HAVEit

Highly automated vehicles for intelligent transport

7th Framework Program
ICT-2007.6.1
ICT for intelligent vehicles and mobility services
Grant agreement no.: 212154

The future of driving.

Deliverable D22.2
Communication Hardware

Version number	Version 1.0
Dissemination level	CO / public version
Lead contractor	Continental Automotive GmbH
Due date	31.01.2010
Date of preparation	31.01.2010

Authors

Hannes Stratil, EFKON

Laurent Bouraoui, INRIA

Paulo Resende, INRIA

Project Manager

Prof. Alfred Hoess

Project Co-ordinator

Dr. Reiner Hoeger

Continental Automotive GmbH, STA EG

Siemensstrasse 12

93055 Regensburg

Germany

Phone +49 941 790 3673

Fax +49 941 790 13 3673

e-mail reiner.hoeger@continental-corporation.com

Revision and history chart

Version	Date	Reason
0.1	2009-12-09	Initial template by EFKON
0.2	2010-01-14	Description of V2V communication component
0.3	2010-01-22	Description of V2I communication component
0.4	2010-01-23	Internal Review
0.5	2010-01-29	Comments of Peer Review implemented
1.0	2010-01-31	Final editing and submission

Table of contents

List of Figures

List of Tables

Executive summary

This document includes the detailed description of the wireless communication components used in the HAVEit demonstrators. The document is split into two main parts: the V2V and the V2I communication components. Each of these chapters contains general information, architecture description, and interface specifications of the modules.

1 V2V Communication

1.1 System Overview

In HAVEit, V2V communication will be applied in the demonstrators of WP5200 and WP5400. The V2V communication module of a HAVEit demonstrator communicates with one dedicated communication module of another demonstrator vehicle.

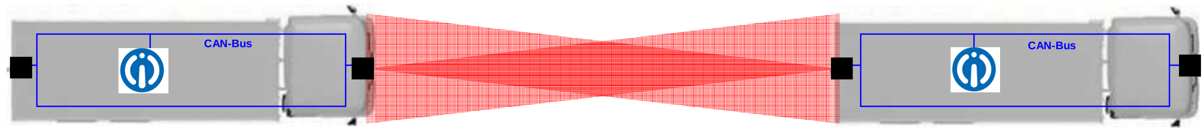

Figure 1: V2V communication

The V2V system consists of two hardware components: a front and rear transceiver. These stand-alone units include the optics, processing and CAN hardware all in one package. Therefore, only power and CAN lines must be routed to the units. This simplifies installation on the vehicle.

When selecting mounting locations, it is important to consider several factors, including:

- Approximate optical alignment with any vehicles it will communicate with
- Remain relatively free from dirt and water spray
- Not be exposed to mechanical damage

The mounting position of the V2V units for WP5200 (AQuA truck) and WP5400 (Active Green Driving Bus) is shown below. During mounting of V2V units, it is important to ensure approximate optical alignment of the sensors. The WP5200 truck and WP5400 bus will be the primary vehicles used to communicate with each other throughout development.

1.1.1 WP5200 AQuA Truck

Integration of the V2V units on the AQuA truck is shown in Figure 2. The units will be mounted high on the vehicle, as this location is most appropriate to allow optical alignment on both the AQuA truck and WP5400 bus. The front unit will be properly integrated into the upper headlight mounting. The rear unit will be mounted at the top of the rear cargo door, using the door locking mechanism as a mounting point.

The figure below contains following: (a) mounting location of the V2V unit at the front of the AQuA truck, fully integrated into the overhead head light module, (b) mounting location of the V2V unit at the rear of the AQuA truck, near the locking mechanism of the cargo door, for the rear of the AQuA truck.

Figure 2: Integration of the V2V units on the AQuA truck.

1.1.2 WP5400 Active Green Driving Bus

Integration of the V2V units on the WP5400 bus is shown in Figure 3. The rear unit will be installed on the mesh above the engine compartment and the front unit will be mounted inside the upper left of the windscreen.

Figure 3: Mounting locations for the V2V units on the WP5400 bus.

1.2 Hardware Architecture Description

The V2V communication hardware is placed within a square metal housing. This metal housing fulfils all requirements for temperature range and outdoor mounting. The metal housing provides gadgets for the mounting of the components. The front plate of the V2V communication module contains the lenses for the wireless Infrared communication.

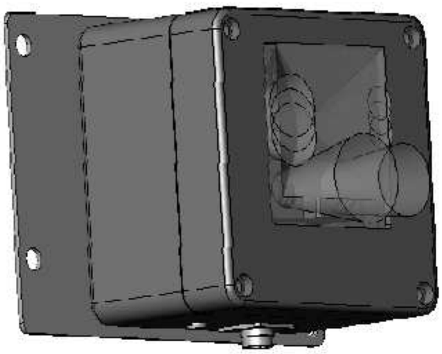

Figure 4: V2V Communication Module

The electronic hardware of the V2V communication module has three main components.

- Controller board
- Base board
- Infrared board

1.2.1 Controller board

The controller board is the brain of the V2V communication components. This PCB is equipped with an ARM-9 µController and the required memory chips. The µController hosts the software modules of the V2V communication board. (see Section 1.3).

