IEEE 802.15

Wireless Specialty Networks

Project	IEEE 802.15 Working Group for Wireless Specialty Networks
Title	UWB Channel Models Document
Date Submitted	September 14th, 2022
Source	Takumi Kobayashi[1,2], Daisuke Anzai[4], Marco Hernandez[1,3], Ryuji Kohno[1,2], Minsoo Kim[2]; [1]YRP-IAI, [2]YNU, Japan, [3]CWC Oulu Univ. Finland, [4]Nagoya Institute of Technology, Japan. Marco.Hernandez@ieee.org Kohno@ynu.ac.jp Kobayashi-Takumi-ch@ynu.ac.jp Minsoo@minsookim.com Anzai@nitech.ac.jp
Response	To call for contributions
Abstract	This document contains the channel models to evaluate proposals.
Purpose	For contributions to P802.15.6ma
Notice	This document has been prepared to assist the IEEE P802.15.6ma. It is offered as a basis for discussion and is not binding on the contributing individual(s) or organization(s). The material in this document is subject to change in form and content after further study. The contributor(s) reserve(s) the right to add, amend or withdraw material contained herein.

Contributors

Name & affiliation	Email
Marco Hernandez, CWC, Oulu Univ. Finland, YRP-IAI Japan.	Marco.Hernandez@ieee.org
Ryuji Kohno, YRP-IAI, YNU, Japan	Kohno@ynu.ac.jp
Takumi Kobayashi, YRP-IAI, YNU, Japan	Kobayashi-Takumi-ch@ynu.ac.jp
Minsoo Kim, YRP-IAI, Japan.	Minsoo@minsookim.com
Kamran Sayrafian, NIST, USA.	kamran.sayrafian@nist.gov
Daisuke Anzai, Nagoya Institute of Technology, Japan.	Anzai@nitech.ac.jp

Revision history

Revision	Date	Notes
0	7/12/2022	Document created from 15-0344-03-006a plus adding edits to Section 6.2 and use cases S2.3. Removing S4.1, updates to S6.1 and Fig 3.
1	3/15/2023	S2.1 Option2, S2.2, S2.3, S6.2, S8, S8.1, S8.2, S9, S10, S11, S12, S13 and S14 models have been added and updated. Simulation MATLAB code program has been added.

Contents

1. Introduction

The channel models used by IEEE Std 802.15.6-2012 Wireless Body Area Network (BAN) during the development of the Std cover use cases for medical and non-medical applications inside or on the surface of the human body. The frequency bands of operation are ISM bands used by narrowband and UWB implementations [1] .

The channel models covered in this document are for the UWB band intended for body surfaces and vehicles. The channel models are aimed for the development of the PHY and MAC for the revision of IEEE 802.15.6 Std.

The channel model document of IEEE Std 802.15.6-2012 defined seven scenarios and four-channel models[1]   as shown in Table 1 for illustration.

Table 1—Channel models used in IEEE Std 802.15.6-2012

Scenario	Description	Frequency Band	Channel Model
S1	Implant to Implant	402-405 MHz	CM1
S2	Implant to Body Surface	402-405 MHz	CM2
S3	Implant to External	402-405 MHz	CM2
S4	Body Surface to Body Surface (LOS)	13.5, 50, 400, 600, 900 MHz 2.4, 3.1-10.6 GHZ	CM3
S5	Body Surface to Body Surface (NLOS)	13.5, 50, 400, 600, 900 MHz 2.4, 3.1-10.6 GHZ	CM3
S6	Body Surface to External (LOS)	900 MHz 2.4, 3.1-10.6 GHZ	CM4
S7	Body Surface to External (NLOS)	900 MHz 2.4, 3.1-10.6 GHZ	CM4

Figure 1 illustrates such channel models.
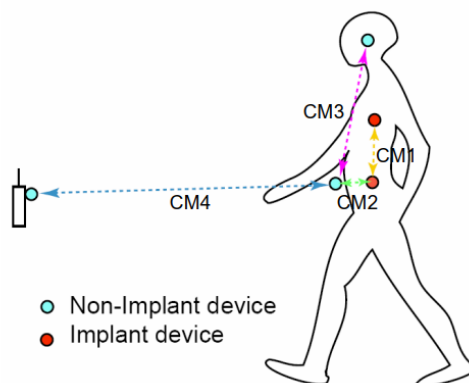
Figure 1—Channel models for IEEE Std 802.15.6-2012

2. Scenarios for 15.6 revision

2.1 Human BAN scenarios and channel models

Table 2—15.6ma HBAN scenarios (notation in [1]).

Scenario	Description	Frequency Band	Channel Model
S1	Implant to Implant	402-405 MHz	CM1
S2	Implant to Body Surface	402-405 MHz	CM2
S2.1	Implant (upper body) to Body Surface	3.1-10.6 GHz	CM2.1
S2.2	Implant (head) to Body Surface	3.1-10.6 GHz	CM2.2
S2.3	Implant (head) to external	3.1-10.6 GHz	CM2.3
S3	Implant to External	402-405MHz	CM3
S3.1	Implant (upper body) to External	3.1-10.6GHz	CM3.1
S4	Body Surface to Body Surface (LOS)	400, 600, 900 MHz 2.4, 3.1-10.6 GHz	CM3
S5	Body Surface to Body Surface (NLOS)	400, 600, 900 MHz 2.4, 3.1-10.6 GHz	CM3
S6	Body Surface to External (LOS)	900MHz, 2.4 GHz, 3.1-10.6 GHz	CM4
S6.1	Body Surface (head) to External (LOS)	3.1-10.6 GHz	CM6.1
S6.2	BAN coordinator to BAN coordinator	3.1-10.6 GHz	CM6.1
S7	Body Surface to External (NLOS)	900MHz , 2.4 GHz, 3.1-10.6 GHz	CM4

The present document describes the new channel models for scenarios S2.1, S2.2, S4.1 and S6.1.

2.2 Scenario 2.1

The new scenario S2.1 is for the implant (upper body) to body surface on the UWB band for applications such as capsule endoscopy.
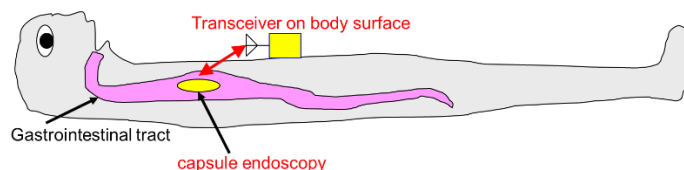
Figure 2—Scenario 2.1: Capsule endoscopy application

2.3 Scenario 2.2

The new scenario S2.2 is for the implant (head) to the body surface on the UWB band for applications such as the brain-computer interface (BCI). Note the transmitting antenna is integrated into the transmitter device over a titanium plaque that replaces the skull bone. In that way, the embedded transmitter is under the skin to avoid infection. Electrodes are implanted on the brain and join by a small cable to the transmitter device. The receiver may be located around the head (headband or other wearable) or torso.

The receiver could be located on the head by using head gear or something wearable to make easy to implementation of the antennas.
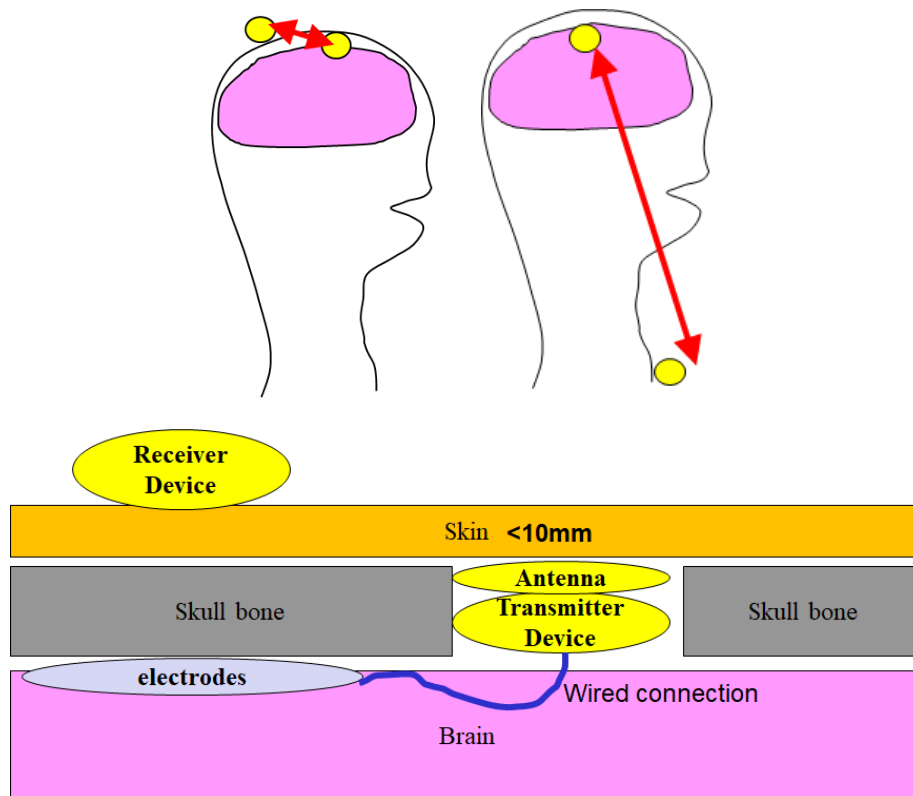
Figure 3—Scenario 2.2: implant (head) in-body to body surface.

2.4 Scenario 2.3

The new scenario S2.3 is for the implant (head) to external on the UWB band for applications such as the brain-computer interface (BCI). Note the transmitting antenna is integrated into the transmitter device over a titanium plaque. In that way, the embedded transmitter is under the skin to avoid infection.

The receiver may be located within a range of 2m in LOS.
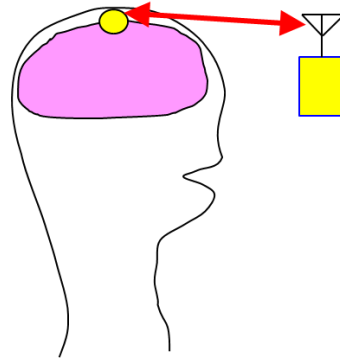
Figure 4—Scenario 2.3: implant (head) to external.

2.5 Scenario 6.1 Body Surface (head) to External (LOS)

The new scenario 6.1 is for the body surface (head) to external on the UWB band for applications such as the brain-computer interface (BCI).
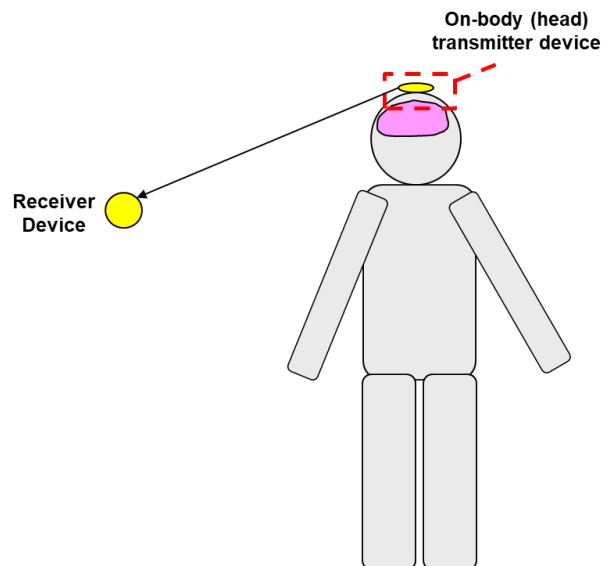
Figure 5—Scenario 6.1: body surface (head) to external.

2.6 Scenario 6.2 HBAN coordinator to HBAN coordinator

The new scenario 6.2 covers HBAN coordinator to HBAN coordinator.
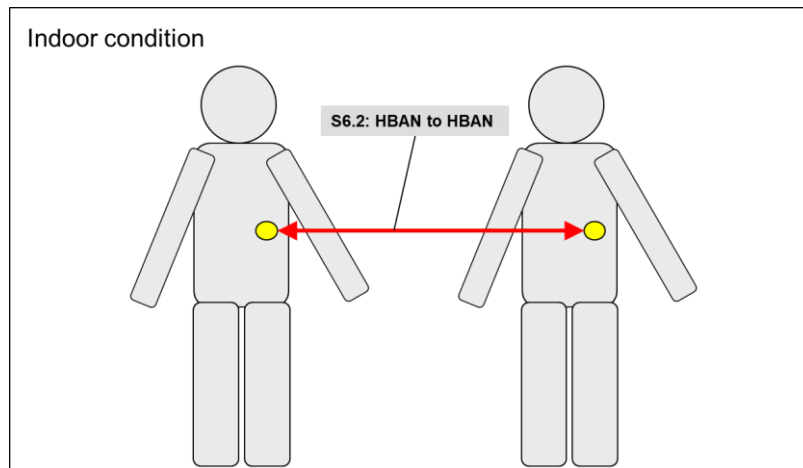
Figure 6—Scenario 6.2: HBAN coordinator to HBAN coordinator.

2.7 Vehicle BAN scenarios and channel models

Table 3—15.6ma VBAN scenarios

Scenario	Description	Frequency Band	Channel Model
S8	In-vehicle to In-vehicle (sedan)	2.4, 3.1-10.6 GHZ	CM8
S8.1	In-vehicle to In-vehicle (passenger bus)	2.4, 3.1-10.6 GHZ	CM8.1
S8.2	In-vehicle to In-vehicle (functional compartment)	2.4, 3.1-10.6 GHZ	CM8.2
S9	In-vehicle to On-vehicle	2.4, 3.1-10.6 GHZ	CM9
S10	In vehicle to External	2.4, 3.1-10.6 GHZ	CM10
S11	On-vehicle to on-vehicle (LOS)	2.4, 3.1-10.6 GHZ	CM11
S12	On-vehicle to on-vehicle (NLOS)	2.4, 3.1-10.6 GHZ	CM12
S13	On-vehicle to external (LOS)	2.4, 3.1-10.6 GHZ	CM13
S14	On-vehicle to external (NLOS)	2.4, 3.1-10.6 GHZ	CM14
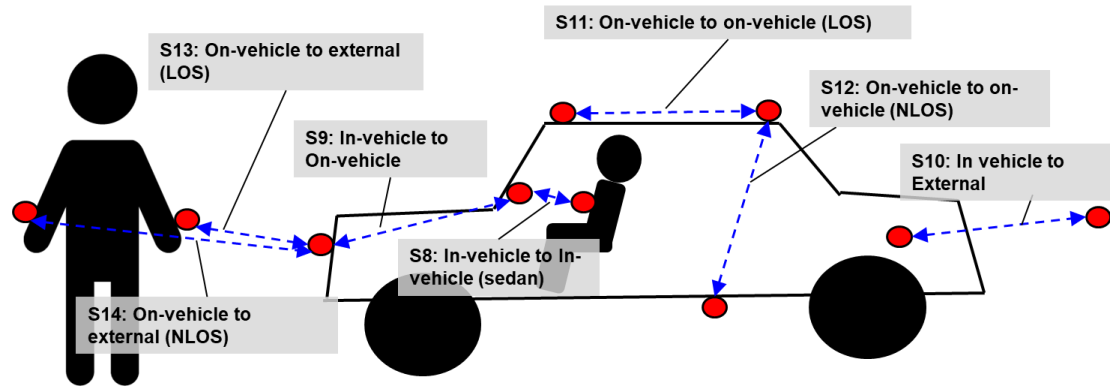
Figure 7—Channel models and scenarios for VBAN

Figure 7 shows an overview of models of VBAN applications.
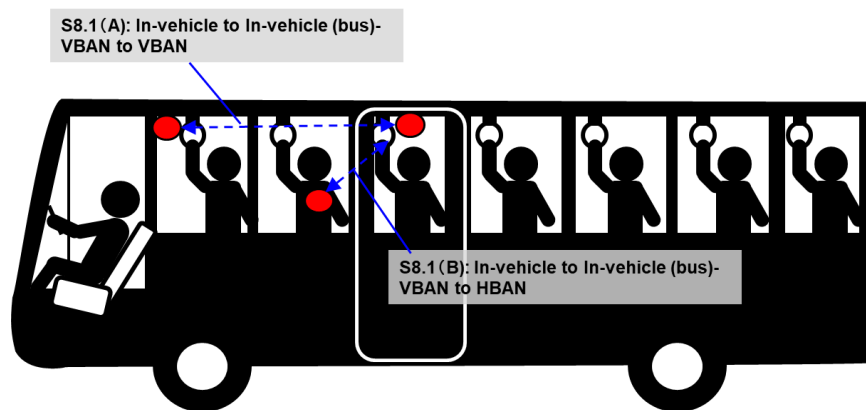
Figure 8—Scenario S8.1: In-vehicle to In-vehicle (bus)

Figure 8 shows scenario S8.1 which is targeting a use case in passenger bus. There are two scenarios S8.1(A) and S8.1(B). Scenario S8.1(A) illustrates the scenario VBAN to VBAN, while S8.1(B) illustrates the scenario of VBAN to HBAN.
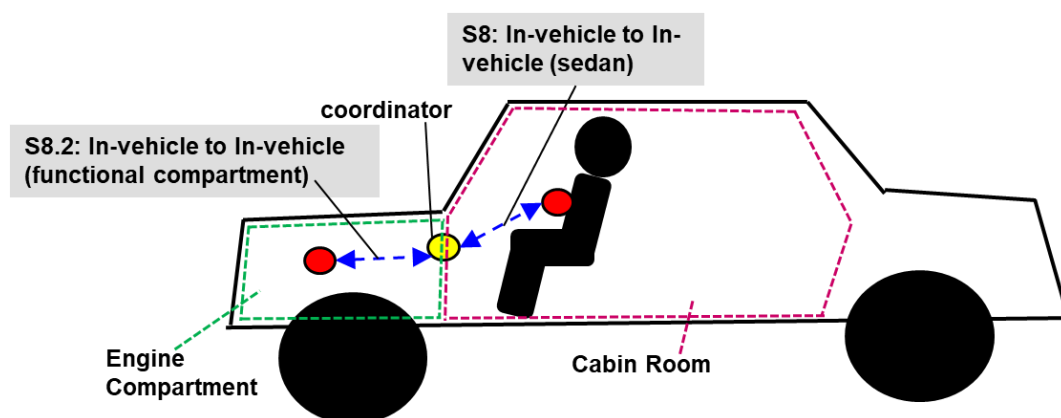
Figure 9—Scenario S8.2: In-vehicle to In-vehicle (functional compartment)

Figure 9 shows scenario S8.2, which can be used for the analysis of wireless connection between cabin's room and the other functional compartments such as the engine compartment and cargo space   as shown in Figure 10.
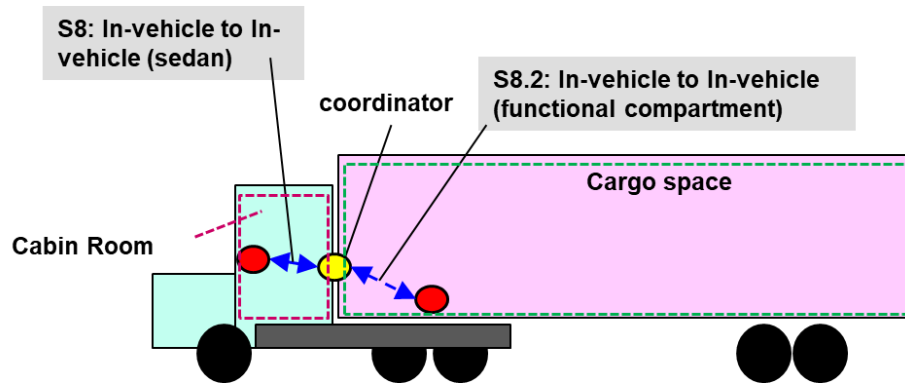
Figure 10 — Scenario S8.2: In-vehicle to In-vehicle (functional compartment) ; Cargo vehicle use case

In order to reduce disconnection between devices in the cabin's room and functional space room, a coordinator may be placed in between these two the cabin's room and cargo space.

3. Antenna Effect

An antenna placed on the surface or inside a body will be heavily influenced by its surroundings [2] . The consequent changes in antenna pattern and other characteristics need to be understood and accounted for during any propagation measurement campaign.

The form factor of an antenna will be highly dependent on the requirements of the application. For MICS applications, for example, a circular antenna may be suitable for a pacemaker implant, while a helix antenna may be required for a stent or urinary implant. The form factor will affect the performance of the antenna and, the antenna performance will be very important to the overall system performance. Therefore, an antenna which has been designed with respect to the body tissues (or considered the effect of human body) shall be used for the channel model measurements [3] .

3.1 Electrical antennas, such as dipole

Electrical antenna typically generates large components of E-field normal to the tissue interface, which overheat the fat tissue. This is because boundary conditions require the normal E-field at the interface to be discontinuous by the ratio of the permittivity, and since fat has a lower permittivity than muscle, the E-field in the fat tissue is higher.

