Technical Proposal for

COMMON-ISDN-API

Version 2.0

Generic Tone Generator and Detector Support for Voice Applications

Extension

August 2009

Dialogic Corporation

COPYRIGHT NOTICE AND LEGAL DISCLAIMER

Fifth Edition (August 2009) **206-462-05**

All contents of this document are furnished for informational use only and are subject to change without notice and do not represent a commitment on the part of Dialogic Corporation or its subsidiaries ("Dialogic"). Reasonable effort is made to ensure the accuracy of the information contained in the document. However, Dialogic does not warrant the accuracy of this information and cannot accept responsibility for errors, inaccuracies or omissions that may be contained in this document.

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH DIALOGIC® PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN A SIGNED AGREEMENT BETWEEN YOU AND DIALOGIC, DIALOGIC ASSUMES NO LIABILITY WHATSOEVER, AND DIALOGIC DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF DIALOGIC PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY INTELLECTUAL PROPERTY RIGHT OF A THIRD PARTY.

Dialogic products are not intended for use in medical, life saving, life sustaining, critical control or safety systems, or in nuclear facility applications.

Due to differing national regulations and approval requirements, certain Dialogic products may be suitable for use only in specific countries, and thus may not function properly in other countries. You are responsible for ensuring that your use of such products occurs only in the countries where such use is suitable. For information on specific products, contact Dialogic Corporation at the address indicated below or on the web at *www.dialogic.com*.

It is possible that the use or implementation of any one of the concepts, applications, or ideas described in this document, in marketing collateral produced by or on web pages maintained by Dialogic may infringe one or more patents or other intellectual property rights owned by third parties. Dialogic does not provide any intellectual property licenses with the sale of Dialogic products other than a license to use such product in accordance with intellectual property owned or validly licensed by Dialogic and no such licenses are provided except pursuant to a signed agreement with Dialogic. More detailed information about such intellectual property is available from Dialogic's legal department at 9800 Cavendish Blvd., 5th Floor, Montreal, Quebec, Canada H4M 2V9. **Dialogic encourages all users of its products to procure all necessary intellectual property licenses required to implement any concepts or applications and does not condone or encourage any intellectual property infringement and disclaims any responsibility related thereto. These intellectual property licenses may differ from country to country and it is the responsibility of those who develop the concepts or applications to be aware of and comply with different national license requirements.**

Dialogic, Dialogic Pro, Brooktrout, Diva, Cantata, SnowShore, Eicon, Eicon Networks, NMS Communications, NMS (stylized), Eiconcard, SIPcontrol, Diva ISDN, TruFax, Exnet, EXS, SwitchKit, N20, Making Innovation Thrive, Connecting to Growth, Video is the New Voice, Fusion, Vision, PacketMedia, NaturalAccess, NaturalCallControl, NaturalConference, NaturalFax and Shiva, among others as well as related logos, are either registered trademarks or trademarks of Dialogic Corporation or its subsidiaries. Dialogic's trademarks may be used publicly only with permission from Dialogic. Such permission may only be granted by Dialogic's legal department at 9800 Cavendish Blvd., 5th Floor, Montreal, Quebec, Canada H4M 2V9. Any authorized use of Dialogic's trademarks will be subject to full respect of the trademark guidelines published by Dialogic from time to time and any use of Dialogic's trademarks requires proper acknowledgement.

The names of actual companies and products mentioned herein are the trademarks of their respective owners.

Motivation:

For some applications, such as the recognition of the "Beep" of an answering machine, it is required to detect the presence of a tone or a set of tones on a voice connection. Since the frequency of such tones is often not known in advance, the detector should work in a way that it determines tones in the signal and reports their frequency.

It is also desirable to support generation of various tones and signals on a voice connection to provide a caller with information or feedback for an action. A generic approach with programmable signal generators would give a high degree of flexibility for the application.

Since these features are intended to be used in server applications, it is desirable that a **COMMON-ISDN-API** controller provides built-in generic tone detector and generator facilities.

The following **COMMON-ISDN-API** extension defines the support for generic tone detection and generation for bit transparent connections. This is done by issuing manufacturer requests *Generic Tone Control* with the function *Enable Generic Tone Operation* when the physical connection is established. An application may search the manufacturer string for "Eicon" and issue a manufacturer request *Generic Tone Control* with the function *Get Supported Services* for a **COMMON-ISDN-API** controller to determine whether it supports generic tone generator and detector facilities and which services are available in which parameter ranges. The parameters required to control each generic tone service are passed via manufacturer request messages *Enable Generic Tone Operation*. The command results are reported via manufacturer confirmation. Generic tone detector events are reported to the **COMMON-ISDN-API** application via manufacturer indication.

