

# Buses, Protocols and Systems for Home and Building Automation

Ondřej Nývlt



**Evropský sociální fond. Praha & EU: Investujeme do vaší budoucnosti.**

Department of Control Engineering  
Faculty of Electrical Engineering  
Czech Technical University in Prague  
2009-2011

## Table of contents

1.	Basic categorization.....	3
1.1.	System openness.....	3
1.2.	System centralization .....	4
1.3.	System complexity and versatility.....	5
1.4.	Physical layer – communication medium.....	6
2.	Closed systems .....	7
2.1.	ABB Ego-N.....	7
2.2.	Elko EP iNels .....	8
2.3.	Eaton/Moeller X-Comfort and Nikobus.....	8
3.	Open standardized protocols .....	11
3.1.	Complex protocols.....	12
3.1.1.	BacNet .....	12
3.1.2.	Konnexbus .....	14
3.1.3.	LonWorks and LonTalk .....	16
3.2.	Specialized and other standardized protocols .....	19
3.2.1.	DALI .....	19
3.2.2.	DMX512.....	23
3.2.3.	EnOcean.....	28
3.2.4.	M-Bus (Meter-Bus).....	31
3.2.5.	OpenTherm.....	34
3.2.6.	SMI (Standard Motor Interface).....	38
4.	Other buses and systems used in home automation.....	42
5.	Bibliography.....	43

## 1. Basic categorization

Building automation systems and protocols can be divided into a plenty of categories using very different rules. In this wide set of classification properties we can determine some basic and interesting classifiers: *openness*, *centralization* or *versatility* of a system. These categories provide us important information which is crucial in assessing usability of a protocol or a system for a project.

### 1.1. System openness

This property describes dependency of a system on a manufacturer. There exist two essential categories:

- **Open protocols** (*KNX, LON, BACnet, DALI, OpenTherm, EnOcean...*)
- **Closed systems** (*Ego-n, iNels, Nikobus, XComfort*)

**Open protocols** are based on open standards or open specifications, which are accessible for everybody (who pay some fee) – not only for the manufacturer who developed the protocol. It is common that everything about the protocol is managed by a special association, not by the developer company. The most significant advantage of this approach is evident: open specification ensures a big flexibility for the building control system designer because there usually exist more than one manufacturer of a device with a desired function. Differences between devices are in a price, design or additional functions. Another advantage is that these protocols are supported by significant academic research producing new non-traditional features to the system (for example *TU Wien* is one of the leading research facilities for the *KNX* protocol). One of disadvantages of open protocols is usually the price of the system for family and small houses. These systems are cost-effective for bigger and large buildings, such as office buildings, hospitals, hotels or airports. Examples of some standards which cover the most important building automation protocols are *KNX - EN 50090* and *ISO/IEC 14543*, *Lon - ANSI/CEA 709.1*, *BACnet – ASHRAE/ANSI 135* and *ISO 16484-5*.

Specification (e.g. communication protocol) of a **closed system** is, in opposite to open protocols, not public to everyone – it is private to the developer company. There exist some exceptions – for example *Eaton* company and their *Xcomfort* system, where they offer the specification of their communication protocol in some cases to manufacturers who can add new features into their system. Closed building automation systems are closer to the end user than the open protocols – you can buy components (sensors and actors) of these systems in “supermarkets” like *OBI, Hornbach* or *Baumax*. The advantage are the prices of the components and of the whole system for family houses, flats or small houses. Usually these systems are very easy to install and to “program” (often without a computer). Closed systems are, in the most of cases, able to solve all the basic task of home automation. But they do not offer big versatility – a user can choose only from a very small group of devices and designs. Another problem is that users are dependent on one producer only and when the manufacturer stops the production of the devices, there will be a problem with the expansion of an installation and with replacement of crashed devices.

## 1.2. System centralization

Another property which divides protocols and systems into very different groups is logical or topology centralization:

- **Centralized systems** (*Ego-n, iNels*, systems based on central PLC)
- **Decentralized/distributed systems** (*KNX, LON, Xcomfort...*)
- **Hybrid systems** (*Nikobus*)

A **centralized system** (Fig. 1.2.1) has a central unit in the middle of the system (one or several units) which controls whole system functions. So the system does not need smart sensors and actors, but is very sensitive to a failure of the central unit. If it fails, then the whole system will not work. Today, these systems usually use a bus topology, but sometimes they still use direct connections to sensors and actors (a star topology – every device has its own connection with the central device).

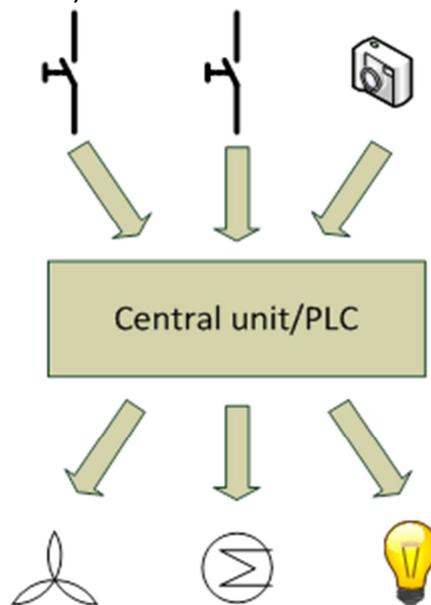


Fig. 1.2.1: Centralized system

**Distributed systems** (Fig. 1.2.2) have no central unit. That means that every unit is intelligent/smart and knows what to do (e.g. when and where to send data). This is of course a big advantage, because the system is robust and more failsafe/reliable – when one unit fails, then the others work on. Distributed systems always use a bus topology (shared medium). The disadvantage is that the devices are more expensive than in the case of “dull” devices of a centralized system.

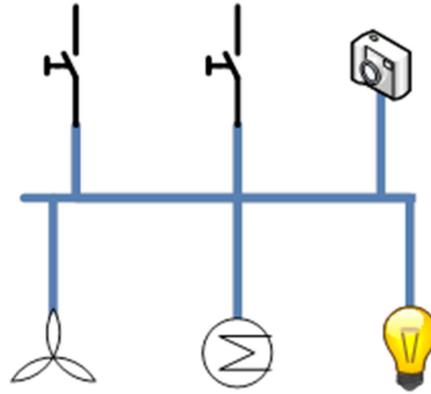


Fig. 1.2.2: Distributed system

**Hybrid systems** (Fig. 1.2.3) are somewhere in between – one example is *Nikobus*, where inputs (sensors) are connected using a bus and outputs (actors) are connected direct by using star topology to semi-central units.

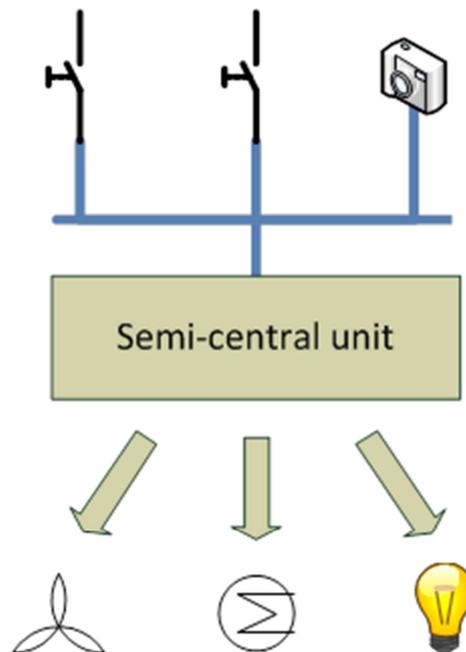


Fig. 1.2.3: Hybrid system

### 1.3. System complexity and versatility

This parameter represents the ability of a system to cover one or more control tasks in building and home automation:

- **Complex control system** (*KNX, LON, Xcomfort, Ego-n*) are able to solve every basic task in building automation (for example HVAC<sup>1</sup>, lights, shutters/blinds, visualization or basic security)
- **System and protocols focused on one control task:** e.g. *DALI* bus specialized on light control or *OpenTherm* focused on heating control

<sup>1</sup> Heating, Ventilating, and Air Conditioning

## 1.4. Physical layer – communication medium

Standardized complex protocols usually offer more than one physical layer (up to 5 or 6), and commercial closed building control systems traditionally offer only one. This situation is changing today, because it is more and more common that a cheap commercial closed system offers, in addition to the communication by one of wired physical layers, also a wireless communication possibility.

Here is a list of some typical physical layers used in building automation:

- **Twisted-pair/RS-485**
- **Powerline 230V**
- **Wireless – infrared, RF, ZigBee...**
- **Ethernet**
- **Coaxial cable**

## 2. Closed systems

Closed building automation systems or proprietary systems are typically focused on family and small houses. There exist tens of these systems on the market today. They are developed not only by “no name” producers, but also by traditional companies like *ABB* or *Moeller/Eaton*. Components for these systems are often offered in hobby retail chains like *OBI*, *Hornbach* and *Baumax*, and in most cases they are cheaper than any system based on an open standard.

Closed systems are usually complex, so they can solve all the basic tasks of home automation – HVAC, light and shutter/blinds control, remote control (PC, PDA, phone) and so on. A disadvantage is that they offer only very limited possibilities in functions, interconnections or designs of devices. An advantage of this kind of systems is that they often do not need any special training for installation and programming (often without PC, only using a screwdriver). We can find centralized, distributed and also hybrid closed systems.

Closed systems are based on a proprietary communication protocol with wired (*Nikobus*) or wireless connection (*Xcomfort*, *Conrad FS20* and *HomeMatic*). Wireless closed systems are becoming very popular today because their price is comparable to the price of a wired solution with the advantage of easier installation (especially for older houses). Also more and more producers start to offer systems with both types of communication – wired and wireless (*iNels*).

### 2.1. ABB Ego-N



Fig. 2.1.1: ABB Ego-n (1)

**Ego-n (2)** is a centralized bus building automation system focused mainly on family houses. It has its origins in the Czech Republic. The bus is physically realized by a 4-core cable with 2 cores for data and 2 cores for a power supply to the devices. There exist two types of buses (Fig. 2.1.2): primary (max 700m length) and secondary (max 2000m length). The primary bus connects sensors and actors (up to 64 devices on the bus) with a central control unit. The secondary bus is optional and it connects up to 8 central control units, a GSM unit, a unit of logic functions, a TCP/IP module and other high-end devices.