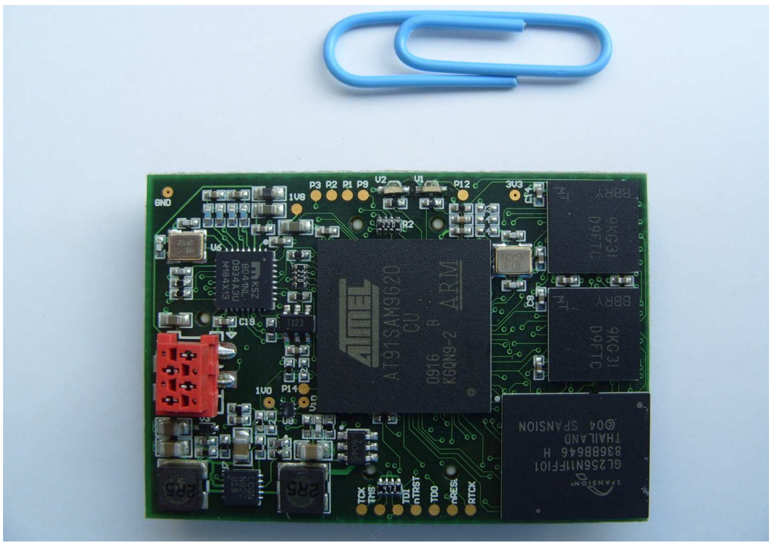

Figure 5: Controller Board

1.2.2 Base board

The base board forms the backbone of the component. Controller board and Infrared board are connected to this PCB. The base board contains the chips for the CAN communication interface and the CALM-IR ASIC. The CALM-IR ASIC was developed by EFKON and contains the CALM Infrared communication protocol as specified in the ISO 21214 Standard [1].

1.2.3 Infrared board

The Infrared PCB hosts the Infrared LEDs for the transmission of Infrared light and PIN-Diodes for the reception of Infrared light. The placement of the diodes is important for the shape of the Infrared beam. Figure 6 shows the shape of the CALM-IR beam and Figure 7 a picture of the Infrared PCB.

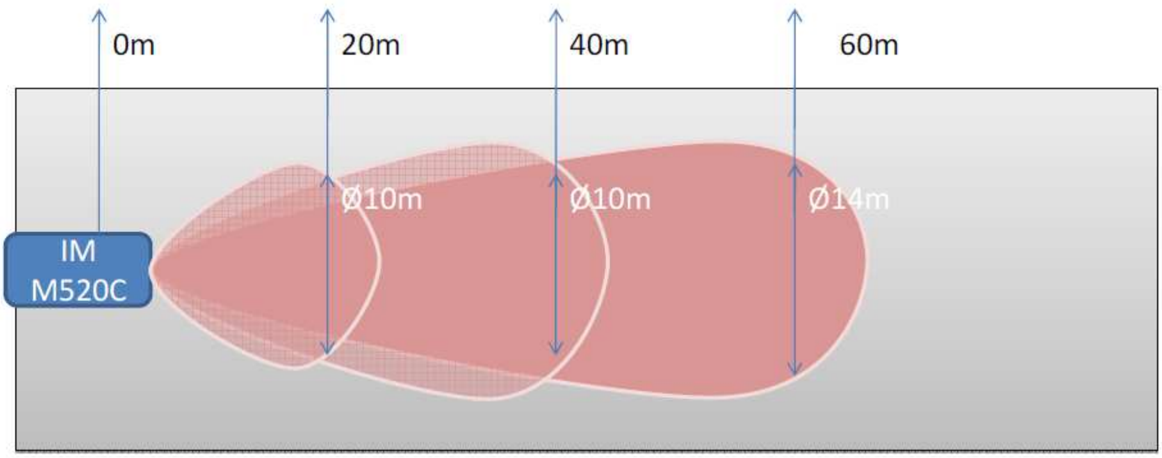

Figure 6: Shape of CALM-IR Beam

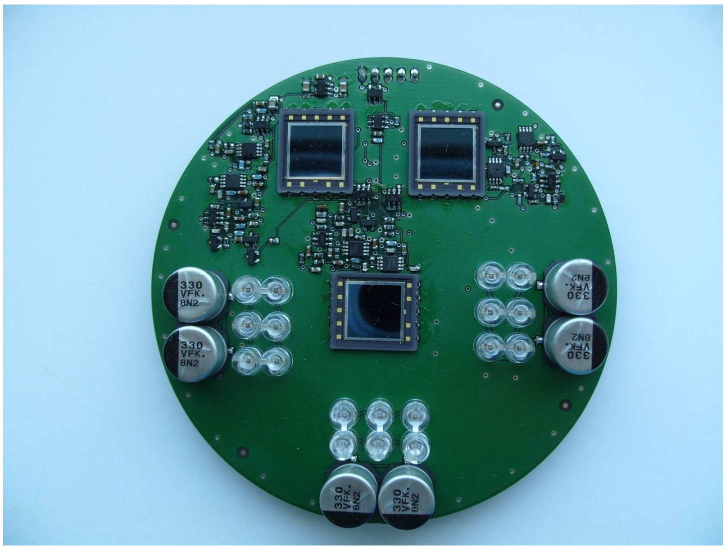

Figure 7: Infrared Board

1.3 Software Architecture Description

The software of the V2V communication component runs on an ARM-9 µController. Linux is used as Operating System for the controller. The software architecture of the V2V communication component consists of three main components.

- HAVEit application
- CAN communication stack
- CALM IR communication stack

An overview of the software modules is given in Figure 8.