The currently defined generic tone services are:

- Sine generators with programmable frequency and amplitude modulation
- Function generators with programmable signal shape, frequency, and amplitude modulation
- Noise generators with programmable crest factor and amplitude modulation
- Universal single tone detection
- Universal dual tone detection

Manu ID (dword)

The parameter *Manu ID* is to communicate a dword, which identifies the manufacturer in MANUFACTURER messages. Every manufacturer supplying MANUFACTURER messages should choose a unique value (such as an abbreviation of the company name).

The manufacturer ID used by Dialogic is:

0x44444944

This information element appears in:

MANUFACTURER_REQ
MANUFACTURER_RESP
MANUFACTURER_CONF
MANUFACTURER_IND

Manufacturer-Specific

The parameter *manufacturer specific* exchanges manufacturer-specific information.

Manufacturer specific information for MANUFACTURER_REQ:

word	manufacturer command	Manufacturer-specific operation requested.
struct	manufacturer command parameters	Command-dependent parameters for manufacturer request.

Manufacturer specific information for MANUFACTURER_CONF:

word	manufacturer command	Manufacturer-specific operation that was requested.
word	info	Result of the operation according to **COMMON-ISDN-API** definition of Info.
struct	manufacturer confirmation parameters	Command-dependent parameters for manufacturer confirmation.

Manufacturer specific information for MANUFACTURER_IND:

word	manufacturer indication	Manufacturer-specific indication code.
struct	manufacturer indication parameters	Indication-dependent parameters of manufacturer indication.

Manufacturer-specific information for MANUFACTURER_RESP:

word	manufacturer indication	Manufacturer-specific indication code where this response belongs to.
struct	manufacturer response parameters	Indication-dependent parameters for response to manufacturer indication.

This information element appears in:

MANUFACTURER_REQ
MANUFACTURER_RESP
MANUFACTURER_CONF
MANUFACTURER_IND

Manufacturer Command

The parameter *manufacturer command* specifies the kind of operation requested in a MANUFACTURER_REQ.

The following manufacturer commands are defined:

1:	**Assign PLCI**
2:	**Advanced Codec control**
3:	**DSP control**
4:	**Signaling control**
5:	**RXT control**
6:	**IDI control**
7:	**Configuration control**
8:	**Remove Codec**
9:	**Options request**
10..14:	**reserved**
15:	**Generic Tone Control**

The *Generic Tone Control* command activates and deactivates generic tone generator and detector facilities on the physical connection specified by the PLCI of the MANUFACTURER_REQ. These facilities are only available with the bit transparent B protocol. The *generic tone control function Get Supported Services* may be used to determine whether the controller specified by the controller number in the MANUFACTURER_REQ provides generic tone generator and detector facilities.

This information element appears in:

Manufacturer Specific

Manufacturer Command Parameters

The parameter *manufacturer command parameters* specifies command-dependent parameters.

Parameters for manufacturer command 15: Generic Tone Control:

word	Generic tone control function	Functions to control generic tone generator and detector operation: 0: Get supported services 1: Enable generic tone operation 2: Disable generic tone operation
	sequence of generic tone requests	

This information element appears in:

Manufacturer Specific

Manufacturer Confirmation Parameters

The parameter *manufacturer confirmation parameters* specifies command-dependent confirmation parameters.

Parameters for confirmation to manufacturer command 15: Generic Tone Control:

word	Generic tone control function	Functions to control generic tone generator and detector operation: 0: Get supported services 1: Enable generic tone operation 2: Disable generic tone operation
	sequence of generic tone confirmations	

This information element appears in:

Manufacturer Specific

Manufacturer Indication Parameters

The parameter *manufacturer indication parameters* specifies command dependent indication parameters.

Parameters for indication to manufacturer command 15: Generic Tone Control:

word	Generic tone control function	Functions to control generic tone generator and detector operation: 0: Get supported services 1: Enable generic tone operation 2: Disable generic tone operation
	sequence of generic tone indications	

This information element appears in:

Manufacturer Specific

Manufacturer Response Parameters

The parameter *manufacturer response parameters* specifies command dependent response parameters.

