

Internet Draft  
Intended status: Experimental  
Expires: September 2019

M. Zeppelzauer  
A. Ringot  
St. Poelten UAS  
March 6, 2019

**SoniTalk: An Open Protocol for Data-Over-Sound Communication  
draft-zeppelzauer-data-over-sound-00.txt**

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of [BCP 78](#) and [BCP 79](#). This document may not be modified, and derivative works of it may not be created, except to publish it as an RFC and to translate it into languages other than English.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/ietf/1id-abstracts.txt>

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>

This Internet-Draft will expire on September 6, 2019.

Copyright Notice

Copyright (c) 2019 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must

include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Abstract

This document defines a new protocol for communication via sound (and in particular via near-ultrasound) that is simple enough to be implemented on devices with limited computational resources, such as Internet-of-Things (IoT) devices. The near-ultrasonic frequency band in the range of 18-22kHz represents a novel and so far hardly used channel for the communication of different devices, such as mobile phones, computers, TVs, personal assistants, and potentially a wide range of IoT devices. Moreover, data-over-sound enables to connect low-end hardware devices to the Internet by near field communication with other Internet-connected devices. Data-over-sound requires only a standard loudspeaker and a microphone for communication, and thus has very low hardware requirements compared to other communication standards such as Bluetooth, WLAN and NFC. "SoniTalk" is designed as an open and transparent near-ultrasonic data transmission protocol for data-over-sound. This document provides a specification of the protocol at the lowest layer (physical layer) in the sense of the OSI model.

Table of Contents

- [1. Introduction.....2](#)
- [2. Details.....3](#)
- [3. Security Considerations.....6](#)
- [4. IANA Considerations.....7](#)
- [5. Conclusions.....7](#)
- [6. References.....7](#)
  - [6.1. Normative References.....7](#)
  - [6.2. Informative References.....7](#)
- [7. Acknowledgments.....7](#)
- [Authors' Addresses.....9](#)

**1. Introduction**

The typical frequency band for data-over-sound starts at 18kHz. This band can be corrupted by noise from the environment, which requires a number of counter measures to ensure a robust signal transmission. Especially the temporally varying characteristics of the channel makes the transmission of messages over longer time-spans more likely

to be corrupted. The proposed protocol tries to mitigate these sources of error by including redundancy in the encoding.  
Redundancy

Zeppelzauer, Ringot Expires September 6, 2019

[Page 2]

is generated by encoding each bit in terms of a Manchester code with a transition from high to low (and vice versa) for bit 1 and 0, respectively. This type of redundancy makes the code not only more robust, but also enables a simpler decoding of the message. To minimize the temporal message duration and maximize data rate, information is sent in multiple channels in parallel.

## 2. Details

Data in the protocol is represented by individual messages. Each message is represented by an acoustic signal that encodes the information contained in the message. A message has a temporal and

a

spectral dimension, i.e., a two-dimensional layout in terms of frequency and time (see Figure 1). Along the temporal dimension, a message is composed of several consecutive blocks. Each message starts with a "start block", followed by M "message blocks" and an "end block". Each message block has a duration of D ms. The start- and end blocks have a duration of D/2 ms. Each block spans multiple carrier frequencies  $F_i$ , where  $F_i$  in  $\{F_1, F_2, \dots, F_C\}$  are C equally-spaced carrier frequencies covering a frequency band of  $B = F_C - F_1$

Hz.

The spacing of the frequencies is  $S = B/(C-1)$  Hz. Each bit in the message can be addressed by a block number and carrier frequency. This layout allows for sending information in parallel on multiple frequencies.

Information is encoded binary. Each message block encodes one bit

at

each carrier frequency  $F_i$ . For a logical "1" the amplitude of the first D/2 ms at frequency  $F_i$  of the block is "high" and the

amplitude

of the second D/2 ms is zero. For a logical "0" the opposite is the case, i.e., the amplitude of the first D/2 ms at carrier frequency

$F_i$

of the block is zero and the amplitude of the second D/2 ms is "high". The magnitude of "high" amplitude is not normative and depends on the actual use case, employed hardware and the targeted transmission range.

The binary message content is encoded across the carrier frequencies (from lowest to highest frequency, i.e.  $F_1$  to  $F_C$ ) starting with the first message block, i.e. the first bit is encoded at message block

1

and carrier frequency  $F_1$ , the second bit is located at message block 2 and carrier frequency  $F_2$ , etc.

