/Quad) |
| *dtmf_a.dsp* | DTMF, cleardown and silence/energy detectors |
| *voice_a.dsp* | NMS ADPCM play and record |
| *mf_a.dsp* | MF detection |

Below is the most optimal task processor assignment for the protocol. It must be included in the AG configuration file for AG E1 boards to perform as desired. Also specified are the assignments of DSP files to task processors on the board:

- `taskprocessor 0..2 = callp_a.dsp,tone_a.dsp,voice_a.dsp`
- `taskprocessor 3 = signal_a.dsp,dtmf_a.dsp,mf_a.dsp`
- `taskprocessor 4..5 = mf_a.dsp,dtmf_a.dsp`

This is the allocation for the AG QUAD E board and is the R1.5 equivalent to `CCMODE = HIGH`. It is used for applications with very little activity in the connected state (ideally none, like trunk-to-trunk switching applications). It gives a maximum of 120 ports of simultaneous call setup per board, but provides a very limited processing power for the application while the call is connected:

- `taskprocessor 0..2 = callp_a.dsp,tone_a.dsp,voice_a.dsp`
- `taskprocessor 3 = dtmf_a.dsp,mf_a.dsp`
- `taskprocessor 4..5 = mf_a.dsp,dtmf_a.dsp`
- `taskprocessor 6..7 = callp_a.dsp,tone_a.dsp,voice_a.dsp,mf_a.dsp,dtmf_a.dsp`

## 2.7.2 Call Setup Resource Diagram

The following figures represent the MIPS consumption and the kind of DSP resources needed by the R1.5 TCPs to place and receive a call, plotted against time. The register signaling consists of seven DID digits, category digit and seven ANI digits, inbound and outbound.

1. AG-E1. In this case, the DSP overlay to receive / transmit bits signaling is on. The DSP overlays for US MF tone detection and tone generation are started in the beginning of call setup, and are kept turned on until connected, for both, inbound and outbound. This is illustrated in Figure 27:

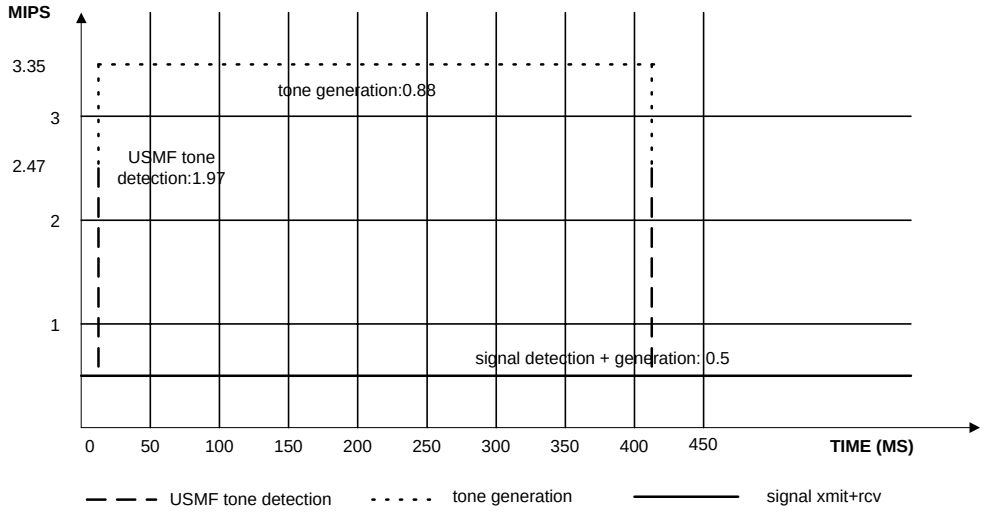



*Figure 27. AG-E1 with DSP Overlay Signaling On*

2. AG-E1. The signaling bits detection / generation function is carried out by the 486 coprocessor directly, and no DSP overlay is required for the task. This reduces the DSP load by 0.5 MIPS, as shown in Figure 28. The DSP overlays for US MF tone detection and tone generation are started in the beginning of call setup, and are kept turned on until connected, for both, inbound and outbound.

