ICL

Title: Telephone Handler

Reference: PSD 76.97.3.3

0 DOCUMENT CONTROL

0.1 Contents

0.2 Changes Since Previous Issue

This is the first issue as a PSD. Apart from changes of layout, technical differences from the last issue of the predecessor document are minor and reflect the release 1 issue state of Telephone Handler.

0.3 Document Predecessor

OPD/TH/1 Issue 4/1 OPD Telephone Interface Procedures Specification.

0.4 Changes Forecast

This first issue as a PSD is intended to define interfaces available in the first release product. No changes are expected except for correction of errors or clarifications.

0.5 Document Cross References

[1] R51002 OPD Handbook.
[2] PSD 76.97.3.1 OPD Kernel Specification.
[3] PSD 76.97.3.2 OPD Director Facilities for Application Writers.

1. GENERAL

1.1 Scope

This document describes the procedure interfaces provided into the Telephone Handler subsystem that are available to OPD applications.

It does not describe, except in very general terms, how the Telephone Handler subsystem works; this is covered in an internal design document.

1.2 Introduction

The Telephone Handler subsystem controls all direct use of telephone facilities by the OPD user. The Telephone Handler interface procedures allow an OPD application to direct the subsystem to carry out a limited set of telephone operations on behalf of the application.

The Telecom authorities restrict the range of possible telephone operations available to applications, thus for example it is not permitted for an application to take a line offhook, dial a number and connect a voice synthesised message to the line.

1.3 Terminology

Kernel Director Application Handler Telephone Handler	Components of OPD system software
Application	A program of the type that can be scheduled (to run, use the screen etc.) through Application Handler facilities. See [3]
Activity	A processing thread with associated registers etc. See [2] for details

2. SUMMARY

This document defines the Telephone Handler interfaces that are available for the use of OPD applications.

3. TELEPHONE HANDLER DESCRIPTION

3.1 General

The Telephone Handler subsystem is that part of the system software responsible for controlling all operations on the telephone lines above the Kernel interface.

In addition to all direct use of the telephone facilities by the end user, e.g. the handset, the speaker and the special telephone keys on the keyboard, the subsystem also provides programming interfaces for other OPD applications to gain access to the phone lines, e.g. to autodial or connect data calls.

A major reason for the existence of the subsystem is to provide an additional level of control of the phone lines above the Kernel interface, to ensure that, where necessary, direct end user phone actions take precedence over any application usage, so that the telephone always behaves as far as possible like an ordinary phone. The usual example given is that the user must at any time be able to make an emergency 999 call.

3.2 Structure

The Telephone Handler subsystem runs as two separate activities, each responding to events on a particular phone line. These two activities share the same code.

Additionally there is the code of the programming interfaces used by other applications, which when executed will run in the activity of the calling application.

Interaction between the interface code and the phone line activities, and between the activities themselves, is signalled using the Kernel event system, and data is passed via a shared telephone database held in fixed system RAM.

The telephone database, amongst other data, holds information on the current status of each phone line, and other related telephony resources. This can be interrogated via an interface procedure by an interested application.

The line activities permanently hold the only Kernel channels to the phone lines, thus applications' use of the lines must also be handled by these activities as directed via the interface procedures. This ensures that priority is always given to direct end user actions. An application's use of a phone line is via a call on an interface procedure, which after some initial checks that the line and resource states are appropriate, will signal the appropriate line activity to take action.

3.3 Telephone Handler Interface

3.3.1 GENERAL

The interface procedures of the Telephone Handler subsystem allow applications to gain access to the telephone lines and configure related resources.

The procedures permit applications to carry out telephone operations as for example, autodialling, manual dial, making data calls and to direct, or obtain information on, call timing.

The interface procedures are called from an activity of any application and therefore run as part of that activity. The operations that a Telephone Handler interface procedure are directed to take on a telephone line, must be communicated to the appropriate line activity for action. This process is accomplished using the common telephony data base of status and action information and the Kernel event system.