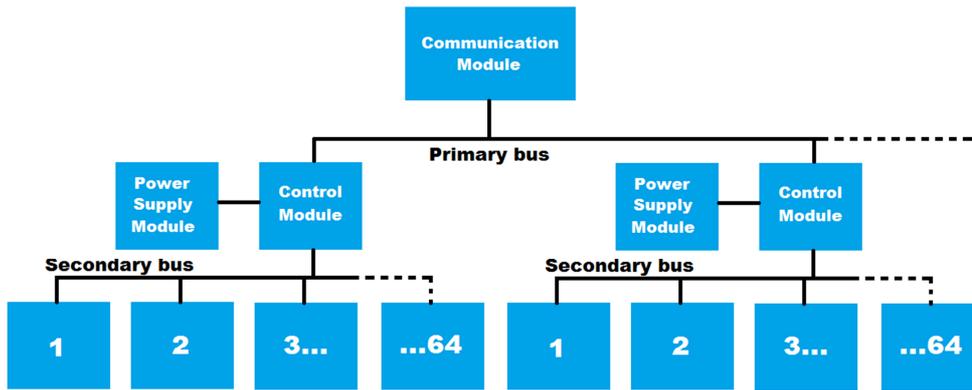


Fig. 2.1.2: Ego-n Bus structure – translated and redrawn from (1)

Ego-n can solve typical simple home automation tasks: HVAC, shutter/blinds and light control, motion detection, simulation of a presence of persons in the house, control over GSM, PC or PDA, fire and security alarms and so on. The configuration of the system can be done using a computer or by so-called “button mode” without a PC.

## 2.2. Elko EP iNels



Fig. 2.2.1: iNels (3)

**iNels** is a typical fully centralized automation system based on the special central PLC-like unit *CU2-01M* or on the modular *PLC Tecomat Foxtrot*. This system is not focused only on family houses, but with a usage of the *PLC Foxtrot* also on BMS (building management systems) of large buildings. In the case of installation with *CU2-01M* (up to 192 connected I/O devices) in family houses, the system is capable to solve every typical task (e.g. HVAC, lights, shutters/blinds) plus some specialties, such as security and fire safety features, energy management, voice control, control over TV, PC or GSM. If *PLC Foxtrot* is used as the control unit, then the possibilities of the system are much wider (up to 288 connected I/O devices using 9 *CIB*) and of course much more expensive. **iNels** offers communication over its own bus called *CIB* (or over *CAN*). A new feature is a wireless connection of sensors and actors using protocol *RFox*. In the case of the *PLC Foxtrot* used as the central unit it is possible to communicate over protocols *LON*, *BACnet*, *DALI*, *M-Bus*, *Profibus DP* or *Modbus*.

## 2.3. Eaton/Moeller X-Comfort and Nikobus



Fig. 2.3.1: Nikobus

**Nikobus** (4) is one of a few *hybrid* building automation systems (in topology). Inputs (up to 256 sensors per one unit) are connected using a bus with central functional units (actor units), and end-devices like shutters/blinds, lights or heaters are connected directly to

functional units (star topology) – Fig. 2.3.2. “Nikobus bus” is physically realized by a twisted-pair and is used also as a power supply for sensors. Nikobus can take care of basic automation tasks, for example: light, HVAC and shutter/blinds control, switching of electric devices, security and alarms or energy management. Central functional units are not universal – these devices are specialized on some kind of tasks: a shutter/blinds unit, a dimming unit or a switching unit. These units have functions prepared inside, so users have to only configure them (by the computer or using the screwdriver). It is possible to incorporate RF components into the *Nikobus* system. Today the *Nikobus* is a little bit old-fashioned and not much adaptive system, but can be cost-effective for some basic, simple installations.

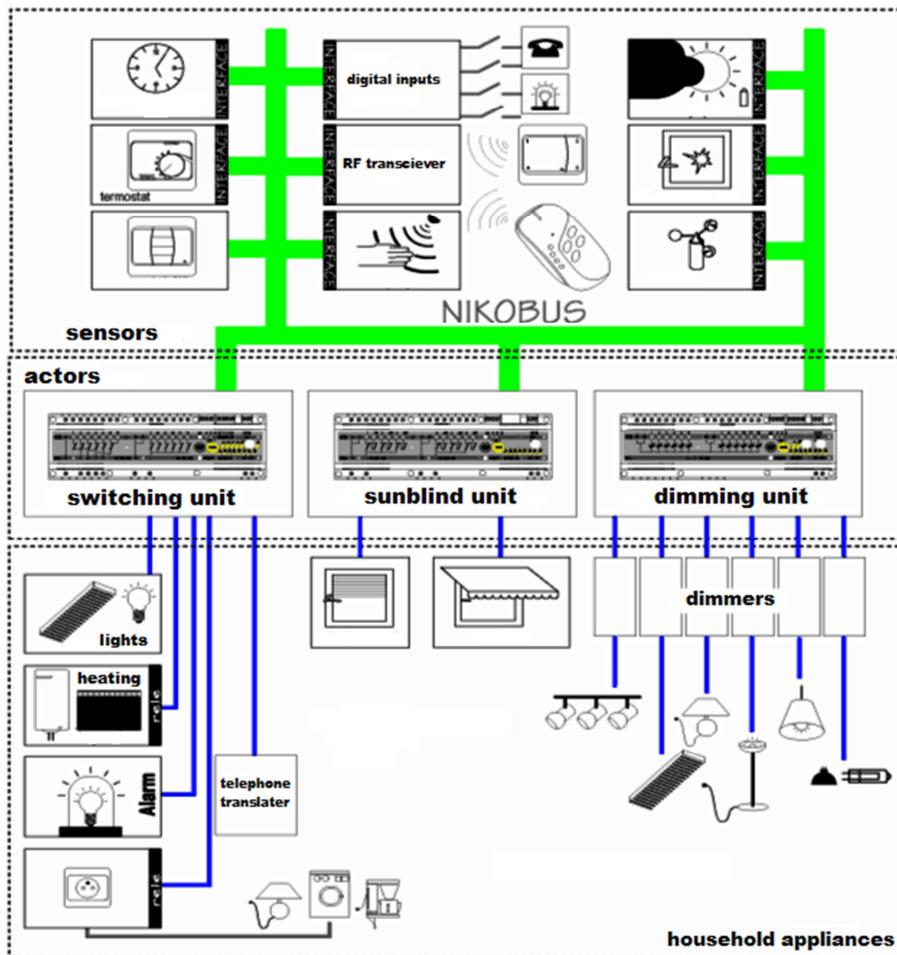


Fig. 2.3.3: Nikobus topology – translated from (5)



Fig. 2.3.4: X-Comfort

**Xcomfort** (6) is an example of relatively cheap complex wireless solution for home automation (especially suitable for building reconstruction). The system is made of fully decentralized units communicating together using a proprietary wireless protocol (with a carrier frequency 868.3 MHz). Both actors and sensors (usually using batteries) are able to route packets, so if one device sends a message for a device which is not in the range of the

transmitter, then other units in range are able to receive the message and send it to the next device until it reaches the desired device. The disadvantage is that the routing vector has to be preprogrammed to the devices (using the computer). Basic configuration of a simple installation (parameterization of predefined functions) can be done without the computer only by a screwdriver. This system is, as usual, capable to solve all the typical tasks of home automation, including basic remote visualization of the house on the computer/PDA or GSM control ability. The advantage of this product is that the producer (*Moeller/Eaton*) of the system offers other companies to share the communication protocol with them if they add some new interesting feature or device into the *Xcomfort* system. There are of course some disadvantages – for example the radiator valve needs external power supply (modern wireless valves use batteries – for example in the system made by *Conrad*).

### 3. Open standardized protocols

Protocols covered by some world-wide standards can be divided, according to the system complexity (Chapter 1.3), into two big groups: *complex and specialized protocols*.

If we apply the well-known three-level hierarchical model of automation and control system on building automation protocols, we get a division shown in Fig. 3.1 and Fig. 3.2. Industrial control system theory traditionally describes these levels this way:

- **Field level** - sensors, actuators and controllers
- **Automation (processing) level** – execution and control systems (SCADA, HMI, MES or databases)
- **Management (information) level** – enterprise applications, ERP, SAP...

KNX (Chapter 3.1.2) and LonMark (Chapter 3.1.3) are more **field** and **automation level** oriented (sensors, actors, higher control algorithms, monitoring, SCADA). BACnet (Chapter 3.1.1) is, as opposite to these two protocols, more **management** level and less **field** level oriented. As we can see from the figure Fig. 3.2, single-purpose protocols are ranked in the **field** level.