3.2 Magnetic antennas, such as loop

Magnetic antenna produces an E-field mostly tangential to the tissue interface, which seem not to couple as strongly to the body as electrical antennas. Therefore, it does not overheat the fat.

There are antennas same as helical-coil, which is similar to a magnetic antenna in some respect, but its heating characteristics appear to be more like an electrical antenna. The strong E-field generated between the turns of coil is mainly responsible for tissue heating

It should be noted that SAR in the near field of the transmitting antenna depends mainly on the H-field; however, SAR in the far field of the transmitting antenna depends mainly on the E-field.

4. Channel Characterization

4.1 Electrical Properties of Body Tissues

The human body is not an ideal medium for radio frequency wave transmission. It is partially conductive and consists of materials of different dielectric constants, thickness, and characteristic impedance. Therefore, depending on the frequency of operation, the human body can lead to high losses caused by power absorption, central frequency shift, and radiation pattern destruction. The absorption effects vary in magnitude with both frequency of applied field and the characteristics of the tissue [[4] [5] [6] [7]].

4.2 Model Types

In all cases, two types of models may be generated:

— A theoretical or mathematical model including numerical simulation based modeling.

— An empirical model

A theoretical model may be traceable back to the fundamental principles of electromagnetic propagation and will permit precise modeling of a specific situation at radio link level. It is intended for detailed exploration of, for example, the influence of body structures on antenna patterns. It will require a detailed description of the propagation environment and is therefore probably not suitable for modeling of macro environments.

An empirical model may be traceable to an agreed set of propagation measurements and is intended to provide a convenient basis for statistical modeling of the channel. Compared to the theoretical model, the empirical model will use a greatly simplified description of the environment and, although statistically accurate at network level, will not be precise at link level. Appropriate efforts could be made to ensure that the two sets of models are consistent with each other.

4.3 Fading

In the body area network communications, propagation paths can experience fading due to different reasons, such as energy absorption, reflection, diffraction, shadowing by body, and body posture. The other possible reason for fading is multipath due to the environment around the body. Fading can be categorized into two categories; small scale and large-scale fading.
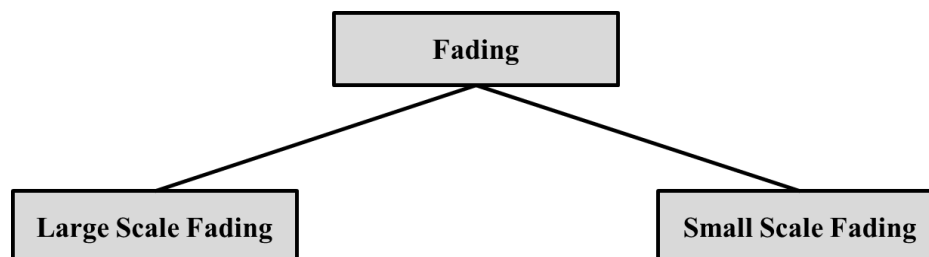
4.3.1 Small-Scale Fading

Small scale fading refers to the rapid changes of the amplitude and phase of the received signal within a small local area due to small changes in location of the on-body device or body positions, in a given short period of time. The small-scale fading can be further divided into flat fading and frequency selective fading.

Averaging the attenuation between each antenna position on the body and each antenna location in the room will remove the effect of small scale fading due to small changes in the body position.

4.3.1.1 UWB Channel model

The UWB channel impulse responses are taken from the channel models document of former IEEE802.15.4a TG. Scenarios, bandwidths and central frequencies are similar to the ones targeted for IEEE802.15.8 TG. The MatLab code is attach in Annex A.

4.3.1.2 Simulation methodology

The power delay profiles are taken from the extended ITU channel models described in [8] with Doppler spectrum characterized by the Jakes spectrum shape and maximum Doppler frequency shift, as suggested in [9] . The spatial correlation is described in [8] . Figure 11 shows the schematic flow for simulation of Kronecker model, already implemented in Matlab Simulink. In Figure 12, the correlated MIMO channel simulation is delineated.
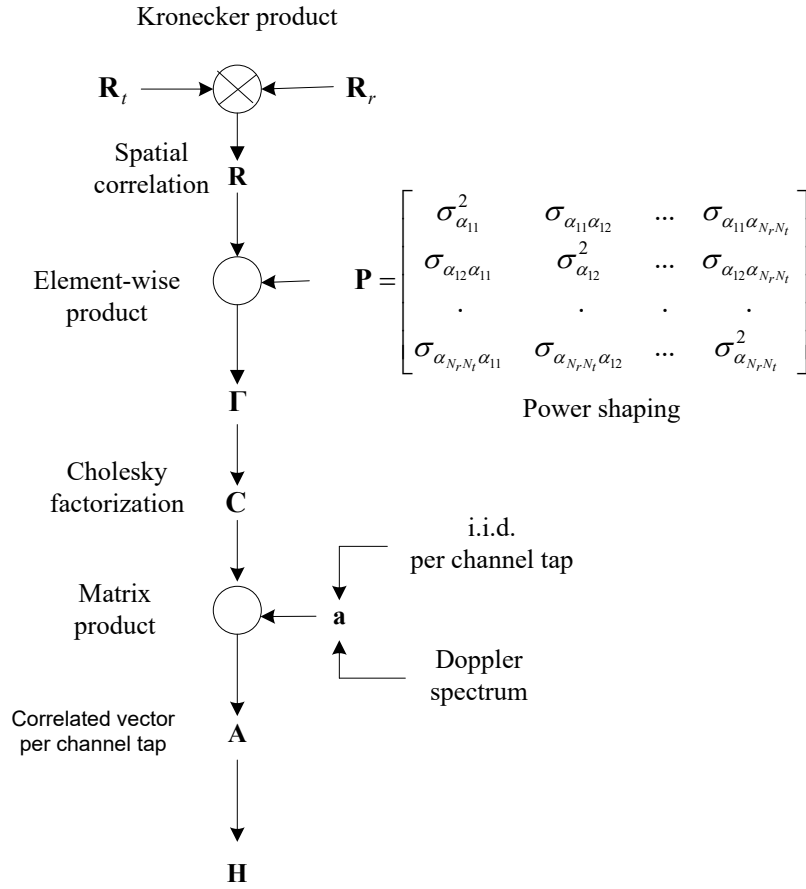
Figure 11 — Schematic flow of simulation for the Kronecker model.

where $\sigma^2_{\alpha_{ij}}(q)$ is the power of the channel coefficient of qth tap resulting between the coupling of ith antenna at transmitter to jth antenna at receiver.
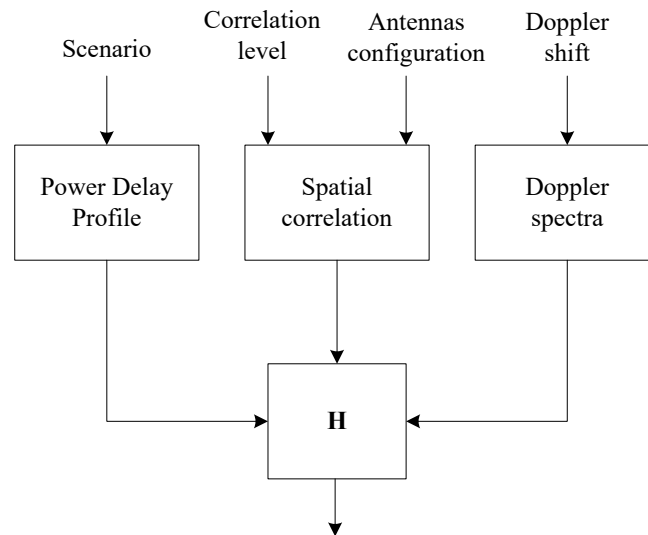
Figure 12 — Channel simulation methodology.

4.3.1.3 MatLab™ code

The following scripts were written in MatLabR2011 and the Communications Toolbox. It exploits the new interface between Matlab and Simulink for optimizing speed.

```matlab
%                                                                          %
% function [H] = ExtendedITU_MIMOChannel(model,correlation,               %
%                                      configuration,fd,fs)               %
%                                                                          %
% Return object for FIR taps of Extended ITU MIMO channel models          %
%                                                                          %
% INPUT PARAMETERS:                                                        %
%   model         = Extended ITU channel model                            %
%   correlation   = Spatial correlation level                             %
%   configuration = Antenna configuration                                 %
%   fd            = Maximum Doppler shift in Hz                            %
%   fs            = Simulation sampling frequency in Hz                    %
%                                                                          %
% OUTPUT VALUE:                                                            %
%   H   =   complex channel taps                                          %
%                                                                          %
% Author: Marco Hernandez, v1.0                                           %
% *********************************************************************** %

function [H] =
ExtendedITU_MIMOChannel(model,correlation,configuration,fd,fs)

    switch model
        case 'pedestrian'
            %  vector of path delays in sec
            tau = [0 30 70 80 110 190 410]*1e-9;
            % vector of average path power gains in dB
            pdb = [0 -1 -2 -3 -8 -17.2 -20.8];
        case 'vehicular'
            tau=[0 30 150 310 370 710 1090 1730 2510]*1e-9;
            pdb=[0 -1.5 -1.4 -3.6 -0.6 -9.1 -7 -12 -16.9];
        case 'urban'
            tau=[0 50 120 200 230 500 1600 2300 5000]*1e-9;
```

```matlab
            pdb=[-1 -1 -1 0 0 0 -3 -5 -7];
        otherwise
            error('Model not implemented.');
    end

    switch correlation
        case 'low'
            a=0; b=0;
        case 'medium'
            a=0.3; b=0.9;
        case 'high'
            a=0.9; b=0.9;
        otherwise
            error('Spatial correlation profile not implemented.');
    end

    switch configuration
        case '1x2'
            Nt=1; Nr=2; Rt=1; Rr=[1 b;b 1];
        case '2x1'
            Nt=2; Nr=1; Rt=[1 a;a 1]; Rr=1;
        case '2x2'
            Nt=2; Nr=2; Rt=[1 a;a 1]; Rr=[1 b;b 1];
        case '4x2'
            Nt=4; Nr=2; Rt=[1 a^(1/9) a^(4/9) a;a^(1/9) 1 a^(1/9)
a^(4/9);a^(4/9) a^(1/9) 1 a^(1/9);a a^(4/9) a^(1/9) 1];
                        Rr=[1 b;b 1];
        case '2x4'
            Nt=2; Nr=4; Rt=[1 a;a 1];
                        Rr=[1 b^(1/9) b^(4/9) b;b^(1/9) 1 b^(1/9)
b^(4/9);b^(4/9) b^(1/9) 1 b^(1/9);b b^(4/9) b^(1/9) 1];
        case '4x4'
            Nt=4; Nr=4; Rt=[1 a^(1/9) a^(4/9) a;a^(1/9) 1 a^(1/9)
a^(4/9);a^(4/9) a^(1/9) 1 a^(1/9);a a^(4/9) a^(1/9) 1];
                        Rr=[1 b^(1/9) b^(4/9) b;b^(1/9) 1 b^(1/9)
b^(4/9);b^(4/9) b^(1/9) 1 b^(1/9);b b^(4/9) b^(1/9) 1];
        otherwise
            error('Antenna configuration not implemented.');
    end

    % Object MIMO multipath fading channel
    H=comm.MIMOChannel(...
        'SampleRate',                   fs,...
        'PathDelays',                   tau,...
        'AveragePathGains',             pdb,...
        'MaximumDopplerShift',          fd,...
        'DopplerSpectrum',              doppler.jakes,...
        'NumTransmitAntennas',          Nt,...
        'NumReceiveAntennas',           Nr,...
        'TransmitCorrelationMatrix',    Rt,...
        'ReceiveCorrelationMatrix',     Rr,...
        'FadingDistribution',           'Rayleigh',...
        'RandomStream',                 'mt19937ar with seed',...
        'Seed',                         99,...
        'NormalizePathGains',           true,...
        'NormalizeChannelOutputs',      true,...
        'PathGainsOutputPort',          false );

end
```

4.3.2 Large-Scale Fading

Large scale fading refers to the fading due to motion over large areas; this is referring to the distance between antenna positions on the body and external node (home, office, or hospital).

4.4 Path loss

Unlike traditional wireless communications, the path loss for body area network system (on body applications), is both distance and frequency dependent. The frequency dependence of bod tissues shall be considered. The path loss model in dB between the transmitting and the receiving antennas as a function of the distance d based on the Friis formula in free space is described by [[10] [11]]:

$$PL(d) = PL_0 + 10n \mathrm{Log}_{10} \left(\frac{d}{d_0} \right) \tag{1}$$

where PL_0 is the path loss at a reference distance d_0, and n is the path-loss exponent.

The path loss near the antenna depends on the separation between the antenna and the body due to antenna mismatch. This mismatch indicates that a body-aware antenna design could improve system performance.

4.5 Shadowing

Due to the variation in the environment surrounding of body or even movement of the body parts, path loss will be different from the mean value for a given distance as shown in equation (1). This phenomenon is called shadowing, and it reflects the path loss variation around the mean. The shadowing should be considered for stationary and non-stationary position of body.

When considering shadowing, the total path loss PL can be expressed by:

$$PL = PL(d) + S \tag{2}$$

where $PL(d)$ is expressed by the Equation (1) and S represents the shadowing component.

5. Models and Scenarios

5.1 Human BAN (HBAN) Channel models

5.2 In-body

5.2.1 Implant to implant CM1 (Scenario S1) for 402 – 405 MHz

Scenario S1 is a model for the communication between implanted device to implanted device. Channel model for this scenario, Implant to Implant CM1 (Scenario S1) for 402-405 MHz which is defined in IEEE 802.15.6 – 2012 Channel Model Document [1] can be applicable.

5.2.2 Implant to body surface CM2 (Scenario S2) for 402 – 405 MHz

Scenario S2 is a model for the communication between implanted device to on-body device. Channel model for this scenario, Implant to body surface CM2 (Scenario S2) for 402-405 MHz which is defined in IEEE 802.15.6 – 2012 Channel Model Document [1] can be applicable.

5.2.3 Implant (upper body) to Body Surface CM2.1 (Scenario S2.1) for 3.1 – 10.6 GHz

Scenario S2.1 is a model for the implant to the body surface on the UWB band for specific applications such as capsule endoscopy.
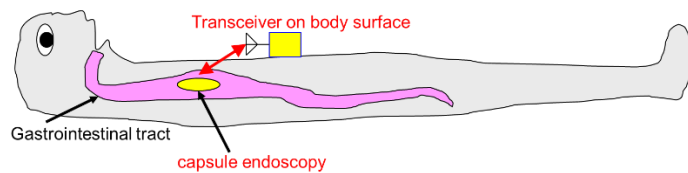
Figure 13 —Scenario 2.1: Capsule endoscopy application

OPTION 1

This model is developed based on doc.# P.802.15-22-0401-00-6a [12] . Path Loss (PL) versus distance *(d)* can be represented by:

$$PL(d) = PL(d_0) + 10n\, log_{10}\left(\frac{d}{d_0}\right) + S \qquad d \geq d_0$$

$$\text{……….. (3)}$$

Where, d_0 is a reference distance, n is the pathloss exponent, S is the random scatter around the regression line with Normal Distribution with standard deviation σ_s

Table 4 Implant to Body Surface CM2.1 for 3.1 – 10.6 GHz parameters (option 1)

Implant to body surface	$PL(d_0)$ (dB)	n	σ_s (dB)
	51.9	8.14	18.19

OPTION 2

This model is developed based on 15-23-0145-00-6a[14] . Path Loss (PL) versus distance *(d)* can be represented by:

$$PL_{\mathrm{dB}} = PL_{0,\mathrm{dB}} + 10n \log_{10}\left[\frac{d}{d_0}\right]$$

………. (4)

and the parameters that calculated by using FDTD simulations [14] . There are two configurations of direction of the implanted antennas with horizontal arrangement and vertical arrangement (Figure 14). Horizontal arrangement and vertical arrangement are shown in Table 5(a) and Table 5(b) respectively.

(a) Horizontal arrangement   (b) Vertical arrangement
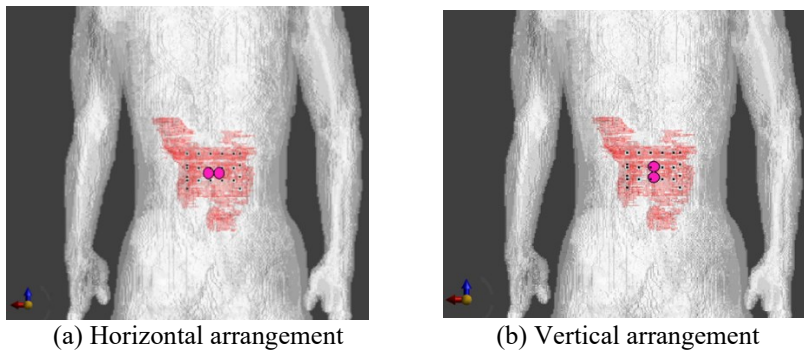
Figure 14—Implanted antenna arrangement

Table 5 Implant to Body Surface CM2.1 for 3.1 – 10.6 GHz parameters (option 2)

(a) Horizontal arrangement

$d_0 = 10$ mm

Transmitting components	PL_0 [dB]	n	σ_s [dB]
x	69.05	18.14	6.21
y	74.10	17.09	7.98
z	71.92	17.74	6.40

(b) Vertical arrangement

$d_0 = 10$ mm

Received components	PL_0 [dB]	n	σ_s [dB]
x	72.95	17.56	6.57
y	75.79	17.07	8.14
z	69.90	18.26	5.64

5.2.4 Implant (head) to Body Surface CM2.2 (Scenario S2.2) for 3.1 – 10.6 GHz

Scenario S2.2 is for the implant to the body surface on the UWB band for specific applications such as the brain-computer interface (BCI). Note the transmitting antenna is integrated into the transmitter device over a titanium plaque that replaces the skull bone. In that way, the embedded transmitter is under the skin to avoid infection.
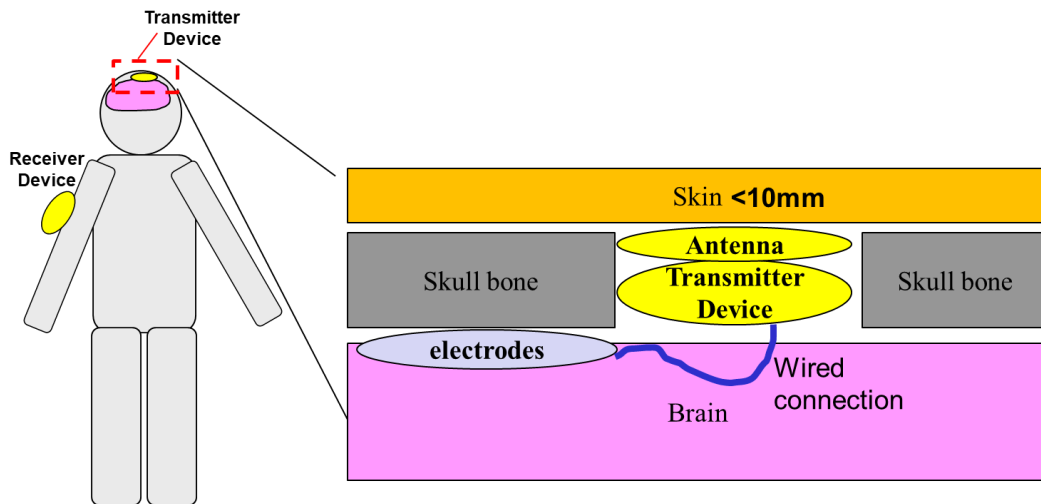
Figure 15 – Scenario S2.2 implant (head) to body surface

Path loss model is defined as

$$PL_{\text{dB}} = PL_{0,\text{dB}} + 10n \log_{10} \left[\frac{d}{d_0} \right] . \qquad (5)$$

Path loss parameters are simulated and provided in [13] ,[16] ,[17] ,[18] ,[19] ,[20] as follows.