The interaction process between these separate activities will usually proceed as follows -

- The interface procedure is called to initiate an operation on a telephone line.
- It will check using the status information in the telephone database as far as it can that the operation is feasible, if not immediately returning to its caller.
- It will store the requested action details in the telephone database.
- It will cause a local event for the telephone activity of the line in question.
- It will then exit to its caller, with a success response.
- At some point the line activity, when checking events received, will action this local event and carry out the actions requested.

Some examples of the use of the Interface Procedures follow:-

(i) Directory Dialling.

The Directory application uses the AUTODIAL-VOICE interface to direct the line activity to autodial the specified telephone number, and then to set up the line for call timing as desired.

The human user of the Directory application will, prior to the AUTODIAL-VOICE call, have picked up the handset (or pressed SPKR key) and thus established the initial voice path manually.

The progress of the call and termination will be, as for any normal manual call, handled by the line activity.

(ii) Data Dialout.

Here the data activity opens a channel to the modem initially, setting up its required configuration and obtaining its access channel number.

It will use the AUTODIAL DATA interface procedure to signal the line activity to acquire a data line, auto dial the number and issue a 'Switch line to Data' command, see [2]. At which point the configured modem is switched onto the line and the data connection attempted by the phone module. Meanwhile following successful exit from the interface procedure, the data activity waits for the specified event from the line activity, this will be signalled when the modem connection succeeds or fails.

The progress of data transfer is the sole responsibility of the data activity. On completion or abandonment of the process, the data activity should close the modem channel and call the ENDCALL Interface Procedure to signal the line activity to terminate the phone line connection. Failure to do this, as for example in an unscheduled abort of the data activity, will leave the line connected until the machine is switched off!

(iii) Data Auto-answer.

The data auto-answer protocol activity is started by Telephone Handler at switch on and is then suspended waiting on local event 13 from the Telephone Handler.

When the Telephone Handler decides to answer a call on a line set up for auto-answer, it causes local event 13 to this data auto-answer activity. The data auto-answer activity then opens the modem and calls the interface procedure DATA AUTO-ANSWER to connect the modem to the line. When the modem connection succeeds or fails, the line activity signals the nominated event to the AA activity.

It should be noted that although an activity may have successfully gained access to and configured a telephone line, a later call on another interface procedure may still fail with ERR.NT (Not a valid phone line) due to the line activity having overridden the connection as a result of direct user action.

The Telephone Handler interface procedures are as follows:-

AUTODIAL_VOICE - Used to acquire a voice configured telephone line and autodial out on it.

AUTODIAL_DATA - Used to acquire a data configured telephone line, autodial out on it and 'switch line to data' to get a modem connection.

DIAL_VOICE - Used to acquire a voice telephone line and allow manual dial out on it.

DIAL_DATA - Used to acquire a data telephone line, allow user to manually dial out and 'switch line to data' to connect modem.

CONFIG_VOICE - Used in DATA in VOICE cases to reconfigure a data call back to voice mode.

CONFIG_DATA - Used in DATA in VOICE cases to configure a voice call to data mode and issue a 'switch line to data'.

TIMECALL - Used to initiate call charging on an outgoing call, or to change to a new charge band or none, or to reset timer. See [1].

CALLDET - Used to obtain call charging details of the last call dialled out on the specified line.

MONITOR - Used to change the monitoring state of a data call, i.e. to connect speaker or disconnect it.

DATA AUTO-ANSWER - Used to provide details of an event number to be used by Telephone Handler to signal modem connection in auto-answer data calls.

ENDCALL - Used to terminate a telephone line connection in data call cases.

CALL TIMING - Used by configurator to switch call timing on/off on a telephone line.

ENQUIRE - Used to provide details of the current state and resource configuration of the specified phone line.

SETUP DATA AA - Used by configurator to record a change of Auto-answer

CHECK CHARGEBAND - Used by messaging to check whether a user-supplied charge band is valid.