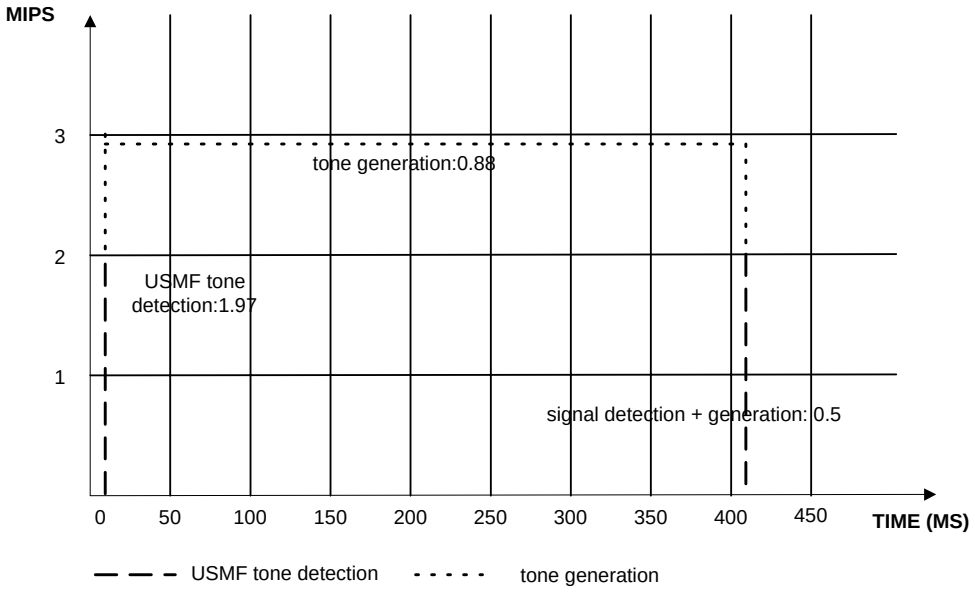



*Figure 28. AG-E1 486 Coprocessor Signaling*

## 2.7.3 Behavior Under Load

The PROTtest utility can be used to test the protocol's behavior under load. It spawns a number of threads, each either placing or receiving calls. Then it collects statistics and errors from the threads. (For more information about PROTtest, refer to the *PROTtest User's Manual*.) The following are examples of PROTtest log files for AG-E1 and AG Quad E boards, running at full capacity.

## Load Test Results: Two AG-E1 Boards, One AG Quad E Board

In this test, two AG-E1 boards and one AG Quad E board are connected back to back. The AG Quad E places calls to the two AG-E1s on all their available channels (60 channels in total). Statistics are gathered every 10 minutes.

```
================================================================================
=
ProtTest - CTA protocols tester V.2.0 (Feb 16 1999) - inbound side
test start time = Wed Feb 17 13:18:41 1999
arguments = -i -fconf4
board: AG-E1, MIPS available: 168
================================================================================
=
================================================================================
=
ProtTest - CTA protocols tester V.2.0 (Feb 16 1999) - outbound side
test start time = Wed Feb 17 13:19:04 1999
arguments: dialstring #12345678#1234567
board: AG Quad E, MIPS available: 400
================================================================================
=
stat:          INBOUND time = 010 minutes
           attempted inbound calls..... 550
           successful inbound calls..... 492
           calls with no resource available for call setup.. 0
           calls with no resource available for application. 0
           calls with an out-of-service event..... 0
           calls with an in-service event..... 0
stat:          OUTBOUND time = 010 minutes
           attempted outbound calls..... 1522
           successful outbound calls..... 1507
           calls with no resource available for call setup.. 0
           calls failed because inbound had no CS resource.. 0
           calls with no resource available for application. 0
           calls with no resource available for CP..... 0
           calls with an out-of-service event..... 0
           calls with an in-service event..... 0
stat:          INBOUND time = 020 minutes
           attempted inbound calls..... 2364
           successful inbound calls..... 2305
           calls with no resource available for call setup.. 0
           calls with no resource available for application. 0
           calls with an out-of-service event..... 0
           calls with an in-service event..... 0
stat:          OUTBOUND time = 020 minutes
           attempted outbound calls..... 3337
           successful outbound calls..... 3320
           calls with no resource available for call setup.. 0
           calls failed because inbound had no CS resource.. 0
```

```
           calls with no resource available for application. 0
           calls with no resource available for CP.......... 0
           calls with an out-of-service event............... 0
           calls with an in-service event................... 0
-----------------------------------------------------------
CASmon - CTA Digital CAS protocols tester V.1(Apr 28 1999)
printouts omitted - no errors
----------------------------
stat:          INBOUND time = 060 minutes
           attempted inbound calls.......................... 9700
           successful inbound calls......................... 9641
           calls with no resource available for call setup.. 0
           calls with no resource available for application. 0
           calls with an out-of-service event............... 0
           calls with an in-service event................... 0
stat:          OUTBOUND time = 060 minutes
           attempted outbound calls......................... 9702
           successful outbound calls........................ 9681
           calls with no resource available for call setup.. 0
           calls failed because inbound had no CS resource.. 0
           calls with no resource available for application. 0
           calls with no resource available for CP.......... 0
           calls with an out-of-service event............... 0
           calls with an in-service event................... 0
-------------------------------------------------
test successfully terminated
-------------------------------------------------
```