According to H. M. Newman from the Cornell University, Ithaca, USA, these three levels of building automation theory can be understood in a little bit different way (7):

*“The management level, for example, is where the majority of operator interface functions reside. Additional functions include communication with controllers, monitoring, alarm annunciation, trend logging and statistical analysis, centralized energy management functions, and communication with, or coordination of, dedicated non-HVAC systems such as fire alarm and security control. As a practical matter, most of the devices at this level are personal computer workstations.*

*The automation level is where the majority of real-time control functions are carried out. The devices tend to be general-purpose, programmable controllers.*

*The field-level contains the devices that connect to sensors and actuators. We would tend to think of field-level devices as unitary or application-specific controllers.”*

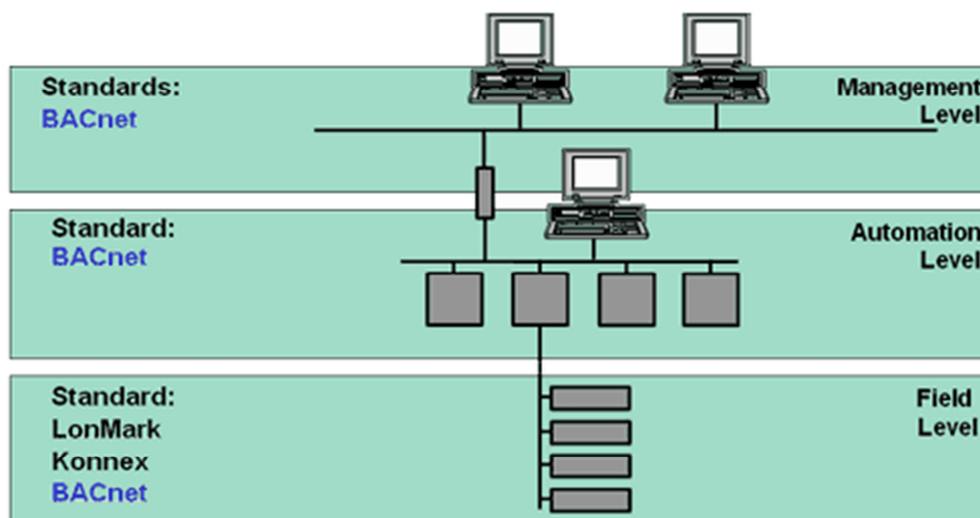


Fig. 3.1: The three level architecture and basic protocols (8)

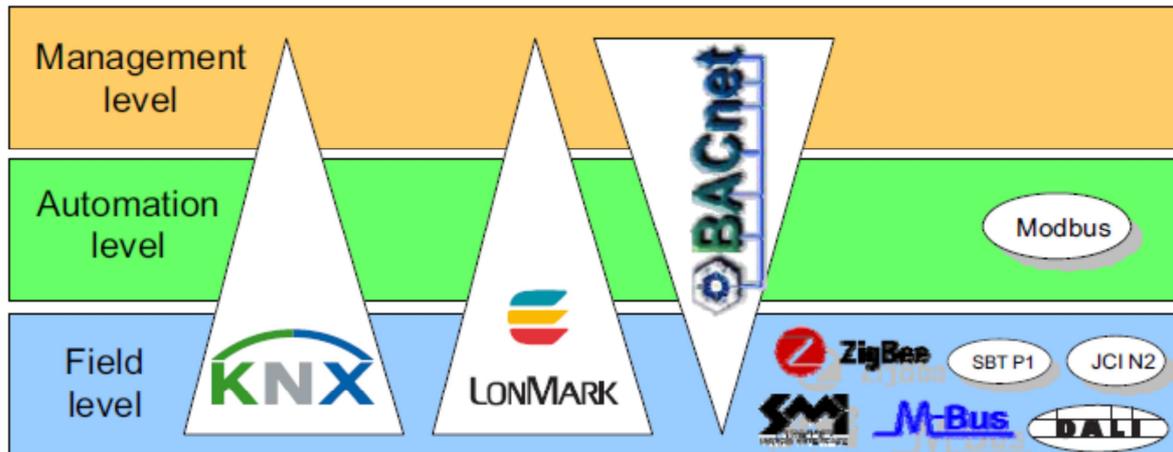


Fig. 3.2: Standards in building automation (9)

### 3.1. Complex protocols

Complex standardized protocols are the most important section of building automation from the global point of view. Today, there exist millions of their applications from light control on an airport or in a city to the room control of luxury ships all over the world. Standardized protocols are often supported by an academic research on one or more universities. The market consists of hundreds of manufacturers with thousands of devices equipped with one of these protocols. In opposite to the systems based on the closed protocols (Chapter 2), systems presented in this chapter are used not only in family houses, but they offer also a sophisticated approach for a complex building management of large buildings (e.g. airports, hospitals, office buildings).

There exists a lot of books in many languages about these famous building automation protocols. We can also find some usable books in the Czech language – for example “*Building automation: communication systems with EIB/KNX, LON and BACnet*” from the authors Hermann Merz, Thomas Hansemann and Christof Hübner (10) which practically shows basic principles of *KNX*, *LonMark* and *BACNet*. This is the reason why these three huge and important protocols will be described only briefly in this chapter.

#### 3.1.1. BacNet



Fig. 3.1.1.1: BACnet

The development of the BACnet protocol started in late eighties in the USA in the society called ASHRAE (*American Society of Heating, Refrigerating and Air-Conditioning Engineers*). From the beginning of its development, the protocol was designed only for the purposes of building automation, as its name suggests: *A data communication protocol for Building Automation and Control networks*. Today, BACnet is covered by two standards: *ISO 16484-5* and *ANSI/ASHRAE STANDARD 135*. The protocol is very different from the other standardized protocols in one aspect: it is focused on the *management and automation level*

in the above described hierarchical model (Fig. 3.1 and Fig. 3.2). This orientation is used mainly in non-residence buildings (BMS).

The BACnet is an example of a completely non-proprietary object-oriented system. That means that there are no proprietary special chips needed (in opposite to *LonTalk*). On the two lowest levels of the ISO/OSI model, the protocol is very flexible, because we can presently choose from a very incompatible group of data link/physical layers including (Fig. 3.1.1.2) *ARCNET*, *Ethernet*, *BACnet/IP*, *Point-To-Point over RS-232*, *Master-Slave/Token-Passing over RS-485*, *LonTalk* or *KNX-IP*.

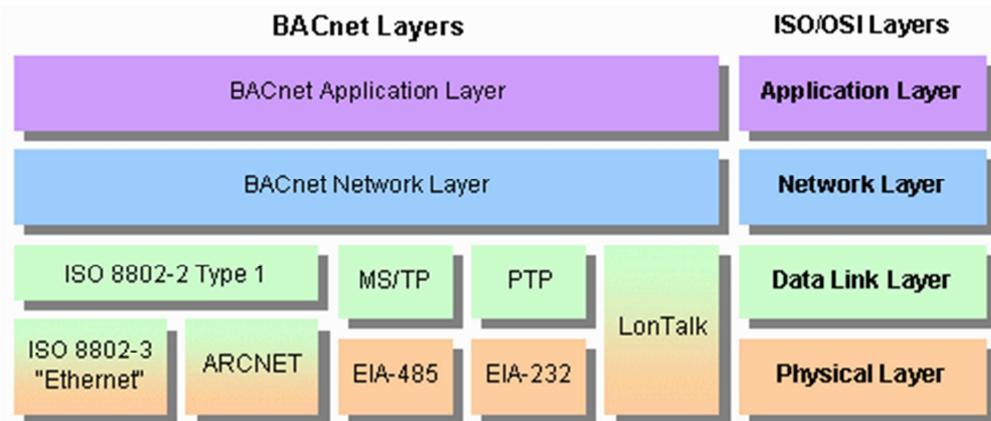


Fig. 3.1.1.2: BACnet protocol layers and equivalent ISO/OSI Layers (11)

The information exchange between devices over the BACnet is basically based on **objects and services**. Today, there exist 49 standard objects in the BACnet and every unit consists of at least one object – for example:

- Device (mandatory for every BACnet device)
- Analog input/output/value
- Binary input/output/value
- Schedule
- Calendar
- Program
- File
- Access Door
- ...

Every object has some common properties like *identifier*, *name* or *type* and some special properties dependent on the type of the object. In the protocol, there are more than one hundred properties defined. Properties can be accessed, read or written by calling special services. The services (in total more than 30) are not made only for accessing the properties, in general, they can be described as follows: *“Services are the means by which one BACnet device acquires information from another device, commands another device to perform some actions, or announces to one or more devices that some event has taken place. Each service request issued and service acknowledgment (reply) returned becomes a message packet transferred over the network from the sending to the receiving device.”* (12)

All these facts show us that besides traditional tasks of small and medium building automation (HVAC and light control), BACnet is able to take care about tasks of BMS and complex building systems (e.g. security and fire safety systems, access control systems, vertical transport systems, elevator control, maintenance or waste management). Another

important usage of the BACnet is an interconnection of incompatible building automation systems and components.

First of disadvantages of the BACnet is that there exists no common configuration tool (in opposite to KNX for example). Therefore almost every producer has to develop his own new tool for its devices. Another big problem with BACnet in recent history was that some BACnet products were not 100% compliant, so the interoperability of the devices was not as good as it should be. This problem was the reason of founding of *BACnet Testing Laboratories* (BTL).

Today, BACnet is supported in devices of more than 500 companies and also universities (February 2011: 191 vendor ids in the USA, 59 in Germany, 4 in the Czech Republic) including Siemens, Honeywell, Hyundai, Mitsubishi, ABB, WAGO, DOMAT Control systems or Teco Kolín.

**Further reading:**

(8) (9) (11) (12)

### 3.1.2. Konnexbus



Fig. 3.1.2.1: KNX (13)

*Konnexbus* (KNX) is the main building automation system (not only) in Europe. Eighty percent of companies on the market offer devices, which are capable to be connected into a KNX network. KNX is also known by its older name *EIB Instabus*. *EIB* is one of three older independent European building automation systems which were joined together in the year 2001 to create a new system called KNX. The other systems were *EHS* and *BatiBus* and a backward compatibility to these older systems is provided. Konnexbus is covered by five standards today:

- ISO/IEC 14543-3
- CSA-ISO/IEC 14543-3 (Canada)
- CENELEC EN 50090 (Europe)
- CEN EN 13321-1 (Europe)
- GB/Z 20965 (China)

The KNX system is a typical example of a fully decentralized complex home and building automation system – every device has its own “intelligence” and knows what to receive/send from/to the bus and how to process the received data.

As is usual in the case of standardized protocols, there exists an association (*KNX association* (13)) which cares about everything around the protocol – organizes a scientific conference every year, users meetings, trainings, publishes its own journal, develops and produces software for the users and manufactures... *The KNX association* has nowadays more than 220 members including companies ABB, Gira, Schneider, Wago, Hager, Siemens, Buderus, Viessman, Somphy, Bosch or Toshiba. There also exist more than 30000 installer companies in 100 countries and 150 training centers all over the world. Strong academic research is very important for the KNX – the scientific club of the Konnexbus comprises about 60 universities (for example *prof. Kastner* in TU Wien). These facts imply that customers can

choose from a very large portfolio of KNX devices (more than 7000) which are the most suitable with a function, price and design for their project. Product lines include not only actors, sensors or visualization panels, but also gateways, which are capable to interconnect the KNX system with all other building automation systems (e.g. *BACnet*, *LON*, *EnOcean*, *DALI*, *OpenTherm*).