UPDATE NOTICEBOARD - Used by Auto-answer control to ensure a change in auto-answer status is reflected in the Noticeboard.

VOICE AA ENQUIRY - Used by Voice response to determine whether a particular response is being output.

These interfaces are described in detail in section 4.

3.3.2 CODE INTERFACE

All the Telephone Handler interface procedures are entered, via the Kernel, using the TRAP instruction mechanism. A single TRAP number is allocated for this purpose, with the different procedures distinguished by different Action Value parameters. The standard interface is -

```
TRAP  #T.PHONE
```

and the action value distinguishing the procedure is always passed in DO.B.

For each call any extra parameters passed in and those returned are always in the registers. Where a parameter value does not occupy the full 32 bits of a register, the contents of leading bits are ignored.

All registers not used as parameters are preserved over the procedure call.

On exit from an interface procedure, its response will be in DO.L always. This response will be 0 or +ve if the call was successful, otherwise it will take the form of a -ve number defined as # ERR.cc where cc is a 2 character error code. (For definitions of #ERR codes see [2] Appendix B.) The condition code register CCR will be set correspondingly.

A general response of ERR.NC will be returned by any Interface procedure if a line activity is busy processing an earlier call.

4. DETAILED INTERFACES

4.1 AUTODIAL VOICE

4.1.1 INPUT PARAMETERS

D0.B	Action Value	= #TH.AUTOV
D1.W	Length	= Length of dial sequence string
D2.W	Timing required	= 0 if not required 1 if timing forced 2 as set on line
D3.W	Charge Band (CB)	= 1/2 character charge band code, left justified with binary zero fill, all binary zero if no CB
D6.W	Line number	= Allowed line number 1, 2, or 0
A0.L	Dial Sequence	= Address of string to be dialled

4.1.2 OUTPUT PARAMETERS

D0.W	Line number	= Line number used for call

Error Responses BP : Bad parameter
IU : Telephone Line activity busy
NT : If a specified line is not in the correct state, but the other line is
NO : Neither phone line is in correct state

4.1.3 DESCRIPTION

This procedure will be called by an activity wishing to acquire a voice telephone line and autodial out on it.

The input parameters are validated, that of the dial sequence string being checked to consist only of the ASCII numerals or *, #, or - (for 4 second pause). An invalid parameter is immediately signalled with an error response of BP.

If the line number parameter is 0, the current 'preferred voice' line will be used for the dial if possible, else if busy the other line will be used. If neither line is in a suitable state then the response ERR.NO will be returned.

If the line number parameter is 1 or 2 then only phone line 1 or 2 will be used, if this specified line is not in the correct state, but the other line is, then a response of ERR.NT is made. If neither line is in the correct state the response will be ERR.NO.

A suitable phone line must be in the PREDIAL state (that is the handset must be up or SPKR pressed and dialling not yet started).

The procedure will 'assign' the telephone line in the PREDIAL state, signal the appropriate Telephone Handler Activity to configure the line and carry out the auto dial, and return a success response of the line number allocated.

It is assumed that an activity calling this procedure will have no further interest in the call once a success response has been returned. It is possible that the Telephone Handler Activity may not succeed in its actions, in which case the user would need to reactivate the originating activity.

Note that the 'Timing Required' parameter may be used to force use of a supplied Charge Band code, or a prompt for one, in the case where the line is not set for call timing. If set to 'as set on line' a CB will be used, or prompted for, if call timing is on, but not if call timing is set off. If set to 'not required' then even if no CB and the line is set for call timing, no prompt will be issued.

The appropriate telephone line window in the Noticeboard is updated as the dial progresses, with the number dialled, and will eventually show the line in voice dial out state, or call timing in progress, see [1].