OPTION 1: Pass loss model of Implanted device in head to head surface (Figure 16) channel has been derived as shown in Table 6.
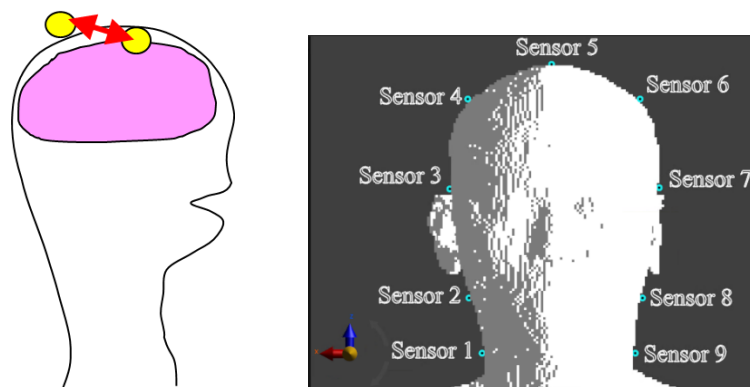
Figure 16 – Scenario S2.2 implant (head) to body surface (head)

Table 6 Implant (head) to Body Surface CM2.2 for 3.1 – 10.6 GHz parameters (head surface)

Transmission antenna direction	d_0 [m]	PL_0 (dB)	n	σ_s [dB]
X		24.84	3.84	9.52
Y	0.01	3.78	6.80	7.27
Z		3.73	6.03	9.31

OPTION 2: Path loss model parameters of Implanted device in head to body surface (head, arm, and chest) (Figure 17) channel has been derived as shown in Table 7.
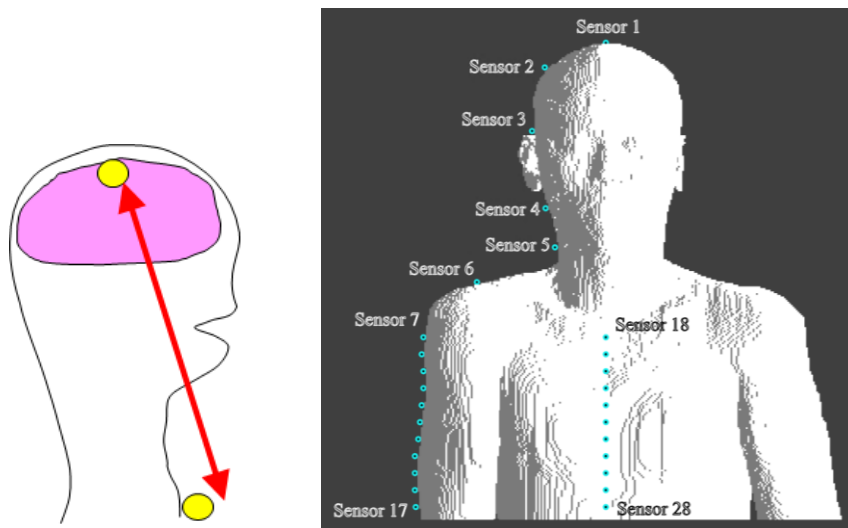
Figure 17 – Scenario S2.2 implant (head) to body surface (head, arm, and chest)

Table 7 Implant (head) to Body Surface CM2.2 for 3.1 – 10.6 GHz parameters (body surface)

Transmission antenna direction	d_0 [m]	PL_0 (dB)	n	σ_s [dB]
X		23.70	9.76	0.74
Y	0.1	102.62	0.57	8.15
Z		104.27	0.59	4.98

5.2.5 Implant (head) to External CM2.3 (Scenario S2.3) for 3.1 – 10.6 GHz

Scenario S2.3 is for the implant (head) to external on the UWB band for applications such as the brain-computer interface (BCI). Note the transmitting antenna is integrated into the transmitter device over a titanium plaque. In that way, the embedded transmitter is under the skin to avoid infection.
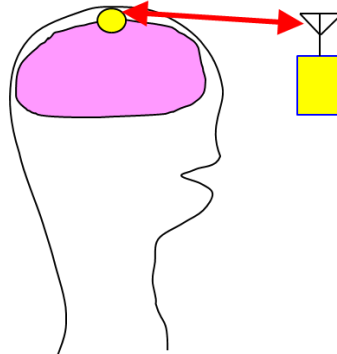
Figure 18 —Scenario 2.3: implant (head) to external.

This model is developed based on 15-23-0145-00-6a[14] . Path Loss (PL) versus distance *(d)* can be represented by:

$$PL_{\mathrm{dB}} = PL_{0,\mathrm{dB}} + 10n \log_{10}\left[\frac{d}{d_0}\right] \qquad \text{………. (6)}$$

and the parameters that calculated by using FDTD simulations [14] .

Table 8 Implant (head) to external CM2.3 for 3.1 – 10.6 GHz parameters

Transmission antenna direction	d_0 [m]	PL_0 (dB)	n	σ_s [dB]
X		48.47	1.45	9.63
Y	0.01	19.13	2.67	2.72
Z		22.81	2.69	7.55

5.2.6 Implant (upper body) to External CM3 (Scenario 3) for 402 – 405 MHz

Scenario S3 is a model for the communication between implanted device in upper body to external body device. Channel model for 402 – 405 MHz of this scenario, Implant to external CM2 (Scenario S3) for 402-405 MHz which is defined in IEEE 802.15.6 – 2012 Channel Model Document [1] can be applicable.

5.2.7 Implant (upper body) to External CM3.1 (Scenario 3.1) for 3.1 – 10.6 GHz

[TBD]

5.3 Body surface

5.3.1 Body surface to body surface (LOS) CM3 (Scenario S4 & S5) for 400 MHz

Scenario S4 is a model for the communication between a device on body surface to body surface device. Channel model for this scenario, Body surface to body surface CM3 (Scenario S4 & S5) for 400 MHz which is defined in IEEE 802.15.6 – 2012 Channel Model Document [1] can be applicable.

5.3.2 Body surface to body surface (LOS) CM3 (Scenario S4 & S5) for 600 MHz

Scenario S4 is a model for the communication between a device on body surface to body surface device. Channel model for this scenario, Body surface to body surface CM3 (Scenario S4 & S5) for 600 MHz which is defined in IEEE 802.15.6 – 2012 Channel Model Document [1] can be applicable.

5.3.3 Body surface to body surface (LOS) CM3 (Scenario S4 & S5) for 900 MHz

Scenario S4 is a model for the communication between a device on body surface to body surface device. Channel model for this scenario, Body surface to body surface CM3 (Scenario S4 & S5) for 900 MHz which is defined in IEEE 802.15.6 – 2012 Channel Model Document [1] can be applicable.

5.3.4 Body surface to body surface (LOS) CM3 (Scenario S4 & S5) for 2.4 GHz

Scenario S4 is a model for the communication between a device on body surface to body surface device. Channel model for this scenario, Body surface to body surface CM3 (Scenario S4 & S5) for 2.4 GHz which is defined in IEEE 802.15.6 – 2012 Channel Model Document [1] can be applicable.

5.3.5 Body surface to body surface CM3 (Scenario S4 & S5) for 3.1-10.6 GHz

The following path loss model is based on measurements that cover frequencies of 3.1-10.6 GHz. Measurement set up, derivation and data analysis can be found in [21] . The table below summarizes the model and corresponding parameters.

Table 9 Parameters of the path loss model for CM3 (S4 & S5) for 3.1 – 10.6 GHz

	Hospital Room	Anechoic Chamber
Path loss model	$PL(d)$[dB] $= a \cdot \log_{10}(d) + b + N$	
a	19.2	34.1
b	3.38	-31.4
σ_N	4.40	4.85

-
- a and b : Coefficients of linear fitting
- d : Tx -Rx distance in mm
- N : Normally distributed variable with zero mean and standard deviation σ_N

A power delay profile (PDP) model for 3.1 – 10.6 GHz is also given in [21] . The table below summarizes this model and corresponding parameter.

Table 10 Parameters of the PDP model for CM3 (S4 & S5) for 3.1 – 10.6 GHz

PDP Model	$$h(t) = \sum_{l=0}^{L-1} a_l \exp(j\phi_l)\delta(t - t_l)$$ $$10\log_{10}|a_l|^2 = \begin{cases} 0 & l = 0 \\ \gamma_0 + 10\log_{10}\left(\exp\left(-\frac{t_l}{\Gamma}\right)\right) + S & l \neq 0 \end{cases}$$ $$p(t_l \mid t_{l-1}) = \lambda \exp(-\lambda(t_l - t_{l-1}))$$ $$p(L) = \frac{\overline{L}^L \exp(\overline{L})}{L!}$$ ϕ_l is modeld by a uniform distribution over $[0, 2\pi)$	
a_l	γ_0	-4.60 dB
	Γ	59.7
	σ_s	5.02 dB
t_l	$1/\lambda$	1.85 ns
L	$\overline{L}$	38.1

- a_l : path amplitude for the l-th path
- t_l : path arrival time for the l-th path
- ϕ_l : phase for the l-th path
- L : the number of the arrival paths
- $\delta(t)$: the Dirac function
- Γ : an exponential decay with a Rician factor γ_0
- S : a normal distribution with zero-mean and standard deviation of σ_S.
- λ: path arrival rate
- $\overline{L}$: the average number of the L

Option 2

The following path loss model is based on measurements that cover frequencies of 3.1-10.6 GHz. Measurement set up, derivation and data analysis can be found in [22] . The table below summarizes the corresponding parameters.

Table 11 Parameters of the path loss model for CM3 (S4 & S5) (Option 2)

Path loss model	$PL(d)[\text{dB}] = P_0[\text{dB}] + 10n\log_{10}\left(d/d_0\right)$	
Around torso		
Antenna separation from body surface	0 mm	5 mm
P_0 [dB]	56.1	48.4
d_0 [m]	0.1	0.1
n	5.8	5.9
Along torso		
Antenna separation from body surface	0 mm	5 mm
P_0 [dB]	56.5	44.6
d_0 [m]	0.1	0.1

- n: Path loss exponent
- P_0: Path loss at the reference distance
- d_0: Reference distance

5.3.6 Dynamic channel model for body surface to body surface CM3 (Scenario S4 & S5) at 4.5 GHz

A real time channel measurements by use of channel sounder has been performed in [24] . The channel measurements were carried out in an anechoic chamber with center frequency of 4.5 GHz and bandwidth of 120 MHz. Details of the measurement set up, derivation and data analysis can be found in [24] . The measurements are focusing on the fading effect due to movements of the human body, therefore conducted in an anechoic chamber, where the multipath from the surrounding objects are negligible.
The transmitter antenna was fixed on around navel. The table below summarize the position of receiving antennas and distance between Tx and Rx antennas.

Table 12 Summary of the position of receiving antennas and distance between Tx and Rx antennas for CM3 (Scenario S4 & S5) at 4.5 GHz

Position	Distance [mm]
A Right wrist	440 ~ 525
B Right upper arm	360
C Left ear	650
D Head	710
E Shoulder	310
F Chest	230
G Right rib	183
H Left waist	140
I Thigh	340
J Ankle	815 ~ 940

To provide a statistical model, the probability distribution functions such as normal, log-normal and Weibull distributions have been tried to fill the measurement results. The tables below summarize the normal, log-normal and Weibull distributions.

Table 13 Summary of the normal distributions parameter for CM3 (Scenario S4 & S5) at 4.5 GHz

Position	(Normal) $f(x\mid\mu,\sigma)=\dfrac{1}{\sigma\sqrt{2\pi}}e^{-\frac{(x-\mu)^2}{2\sigma^2}}$		
	Still μ/σ [dB] (-log L)	Walking μ/σ [dB] (-log L)	Up-down μ/σ [dB] (-log L)
Right wrist	1.0000 / 0.1279 (-6303)	3.0875 / 4.4063 (28972)	8.9461 / 5.1576 (30220)
Right upper arm	1.0000 / 0.1500 (-4735)	3.2443 / 1.7012 (19332)	7.3374 / 6.1415 (32061)
Right ear	1.0000 / 0.1349 (-5826)	0.7303 /.3014 (101)	0.8205 / 0.8232 (12148)
Head	1.0000 / 0.0666 (-12706)	0.4477 / 0.24450 (4535)	0.9058 / 1.2629 (16207)
Shoulder	1.0000 / 0.0335 (-19782)	2.6849 / 1.4627 (2153)	1.5437 / 1.3265 (17519)
Chest	1.0000 / 0.3481 (3600)	3.6360 / 2.4688 (17849)	7.1082 / 9.2594 (36609)
Right rib	1.0000 / 0.1654 (-3812)	0.8572 / 0.2744 (1256)	4.7043 / 2.9597 (24784)
Left waist	1.0000 / 0.0399 (-17978)	0.7264 / 0.1609 (-4018)	0.5458 / 0.3105 (2456)
Right thigh	1.0000 / 0.0964 (-9254)	0.6500 / 0.4388 (5831)	1.1357 / 0.6949 (10272)
Right ankle	1.0000 / 0.0787 (-11025)	1.6070 / 0.9968 (14248)	1.2489 / 1.2220 (16122)

- μ : Mean value
- σ : Variable value
- *-LogL*: Negative log liklihood

**Table 14 Summary of the log-normal distribution parameters for CM3
(Scenario S4 & S5) at 4.5 GHz**

Position	(Log-Normal) $f(x\mid\mu,\sigma)=\dfrac{10/\ln 10}{x\sigma\sqrt{2\pi}}e^{-\frac{(10\log_{10}x-\mu)^2}{2\sigma^2}}$		
	Still μ/σ [dB] (-log L)	Walking μ/σ [dB] (-log L)	Up-down μ/σ [dB] (-log L)
Right wrist	-0.0652 / 0.9531 (-1114.1)	-3.9908 / 11.7727 (14948)	8.4456 / 3.4836 (31047)
Right upper arm	-0.0491 / 0.6544 (-4800.9)	4.4491 / 2.4972 (18735)	6.3009 / 5.6692 (31093)
Right ear	-0.0400 / 0.5921 (-5815.6)	-4.1465 / 2.4654 (-1005.5)	-3.3130 / 5.2458 (8384.1)
Head	-0.0100 / 0.2969 (-12476)	-3.5392 / 3.2052 (2983.8)	-3.6932 / 5.3385 (7601)
Shoulder	-0.0024 / 0.1453 (-19806)	-1.8011 / 2.0549 (2507.8)	0.2165 / 4.1023 (14537)
Chest	-0.2359 / 1.4019 (2315)	3.6360 / 2.4688 (16779)	5.0788 / 6.4077 (29907)
Right rib	-0.0470 / 0.6043 (-5657.3)	-0.8883 / 1.3994 (818.79)	5.7033 / 3.2101 (24051)
Left waist	-0.0035 / 0.1742 (-17942)	-1.4965 / 0.9770 (-4109.5)	-3.5588 / 3.3466 (3337.8)
Right thigh	-0.0191/ 0.4017 (-9717)	-2.6924 / 2.6453 (2970.7)	-0.5995 / 3.7712 (11098)
Right ankle	-0.0131/ 0.3352 (-11251)	0.9424 / 3.7030 (14861)	-1.1303 / 4.7167 (12357)

- μ : Mean value
- σ : Variable value
- *-LogL*: Negative log loklihood

**Table 15 Summary of the Weibull distribution parameters for CM3
(Scenario S4 & S5) at 4.5 GHz**

Position	(Weibull) $f(x\mid a,b) = \dfrac{b}{a}\left(\dfrac{x}{a}\right)^{b-1} e^{-\left(\frac{x}{a}\right)^{b}}$		
	Still a/b (_-LogL)	Walking a/b (_-LogL)	Updn a/b (_-LogL)
Right wrist	1.0478 / 7.7411 (-5770.3)	1.4690 / 0.4510 (14629)	10.0006 / 1.7319 (29786)
Right upper arm	1.0655 / 7.3618 (-4529.3)	3.6759 / 2.0325 (18625)	7.5368 / 1.0787 (29631)
Right ear	1.0594 / 7.7870 (-5394.6)	0.5071 / 1.9516 (-840.12)	0.8224 / 1.0055 (7959.7)
Head	1.0297 / 17.9473 (-13029)	0.6360 / 1.5519 (3061.7)	0.8015 / 0.8228 (8458.3)
Shoulder	1.0165 / 29.8877 (-18940)	0.8237 / 2.6536 (1913.3)	1.6534 / 1.2203 (14452)
Chest	1.1203 / 2.9913 (3487.2)	3.0419 / 1.9575 (16909)	6.3808 / 0.8247 (29393)
Right rib	1.0742 / 4.9107 (-1979.2)	0.9539 / 3.2009 (1357.2)	5.2709 / 1.6440 (23654)
Left waist	1.0190 / 28.9055 (-17946)	0.7915 / 4.9572 (-3903.2)	0.6117 / 1.7907 (1823.7)
Right thigh	1.0464 / 9.6410 (-7951.2)	0.7325 / 1.6297 (3867.9)	1.2626 / 1.6251 (9539.3)
Right ankle	1.0379 / 11.8572 (-9813.1)	1.7880 / 1.6227 (13323)	1.2867 / 1.0759 (12124)

- *a* : Scale factor
- *b*: Shape factor
- *-LogL*: Negative log loklihood

The below table summarize the best fitting distributions for dynamic channel at 4.5 GHz.

**Table 16 Summary of the best fitting distributions for dynamic channel parameters for CM3
(Scenario S4 & S5) at 4.5 GHz**

Position	Still	Walking	Up-down
Right wrist	Normal	Weibull	Weibull
Right upper arm	Log-normal	Weibull	Weibull
Right ear	Normal	Log-normal	Weibull
Head	weibull	Log-normal	Log-normal
Shoulder	Log-normal	Weibull	Weibull
Chest	Log-normal	Log-normal	Weibull
Right rib	Log-normal	Log-normal	Weibull
Left waist	Normal	Log-normal	Weibull
Right thigh	Log-normal	Log-normal	Weibull
Right ankle	Log-normal	Weibull	Weibull

- Normal distribution seems to fit the still posture best, but it can be seen that fittings with any distributions have large from the PDFs.

- Log-normal distribution shows good match in cases of still postures and small movements such as walking posture in case of head , right ear, chest, right rib, left waist, right thigh, and stand up/down posture in case of head.
- Weibull distribution can represent much better large movement behaviors such as walking posture in case of right wrist, right upper arm, shoulder, right ankle, and all stand up/down postures except for head.

Based on statistical analysis of fading duration a 5-state Fritchman model for dynamic on-body channels, as shown in the below Figure, has been used to classifies channel states according to the dwelling time (Θ) in different channel qualities to quantitatively describe the time-varying property of on-body channels.
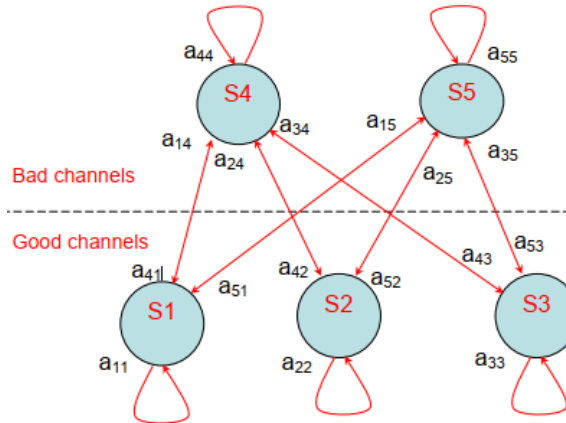

Figure 19 Five-state Fritchman model for dynamic on-body channels

The five-state Fritchman model for describing the burst behaviors of on-body channels are:

- S1: unstable error-free state, good channels which last less than 20 ms;
- S2: semi-constant error-free state, good channel which are over 20 ms and less than 400
- ms;
- S3: constant error-free state, good channel which are over 400 ms;
- S4: unstable error state, bad channel which last less than 20 ms, and
- S5: semi-constant error state, bad channel which are less than 400 ms.