The system can be used for automation of every type and size of a building – from family houses to office complexes and airports. A classical and basic application of the protocol is a sophisticated light control, but today it covers all main tasks in building automation: energy management; energy consumption measure; HVAC control (using KNX meteo-station); advanced shutter and jalousie control; control of electronic devices; security and safety tasks...

Similarly to other typical complex standardized protocol KNX uses more than one physical layer:

- **Twisted pair wiring** (inherited from the BatiBUS and EIB)
- **Powerline networking 230V** (inherited from the EIB and EHS)
- **Wireless KNX-RF** (carrier frequency 868.3 MHz)
- **Infrared**
- **Ethernet** (KNXnet/IP)

The most frequently used medium is a twisted pair (called TP1) with a speed of 9600 bit/s, which enables an interconnection of basic sensors and actors (**field level**). TP1 is also used as a power supply for sensors. Ethernet physical layer is used mainly for a connection of “higher” devices (**automation level**) like gateways, touch-panels, computers or PLCs with the rest of the system.

The communication itself is based on two types of addresses: *individual* and *group address*. Every device in a KNX net has a unique individual address. This address is used only for programming, configuration and monitoring of the device – not for transferring process data. For this purpose, there are dedicated group addresses. This type of address is not bundled with a device, but with a “shared” variable of a specified type (we can also call them net-variables). This variable forms the connection between sensors and actors, because it logically connects together a group of variables of the same type located in different devices. More than one device can listen to one group address and react on changes, but only one device can write data to the group address (1 producer and 1 or more consumers). So, for example, when a push-button is pressed, then it sends a packet with some group address. Then a light actor, which is listening to this group address on the bus, recognizes the change of the variable and switches the light on. Important fact is that there is no packet sent directly (a packet with destination address equals to the individual address of the device) from the push-button to the actor.

The KNX system can be configured and parameterized very easily. For this purpose, there exists a tool called *Engineering Tool Software* (ETS) which is manufacturer independent and easy to learn. So the KNX system avoids typical problems of other standardized systems (e.g. BACnet) where exists a special configuration tool for every manufacturer quite often (you cannot use a tool from one producer for devices using the same protocol/bus from another producer). The ETS tool is managed by the *KNX association* and the manufacturers only have to add a plug-in into the ETS tool. This plug-in is designed for a detailed configuration of a device and its functions. The ETS tool exists in several modifications based on skills of a user. Of course, there are still some bugs in the ETS tool. If the possibilities of the



- GB/T 20299.4-2006. Standardization Administration of China Digital Technique Application of Building and Residence Community
- SEMI E54.16. Semiconductor equipment manufacturers standard for sensor-actuator networks

To support the protocol and the standards, an association called LonMark was founded. It has more than 400 members and it maintains the standards and the interoperability of devices. Besides companies like ABB, Honeywell, Thermokon, Shneider, Somfy or WAGO, LonWorks devices are also produced by Czech companies e.g. ZPA or ATD. Today, we can say that the platform is more popular in North America than in Europe, where the leader on the market of building automation is KNX.

The LonWorks platform is focused mainly on the **automation** and **field** level of the hierarchical model. The system is based on a completely decentralized peer-to-peer net of smart nodes (Neuron chips). Every Neuron chip has a 48-bit unique serial number, which is very important for the identification in an installation. The dependency on a proprietary Neuron chip, which is the heart of every LonTalk device, is the most important disadvantage of the platform. This chip (developed with the companies Toshiba and Motorola) is based on three 8-bit processors, where two are dedicated for the protocol itself and the third one for the application:

CPU1 – Medium Access Control

CPU2 – Dedicated to processing Net variables

CPU3 – User application programmed in a language called “Neuron C”

The communication between nodes is realized by a logical linkage of network variables. Every device has a set of inputs, outputs and configuration network variables. These variables can only be of a type defined in the standard – the standardized types are called SNVT (Standard Network Variable Types). Every SNVT contains some specific properties, e.g. upper/down limit. So, for example, if a thermostat wants to read a temperature from some sensor, then a virtual logical linkage between an output variable of the sensor and an input variable of the thermostat has to be created (the variables have to be of compatible type). One output variable can have a linkage with more than one input variable. This is a little bit different from the approach of the KNX, where the interconnection of input/output variables of devices is created by a group address of some standardized type. Measured values can be sent automatically in the case of some percentage change (analog variables), state change (binary variables), in some specified time intervals (using timer) or after some explicit request from another device.

The LonTalk protocol can use one of six physical layers for transferring its data:

- Twisted pair (common)
- Power line 230V (common)
- Radio frequency
- Infrared
- Coaxial cable
- Fiber optics

One of the newer features of Lon is LonTalk over IP or so-called IP-tunneling, which is very popular. There exists a special transceiver for every physical layer which is transforming physical layer specific signals to standardized input signals for the connected Neuron chip.

The LonWorks platform can manage almost every task and wish in building and home automation (BMS) from room temperature control to security and access control systems.

But because of the cost of devices, LonWorks (similarly to KNX) is still not much suitable for “normal” family houses.

**Further reading:**

(19) (10) (20) (18)

## 3.2. Specialized and other standardized protocols

This chapter discusses a major part of important single-purpose standardized protocols in building automation (houses, discotheques, offices...). In addition to these protocols, we will also discuss the *EnOcean* protocol in this chapter, which is not single-purpose, but it is a little bit out of every category. Protocols presented here are often used by systems based on one of the complex protocols described in Chapter 3.1, or by a system with a programmable logic controller (PLC).

Presented protocols:

**DALI** – light control

**DMX512** – light and effects control for concerts, discotheques...

**EnOcean** – general I/O

**M-bus** – primary for remote reading of consumption meters

**OpenTherm** – heating control

**SMI** – drives and motors control (shutters)

### 3.2.1. DALI



Fig. 3.2.1.1: DALI (21)

#### DALI

DALI (Digital Addressable Lighting Interface) is a standardized bus and protocol for controlling (switching and dimming) of light sources in a building. This protocol is intended as a replacement of classical light control possibilities, such as the analog 1-10V dimming or relay switching. DALI is based on the DSI (Digital Signal Interface) protocol, which is also a competitor to this solution.

With this protocol, light sources can be switched and dimmed (a logarithmic dimming with the range 0.1-100 %), whole light-scenes can be defined and controlled. Another feature is a feedback from the bus and from every light, so it is possible to read-out a state of the ballast and possible errors.

One DALI line supports:

- Max 64 devices (masters + slaves)
- Max 16 groups of devices – any slave can belong to as many as 16 groups
- Max 16 light scenes

Commands (e.g. power on/off, dimming, setting of a specific brightness) can be addressed to one device, to the whole group or to be broadcasted to every slave.

Every ballast (slave) has configuration information saved in its memory, which can be edited by a master, and consists of:

- Address of a device
- Group assignment of a device

- Fade time and rate
- Light levels (Power On, Maximum, System failure)
- Values of brightness in assigned scenes

Every slave device provides this feedback information to the master:

- Lamp/brightness state (on/off, brightness)
- Lamp energy level
- Lamp condition – failure

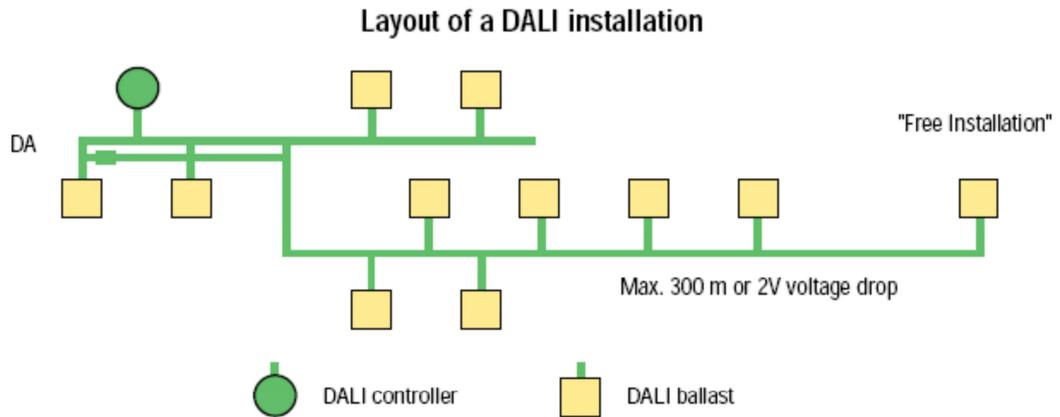


Fig. 3.2.1.2: Example of a “free” DALI system installation (22)

### Standards

The DALI interface will be included in the international standard IEC 62386. Another standard which covers this topic is IEC 60929 (the fluorescent lamp ballast) under Annex E.

### DALI Action Group

DALI AG is a working group of manufacturers and institutes which takes care of the DALI technology, defines technical principles and concepts of use, etc. DALI AG consists of more than 40 companies, for example: WAGO, ABB, Helvar, OSRAM, Philips Lighting, Tridonic ATCO or Zumbotel.

### A Master-slave system

DALI is a strictly master-slave system, where slaves are light sources and masters are PLCs, PCs or sensors and switches. There are two possibilities of the master/slave DALI system:

- 1) Single master (monomaster) system – only one master on the DALI line.
- 2) Multi-master system – more than one device operates as a DALI master on one line (sensors, switches, PLCs...). These devices must follow some “traffic rules” to avoid data collision and to maintain the correct system function.

Every master supports the monomaster system, but only newer master devices support multimaster systems.

The basic principle is, that a master gets signals from sensors, or higher control systems, and after processing this data, it creates and sends commands for slaves or groups of slaves (and waits for possible answers with state information). The possibility of the feedback is very beneficial with regard to the fact, that bulbs and lights get broken often.

### DALI system and a building management system (BMS)

- a) A completely stand-alone DALI system - Fig. 3.2.1.3. There is no connection with the BMS. All slaves are controlled by masters completely locally.