## Load Test Results: One AG Quad E Board (Back to Back)

In this test, one AG Quad E board places calls to itself. The first two trunks of the board receive calls, the second two trunks place calls. Resource management is used in this case, but the application is inert in the connected phase. This simulates a gateway application, that only switches calls through other boards / trunks, without playing or recording voice files, or detecting DTMFs. Statistics are gathered every 10 minutes.

```
================================================================================
=
ProtTest - CTA protocols tester V.2.0 (Feb  8 1999) - inbound side
test start time = Wed Feb 10 16:38:28 1999
arguments = -i -fconf7
board: AG Quad E, MIPS available: 400
================================================================================
=
================================================================================
=
ProtTest - CTA protocols tester V.2.0 (Feb  8 1999) - outbound side
test start time = Wed Apr 28 10Feb 10 16:38:44:01 29 1999
arguments = -fconf7
board: AG Quad E, MIPS available: 400
================================================================================
=
stat:   INBOUND time = 010 minutes
    attempted inbound calls..... 1242
    successful inbound calls..... 1190
    calls with no resource available for call setup.. 0
    calls with no resource available for application. 0
    calls with an out-of-service event..... 0
    calls with an in-service event..... 0
stat:   OUTBOUND time = 010 minutes
    attempted outbound calls..... 1250
    successful outbound calls..... 1235
    calls with no resource available for call setup.. 0
    calls failed because inbound had no CS resource.. 0
    calls with no resource available for application. 0
    calls with no resource available for CP..... 0
    calls with an out-of-service event..... 0
    calls with an in-service event..... 0
stat:   INBOUND time = 020 minutes
    attempted inbound calls..... 2479
    successful inbound calls..... 2423
    calls with no resource available for call setup.. 0
    calls with no resource available for application. 0
    calls with an out-of-service event..... 0
    calls with an in-service event..... 0
```

```
stat:   OUTBOUND time = 020 minutes
    attempted outbound calls..................... 2484
    successful outbound calls.................... 2463
    calls with no resource available for call setup.. 0
    calls failed because inbound had no CS resource.. 0
    calls with no resource available for application. 0
    calls with no resource available for CP..... 0
    calls with an out-of-service event.............. 0
    calls with an in-service event.................. 0
----------------------------
printouts omitted - no errors
----------------------------
stat:   INBOUND time = 120 minutes
    attempted inbound calls...................... 14586
    successful inbound calls..................... 14527
    calls with no resource available for call setup.. 0
    calls with no resource available for application. 0
    calls with an out-of-service event.............. 0
    calls with an in-service event.................. 0
stat:   OUTBOUND time = 120 minutes
    attempted outbound calls..................... 14593
    successful outbound calls.................... 14566
    calls with no resource available for call setup.. 0
    calls failed because inbound had no CS resource.. 0
    calls with no resource available for application. 0
    calls with no resource available for CP..... 0
    calls with an out-of-service event.............. 0
    calls with an in-service event.................. 0
-------------------------------------------------
test successfully terminated
-------------------------------------------------
```

## Quad E1 Call Setup

Tests on Quad E1 boards were carried out using the CASmon utility. CASmon is included in the GTP Developers Kit. (For more information about CASmon, refer to the *CASmon User's Manual*.) CASmon can monitor any number of DSP or trunk timeslots, detecting tones and signaling bit changes. It saves the monitor results in a log file.

CASmon has an option to connect to two different ends of a connection, by monitoring a DSP resource and the corresponding trunk timeslot. This is useful to track both parties of a call setup. This option has been used to produce the following table.

The test setup is a Quad E1 board with two trunks connected with a loopback cable. Channel 60 places calls to channel 0 using R1.5.