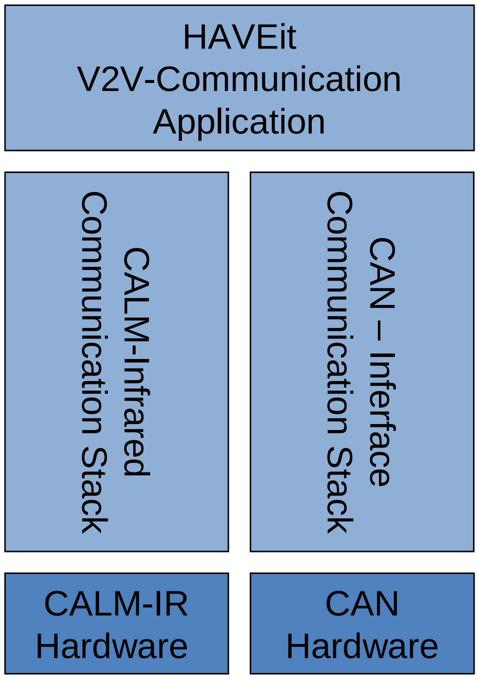

Figure 8: V2V Communication Module - SW Architecture

1.3.1 HAVEit Application

The HAVEit application of the V2V communication component is the responsible for the message generation and for the exact timing of the communication process. The application gets the required data from the CAN communication stack and builds, with the data of the CAN messages, the messages for the wireless Infrared communication. These messages will be forwarded to the CALM Infrared communication stack.

The messages received from the CALM Infrared communication stack will be forwarded to the CAN communication stack.

1.3.2 CAN Communication Stack

The CAN communication stack reads the required messages from the in-vehicle CAN bus of the HAVEit demonstrator and forwards these messages to the HAVEit application. The messages received from the HAVEit application will be written to the in-vehicle CAN bus.

1.3.3 CALM IR Communication Stack

The CALM IR communication stack is responsible for the wireless Infrared communication. The CALM IR communication stack is implemented according to the CALM ISO Standards.

The family of International Standards based on the CALM (Communications access for land mobiles) concept are developed by ISO TC204 WG16. This family of standards specifies a common architecture, network protocols and communication interface definitions for wired and wireless communications using various access technologies including cellular 2nd generation, cellular 3rd generation, satellite, Infrared, 5 GHz micro-wave, 60 GHz millimeter-wave, and mobile wireless broadband. These and other access technologies that can be incorporated are designed to provide broadcast, unicast and multicast communications between mobile stations, between mobile and fixed stations and between fixed stations in the "Intelligent Transport Systems" (ITS) sector. An overview of the CALM family is given in Figure 9.

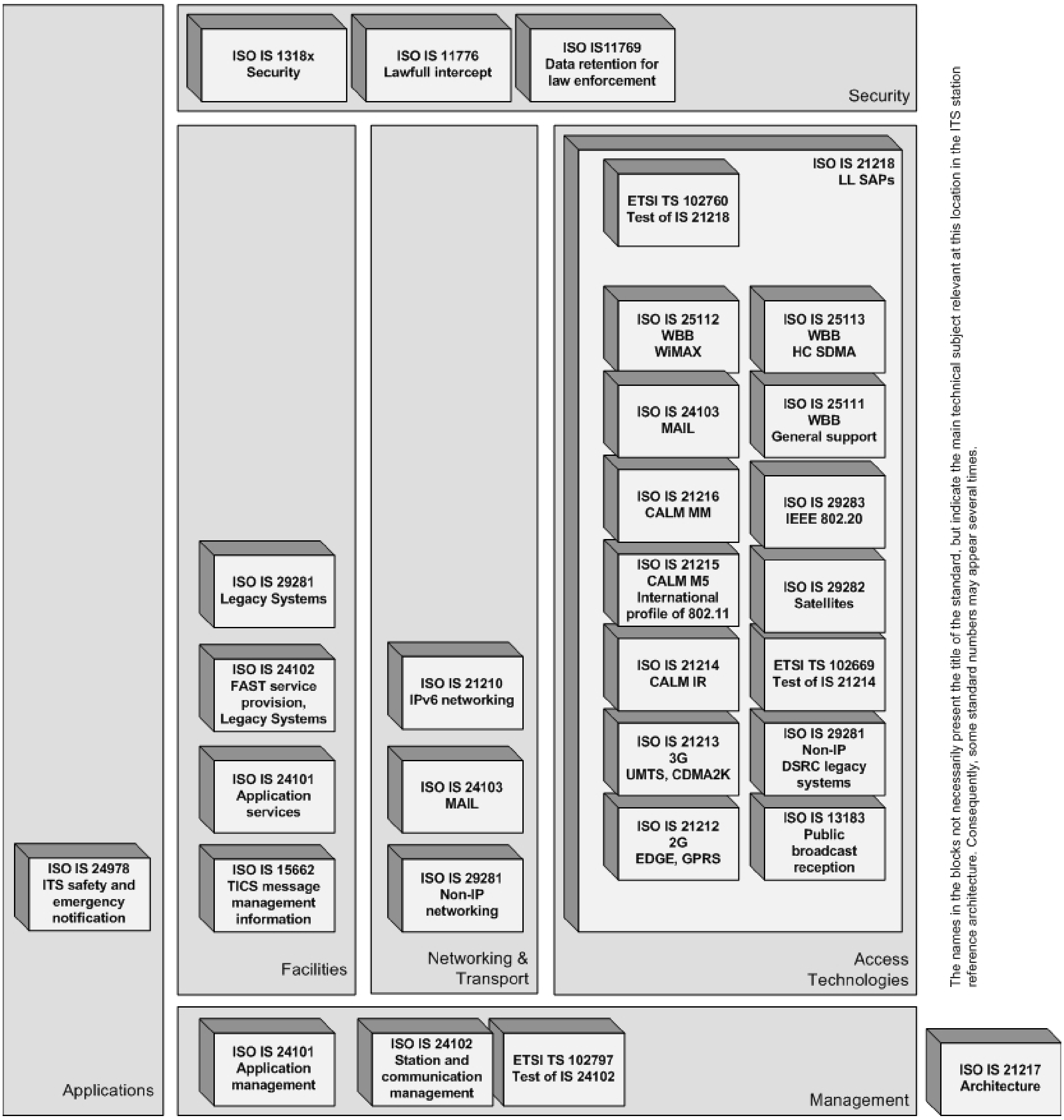

Figure 9: Family of CALM Standards

The lower layers of the Infrared communication are standardized within the ISO 21214 CALM Infrared Standard [1]. The CALM Architecture is described in ISO 21217 [2] and the communication protocol for non-IP communication is standardized in ISO 29281. [3]

