D2.3 Communication Needs and Acoustic Processing Unit Requirements

Author: Sebastian Uziel (IMMS), Wolfram Kattanek (IMMS)
Contributor (s): Andrea Cavallaro (DAP), Marcella Scuccimarra (SEA)
Project Acronym: S4ECoB
Grant Agreement Number: 284628

Issue Date:	**M22**
Deliverable Number:	**D2.3**
Work Package Number:	**WP2**
Status:	**Final version**

DISSEMINATION LEVEL	
X	**PU = Public**
	PP = Restricted to other programme participants (including the EC)
	RE = Restricted to a group specified by the consortium (including the EC)
	CO = Confidential, only for members of the consortium (including the EC)

Document History			
Version	**Date**	**Author**	**Description**
1	30.09.2012	suziel	Draft version
2	12.02.2013	suziel	Preliminary version
3	15.07.2013	suziel, wkattanek	Final version submitted to QCC for review
4	24.07.2013	suziel, wkattanek	Final version after review

Disclaimer

The information proposed in this document is provided as a generically explanation on the proposed topic. No guarantee or warranty is given that the information fits for any particular purpose. The user thereof must assume the sole risk and liability of this report practical implementation. The document reflects only the author's views and the whole work is not liable for any empirical use of the information contained therein.

CONTENTS

TERMINOLOGY AND ABBREVIATIONS

BEMO	Building Energy Management System Optimizer = overall S4ECoB system
BMS	Building Management System
BEMO server	central component of each BEMO installation (usually 1 BEMO server per building / site); consists of several software components (e.g. data base, data fusioning module, GUI and BMS interface, gateway to occupancy sensor network, internet remote access) running on a dedicated PC / server or on PC / server which is already part of the BMS
APU	Acoustic Processing Unit
ASU	Audio Satellite Unit
PTP	Precision Time Protocol
GPMC	General Purpose Memory Controller
FPGA	Field Programmable Gate Array

LIST OF FIGURES

LIST OF TABLES

1 INTRODUCTION

1.1 Purpose of this Document

The S4ECoB system aims at monitoring and processing sounds and noises for an accurate determination of the types of occupancy and activities inside smart buildings in order to enhance the quality of the Building Energy Management (BEM) and in consequence to optimize the energy efficiency in buildings.

The S4ECoB system consists of two main components:

- Occupancy sensors with integrated Acoustic Processing Unit (APU) and up to 3 Audio Satellite Units (ASU) each connected with up to 8 microphones
- Building Energy Management Optimizer server (BEMO server)

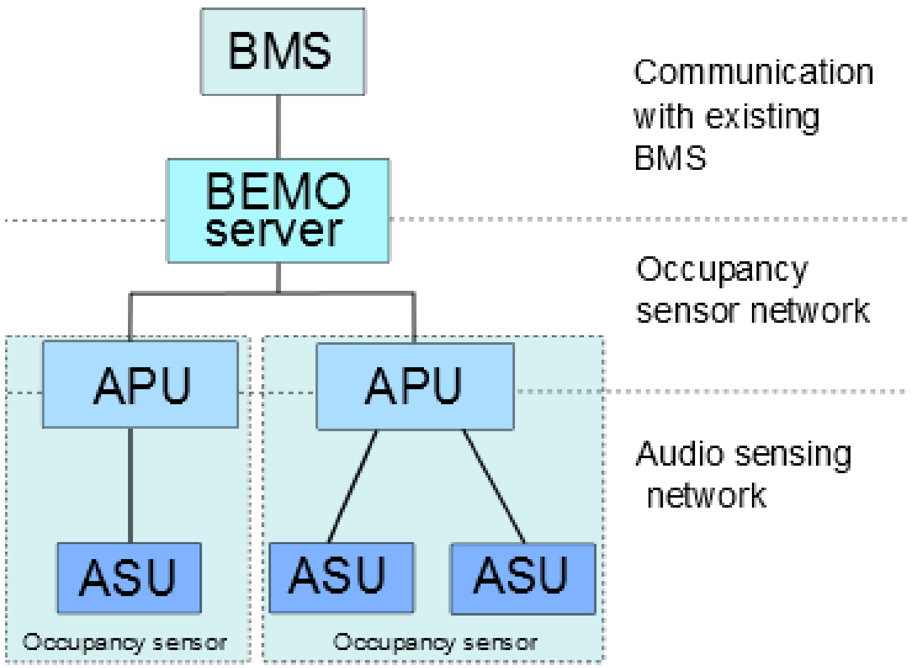

Figure 1: S4ECoB system overview

Between the components of the S4ECoB system, namely APU, ASU, BEMO server and between the BEMO server and the existing building management system, different types and amount of data have to be transferred. Therefore communication channels for the audio sensor network and occupancy network and the communication with the existing building management system have to be implemented. Each of these channels has different requirements, which have to be taken into account. In this deliverable the communication and sensing concept and the requirements for the different communication channels of the S4ECoB system will be defined, suitable solutions will be listed, compared and adopted solutions for each communication channel will be presented.

Based on the communication requirements presented in this deliverable and the acoustic processing and system requirements provided in the deliverables D2.2 and D2.4 the hardware and software requirements for the embedded acoustic processing unit and the APU hardware architecture will be defined in this document.

The requirements will be divided into three classes:

- Functional requirements (R_F), possibly complemented with required quantitative parameters, describing specific functionality of the APU and the sensor network communication
- Non-functional requirements (R_N), describing general properties and the operational behaviour of the APU and the sensor network
- Architectural and interface requirements (R_A), describing required architectural properties of the APU, its interfaces and the sensor network communication system in general

Finally the test methodology for the technical system validation will be presented.

1.2 Structure of the Deliverable

This document is structured as follows:

- Chapter 2 gives an overview about the S4ECoB system and the sensing concept.
- Chapter 3 presents the requirements for the individual communication channels and lists and compares suitable communication solutions and an adopted solution for each communication channel.
- In Chapter 4 the hardware and software requirements for the audio processing unit are provided.
- In Chapter 5 the requirements and planned tests for the validation of the communication and processing system are defined.
- Chapter 6 summarizes the key conclusions of the achieved results.

1.3 Relationship to the Project Objectives

The requirements, specifications and architecture defined in this document are essential for the development of the occupancy sensors (Objective 2), including the audio satellite and acoustic processing unit (Objective 1), the communication interfaces between these components and the BEMO server. Furthermore the validation methods and test cases specified in this document are fundamental for the proper work of the S4ECoB system in the demo sites.

1.4 Relationship to other Deliverables and Tasks

In D2.2 the requirements, architecture and specification of the overall S4ECoB system are documented, which is a basic input for this deliverable. Audio recording and processing parameters are defined in D2.4, on this basis the interfaces and protocols for audio transmission could be selected.

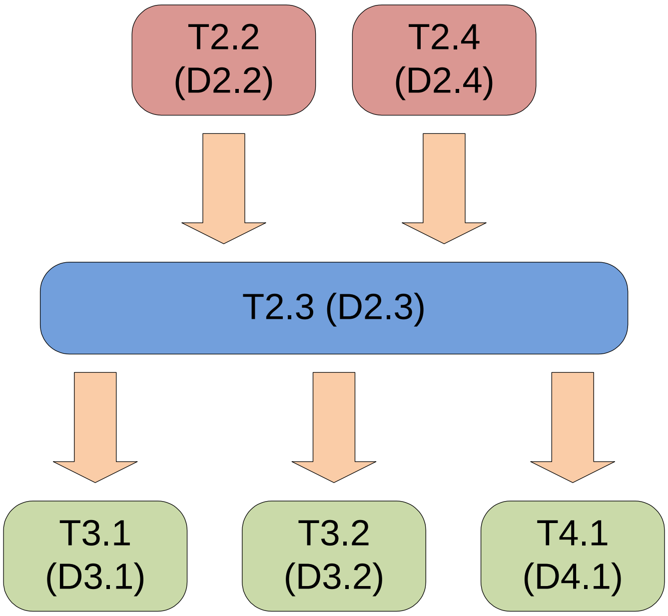

Figure 2: Relation of D2.3 to previous and successive deliverables and tasks

The information presented in this deliverable is an important input for T3.1, T3.2 and T4.1 in order to enable the development of the communication between the audio satellite unit, acoustic processing unit and the APU gateway running on the BEMO server.

1.5 Contributions of Partners

IMMS INSTITUT FUER MIKROELEKTRONIK- UND MECHATRONIK-SYSTEME GMBH has had the main responsibility to prepare this document.

DAP has contributed information regarding the communication with existing BEM systems, SEA has contributed general requirements from the end-user point of view and FHG has contributed specific communication and functional requirements regarding the audio processing.