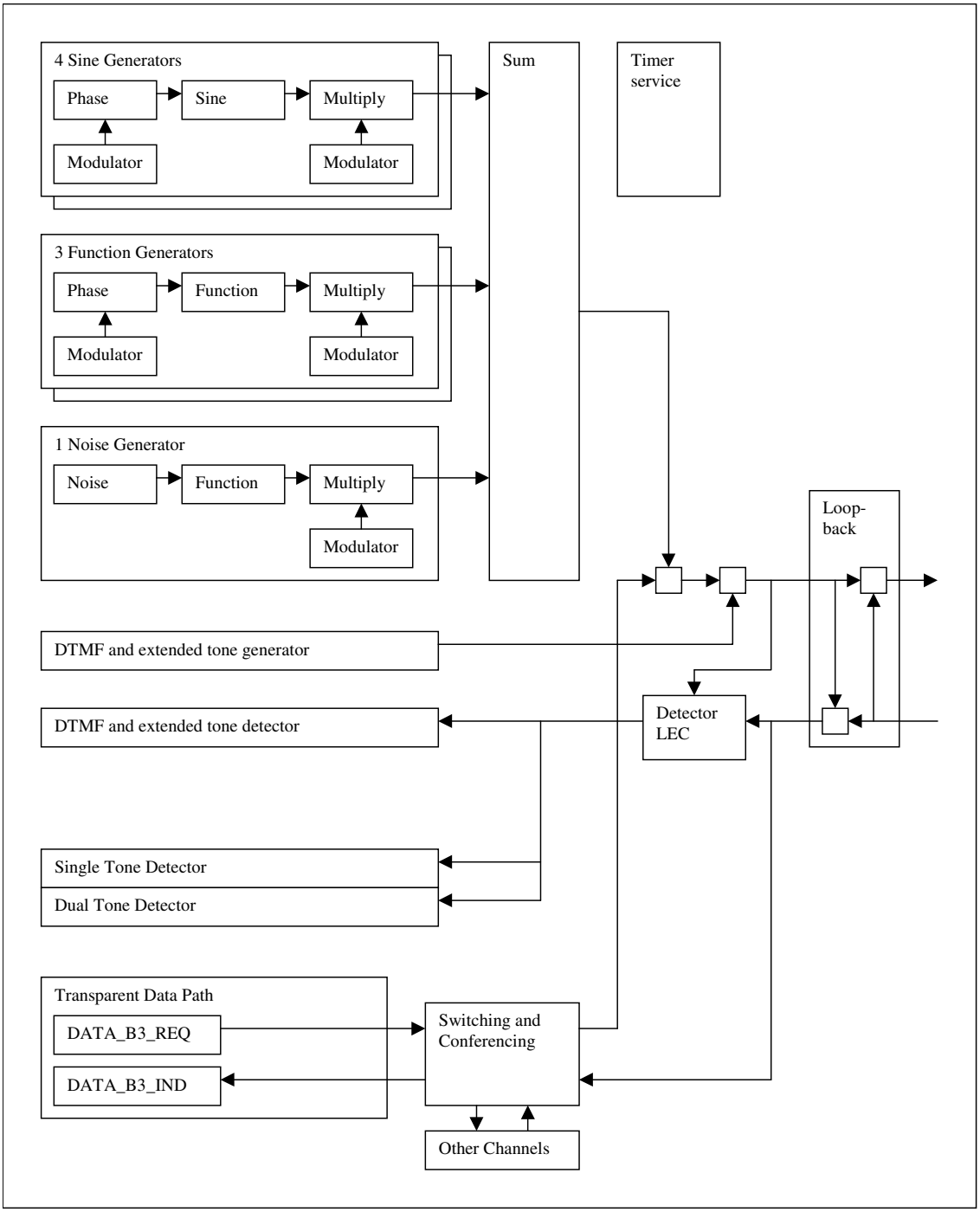
Parameters for response to manufacturer command 15: Generic Tone Control:

word	Generic tone control function	Functions to control generic tone generator and detector operation: 0: Get supported services 1: Enable generic tone operation 2: Disable generic tone operation
	sequence of generic tone responses	

This information element appears in:

Manufacturer Specific

ANNEX G: PARAMETERS FOR GENERIC TONE FACILITIES



Outline of the generic tone generator and detector system.

The **COMMON-ISDN-API** controller implements a generic tone generator and detector system as shown in the outline above.
A timer service can be set up to issue regular timeout indications. This enables an application to keep track of the elapsed time when timestamp counters wrap around.
The transmit section consists of a set of generators. Currently 4 sine generators, 3 function generators, and 1 noise generator are supported. The generators are controlled by programmable modulators. A modulator enables the application to specify in advance an arbitrary curve describing the value of the generator parameter in

relation to the time that passed. It is programmed with a set of points. Every point consists of a pair that specifies the value and the distance in time to the next point. The last point specifies the distance to the first point. An infinite distance can be set when appropriate. Linear interpolation is applied to form the shape of the curve. Modulators can be freely assigned to the generator parameters, e.g., for amplitude and frequency modulation.