The statistical parameters for the on-body dynamic channel at Θ=-10dB for different locations are;

Table 17 statistical parameters for the on-body dynamic channel

Positions	Distance (mm)	Walking			up/down		
		Relative path gain (dB)	Duration (ms)		Relative path gain (dB)	Duration (ms)	
			Good channels	Bad channels		Good channels	Bad channels
A	440~525	-3.99±11.8	204.23±252.37	99.03±155.97	8.45±3.48	10,000	0
B	360	4.45±2.50	10,000	0	6.30±5.67	423.33±851.80	12.6±21.98
C	710	-4.15±2.47	159.03±509.90	1.65±2.68	-3.13±5.25	37.56±171.64	7.15±25.70
D	650	-3.54±3.21	67.64±251.40	2.83±3.75	-3.69±3.21	62.06±265.87	10.23±43.57
E	310	-1.80±2.05	10,000	0	0.22±4.10	1666.7±250.6	2.1±5.8
F	230	3.64±2.47	10,000	0	5.08±6.41	572.56±779.69	14.12±18.13
G	183	-0.89±1.40	10,000	0	5.70±3.21	10,000	0
H	140	-1.50±0.98	10,000	0	-3.56±3.35	862.71±1174.41	51.88±71.44
I	340	-2.69±2.65	750.35±1403.24	2.00±2.45	-0.60±3.77	710.08±840.69	25.15±26.94
J	815~940	0.94±3.70	574.12±459.97	10.82±4.75	-1.13±4.72	262.48±538.37	20.17±47.19

The statistical parameters for the on-body dynamic channel at Θ=-10dB in different states are

Table 18 — statistical parameters for the on-body dynamic channel

Parameters		State samples		State durations (ms)	
		Walking	up/down	Walking	up/down
Good channels	S1	93	337	3.31±3.66	2.38±3.05
	S2	29	34	99.41±65.15	161.1±132.3
	S3	32	38	2515.5±3342.9	1771.7±2082.4
Bad channels	S4	145	371	3.45±4.38	3.48±4.20
	S5	10	38	287.1±139.3	80.68±87.26

Parameters of the 5-state Fritchman model for different action scenarios are

Table 19 Parameters of the five-state Fritchman model for different action scenarios

Threshold $\theta=-10$ (dB)	A and Π_0
Walking	$A = \begin{bmatrix} 0.772 & 0 & 0 & 0.223 & 0.005 \\ 0 & 0.991 & 0 & 0.0082 & 0.00087 \\ 0 & 0 & 0.9997 & 0.00025 & 0.00007 \\ 0.137 & 0.047 & 0.043 & 0.773 & 0 \\ 0.00189 & 0.0009 & 0.00063 & 0 & 0.996 \end{bmatrix}$ $\Pi_0 = \begin{bmatrix} 0.00415 & 0.0346 & 0.923 & 0.0064 & 0.032 \end{bmatrix}$
Standing up/sitting down	$A = \begin{bmatrix} 0.69 & 0 & 0 & 0.29 & 0.019 \\ 0 & 0.993 & 0 & 0.005 & 0.0017 \\ 0 & 0 & 0.9995 & 0.00032 & 0.00016 \\ 0.195 & 0.0209 & 0.0183 & 0.766 & 0 \\ 0.0076 & 0.00147 & 0.0044 & 0 & 0.986 \end{bmatrix}$ $\Pi_0 = \begin{bmatrix} 0.011 & 0.058 & 0.882 & 0.0158 & 0.034 \end{bmatrix}$
Actions combined	$A = \begin{bmatrix} 0.7131 & 0 & 0 & 0.2715 & 0.01542 \\ 0 & 0.9924 & 0 & 0.006169 & 0.00140 \\ 0 & 0 & 0.9996 & 0.000283 & 0.00012 \\ 0.1783 & 0.0284 & 0.0252 & 0.768 & 0 \\ 0.00486 & 0.00122 & 0.00258 & 0 & 0.9913 \end{bmatrix}$ $\Pi_0 = \begin{bmatrix} 0.0075 & 0.046 & 0.902 & 0.011 & 0.033 \end{bmatrix}$

5.3.7 Body surface to external CM4 (Scenario S6 & S7) for 900MHz

Channel model for Body surface to external CM4 (Scenario S6 & S7) for 900 MHz which is defined in IEEE 802.15.6 – 2012 Channel Model Document [1] .

5.3.8 Body surface to external CM4 (Scenario S6 & S7) for 2.4 GHz

Channel model for Body surface to external CM4 (Scenario S6 & S7) for 2.4 GHz which is defined in IEEE 802.15.6 – 2012 Channel Model Document [1] .

5.3.9 Body surface to external CM4 (Scenario S6 & S7) for 3.1-10.6 GHz

Measurement for the UWB frequency band of 3.1-10.6 GHz has been performed in [23] . On body antenna characteristics were measured in anechoic chamber, while channel measurements were done in office environment. For this measurement, the Tx antenna is fixed near to wall, while the Rx antenna (placed on body) positions were changed in office area. The effect of ground is considered in measurements. All data were averaged for statically analysis; therefore, the detail data of each measurement is not provided. The Further detail on set-up, derivation and data analysis can be found in [23] . The following figures summarize the delay profile of front, side, and backside of body while the Tx antenna is in the front of body.

Table 20 Parameters of the path loss model for CM4 (Scenario S6 & S7) for 3.1-10.6 GHz

Model	$$h(t) = \sum_{m=0}^{L-1} \alpha_m \delta(t - \tau_m)$$ $$\|\alpha_m\|^2 = \Omega_0 e^{-\frac{\tau_m}{\Gamma} - k[1-\delta(m)]} \beta$$ $$k = \Delta k (\ln 10 / 10)$$ $$\tau_0 = d/c$$ $$\beta \sim \log normal(0, \sigma)$$		
Direction of Body	**Γ[ns]**	**k (Δk[dB])**	**σ[dB]**
0	44.6346	5.111 (22.2)	7.30
90	54.2868	4.348 (18.8)	7.08
180	53.4186	3.638 (15.8)	7.03
270	83.9635	3.983 (17.3)	7.19

- $h(t)$: Complex impulse response
- L : Number of arrival paths, modelled as Poisson random variable with mean $\bar{L} = 400$
- a_m : Amplitude of each path
  τ_m,m $=1....L-1$: Timing of path arrivals, modelled as Poisson random process with arrival rate $\lambda = 1/(0.50125$ ns)
- k: Effect of K-factor (NLOS)
- Ω_0: Path loss (can be assumed as free space. In CM4 Matlab code Ω_0 is normalized to unity; Ω_0=1)
- d: Tx-Rx distance
- c: Velocity of light

5.3.10 Body surface (head) to external CM6.1 (Scenario S6.1) for 3.1 – 10.6 GHz

Scenario S6.1 is for the implant to the body surface on the UWB band for specific applications of brain-computer interface (BCI). Note the transmitting antenna is integrated into the transmitter device over a titanium plaque that replaces the skull bone. In that way, the embedded transmitter is under the skin to avoid infection.
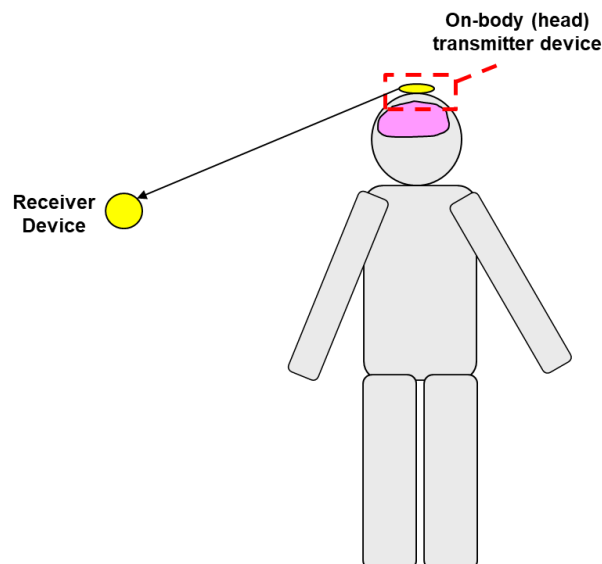

Figure 20 – Scenario 6.1: body surface (head) to external.

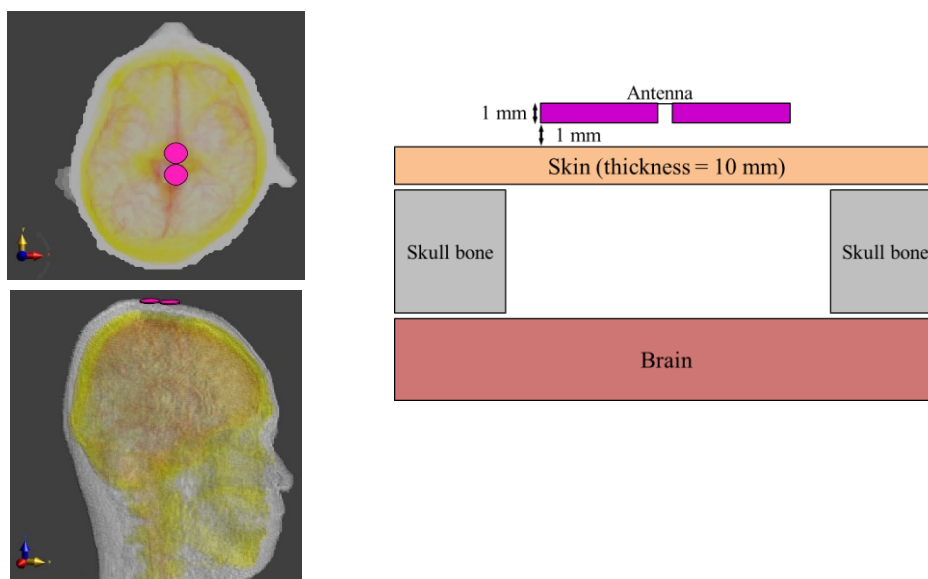**Figure 21 – Scenario 6.1: On-body(head) antenna location.**

Path loss model is defined as

$$PL_{\mathrm{dB}} = PL_{0,\mathrm{dB}} + 10n \log_{10}\left[\frac{d}{d_0}\right] . \qquad (7)$$

Path loss parameters are simulated and provided in,[16] as follows.

Pass loss model parameters of on body (head) device to external (Figure 22) channel has been derived as shown in Table 21.
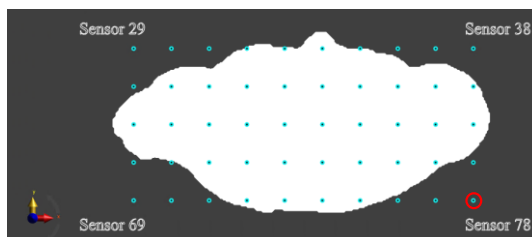
Figure 22 – External device locations (Top view of human body)

Table 21 5.3.10 Body surface (head) to external CM6.1 (Scenario S6.1) for 3.1 – 10.6 GHz parameters

Transmission antenna direction	d_0 [m]	PL_0 (dB)	n	σ_s [dB]
X		40.26	1.65	6.19
Y	0.01	23.09	2.25	3.85
Z		22.12	2.66	5.81

5.3.11 BAN coordinator to BAN coordinator CM6.2 (Scenario S6.2) for 3.1 – 10.6 GHz

Figure 23 — HBAN coordinator to HBAN coordinator (Scenario 6.2, CM 6.2)

Channel models for HBAN coordinator to HBAN coordinator was provided by simulation result [14] . Simulation was performed by using simplified car 3D model and pathloss characteristics has been calculated as follows.

Table 22 BAN coordinator to BAN coordinator model parameters in S6.2

	Path loss (dB)	Distance (mm)
S6.2	-61.69	4000

5.3.12 Body surface to external CM4 (Scenario S7) for 900 MHz

Scenario S7 is a model for the communication between a device on body surface to external device. Channel model for this scenario, Body surface to external CM4 (Scenario S6 & S7) for 900 MHz which is defined in IEEE 802.15.6 – 2012 Channel Model Document [1]   can be applicable.
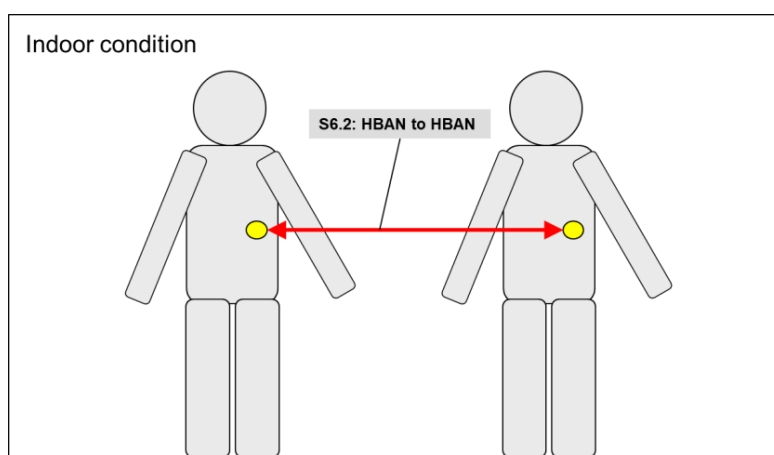
5.3.13 Body surface to external CM4 (Scenario S7) for 2.4 GHz

Scenario S7 is a model for the communication between a device on body surface to external device. Channel model for this scenario, Body surface to external CM4 (Scenario S6 & S7) for 2.4 GHz which is defined in IEEE 802.15.6 – 2012 Channel Model Document [1]   can be applicable.

5.3.14 Body surface to external CM4 (Scenario S7) for 3.1 – 10.6 GHz

Scenario S7 is a model for the communication between a device on body surface to external device. Channel model for this scenario, Body surface to external CM4 (Scenario S6 & S7) for 3.1 – 10.6 GHz which is defined in IEEE 802.15.6 – 2012 Channel Model Document [1]   can be applicable.

5.4 Vehicle BAN (VBAN) channel models

VBAN channel models are defined as four cases; in-vehicle to in-vehicle, on-vehicle to on-vehicle and on-vehicle to surrounding vehicle cases.

The considered channel models are based on the channel model parameters contribution for VBAN [15] .

Scenarios 8 to 14 channel model parameters was calculated based on ray-tracing method simulations using simplified 3D model as shown in Figure 24.
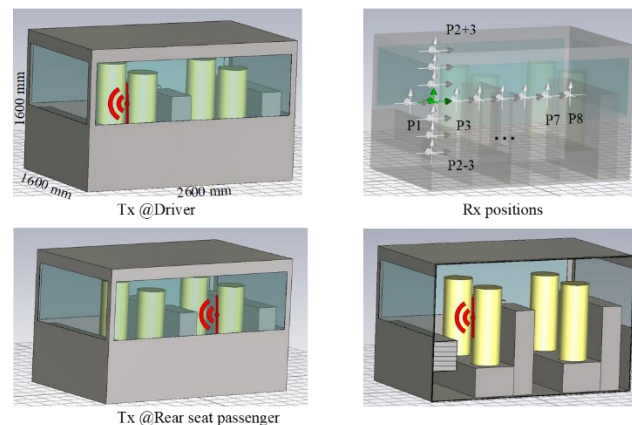
Figure 24 — Simplified car 3D model used for simulation of Scenario 8 to 14

5.5 In Vehicle models

5.5.1 In-vehicle to In-vehicle (Scenario 8, CM 8) 3.1 - 10.6GHz
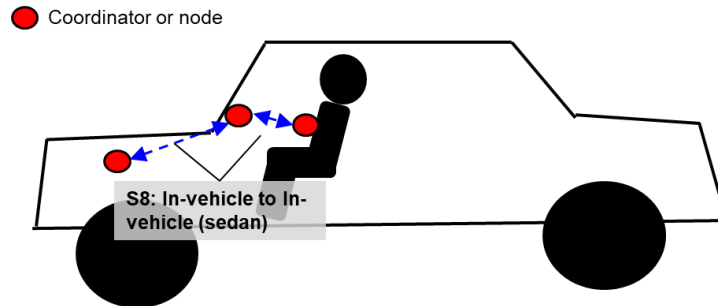
Figure 25 —In-vehicle to In-vehicle (Scenario 8, CM 8)

Channel models for In-vehicle to In-vehicle was provided by the ray-tracing simulation result[15] . Simulation was performed by using simplified car 3D model and pathloss characteristics has been calculated as follows.

Table 23 In-vehicle to In-vehicle pathloss model parameters in S8

d_0=1000 (mm)

Tx	Driver	Rear seat
n	1.28	1.73
PL_0 [dB]	65.35	60.93

$$PL = PL_0 + 10n \log_{10}\left(\frac{d}{d_0}\right) \text{ (dB)}$$

5.5.2 Cabin room to cabin room in omnibus (Scenario 8.1) 3.1 - 10.6GHz

In large vehicle like omnibus type of vehicle, multiple HBAN is in the same cabin room. In such situation, channel model CM8.1 can be applied to analyze communication between VBAN and multiple HBANs.

Nodes and coordinator, VBAN and HBAN are in bus cabin room
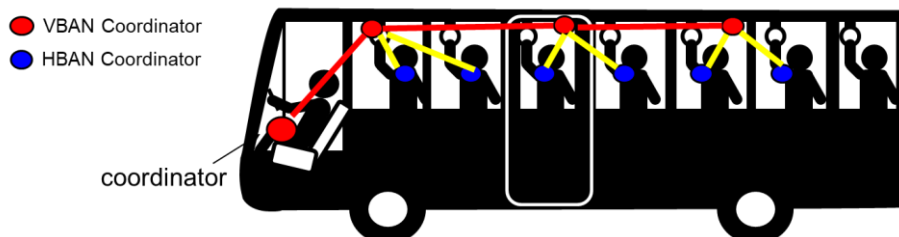
Figure 26 —Scenario8.1 for Omnibus HBANs to VBAN connection

In multiple passengers on a vehicle such as omnibus, VBAN coordinator is connected with multiple VBAN nodes and also connected with HBAN coordinator as well as HBAN nodes that is used passengers personally. In this situation, channel models are defined based on simulation results [13] and [15] .

OPTION 1

This channel model can be used to estimate pathloss in situation that transmitter (or node) is on passenger's body surface and receiver (or coordinator) is on the other passenger's body surface. Also, this channel model can be applicable to use to estimate pathloss between transmitter (or node) to receiver (or coordinator) on inner surface of bus body. This channel model has been derived based on a ray-tracing method simulation with simplified bus and human body tissue models[13] [15] . This model provides simulated path-loss in omnibus environment.

✓ Frequency band: 3.1 – 10.6 GHz
✓ Ray-tracing method
✓ Cylinder model employed as a human model for simplification
✓ HBAN node on the chest of the human model
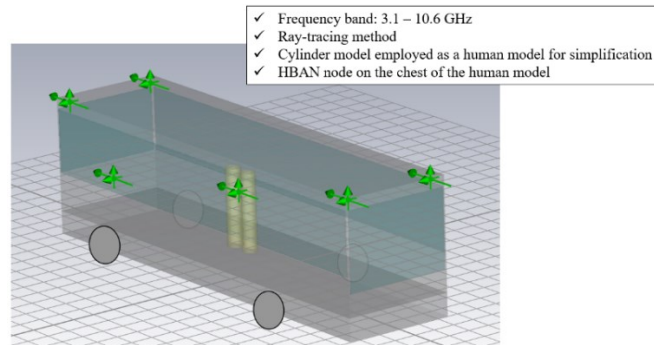
Figure 27 —simplified omnibus 3D models for ray-tracing simulation
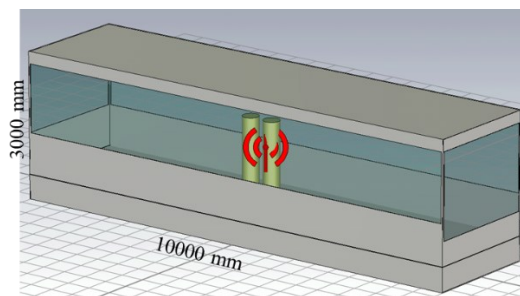
Figure 28 —transmitter location for 2 persons model
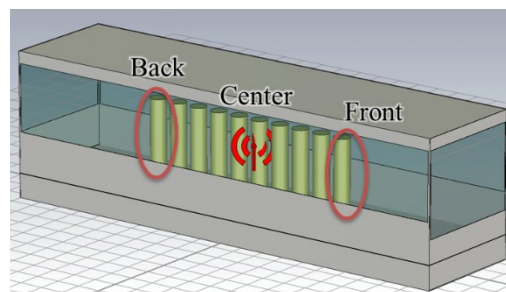
Figure 29 —transmitter location for 10 persons model
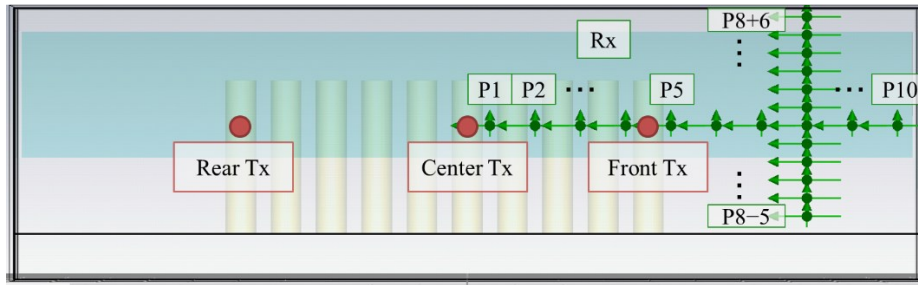
Figure 30 —transmitter and receiver positions

Table 24 On-vehicle to on-vehicle pathloss model parameters in S8.1 (Option 1)

Channel parameter (d_0=1000 (mm))

Tx	Center				Front		Rear	
direction	Forward		Sideways		Forward	Sideways	Forward	Sideways
persons	10	2	10	2	10	10	10	10
n	1.31	1.96	0.33	1.48	1.01	0.70	1.22	2.07
PL_0[dB]	60.98	63.54	65.03	73.49	58.31	63.18	47.21	52.65

$$PL = PL_0 + 10n \log_{10}\left(\frac{d}{d_0}\right) \text{ (dB)}$$

OPTION 2

Additionally, following channel models can be used for the use case that node or coordinator supposed to be located at the wall side of bus, or driver's sheet as shown in Figure 31.