2 COMMUNICATION AND SENSING CONCEPT

The S4ECoB system is aimed at monitoring and processing sounds and noises for an accurate determination of the types of occupancy and activities inside smart buildings in order to improve the Building Energy Management (BEM) systems and in consequence to optimize the Energy efficiency in Buildings (EeB). The S4ECoB platform will integrate other existing HVAC systems, such as heating, ventilation, air conditioning, etc. and will interoperate with other ICT-based subsystems.

A schematic view of the S4ECoB system is shown in Figure 3.

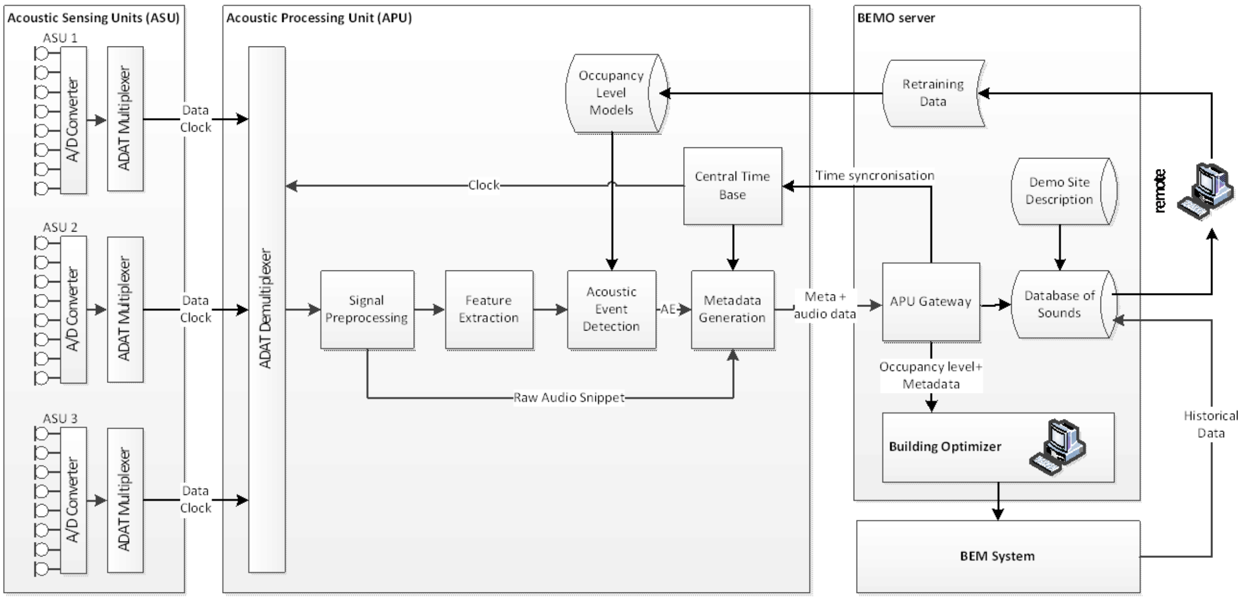

Figure 3: Schematic view of the S4ECoB system

The audio sensing is performed by a network of microphones. Up to 8 microphones are grouped into microphone arrays. The analogue microphone signals are amplified, converted to digital signals and encoded using the Audio Satellite Unit (ASU) integrated electronics.

Up to 3 audio satellite units can be connected to one Acoustic Processing Unit (APU), an energy-efficient but capable, adaptable and scalable embedded audio processing HW/SW platform. The APU extracts and processes the audio streams in near real time and detects events to discriminate the level of occupancy in the corresponding room or area.

The calculated occupancy level will be sent to the BEMO server, a central component of each installation, consisting of several software components (e.g. data base, data fusioning module, GUI and BMS interface, gateway to occupancy sensor network, internet remote access), running on a dedicated PC and receiving all information from the occupancy sensors. Its main purpose is the selection and/or calculation of optimized building automation parameters (based on the actual occupancy level) in order to control existing building automation systems in a way suitable for lowering the building energy consumption while preserving functional (e.g. comfort) and safety requirements.

For the overall S4ECoB system three communication channels have to be implemented:

- Audio sensor network communication: a real time audio data stream from audio satellite units to the APU
- Occupancy sensor network communication: a protocol to integrate the occupancy sensors (i.e. APUs) into the building communication system
- Building automation communication: communication with existing Building Management Systems (BMS) and/or Building Energy Management Systems (BEM)

3 COMMUNICATION CHANNELS

3.1 Audio Sensor Network Communication Requirements

The physical connection between the APU and one Audio Satellite Unit (ASU) consists of a communication part transmitting 8 channel audio data from ASUs to APU, a word clock (sync signal) from APU to ASUs. Optionally a power supply for the ASUs should be foreseen in the APU.

Audio data from the audio satellite unit has to be transferred to the APU in real time. A communication protocol and electrical interface standard has to be selected.

As defined in deliverable D2.4 there are at maximum 8 microphones connected to one satellite unit. Using a sample rate of 48 kHz and a resolution of 24 bits, a maximum net data rate of

$$data\ rate = 48\ kHz\ \times\ 24\ bits\ \times\ 8\ channels = 9.216\ {}^{Mbit}/_{s}$$

raw audio data stream have to be transferred from one ASU to the APU.

As described before up to three ASUs could be connected to one APU. The ASUs are spatially distributed inside the demo sites. The selected communication protocol and the corresponding electrical interface standard should be able to transfer data at the specified data rate over a distance of minimum 40m (in order to also cover large rooms and areas with only one APU and up to three connected ASUs.

The S4ECoB system will be installed in an electrically noisy environment. In the demo sites wireless LANs and other radio communication equipment are used, the audio sensor network communication should be immune to this impact.

The system should be easy to install, if possible only one standard cable should be used for the communication between the satellite units and the APU. This minimizes the possibility of a failure during the installation process.

In summary the requirements in Table 1 have to be fulfilled by the audio sensor network in the S4ECoB system.

Requirement	Description
R_F1.1	Adequate data rate of about 10 Mbit/s
R_F1.2	Cable length >= 40m
R_A1.3	Ensure proper signal integrity
R_A1.4	Easy to use / install in the demo sites

Table 1: Requirements for the audio sensor network communication

3.1.1 Available Audio Protocols and Standards

There are numerous digital audio interfaces and communication standards to stream audio data in real time. Following some common and frequently used standards are outlined.

- AES3 (AES/EBU) / S/PDIF (Sony/Philips Digital Interconnect Format)

  AES3 was designed to transmit 2 audio channels over balanced, unbalanced lines or optical fiber at various sampling frequencies and a resolution from 16 to 24 bit[1]. S/PDIF is a consumer variant of AES3.

- I^2S (Inter-IC Sound)

  The I^2S serial bus is commonly used to transfer audio signal in a device between different integrated circuits. The I^2S bus uses separate clock and data lines[2].

- MADI (Multichannel Audio Digital Interface)

  MADI supports the serial digital transmission over fiber or coax cable of 24, 56 or 64 channels of digital data at a common sampling frequency within the range from 32 kHz to 96 kHz having a resolution of up to 24 bit per channel. The audio engineering society (AES) documented the MADI standard as AES10-2003[3].

- ADAT (Alesis Digital Audio Tape)

  The ADAT optical or ADAT Lightpipe interface can carry 8 digital audio channels with 24-bit resolution and 44.1 or 48 kHz sample rate[4].

In Table 2 the features of the different protocols are summarized.

Protocol	**AES3/SPDIF**	**MADI**	**ADAT**	**I^2S**
Channels	2	24 to 64	8	2
Coding	Biphase mark	RLL	NRZI	None
Sample Frequencies	32 – 96 kHz	32 – 48 kHz	44.1, 48 kHz	Not specified
Data Rate	6.144 Mbit/s	125 Mbit/s	12 Mbit/s	Not specified

Table 2: Audio protocol summary

[1] http://tech.ebu.ch/docs/tech/tech3250.pdf

[2] https://www.sparkfun.com/datasheets/BreakoutBoards/I2SBUS.pdf

[3] http://www.iis.ee.ethz.ch/%7Efelber/DataSheets/AES-EBU/aes10-2003.pdf

[4] http://www.albert-av.de/htm/adat/interface.htm

3.1.2 Selected Solution

Using AES3, SPDIF or I2S protocol only 2 audio channels can be transmitted simultaneously, so these protocols are not qualified for the S4ECoB system.

The MADI protocol was defined for the transmission of a large number of audio channels. This protocol is oversized for the use in the S4ECoB system and its implementation is too costly. The MADI protocol uses a 10 times higher data rate than the ADAT protocol. The higher the data rate of a used transmission protocol the more effort is necessary to ensure electromagnetic compatibility and immunity against impact from external disturbing sources, like WLAN and cell phones.

ADAT was chosen as interface protocol, because it uses a simple NRZI code, receiver and transceiver ICs are available and are easy to implement. ADAT is protected by a European patent (EP0621994B1), which expired in the EU in January 2005.