The shape of the function generator signal and the post processing of noise generator samples can be defined via programmable functions. A function is defined by a set of points. Each point consists of a pair that specifies the value at a position in the range 0x0000 to 0xffff. The continuous shape of the function is formed by linear interpolation between each point and the next wrapping around from the last point to the first. There is a set of functions, each of which can be assigned for the desired purpose.

The noise generator calculates a stream of equally distributed samples. With a post-processing function, the application can influence the distribution, e.g., to meet a desired crest factor requirement. All generators can be enabled with one request and they will start in coincidence. If any generator is enabled, the data feed via DATA_B3_REQ will be overridden by the sum of the generator signals.

In the receive direction, the signal is fed into the detectors. The single tone detector is able to find single tone signals that fulfill configurable requirements for minimum duration, minimum signal to noise ratio (SNR), minimum signal level, maximum amplitude modulation and maximum frequency modulation. When a tone is detected, the precise start time is reported, and depending on the configuration, the measured SNR, mean signal level, mean frequency, and the variation in signal level and frequency are reported as well. On termination of the tone, a second indication is generated with the tone end timestamp. The dual tone detector provides a similar service to find dual tone signals. The minimum duration, minimum SNR, minimum signal level, maximum high-low twist, and maximum low-high twist are configurable parameters, where high-low twist signifies the difference in signal level between the higher frequency tone and the lower frequency tone. When a dual tone is detected, the start time is reported, and depending on the configuration, the measured SNR, the signal level and frequency of each tone are reported as well. The tone end is reported with a second indication. Note that the tone start and end timestamps are 16 bit values that wrap around. A periodic timeout from the timer service about every 1 or 2 seconds is an acceptable means to extend this timestamp if necessary.

The signal of the transparent data path transferred via DATA_B3_REQ and DATA_B3_IND passes the switching and conferencing block. The transmit signal is fed to the B-channel if it is not overridden by the signal generators of the generic tone generator facility or from the regular DTMF and extended tone generator. The receive signal is taken from the current B-channel. Before it is fed into the detectors, it passes a line echo canceller to avoid unwanted detections due to echoes of its own outbound signal.

Generic Tone Request

The parameter *generic tone request* controls a generic tone generator and detector service and provides the required parameters.

Generic tone request

byte	service request	Service request number
struct	Service request parameters	Service request dependent parameters.

The following service requests are defined for generic tone control function 0: Get Supported Services

0: Timer support
96: Single tone detector support
100: Dual tone detector support
144: Generator support
148: Modulator support
152: Function support
160: Sine generator support
164: Function generator support
168: Noise generator support

Service request parameters for service request 0: Timer support

coded as an empty struct

Service request parameters for service request 96: Single tone detector support

coded as an empty struct

Service request parameters for service request 100: Dual tone detector support

coded as an empty struct

Service request parameters for service request 144: Generator support

coded as an empty struct

Service request parameters for service request 148: Modulator support

coded as an empty struct

Service request parameters for service request 152: Function support

coded as an empty struct

Service request parameters for service request 160: Sine generator support

coded as an empty struct

Service request parameters for service request 164: Function generator support

coded as an empty struct

Service request parameters for service request 168: Noise generator support

coded as an empty struct

The following service requests are defined for generic tone control function 1: Enable Generic Tone Operation

1: Timer setup
97: Single tone detector setup
101: Dual tone detector setup
145: Generator setup
149: Modulator setup
153: Function setup
161: Sine generator setup
165: Function generator setup
169: Noise generator setup

Service request parameters for service request 1: Timer setup

word	Instance	Instance to address
word	Timer period	Time interval for timeout indications: 0: Stop timeout indications 1: Indicate timeout every 0.125 ms ... 64000: Indicate timeout every 8000 ms