(Dialing: 1234567, Category 1, ANI 2345678)

```
-----------------------------------------------------------
CASmon - CTA Digital CAS protocols tester V.1(Apr 28 1999)
test start time = Fri Apr 30 10:46:16 1999
-----------------------------------------------------------
01:trunk 1:000  10:46:23.420  Event: 0x120a0    1001 (B OFF)
01:dsp:000      10:46:23.460  Event: 0x120a0    1101 (A ON)
01:dsp:000      10:46:23.550  Event: 0x12048    USMF-1
01:dsp:000      10:46:23.610  Event: 0x12049    USMF_END
01:trunk 1:000  10:46:23.740  Event: 0x12048    USMF-1
01:trunk 1:000  10:46:23.800  Event: 0x12049    USMF_END
01:dsp:000      10:46:23.930  Event: 0x12048    USMF-2
01:dsp:000      10:46:23.990  Event: 0x12049    USMF_END
01:trunk 1:000  10:46:24.120  Event: 0x12048    USMF-2
01:trunk 1:000  10:46:24.180  Event: 0x12049    USMF_END
01:dsp:000      10:46:24.310  Event: 0x12048    USMF-2
01:dsp:000      10:46:24.370  Event: 0x12049    USMF_END
01:trunk 1:000  10:46:24.500  Event: 0x12048    USMF-3
01:trunk 1:000  10:46:24.560  Event: 0x12049    USMF_END
01:dsp:000      10:46:24.690  Event: 0x12048    USMF-2
01:dsp:000      10:46:24.750  Event: 0x12049    USMF_END
01:trunk 1:000  10:46:24.880  Event: 0x12048    USMF-4
01:trunk 1:000  10:46:24.940  Event: 0x12049    USMF_END
01:dsp:000      10:46:25.070  Event: 0x12048    USMF-2
01:dsp:000      10:46:25.130  Event: 0x12049    USMF_END
01:trunk 1:000  10:46:25.260  Event: 0x12048    USMF-5
01:trunk 1:000  10:46:25.320  Event: 0x12049    USMF_END
01:dsp:000      10:46:25.450  Event: 0x12048    USMF-2
01:dsp:000      10:46:25.510  Event: 0x12049    USMF_END
```

```
01:trunk 1:000  10:46:25.640  Event: 0x12048    USMF-6
01:trunk 1:000  10:46:25.700  Event: 0x12049    USMF_END
01:dsp:000      10:46:25.830  Event: 0x12048    USMF-2
01:dsp:000      10:46:25.890  Event: 0x12049    USMF_END
01:trunk 1:000  10:46:26.020  Event: 0x12048    USMF-7
01:trunk 1:000  10:46:26.080  Event: 0x12049    USMF_END
01:dsp:000      10:46:26.210  Event: 0x12048    USMF-4
01:dsp:000      10:46:26.270  Event: 0x12049    USMF_END
01:trunk 1:000  10:46:26.400  Event: 0x12048    USMF-C
01:trunk 1:000  10:46:26.460  Event: 0x12049    USMF_END
01:dsp:000      10:46:26.610  Event: 0x120a0    1001 (B OFF)
01:trunk 1:000  10:46:27.000  Event: 0x12048    USMF-D
01:trunk 1:000  10:46:27.030  Event: 0x12049    USMF_END
01:trunk 1:000  10:46:27.040  Event: 0x12048    USMF-1
01:trunk 1:000  10:46:27.070  Event: 0x12049    USMF_END
01:trunk 1:000  10:46:27.080  Event: 0x12048    USMF-8
01:trunk 1:000  10:46:27.110  Event: 0x12049    USMF_END
01:trunk 1:000  10:46:27.120  Event: 0x12048    USMF-7
01:trunk 1:000  10:46:27.150  Event: 0x12049    USMF_END
01:trunk 1:000  10:46:27.160  Event: 0x12048    USMF-6
01:trunk 1:000  10:46:27.190  Event: 0x12049    USMF_END
01:trunk 1:000  10:46:27.200  Event: 0x12048    USMF-5
01:trunk 1:000  10:46:27.230  Event: 0x12049    USMF_END
01:trunk 1:000  10:46:27.240  Event: 0x12048    USMF-4
01:trunk 1:000  10:46:27.270  Event: 0x12049    USMF_END
01:trunk 1:000  10:46:27.280  Event: 0x12048    USMF-3
01:trunk 1:000  10:46:27.310  Event: 0x12049    USMF_END
01:trunk 1:000  10:46:27.320  Event: 0x12048    USMF-2
01:trunk 1:000  10:46:27.350  Event: 0x12049    USMF_END
01:trunk 1:000  10:46:27.360  Event: 0x12048    USMF-D
01:trunk 1:000  10:46:27.390  Event: 0x12049    USMF_END
01:trunk 1:000  10:46:27.400  Event: 0x12048    USMF-1
01:trunk 1:000  10:46:27.430  Event: 0x12049    USMF_END
01:trunk 1:000  10:46:27.440  Event: 0x12048    USMF-8
01:trunk 1:000  10:46:27.480  Event: 0x12049    USMF_END
01:dsp:000      10:46:27.640  Event: 0x120a0    1101 (B ON)
01:dsp:000      10:46:28.780  Event: 0x120a0    1001 (B OFF)
```