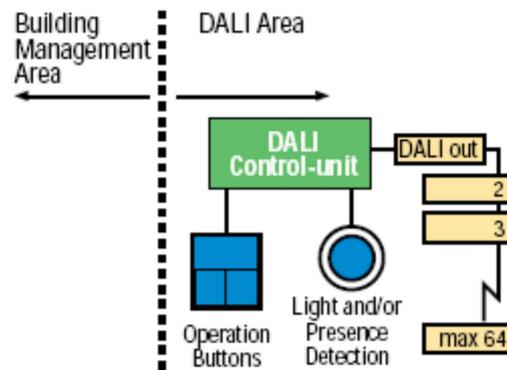


Fig. 3.2.1.3: A stand-alone DALI system (22)

- b) A stand-alone subsystem - Fig. 3.2.1.4. A DALI system is connected to the BMS, but sensors/switches of the DALI system are connected directly to the DALI control units or to the DALI line. Only information about faults, states and most important commands (central on/off) is communicated between the DALI system and the BMS. Therefore, most of the commands are issued locally (from DALI area – Fig. 3.2.1.4). The DALI system is not dependent on the BMS and it can work correctly without a BMS.

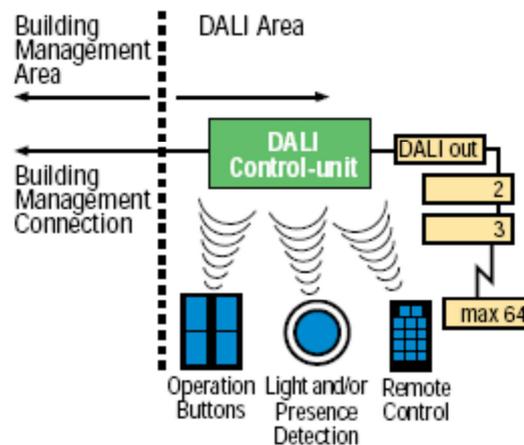


Fig. 3.2.1.4: DALI as a stand-alone subsystem of the BMS (22)

- c) A pure subsystem of BMS - Fig. 3.2.1.5. A DALI system is fully dependent on the BMS – all commands come from the BMS through the DALI/BMS gateway (= DALI master). That means that without the BMS, the DALI system does not work. From these three possibilities, this one is the most frequent solution.

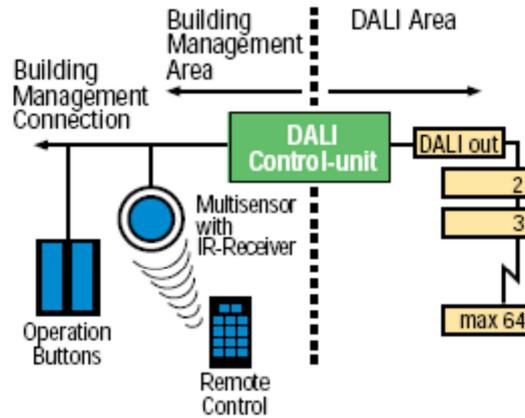


Fig. 3.2.1.5: DALI system like a pure subsystem of the BMS (22)

### Bus - parameters

A DALI line is a two-wire (polarity non-sensitive = non-polarized) line with max 300 meters length (the maximal voltage drop is 2 V), which consists of up to 64 devices. There are no special wiring requirements due to the low transmission rate, and the line also does not need to be terminated by any resistor. The topology of the system is absolutely free (line, star, tree, ring) and the bus is powered by the voltage 9.5-22.4 V (usually 16 V). Maximal transmission speed is 1200 b/s. DALI uses Manchester encoding, and together with its high SNR, this bus is very robust and “foolproof”. Maximal current input of 250 mA is allowed on the DALI line and each device may consume a maximum of 2 mA. The low level on bus is 0 V (-4.5 V to +4.5 V) and the high level is 16 V (9.5 V to 22.5 V).

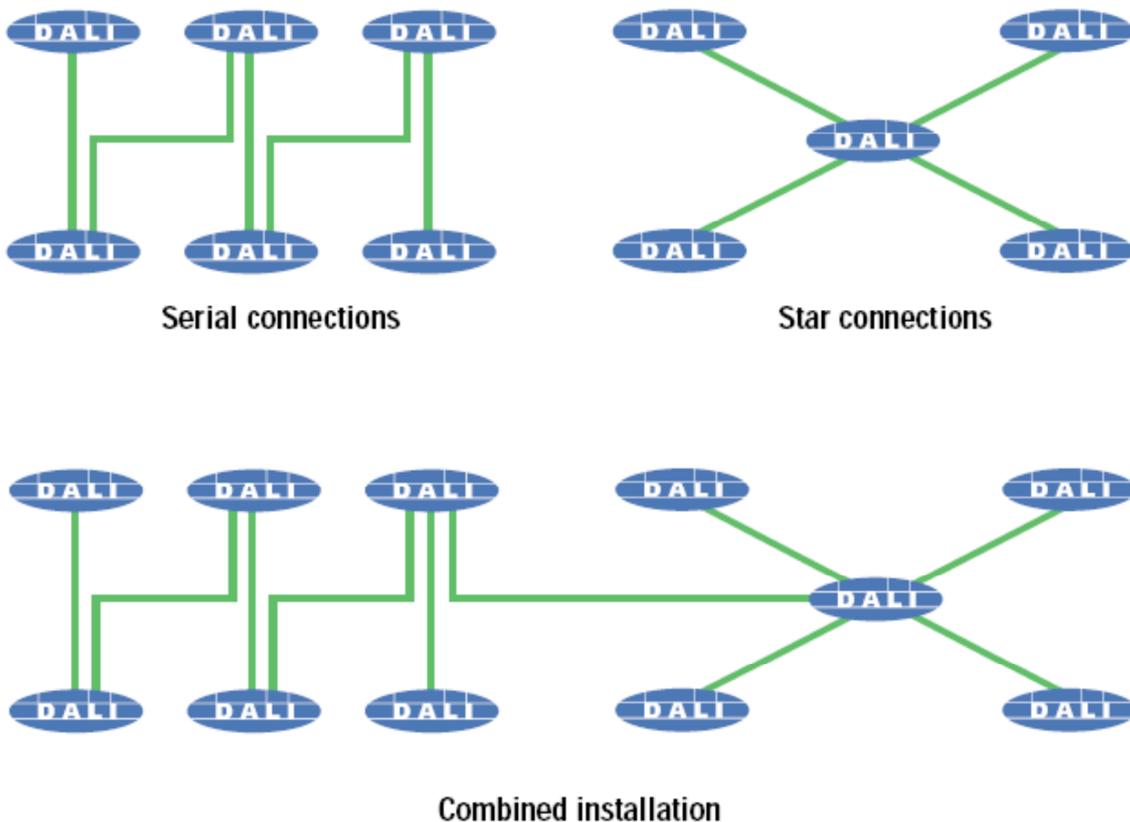


Fig. 3.2.1.6: Topology possibilities of the DALI system (22)

### Advantages

- Easy modification – no need of rewiring in case of a change
- Energy savings – 30-60 % through dimming lights in response to actual brightness of natural light and switching by occupancy sensors.
- More comfortable lighting (scenes, individual control...)
- Automatic identification of failed lamps and ballasts
- Robustness, an interference free operation

### Disadvantages

From the point of view of today's state-of-the-art, DALI is a little bit "archaic" system, which should be upgraded (new specification 3.0 should come in near future). DALI cannot be used for controlling fast changes of brightness because of its very slow communication speed 1200 b/s. Another disadvantage is a high price of devices, which are equipped with DALI. A lot of DALI master units still support only a single-master system. Maximal length of the line is 300 meters and only 64 devices can be connected. Also adding more than one sensor or switch directly into the DALI bus is not easy, because they have to support the multi-master structure. So an easier way how to solve this problem is to keep the sensors separate from DALI line and send commands through a central controller (Fig. 3.2.1.5). Moreover, the market offer of DALI sensors is not very wide.

### Interconnections

There exist a gateway to almost every important building and home automation system, bus or protocol: *Modbus*, *DMX512*, *KNX*, *BacNet (Loytec)*, *LON* or *Xcomfort/Nikobus*. Another possible option of interconnection is a modular PLC used as an Ethernet gateway for DALI or to connect DALI with EnOcean and other protocols.

### Future (autumn 2009)

Because DALI is today "a little bit archaic", but relatively often used system for controlling lights, there should come some new specification of the DALI bus soon, with new improvements. Besides a new specification, the WAGO Company is preparing (release in 2 years) a new master PLC card, which will support the multi-master DALI.

### Further reading:

(21) (22) (23) (24)

## 3.2.2. DMX512

DMX512 is a simple serial standard protocol designed especially for controlling light sources (dimmers and other light effects in theaters, discotheques...). DMX512 was developed in the year 1986 by the institute USITT (United States Institute for Theatre Technology) and updated in the year 1990. The purpose of the developing DMX512 was the same as in the case of DALI – to replace old-fashioned analog 1-10 V dimming, which has problems with interference and every device has to have its own wires connected to a controller (a star topology).

A basic principle of this protocol is very simple: a master periodically sends a packet of 512 bytes containing commands for slaves over the RS-485 line. That means that DMX512 is a master-slave system without a feedback from slaves (light sources) to the master (controller), and without an automatic error checking or correction. We call this type of communication a simplex (one-way) communication.

### A physical layer

DMX512 uses the RS-485 (EIA-485) standard for communication. So a DMX152 line (a segment) is made of twisted-pair wires and has to be terminated with a resistor with a value  $120 \Omega$  (a terminator). The maximal length of the DMX512 cable is 1,200 m (3,900 ft.). A number of segments is not limited. Standard connectors for DMX512 are with 5 or 3 pins, i.e. XLR style microphone connectors. Originally, there was used only the 5 pin connector with 2 twisted pairs (the first pair is used for DMX512 data and the second pair can be freely used by a manufacturer). Now 3 pin connectors are commonly used with only 1 twister pair reserved only for DMX data. The pin out for the 5-pin and 3-pin connector is showed in Tab. 3.2.2.1 and on Fig. 3.2.2.1 and Fig. 3.2.2.2.