Service request parameters for service request 97: Single tone detector setup

word	Instance	Instance to address
word	Reserved	Must be coded as 0
word	Single tone result flags	Result values that should be included in every result packet: [Bit 0]: Reserved [Bit 1]: Mean SNR [Bit 2]: Mean energy [Bit 3]: Mean frequency [Bit 4]: Energy variation [Bit 5]: Frequency variation [Bit 6..15]: Reserved
word	Minimum duration	Minimum tone duration in 8kHz samples: 0: 0ms ... 64000: 8000 ms
word	Minimum SNR	Minimum required signal to noise ratio: 0x8000: -128 dB ... 0x7fff: 127.996 dB
word	Minimum level	Minimum required signal level: 0x8000: -infinity 0x8001: -127.996 dBm ... 0x7fff: 127.996 dBm
word	Maximum amplitude modulation	Maximum allowed variation in signal level: 0: 0 dB ... 0xffff: 255.996 dB
word	Maximum frequency modulation	Maximum allowed variation in signal frequency: 0: 0 Hz ... 32000: 4000 Hz

Service request parameters for service request 101: Dual tone detector setup

word	Instance	Instance to address
word	Reserved	Must be coded as 0
word	Dual tone result flags	Result values that should be included in every result packet: [Bit 0]: Reserved [Bit 1]: Mean SNR [Bit 2]: Energy of lower frequency tone [Bit 3]: Energy of higher frequency tone [Bit 4]: Frequency of lower frequency tone [Bit 5]: Frequency of higher frequency tone [Bit 6..15]: Reserved
word	Minimum duration	Minimum tone duration in 8kHz samples: 0: 0ms ... 64000: 8000 ms
word	Minimum SNR	Minimum required signal to noise ratio of the sum of both tones: 0x8000: -128 dB ... 0x7fff: 127.996 dB
word	Minimum level	Minimum required signal level for each tone: 0x8000: -infinity 0x8001: -127.996 dBm ... 0x7fff: 127.996 dBm
word	Maximum high-low twist	Maximum allowed difference between higher and lower frequency tone: 0x8000: -infinity 0x8001: -127.996 dB ... 0x7fff: 127.996 dB
word	Maximum low-high twist	Maximum allowed difference between lower and higher frequency tone: 0x8000: -infinity 0x8001: -127.996 dB ... 0x7fff: 127.996 dB

Service request parameters for service request 145: Generator setup

word	Instance	Instance to address
word	Enabled sine generators	Bit mask with bits set to 1 for every sine generator that should be enabled. If any generator is enabled, the transmit signal feed via DATA_B3_REQ will be overridden.
word	Enabled function generators	Bit mask with bits set to 1 for every function generator that should be enabled. If any generator is enabled, the transmit signal feed via DATA_B3_REQ will be overridden.
word	Enabled noise generators	Bit mask with bits set to 1 for every noise generator that should be enabled. If any generator is enabled, the transmit signal feed via DATA_B3_REQ will be overridden.

Service request parameters for service request 149: Modulator setup

word	Instance	Instance to address
word	Point offset	Offset of the first curve point in case the modulation curve is constructed in multiple requests.
word	Number of points	Number of modulation curve points that follow.
	Sequence of modulation curve points	Curve points that describe the desired modulation curve. Linear interpolation between each point and the next and between the last point and the first are applied in order to form a continuous shape.

Modulation curve point:

word	Distance	Distance to the next curve point (or the first if no next is specified): 0x0000: infinite 0x0001: 0.125 ms ... 0xffff: 8191.875 ms
word	Value	Value of the curve point at the beginning of the interval: If used as frequency modulator, it specifies delta phi 0x8000: -180.000 degree ... 0x7fff: +179.995 degree If used as amplitude modulator, it specifies the gain 0x8000: -1.00000 ... 0x7fff: +0.99997

Service request parameters for service request 153: Function setup

word	Instance	Instance to address
word	Point offset	Offset of the first curve point in case the function curve is constructed in multiple requests.
word	Number of points	Number of function curve points that follow.
	Sequence of function curve points	Curve points that describe the desired function curve. Linear interpolation between each point and the next and wraparound from the last point to the first are applied in order to form a continuous shape.

Function curve point:

word	Position	Position of the curve point, phi: 0x0000: 0 degree ... 0xffff: 359,995 degree
word	Value	Value of the curve point at the specified position: 0x8000: -1.00000 ... 0x7fff: +0.99997

Service request parameters for service request 161: Sine generator setup

word	Instance	Instance to address
word	Amplitude modulator ID	Instance number of the modulator to be used for amplitude modulation.
word	Frequency modulator ID	Instance number of the modulator to be used for frequency modulation.

Service request parameters for service request 165: Function generator setup

word	Instance	Instance to address
word	Function ID	Instance number of the function that forms the signal.
word	Amplitude modulator ID	Instance number of the modulator to be used for amplitude modulation.
word	Frequency modulator ID	Instance number of the modulator to be used for frequency modulation.