Results are summarized in the following table, showing the first three cycles (shaded differently):

| Event | Timing (ms) | Comments |
|---|---|---|
| Seizure | 0 | Taken as base of timing |
| Seizure Acknowledge | 40 | |
| MF 1 from inbound side | 130 | Request first digit |
| MF end from inbound side | 190 | |
| MF 1 from outbound side | 320 | First DID |
| MF end from outbound side | 380 | |
| MF 2 from inbound side | 510 | Request next digit |
| MF end from inbound side | 570 | |
| MF 2 from outbound side | 800 | Second DID |
| MF end from outbound side | 860 | |
| MF 2 from inbound side | 990 | Request next digit |
| MF end from inbound side | 1050 | |
| MF 3 from outbound side | 1180 | Third DID |
| MF end from outbound side | 1240 | |
| … | … | … |
| Answer code | 3190 | And 500 Hz tone for ANI request |
| Beginning of ANI | | |
| MF D | 3580 | Beginning of series |
| MF 1 | 3620 | Category |
| MF 8 | 3660 | last ANI |
| MF 7 | 3700 | (last–1) ANI |
| … | … | … |

## AG 4000 Tests *Preliminary Results*

Tests on AG 4000 boards were carried out using the CASmon utility.

The test setup is an AG 4000 with two trunks connected with a loopback cable. Channel 60 places calls to channel 0 using R1.5.

(Dialing: 1234567, Category 1, ANI 2345678)

```
------------------------------------------------------------
CASmon - CTA Digital CAS protocols tester V.1(Apr 28 1999)
test start time = Wed Apr 28 15:02:12 1999
------------------------------------------------------------
00:trunk 1:000  15:02:33.270  Event: 0x120a0        1001 (B OFF)
00:dsp:000      15:02:33.300  Event: 0x120a0        1101 (A ON)
00:dsp:000      15:02:33.370  Event: 0x12048        USMF-1
00:dsp:000      15:02:33.430  Event: 0x12049        USMF_END
00:trunk 1:000  15:02:33.530  Event: 0x12048        USMF-1
00:trunk 1:000  15:02:33.590  Event: 0x12049        USMF_END
00:dsp:000      15:02:33.690  Event: 0x12048        USMF-2
00:dsp:000      15:02:33.750  Event: 0x12049        USMF_END
00:trunk 1:000  15:02:33.850  Event: 0x12048        USMF-2
00:trunk 1:000  15:02:33.910  Event: 0x12049        USMF_END
00:dsp:000      15:02:34.010  Event: 0x12048        USMF-2
00:dsp:000      15:02:34.070  Event: 0x12049        USMF_END
00:trunk 1:000  15:02:34.170  Event: 0x12048        USMF-3
00:trunk 1:000  15:02:34.230  Event: 0x12049        USMF_END
00:dsp:000      15:02:34.330  Event: 0x12048        USMF-2
00:dsp:000      15:02:34.390  Event: 0x12049        USMF_END
00:trunk 1:000  15:02:34.490  Event: 0x12048        USMF-4
00:trunk 1:000  15:02:34.550  Event: 0x12049        USMF_END
00:dsp:000      15:02:34.650  Event: 0x12048        USMF-2
00:dsp:000      15:02:34.710  Event: 0x12049        USMF_END
00:trunk 1:000  15:02:34.810  Event: 0x12048        USMF-5
00:trunk 1:000  15:02:34.870  Event: 0x12049        USMF_END
00:dsp:000      15:02:34.970  Event: 0x12048        USMF-2
00:dsp:000      15:02:35.030  Event: 0x12049        USMF_END
00:trunk 1:000  15:02:35.130  Event: 0x12048        USMF-6
00:trunk 1:000  15:02:35.190  Event: 0x12049        USMF_END
00:dsp:000      15:02:35.290  Event: 0x12048        USMF-2
00:dsp:000      15:02:35.350  Event: 0x12049        USMF_END
00:trunk 1:000  15:02:35.450  Event: 0x12048        USMF-7
00:trunk 1:000  15:02:35.510  Event: 0x12049        USMF_END
00:dsp:000      15:02:35.610  Event: 0x12048        USMF-4
00:dsp:000      15:02:35.670  Event: 0x12049        USMF_END
```

```
00:trunk 1:000  15:02:35.770  Event: 0x12048        USMF-C
00:trunk 1:000  15:02:35.830  Event: 0x12049        USMF_END
00:dsp:000      15:02:35.960  Event: 0x120a0        1001 (B OFF)
00:trunk 1:000  15:02:36.280  Event: 0x12048        USMF-D
00:trunk 1:000  15:02:36.290  Event: 0x12049        USMF_END
00:trunk 1:000  15:02:36.300  Event: 0x12048        USMF-1
00:trunk 1:000  15:02:36.330  Event: 0x12049        USMF_END
00:trunk 1:000  15:02:36.340  Event: 0x12048        USMF-8
00:trunk 1:000  15:02:36.370  Event: 0x12049        USMF_END
00:trunk 1:000  15:02:36.380  Event: 0x12048        USMF-7
00:trunk 1:000  15:02:36.410  Event: 0x12049        USMF_END
00:trunk 1:000  15:02:36.420  Event: 0x12048        USMF-6
00:trunk 1:000  15:02:36.450  Event: 0x12049        USMF_END
00:trunk 1:000  15:02:36.460  Event: 0x12048        USMF-5
00:trunk 1:000  15:02:36.490  Event: 0x12049        USMF_END
00:trunk 1:000  15:02:36.500  Event: 0x12048        USMF-4
00:trunk 1:000  15:02:36.530  Event: 0x12049        USMF_END
00:trunk 1:000  15:02:36.540  Event: 0x12048        USMF-3
00:trunk 1:000  15:02:36.570  Event: 0x12049        USMF_END
00:trunk 1:000  15:02:36.580  Event: 0x12048        USMF-2
00:trunk 1:000  15:02:36.610  Event: 0x12049        USMF_END
00:trunk 1:000  15:02:36.620  Event: 0x12048        USMF-D
00:trunk 1:000  15:02:36.650  Event: 0x12049        USMF_END
00:trunk 1:000  15:02:36.660  Event: 0x12048        USMF-1
00:trunk 1:000  15:02:36.690  Event: 0x12049        USMF_END
00:trunk 1:000  15:02:36.700  Event: 0x12048        USMF-8
00:trunk 1:000  15:02:36.730  Event: 0x12049        USMF_END
00:dsp:000      15:02:36.860  Event: 0x120a0        1101 (B ON)
00:dsp:000      15:02:37.910  Event: 0x120a0        1001 (B OFF)
```