Instead of the ADAT specified transmission over optical fiber the electrical RS485 standard was chosen, because it can be used over long distances and in environments with a high electrical noise level. The RS485 interface standard supports data rates up to 12 Mbit/s, so it is sufficient for the use with the ADAT protocol.

The ADAT protocol uses NRZI (no return to zero) coding. A digital “1” is coded by a level change (high-to-low or low-to-high). A digital “0” doesn’t change the signal. For clock recovery “1”s are periodically inserted. One ADAT data frame consists of 256 bits. It starts with 5 “0”s as initial sequence, then a “1” (as clock pulse) is followed by 4 user bits. The following audio data is split into 4 bit words which are inserted between clock pulses. Table 3 shows the ADAT protocol structure.

Bit	Function	Data
0 to 4	Frame start	Digital zero
5	clock	Digital one
6	User bit 3	Reserved zero
7	User bit 2	
8	User bit 1	
9	User bit 0	
10	Clock	Digital one
11 to 14	Channel 1 bit 23 to 20	Audio
15	Clock	Digital one
16 to 19	Channel 1 bit 19 to 16	Audio
20	Clock	Digital one
21 to 24	Channel 1 bit 15 to 12	Audio
25	Clock	Digital one
26 to 29	Channel 1 bit 11 to 8	Audio
30	Clock	Digital one
31 to 34	Channel 1 bit 7 to 4	Audio
35	Clock	Digtial one

36 to 39	Channel 1 bit 3 to 0	Audio
40 to 69	Channel 2	Clock and audio
70 to 99	Channel 3	Clock and audio
100 to 129	Channel 4	Clock and audio
130 to 159	Channel 5	Clock and audio
160 to 189	Channel 6	Clock and audio
190 to 219	Channel 7	Clock and audio
220 to 240	Channel 8	Clock and audio
250	Clock	Digital one
251 - 255	End frame	Digital zero

Table 3: ADAT protocol

3.2 Occupancy Sensor Network Communication Requirements

In the occupancy sensor network the APU communicates with the APU gateway running on the BEMO server. The calculated occupancy levels, status information and retraining data have to be transmitted from the APUs to the BEMO server. Furthermore remote access to the APUs and a possibility for parameter and firmware updates of the APUs is necessary.

Unknown or unrecognized sounds (short audio samples) have to be transferred from the APU to a database running on the BEMO server for training purposes. It should be possible to use encrypted communication between the APU and the BEMO server, to deny access for unauthorized persons.

The calculated current occupancy level and status information have to be transferred immediately from APU to the BEMO server. If unknown sounds are detected, short audio samples have to be transmitted to the BEMO server in non real-time. The audio samples are stored in a database running on the BEMO server, therefore the data could transfer in idle periods of the APU. Consequently this communication interface should support different data types and different transfer mechanisms.

In the development and validation phase algorithms and firmware on the APU have to be regularly updated remotely. The retraining process is also necessary after the validation phase to incorporate formerly unknown sounds into the models of the acoustic event detector and improve the performance of these algorithms. Therefore a stable process for remote access and parameter update of the acoustic event detection should be implemented in the fully working system.

The status of the connected APUs in the sensor network has to be monitored, therefore a cyclic status check of the connected APUs has to be implemented. If an APU did not respond in a defined time interval a warning message should be generated, to inform maintenance personnel.

For optional distributed audio sensing and processing the occupancy sensor network communication solution should be able to support time synchronization between APUs (see also chapter 4.2).

Installation effort should be low, existing wiring installations and standard technologies should be used as much as possible.

In Table 4 the requirements for the occupancy sensor network are summarized.

Requirement	Description
$R_A2.1$	Respect of ethical issues – secure data transmission
$R_A2.2$	Support for different data types
$R_F2.3$	Remote access to the occupancy sensor / APU
$R_F2.4$	Cyclic alive check of occupancy sensor / APU
$R_F2.5$	Time synchronization of all components in the occupancy sensor network

R_N2.6	Low installation effort

Table 4: Requirements for the occupancy sensor network

3.2.1 Available Protocols

There are numerous communication protocols and standards used in the building and home automation sector. Following the three widespread used building automation protocols KNX, LonWorks and BACnet are listed and explained.

- EIB/KNX

  KNX is a network communication protocol for intelligent buildings. It defines several physical communication media, such as twisted pair wiring, power line networking and wireless[5].

  The EIB/KNX technology is well adjusted to the electrical installation industry and to the smaller building area. The geographical bulk of the EIB/KNX products are in Europe.

- LonWorks

  The LonWorks (local operation network) technology enables a peer-to-peer data exchange at the field level, creating a flat network architecture with the potential for multi-vendor interoperable networks. The platform is built on a protocol created by Echelon Corporation for networking devices over media such as twisted pair, powerline, fiber optics, and RF. It is used for the automation of various functions within buildings such as lighting and HVAC[6].

  Products incorporating LonWorks technology are suitable for both larger and small buildings enabling the use of a single technology for all control applications in both commercial buildings and in homes.

- BACnet

  BACnet is a communications protocol for Building Automation and Control Networks. BACnet was designed to allow communication of building automation and control systems for applications such as heating, ventilating, and air-conditioning control, lighting control, access control, and fire detection systems and their associated equipment[7].

  The BACnet solutions for network management rely on a network management tool that is specific to the vendor supplying the solution. Each vendor supplies a custom network operating system and network database (which may require multiple tools to implement).

  BACnet solutions are available from several manufacturers, but the domain of component products available from independent distributors is quite small and restricted. The BACnet

[5] http://www.knx.org/knx-standard/main-advantages

[6] http://www.lonmark.org/technical_resources/guidelines/master_system_specs

[7] BACnet Gebäudeautomation, Hans W. Kranz, ISBN 3-922420-02-8

community has been slower to adopt new technologies, resulting in longer time to market for new products.

In contrast to the above mentioned domain-specific protocols generic computer communication protocols could be taken into account for the occupancy sensor network communication. As the by far most prominent and in general best supported protocol TCP/IP is a natural choice here. Due to its proven stability and flexibility and furthermore due to its excellent support from operating systems and embedded system platforms the use of TCP/IP can greatly lower implementation risks. As an open and vendor independent standard it is future proof and is free of licensing costs.

In Table 5 the S4ECoB relevant properties of these protocols are summarized.

3.3 Protocol	KNX	LonWorks	BACnet	TCP/IP
Application area	building automation	building automation industrial automation	building automation	standard communication protocol
Use of existing infrastructure	partial	partial	partial	yes
Time synchronization available	no	no	no	yes
APU remote access possible	no	no	no	yes
secure data transmission	no	no data encryption, sender authentification implemented	yes	yes

Table 5: Summary of available protocols for the occupancy sensor network

3.3.1 Selected Solution

As presented in deliverable D2.2 in the three demo sites different building management systems are employed using different building automations protocols. Furthermore the building automation protocols KNX, LonWorks and BACnet meet not all the requirements for the S4ECoB system (see Table 5). The remote access to the APUs cannot be implemented using KNX, LonWorks or BACnet.

A standard Ethernet connection using the TCP/IP protocol was chosen for the occupancy sensor network communication. This allows the use of available wiring infrastructures already existing within buildings (demo sites). It is more robust and resistant against transmission errors than WLAN or other wireless standards. Additionally, a higher data transfer bandwidth can be achieved with Ethernet connections. Proven TCP/IP extensions and standards building upon it can be easily used such as the Secure Sockets Layer (SSL) protocol for example, which supports the establishment of secure data connections. Remote access to the APUs via secure shell protocol is possible.

3.4 Building Automation Communication Requirements

The BEMO server communicates with the existing building management systems of the demo sites to include available data from sensors already installed in the buildings, such as energy meters, temperature sensors and person counting systems.

In the deliverable D2.2 corresponding system requirements, standards and architectures are described. In chapter 3.2 of D2.2 the communication with the existing building management system is described in detail.

4 ACOUSTIC PROCESSING UNIT

4.1 APU Hardware Requirements

The acoustic processing unit receives the raw audio data stream coming from the satellite units and extracts and processes the stream in near real time. An audio event detection algorithm running on the APU discriminates the level of occupancy in a room or area. The calculated occupancy level has to be transferred to the BEMO server.

The occupancy level should be calculated in near real-time in the APU. A hardware platform with sufficient processing power has to be selected to achieve this requirement.

Main objective of the S4ECoB system is to optimize the energy efficiency in buildings, therefore the installed components have to consume power as low as possible, so this fact should be considered by the selection of the hardware platform.

The APU has to provide the following interfaces:

- ADAT

  As specified in chapter 3.1.2 the ADAT protocol and RS485 standard is used to transfer the raw audio data from the ASUs to the APU. RJ45 connectors are used for this interface. Three ASUs can be connected to one APU, therefore three audio interfaces have to be implemented.

- Ethernet

  A standard 10/100 Ethernet interface with RJ45 connector should be used for the occupancy sensor network communication.