Service request parameters for service request 169: Noise generator setup

word	Instance	Instance to address
word	Function ID	Instance number of the function to be used to form the desired distribution, e.g. the crest factor.
word	Amplitude modulator ID	Instance number of the modulator to be used for amplitude modulation.

Generic Tone Confirmation

The parameter *generic tone confirmation* reports the result of the command issued to a generic tone generator and detector service.

Generic tone confirmation

byte	service confirmation	Service confirmation number
struct	Service confirmation parameters	Service confirmation dependent parameters

The following service confirmations are defined for generic tone control function 0: Get Supported Services

0: Timer support
96: Single tone detector support
100: Dual tone detector support
144: Generator support
148: Modulator support
152: Function support
160: Sine generator support
164: Function generator support
168: Noise generator support

Service confirmation parameters for service confirmation 0: Timer support

byte	Supported instances	Number of instances supported.
byte	Generic tone info	Result of the operation (see below)
word	Supported timer period	Maximum supported timer period in 0.125 ms units.

Service confirmation parameters for service confirmation 96: Single tone detector support

byte	Supported instances	Number of instances supported.
byte	Generic tone info	Result of the operation (see below)
word	Reserved	
word	Supported single tone result flags	Result values that might be requested: [Bit 0]: Reserved [Bit 1]: Mean SNR [Bit 2]: Mean energy [Bit 3]: Mean frequency [Bit 4]: Energy variation [Bit 5]: Frequency variation [Bit 6..15]: Reserved
word	Supported minimum tone duration	Supported minimum tone duration in 8kHz samples: 0: 0ms ... 64000: 8000 ms
word	Supported minimum SNR	Highest supported value for the minimum required SNR: 0x8000: -128 dB ... 0x7fff: 127.996 dB
word	Supported minimum level	Highest supported value for the minimum required signal level: 0x8000: -128 dBm ... 0x7fff: 127.996 dBm
word	Supported maximum amplitude modulation	Highest supported maximum amplitude modulation: 0: 0 dB ... 0xffff: 255.996 dB
word	Supported maximum frequency modulation	Highest supported maximum frequency modulation: 0: 0 Hz ... 32000: 4000 Hz

Service confirmation parameters for service confirmation 100: Dual tone detector support

byte	Supported instances	Number of instances supported.
byte	Generic tone info	Result of the operation (see below)
word	Reserved	
word	Supported dual tone result flags	Result values that might be requested: [Bit 0]: Reserved [Bit 1]: Mean SNR [Bit 2]: Energy of lower frequency tone [Bit 3]: Energy of higher frequency tone [Bit 4]: Frequency of lower frequency tone [Bit 5]: Frequency of higher frequency tone [Bit 6..15]: Reserved
word	Supported minimum tone duration	Supported minimum tone duration in 8kHz samples: 0: 0ms ... 64000: 8000 ms
word	Supported minimum SNR	Highest supported value for the minimum required SNR: 0x8000: -128 dB ... 0x7fff: 127.996 dB
word	Supported minimum level	Highest supported value for the minimum required signal level: 0x8000: -128 dBm ... 0x7fff: 127.996 dBm
word	Supported maximum high-low twist	Highest supported maximum high-low twist: 0: 0 dB ... 0xffff: 255.996 dB
word	Supported maximum low-high twist	Highest supported maximum low-high twist: 0: 0 dB ... 0xffff: 255.996 dB

Service confirmation parameters for service confirmation 144: Generator support

byte	Supported instances	Number of instances supported.
byte	Generic tone info	Result of the operation (see below)
word	Supported sine generator mask	Bit mask with bits set to 1 for every supported sine generator.
word	Supported function generator mask	Bit mask with bits set to 1 for every supported function generator.
word	Supported noise generator mask	Bit mask with bits set to 1 for every supported noise generator.

Service confirmation parameters for service confirmation 148: Modulator support

byte	Supported instances	Number of instances supported.
byte	Generic tone info	Result of the operation (see below)
word	Supported curve points	Maximum number of curve points supported in total for all modulator instances.

Service confirmation parameters for service confirmation 152: Function support

byte	Supported instances	Number of instances supported.
byte	Generic tone info	Result of the operation (see below)
word	Supported curve points	Maximum number of curve points supported in total for all function instances.