The CALM IR communication stack is continuously trying to build up a communication link with another V2V communication module. The communication link would be established as soon as another V2V communication module is within the Infrared beam of the communication module. The next step is to run a master election algorithm to identify the master and the slave of this communication station. The complexity of the master election algorithm increases with the number of modules within the communication beam. The transmission of the messages starts immediately after the required communication management tasks. The Infrared communication is described in Section 1.5.

1.4 Message Description

This Section describes messages and data that are exchanged between the V2V communication modules mounted in the HAVEit demonstrators of WP 52000 and WP5400.

The transmitted data are combined within three messages:

- Vehicle Data
- Driver Data
- Vehicle Position

1.4.1 Vehicle Data Message

Data	Description
Velocity	Velocity of vehicle
Yaw Rate	Yaw rate of vehicle (left is positive – ISO)
Steering Angle	Steering angle (left is positive – ISO)
LongAcc	Longitudinal acceleration of vehicle

Table 1: Vehicle Data

1.4.2 Driver Data Message

Data	Description
BrakeActive	Are brakes active
AccSetspeed	ACC setspeed
AccOn	Is ACC active
Turn Indicator	Turn Indicator mode

Table 2: Driver Data

1.4.3 Vehicle Position Data

Data	Description
Longitude	Longitude according to WGS84
Latitude	Latitude according to WGS84

Table 3: Position Data

1.5 Interface Description

This section gives a description of the interfaces provided by the V2V communication component. The focus is on the wireless Infrared interface in (Section 1.3.1), the CAN interface is briefly described in Section 1.3.2. A more detailed description of this interface is given in the D.22.1 document [4].

1.5.1 CALM Infrared Interface

The Infrared communication technology offers the capability to communicate in predefined directions. The physical medium INFRARED at 850 nm is used in incoherent mode (± 50nm). The IR-receiver is a detector which is sensitive to Infrared modulated power, no phase is necessary to be encountered at all. This matter of fact turns out very positive in respect to reflected signals from the road, traffic geometry or neighbor lanes – no destructive superposition takes place.

The Infrared communication technology used in HAVEit is standardized within the ISO 21214 standard. This standard is part of the family of International Standards for CALM (Communications access for land mobiles) which are the ISO-Standards for vehicle-to-vehicle and vehicle-to-infrastructure communication.

The CALM-group creates and standardizes an automotive, wireless and media-spanning communication environment. CALM (Communications access for land mobiles) is part of the ISO-standardization-program (TC204-WG16). The ultimate target is, to provide automatically the most appropriate communication-medium to the user. The deployment scenario for the CALM communication can be seen in Figure 10. The CALM-program standardizes also the necessary networking, switching and management functionality.

The CALM-IR standard defines a Time Division Muliple Access (TDMA) scheme as media access method for synchronized communication of multiple communication partners. In a communication environment with two or more communication partners there exists exact one master, which controls the organization of the TDMA sequence. If no dedicated master exists, a procedure is provided to establish a new master.

The CALM IR-frame consists of several time slots and is defined and organized by the master. The frame starts with a sequence of reserved signals which never can occur in a data stream. This allows simple detection circuitry without the necessity to constantly supervise and analyze the data stream.

The CALM-IR TDMA frame is generated by the master and starts with a specific frame synchronization signal and is subdivided into several timeslots. The maximum length of the frame is 256 time slots. The CALM-IR TDMA frame is subdivided in communication windows, and each frame contains at least one window (see Figure 11).