4.2 AUTODIAL DATA

4.2.1 INPUT PARAMETERS

D0.B	Action Value	= #TH.AUTOD
D1.W	Length	= Length of dial sequence string
D2.W	Timing required	= 0 if not required 1 if timing forced 2 as set on line
D3.W	Charge band (CB)	= 1/2 character charge band code, left justified with binary zero fill, all binary zero if no CB
D5.W	Monitor/shortcode	= Bit 0 set if monitor is required, else clear Bit 1 set if number is derived from a shortcode, else clear
D6.W	Line Number	= Allowed line number 1, 2 or 0
D7.W	Event Number	= Event number in binary (not a mask)
A0.L	Dial Sequence	= Address of string to be dialled

4.2.2 OUTPUT PARAMETERS

D0.W	Line number	= Line number used for call

Error Responses BP : Bad parameter
IU : Telephone Line activity busy
NT : If a specified line is not in the correct state, but the other line is
NO : Neither phone line is in the correct state

4.2.3 DESCRIPTION

This procedure will be called by an activity wishing to acquire a data telephone line, autodial out on it and start data transfer (i.e. establish data connection).

The calling activity will have previously opened a channel to the modem specifying the configuration it requires.

The input parameters are validated (as for AUTODIAL_VOICE), and errors return response BP.

The choice of a suitable phone line is made in a similar fashion to

AUTODIAL VOICE in this case choosing the preferred data line, with possible responses of NT or NO returned in the same cases.

The procedure will 'assign' the chosen telephone line, in this case in the ONHOOK state, signal the appropriate Telephone Handler Activity to configure the line for data, (monitoring set if specified), carry out the autodial sequence and attempt to establish data connection by using the 'switch line to data' command see [2]. A success response of the line number allocated is returned by the Interface Procedure immediately it has signalled the telephone line activity as above.

The calling activity after a success response from the interface will wait on the event nominated in D7; this event will be sent by the Telephone Handler to the activity calling this procedure when the modems at each end of the line have established connection, or have given up the attempt. The activity should do a GIVE CHANNEL STATUS on the modem to see whether connection has occurred. If the connection attempt fails then the line will be put ONHOOK again.

If the specified line is "ringing" when the Telephone Handler activity starts to process this request, the requested data call will not be proceeded with and the caller's event will be signalled.

If monitoring is requested but the loudspeaker is not available, Telephone Handler will still proceed with the call, but it will not be monitored.

The calling activity may subsequently use the MONITOR interface procedure to change the monitoring state on the line.

Note that the 'timing required' parameter may be used to force use of a supplied CB code, or a prompt for one, in the case where the line is not set for call timing. If set to 'as set on line' a CB will be used, or prompted for, if call timing on but not if it is set off. If set to 'not required' then even if no CB and line set for call timing no prompt will be issued.

The appropriate telephone line window in the Noticeboard is updated as the dial progresses, with the number dialled, and eventually will show the line in data dial out or call timing in progress, see [1].

If bit 1 of the monitor/shortcode parameter is set, the data number will be assumed to have been derived from a shortcode entry, and the number will not be inserted into the LAST 6 list for re-dialling.

4.3 DIAL VOICE

4.3.1 INPUT PARAMETERS

D0.B	Action Value	= #TH.DIALV

4.3.2 OUTPUT PARAMETERS

D0.W	Line Number	= Line number used for call

Error Responses NO : Lines in use, no line allocated.

4.3.3 DESCRIPTION

This procedure will be called by an activity wishing to acquire a voice line and allow manual dialout on it.

If no suitable free line exists, the procedure will exit immediately with response NO.

Otherwise the procedure will 'assign' a free telephone line in the PREDIAL state, signal the appropriate Telephone Handler activity, and return a success response of the line number allocated.

The appropriate telephone line window in the Noticeboard is updated to the dialling state and will show the number manually dialled, see [1].

4.4 DIAL DATA

4.4.1 INPUT PARAMETERS

D0.B	Action Value	= #TH.DIALD
D5.W	Monitor	= 0 if no monitor required 1 if it is
D7.W	Event Number	= Event number in binary (not a mask)

4.4.2 OUTPUT PARAMETERS

D0.W	Line Number	= Line number used for call

Error Responses BP : Bad parameter
NO : Lines in use, no line allocated

4.4.3 DESCRIPTION

The procedure will be called by an activity wishing to acquire a data line, allow the user to manually dial the number and start data transfer.