Tab. 3.2.2.1: Pin out of DMX512 connectors

Pin	3-pin XLR	5-pin XLR
1	Ground/shield	
2	Data 1 - (first TP - DMX data)	
3	Data 1 + (first TP - DMX data)	
4	-	Data 2 - (second TP)
5	-	Data 2 + (second TP)

#### 5 pin XLR

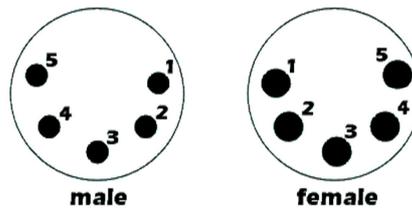


Fig. 3.2.2.1: Pin out of 5-pin XLR (25)

#### 3 pin XLR

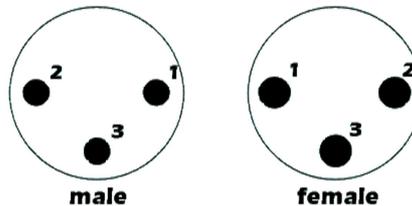


Fig. 3.2.2.2: Pin out of 3-pin XLR (26)

### Networking

DMX512 devices are connected using a daisy-chain method. Every device has a DMX512 “in” connector and often a DMX512 “out” or “thru” connector. A DMX controller has only DMX512 “out” connector. Devices at the end of the segment have to have the terminator in “out” connector placed (i.e. the resistor  $120 \Omega$  between pins 2 and 3). Some of the devices

have built-in software-operated terminator. A basic connection is showed in the Fig. 3.2.2.3. On one DMX512 line (RS-485), up to the 32 devices can be connected. When splitters and repeaters are used, then more devices can be placed in a DMX512 system.

### A Simple DMX512 Controlled Dimmer System

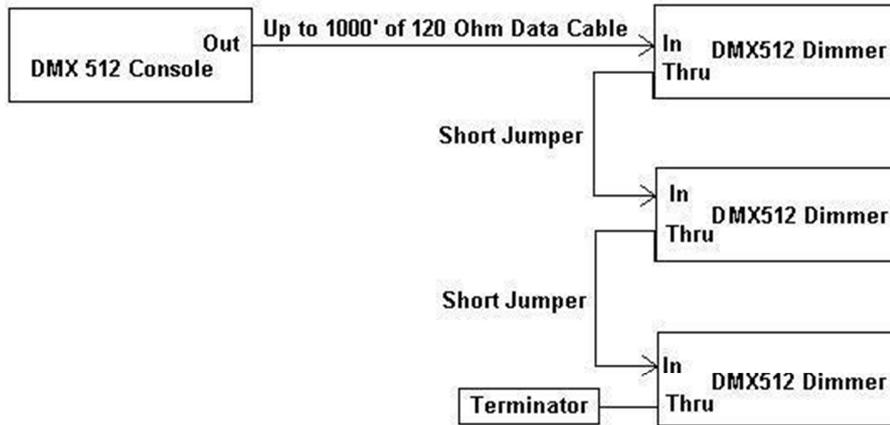


Fig. 3.2.2.3: A basic DMX512 system (27)

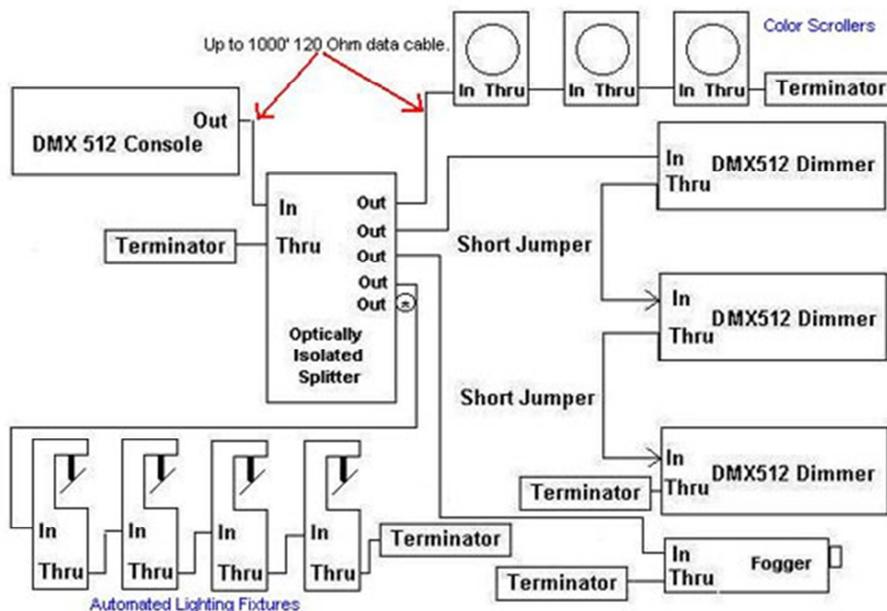


Fig. 3.2.2.4: More complex DMX512 system (27)

### Data format

Data (in the little endian format) transmission is serial and asynchronous with a speed of 250 kb/s. A DMX512's data packet consists of up to 512 data bytes carrying commands for devices without an address. There is no need of transmitting addresses through a DMX512 packet because every device (slave) knows his own base address (0-511).

From this address, every slave reads his data (one or more bytes with a meaning of a command) and reacts on them (e.g. set brightness on a specified level). There can be more

devices with the same base address. These devices will react on the same data/command, so they are controlled as a group.

A data packet (Fig. 3.2.2.5) starts with the zero level (min 88 ms on the receiver side) called *reset (break)* followed by high level called *MAB (Mark after break)* taking at least 8 ms on the receiver side. By these two first parts, the transmission is synchronized. The first data frame called *start code* takes place after *break* and *MAB*. Then up to 512 data frames are transmitted, where every frame consists of 1 start bit, 8 data bits (without parity) and 2 stop bits. Between 2 frames, there could be a timeout *MTBF (Mark Time Between Frames)* with maximal length of 1 second. At the end of a packet, a timeout *MTBP (Mark Time Between Packets)* is placed with max length of 1 second. If there appears any level (high or low) on the line for more than 1 second, then this situation is evaluated as a loss of the signal (error), which is a dangerous situation.

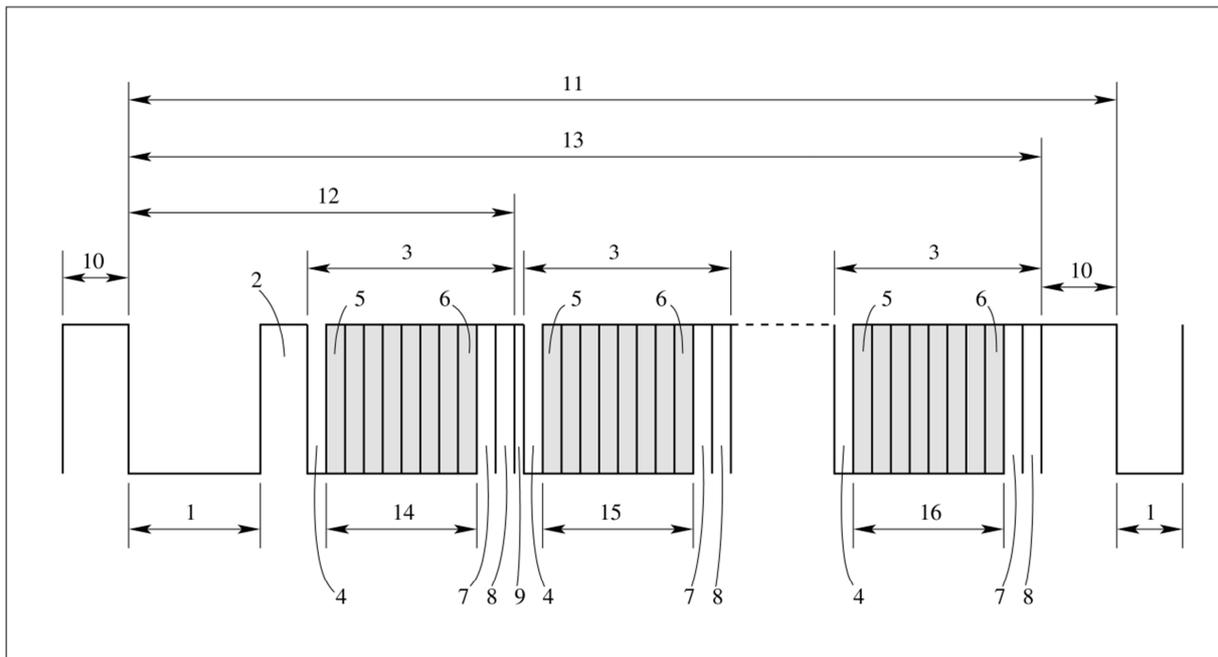


Fig. 3.2.2.5: Composition of the DMX512 data packet (28)

Tab. 3.2.2.2: Parts and lengths of the DMX512 data packet (28)

		Transmitter		Receiver	
		min	max	min	max
1.	Break - reset	92 μs	-	88 μs	-
2.	Mark after Break (MAB)	12 μs	1 s	8 μs	1 s
3.	Slot Time	-	-	-	-
4.	Start bit	-	-	-	-
5.	Least Significant Data Bit (LSB)	-	-	-	-
6.	Most Significant Data Bit (MSB)	-	-	-	-
7.	Stop Bit	-	-	-	-
8.	Stop bit	-	-	-	-
9.	Mark Time between Slots	0 s	1 s	0 s	1 s
10.	Mark before Break (MBB)	0 s	1 s	0 s	1 s
11.	Break to Break time	1203 μs	1 s	1196 μs	1,25 s
12.	Reset Sequence (Break, MAB, Start Code)	-	-	-	-
13.	DMX512 Packet	1204 μs	1 s	1196 μs	1,25 s
14.	Start Code (SLOT 0 Data)	-	-	-	-
15.	Slot 1 Data	-	-	-	-
16.	Slot n Data ( Max. 512)	-	-	-	-

With the transmission speed 250 kb/s, every bit lasts 4 μs, so data frame's length (11 bits) is 44 μs. If we have 512 data frames, then a data packet with a *break* and *MAB* lasts:

$$\text{Break} + \text{MAB} + (1 + 512) * \text{frame} = \text{min } 88 + \text{min } 8 + 513 * 44 = \text{min } 22668 \mu\text{s}$$

So the highest frequency of the data packet sending is about 44,12 Hz = 44 updates/second.