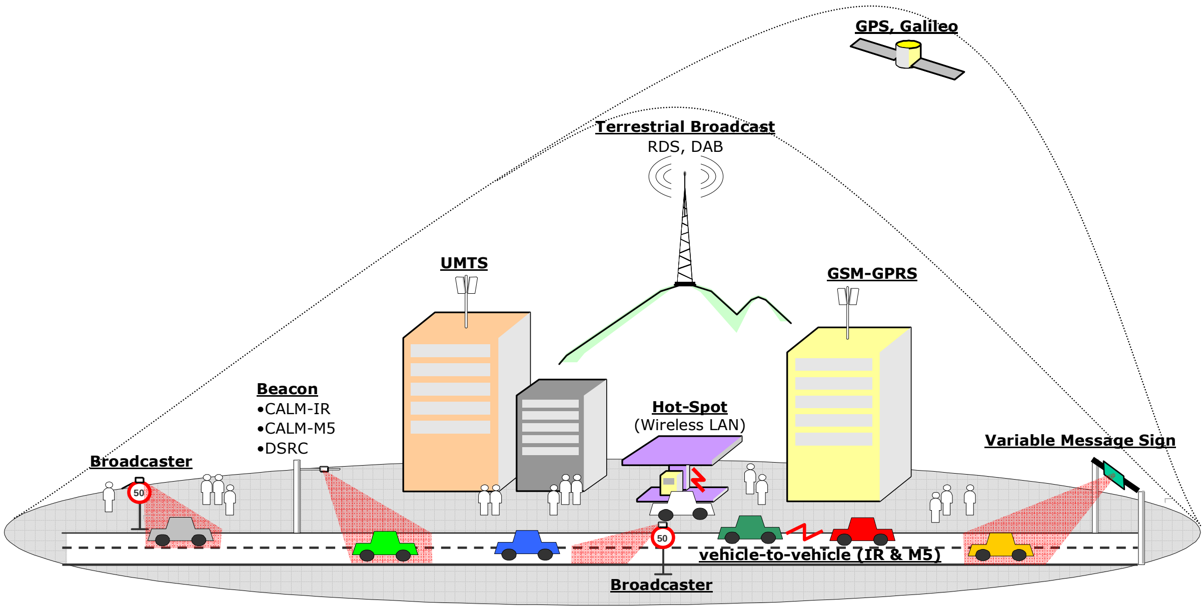

Figure 10: Scope of CALM

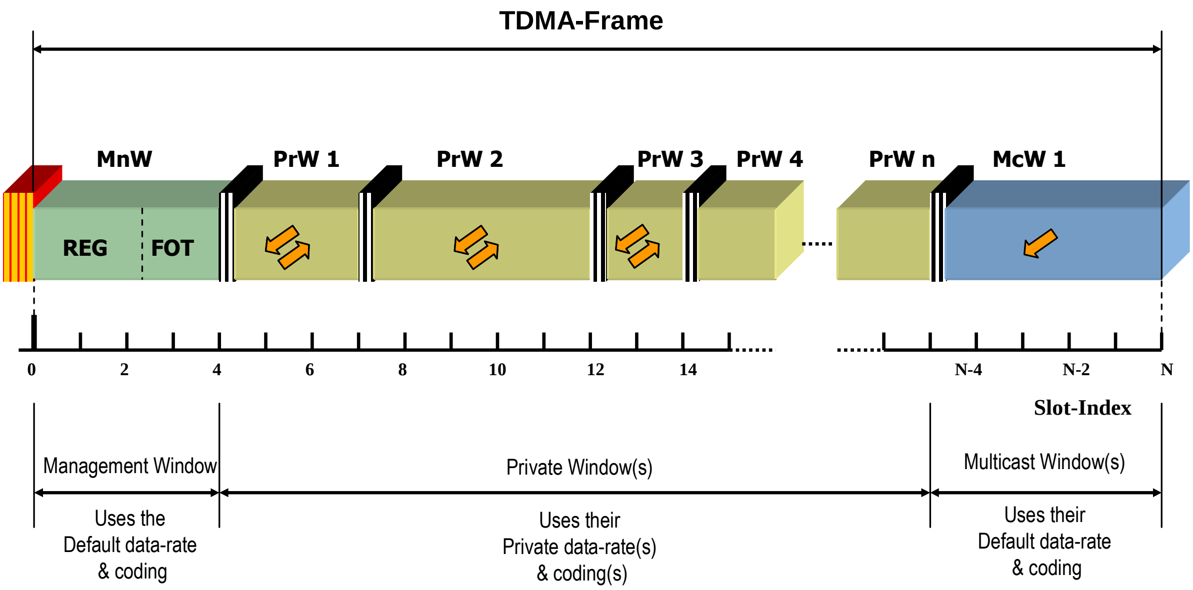

Figure 11: CALM IR TDMA Frame

Each window is initiated by a specific window synchronization signal. The maximum number of windows within one frame is a dynamic parameter and depends on the size of the windows. The first window of a CALM-IR TDMA frame is always the management window.

The management window contains some general information as well as size and owner of the communication windows. The respective owners can use their communication windows to transmit their messages.

1.5.2 CAN Interface

The V2V communication module developed within HAVEit reads messages from the in-vehicle CAN bus and transmits the data via an Infrared communication link to another V2V communication module. The receiving communication module writes the CAN message to the CAN bus and uses the ID of the transmitting communication module to indicate the origin of the written data. A detailed description of the communicated CAN messages can be found in the D.22.1 document [4].

The CAN Network of the WP5200 demonstrator vehicle is shown in Figure 12. The data received by the V2V communication unit are used by the Sensor Data Fusion ECU to generate an “object tracker” of the other vehicle. D.52.2 [5] gives a detailed description of the Sensor Data Fusion algorithms used in HAVEit.