Tx position at a person boarding on the bus

Tx position at a driver
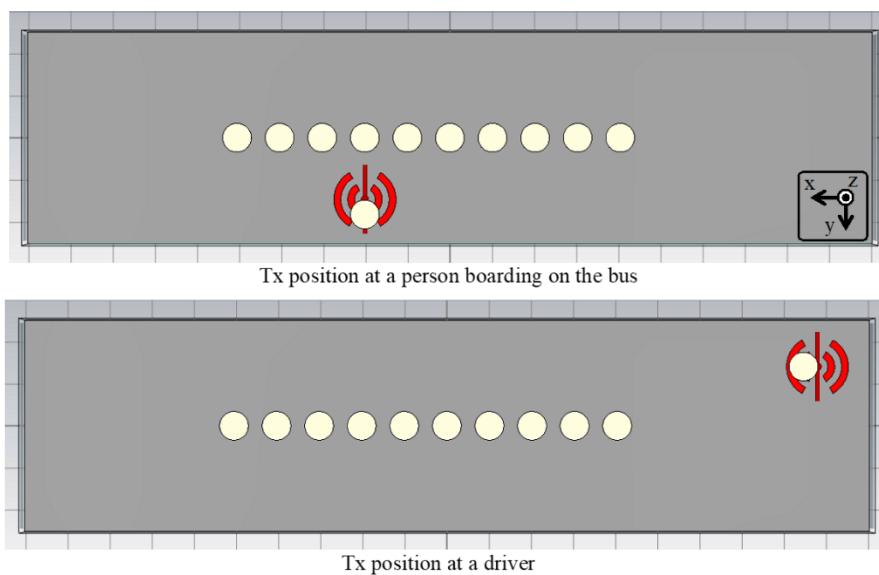
Figure 31 —transmitter positions

This channel model can be represented as follows.

Table 25 On-vehicle to on-vehicle pathloss model parameters in S8.1 (Option 2)

		$d_0 = 1000$ (mm)
Tx	Driver	Getting on
n	1.01	1.19
PL_0 [dB]	70.29	71.82

$$PL = PL_0 + 10n \log_{10}\left(\frac{d}{d_0}\right) \text{ (dB)}$$

5.5.3 In-vehicle to In-vehicle (functional compartment) (Scenario 8.2, CM 2)
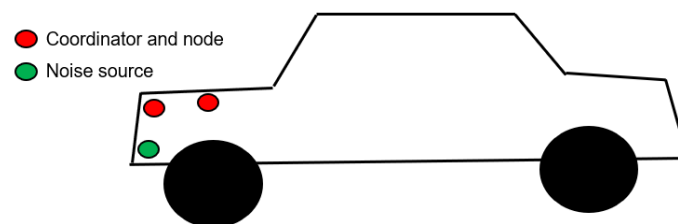
Figure 32 — 6.5.1.1 Engine compartment to engine compartment (Scenario 8.2, CM 8.2)

Channel models for in functional compartment such as engine compartment of the vehicle is defined separately. The model was provided by the ray-tracing simulation result[15] . Simulation was performed by using simplified car 3D model and pathloss characteristics has been calculated.
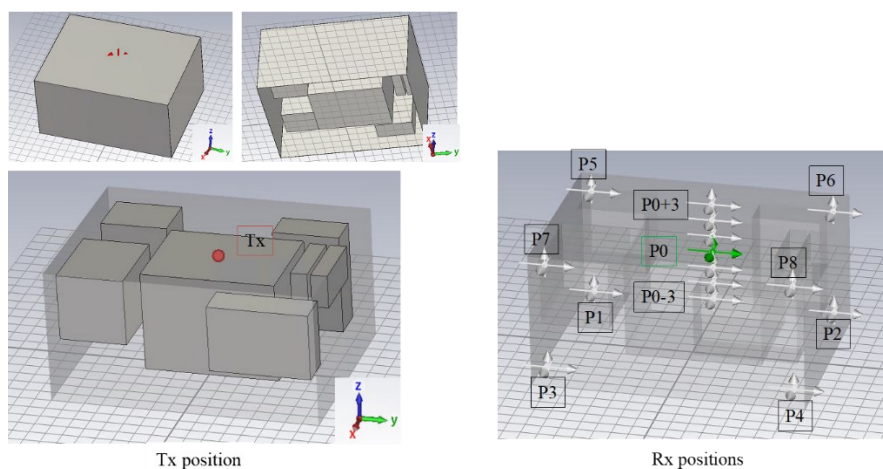
Figure 33 — 6.5.1.1 Scenario 8.2 simulation model

From the results of simulations, pathloss model is defined as follows.

Table 26 In-vehicle to In-vehicle (functional compartment) (Scenario 8.2, CM 8.2) pathloss parameters

$d_0 = 250$ (mm)

Tx	Engine
n	8.0×10^{-5}
PL_0 [dB]	54.18

$$PL = PL_0 + 10n \log_{10}\left(\frac{d}{d_0}\right) \text{ (dB)}$$

5.6 On vehicle

5.6.1 On-vehicle to on-vehicle (Scenario 11, 12, CM11, 12)
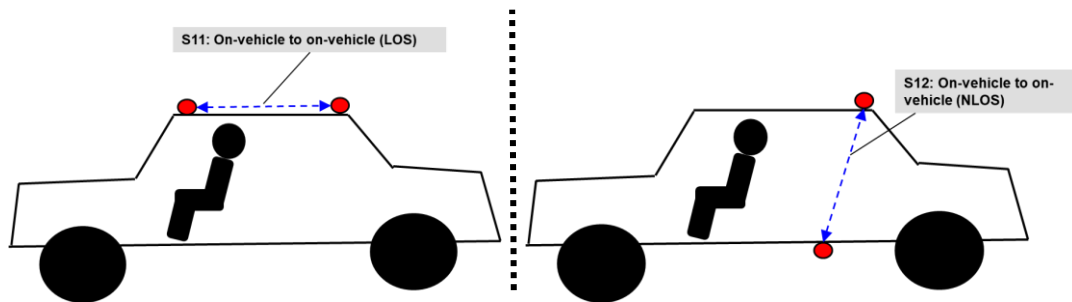
Figure 34 —On-vehicle to on-vehicle (Scenario 11, 12, CM 11, 12)

Channel models for on-vehicle to on-vehicle was provided by the ray-tracing simulation result[15]. Simulation was performed by using simplified car 3D model and pathloss characteristics has been calculated as follows.

Table 27 On-vehicle to on-vehicle pathloss model parameters in S11 & S12

	Path loss (dB)	Distance (mm)
S11	-69.04	1500
S12	-102.08	1800

5.6.2 In-vehicle to on vehicle (Scenario 9, CM9) 3.1 – 10.6 GHz
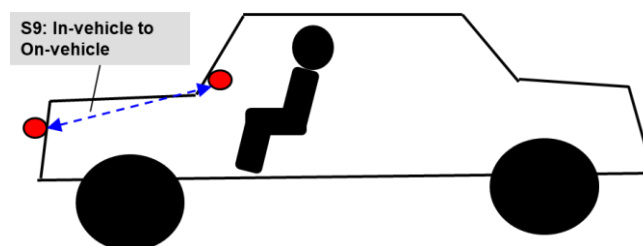
Figure 35 ―In-vehicle to on vehicle (Scenario 9, CM9) 3.1 – 10.6 GHz

Channel models for In-vehicle to on-vehicle was provided by the ray-tracing simulation result [15]. Simulation was performed by using simplified car 3D model and pathloss characteristics has been calculated as follows.

Table 28 In-vehicle to On-vehicle pathloss model parameters in S9

Scenario	Path loss (dB)	Distance (mm)
S09	-74.75	2077

5.7 External

5.7.1 In-vehicle to External (Scenario 10, CM 10) 3.1 – 10.6GHz

Engine compartment / body surface (outside)
- 🔴 Coordinator and node
- 🟢 Noise source
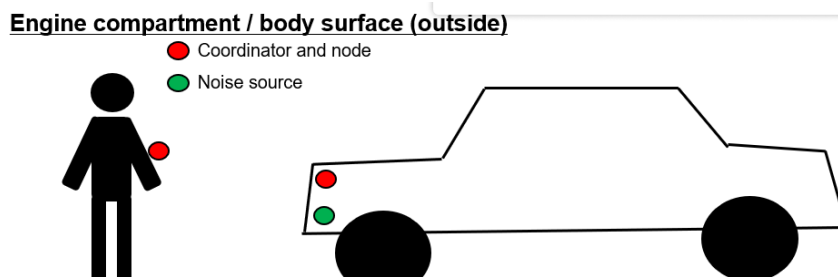
Figure 36 —In-vehicle to external (Scenario 10, CM10) 3.1 – 10.6 GHz

Channel models for In-vehicle to external was provided by the ray-tracing simulation result [15] . Simulation was performed by using simplified car 3D model and pathloss characteristics has been calculated as follows.

Table 29 In-vehicle to external pathloss model parameters in S10

Scenario	Path loss (dB)	Distance (mm)
S10	-86.45	3536

5.7.2 On-vehicle to External (LOS) (Scenario 13, CM 13) 3.1 – 10.6GHz

S13: On-vehicle to external (LOS)
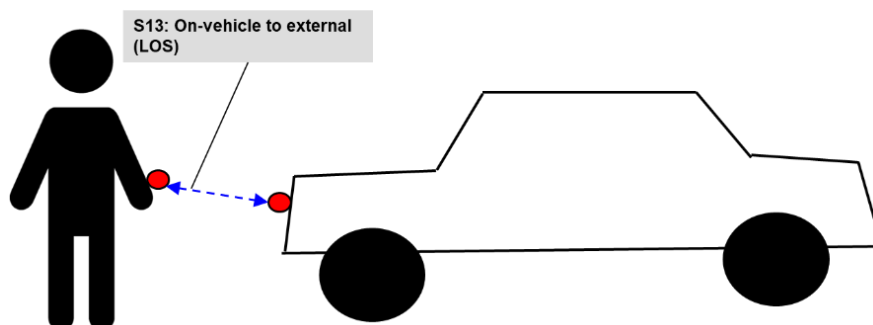
Figure 37 — On-vehicle to External (LOS) (Scenario 13, CM 13) 3.1 – 10.6GHz

Channel models for on-vehicle to external (LOS) was provided by the ray-tracing simulation result [15] . Simulation was performed by using simplified car 3D model and pathloss characteristics has been calculated as follows.

Table 30 On-vehicle to external (LOS) pathloss model parameters in S13

Scenario	Path loss (dB)	Distance (mm)
S13	-70.19	1509

5.7.3 On-vehicle to External (NLOS) (Scenario 13, CM 13) 3.1 – 10.6GHz
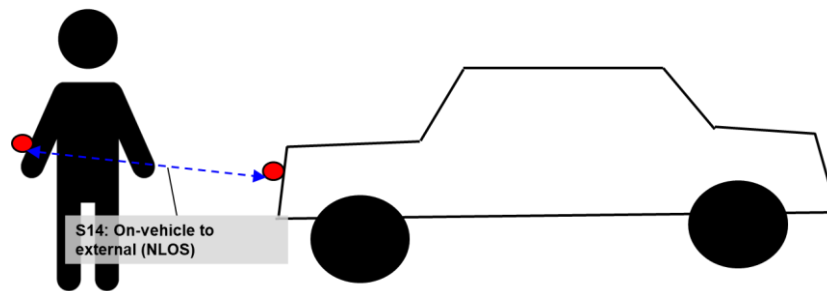
Figure 38 — On-vehicle to External (NLOS) (Scenario 14, CM 14) 3.1 – 10.6GHz

Channel models for on-vehicle to external (NLOS) was provided by the ray-tracing simulation result [15] . Simulation was performed by using simplified car 3D model and pathloss characteristics has been calculated as follows.

Table 31 On-vehicle to external (NLOS) pathloss model parameters in S14

Scenario	Path loss (dB)	Distance (mm)
S14	-92.79	1897

6. MATLAB code program

6.1 Path loss model

The The following MATLAB code can be used to generate path loss model for UWB communication in scenarios S2.1, S2.2, S2.3, S3.1, S6.1 and S6.2 for HBAN. The MATLAB code is based on Nagoya Institute of Technology's simulation results [13] [14] [16]   and provided by Nagoya Institute of Technology.

This MATLAB code can be used for the calculation of path loss for UWB band in VBAN scenarios and channel models such as S8, S8.1 and S8.2 as well as the HBAN models by replacing the channel parameters `PL_0`, `n`, and `sigma_s` in following code.

```
%% 1.
%%Written by Sho Asano, Daisuke Anzai on 3/15/2023

N_ch = 100; % # of path loss realizations
distance = 5; % unit: mm (the distance should be at least 10 mm.)
Transmitting_direction = 1; % 1: Position A (Front-back direction), 2:
Position B (Left-right direction)
Transmitting_antenna = 1; % 1: Inside the head, 2: Outside the head
Received_components = 1; % 1: X-axis, 2: Y-axis, 3: Z-axis
Receiving_antenna = 1; % 1: Whole head, 2: Scalp, 3: Arm, 4: Torso, 5:
Off-body

if distance < 10
    fprintf("Warning!: The distance should be at least 10 mm.\n")
end
% ================== Channel parameter ================
if Transmitting_direction == 1
    if Transmitting_antenna == 1
        if Received_components == 1
            if Receiving_antenna == 1
                d_0 = 10;
                PL_0 = 24.84;  n = 3.84;  sigma_s = 9.52;
            elseif Receiving_antenna == 2
                d_0 = 10;
                PL_0 = 27.13;  n = 3.36;  sigma_s = 10.18;
            elseif Receiving_antenna == 3
                d_0 = 100;
                PL_0 = 23.70;  n = 9.76;  sigma_s = 0.74;
            elseif Receiving_antenna == 4
                d_0 = 100;
                PL_0 = 69.31;  n = 3.85;  sigma_s = 3.02;
            elseif Receiving_antenna == 5
                d_0 = 10;
                PL_0 = 48.47;  n = 1.45;  sigma_s = 9.63;
            end
        elseif Received_components == 2
            if Receiving_antenna == 1
                d_0 = 10;
                PL_0 = 3.78;  n = 6.80;  sigma_s = 7.27;
            elseif Receiving_antenna == 2
                d_0 = 10;
                PL_0 = 11.44;  n = 5.32;  sigma_s = 6.96;
            elseif Receiving_antenna == 3
                d_0 = 100;
                PL_0 = 102.62;  n = 0.57;  sigma_s = 8.15;
            elseif Receiving_antenna == 4
                d_0 = 100;
```

```
            PL_0 = 50.24;  n = 2.76;  sigma_s = 0.65;
        elseif Receiving_antenna == 5
            d_0 = 10;
            PL_0 = 19.13;  n = 2.67;  sigma_s = 2.72;
        end
    elseif Received_components == 3
        if Receiving_antenna == 1
            d_0 = 10;
            PL_0 = 3.73;  n = 6.03;  sigma_s = 9.31;
        elseif Receiving_antenna == 2
            d_0 = 10;
            PL_0 = 17.58;  n = 3.31;  sigma_s = 7.52;
        elseif Receiving_antenna == 3
            d_0 = 100;
            PL_0 = 104.27;  n = 0.59;  sigma_s = 4.98;
        elseif Receiving_antenna == 4
            d_0 = 100;
            PL_0 = 27.52;  n = 8.61;  sigma_s = 1.98;
        elseif Receiving_antenna == 5
            d_0 = 10;
            PL_0 = 22.81;  n = 2.69;  sigma_s = 7.55;
        end
    end
elseif Transmitting_antenna == 2
    if Received_components == 1
        if Receiving_antenna == 1
            d_0 = 10;
            PL_0 = 32.21;  n = 2.55;  sigma_s = 12.94;
        elseif Receiving_antenna == 2
            d_0 = 10;
            PL_0 = 33.07;  n = 2.40;  sigma_s = 14.17;
        elseif Receiving_antenna == 3
            d_0 = 100;
            PL_0 = 60.60;  n = 2.01;  sigma_s = 1.10;
        elseif Receiving_antenna == 4
            d_0 = 100;
            PL_0 = 70.72;  n = 5.04;  sigma_s = 2.69;
        elseif Receiving_antenna == 5
            d_0 = 10;
            PL_0 = 40.26;  n = 1.65;  sigma_s = 6.19;
        end
    elseif Received_components == 2
        if Receiving_antenna == 1
            d_0 = 10;
            PL_0 = 10.72;  n = 5.63;  sigma_s = 8.02;
        elseif Receiving_antenna == 2
            d_0 = 10;
            PL_0 = 14.08;  n = 4.89;  sigma_s = 7.88;
        elseif Receiving_antenna == 3
            d_0 = 100;
            PL_0 = 90.00;  n = 0.66;  sigma_s = 10.08;
        elseif Receiving_antenna == 4
            d_0 = 100;
            PL_0 = 59.62;  n = 3.02;  sigma_s = 0.61;
        elseif Receiving_antenna == 5
            d_0 = 10;
            PL_0 = 23.09;  n = 2.25;  sigma_s = 3.85;
        end
    elseif Received_components == 3
        if Receiving_antenna == 1
            d_0 = 10;
```

```
            PL_0 = 13.00;   n = 4.62;   sigma_s = 8.86;
        elseif Receiving_antenna == 2
            d_0 = 10;
            PL_0 = 18.04;   n = 3.52;   sigma_s = 8.75;
        elseif Receiving_antenna == 3
            d_0 = 100;
            PL_0 = 87.49;   n = 0.45;   sigma_s = 5.35;
        elseif Receiving_antenna == 4
            d_0 = 100;
            PL_0 = 32.56;   n = 9.25;   sigma_s = 1.61;
        elseif Receiving_antenna == 5
            d_0 = 10;
            PL_0 = 22.12;   n = 2.66;   sigma_s = 5.81;
        end
    end
end
elseif Transmitting_direction == 2
    if Transmitting_antenna == 1
        if Received_components == 1
            if Receiving_antenna == 1
                d_0 = 10;
                PL_0 = 20.37;   n = 3.80;   sigma_s = 7.47;
            elseif Receiving_antenna == 2
                d_0 = 10;
                PL_0 = 10.97;   n = 5.63;   sigma_s = 6.89;
            elseif Receiving_antenna == 3
                d_0 = 100;
                PL_0 = 51.42;   n = 3.14;   sigma_s = 1.13;
            elseif Receiving_antenna == 4
                d_0 = 100;
                PL_0 = 16.68;   n = 10.59;  sigma_s = 2.42;
            elseif Receiving_antenna == 5
                d_0 = 10;
                PL_0 = 0.05;   n = 4.95;  sigma_s = 2.66;
            end
        elseif Received_components == 2
            if Receiving_antenna == 1
                d_0 = 10;
                PL_0 = 19.60;   n = 4.96;   sigma_s = 7.59;
            elseif Receiving_antenna == 2
                d_0 = 10;
                PL_0 = 25.39;   n = 3.86;   sigma_s = 7.31;
            elseif Receiving_antenna == 3
                d_0 = 100;
                PL_0 = 80.65;   n = 1.99;   sigma_s = 11.42;
            elseif Receiving_antenna == 4
                d_0 = 100;
                PL_0 = 67.02;   n = 3.72;   sigma_s = 1.80;
            elseif Receiving_antenna == 5
                d_0 = 10;
                PL_0 = 34.94;   n = 2.74;   sigma_s = 5.65;
            end
        elseif Received_components == 3
            if Receiving_antenna == 1
                d_0 = 10;
                PL_0 = 13.36;   n = 4.54;   sigma_s = 7.09;
            elseif Receiving_antenna == 2
                d_0 = 10;
                PL_0 = 19.77;   n = 3.34;   sigma_s = 7.03;
            elseif Receiving_antenna == 3
                d_0 = 100;
```

```
            PL_0 = 76.35;  n = 1.85;  sigma_s = 5.12;
        elseif Receiving_antenna == 4
            d_0 = 100;
            PL_0 = 62.07;  n = 6.87;  sigma_s = 2.61;
        elseif Receiving_antenna == 5
            d_0 = 10;
            PL_0 = 44.11;  n = 0.56;  sigma_s = 5.84;
        end
    end
elseif Transmitting_antenna == 2
    if Received_components == 1
        if Receiving_antenna == 1
            d_0 = 10;
            PL_0 = 17.71;  n = 4.01;  sigma_s = 7.33;
        elseif Receiving_antenna == 2
            d_0 = 10;
            PL_0 = 14.01;  n = 4.84;  sigma_s = 7.51;
        elseif Receiving_antenna == 3
            d_0 = 100;
            PL_0 = 51.22;  n = 3.28;  sigma_s = 1.09;
        elseif Receiving_antenna == 4
            d_0 = 100;
            PL_0 = 15.68;  n = 11.30;  sigma_s = 2.58;
        elseif Receiving_antenna == 5
            d_0 = 10;
            PL_0 = -3.03;  n = 5.25;  sigma_s = 4.76;
        end
    elseif Received_components == 2
        if Receiving_antenna == 1
            d_0 = 10;
            PL_0 = 26.20;  n = 4.19;  sigma_s = 12.94;
        elseif Receiving_antenna == 2
            d_0 = 10;
            PL_0 = 33.17;  n = 2.65;  sigma_s = 12.88;
        elseif Receiving_antenna == 3
            d_0 = 100;
            PL_0 = 78.08;  n = 2.63;  sigma_s = 12.69;
        elseif Receiving_antenna == 4
            d_0 = 100;
            PL_0 = 56.25;  n = 4.89;  sigma_s = 1.52;
        elseif Receiving_antenna == 5
            d_0 = 10;
            PL_0 = 45.33;  n = 1.24;  sigma_s = 6.31;
        end
    elseif Received_components == 3
        if Receiving_antenna == 1
            d_0 = 10;
            PL_0 = 20.14;  n = 4.09;  sigma_s = 8.89;
        elseif Receiving_antenna == 2
            d_0 = 10;
            PL_0 = 23.42;  n = 3.39;  sigma_s = 9.35;
        elseif Receiving_antenna == 3
            d_0 = 100;
            PL_0 = 77.09;  n = 1.87;  sigma_s = 5.07;
        elseif Receiving_antenna == 4
            d_0 = 100;
            PL_0 = 36.17;  n = 10.07;  sigma_s = 1.61;
        elseif Receiving_antenna == 5
            d_0 = 10;
            PL_0 = 42.50;  n = 0.80;  sigma_s = 4.96;
        end
```

```
        end
    end
end
% =====================================================
% =============== Path loss realizations ===============
% -- Path loss in dB
PL=PL_0+10*n*log10(distance/d_0)+sigma_s*randn(1,N_ch);
% =====================================================
```

Annex A

Bibliography

[1]   Kamya Yekeh Yazdandoost, Kamran Sayrafian-Pour, "Channel Model for Body Area Network (BAN)," IEEE P802.15 Working Group for Wireless Personal Area Networks (WPANs), IEEE P802.15-08-0780-12-0006, Nov., 2012.