The calling activity will have previously opened a channel to the modem specifying the configuration it requires.

The input parameters are validated and if there are any errors a response of BP is made.

If no suitable free telephone line exists, the procedure immediately returns a response of NO.

Otherwise the procedure will 'assign' a free telephone line in the ONHOOK state, and return a success response of the line number used - after signalling the Telephone Handler Activity to configure the line for data (with monitoring set if specified), switch into PREDIAL state with keyboard (ready for manual dialout). The Telephone Handler Activity, after each numeric digit dialled on such a line, will issue the 'switch line to data' command, to be repeated on failure after any subsequent digit is dialled.

If the line chosen is "ringing" or the keyboard is not free for dialling when the Telephone Handler activity starts to process this request, then the requested data call will not be proceeded with and the caller's event will be signalled.

If monitoring is requested but the loudspeaker is not available, Telephone Handler will still proceed with the call, but it will not be monitored.

When the Telephone Handler Activity has successfully established a data connection, it will release the keyboard from the line.

After a success response from DIAL DATA the calling activity will wait on the event nominated in D7. This event will be sent by the Telephone Handler to the activity calling this procedure when the modems at each end of the line have established connection, or have given up the attempt. The activity should do a GIVE CHANNEL STATUS on the modem to see whether connection has occurred. If the connection attempt fails then the line will be put ONHOOK again.

The appropriate telephone line window in the Noticeboard will be updated to the dial state, and then will be updated in the usual way as the manual dial progresses, see [1].

4.5 CONFIG VOICE

4.5.1 INPUT PARAMETERS

DO.B	Action Value	= #TH.CONFV
D1.W	Line Number	= Telephone line to be reconfigured, 1/2 or 0

4.5.2 OUTPUT PARAMETERS

DO.W	Line Number	= Line number reconfigured

Error Responses BP : Bad parameter
NT : Not a valid phone line
IU : Handset/loudspeaker in use

4.5.3 DESCRIPTION

The procedure will be called by an activity wishing to reconfigure the specified line from data mode to voice mode.

The line number parameter may be 1 or 2 to specifically select phone line number 1 or 2, or 0 in which case the phone line currently connected to the modem will be used. If there is no such line a response of NT is returned.

The input parameter will be validated and BP returned if the line number is not valid, or NT if the line is not connected in data mode, or IU if handset/loudspeaker in use.

Otherwise the procedure will return a success response after signalling the Telephone Handler Activity to 'terminate the modem connection' and reconnect the handset/speaker to the line.

If Telephone Handler finds the handset still onhook and the speaker in use on the other line when it comes to connect the audio path, it will put the line into HOLD.

This procedure is used to reconfigure a DATA in VOICE call back to VOICE, and the handset or loudspeaker will have originally been connected to the line.

The Noticeboard window of the line will be changed from DATA CALL to VOICE CALL.

4.6 CONFIG DATA

4.6.1 INPUT PARAMETERS

D0.B	Action Value	= #TH.CONFD
D1.W	Line Number	= Telephone line to be reconfigured 1, 2 or 0
D5.W	Monitor	= 1 if monitoring required, else 0
D7.W	Event number	= Event number in binary (not a mask)

4.6.2 OUTPUT PARAMETERS

D0.W	Line Number	= Line number reconfigured

Error Responses BP : Bad parameter
NT : Not a valid phone line

4.6.3 DESCRIPTION

The procedure will be called in the DATA in VOICE call case to reconfigure the specified line from voice mode to data mode.

The line number parameter may be 1 or 2 to specifically select phone line 1 or 2, or 0 in which case the phone line currently connected to the handset/SPKR will be used. If there is no such line then a response of NT is returned.