Results are summarized in the following table, showing the first three compelled cycles (differentiated by shading):

| Event | Timing (ms) | Comments |
|---|---|---|
| Seizure | 0 | Taken as base of timing |
| Seizure Acknowledge | 30 | |
| MF 1 from inbound side | 100 | Request first digit |
| MF end from inbound side | 160 | |
| MF 1 from outbound side | 260 | First DID |
| MF end from outbound side | 320 | |
| MF 2 from inbound side | 420 | Request next digit |
| MF end from inbound side | 480 | |
| MF 2 from outbound side | 580 | Second DID |
| MF end from outbound side | 640 | |
| MF 2 from inbound side | 740 | Request next digit |
| MF end from inbound side | 800 | |
| MF 3 from outbound side | 900 | Third DID |
| MF end from outbound side | 960 | |
| … | … | … |
| Answer code | 2690 | And 500 Hz tone for ANI request |
| Beginning of ANI | | |
| MF D | 3010 | Beginning of series |
| MF 1 | 3030 | Category |
| MF 8 | 3070 | last ANI |
| MF 7 | 3110 | (last–1) ANI |
| … | … | … |

**Appendix A**

# Glossary

**ABCD bits** The four signaling bits used in channel associated signaling. The value that these bits take determines or indicates the state of the telephone channel. Many protocols do not use all four bits. Usually, only the A and B bits are used in North America, but all four are used in Europe. See also **CAS**.

**A bit** See **ABCD bits**.

**A-law** An encoding scheme that determines how an analog speech signal is converted to a digital signal. A-law encoding is used in Europe. The other algorithm, mu-law, is used in North America and Japan. See also **mu-law**.

**AG** The NMS **A**lliance **G**eneration family of telephony and voice processing hardware.

**AG configuration file**
Text file (usually named *ag.cfg*) that provides configuration information for all AG boards in a system. The AG configuration file also specifies runtime software such as DSP files and TCPs to download to the AG board(s) as part of board initialization. *agmon* uses the AG configuration file to initialize and load the AG boards in the system. See also **agmon**, **DSP file**, **TCP**.

**agmon** Utility program which reads a configuration file and uses it to load and monitor AG hardware.

**ANI** **A**utomatic **N**umber **I**dentification; a network service that provides a called party with the phone number of the calling party. It is typically available to the called party in R1, R2, and ISDN signaling systems. The implementation of ANI can vary between different telephone companies and even different central office switches. See also **CAS**.

**API** **A**pplication **P**rogramming **I**nterface; a library of function calls that allow an application developer to access functionality in a uniform and consistent way.

**blocking (function call)**
Stopping the executing process or thread while waiting for an internal completion event or signal.

**blocking (telephone call)**
Call control action taken by an application to refuse any further incoming calls. Depending on the telephony protocol, blocking calls on a line tells the CO (Central Office) or switch not to offer any calls, or causes the line to reject every call that is being offered. Some protocols allow an application to designate a particular rejection method, such as playing a busy signal or a reverse-make busy signal.

**call control** Any part of telephone call establishment, including setting up, blocking, monitoring, transferring, releasing, or tearing down the telephone call.

**call progress** In CT Access, functions that allow the application to control and supervise call placement. Call progress monitors in-band energy to detect network tones, voice, and other tones such as modems.