### Application of the protocol

- Dimmers
- Mirror positioning
- Gobo positioning
- Gelstring positioning
- Shutter positioning
- Focus motor positioning
- Fog and smoke machines
- ...

Because there is no error checking, DMX cannot be used for pyrotechnics or laser control and other safety-problematical applications.

### Standard

In the year 2004, DMX512 standardized by ANSI under the standard "E1.11-2004, USITT DMX512-A".

### Remote Device Management (29)

The disadvantage of the simplex communication of DMX512 was eliminated in the year 2006 (approved by ANSI) with an enhancement of the DMX512 protocol called *Remote Device Management* (RDM). RDM allow bi-directional communication between a light source and a controller over standard DMX512 line. Devices that support RDM can be placed on the same line with devices that do not support it – operation of these devices is not disturbed (RDM has a different start code). So this enhancement allows DMX512 to manage and configure devices or process a status monitoring. RDM has three basic communication types – *unicast*, *broadcast* and *discovery*.

### Art-Net (30)

Art-Net is an Ethernet based protocol for transporting DMX512 data. This protocol is also supported by over 20 manufacturers and also by the *Wireshark software*.

### Further reading:

(25) (26) (27) (28) (29) (31) (30) (32) (33) (34) (35) (36) (37)

## 3.2.3. EnOcean



Fig. 3.2.3.1: EnOcean (38)

EnOcean (38) is an example of the modern popular completely wireless “green” technology. EnOcean is a system of a distributed intelligence based on a communication between wireless intelligent sensors and actuators (every node has its own processor/microcontroller, so it can make its own decisions). This system is often used only as an I/O subsystem (a building fieldbus) for a larger building automation system.

### Advantages of an EnOcean technology:

- a) Wireless communication
- b) Smart energy management with a low consumption of energy
- c) Innovative energy harvesting
- d) A lot of service-free/maintenance-free battery-free nodes
- e) Reduced fire risk, inductive fields and low electromagnetic pollution

**The disadvantages** of the system are on the first place prices of devices equipped with this protocol, relatively small group of offered products, and especially in the Czech Republic, the low availability of the products on the market.

### An alliance

The EnOcean Alliance was founded in the year 2001 as a spin-off company of *Siemens AG*, who created this technology. EnOcean Alliance contains nowadays more than 100 integration partners (Siemens, WAGO, Osram, Steute, Thermokon, VIPA, Phoenix Contact...), which offer more than 300 products with the EnOcean technology. There exist a lot of

interfaces and gateways for its interconnection with buses such as KNX, LON or with TCP/IP and communication modules into PLC.

### Nodes

There are 3 types of nodes – transmit only (wireless sensors), receive only (actuators, gateways) and bidirectional (wireless sensors, wireless actuators, gateways, repeaters) nodes. Nodes can be harvester powered (sensors, actuators), battery powered (sensors, actuators) or line powered (actuators, sensors, gateways, repeaters). A lot of modules (mainly sensors) are service-free and self-powered (without battery), so they use some kind of energy from the surrounding environment with an optimal energy consumption management. Every self-powered sensor contains an energy harvester and converter.

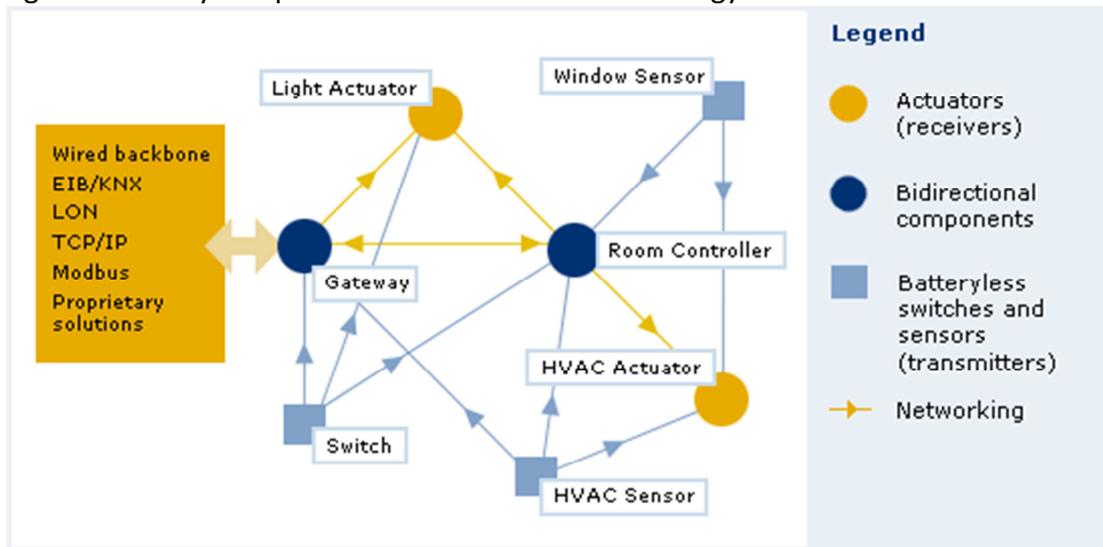


Fig. 3.2.3.2: Example of a control system based on the EnOcean (38)

Types of energy harvesters for battery-less nodes:

- Motion converter* – gains energy from switching operations/button motions (piezoelectric crystal deformation). Service-free for 50,000 switching cycles.
- Solar cells* – used for example for a window monitoring
- Thermal converter* – heat dissipation is used as an energy source (Peltier element)
- Rotation converter* – used for example for monitoring of car wheels
- Vibration converter* – used in industry

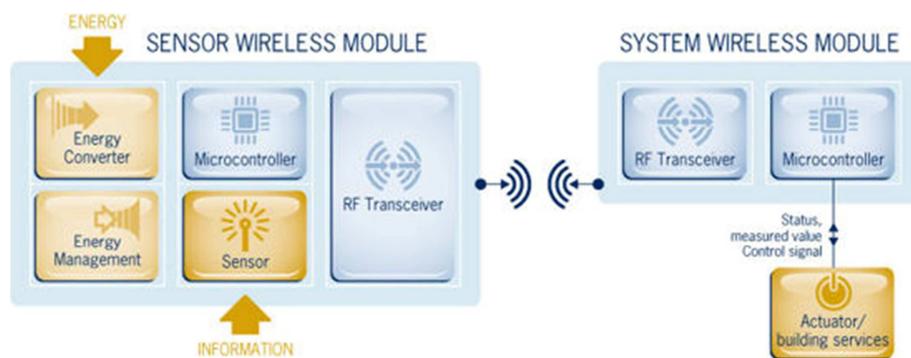


Fig. 3.2.3.3: Principal structure of the EnOcean sensor node (38)

### Wireless technology

EnOcean modules use two frequencies: 868.3 MHz for Europe and 315 MHz for North America and other non-FCC countries.

Technical parameters (Europe):

- a) License-free 868.3 MHz with 1% duty cycle with ASK modulation (for example KNX RF implementation uses 868.3 MHz with 1% and FSK modulation)
- b) Data packets consist of 14 bytes, where 4 bytes are data (39). For example, to push a button means to send 3 packets. These 3 packets are sent at pseudo-random intervals reducing the possibility of packet collisions.
- c) Range: 30 m inside buildings and 300 meters in free field. For range extension, repeaters can be used.
- d) Signal duration is approximately 1 ms
- e) Speed: 125 kbit/s
- f) No interference with WLAN or other wireless systems
- g) Every device has a 32-bit unique serial identification number
- h) 50  $\mu$ Ws is a minimal energy that is needed for sending packets

### Application of the EnOcean technology

The EnOcean technology is mainly used in a home and building automation, where it is an ideal solution for a building reconstruction. The technology is able to solve some “classical” tasks of the building control:

- a) Lighting
- b) Shading
- c) Presence detection
- d) Window monitoring
- e) Room temperature sensing
- f) Measured data acquisition

But buildings are not the only possible usage of this system. The EnOcean technology can be well used for example in cars, where the combination of the wireless communication and of a battery-free solution harvesting the energy from a rotation motion is ideal for wheels monitoring. Another interesting application is an audience voting system developed by the EnOcean's UK distributor (40). In this system, each member of the audience receives a four-button remote control (Fig. 3.2.3.4) with an EnOcean transmitter which has a unique identification number. The signals from the controllers are then decoded by a receiver connected to a PC. Other examples of the usage of EnOcean are listed in (41).

### Further reading:

(38) (39) (40) (42)



Fig. 3.2.3.4: EnOcean Voting System (40)

### 3.2.4. M-Bus (Meter-Bus)



Fig. 3.2.4.1: M-Bus (43)

M-Bus is the first communication protocol primarily designed for remote reading of consumption meters (water, heat, electricity or gas). This protocol can be used in industry, as well as in building automation. Meter-Bus was developed by Dr. Horst Ziegler of the University of Paderborn in cooperation with Texas Instruments Deutschland GmbH and Techem GmbH (actual version is 4.8). The usage of the M-bus and the requirements for the protocol are very different from the other systems presented here: the protocol offers fail-safe data collecting from consumption meters on a long distance with a very low speed, which is the effect of the robust, disturbance non-sensitive communication (the same approach as the DALI bus (3.2.1)).

Today, M-Bus is strongly supported by traditional automation companies like Siemens (DESIGO), WAGO or ABB, and also by a lot of smaller companies, which are, besides sensors, usually selling gateways for every important communication bus.

#### Wired M-Bus

The basic wired version of the M-Bus has a physical, link (both covered by the standard EN 13757-2) and application layer (EN 13757-3) implemented. This basic version can have up to 250 slaves connected to the master by the bus. If this amount is not enough, then the network layer can be implemented. M-Bus is a serial bus with 8-bit asynchronous half-duplex communication equipped by two basic communication mechanisms: request/response and acknowledged communication.