[2]   W.-T. Chen; H.-R. Chuang, "Numerical computation of human interaction with arbitrarily oriented superquadric loop antennas in personal communications," IEEE Trans. on Antenna and Propagation, vol.46, no. 6, pp. 821-828, June 1998.

[3]   Kamya Y. Yazdandoost and Ryuji Kohno, "The Effect of Human Body on UWB BAN Antennas," IEEE802.15-07-0546-00-0ban.

[4]   C. H. Duney, H. Massoudi, and M. F. Iskander, "Radiofrequency radiation dosimetry handbook," USAF School of Aerospace Medicine, October 1986.

[5]   C. Gabriel and S. Gabriel, "Compilation of the dielectric properties of body tissues at RF and microwave frequencies," AL/OE-TR-1996-0037, June 1996, http://www.brooks.af.mil/AFRL/HED/hedr/reports/dielectricReport/Report.html.

[6]   Italian National Research Council, Institute for Applied Physics, "Dielectric properties of body tissues," http://niremf.ifac.cnr.it

[7]   P. Gandhi, ".Biological effects and medical applications of electromagnetic energy," Prentice Hall, Englewood Cliffs, N.J., 1990.

[8]   Marco Hernandez, Huan-Bang Li, Igor Dotlić, Ryu Miura, "Channel Models for TG8," IEEE P802.15 Working Group for Wireless Personal Area Networks (WPANs), IEEE P802. 15-12-0459-07-0008, Sept., 2012

[9]   3GPP TS 36.104, Evolved Universal Terrestrial Radio Access; Base Station radio transmission and reception

[10]   E. Reusens, W. Joseph, G. Vermeeren, and L. Martens, „On-body measurements and characterization of wireless communication channel fro arm and torso of human," International Workshop on Wearable and Implantabel Body Sensor Networks, BSN07, Achen, March 2007, pp. 26-28.

[11]   A. Fort, J. Ryckaert, C. Desset, P. De Doncker, P. Wambacq, and L. Van Biesen, " Ultrawideband channel model for communication around the human body, " IEEE Journal on Selected Areas in Communications, vol. 24, pp.927-933, April 2006.

[12]   Kamran Sayrafian, "Statistical Pathloss Model for UWB Wireless Capsule Endoscopy," doc.# P.802.15-22-0401-00-6a, 2022.

[13]   Daisuke Anzai, "Propagation characteristics of UWB communication applications including medical implants, BCI and passengers bus," doc.# P.802. 15-22-0469-00-006a, 2022.

[14]   Daisuke Anzai, Sho Asano, "Propagation Channel Parameters of UWB Communication Applications for Human BAN (HBAN) Use Cases," doc.# P.802.15-23-0145-00-006a, 2023.

[15]   Daisuke Anzai, Sho Asano, "Propagation Channel Parameters of UWB Communication Applications for Vehicle BAN (VBAN) Use Cases," doc.# P.802.15-23-0146-00-006a, 2023.

[16]   Daisuke Anzai, "Propagation Characteristics of UWB Communication Applications for Human BAN (HBAN) Use Cases, doc.# P.802. 15-23-0018-00-006a, 2023.

[17]   D. Anzai, I. Balasingham, G. Fischer, J. Wang, "Reliable and High-Speed Implant Ultra-Wideband Communications with Transmit-Receive Diversity," EAI/Springer Innovations in Communication and Computing, pp. 27-32, March 2020.

[18]   Y. Shimizu, D. Anzai, R. C-Santiago, P. A. Floor, I. Balasingham, and J. Wang, "Performance evaluation of an ultra-wideband transmit diversity in a living animal experiment" IEEE Trans. Microw. Theory Tech., vol. 65, no. 7, pp. 2596-2606, July 2017.

[19]   D. Anzai, K. Katsu, R. Chavez-Santiago, Q. Wang, D. Plettemeier, J. Wang, and I. Balasingham, "Experimental evaluation of implant UWB-IR transmission with living animal for body area networks," IEEE Trans. Microw. Theory Tech., vol. 62, no. 1, pp. 183-192, Jan. 2014.

[20]   J. Shi, D. Anzai, and J. Wang, "Channel modeling and performance analysis of diversity reception for implant UWB wireless link," IEICE Trans. Commun., no. E95-B, vol. 10, pp. 3197-3205, Oct. 2012.

[21]   Takahiro Aoyagi, Jun-ichi Takada, Kenichi Takizawa, Norihiko Katayama, Takehiko Kobayashi, Kamya Yekeh Yazdandoost, Huan-bang Li and Ryuji Kohno, "Channel model for wearable and implantable WBANs," IEEE 802.15-08-0416-04-0006, November 2008.

[22]   Guido Dolmans and Andrew Fort, "Channel models WBAN-Holst centre/IMEC-NL," IEEE 802.15-08-0418-01-0006, July 2008.

[23]   Hirokazu Sawada, Takahiro Aoyagi, Jun-ichi Takada, Kamya Yekeh Yazdandoost, Ryuji Kohno, "Channel model between body surface and wireless access point for UWB band," IEEE 802.15-08-0576-00-0006, August 2008.

[24]   Minseok Kim, Jun-ichi Takada, Bin Zhen, Lawrence Materum, Tomoshige Kan, Yuuki Terao, yohei Konishi, Kenji Nakai, Takahiro Aoyagi, Ryuji Kohno, "Statistical property of dynamic BAN channel gain at 4.5 GHz," IEEE 802-.15-08-0489-02-0006, June 2010.

[25]   Andreas F. Molisch, Kannan Balakrishnan, Dajana Cassioli, Chia-Chin Chong, Shahriar Emami, Andrew Fort,Johan Karedal, Juergen Kunisch, Hans Schantz, Ulrich Schuster, Kai Siwiak, "IEEE 802.15.4a channel model - final report,"

[26]   J. Karedal, S. Wyne, P. Almers, F. Tufvesson, and A. F. Molisch, "Statistical analysis of the uwb channel in an industrial environment," in Proc. VTC fall 2004, 2004.

[27]   T. S. Rappaport 1989.

[28]   W. Pietsch 1994.

Annex B

(informative)

Matlab code for IEEE802.15.4a UWB channel model

```
% modified S-V channel model evaluation
%%Written by Sun Xu, Kim Chee Wee, B. Kannan & Francois Chin on 22/02/2005
clear;
no_output_files = 1; % non-zero: avoids writing output files of continuous-time
responses
num_channels = 100; % number of channel impulse responses to generate
randn('state',12); % initialize state of function for repeatability
rand('state',12); % initialize state of function for repeatability
cm_num = 6; % channel model number from 1 to 8
% get channel model params based on this channel model number
[Lam,Lmean,lambda_mode,lambda_1,lambda_2,beta,Gam,gamma_0,Kgamma, ...
sigma_cluster,nlos,gamma_rise,gamma_1,chi,m0,Km,sigma_m0,sigma_Km, ...
sfading_mode,m0_sp,std_shdw,kappa,fc,fs] = uwb_sv_params_15_4a( cm_num );
fprintf(1,['Model Parameters\n' ...
' Lam = %.4f, Lmean = %.4f, lambda_mode(FLAG) = %d\n' ...
' lambda_1 = %.4f, lambda_2 = %.4f, beta = %.4f\n' ...
' Gam = %.4f, gamma0 = %.4f, Kgamma = %.4f, sigma_cluster = %.4f\n' ...
' nlos(FLAG) = %d, gamma_rise = %.4f, gamma_1 = %.4f, chi = %.4f\n' ...
' m0 = %.4f, Km = %.4f, sigma_m0 = %.4f, sigma_Km = %.4f\n' ...
' sfading_mode(FLAG) = %d, m0_sp = %.4f, std_shdw = %.4f\n', ...
' kappa = %.4f, fc = %.4fGHz, fs = %.4fGHz\n'], ...
Lam,Lmean,lambda_mode,lambda_1,lambda_2,beta,Gam,gamma_0,Kgamma, ...
sigma_cluster,nlos,gamma_rise,gamma_1,chi,m0,Km,sigma_m0,sigma_Km,...
sfading_mode,m0_sp,std_shdw,kappa,fc,fs);
ts = 1/fs; % sampling frequency
% get a bunch of realizations (impulse responses)
[h_ct,t_ct,t0,np] = uwb_sv_model_ct_15_4a(Lam,Lmean,lambda_mode,lambda_1, ...
lambda_2,beta,Gam,gamma_0,Kgamma,sigma_cluster,nlos,gamma_rise,gamma_1, ...
chi,m0,Km,sigma_m0,sigma_Km,sfading_mode,m0_sp,std_shdw,num_channels,ts);
% change to complex baseband channel
h_ct_len = size(h_ct, 1);
phi = zeros(h_ct_len, 1);
for k = 1:num_channels
phi = rand(h_ct_len, 1).*(2*pi);
h_ct(:,k) = h_ct(:,k) .* exp(phi .* i);
end
% now reduce continuous-time result to a discrete-time result
[hN,N] = uwb_sv_cnvrt_ct_15_4a( h_ct, t_ct, np, num_channels, ts );
if N > 1,
h = resample(hN, 1, N); % decimate the columns of hN by factor N
else
```

```
h = hN;
end
% add the frequency dependency
[h]= uwb_sv_freq_depend_ct_15_4a(h,fc,fs,num_channels,kappa);
%**********************************************************
% Testing and ploting
%**********************************************************
% channel energy
channel_energy = sum(abs(h).^2);
h_len = length(h(:,1));
t = [0:(h_len-1)] * ts; % for use in computing excess & RMS delays
excess_delay = zeros(1,num_channels);
RMS_delay = zeros(1,num_channels);
num_sig_paths = zeros(1,num_channels);
num_sig_e_paths = zeros(1,num_channels);
for k=1:num_channels
% determine excess delay and RMS delay
sq_h = abs(h(:,k)).^2 / channel_energy(k);
t_norm = t - t0(k); % remove the randomized arrival time of first cluster
excess_delay(k) = t_norm * sq_h;
RMS_delay(k) = sqrt( ((t_norm-excess_delay(k)).^2) * sq_h );
% determine number of significant paths (paths within 10 dB from peak)
threshold_dB = -10; % dB
temp_h = abs(h(:,k));
temp_thresh = 10^(threshold_dB/20) * max(temp_h);
num_sig_paths(k) = sum(temp_h > temp_thresh);
% determine number of sig. paths (captures x % of energy in channel)
x = 0.85;
temp_sort = sort(temp_h.^2); % sorted in ascending order of energy
cum_energy = cumsum(temp_sort(end:-1:1)); % cumulative energy
index_e = min(find(cum_energy >= x * cum_energy(end)));
num_sig_e_paths(k) = index_e;
end
energy_mean = mean(10*log10(channel_energy));
energy_stddev = std(10*log10(channel_energy));
mean_excess_delay = mean(excess_delay);
mean_RMS_delay = mean(RMS_delay);
mean_sig_paths = mean(num_sig_paths);
mean_sig_e_paths = mean(num_sig_e_paths);
fprintf(1,'Model Characteristics\n');
fprintf(1,' Mean delays: excess (tau_m) = %.1f ns, RMS (tau_rms) = %1.f\n', ...
mean_excess_delay, mean_RMS_delay);
fprintf(1,' # paths: NP_10dB = %.1f, NP_85%% = %.1f\n', ...
mean_sig_paths, mean_sig_e_paths);
fprintf(1,' Channel energy: mean = %.1f dB, std deviation = %.1f dB\n', ...
energy_mean, energy_stddev);
figure(1); clf; plot(t, abs(h)); grid on
```

```
title('Impulse response realizations')
xlabel('Time (nS)')
figure(2); clf; plot([1:num_channels], excess_delay, 'b-', ...
[1 num_channels], mean_excess_delay*[1 1], 'r--' );
grid on
title('Excess delay (nS)')
xlabel('Channel number')
figure(3); clf; plot([1:num_channels], RMS_delay, 'b-', ...
[1 num_channels], mean_RMS_delay*[1 1], 'r--' );
grid on
title('RMS delay (nS)')
xlabel('Channel number')
figure(4); clf; plot([1:num_channels], num_sig_paths, 'b-', ...
[1 num_channels], mean_sig_paths*[1 1], 'r--');
grid on
title('Number of significant paths within 10 dB of peak')
xlabel('Channel number')
figure(5); clf; plot([1:num_channels], num_sig_e_paths, 'b-', ...
[1 num_channels], mean_sig_e_paths*[1 1], 'r--');
grid on
title('Number of significant paths capturing > 85% energy')

xlabel('Channel number')
temp_average_power = sum((abs(h))'.*(abs(h))', 1)/num_channels;
temp_average_power = temp_average_power/max(temp_average_power);
average_decay_profile_dB = 10*log10(temp_average_power);
threshold_dB = -40;

above_threshold = find(average_decay_profile_dB > threshold_dB);

ave_t = t(above_threshold);
apdf_dB = average_decay_profile_dB(above_threshold);
figure(6); clf; plot(ave_t, apdf_dB); grid on
title('Average Power Decay Profile')
xlabel('Delay (nsec)')
ylabel('Average power (dB)')
if no_output_files,
return
end
%*************************************************************************
*******
%Savinge the data
%*************************************************************************
*******
%%% save continuous-time (time,value) pairs to files
save_fn = sprintf('cm%d_imr', cm_num);
% A complete self-contained file for Matlab users
save([save_fn '.mat'], 't', 'h','t_ct', 'h_ct', 't0', 'np', 'num_channels', 'cm_num');
% Three comma-delimited text files for non-Matlab users:
% File #1: cmX_imr_np.csv lists the number of paths in each realization
dlmwrite([save_fn '_np.csv'], np, ','); % number of paths
% File #2: cmX_imr_ct.csv can open with Excel
```

```
% n'th pair of columns contains the (time,value) pairs for the n'th realization
% save continous time data
th_ct = zeros(size(t_ct,1),3*size(t_ct,2));
th_ct(:,1:3:end) = t_ct; % time
th_ct(:,2:3:end) = abs(h_ct); % magnitude
th_ct(:,3:3:end) = angle(h_ct); % phase (radians)
fid = fopen([save_fn '_ct.csv'], 'w');

if fid < 0,

error('unable to write .csv file for impulse response, file may be open in another application');
end
for k = 1:size(th_ct,1)
fprintf(fid,'%.4f,%.6f,', th_ct(k,1:end-2));
fprintf(fid,'%.4f,%.6f\r\n', th_ct(k,end-1:end)); % \r\n for Windoze end-of-line
end
fclose(fid);
% File #3: cmX_imr_dt.csv can open with Excel
% discrete channel impulse response magnitude and phase pair realization.
% the first column is time. phase is in radians
% save discrete time data
th = zeros(size(h,1),2*size(h,2)+1);
th(:,1) = t'; % the first column is time scale
th(:,2:2:end) = abs(h); % even columns are magnitude
th(:,3:2:end) = angle(h); % odd columns are phase
fid = fopen([save_fn '_dt.csv'], 'w');

if fid < 0,

error('unable to write .csv file for impulse response, file may be open in another application');
end
for k = 1:size(th,1)
fprintf(fid,'%.4f,%.6f,', th(k,1:end-2));
fprintf(fid,'%.4f,%.6f\r\n', th(k,end-1:end)); % \r\n for Windoze end-of-line
end
fclose(fid);
return; % end of program
function
[Lam,Lmean,lambda_mode,lambda_1,lambda_2,beta,Gam,gamma_0,Kgamma, ...
sigma_cluster,nlos,gamma_rise,gamma_1,chi,m0,Km,sigma_m0,sigma_Km, ...
sfading_mode,m0_sp,std_shdw,kappa,fc,fs] = uwb_sv_params_15_4a( cm_num )
% Written by Sun Xu, Kim Chee Wee, B. Kannan & Francois Chin on 22/02/2004
% Return modified S-V model parameters for standard UWB channel models
%─────────────────────────────────────────
% Lam Cluster arrival rate (clusters per nsec)
% Lmean Mean number of Clusters
% lambda_mode Flag for Mixture of poission processes for ray arrival times

% 1 -> Mixture of poission processes for the ray arrival times

% 2 -> tapped delay line model

% lambda_1 Ray arrival rate for Mixture of poisson processes (rays per nsec)
```

% lambda_2 Ray arrival rate for Mixture of poisson processes (rays per nsec)
% beta Mixture probability
%────────────────────────────
% Gam Cluster decay factor (time constant, nsec)
% gamma0 Ray decay factor (time constant, nsec)
% Kgamma Time dependence of ray decay factor
% sigma_cluster Standard deviation of normally distributed variable for cluster energy
% nlos Flag for non line of sight channel
% 0 -> LOS

% 1 -> NLOS with first arrival path starting at t ~= 0

% 2 -> NLOS with first arrival path starting at t = 0 and diffused first cluster

% gamma_rise Ray decay factor of diffused first cluster (time constant, nsec)
% gamma_1 Ray decay factor of diffused first cluster (time constant, nsec)
% chi Diffuse weight of diffused first cluster
%────────────────────────────
% m0 Mean of log-normal distributed nakagami-m factor
% Km Time dependence of m0
% sigma_m0 Standard deviation of log-normal distributed nakagami-m factor
% sigma_Km Time dependence of sigma_m0
% sfading_mode Flag for small-scale fading
% 0 -> All paths have same m-factor distribution

% 1 -> LOS first path has a deterministic large m-factor

% 2 -> LOS first path of each cluster has a deterministic

% large m-factor
% m0_sp Deterministic large m-factor
%────────────────────────────
% std_shdw Standard deviation of log-normal shadowing of entire impulse response
%────────────────────────────
% kappa Frequency dependency of the channel
%────────────────────────────
% fc Center Frequency
% fs Frequency Range
%%
modified by I2R
if cm_num == 1, % Residential LOS
% MPC arrival
Lam = 0.047; Lmean = 3;
lambda_mode = 1;
lambda_1 = 1.54; lambda_2 = 0.15; beta = 0.095;
%MPC decay
Gam = 22.61; gamma_0 = 12.53; Kgamma = 0; sigma_cluster = 2.75;
nlos = 0;
gamma_rise = NaN; gamma_1 = NaN; chi = NaN; % dummy in this scenario
% Small-scale Fading
m0 = 0.67; Km = 0; sigma_m0 = 0.28; sigma_Km = 0;
sfading_mode = 0; m0_sp = NaN;