The input parameters are validated, with BP being returned for invalid parameter values, or NT if a specified line is not found in voice mode.

Otherwise the procedure returns a success response after signalling the Telephone Handler Activity to reconfigure the line for data, with monitoring if required, and to issue a command to switch the modem to the line and attempt a data connection.

After a success response from CONFIG DATA the calling activity will wait on the event nominated in D7; this event will be sent by the Telephone Handler to the activity calling this procedure when the modems at each end of the line have established connection, or have given up the attempt. The activity should do a GIVE CHANNEL STATUS on the modem to see whether connection has occurred. If the connection attempt fails then the line will be reconfigured to voice.

The handset, if connected to the line, will be left in the LIMBO state ready for subsequent reconnection unless it is replaced on its rest by the user.

The Noticeboard window of the line will be changed from VOICE CALL to DATA CALL.

4.7 TIMECALL

4.7.1 INPUT PARAMETERS

D0.B	Action Value	= #TH.TIMEC
D1.W	Line Number	= Telephone line identifier (1/2)
D3.W	Timing CB	= 1/2 character charge band code, left justified binary zero fill, or all binary 0 or all spaces

4.7.2 OUTPUT PARAMETERS

None

Error Responses BP : Bad parameter
NT : Not a valid phone line

4.7.3 DESCRIPTION

The procedure is used to initiate call timing on the specified telephone line, or to alter the charging to a different charge band code.

The procedure does not interact with the Telephone Handler Activity. It will cause similar results to a user press of the TIMING key, the exception being that the charge band code is already supplied so no prompt will appear.

That is - if this is the first call of the procedure since the start of call timing and no previous charge band code was available, the timer is not reset and all timing up to this point and from now on will be charged to the supplied code. If this is not the first call, or there was a previous code, then charging to the previous code is completed and the timer reset to recommence charging to the new code.

The procedure has two special modes of use. First, if the charge band code parameter is zero then the timer will be reset, but the current charge band code is unchanged. Note that all, possibly separate, chargings within the current call to this charge band will be cleared.

Second, if the charge band code parameter is all spaces, then charging to any current code will be completed and although timing continues with timer reset, no specific charge band code will be charged.

The appropriate telephone line window of the Noticeboard is updated to show the new timing state, see [1].

Failure responses from the procedure are -

ERR.NT The specified line is not in the required state for call timing, that is, it is not in the offhook/DIALLING or POSTDIAL states.

ERR.BP An invalid parameter value has been supplied.

4.8 CALLDET

4.8.1 INPUT PARAMETERS

D0.B	Action Value	= #TH.CALLD
D2.W	Buffer size	= Size of buffer below in bytes
A0.L	Buffer address	= Address of buffer area to receive call timing details

4.8.2 OUTPUT PARAMETERS

None

Error Responses BP : Bad Parameter
NT : Not a valid phone line
BO : Buffer overflow

4.8.3 DESCRIPTION

The procedure is used to obtain the call timing details of the last number dialled out on either line.

The procedure has no direct interaction with the Telephone Handler Activity. It will validate the buffer address supplied, and if correct the procedure will return the charging information as below in the buffer, together with a success response of 0.

The buffer contents will be:-

SIZE	one word	The byte size of the telephone number string.
CB	two bytes	Charge Band code, left justified binary zero filled. All binary zero if no band specified.
LTN	26 bytes	The last telephone number dialled. Left justified, right space filled in normal telephone number format.
LT	one word	Length of time in seconds bought by 1 charge unit at this rate.
TIME	one long word	Number of seconds charged at this rate.
COST	one word	Cost in pence of 1 charge unit at this charge rate.

Note in the above LT and COST are not valid for the case where CB = All zero, i.e. no charge band specified.

Note that the CB returned will be that last specified for the call.

Failure responses from the procedure are -

ERR.NT There has been no call dialled out on either line. Buffer contents are not valid.

ERR.BP A parameter value is invalid.