Service confirmation parameters for service confirmation 160: Sine generator support

byte	Supported instances	Number of instances supported.
byte	Generic tone info	Result of the operation (see below)

Service confirmation parameters for service confirmation 164: Function generator support

byte	Supported instances	Number of instances supported.
byte	Generic tone info	Result of the operation (see below)

Service confirmation parameters for service confirmation 168: Noise generator support

byte	Supported instances	Number of instances supported.
byte	Generic tone info	Result of the operation (see below)

The following service confirmations are defined for generic tone control function 1: Enable Generic Tone Operation

1: Timer setup
97: Single tone detector setup
101: Dual tone detector setup
145: Generator setup
149: Modulator setup
153: Function setup
161: Sine generator setup
165: Function generator setup
169: Noise generator setup

Service confirmation parameters for service confirmation 1: Timer setup

byte	Instance	Addressed instance
byte	Generic tone info	Result of the operation (see below)
word	Accepted timer period	Accepted timer period in 0.125 ms units. Timer timeout indications will follow in this interval.

Service confirmation parameters for service confirmation 97: Single tone detector setup

byte	Instance	Addressed instance
byte	Generic tone info	Result of the operation (see below)
word	Reserved	
word	Accepted single tone result flags	Result values that will be included in every result packet: [Bit 0]: Reserved [Bit 1]: Mean SNR [Bit 2]: Mean energy [Bit 3]: Mean frequency [Bit 4]: Energy variation [Bit 5]: Frequency variation [Bit 6..15]: Reserved
word	Accepted minimum tone duration	Accepted minimum tone duration in 8kHz samples: 0: 0ms ... 64000: 8000 ms
word	Accepted minimum SNR	Accepted minimum necessary SNR: 0x8000: -128 dB ... 0x7fff: 127.996 dB
word	Accepted minimum level	Accepted minimum necessary signal level: 0x8000: -128 dBm ... 0x7fff: 127.996 dBm
word	Accepted maximum amplitude modulation	Accepted maximum amplitude modulation: 0: 0 dB ... 0xffff: 255.996 dB
word	Accepted maximum frequency modulation	Accepted maximum amplitude modulation: 0: 0 Hz ... 32000: 4000 Hz

Service confirmation parameters for service confirmation 101: Dual tone detector setup

byte	Instance	Addressed instance
byte	Generic tone info	Result of the operation (see below)
word	Reserved	
word	Accepted dual tone result flags	Result values that will be included in every result packet: [Bit 0]: Reserved [Bit 1]: Mean SNR [Bit 2]: Energy of lower frequency tone [Bit 3]: Energy of higher frequency tone [Bit 4]: Frequency of lower frequency tone [Bit 5]: Frequency of higher frequency tone [Bit 6..15]: Reserved
word	Accepted minimum tone duration	Accepted minimum tone duration in 8kHz samples: 0: 0ms ... 64000: 8000 ms
word	Accepted minimum SNR	Accepted minimum necessary SNR: 0x8000: -128 dB ... 0x7fff: 127.996 dB
word	Accepted minimum level	Accepted minimum necessary signal level: 0x8000: -128 dBm ... 0x7fff: 127.996 dBm
word	Accepted maximum high-low twist	Accepted maximum high-low twist: 0: 0 dB ... 0xffff: 255.996 dB
word	Accepted maximum low-high twist	Accepted maximum low-high twist: 0: 0 dB ... 0xffff: 255.996 dB

Service confirmation parameters for service confirmation 145: Generator setup

byte	Instance	Addressed instance
byte	Generic tone info	Result of the operation (see below)
word	Accepted sine generator mask	Accepted bit mask with bits set to 1 for every enabled sine generator. All generators newly enabled in the same generator setup request start in coincidence.
word	Accepted function generator mask	Accepted bit mask with bits set to 1 for every enabled function generator. All generators newly enabled in the same generator setup request start in coincidence.
word	Accepted noise generator mask	Accepted bit mask with bits set to 1 for every enabled noise generator. All generators newly enabled in the same generator setup request start in coincidence.

Service confirmation parameters for service confirmation 149: Modulator setup

byte	Instance	Addressed instance
byte	Generic tone info	Result of the operation (see below)
word	Accepted curve points	Number of curve points accepted.

Service confirmation parameters for service confirmation 153: Function setup

byte	Instance	Addressed instance
byte	Generic tone info	Result of the operation (see below)
word	Accepted curve points	Number of curve points accepted.

Service confirmation parameters for service confirmation 161: Sine generator setup

byte	Instance	Addressed instance
byte	Generic tone info	Result of the operation (see below)

Service confirmation parameters for service confirmation 165: Function generator setup

byte	Instance	Addressed instance
byte	Generic tone info	Result of the operation (see below)

Service confirmation parameters for service confirmation 169: Noise generator setup

byte	Instance	Addressed instance
byte	Generic tone info	Result of the operation (see below)

Generic Tone Indication

The parameter *generic tone indication* reports events from a generic tone generator and detector service.