% Large-scale Fading – Shadowing
std_shdw = 2.22;
% Frequency Dependence
kappa = 1.12;
fc = 6; % GHz
fs = 8; % 2 - 10 GHz
elseif cm_num == 2, % Residential NLOS
% MPC arrival
Lam = 0.12; Lmean = 3.5;
lambda_mode = 1;
lambda_1 = 1.77; lambda_2 = 0.15; beta = 0.045;
%MPC decay
Gam = 26.27; gamma_0 = 17.5; Kgamma = 0; sigma_cluster = 2.93;
nlos = 1;
gamma_rise = NaN; gamma_1 = NaN; chi = NaN; % dummy in this scenario
% Small-scale Fading
m0 = 0.69; Km = 0; sigma_m0 = 0.32; sigma_Km = 0;
sfading_mode = 0; m0_sp = NaN;
% Large-scale Fading – Shadowing
std_shdw = 3.51;
% Frequency Dependence
kappa = 1.53;
fc = 6; % GHz
fs = 8; % 2 - 10 GHz
elseif cm_num == 3, % Office LOS
% MPC arrival
Lam = 0.016; Lmean = 5.4;
lambda_mode = 1;
lambda_1 = 0.19; lambda_2 = 2.97; beta = 0.0184;
%MPC decay
Gam = 14.6; gamma_0 = 6.4; Kgamma = 0; sigma_cluster = 3; % assumption
nlos = 0;
gamma_rise = NaN; gamma_1 = NaN; chi = NaN; % dummy in this scenario
% Small-scale Fading
m0 = 0.42; Km = 0; sigma_m0 = 0.31; sigma_Km = 0;
sfading_mode = 2; m0_sp = 3; % assumption
% Large-scale Fading – Shadowing
std_shdw = 0; %1.9;
% Frequency Dependence
kappa = 0.03;
fc = 6; % GHz
fs = 8; % 3 - 6 GHz
elseif cm_num == 4, % Office NLOS
% MPC arrival
Lam = 0.19; Lmean = 3.1;
lambda_mode = 1;
lambda_1 = 0.11; lambda_2 = 2.09; beta = 0.0096;
%MPC decay
Gam = 19.8; gamma_0 = 11.2; Kgamma = 0; sigma_cluster = 3; % assumption
nlos = 2;

gamma_rise = 15.21; gamma_1 = 11.84; chi = 0.78;
% Small-scale Fading
m0 = 0.5; Km = 0; sigma_m0 = 0.25; sigma_Km = 0;
sfading_mode = 0; m0_sp = NaN; % assumption
% Large-scale Fading – Shadowing
std_shdw = 3.9;
% Frequency Dependence
kappa =0.71;
fc = 6; % GHz
fs = 8; % 3 - 6 GHz
elseif cm_num == 5, % Outdoor LOS
% MPC arrival
Lam = 0.0448; Lmean = 13.6;
lambda_mode = 1;
lambda_1 = 0.13; lambda_2 = 2.41; beta = 0.0078;
%MPC decay
Gam = 31.7; gamma_0 = 3.7; Kgamma = 0; sigma_cluster = 3; % assumption
nlos = 0;
gamma_rise = NaN; gamma_1 = NaN; chi = NaN; % dummy in this scenario
% Small-scale Fading
m0 = 0.77; Km = 0; sigma_m0 = 0.78; sigma_Km = 0;
sfading_mode = 2; m0_sp = 3; % assumption
% Large-scale Fading – Shadowing
std_shdw = 0.83;
% Frequency Dependence
kappa = 0.12;
fc = 6; % GHz
fs = 8; % 3 - 6 GHz
elseif cm_num == 6, % Outdoor NLOS
% MPC arrival
Lam = 0.0243; Lmean = 10.5;
lambda_mode = 1;
lambda_1 = 0.15; lambda_2 = 1.13; beta = 0.062;
%MPC decay
Gam = 104.7; gamma_0 = 9.3; Kgamma = 0; sigma_cluster = 3; % assumption
nlos = 1;
gamma_rise = NaN; gamma_1 = NaN; chi = NaN; % dummy in this scenario
% Small-scale Fading
m0 = 0.56; Km = 0; sigma_m0 = 0.25; sigma_Km = 0;
sfading_mode = 0; m0_sp = NaN; % assumption
% Large-scale Fading – Shadowing
std_shdw = 2; % assumption
% Frequency Dependence
kappa = 0.13;
fc =6; % GHz
fs = 8; % 3 - 6 GHz
elseif cm_num == 7, % Industrial LOS
% MPC arrival
Lam = 0.0709; Lmean = 4.75;
lambda_mode = 2;

lambda_1 = 1; lambda_2 = 1; beta = 1; % dummy in this scenario
%MPC decay
Gam = 13.47; gamma_0 = 0.615; Kgamma = 0.926; sigma_cluster = 4.32;
nlos = 0;
gamma_rise = NaN; gamma_1 = NaN; chi = NaN; % dummy in this scenario
% Small-scale Fading
m0 = 0.36; Km = 0; sigma_m0 = 1.13; sigma_Km = 0;
sfading_mode = 1; m0_sp = 12.99;
% Large-scale Fading – Shadowing
std_shdw = 6;
% Frequency Dependence
kappa = -1.103;
fc = 6; % GHz
fs = 8; % 2 - 8 GHz
elseif cm_num == 8, % Industrial NLOS
% MPC arrival
Lam = 0.089; Lmean = 1;
lambda_mode = 2;
lambda_1 = 1; lambda_2 = 1; beta = 1; % dummy in this scenario
%MPC decay
Gam = 5.83; gamma_0 = 0.3; Kgamma = 0.44; sigma_cluster = 2.88;
nlos = 2;
gamma_rise = 47.23; gamma_1 = 84.15; chi = 0.99;
% Small-scale Fading
m0 = 0.3; Km = 0; sigma_m0 = 1.15; sigma_Km = 0;
sfading_mode = 0; m0_sp = NaN; % m0_sp is assumption
% Large-scale Fading – Shadowing
std_shdw = 6;
% Frequency Dependence
kappa = -1.427;
fc = 6; % GHz
fs = 8; % 2 - 8 GHz
elseif cm_num == 9, % Open Outdoor Environment NLOS (Fram, Snow-Covered Open Area)
% MPC arrival
Lam = 0.0305; Lmean = 3.31;
lambda_mode = 1;
lambda_1 = 0.0225; lambda_2 = 1; beta = 1;
%MPC decay
Gam = 56; gamma_0 = 0.92; Kgamma = 0; sigma_cluster = 3; % sigma_cluster is assumption
nlos = 1;
gamma_rise = NaN; gamma_1 = NaN; chi = NaN;
% Small-scale Fading
m0 = 4.1; Km = 0; sigma_m0 = 2.5; sigma_Km = 0;
sfading_mode = 0; m0_sp = NaN; % m0_sp is assumption
% Large-scale Fading – Shadowing
std_shdw = 3.96;
% Frequency Dependence
kappa = -1; % Kappa is assumption

```
fc = 6; % GHz
fs = 8; % 2 - 8 GHz
else
error('cm_num is wrong!!')
end
return
function [h]= uwb_sv_freq_depend_ct_15_4a(h,fc,fs,num_channels,kappa)
% This function is used to include the frequency dependency
f0 = 5; % GHz
h_len = length(h(:,1));
f = [fc-fs/2 : fs/h_len/2 : fc+fs/2]./f0;
f = f.^(-2*(kappa));
f = [f(h_len : 2*h_len), f(1 : h_len-1)]';
i = (-1)^(1/2); % complex i
for c = 1:num_channels
% add the frequency dependency
h2 = zeros(2*h_len, 1);
h2(1 : h_len) = h(:,c); % zero padding
fh2 = fft(h2);
fh2 = fh2 .* f;
h2 = ifft(fh2);
h(:,c) = h2(1:h_len);
% Normalize the channel energy to 1
h(:,c) = h(:,c)/sqrt(h(:,c)' * h(:,c) );
end
return
function [h,t,t0,np] =
uwb_sv_model_ct_15_4a(Lam,Lmean,lambda_mode,lambda_1, ...
lambda_2,beta,Gam,gamma_0,Kgamma,sigma_cluster,nlos,gamma_rise,gamma_1, ...
chi,m0,Km,sigma_m0,sigma_Km,sfading_mode,m0_sp,std_shdw,num_channels,ts)
% Written by Sun Xu, Kim Chee Wee, B. Kannan & Francois Chin on 22/02/2005
% IEEE 802.15.4a UWB channel model for PHY proposal evaluation
% continuous-time realization of modified S-V channel model
% Input parameters:
% detailed introduction of input parameters is at uwb_sv_params.m
% num_channels number of random realizations to generate
% Outputs
% h is returned as a matrix with num_channels columns, each column
% holding a random realization of the channel model (an impulse response)
% t is organized as h, but holds the time instances (in nsec) of the paths whose
% signed amplitudes are stored in h
% t0 is the arrival time of the first cluster for each realization
% np is the number of paths for each realization.
% Thus, the k'th realization of the channel impulse response is the sequence
% of (time,value) pairs given by (t(1:np(k),k), h(1:np(k),k))
%%%
modified by I2R
% initialize and precompute some things
std_L = 1/sqrt(2*Lam); % std dev (nsec) of cluster arrival spacing
std_lam_1 = 1/sqrt(2*lambda_1);
```

```
std_lam_2 = 1/sqrt(2*lambda_2);
% std_lam = 1/sqrt(2*lambda); % std dev (nsec) of ray arrival spacing
h_len = 1000; % there must be a better estimate of # of paths than this
ngrow = 1000; % amount to grow data structure if more paths are needed
h = zeros(h_len,num_channels);
t = zeros(h_len,num_channels);
t0 = zeros(1,num_channels);
np = zeros(1,num_channels);
for k = 1:num_channels % loop over number of channels
tmp_h = zeros(size(h,1),1);
tmp_t = zeros(size(h,1),1);
if nlos == 1,
Tc = (std_L*randn)^2 + (std_L*randn)^2; % First cluster random arrival
else
Tc = 0; % First cluster arrival occurs at time 0
end
t0(k) = Tc;
if nlos == 2 & lambda_mode == 2
L = 1; % for industrial NLOS environment
else
L = max(1, poissrnd(Lmean)); % number of clusters
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
if Kgamma ~= 0 & nlos == 0
Tcval = []; Tc_cluster= [];
Tc_cluster(1,1)=Tc;
for i_Tc=2:L+1
Tc_cluster(1,i_Tc)= Tc_cluster(1,i_Tc-1)+(std_L*randn)^2 + (std_L*randn)^2;
end
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
cluster_index = zeros(1,L);
path_ix = 0;
nak_m = [];
for ncluster = 1:L
% Determine Ray arrivals for each cluster
Tr = 0; % first ray arrival defined to be time 0 relative to cluster
cluster_index(ncluster) = path_ix+1; % remember the cluster location
gamma = Kgamma*Tc + gamma_0; % delay dependent cluster decay time
if nlos == 2 & ncluster == 1
gamma = gamma_1;
end
Mcluster = sigma_cluster*randn;
Pcluster = 10*log10(exp(-1*Tc/Gam))+Mcluster; % total cluster power
Pcluster = 10^(Pcluster*0.1);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
if Kgamma ~= 0 & nlos == 0
Tr_len=Tc_cluster(1,ncluster+1)-Tc_cluster(1,ncluster);
```

```
else
Tr_len = 10*gamma;
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

while (Tr < Tr_len),
t_val = (Tc+Tr); % time of arrival of this ray
if nlos == 2 & ncluster == 1
% equation (22)
h_val = Pcluster*(1-chi*exp(-Tr/gamma_rise))*exp(-Tr/gamma_1) ...
*(gamma+gamma_rise)/gamma/(gamma+gamma_rise*(1-chi));
else
% equation (19)
h_val = Pcluster/gamma*exp(-Tr/gamma)/(beta*lambda_1+(1-beta)*lambda_2+1);
end
path_ix = path_ix + 1; % row index of this ray

if path_ix > h_len,
% grow the output structures to handle more paths as needed
tmp_h = [tmp_h; zeros(ngrow,1)];
tmp_t = [tmp_t; zeros(ngrow,1)];
h = [h; zeros(ngrow,num_channels)];
t = [t; zeros(ngrow,num_channels)];
h_len = h_len + ngrow;
end
tmp_h(path_ix) = h_val;
tmp_t(path_ix) = t_val;
% if lambda_mode == 0
% Tr = Tr + (std_lam*randn)^2 + (std_lam*randn)^2;
if lambda_mode == 1

if rand < beta
Tr = Tr + (std_lam_1*randn)^2 + (std_lam_1*randn)^2;
else
Tr = Tr + (std_lam_2*randn)^2 + (std_lam_2*randn)^2;
end
elseif lambda_mode == 2
Tr = Tr + ts;
else
error('lambda mode is wrong!')
end
% generate log-normal distributed nakagami m-factor
m_mu = m0 - Km*t_val;
m_std = sigma_m0 - sigma_Km*t_val;
nak_m = [nak_m, lognrnd(m_mu, m_std)];
end
Tc = Tc + (std_L*randn)^2 + (std_L*randn)^2;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%
if Kgamma ~= 0 & nlos == 0
Tc = Tc_cluster(1,ncluster+1);
end
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%
end
% change m value of the first multipath to be the deterministic value
if sfading_mode == 1
nak_ms(cluster_index(1)) = m0_sp;
elseif sfading_mode == 2
nak_ms(cluster_index) = m0_sp;
end
% apply nakagami
for path = 1:path_ix
h_val = (gamrnd(nak_m(path), tmp_h(path)/nak_m(path))).^(1/2);
tmp_h(path) = h_val;
end
np(k) = path_ix; % number of rays (or paths) for this realization
[sort_tmp_t,sort_ix] = sort(tmp_t(1:np(k))); % sort in ascending time order
t(1:np(k),k) = sort_tmp_t;
h(1:np(k),k) = tmp_h(sort_ix(1:np(k)));
% now impose a log-normal shadowing on this realization
% fac = 10^(std_shdw*randn/20) / sqrt( h(1:np(k),k)' * h(1:np(k),k) );
% h(1:np(k),k) = h(1:np(k),k) * fac;
end
return
function [hN,N] = uwb_sv_cnvrt_ct_15_4a( h_ct, t, np, num_channels, ts )
% convert continuous-time channel model h_ct to N-times oversampled discrete-time
samples
% h_ct, t, np, and num_channels are as specified in uwb_sv_model
% ts is the desired time resolution
%%hN will be produced with time resolution ts /
N.
% It is up to the user to then apply any filtering and/or complex downconversion and
then
% decimate by N to finally obtain an impulse response at time resolution ts.
min_Nfs = 100; % GHz
N = max( 1, ceil(min_Nfs*ts) ); % N*fs = N/ts is the intermediate sampling frequency
before decimation
N = 2^nextpow2(N); % make N a power of 2 to facilitate efficient multi-stage
decimation
Nfs = N / ts;
t_max = max(t(:)); % maximum time value across all channels
h_len = 1 + floor(t_max * Nfs); % number of time samples at resolution ts / N
hN = zeros(h_len,num_channels);
for k = 1:num_channels
np_k = np(k); % number of paths in this channel
t_Nfs = 1 + floor(t(1:np_k,k) * Nfs); % vector of quantized time indices for this
channel
for n = 1:np_k
hN(t_Nfs(n),k) = hN(t_Nfs(n),k) + h_ct(n,k);
end
end
```

Annex C

(informative)