- Power supply

  A standard external wall power supply with 2.5 mm DC plug connector should be used.

Requirement	**Description**
R_F3.1	Sufficient processing performance for audio processing and event detection algorithms
R_N3.2	Low energy consumption
R_A3.3	APU interfaces (3 ADAT, 1 Ethernet, 1 power suppy)

Table 6: Hardware requirements

4.1.1 CPU Performance Tests

To select a suitable embedded CPU for the acoustic processing unit, a benchmarking of different embedded platforms was performed. Following the benchmarked hardware platforms are described:

- Armadeus APF51[8], a reduced size processor board benefiting from a highly competitive price/performance ratio additionally equipped with a Spartan 6 FPGA.
  - Freescale 800 MHz i.MX515 processor: ARM Cortex A8 with 32KB I and D cache, 256KB L2 and FPU (floating point co-processor)
  - Xilinx Spartan 6A XC6SLX9
  - 256 MB 32 bits LPDDR
  - 512 MB SLC NAND

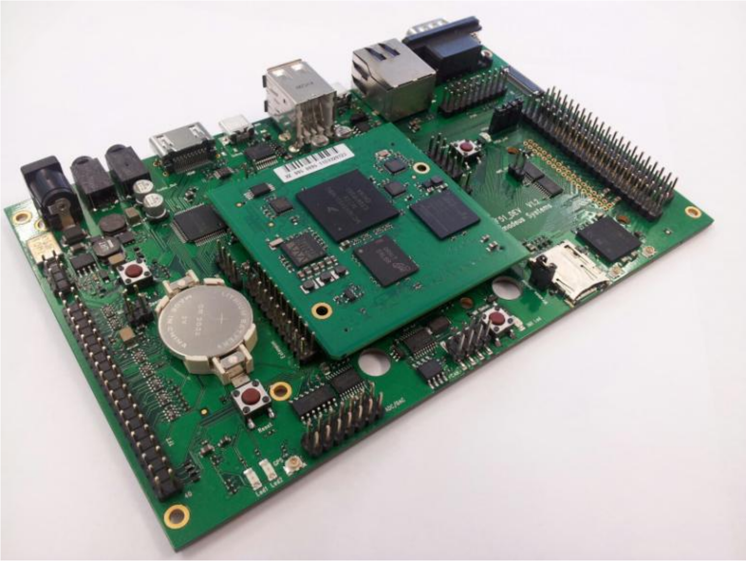

Figure 4: Armadeus APF51

- BeagleBone[9], a low cost, credit card sized evaluation board.
  - AM335x 720MHz ARM Cortex-A8
  - 256MB DDR2 400MHZ

[8] http://www.armadeus.com/english/products-processor_boards-apf51.html

[9] http://beagleboard.org/Products/BeagleBone

Figure 5: BeagleBone

- phyCORE®-OMAP4430[10], a System on Module for deployment in wireless, industrial, video/imaging, medical and other vertical embedded applications.
    - OMAP4430
    - 1 GB LPDDR2
    - 1 GB NAND
    - PowerVR SGX540 3D graphics

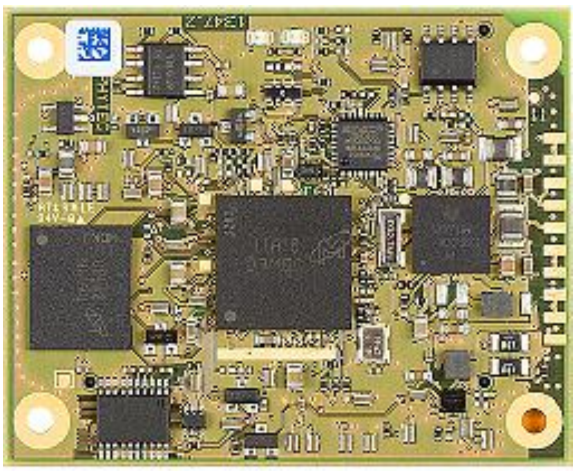

Figure 6: phyCore–OMAP4433

10 http://www.phytec.com/products/system-on-modules/phycore/omap4430

- PandaBoard ES[11], an open OMAP™ 4 mobile software development platform.
  - OMAP4460
  - 1 GB LPDDR2
  - PowerVR SGX540 3D graphics

Figure 7: Pandaboard ES

- Zotac ID41[12], an Intel Atom based Mini-PC.
  - Intel® Atom™ D525
  - 4 GB DDR3
  - Next-generation NVIDIA ION

[11] http://pandaboard.org/content/pandaboard-es

[12] http://www.zotac.com/products/mini-pcs/zbox/intel/product/intel/detail/zbox-id41.html

Figure 8: Zotac ID41

- Desktop PC, as a reference system.

In Table 7 key parameters of the analysed embedded boards are summarized.

Board	CPU	Core	#Cores	Core Clock (MHz)
APF51	i.MX51	Cortex-A8	1	800
BeagleBone	AM3359	Cortex-A8	1	720
phyCore OMAP4	OMAP4430	Cortex-A9	2	1008
PandaBoard ES	OMAP4460	Cortex-A9	2	1200
(Zotac ID41)	D525	Atom	2	1800
(PC)	Q6600	Core 2	4	2400

Table 7: Evaluated systems for CPU performance tests

For choosing a suitable APU system architecture three different benchmarks were used. As it is not known at the time of the benchmarking if the final audio processing can be distributed to different CPU cores only single core performance was measured in all benchmarks.

4.1.1.1 Generic Performance (nbench)

To estimate the overall performance of the selected embedded systems the generic benchmark *nbench*[13] was chosen. *nbench* is designed to expose the capabilities of a system's CPU, FPU, and memory system. It provides a summarized result for the fix point (Int-Idx), floating point (FP-Idx) and memory bandwidth (Mem-Idx) performance of the system under test.

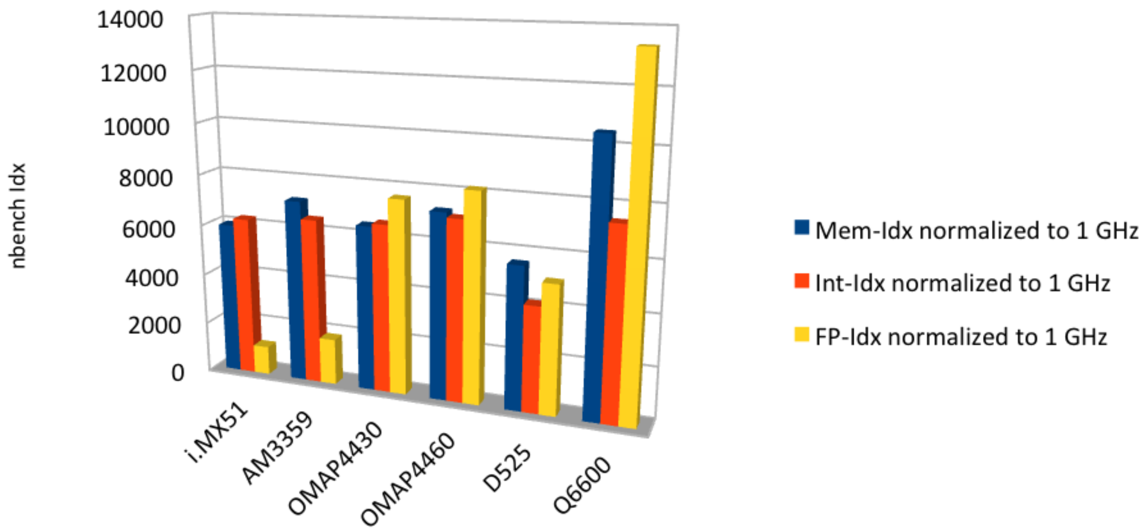

Figure 9: Generic processing performance with *nbench*

The *nbench* benchmark was compiled using different toolchains and various compiler optimization flags for each of the embedded boards. The best result for each system was logged and is shown in Figure 9. To allow a better comparison of the relative processing performance the index numbers in the chart are normalized to a CPU clock of 1 GHz. Considering the power consumption of the corresponding boards and systems would lead to significant advantages of the ARM-based embedded platforms (e.g. the PC with the Q6600 has power requirements of about 120 W whereas the Pandaboard with the OMAP4430 processor only consumes about 4 W).

4.1.1.2. Acoustic Localization Algorithm

To allow a more realistic estimation of the required APU performance a test algorithm for acoustic localization (provided by project partner FHG) was used. Using pre-recorded 6 channel 48 kHz audio data as input the position of a sound source was calculated. In order to get a meaningful parameter for performance comparison purposes of the different embedded platforms the runtime of the algorithm

[13] http://www.tux.org/~mayer/linux/bmark.html

analysing a 28 second piece of audio data was measured. In Figure 10 the analysed sample time divided by the runtime of the algorithm is shown. Hence a value greater "1" means that on this system the algorithm is able to perform the localization at least in real time.