**CAS** **C**hannel **A**ssociated **S**ignaling; a signaling method where signaling information is provided on a line-by-line or channel-by-channel basis. Signaling information is sent for each channel at regular intervals, regardless of state changes. Each transmission channel is associated with one or more transmission paths.

**CEPT** **C**onference of **E**uropean **P**ostal and **T**elecommunications administrations; a European telephony standards committee.

**CEPT E1** A standard used in Europe for digital telephone carriers. CEPT E1 is similar to T1 but operates at 2.048 Mbps and has 32 channels instead of T1's 24. See also **T1**.

**CCITT** **C**omité **C**onsultatif **I**nternationale de **T**élégraphique et **T**éléphonique, now called the ITU-T (International Telecommunications Union-Telecommunications Services Sector); the United Nations agency concerned with telecommunications standards, the harmonized evolution of networks and services, and support for developing countries.

**channel** An electronic communications path. In digital telephony, a channel usually refers to a separate connection carried on a digital trunk.

**Channel Associated Signaling**
See **CAS**.

**compelled sequence**
An exchange of in-band signaling during call setup, where each party waits for receipt acknowledgment from the other party before sending the next signal.

**CT Access** Natural MicroSystems development environment for telephony applications.

**DID number** **D**irect **I**nward **D**ial number; the called number which is collected by the TCP and is passed to the application.

**digital** Information in a discrete or quantized form.

**Digital Signal Processor**
See **DSP**.

**DSP** In telephony, a **D**igital **S**ignal **P**rocessor; a microprocessor that is designed to perform the calculations required for voice processing.

**DSP file** A low-level program, downloaded to a board at initialization, that enables the board's DSPs to perform tasks such as DTMF signaling, voice recording, and playback.

**DTMF** **D**ual-**T**one-**M**ulti-**F**requency; an in-band signaling system that uses two simultaneous voice-band tones for dialing. Also called touchtone.

**E1** See **CEPT E1**.

**E&M** **E** lead **and M** lead; a telephone signaling system used between two switches that uses two wires called the E lead and the M lead. In digital E&M signaling, the A bits and B bits are used.

**EUC protocols** **EU**ropean digital **C**hannel associated signaling protocols. A protocol family including special channel associated signaling protocols used in certain European countries. The protocols use two-bit line signaling, not as specified by CCITT, but as specified by national documents. The register signaling is either carried by in-band DTMF tones (not compelled) or by out-of-band decadic pulses. Also known as EuroCAS. See also **CAS**.

**events** In CT Access, data structures that contain notification of certain conditions or state changes. Events can be sent from the hardware or from another software module to the application program. The event structure contains information about the specific event, including a timestamp, associated CTA context, specific data, error codes, or reason codes.

**full-duplex** Simultaneous two-way voice and two-way signal data transmission.

**gateway application**
Generic term for a host-based application with interfaces to more than one network (for example, packet-switched and circuit-switched). Gateway applications are used to route data from one network to another and/or from one protocol to another.

**H.100 bus** A PCM digital bus standard for integrating hardware from various PC board vendors which enables boards to share voice data, signaling data, and switching information. The H.100 bus is an interoperable superset of the H-MVIP and MVIP-90 telephony buses. It can be addressed using the MVIP-95 switch model. See also **MVIP-90**, **MVIP-95**.

**host** The PC on which an application runs. Also used to designate a computer with full two-way access to a network such as a local area network or the Internet.

**inbound application**
An application designed to receive inbound calls.

**inbound call** A call received by an inbound application, placed by a remote party.

**IVR** **I**nteractive **V**oice **R**esponse; a telephony application in which callers interact with a program using recorded or synthesized voice prompts, DTMF digits, or speech recognition to query or deliver information.

**IWK** **I**nternational **W**in**K** Start protocols; a protocol family that includes country-specific variations of the US Wink Start protocol used in certain countries on T1 trunks. The protocol uses one-bit signaling, and performs register signaling by in-band DTMF or MF tones, or by out-of-band decadic pulses.

**line** A logical telephone connection on which a call can take place.

**loopback** A local connection between the input and output of a device. This is typically used for testing purposes.

**MF** **M**ulti-**F**requency; an in-band interoffice tone-based signaling method using pairs of frequencies from a set of six available frequencies.

**MFC-R2** ITU standard line protocol for E1 lines using compelled MF tones and channel-associated signaling. Almost every country has a national variant of this standard. See also **CAS**.

**MFS** **M**ulti-**F**requency compelled **S**ocotel standard protocol: a protocol that uses one-bit steady-state line signaling and performs register signaling using MF tones. It uses a compelled scheme that is different from the CCITT or MFC-R2 schemes. MFS is used in only one country (Spain) and only on E1 lines.