MATLAB script for the IEEE 802.15.4a channel model

```
% modified S-V channel model evaluation
%%Written by Sun Xu, Kim Chee Wee, B. Kannan & Francois Chin on 22/02/2005
clear;
no_output_files = 1; % non-zero: avoids writing output files of continuous-time
responses
num_channels = 100; % number of channel impulse responses to generate
randn('state',12); % initialize state of function for repeatability
rand('state',12); % initialize state of function for repeatability
cm_num = 6; % channel model number from 1 to 8
% get channel model params based on this channel model number
[Lam,Lmean,lambda_mode,lambda_1,lambda_2,beta,Gam,gamma_0,Kgamma, ...
sigma_cluster,nlos,gamma_rise,gamma_1,chi,m0,Km,sigma_m0,sigma_Km, ...
sfading_mode,m0_sp,std_shdw,kappa,fc,fs] = uwb_sv_params_15_4a( cm_num );
fprintf(1,['Model Parameters\n' ...
' Lam = %.4f, Lmean = %.4f, lambda_mode(FLAG) = %d\n' ...
' lambda_1 = %.4f, lambda_2 = %.4f, beta = %.4f\n' ...
' Gam = %.4f, gamma0 = %.4f, Kgamma = %.4f, sigma_cluster = %.4f\n' ...
' nlos(FLAG) = %d, gamma_rise = %.4f, gamma_1 = %.4f, chi = %.4f\n' ...
' m0 = %.4f, Km = %.4f, sigma_m0 = %.4f, sigma_Km = %.4f\n' ...
' sfading_mode(FLAG) = %d, m0_sp = %.4f, std_shdw = %.4f\n', ...
' kappa = %.4f, fc = %.4fGHz, fs = %.4fGHz\n'], ...
Lam,Lmean,lambda_mode,lambda_1,lambda_2,beta,Gam,gamma_0,Kgamma, ...
sigma_cluster,nlos,gamma_rise,gamma_1,chi,m0,Km,sigma_m0,sigma_Km,...
sfading_mode,m0_sp,std_shdw,kappa,fc,fs);
ts = 1/fs; % sampling frequency
% get a bunch of realizations (impulse responses)
[h_ct,t_ct,t0,np] = uwb_sv_model_ct_15_4a(Lam,Lmean,lambda_mode,lambda_1, ...
lambda_2,beta,Gam,gamma_0,Kgamma,sigma_cluster,nlos,gamma_rise,gamma_1, ...
chi,m0,Km,sigma_m0,sigma_Km,sfading_mode,m0_sp,std_shdw,num_channels,ts);
% change to complex baseband channel
h_ct_len = size(h_ct, 1);
phi = zeros(h_ct_len, 1);
for k = 1:num_channels
phi = rand(h_ct_len, 1).*(2*pi);
h_ct(:,k) = h_ct(:,k) .* exp(phi .* i);
end
% now reduce continuous-time result to a discrete-time result
[hN,N] = uwb_sv_cnvrt_ct_15_4a( h_ct, t_ct, np, num_channels, ts );
if N > 1,
h = resample(hN, 1, N); % decimate the columns of hN by factor N
else
```

```
h = hN;
end
% add the frequency dependency
[h]= uwb_sv_freq_depend_ct_15_4a(h,fc,fs,num_channels,kappa);
%*************************************************************
% Testing and ploting
%*************************************************************
% channel energy
channel_energy = sum(abs(h).^2);
h_len = length(h(:,1));
t = [0:(h_len-1)] * ts; % for use in computing excess & RMS delays
excess_delay = zeros(1,num_channels);
RMS_delay = zeros(1,num_channels);
num_sig_paths = zeros(1,num_channels);
num_sig_e_paths = zeros(1,num_channels);
for k=1:num_channels
% determine excess delay and RMS delay
sq_h = abs(h(:,k)).^2 / channel_energy(k);
t_norm = t - t0(k); % remove the randomized arrival time of first cluster
excess_delay(k) = t_norm * sq_h;
RMS_delay(k) = sqrt( ((t_norm-excess_delay(k)).^2) * sq_h );
% determine number of significant paths (paths within 10 dB from peak)
threshold_dB = -10; % dB
temp_h = abs(h(:,k));
temp_thresh = 10^(threshold_dB/20) * max(temp_h);
num_sig_paths(k) = sum(temp_h > temp_thresh);
% determine number of sig. paths (captures x % of energy in channel)
x = 0.85;
temp_sort = sort(temp_h.^2); % sorted in ascending order of energy
cum_energy = cumsum(temp_sort(end:-1:1)); % cumulative energy
index_e = min(find(cum_energy >= x * cum_energy(end)));
num_sig_e_paths(k) = index_e;
end
energy_mean = mean(10*log10(channel_energy));
energy_stddev = std(10*log10(channel_energy));
mean_excess_delay = mean(excess_delay);
mean_RMS_delay = mean(RMS_delay);
mean_sig_paths = mean(num_sig_paths);
mean_sig_e_paths = mean(num_sig_e_paths);
fprintf(1,'Model Characteristics\n');
fprintf(1,' Mean delays: excess (tau_m) = %.1f ns, RMS (tau_rms) = %1.f\n', ...
mean_excess_delay, mean_RMS_delay);
fprintf(1,' # paths: NP_10dB = %.1f, NP_85%% = %.1f\n', ...
mean_sig_paths, mean_sig_e_paths);
fprintf(1,' Channel energy: mean = %.1f dB, std deviation = %.1f dB\n', ...
energy_mean, energy_stddev);
figure(1); clf; plot(t, abs(h)); grid on
```

```
title('Impulse response realizations')
xlabel('Time (nS)')
figure(2); clf; plot([1:num_channels], excess_delay, 'b-', ...
[1 num_channels], mean_excess_delay*[1 1], 'r–' );
grid on
title('Excess delay (nS)')
xlabel('Channel number')
figure(3); clf; plot([1:num_channels], RMS_delay, 'b-', ...
[1 num_channels], mean_RMS_delay*[1 1], 'r–' );
grid on
title('RMS delay (nS)')
xlabel('Channel number')
figure(4); clf; plot([1:num_channels], num_sig_paths, 'b-', ...
[1 num_channels], mean_sig_paths*[1 1], 'r–');
grid on
title('Number of significant paths within 10 dB of peak')
xlabel('Channel number')
figure(5); clf; plot([1:num_channels], num_sig_e_paths, 'b-', ...
[1 num_channels], mean_sig_e_paths*[1 1], 'r–');
grid on
title('Number of significant paths capturing > 85% energy')
xlabel('Channel number')
temp_average_power = sum((abs(h))'.*(abs(h))', 1)/num_channels;
temp_average_power = temp_average_power/max(temp_average_power);
average_decay_profile_dB = 10*log10(temp_average_power);
threshold_dB = -40;
above_threshold = find(average_decay_profile_dB > threshold_dB);
ave_t = t(above_threshold);
apdf_dB = average_decay_profile_dB(above_threshold);
figure(6); clf; plot(ave_t, apdf_dB); grid on
title('Average Power Decay Profile')
xlabel('Delay (nsec)')
ylabel('Average power (dB)')
if no_output_files,
return
end
%****************************************************************
*******
%Savinge the data
%****************************************************************
*******
%%%% save continuous-time (time,value) pairs to files
save_fn = sprintf('cm%d_imr', cm_num);
% A complete self-contained file for Matlab users
save([save_fn '.mat'], 't', 'h','t_ct', 'h_ct', 't0', 'np', 'num_channels', 'cm_num');
% Three comma-delimited text files for non-Matlab users:
% File #1: cmX_imr_np.csv lists the number of paths in each realization
dlmwrite([save_fn '_np.csv'], np, ','); % number of paths
% File #2: cmX_imr_ct.csv can open with Excel
```

```
% n'th pair of columns contains the (time,value) pairs for the n'th realization
% save continous time data
th_ct = zeros(size(t_ct,1),3*size(t_ct,2));
th_ct(:,1:3:end) = t_ct; % time
th_ct(:,2:3:end) = abs(h_ct); % magnitude
th_ct(:,3:3:end) = angle(h_ct); % phase (radians)
fid = fopen([save_fn '_ct.csv'], 'w');

if fid < 0,

error('unable to write .csv file for impulse response, file may be open in another
application');
end
for k = 1:size(th_ct,1)
fprintf(fid,'%.4f,%.6f,', th_ct(k,1:end-2));
fprintf(fid,'%.4f,%.6f\r\n', th_ct(k,end-1:end)); % \r\n for Windoze end-of-line
end
fclose(fid);
% File #3: cmX_imr_dt.csv can open with Excel
% discrete channel impulse response magnitude and phase pair realization.
% the first column is time. phase is in radians
% save discrete time data
th = zeros(size(h,1),2*size(h,2)+1);
th(:,1) = t'; % the first column is time scale
th(:,2:2:end) = abs(h); % even columns are magnitude
th(:,3:2:end) = angle(h); % odd columns are phase
fid = fopen([save_fn '_dt.csv'], 'w');

if fid < 0,

error('unable to write .csv file for impulse response, file may be open in another
application');
end
for k = 1:size(th,1)
fprintf(fid,'%.4f,%.6f,', th(k,1:end-2));
fprintf(fid,'%.4f,%.6f\r\n', th(k,end-1:end)); % \r\n for Windoze end-of-line
end
fclose(fid);
return; % end of program
function
[Lam,Lmean,lambda_mode,lambda_1,lambda_2,beta,Gam,gamma_0,Kgamma, ...
sigma_cluster,nlos,gamma_rise,gamma_1,chi,m0,Km,sigma_m0,sigma_Km, ...
sfading_mode,m0_sp,std_shdw,kappa,fc,fs] = uwb_sv_params_15_4a( cm_num )
% Written by Sun Xu, Kim Chee Wee, B. Kannan & Francois Chin on 22/02/2004
% Return modified S-V model parameters for standard UWB channel models
%————————————————————————————
% Lam Cluster arrival rate (clusters per nsec)
% Lmean Mean number of Clusters
% lambda_mode Flag for Mixture of poission processes for ray arrival times

% 1 -> Mixture of poission processes for the ray arrival times

% 2 -> tapped delay line model

% lambda_1 Ray arrival rate for Mixture of poisson processes (rays per nsec)
```

```
% lambda_2 Ray arrival rate for Mixture of poisson processes (rays per nsec)
% beta Mixture probability
%───────────────────────────────
% Gam Cluster decay factor (time constant, nsec)
% gamma0 Ray decay factor (time constant, nsec)
% Kgamma Time dependence of ray decay factor
% sigma_cluster Standard deviation of normally distributed variable for cluster
energy
% nlos Flag for non line of sight channel
% 0 -> LOS

% 1 -> NLOS with first arrival path starting at t ~= 0

% 2 -> NLOS with first arrival path starting at t = 0 and diffused first cluster

% gamma_rise Ray decay factor of diffused first cluster (time constant, nsec)
% gamma_1 Ray decay factor of diffused first cluster (time constant, nsec)
% chi Diffuse weight of diffused first cluster
%───────────────────────────────
% m0 Mean of log-normal distributed nakagami-m factor
% Km Time dependence of m0
% sigma_m0 Standard deviation of log-normal distributed nakagami-m factor
% sigma_Km Time dependence of sigma_m0
% sfading_mode Flag for small-scale fading
% 0 -> All paths have same m-factor distribution

% 1 -> LOS first path has a deterministic large m-factor

% 2 -> LOS first path of each cluster has a deterministic

% large m-factor
% m0_sp Deterministic large m-factor
%───────────────────────────────
% std_shdw Standard deviation of log-normal shadowing of entire impulse response
%───────────────────────────────
% kappa Frequency dependency of the channel
%───────────────────────────────
% fc Center Frequency
% fs Frequency Range
%%
modified by I2R
if cm_num == 1, % Residential LOS
% MPC arrival
Lam = 0.047; Lmean = 3;
lambda_mode = 1;
lambda_1 = 1.54; lambda_2 = 0.15; beta = 0.095;
%MPC decay
Gam = 22.61; gamma_0 = 12.53; Kgamma = 0; sigma_cluster = 2.75;
nlos = 0;
gamma_rise = NaN; gamma_1 = NaN; chi = NaN; % dummy in this scenario
% Small-scale Fading
m0 = 0.67; Km = 0; sigma_m0 = 0.28; sigma_Km = 0;
sfading_mode = 0; m0_sp = NaN;
```

```
% Large-scale Fading – Shadowing
std_shdw = 2.22;
% Frequency Dependence
kappa = 1.12;
fc = 6; % GHz
fs = 8; % 2 - 10 GHz
elseif cm_num == 2, % Residential NLOS
% MPC arrival
Lam = 0.12; Lmean = 3.5;
lambda_mode = 1;
lambda_1 = 1.77; lambda_2 = 0.15; beta = 0.045;
%MPC decay
Gam = 26.27; gamma_0 = 17.5; Kgamma = 0; sigma_cluster = 2.93;
nlos = 1;
gamma_rise = NaN; gamma_1 = NaN; chi = NaN; % dummy in this scenario
% Small-scale Fading
m0 = 0.69; Km = 0; sigma_m0 = 0.32; sigma_Km = 0;
sfading_mode = 0; m0_sp = NaN;
% Large-scale Fading – Shadowing
std_shdw = 3.51;
% Frequency Dependence
kappa = 1.53;
fc = 6; % GHz
fs = 8; % 2 - 10 GHz
elseif cm_num == 3, % Office LOS
% MPC arrival
Lam = 0.016; Lmean = 5.4;
lambda_mode = 1;
lambda_1 = 0.19; lambda_2 = 2.97; beta = 0.0184;
%MPC decay
Gam = 14.6; gamma_0 = 6.4; Kgamma = 0; sigma_cluster = 3; % assumption
nlos = 0;
gamma_rise = NaN; gamma_1 = NaN; chi = NaN; % dummy in this scenario
% Small-scale Fading
m0 = 0.42; Km = 0; sigma_m0 = 0.31; sigma_Km = 0;
sfading_mode = 2; m0_sp = 3; % assumption
% Large-scale Fading – Shadowing
std_shdw = 0; %1.9;
% Frequency Dependence
kappa = 0.03;
fc = 6; % GHz
fs = 8; % 3 - 6 GHz
elseif cm_num == 4, % Office NLOS
% MPC arrival
Lam = 0.19; Lmean = 3.1;
lambda_mode = 1;
lambda_1 = 0.11; lambda_2 = 2.09; beta = 0.0096;
%MPC decay
Gam = 19.8; gamma_0 = 11.2; Kgamma = 0; sigma_cluster = 3; % assumption
nlos = 2;
```

gamma_rise = 15.21; gamma_1 = 11.84; chi = 0.78;
% Small-scale Fading
m0 = 0.5; Km = 0; sigma_m0 = 0.25; sigma_Km = 0;
sfading_mode = 0; m0_sp = NaN; % assumption
% Large-scale Fading – Shadowing
std_shdw = 3.9;
% Frequency Dependence
kappa =0.71;
fc = 6; % GHz
fs = 8; % 3 - 6 GHz
elseif cm_num == 5, % Outdoor LOS
% MPC arrival
Lam = 0.0448; Lmean = 13.6;
lambda_mode = 1;
lambda_1 = 0.13; lambda_2 = 2.41; beta = 0.0078;
%MPC decay
Gam = 31.7; gamma_0 = 3.7; Kgamma = 0; sigma_cluster = 3; % assumption
nlos = 0;
gamma_rise = NaN; gamma_1 = NaN; chi = NaN; % dummy in this scenario
% Small-scale Fading
m0 = 0.77; Km = 0; sigma_m0 = 0.78; sigma_Km = 0;
sfading_mode = 2; m0_sp = 3; % assumption
% Large-scale Fading – Shadowing
std_shdw = 0.83;
% Frequency Dependence
kappa = 0.12;
fc = 6; % GHz
fs = 8; % 3 - 6 GHz
elseif cm_num == 6, % Outdoor NLOS
% MPC arrival
Lam = 0.0243; Lmean = 10.5;
lambda_mode = 1;
lambda_1 = 0.15; lambda_2 = 1.13; beta = 0.062;
%MPC decay
Gam = 104.7; gamma_0 = 9.3; Kgamma = 0; sigma_cluster = 3; % assumption
nlos = 1;
gamma_rise = NaN; gamma_1 = NaN; chi = NaN; % dummy in this scenario
% Small-scale Fading
m0 = 0.56; Km = 0; sigma_m0 = 0.25; sigma_Km = 0;
sfading_mode = 0; m0_sp = NaN; % assumption
% Large-scale Fading – Shadowing
std_shdw = 2; % assumption
% Frequency Dependence
kappa = 0.13;
fc =6; % GHz
fs = 8; % 3 - 6 GHz
elseif cm_num == 7, % Industrial LOS
% MPC arrival
Lam = 0.0709; Lmean = 4.75;
lambda_mode = 2;

lambda_1 = 1; lambda_2 = 1; beta = 1; % dummy in this scenario
%MPC decay
Gam = 13.47; gamma_0 = 0.615; Kgamma = 0.926; sigma_cluster = 4.32;
nlos = 0;
gamma_rise = NaN; gamma_1 = NaN; chi = NaN; % dummy in this scenario
% Small-scale Fading
m0 = 0.36; Km = 0; sigma_m0 = 1.13; sigma_Km = 0;
sfading_mode = 1; m0_sp = 12.99;
% Large-scale Fading – Shadowing
std_shdw = 6;
% Frequency Dependence
kappa = -1.103;
fc = 6; % GHz
fs = 8; % 2 - 8 GHz
elseif cm_num == 8, % Industrial NLOS
% MPC arrival
Lam = 0.089; Lmean = 1;
lambda_mode = 2;
lambda_1 = 1; lambda_2 = 1; beta = 1; % dummy in this scenario
%MPC decay
Gam = 5.83; gamma_0 = 0.3; Kgamma = 0.44; sigma_cluster = 2.88;
nlos = 2;
gamma_rise = 47.23; gamma_1 = 84.15; chi = 0.99;
% Small-scale Fading
m0 = 0.3; Km = 0; sigma_m0 = 1.15; sigma_Km = 0;
sfading_mode = 0; m0_sp = NaN; % m0_sp is assumption
% Large-scale Fading – Shadowing
std_shdw = 6;
% Frequency Dependence
kappa = -1.427;
fc = 6; % GHz
fs = 8; % 2 - 8 GHz
elseif cm_num == 9, % Open Outdoor Environment NLOS (Fram, Snow-Covered Open Area)
% MPC arrival
Lam = 0.0305; Lmean = 3.31;
lambda_mode = 1;
lambda_1 = 0.0225; lambda_2 = 1; beta = 1;
%MPC decay
Gam = 56; gamma_0 = 0.92; Kgamma = 0; sigma_cluster = 3; % sigma_cluster is assumption
nlos = 1;
gamma_rise = NaN; gamma_1 = NaN; chi = NaN;
% Small-scale Fading
m0 = 4.1; Km = 0; sigma_m0 = 2.5; sigma_Km = 0;
sfading_mode = 0; m0_sp = NaN; % m0_sp is assumption
% Large-scale Fading – Shadowing
std_shdw = 3.96;
% Frequency Dependence
kappa = -1; % Kappa is assumption

```
fc = 6; % GHz
fs = 8; % 2 - 8 GHz
else
error('cm_num is wrong!!')
end
return
function [h]= uwb_sv_freq_depend_ct_15_4a(h,fc,fs,num_channels,kappa)
% This function is used to include the frequency dependency
f0 = 5; % GHz
h_len = length(h(:,1));
f = [fc-fs/2 : fs/h_len/2 : fc+fs/2]./f0;
f = f.^(-2*(kappa));
f = [f(h_len : 2*h_len), f(1 : h_len-1)]';
i = (-1)^(1/2); % complex i
for c = 1:num_channels
% add the frequency dependency
h2 = zeros(2*h_len, 1);
h2(1 : h_len) = h(:,c); % zero padding
fh2 = fft(h2);
fh2 = fh2 .* f;
h2 = ifft(fh2);
h(:,c) = h2(1:h_len);
% Normalize the channel energy to 1
h(:,c) = h(:,c)/sqrt(h(:,c)' * h(:,c) );
end
return
function [h,t,t0,np] =
uwb_sv_model_ct_15_4a(Lam,Lmean,lambda_mode,lambda_1, ...
lambda_2,beta,Gam,gamma_0,Kgamma,sigma_cluster,nlos,gamma_rise,gamma_1, ...
chi,m0,Km,sigma_m0,sigma_Km,sfading_mode,m0_sp,std_shdw,num_channels,ts)
% Written by Sun Xu, Kim Chee Wee, B. Kannan & Francois Chin on 22/02/2005
% IEEE 802.15.4a UWB channel model for PHY proposal evaluation
% continuous-time realization of modified S-V channel model
% Input parameters:
% detailed introduction of input parameters is at uwb_sv_params.m
% num_channels number of random realizations to generate
% Outputs
% h is returned as a matrix with num_channels columns, each column
% holding a random realization of the channel model (an impulse response)
% t is organized as h, but holds the time instances (in nsec) of the paths whose
% signed amplitudes are stored in h
% t0 is the arrival time of the first cluster for each realization
% np is the number of paths for each realization.
% Thus, the k'th realization of the channel impulse response is the sequence
% of (time,value) pairs given by (t(1:np(k),k), h(1:np(k),k))
%%
modified by I2R
% initialize and precompute some things
std_L = 1/sqrt(2*Lam); % std dev (nsec) of cluster arrival spacing
std_lam_1 = 1/sqrt(2*lambda_1);
```

```
std_lam_2 = 1/sqrt(2*lambda_2);
% std_lam = 1/sqrt(2*lambda); % std dev (nsec) of ray arrival spacing
h_len = 1000; % there must be a better estimate of # of paths than this
ngrow = 1000; % amount to grow data structure if more paths are needed
h = zeros(h_len,num_channels);
t = zeros(h_len,num_channels);
t0 = zeros(1,num_channels);
np = zeros(1,num_channels);
for k = 1:num_channels % loop over number of channels
tmp_h = zeros(size(h,1),1);
tmp_t = zeros(size(h,1),1);
if nlos == 1,
Tc = (std_L*randn)^2 + (std_L*randn)^2; % First cluster random arrival
else
Tc = 0; % First cluster arrival occurs at time 0
end
t0(k) = Tc;
if nlos == 2 & lambda_mode == 2
L = 1; % for industrial NLOS environment
else
L = max(1, poissrnd(Lmean)); % number of clusters
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
if Kgamma ~= 0 & nlos == 0
Tcval = []; Tc_cluster= [];
Tc_cluster(1,1)=Tc;
for i_Tc=2:L+1
Tc_cluster(1,i_Tc)= Tc_cluster(1,i_Tc-1)+(std_L*randn)^2 + (std_L*randn)^2;
end
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
cluster_index = zeros(1,L);
path_ix = 0;
nak_m = [];
for ncluster = 1:L
% Determine Ray arrivals for each cluster
Tr = 0; % first ray arrival defined to be time 0 relative to cluster
cluster_index(ncluster) = path_ix+1; % remember the cluster location
gamma = Kgamma*Tc + gamma_0; % delay dependent cluster decay time
if nlos == 2 & ncluster == 1
gamma = gamma_1;
end
Mcluster = sigma_cluster*randn;
Pcluster = 10*log10(exp(-1*Tc/Gam))+Mcluster; % total cluster power
Pcluster = 10^(Pcluster*0.1);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
if Kgamma ~= 0 & nlos == 0
Tr_len=Tc_cluster(1,ncluster+1)-Tc_cluster(1,ncluster);
```

```
else
Tr_len = 10*gamma;
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
while (Tr < Tr_len),
t_val = (Tc+Tr); % time of arrival of this ray
if nlos == 2 & ncluster == 1
% equation (22)
h_val = Pcluster*(1-chi*exp(-Tr/gamma_rise))*exp(-Tr/gamma_1) ...
*(gamma+gamma_rise)/gamma/(gamma+gamma_rise*(1-chi));
else
% equation (19)
h_val = Pcluster/gamma*exp(-Tr/gamma)/(beta*lambda_1+(1-beta)*lambda_2+1);
end
path_ix = path_ix + 1; % row index of this ray
if path_ix > h_len,
% grow the output structures to handle more paths as needed
tmp_h = [tmp_h; zeros(ngrow,1)];
tmp_t = [tmp_t; zeros(ngrow,1)];
h = [h; zeros(ngrow,num_channels)];
t = [t; zeros(ngrow,num_channels)];
h_len = h_len + ngrow;
end
tmp_h(path_ix) = h_val;
tmp_t(path_ix) = t_val;
% if lambda_mode == 0
% Tr = Tr + (std_lam*randn)^2 + (std_lam*randn)^2;
if lambda_mode == 1
if rand < beta
Tr = Tr + (std_lam_1*randn)^2 + (std_lam_1*randn)^2;
else
Tr = Tr + (std_lam_2*randn)^2 + (std_lam_2*randn)^2;
end
elseif lambda_mode == 2
Tr = Tr + ts;
else
error('lambda mode is wrong!')
end
% generate log-normal distributed nakagami m-factor
m_mu = m0 - Km*t_val;
m_std = sigma_m0 - sigma_Km*t_val;
nak_m = [nak_m, lognrnd(m_mu, m_std)];
end
Tc = Tc + (std_L*randn)^2 + (std_L*randn)^2;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
if Kgamma ~= 0 & nlos == 0
Tc = Tc_cluster(1,ncluster+1);
end
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%
end
% change m value of the first multipath to be the deterministic value
if sfading_mode == 1
nak_ms(cluster_index(1)) = m0_sp;
elseif sfading_mode == 2
nak_ms(cluster_index) = m0_sp;
end
% apply nakagami
for path = 1:path_ix
h_val = (gamrnd(nak_m(path), tmp_h(path)/nak_m(path))).^(1/2);
tmp_h(path) = h_val;
end
np(k) = path_ix; % number of rays (or paths) for this realization
[sort_tmp_t,sort_ix] = sort(tmp_t(1:np(k))); % sort in ascending time order
t(1:np(k),k) = sort_tmp_t;
h(1:np(k),k) = tmp_h(sort_ix(1:np(k)));
% now impose a log-normal shadowing on this realization
% fac = 10^(std_shdw*randn/20) / sqrt( h(1:np(k),k)' * h(1:np(k),k) );
% h(1:np(k),k) = h(1:np(k),k) * fac;
end
return
function [hN,N] = uwb_sv_cnvrt_ct_15_4a( h_ct, t, np, num_channels, ts )
% convert continuous-time channel model h_ct to N-times oversampled discrete-time
samples
% h_ct, t, np, and num_channels are as specified in uwb_sv_model
% ts is the desired time resolution
%%hN will be produced with time resolution ts /
N.
% It is up to the user to then apply any filtering and/or complex downconversion and
then
% decimate by N to finally obtain an impulse response at time resolution ts.
min_Nfs = 100; % GHz
N = max( 1, ceil(min_Nfs*ts) ); % N*fs = N/ts is the intermediate sampling frequency
before decimation
N = 2^nextpow2(N); % make N a power of 2 to facilitate efficient multi-stage
decimation
Nfs = N / ts;
t_max = max(t(:)); % maximum time value across all channels
h_len = 1 + floor(t_max * Nfs); % number of time samples at resolution ts / N
hN = zeros(h_len,num_channels);
for k = 1:num_channels
np_k = np(k); % number of paths in this channel
t_Nfs = 1 + floor(t(1:np_k,k) * Nfs); % vector of quantized time indices for this
channel
for n = 1:np_k
hN(t_Nfs(n),k) = hN(t_Nfs(n),k) + h_ct(n,k);
end
end
```

Sample code