Between two message blocks and in the middle of each block (i.e. after the first D/2 ms of a message block) a pause can be inserted

of

duration P with  $P \geq 0$ . For a pause, the sending amplitude is set

to

Zeppelzauer, Ringot Expires September 6, 2019

[Page 3]

zero. The overall message duration is thus:  $D/2 + P + D*M + P*(2*M-1) + P + D/2 = D*(M+1) + P(2*M+1)$  ms.

The first and last blocks of a message represent the start- and end blocks. Start and end blocks are represented by the following encoding: a start block has "high" amplitude at the higher  $C/2$  frequencies ( $C/2$  rounded up in case  $C$  is an odd number) and zero amplitude at the remaining frequencies. For the end blocks the opposite is the case, i.e. "high" amplitude is present at the lower  $C/2$  ( $C/2$  rounded down) carrier frequencies and zero amplitude for the remaining frequencies.

From the above specification it follows that the number of bits that can be represented by a message is:  $M*C$ . The theoretical maximal data rate corresponds to  $1000 / (D*(M+1) + P*(2*M+1)) * (M*C)$  bits per second.

The schematic two-dimensional spectro-temporal layout (time at the x-axis and frequency on the y-axis) of a message for parameters:

$M=4$  blocks,

$C=8$  frequencies,

$D=2$  (corresponding to the spacing of 2 characters along the temporal axis: "--"),

$P=4$  (corresponding to the spacing of 4 characters along the temporal axis: "----"),

encoding the following binary information:

"01010011 01101111 01101110 01101001"

is provided in the following. Character "+" indicates "high" amplitude and "0" indicates zero amplitude. Pause periods are indicated with the following pattern "...." for better visibility:

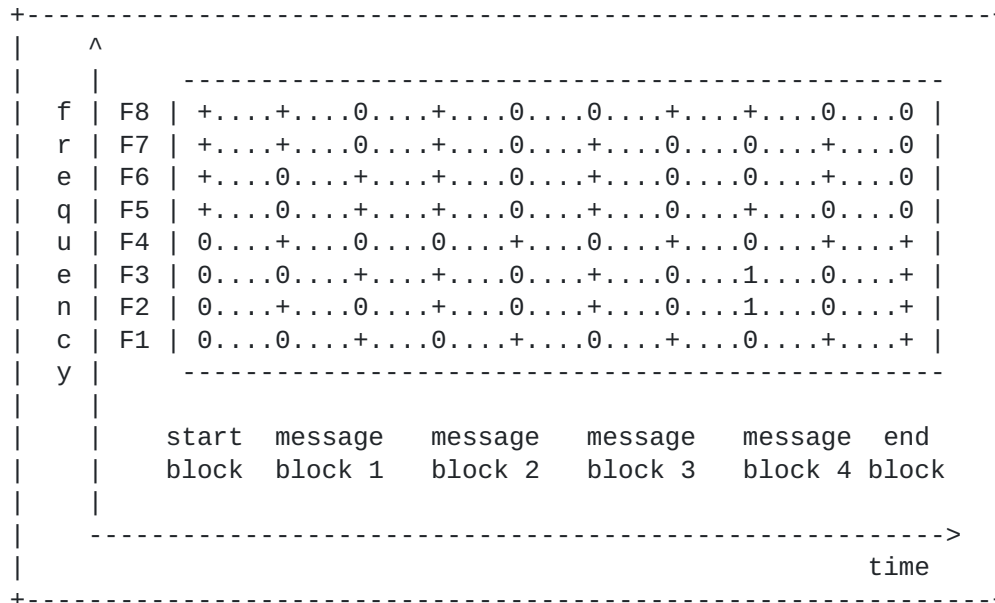


Figure 1 The spectro-temporal layout of a single message, "msg"

Note, the first eight bits of the message are encoded by the first half of message block 1 from low to high frequency. The second half of message block 1 represents the inverted information. The second eight bits are encoded in the first half of message block 2 from low to high frequency, etc.