ERR.BO The specified buffer was not long enough. Information will have been inserted into the buffer with right truncation.

4.9 MONITOR

4.9.1 INPUT PARAMETERS

D0.B	Action Value	= #TH.MONIT
D1.W	Line Number	= Telephone line to be monitored, 1/2
D5.W	On/Off	= 1 switch loudspeaker to line 0 switch loudspeaker off

4.9.2 OUTPUT PARAMETERS

None

Error Responses	BP	: Bad parameter
	NT	: Not a valid phone line
	IU	: Loudspeaker in use elsewhere

4.9.3 DESCRIPTION

The procedure will be called to change the monitoring state on an already connected data telephone line.

The input parameters are validated, with BP being returned for invalid parameter values, NT if the line specified is not set up for data, or the monitoring state is not as expected, and IU if the loudspeaker is already assigned elsewhere.

Otherwise the procedure returns a success response, after signalling the Telephone Handler Activity to reconfigure the monitor state as specified.

The attachment/detachment of the loudspeaker from the line will be displayed in the appropriate Noticeboard area, see [1].

4.10 DATA AUTO-ANSWER

4.10.1 INPUT PARAMETERS

D0.B Action Value	= #TH.DTAA
D1.W Event Number	= -1 or event number in binary

4.10.2 OUTPUT PARAMETERS

D0.W Line Number	= Line number that is set for data auto-answer

Error Response BP : Bad parameter

4.10.3 DESCRIPTION

When an incoming call is detected by Telephone Handler on a line previously set for Data Auto-Answer, it will cause event 13 to be sent to the nominated Protocol Server.

DATA AUTO-ANSWER procedure should then be called by that server activity that opens the modem to signal to Telephone Handler that the modem is open.

The event number parameter will be used by Telephone Handler to signal that event to the server activity (the caller of DATA AUTO-ANSWER), when the data call has been answered and the modems at each end of the line have either successfully established the link, or have failed to do so. The modem state can subsequently be obtained from a GIVE CHANNEL STATUS call on the modem channel.

If the protocol server has not opened the modem successfully, or otherwise decides it does not wish to answer the data call, it should call DATA AUTO-ANSWER with a -1 event number parameter. Telephone Handler, if it receives a -1 parameter, will abandon its attempt to auto-answer the current call in data mode and either leave the line ringing or attempt voice auto-answer.

4.11 ENDCALL

4.11.1 INPUT PARAMETERS

D0.B	Action Value	= #TH.ENDC
D1.W	Line	= Telephone line number (1/2)

4.11.2 OUTPUT PARAMETERS

None

Error Responses	BP : Bad parameter
	NT : Not a valid phone line

4.11.3 DESCRIPTION

This procedure may be used to terminate a telephone call on the specified line, in auto-answer/autodial data cases where user manual termination is not expected.

The procedure will check the line is in an offhook state and, if it is, signal the appropriate telephone line activity to terminate the line and ENDCALL will return a success response. The Telephone Handler Activity will, asynchronously, terminate the call and free any resources held. Note that the calling activity should itself first close any channels it holds to the modem or the speech synthesiser.

The appropriate telephone line window in the Noticeboard is updated to show the line is now ONHOOK.

Failure responses from the procedure are -

ERR.NT If the specified line is already onhook.

ERR.BP If the line parameter value is invalid.

4.12 CALL TIMING

4.12.1 INPUT PARAMETERS

D0.B Action Value	= #TH.CTIME
D1.W Line number	= Number of telephone line
D2.W On/Off	= 1 set timing on, 0 set timing off

4.12.2 OUTPUT PARAMETERS

None

Error Response BP : Bad parameter

4.12.3 DESCRIPTION

This procedure will be used, at configuration time, to switch call timing on or off on the specified telephone line.