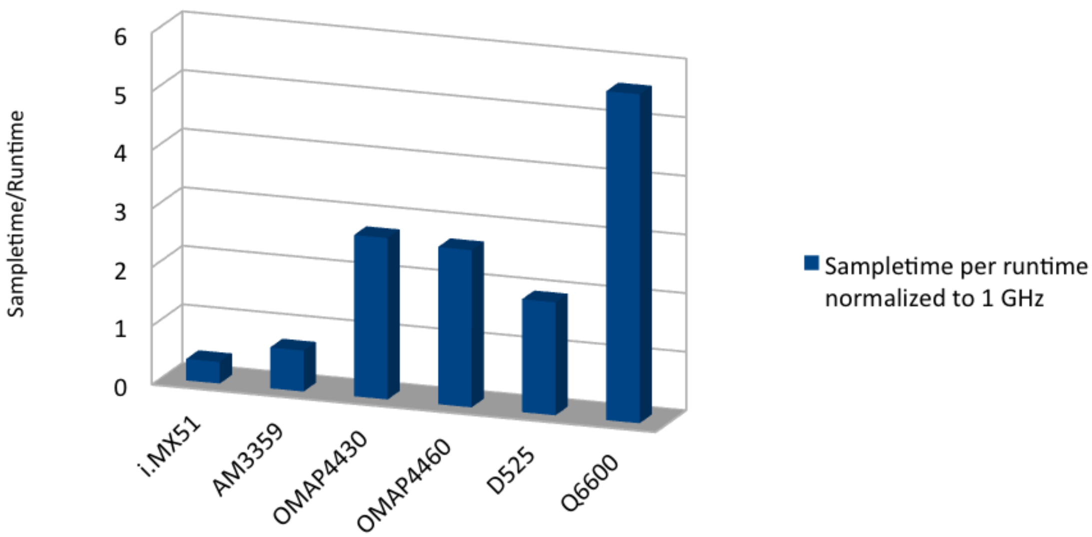

Figure 10: Performance of audio localization algorithm processing

4.1.1.2 IIR Filter and Level Normalization

As a third test an algorithm calculating an IIR filter after normalizing the input signal was used. This benchmark also uses a pre-recorded audio signal as input to make the results on the different platforms comparable. In Figure 11 the number of algorithm runs per second is shown.

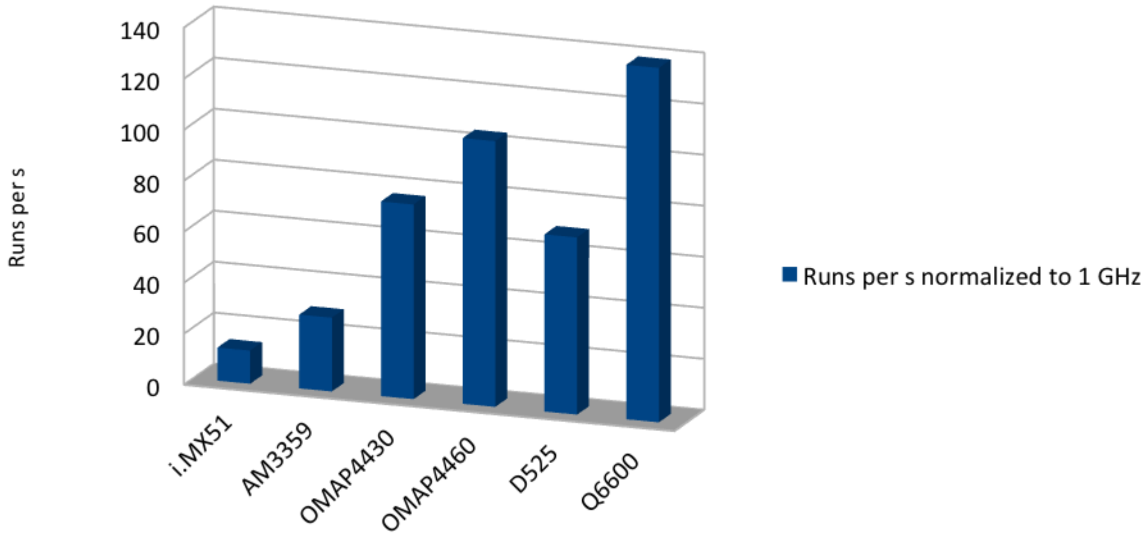

Figure 11: Performance of IIR filter and level normalization processing

4.1.1.3 Results and Conclusion

The generic benchmark indicates an approximately equal fixed point performance of the tested embedded systems whereas floating point operations of Cortex-A9 based CPUs show a significant performance gain over the Cortex-A8 architecture. Using tests 2 and 3 as a reference this difference becomes even more prominent, as the audio processing algorithms used herein heavily rely on floating point math, too. The reason for this performance gain is the greatly improved floating-point unit of the Cortex-A9 CPUs (more registers, better cycles per instruction value).

With the significant performance advantage in floating point calculations in mind the Pandaboard (OMAP4460 with Cortex-A9 architecture) was chosen as a base platform for the APU prototype. In addition, the OMAP4460 implements a second CPU core so that in theory the achievable performance could be doubled.

4.1.2 APU Hardware Architecture

Based on the defined APU hardware requirements and the performance analyses presented in the previous section the Pandaboard with the Texas Instruments OMAP4460 is used as processing platform. The board provides an onboard 10/100Mb Ethernet interface which can be used for the occupancy sensor network interface. The OMAP 4460 processor was designed for use in modern mobile communication devices, such as cell phones and tablets, therefore the Pandaboard is an energy efficient processing system, that offers sufficient performance for audio algorithm performance at a low energy consumption.

There is no interface for receiving 24 audio channels, respective 3 ADAT channels available on the Pandaboard. Therefore for the audio sensor network interface an expansion board has to be developed and connected via the expansions connectors to the Pandaboard.

ADAT receiver ICs are available from company Wavefront (AL1402G). Instead of using 3 of these ADAT receiver chips a field programmable gate array (FPGA) will be implemented. This programmable hardware circuit will be used to implement the audio protocol (ADAT) decoding and additionally provides implementation capabilities for various pre-processing functionalities (e.g. IIR filters). This leads to an optimized and more flexible system architecture. In Figure 12 a schematic view of the APU hardware is shown.

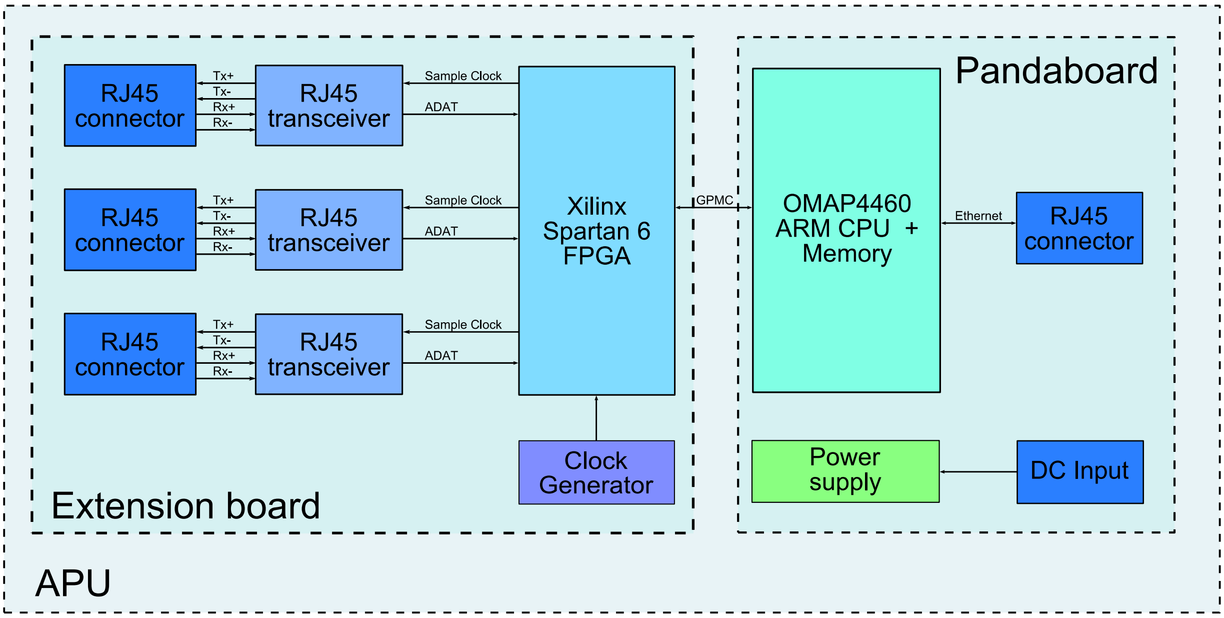

Figure 12: Schematic view of the APU hardware architecture

The FPGA will be connected to the General Purpose Memory Controller (GPMC) of the OMAP processor. The 16 bit synchronous address / data – multiplexed mode of the GPMC will be used. It features synchronous read and write as well as burst read access to the FPGA. The audio data has a resolution of 24 bit. The GPMC interface supports data transfers which are a multiple of 16 bit, therefore one audio sample has to be splitted in two 16 bit values. From FPGA to the OMAP processor 24 audio data streams with a resulting data rate of

$$data\ rate = 48\ kHz\ \times\ 32\ bits\ \times\ 24\ channels = 36.864\ {}^{Mbit}/_{s}$$

have to be transferred. The GPMC interface is capable of a data rate up to 160 MB/s, which is sufficient for the APU.