Generic tone indication

byte	service indication	Service indication number
struct	Service indication parameters	Service indication dependent parameters

The following service indications are defined for generic tone control function 1: Enable Generic Tone Operation

2: Timer timeout
98: Single tone start
99: Single tone end
102: Dual tone start
103: Dual tone end

Service indication parameters for service indication 2: Timer timeout

byte	Instance	Addressed instance
byte	Generic tone info	Result of the operation (see below)
word	Timestamp	Timestamp derived from the 8000 Hz sample clock starting with an arbitrary value.

Service indication parameters for service indication 98: Single tone start

byte	Instance	Addressed instance
byte	Generic tone info	Result of the operation (see below)
word	Timestamp	Timestamp when the tone started sampled from the 8000 Hz sample clock. This timestamp is correlated with the timestamp of the timer service.
word	Mean SNR	Only present if selected in the single tone detector setup: Mean SNR of the detected signal: 0x8000: -infinity 0x8001: -127.996 dB ... 0x7fff: 127.996 dB

word	Mean energy	Only present if selected in the single tone detector setup: Mean energy of the detected tone: 0x8000: -infinity 0x8001: -127.996 dBm ... 0x7fff: 127.996 dBm
word	Mean frequency	Only present if selected in the single tone detector setup: Mean frequency of the detected tone: 0: 0 Hz ... 32000: 4000 Hz
word	Amplitude modulation	Only present if selected in the single tone detector setup: Amplitude variation of the detected tone: 0: 0 dB ... 0xffff: 255.996 dB
word	Frequency modulation	Only present if selected in the single tone detector setup: Frequency variation of the detected tone: 0: 0 Hz ... 32000: 4000 Hz

Service indication parameters for service indication 99: Single tone end

byte	Instance	Addressed instance
byte	Generic tone info	Result of the operation (see below)
word	Timestamp	Timestamp when the tone stopped sampled from the 8000 Hz sample clock. This timestamp is correlated with the timestamp of the timer service.

Service indication parameters for service indication 102: Dual tone start

byte	Instance	Addressed instance
byte	Generic tone info	Result of the operation (see below)
word	Timestamp	Timestamp when the dual tone started sampled from the 8000 Hz sample clock. This timestamp is correlated with the timestamp of the timer service.
word	Mean SNR	Only present if selected in the dual tone detector setup: Mean SNR of the detected signal: 0x8000: -infinity 0x8001: -127.996 dB ... 0x7fff: 127.996 dB
word	Lower frequency tone energy	Only present if selected in the dual tone detector setup: Energy of the lower frequency tone: 0x8000: -infinity 0x8001: -127.996 dBm ... 0x7fff: 127.996 dBm
word	Higher frequency tone energy	Only present if selected in the dual tone detector setup: Energy of the higher frequency tone: 0x8000: -infinity 0x8001: -127.996 dBm ... 0x7fff: 127.996 dBm
word	Lower frequency tone frequency	Only present if selected in the dual tone detector setup: Frequency of the lower frequency tone: 0: 0 Hz ... 32000: 4000 Hz
word	Higher frequency tone frequency	Only present if selected in the dual tone detector setup: Frequency of the higher frequency tone: 0: 0 Hz ... 32000: 4000 Hz

Service indication parameters for service indication 103: Dual tone end

byte	Instance	Addressed instance
byte	Generic tone info	Result of the operation (see below)
word	Timestamp	Timestamp when the dual tone stopped sampled from the 8000 Hz sample clock. This timestamp is correlated with the timestamp of the timer service.

Generic Tone Response

The parameter *generic tone response* acknowledges events from a generic tone generator and detector service.

Generic tone response

byte	service response	Service response number
struct	Service response parameters	Service response dependent parameters

Currently, none of the generic tone indications requires that an acknowledge is sent.

Generic Tone Info

The parameter *generic tone info* reports error conditions for the commands issued to a generic tone generator and detector service.

Generic tone info

Success (0x00):
0x00 operation initiated successfully

Warings (0x01..0x7f):
[Bit 7] 0
[Bit 6..2] reserved
[Bit 1] parameters have been shrunk
[Bit 0] parameters have been ignored

Errors (0x80..0xff):
0x80..0xfb reserved
0xfc invalid parameter value
0xfd out of resources
0xfe parameters are missing
0xff unknown request