4.13 ENQUIRE

4.13.1 INPUT PARAMETERS

DO.B Action Value	= #TH.ENQU
D1.W Line Number	= Number of telephone line
D2.W Event	= Event number as a binary number, or -1
D3.W Line State Mask	= Mask with bits set as below for line/resource states to be monitored

4.13.2 OUTPUT PARAMETERS

DO.W Line state	= Current state of line.
D1.W Line Resourcing	= Current resources of line (see below)

Error Responses BP : Bad parameter
BO : Buffer overflow
NT : Not a valid telephone line

4.13.3 DESCRIPTION

There are two modes of use of this procedural interface.

1. The procedure may be called in order to determine the current state and resourcing of the specified telephone line. In this mode the event number parameter must be -1 and the line state mask is ignored. For this mode the current line state is returned as bit settings in DO, and resourcing similarly in D1, as defined below.

The line number parameter is validated and if it is valid, the current line state is returned in DO.W with settings defined as follows:-

The line state bit definitions in DO:

```
TH.ONHK     EQU    0      ONHOOK
TH.ANSW     EQU    3      ANSWER (an incoming call has been
                          answered)
TH.PRED     EQU    5      PREDIAL (line taken offhook but dialling
                          not started)
TH.DIAL     EQU    6      DIALLING
TH.PROM     EQU    7      PROMPTCB (TIMING key pressed and CB not
                          yet entered)
TH.SCDL     EQU    8      SCDIAL
TH.DIGDL    EQU    9      Digits still to be dialled
TH.AUTD     EQU   10      AUTODIAL
TH.TIMI     EQU   11      TIMING
```

4.15 CHECK CHARGE BAND

4.15.1 INPUT PARAMETERS

D0.B Action Value	= #TH.CHKCB
D3.W Charge Band Code	= 1/2 ASCII character, charge band code, left justified, right binary zero filled
A0.L Buffer	= Address of a frozen work area at least 28 bytes long to be used during CMOS record reads

4.15.2 OUTPUT PARAMETERS

None

Response in D0.L	= 0 if charge band found in CMOS
	= ERR.NF otherwise

4.15.3 DESCRIPTION

This procedure is provided for applications like Messaging which need to check whether a user supplied charge band code is legal, i.e. has been defined via configurator and exists in CMOS.

The procedure's check against charge bands stored in CMOS is case independent.

There is no check on the validity of the address in A0, nor its length.

4.14 SETUP DATA AUTO-ANSWER

4.14.1 INPUT PARAMETERS

DO.B Action Value = #TH.SETDA

4.14.2 OUTPUT PARAMETERS

None

4.14.3 DESCRIPTION

This procedure is provided for the use of the configurator only.

It is called by configurator when the name of the data auto-answer protocol application has been changed by the user.

The Telephone Handler's action is to send event 10 to any existing DAA protocol activity to cause it to abort, and to "Start Application" (see [3] for the new DAA protocol whose name is taken from CMOS record 7).

A success response is always returned by this procedure.

4.17 VOICE AA ENQUIRY

4.17.1 INPUT PARAMETERS

D0.B Action Value	= #TH.VAANQ
D1.L Message Identifier	= Voice auto-answer message identifier

4.17.2 OUTPUT PARAMETERS

None

Error Response NC : Message not currently in use

4.17.3 DESCRIPTION

This procedure is used by the Voice Auto-answer application to determine whether a specified message is currently being used by Telephone Handler for voice auto-answer.

If the message is in use currently then the procedure returns 0 in D0, else ERR.NC.

4.16 UPDATE NOTICEBOARD

4.16.1 INPUT PARAMETERS

D0.B Action Value	= #TH.UPNB
D1.W Line Number	= Telephone line 1 or 2, binary value

4.16.2 OUTPUT PARAMETERS

None

Error Responses BP : Bad parameter if D1 not 1 or 2

4.16.3 DESCRIPTION

This procedure is used to update the Noticeboard telephone status window for the specified line.

No action is taken if the specified line is already offhook.

The procedure is provided for the Voice Auto-Answer application to use when the user alters the auto-answer state on the line, to ensure changed state is immediately displayed in the Noticeboard.