**mu-law** One of two algorithms used in telephony to logarithmically compress or expand digitized speech. mu-law is used in North America and Japan. A-law is the other algorithm, used in European networks. See also **A-law**.

**MVIP** **M**ulti-**V**endor **I**ntegration **P**rotocol; PCM digital bus standard for integrating various board vendors. Facilitates software-controlled digital switching within the PC chassis. MVIP interconnects switching and telephony processing boards on a PC. MVIP is an interoperable subset of the H.100 bus standard. See also **H.100 bus**, **host**.

**MVIP-90** Original MVIP standard for hardware and software requirements for a standard telephony bus.

**MVIP-95** Device driver specification for H-MVIP, H.100, and H.110 telephony buses.

**MVIP bus** A high-speed, multiplexed digital telephony “highway” which allows boards to share data, signaling information, and switching information.

**NMS** **N**atural **M**icro**S**ystems Corporation.

**off-hook** The active state of a telephone circuit. The term is derived from old telephone sets where the receiver, when in use, was lifted from a hook attached to a switch. The opposite condition is on-hook.

**outbound application**
An application designed to place outbound calls.

**outbound call** A call placed by an application to a remote party.

**out-of-band** In telephony, information carried outside of the audio or voice channel.

**pattern** In telephony, a repeated 8-bit sample applied to an output terminus of a switch block.

**PCI** **P**eripheral **C**omponent **I**nterconnect; a 32/64 bit local bus inside a PC.

**port** A point of connection; refers to a logical entity on a DSP board which can run telephony protocols, play and record voice files, or execute other DSP functions. It is typical to connect such a DSP port to a line interface or a trunk by using CT bus switching. This type of port generally corresponds to a CTA context in the ADI service. Refers to telephone connections supported by a board (for example, the AG 2000 board is an 8-port board).

**protocol (telephony)**
Defined procedure for call setup and call teardown.

**Pulsed E and M** A telephone signaling system used between two switches that uses two wires called the E lead and the M lead. In analog E and M, the M lead of one switch is connected to the E lead of the other, and vice versa. In digital E and M signaling, a single bit is used to convey signaling states. See also **E&M**.

**R1.5** System R1.5; an international protocol family that includes channel associated signaling protocols used for E1 lines in Russia.

**R2 signaling** One of several standards for channel associated signaling. See also **CAS**.

**release guard time**
An interval in which an inbound protocol remains in blocking state after both parties have hung up.

**runfile** Mandatory low-level runtime software which is downloaded to an AG board as part of its initialization. The runfile for a board is specified in the AG configuration file. When *agmon* runs, it transfers the runfile from the host into on-board memory. See also **AG configuration file**, **agmon**.

**SIT** **S**pecial **I**nformation **T**one; a series of three tones followed by an informational voice message indicating a network error such as congestion or an unknown telephone number.

**solicited event** An event that occurs as a consequence of some function call.

**special information tone**
See **SIT**.

**stream** A grouping of timeslots that usually corresponds to a particular bit-stream of time-domain multiplexed (TDM) serial data on an individual track or wire of a bus.

**switch** In telephony, a device that can connect one of several inputs to one of several outputs.

**synchronous function**
A function that does not return until the operation it initiated has completed.

**T1** A standard for telephone transmission that multiplexes 24 digital voice channels on a single 1.544 Mbyte/s carrier.

**TCP** **T**runk **C**ontrol **P**rogram; a downloadable module containing the low-level code to interface with an analog or digital trunk running a certain protocol. TCPs are specific to a trunk protocol; for example, one-way inbound with wink start (which is used for DID and DNIS).

**timeslot** Specifies a particular 64 kbit/s subdivision of a TDM bus stream. Timeslots number from zero (0) to ***n*** where ***n*** is stream-dependent.

**toll category** In MFC-R2 and E&M protocols, the digit received by the inbound application that indicates the Group II category of incoming ANI digits. Usually this is the same as the user category, but in some countries it carries the toll category of the call. See also **ANI**, **E&M**, **MFC-R2**, **user category**.

**trunk** For NMS products, a trunk isdefined as the physical interface between the telephone network and the board. It can be used interchangeably with line because it could be connected to either a trunk or a line.

In telephone networks, a trunk is defined as a shared connection between two switches. It differs from a line in that it is not dedicated to one subscriber or extension. Trunks should have good machine-to-machine interfaces. Compared to lines, trunks have different signaling requirements, possibly including billing information.

**Trunk Control Program**
See **TCP**.

**unsolicited events**
Events that are not generated in response to a command.

**user category** In MFC-R2 and E&M protocols, the digit received by the inbound application that indicates the Group I category of incoming ANI digits; e.g., normal subscriber, operator, maintenance equipment. See also **ANI**, **E&M**, **MFC-R2**, **toll category**.