The FPGA firmware should be loadable via the GPMC interface from the processor.

An external clock generator circuit provides a low jitter sample clock which has to be distributed to the audio satellite units.

Power should be delivered with an external wall plug supply via a 2.5 mm DC connector.

In Table 8 the voltage and current requirements for the different APU components are listed.

Name	Voltage	Current	Usage / Description
VCC_48V	48 V	3 x 200 mA	Voltage supply for audio satellite units
VCC_5V	5 V	600 mA	Voltage supply for Pandaboard
VCC_3V3	3.3 V	800 mA	Supply for FPGA IO, RS485 ICs, locking circuits
VCCAUX	2.5 V	300 mA	FPGA auxiliary supply voltage
VCC_1V2	1.2 V	800 mA	FPGA core supply voltage

Table 8: APU voltage supply requirements

4.2 Software Requirements

In order to realize low-cost, vendor-independent, open and secure firmware and system software for the APU open-source software should be preferably used in general. Availability of source code ensures adaptability and therefore optimal support for S4ECoB specific hardware modules like APU FPGA board. An operating system well suited for the embedded hardware platform of the APU has to be chosen. Availability of low-cost and productive development and supporting tools has to be taken in consideration, too.

The use of sounds and audio data often includes the recording and processing of multiple audio channels. These can be locally connected to one APU or distributed between several APUs in the same room or area. Recorded audio data has to be synchronized (or at least time stamped) with high precision to be able to correlate signals from different channels. This is especially important for algorithms calculating signal runtime differences. In contrast to a hardware based time synchronization between multiple channels on a single APU in the case of physically distributed APUs a means of software based clock synchronization scheme is required.

Remote access to the occupancy sensor network (including the BEMO server and each individual APU) is required for several reasons. After the installation of the occupancy sensor network in the demo sites every retraining cycle of the network requires an update of the audio processing or occupancy level algorithms or the corresponding parameters sets or classification information. Another use of the remote access feature is the monitoring of each hardware and software component for debugging and performance optimization reasons. Furthermore updating software components on the APUs or the BEMO server up to a complete remote update of the APU firmware including operating system kernel, device drivers and FPGA configuration should be possible. This ensures maximum flexibility for example in the distribution of different parts of the occupancy level detection algorithms between APU FPGA, APU CPU and BEMO server, if it should be necessary to adapt the sensor networks setup regarding this aspect after deployment in the demo sites.

Several scenarios have to be considered regarding the network security of an S4ECoB installation to prevent unauthorized or unintended access. The occupancy sensor network will be providing input data for the building management system. Manipulation of this input data has to be avoided or to be made as hard as possible be using network security means, as forged input data could lead to malfunctions in the building management systems. This could lead to the wasting of power because wrong parameters for building management system are calculated in the best and even possible damage to physical equipment in the worst case. On the other hand access to the sensor network (to the APUs in particular) would make it possible to monitor or record live audio data captured by the audio satellite units. For privacy reasons this has obviously to be avoided. Last but not least the occupancy sensor network or components of the S4ECoB (software) installation (for example the remote access feature, see 4.2.3) could be used as a gateway to access other parts of the IT infrastructure in the demo sites.

In summary the following APU software requirements have to be fulfilled by the S4ECoB system (Table 9):

Requirement	Description
$R_A4.1$	Use of open-source software for APU wherever possible
$R_F4.2$	Clock synchronization between APUs with a clock deviation < 1 ms
$R_F4.3$	Secure remote APU access for audio algorithm and APU firmware update using Internet and demo site specific networks (including firewalls and VPNs)
$R_A4.4$	Secure network communication and APU access restrictions

Table 9: Software requirements

4.2.1 Operating System / Device Drivers

Considering the above mentioned requirements Linux was chosen as operating system for the APU. This decision was made for various reasons. Besides its open-source nature, its vendor-independence and the absence of any licensing costs Linux offers an excellent hardware support for embedded systems that were taken into account as possible base platforms for the APU. Many standard interfaces of the APU (e.g. SD/MMC or network interfaces) are already very well supported by the Linux kernel.

Apart from hardware support a Linux environment offers a wide range of ready to use software components. All kinds of libraries, protocol stacks, utility programs, etc. are available and could greatly speed up the software development.

In its nature as open source software the Linux kernel and almost all of the other software components of a Linux system are very flexible. An active community offers quick help regarding questions and problems.

One special device driver has to be developed for the APU. It uses the general purpose memory controller (GPMC) of the OMAP4460 CPU to communicate with the FPGA. The DMA controller of the CPU is used to transfer audio data from the FPGA into the APUs DDR2 memory. In addition, the driver allows exchanging control and status information over a register based interface with the FPGA.

Communication with a user space program takes places over the Linux character device interface using IOCTLs and memory mapping. A typical communication sequence between a user space program, the GPMC device driver and the FPGA is shown in Figure 13.

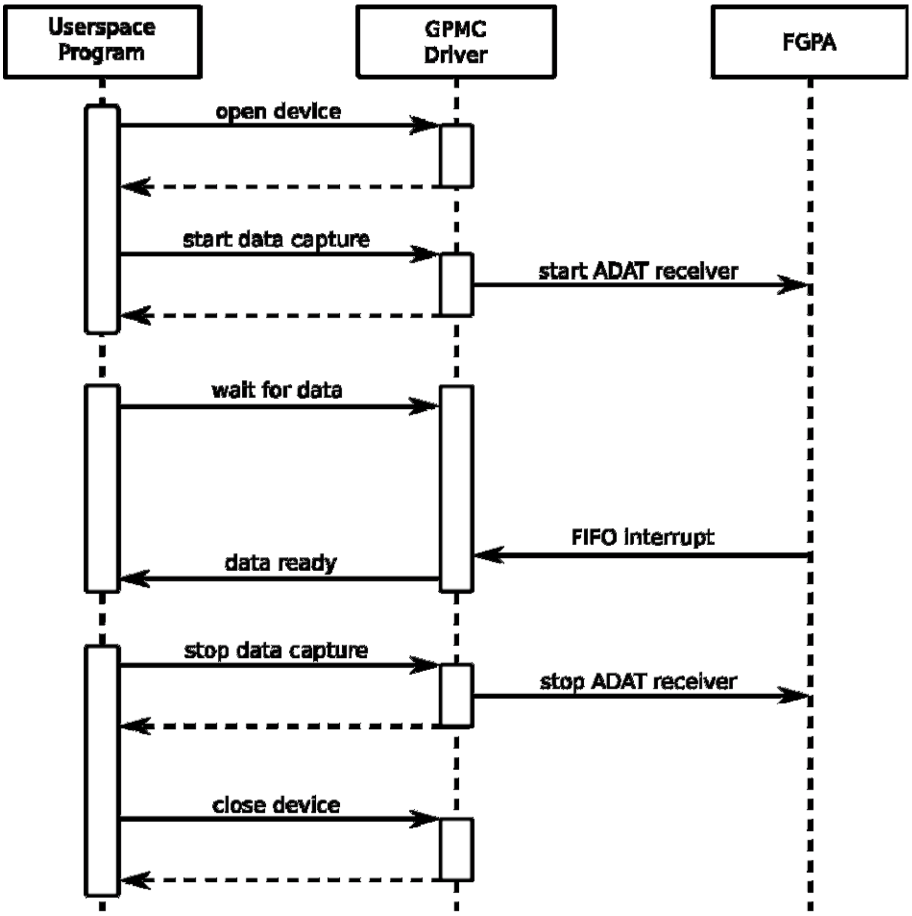

Figure 13: GPMC driver communication sequence

4.2.2 Clock Synchronization

The clock synchronization between the BEMO server and all connected APUs can be achieved through different means. One way would be the use of individual GPS based clocks per APU and server. Hence a very small clock deviation could be reached. But in addition to the resulting extra costs it can't be guaranteed for indoor applications to receive the GPS signal.

As all components of the occupancy detection system are connected, clock synchronization via data network can be used. There are several protocols available. With the most commonly used, the network time protocol (NTP), clock synchronization in a millisecond range can be achieved. Better results may be reached by using the precision time protocol (PTP, IEEE 1588), which promises a clock synchronization in microsecond range.