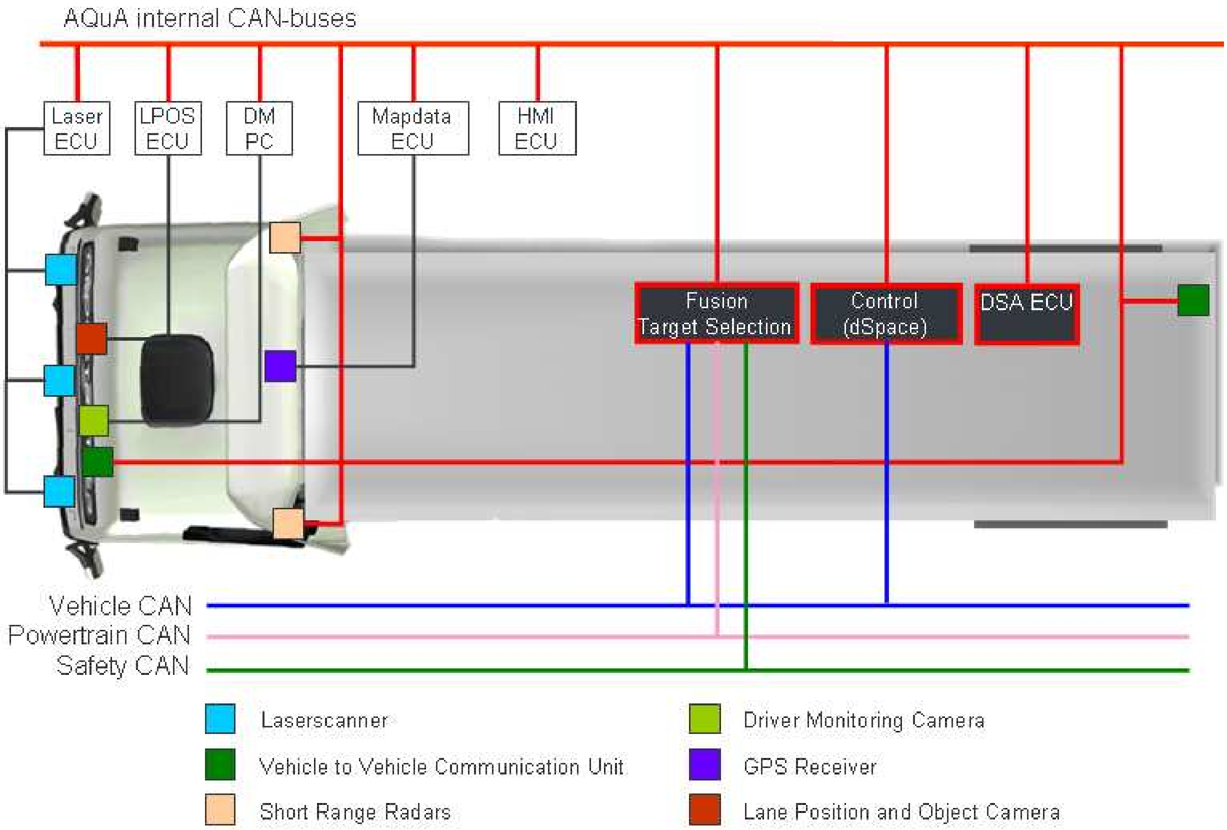

Figure 12: In-Vehicle CAN Buses

2 V2I Communication

2.1 System Overview

Vehicle to infrastructure (V2I) communications can provide useful information to improve the world perception that on-board sensors cannot provide due to their limited range. To demonstrate the concept of V2I communications, where data is exchanged between the infrastructure and the vehicle to improve traffic management and safety, a wireless communication system is developed by INRIA within the HAVEit project.

Figure 13: 4G Cube – compact wireless router

This communications system is a Vehicle Web Service Communication Framework (VWSCF) also called SCOPE that has automatic service discovery. The proof-of-concept consisted of setting a dynamic speed limit sent by the infrastructure to the HAVEit FASCar demonstrator vehicle. This received speed was then used by the CoPilot component to adjust the current vehicle speed set point. A first version of this communication system was already integrated into the HAVEit Joint System Framework that is running in the FASCar demonstrator vehicle [6].

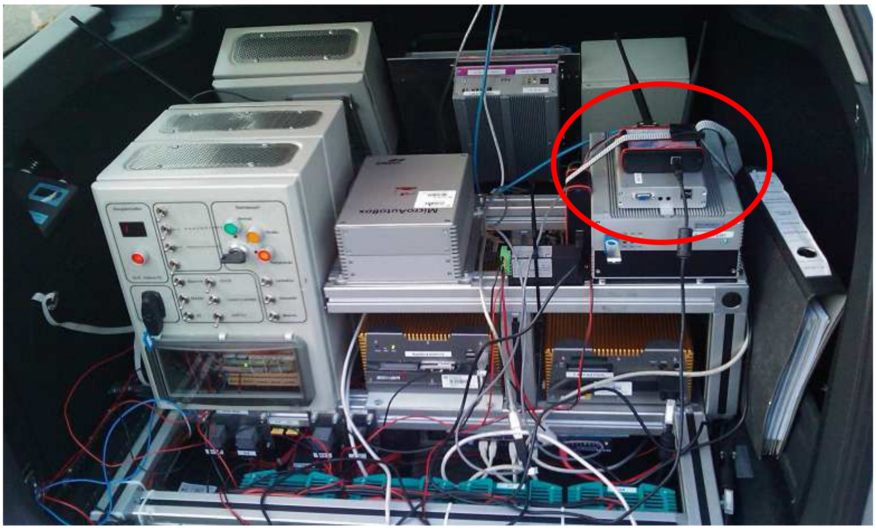

Figure 14: 4G Cube integrated into the FASCar

The hardware added to the FASCar consists of a 4G Cube (Figure 13) – a compact wireless router running SCOPE and OPENWRT router operating system – connected by Ethernet to the already existing onboard computer where the HAVEit Joint System Framework is running (see Figure 15).

2.2 Architecture Description

The communication architecture is modular and is based on embedded Linux boxes (4G Cubes) that are used to provide automatic connectivity for V2I applications.