Different profiles (configurations) of the protocol can be defined to adapt it to the specific requirements of the respective use-cases. The definition of a profile requires the following information:

D: the duration of a bit (i.e. a message block) in ms

P: the pause period in ms

F1: the lowest frequency in Hz

C: the number of frequencies

S: the spacing between successive frequencies  $F_i$  and  $F_{i+1}$  in Hz

M: the number of message blocks

### 3. Security Considerations

This specification is targeting solely the physical layer of the protocol. Thus SoniTalk itself provides no communications security, and therefore a large number of attacks are possible including replay attacks, sniffing, eavesdropping, denial of service attacks, message destruction and message insertion. A passive attack is sufficient to recover the binary information of messages transmitted with SoniTalk.

No endpoint authentication is provided by the protocol as this definition only targets the physical layer. Sender jamming is trivial, and therefore making messages unreadable is trivial. Attacks are however limited to the local environment around the communicating parties (usually within a few meters). If the communication takes place in a room, possible attacks are most likely successful from inside the room and unlikely from outside the room as near-ultrasonic signals hardly pass through walls.

Unlikely attacks are message deletion and message modification as this would require to acoustically manipulate the message while it is sent over the air. While it cannot be guaranteed with absolute certainty such attacks would be extremely difficult, e.g. sending interference sound to cancel out a message acoustically. Furthermore acoustically modifying individual bits of a message for message modification would require precise timing and would very likely destroy the integrity of the message since the acoustic overlay would introduce interferences.

To ensure data integrity the use of an error detecting (e.g. a CRC code) or an error correcting code is highly recommended when encoding the message. To establish confidentiality the binary message should further be encrypted, e.g. by a symmetric or asymmetric encryption scheme where the keys should be exchanged over an out-of-band channel (e.g. Bluetooth). Peer entity authentication is also not implemented at the physical layer and needs to be provided at a higher layer.

It is the particular duty of the developers of applications using the protocol to comprehensively inform the user about the near-ultrasonic data exchange (both sending and receiving) and moreover to inform the



users when personal information is sent over the protocol.

Particular care has to be taken in selecting the carrier frequencies for the data transmission so that no actively or passively participating party is disturbed by potential hearable artifacts of the acoustic data transmission. This in particular includes children as well as animals in the environment.

#### **4. IANA Considerations**

This document has no actions for IANA.

#### **5. Conclusions**

This internet draft introduces SoniTalk, which is the first open protocol for acoustic near field communication via the near-ultrasonic band. Near-ultrasound communication represents an alternative and complement to other existing near-field communication protocols, such as Bluetooth, radio-based NFC and WLAN and is particularly well-suited for IoT devices thanks to its low hardware requirements. This document specifies the protocol at the physical layer and thus primarily focuses on the definition of the message structure for information exchange. Extensions on top of this layer are subject to future specification efforts.

#### **6. References**

##### **6.1. Normative References**

##### **6.2. Informative References**

- [1] Hubert Zimmermann, OSI Reference Model - The ISO Model of Architecture for Open Systems Interconnection, IEEE Transactions on Communications, vol. 28, no. 4, April 1980, pp. 425-432

#### **7. Acknowledgments**

The work which led to this protocol specification was funded by netidee Open Innovations of the Internet Foundation Austria.

This document was prepared using 2-Word-v2.0.template.dot.

**Appendix A.****Scope and Remarks**

## A.1. Remarks

It is recommended to split the M\*C bits of a message into E parity bits for error detection and error correction and M\*C-E bits for the payload of the message. The size of the parity information is not normative and depends on the actual application (e.g. environmental conditions etc.)

The message length is fixed and must not vary. In case the specified message length is longer than the actual information to be sent, the remaining bits must be filled (e.g. by some special symbol) to comply with the protocol specifications.

## A.2. Out of Scope

The spacing of carrier frequencies, the actual height of the frequencies, the pause duration P inside a message as well as the spacing between successive messages is not part of this specification.

This protocol specification focuses exclusively on the lowest network layer (i.e. physical layer according to the OSI reference model [1]).

A protocol for distributing information across several messages, session handling, addressing, error detection and correction as well as synchronous and asynchronous communication is beyond this specification and subject to future norming initiatives.



**Appendix B.**                      **Comments and Feedback**

Please address all comments, discussions, and questions to  
matthias.zeppelzauer@fhstp.ac.at

Authors' Addresses

Matthias Zeppelzauer  
St. Poelten University of Applied Sciences  
Matthias Corvinus-Strasse 15, 3100 St. Poelten  
Austria

Email: matthias.zeppelzauer@fhstp.ac.at

Alexis Ringot  
St. Poelten University of Applied Sciences  
Email: alexis.ringot@fhstp.ac.at