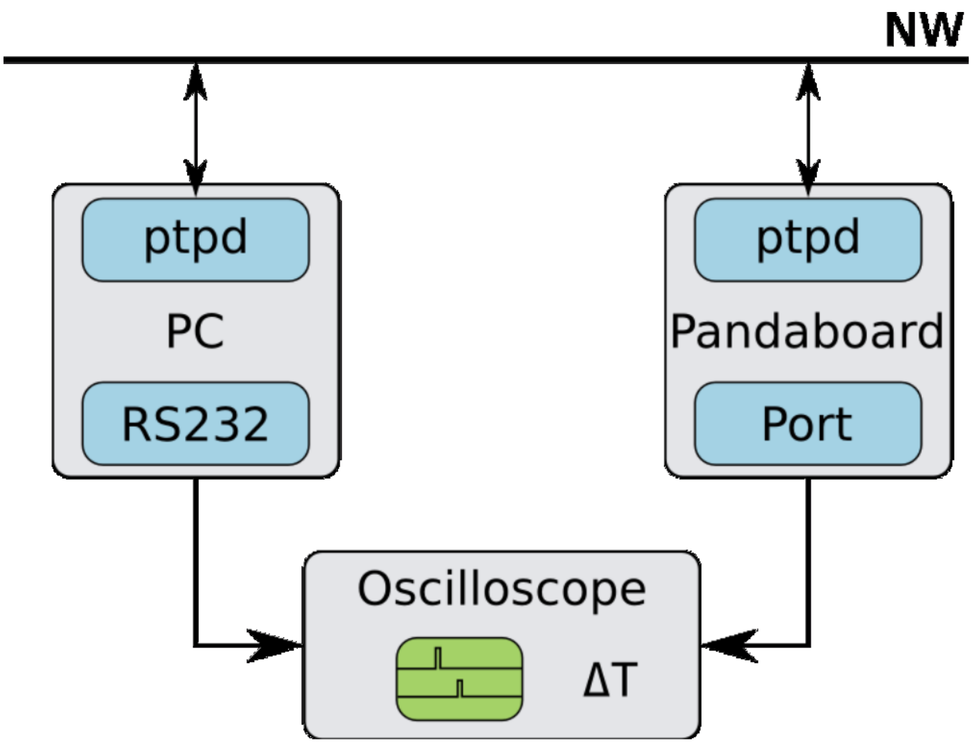

Figure 14: PTP test setup

To verify the usability of the PTP protocol for APU and server clock synchronization in advance and to specify the actual clock deviation several tests were performed using the Pandaboard evaluation hardware (see figure 9). For these tests the PTP Daemon 2 (ptpd2[14]) implementing the IEEE-1588-2008 standard was used to synchronize a PC and a Pandaboard. To be able to measure the actual clock deviation a small test program was implemented. This program toggles a pin on the RS232 of the PC and a port pin on the Pandaboard in a static interval based on each local clock. Using two channels of a digital oscilloscope these two signals can be visualized and the synchronization deviation and jitter can be measured. One such measurement of the clock deviation between APU and BEMO server (approximately 235 µs) is shown in Figure 15.

[14] , http://ptpd.sourceforge.net/

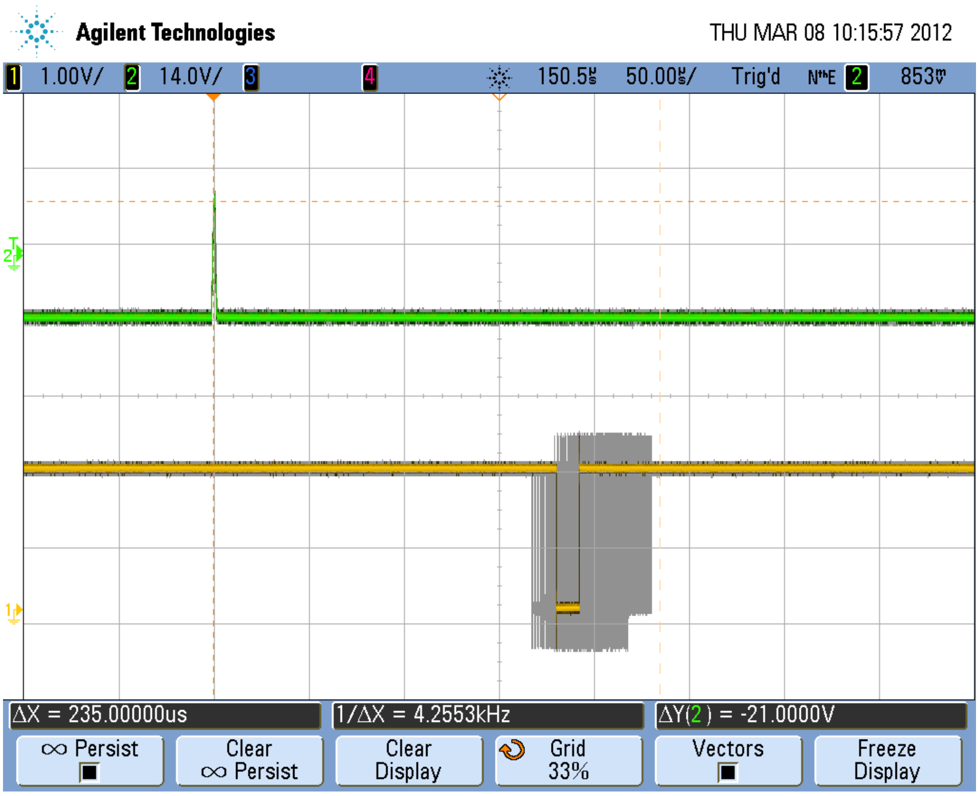

Figure 15: Clock deviation measurement

4.2.3 Remote Access

Taking the above mentioned remote access requirements into account, the remote access feature to the APUs should be at least capable of providing a command interface to the BEMO server and the APUs and a means of transferring files between project members IT infrastructure and components of the occupancy sensor network in every demo site. Figure 16 shows an example setup of an installation providing the required remote access features. The occupancy sensor network is accessible through the firewall and/or "demilitarized zone" (DMZ) of the IT network of the demo site through the Internet by using for example a secure channel like a VPN by the S4ECoB partners requiring access the network. The kind of secure connection between project partner and demo site depends of course from the solution the IT of the demo site can provide.

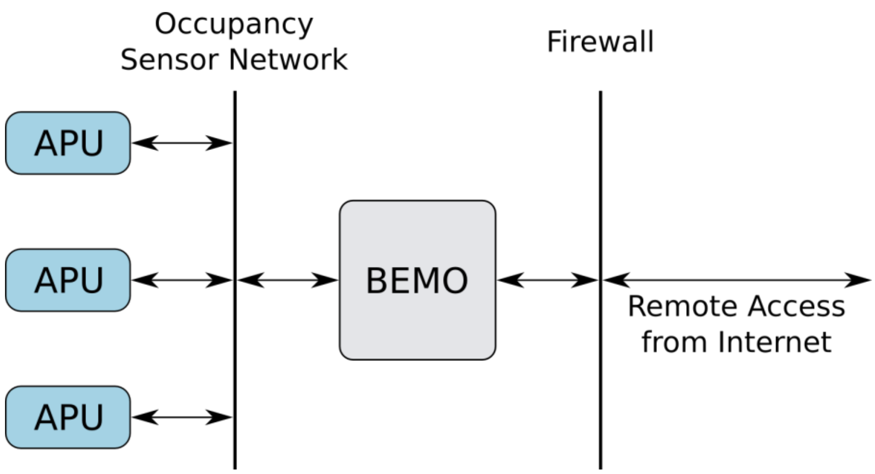

Figure 16: Possible remote access to the occupancy sensor network

4.2.4 Security Requirements

To ensure network security several measures have to be taken into account. First of all the application of basic security policies has to be ensured. This includes for example the disabling of unused wireless interfaces like WLAN or Bluetooth of the APUs, make sure unneeded (network) services are not started (on the APUs and the BEMO server) and network access is protected by a firewall (BEMO server). Access to command interfaces of the APUs and the BEMO server has to be protected by authentication using (secure) passwords or certificates. If (public key) certificates are used for authentication great care has to be taken on choosing the PKI algorithm, the certificate provider and/or handling the private keys.

Avoiding eavesdropping and man-in-the-middle attacks, communication between the components of the occupancy sensor network should be encrypted in general. A method of encryption has to be chosen, which takes into account not only the security of the transferred data but also the amount of data which has to be transferred in respect to the processing power of the APUs, as the encryption process may be limited by it.

As remote access from the Internet is required (see 4.2.3), this communication channel has to be paid attention in particular, as it is the likeliest to be exposed to attacks. A VPN solution compatible with the IT infrastructure of the demo site should be considered to enable a secure access of the S4ECoB partners to the sensor network.