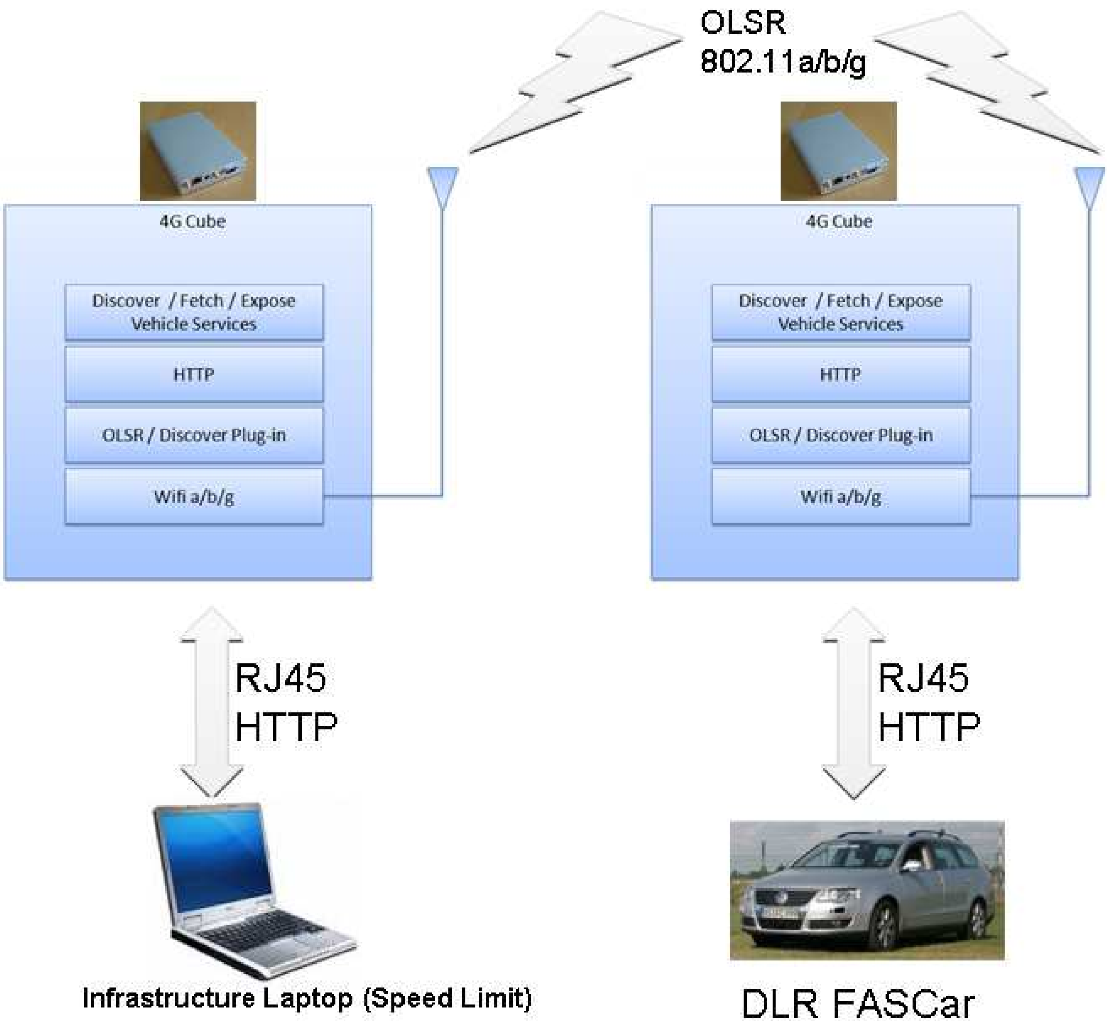

Figure 15: Communication architecture for V2I

The 4G cube features are:

- CPU: x86 500 MHz AMD Geode LX800
- IPV4 / IPV6
- OLSR Mesh

- Service discovery using OLSR
- 1 mini-PCI a/b/g
- HTTP Web services Scope Server
- Serial port
- USB
- VGA output

These boxes can be used as simple routers, or like advanced communication units.

2.2.1 Web services

The embedded communication software uses web services. These web services are scalable and any new web services can be easily developed and fast deployed. In the current version it consists of three web services: **Discover**, **Expose** and **Fetch** and can deal with different kind of data (structures, xml etc.).

For example, in a system with 4G cubes the local URL: `http://cube-xxxx:8000/discover/` can be used to discover available services, by using a standard browser. The system will reply with the following XML answer:

```
/discover/service/laser/
/discover/node/fascar/
/discover/ ?status=new&reconnect
<service status="new">
    <name>laser</name>
<node>fascar</node>
    <url>/fetch/service/laser/fascar/</url>
</service>
```

The Fetch web service is used to receive data stream exposed on the network, for example, data of one or many sensors, from one or many vehicles. This web service can only be contacted by an HTTP GET request. The URL of the position service must contain a node name (or vehicle/infrastructure name), and the name of the service (data exposed by the node). If this data are not specified, an XML listing is sent by the server containing the node list and the list of services currently discovered on the network.

The fetch service is done using the following URL: `http://cube-xxxx:8000/fetch/`

Each vehicle can expose, or send, its data through the Expose web service; it's usable only with HTTP POST request. Once the request is made, the connection is blocking and a data stream can be sent in HTTP chunk.

The expose service is done using the following URL: `http://cube-xxxx:8000/expose/`

2.2.2 OLSR

An “Optimized Link State Routing” protocol (OLSR) was also developed. Its proactive behavior is appropriate for vehicle communications. OLSR allows vehicle mesh networks to be quickly created and dynamically reconfigured, since it is designed for multi-hop networks with a strong and efficient mechanism for data flow delivery.

This protocol can run over WIFI and use UDP or Signal Noise Ratio to measure the quality of the link to establish the best route to transfer data. We use regular WIFI 802.11b/g that can have up to 54Mbit/s bandwidth (802.11a is also possible at 54Mbit/s).

Each user is able to request information to the communication hardware using a simple HTTP 1.1 protocol.

It is not necessary to deal directly with the Communication API (implemented in C++). The communication between the cubes is completely transparent and they can use either IPv4 or IPv6.

The computers or microcontrollers plugged into the 4G Cubes receive their IP by DHCP and they only need to deal with HTTP protocol to send, get or discover data on the network, no fixed IP is required.