4.3 General System Level Requirements

First of all the electrical safety of all components of the occupancy sensor network has to be guaranteed. This involves mainly the power supply of the APUs and ASUs, as they will be installed at various locations in the demo sites.

As the environment conditions at the installation spots of the sensor network components could vary, a sufficient protection against such conditions like for example humidity or dust must be taken into account for example in choosing an appropriate housing for the components of the network. Another system requirement is covered by the robustness of an S4ECoB installation.

Another important point is the resilience to power failures in several ways. For example the power failure of one or more components of the S4ECoB installation should be detected by all other parts and does not lead to unintended behaviour of the complete occupancy network. After a reestablishment of the power supply of the disconnected unit, it should be automatically reintegrated into the sensor network. Special attention has to be paid at possible power loss situations in critical system states like for example a firmware update of the APUs. Even after such an event the unit should be able to recover and of course restore its remote accessibility.

The same rules apply to the complete occupancy sensor network. After a black out of the complete system, all components have to recover completely and restore their respective states and communication channels.

In summary the following system level requirements have to be fulfilled by the S4ECoB sensor network communication and acoustic processing system (Table 10):

Requirement	**Description**
$R_N5.1$	Electrical safe power supply for APU
$R_N5.2$	Robust APU housing
$R_F5.3$	Self-setup of sensor network after temporal power outages
$R_F5.4$	Robust and functional safe APU firmware update process

Table 10: System level requirements

5 TECHNICAL SYSTEM VALIDATION

5.1 Background and Introduction

In this chapter the validation methods and tests of the technical components in the S4ECoB system, which have to be performed to ensure a proper working system, are described. Here the focus lies on the validation of the technical system that includes the communication between ASU, APU and the software components running on the BEMO server (namely the APU gateway). Furthermore the ruggedness of the components against power failure, proper operation under various climatic conditions and long-term stability has to be tested. The validation methods for the acoustic occupancy detection system are described in D2.4. The results of the tests will be subsequently reported in the corresponding deliverables (D3.2, D4.1).

5.2 Validation of Communication and Processing

The technical system validation is part of the pre-validation of the acoustic system. Some more details regarding the pre-validation are given in D2.4 (chapter 6.3.1).

5.2.1 APU Communication Test

In this test the basic communication (receiving and encoding of the ADAT protocol and transmission to the APU processor) should be validated. Therefore a commercially available ADAT PCI card (assembled in a PC for test purposes) should be connected to the APU. Different audio data streams (e.g. sinusoidal, ramp signals) have to transmitted and recorded in the APU. Finally the data has to be analysed regarding correct data decoding and transmission.

Tested requirements: R_F1.1

Results documented: D3.2

5.2.2 Audio Sensor Network Communication Test

With this test the communication in the audio sensor network (data transfer between ASU and APU) has to be verified. The following transmission lines have to be checked:

- Transmission of the sample clock from APU to ASU

  The sample clock is provided by a clock generator IC on the expansion board of the APU and distributed to the connected ASUs. The clock frequency, noise and jitter have to be measured.

- Transfer of 8 channel audio data, using the ADAT protocol from ASU to APU

  In deliverable D3.1 the architecture of the ASU is described. The digital audio signals are encoded into the ADAT protocol using a Wavefront ALG1401 ADAT IC. A transceiver IC adapts the signal to RS485 level. To check the proper audio transmission to the APU a sinusoidal signal should be applied to the line inputs of one audio satellite unit, the raw data has to be recorded on the APU and analysed afterwards.

To validate the requirement R_F1.2 this test should be performed with a 40 m cat 5 cable, connecting the APU and the audio satellite unit.

The signal integrity of the connection between ASU and APU should be tested using a 40 m cat 5 cable by performing a sample clock and audio data stream test described above in a noisy environment e.g. in the presence of a cell phone network and WLAN near the transmission path.

Tested requirements: R_F1.1, R_F1.2, R_A1.3, R_A3.3

Results documented: D3.2

5.2.3 Time Synchronization

As explained in chapter 4.2 time synchronization between the different components in the occupancy sensor network is necessary. To check this requirement the clock deviation and clock jitter between the BEMO server and one APU have to be measured.

Tested requirement: R_F4.2

Results documented: D3.2

5.2.4 Sensor Signal Propagation Delay

The estimated occupancy level should be transferred to the BEMO server in near real time. Therefore the propagation delay of the occupancy sensor, ASU and APU should be measured. This should be done by generating a signal with a loudspeaker and logging the arrival of the message on the APU gateway. The time difference between these two events has to be measured.

Tested requirements: R_F3.1

Results documented: D3.2

5.2.5 Remote Access

The remote access to the APUs is an important feature to validate and debug the S4ECoB system. The following test should be performed:

- Remote access and firmware update in the lab
- Remote access and firmware update of one APU installed by a S4ECoB partner

Tested requirements: R_F2.3,

Results documented: D4.1, D3.2

5.2.6 Robust Firmware Update Process

As explained before the remote access is an important feature to ensure firmware and parameter updates of the APU. To test the safety operation after a power failure or network failure during a firmware update the following tests have to be performed:

- Switch of the power supply during a firmware update
- Disconnect the APU from the occupancy network during a firmware update

Tested requirements: R_F5.4

Results documented: D3.2

5.2.7 BEMO Communication

In this test the occupancy network communication, the data transfer between APUs and APU gateway will be validated. The APU gateway is a software module running on the BEMO server, the design and development of this gateway are described in deliverable D4.1. The following tests have to be performed:

- Data transmission of occupancy level, APU status and raw audio data
- Encryption of the data transfer
- Test with multiple APUs in one network

These tests should be performed in a lab test and realistic conditions, like the IMMS company network.

Tested requirements: $R_A2.1$, $R_A2.2$, $R_A3.3$

Results documented: D4.1, D3.2

5.2.8 APU Network Log-in

The APUs automatically log-in on the APU gateway running on the BEMO server. To ensure a stable communication each APU is required to send a heartbeat message in a regular interval (every 7 seconds). If the deadline for this message is missed the network connection to this APU is aborted and it gets removed from the gateways list. The following test should be performed to check these features:

- Connecting one APU to the occupancy sensor network and check the login on the APU gateway
- Connecting multiple APUs to the network and check the login on the gateway
- Remove one APU from the network and check the status change in the APU gateway and the log file.

Tested requirements: $R_F2.4$

Results documented: D3.2

5.2.9 Environmental Conditions and Power Consumption

The S4ECoB system will be installed in three demo sites with different environmental conditions. The following tests should be performed to ensure a proper work of the APU under different conditions.

- Temperature Test

  Test of the APU in a conditioning cabinet by changing the environmental temperature from 0° to 50° Celsius.

- Power consumption

  The power consumption of the APU and ASU in different configuration has to be measured.

Tested requirements: $R_N3.2$, $R_N5.2$

Results documented: D3.2

5.2.10 Long-term Test

To ensure the proper work of the S4ECoB system over a longer time period a long term test should be performed.

Tested requirements: R_F5.3

Results documented: D3.2

5.3 Validation at the Demo Sites

5.3.1 Remote Access

After the installation of the APU in one demo site firstly the remote access to the APU has to be checked. Therefore a remote access from the internet through the BEMO server to the APU has to be performed.

5.3.2 Communication Test

After the installation of the S4ECoB system in a demo site the basic communication has to be checked, the following tests have to be performed:

- Log-in of the installed APUs on the APU gateway

  After the APUs are installed and power supplied they have to automatically log-in the APU gateway. The APUs should be disconnected and connected again to verify this feature.

- ASU – APU communication

  Each installed microphone should be tested by generating a sound nearby and check in the APU gateway GUI by using the level test plugin on the APU if the sound was detected.

6 CONCLUSION

The purpose of this document is to define requirements on the communication channels between the components of the S4ECoB system, namely APU, ASU, BEMO server and between the BEMO server and the existing building management system. Therefore, different types and amount of data had to be respected.

In this deliverable the communication and sensing concept as well as the overall architecture of the S4ECoB system were described. Based on the sensing concept presented and the requirements defined in the deliverables D2.2, System Requirements, Standards and Architecture Sensing Specifications, and D2.4, Acoustic Processing System Specification, Validation, and Preparation for Deployment the requirements for the audio sensing network, the communication between ASU and APU and the occupancy sensor network, the communication between APU and BEMO server were defined. For these communication channels available protocols and interface standards were presented and suitable solutions were selected.

Furthermore, the hardware and software requirements, which are the basis for the development of the embedded acoustic processing unit, were specified. Based on this the APU hardware architecture was developed. Various software requirements were discussed and results of benchmarking and preliminary test activities were presented. Additionally, system level requirements for the APU and the sensor network were defined.

Finally, the technical system validation and corresponding tests, which have to be performed to validate the defined requirements, were described.