OLSR is responsible of routing correctly and managing multi-hop data transfer. To transfer data, the user can use for example LIBCURL, a HTTP dedicated library to discover, expose or receive data. This library is multi operating system and can be embedded on C or C++ software.

If the infrastructure communication box has access to the internet, it can give the connectivity to the connected vehicles.

In the future, IPV6 coupled with NEMO will manage different media (3G, WIFI, etc...) to maintain a permanent network access to the infrastructure.

2.3 Interface Specification

The communications component is connected to the data fusion component. There is no direct link between the on board computer and infrastructure. The communication is managed by the 4G Cubes and vehicle/infrastructure data are propagated to the dynamic mesh network.

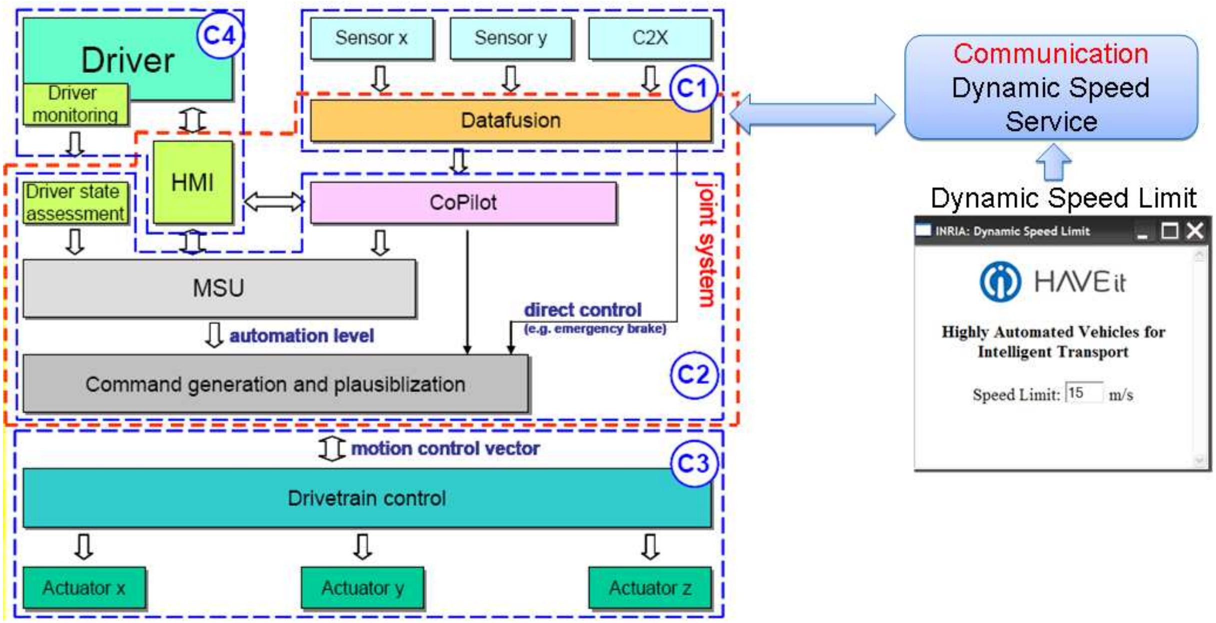

Figure 16: Interface between Joint System and Communication Component

The communications systems can send and receive any type of binary data: custom data structures, video stream, laser scans data, etc. For the following proof-of-concept, a simple data structure containing a speed limit variable in *short int format* is used.

Example of the used binary data :

```
#pragma pack(push)  /* push current alignment to stack */
#pragma pack(1)     /* set alignment to 1 byte boundary */
typedef struct Data_Communication
{
      short int speedlimit; //m/s
} Data_Communication;
#pragma pack(pop)   /* restore original alignment from stack */
```

3 References

[1] ISO 21214 CALM Infrared

[2] ISO 21217 CALM Architecture

[3] ISO 29281 CALM non-IP networking

[4] D.22.1 “Communication Specification”, HAVEit Deliverable 2009

[5] D.52.2 “Automated Queue Assistance Components installed, working and tested”, HAVEit deliverable 2010

[6] D41.1 “Concept and algorithm (1st version) validated in vehicle”, HAVEit deliverable 2009

Annex 1 Keywords

3G	Cellular Network - 3rd Generation
4G	Cellular Network – 4th Generation
API	Application Programming Interface
AQuA	Automated Queue Assistant
ASIC	Application Specific Integrated Circuit
CALM	Communications access for land mobiles
CAN	Controller Area Network
CPU	Central Processing Unit
DHCP	Dynamic Host Configuation Protocol
EBA	Electronic brake actuator
ECU	Electronic computation unit
HAVEit	Highly automated vehicles for intelligent transport
HTTP	Hypertext Transfer Protocol
IP	Internet Protocol
IPv4	Internet Protocol version 4
IPv6	Internet Protocol version 6
IR	Infrared
ISO	International Standardization Organization
ITS	Intelligent Transport Systems
NEMO	Network Mobility
OLSR	Optimized Link State Routing
PCB	Printed Circuit Board
PCI	Peripheral Component Interconnect
TDMA	Time Division Multiple Access
UDP	User Datagram Protocol
URL	Uniform Resource Locator
USB	Universal Serial Bus
V2V	Vehicle to Vehicle
V2I	Vehicle to Infrastructure
VWSCF	Vehicle Web Service Communication Framework
VGA	Video Graphics Array
WiFi	Wireless Fidelity
XML	Extensible Markup Language
t.b.d.	To be defined (later)