The communication itself is based on the master-slave model where the master unit is also called a Repeater. In the wired version, the master unit starts the communication and asks slaves for measurements. The advantage is that the slaves can be powered directly by the bus instead of batteries. The wired bus is a physically shielded twisted-pair and the distance between two devices can be up to 1000 meters (with max speed 300 bauds). The highest speed of the communication through the wired M-Bus is 9600 bauds when the distance between two devices is no longer than 350 meters.

The exchange of data is similar to the OpenTherm technology (3.2.5): The master is changing the voltage when it sends the data ( $\log.1 = 36\text{ V}$  and  $\log.0 = 24\text{ V}$ ) and slaves are responding by the change of the current on the bus ( $\log.1 = 1.5\text{ mA}$  and  $\log.0$  is represented by consumption higher by 11 to 20 mA than in the case of  $\log.1$ ). There exist four types of frames which are traveling between master and slaves: Single Character, Short Frame, Control Frame and Long Frame.

There exist also a lot of converters and gateways which are interconnecting M-Bus with other protocols and buses, e.g. M-Bus to Modbus, M-Bus to IP/Ethernet or M-Bus to KNX. In Fig. 3.2.4.2 and Fig. 3.2.4.3 an example of an Ethernet gateway is depicted, equipped by a webserver, SNMP, XML, Syslog, Modbus/TCP and automatic graph generating.

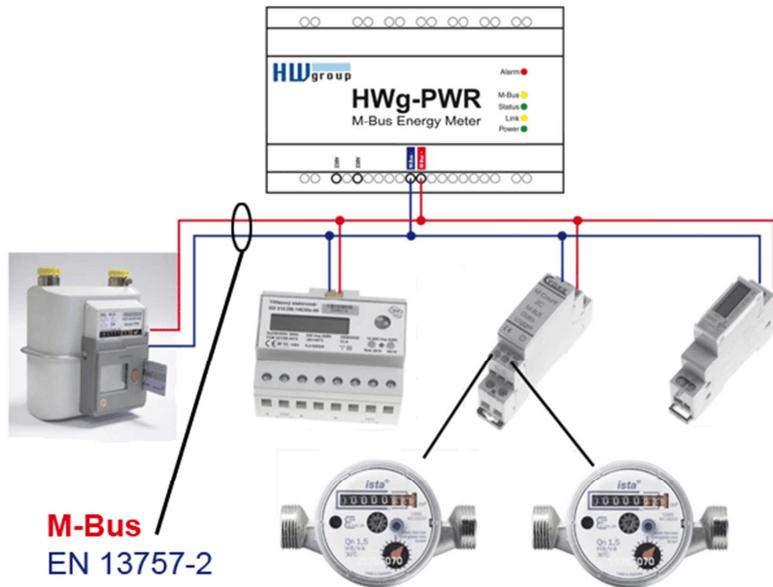


Fig. 3.2.4.2: An example of an Ethernet gateway (44)

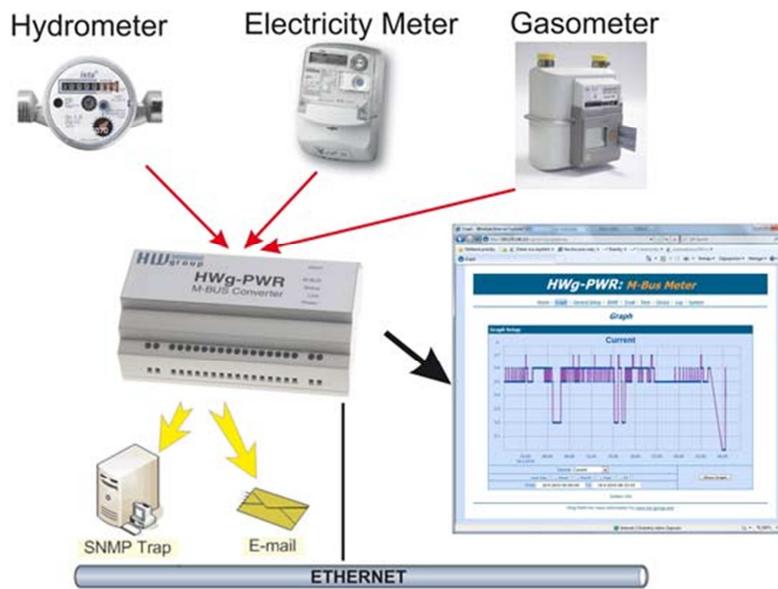


Fig. 3.2.4.3: Connection of the Ethernet gateway (44)

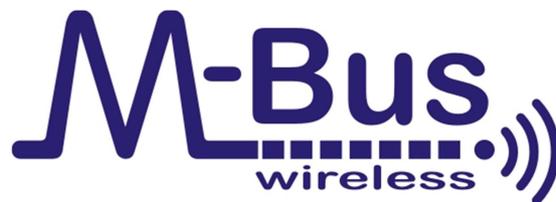


Fig. 3.2.4.4: Wireless M-Bus (45)

### Wireless M-Bus

The newest approach (year 2007) in the protocol is Wireless M-Bus (standard EN 13757-4) developed by a Norwegian company *Radiocrafts AS*. This new feature uses a frequency 868

MHz with a max speed from 4.8 kb/s to 100 kb/s. Wireless M-Bus has the same application layer as the wired version but it has its own physical and link layer.

### Wireless M-Bus modules

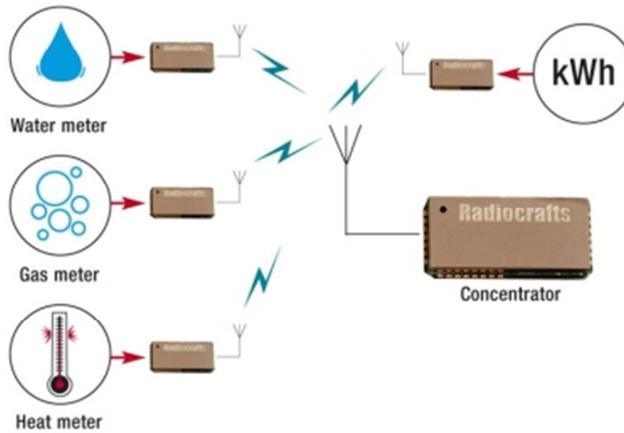


Fig. 3.2.4.5: Radiocrafts AS Wireless M-Bus (45)

One of the most important differences between the wired and wireless version is, that in the wireless approach, the master is not initiating the communication. The master is now only a collector or a concentrator of measurements, and the meters themselves start periodically the communication and send him the measurements. The reason for this approach is the need of low energy consumption, because the devices are usually powered by batteries. The device is usually in the sleep mode, where it saves the battery (the battery then can power the device up to 20 years). After the predefined interval, a timer wakes the device up and it sends the measurement to the master.

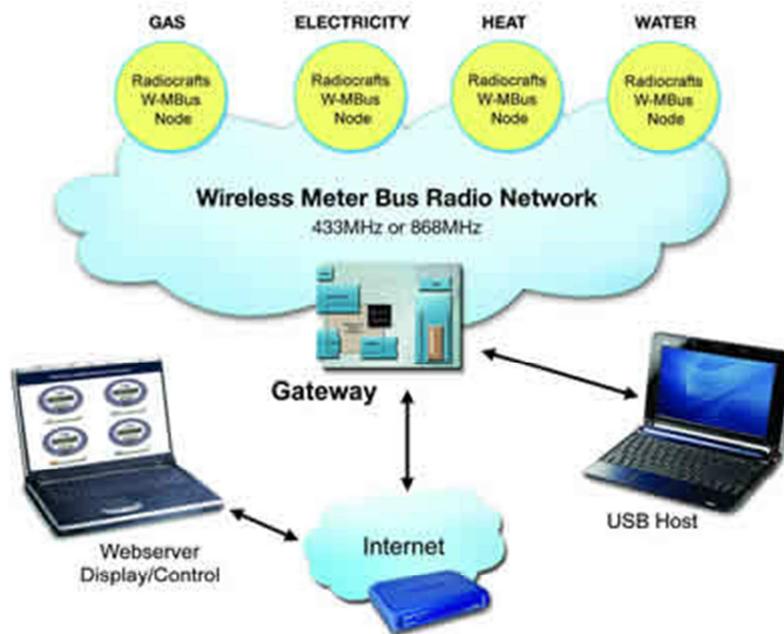


Fig. 3.2.4.6: Basic principles of the Wireless M-Bus (45)

This solution can be used for a lot of interesting applications: for example for a realization of a reading of water consumption meters in houses from a car – the car is going through the street and wirelessly reads the meters, so there is no need of personal visit to the house.

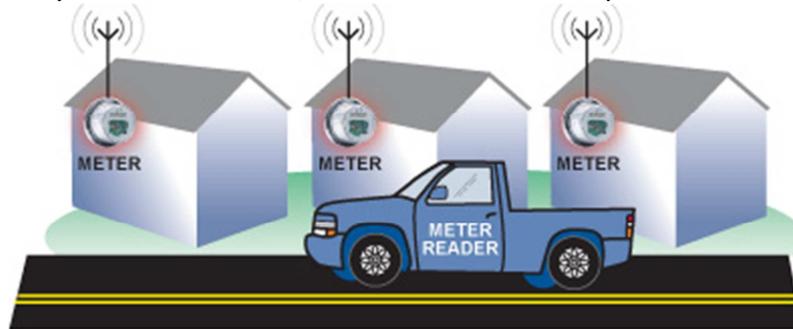


Fig. 3.2.4.7: An example of the usage of the Wireless M-Bus (45)

#### Further reading:

(43) (46) (47) (48) (49) (45)

### 3.2.5. OpenTherm



Fig. 3.2.5.1: OpenTherm (50)

OpenTherm (50) is a manufacturer-independent protocol for communication in HVAC and heating systems. OpenTherm products are typically room thermostats, heating control units, boilers, air heaters, programmers, weather compensators or cascade controllers. It is an open standard, which specification is available to all members (and also for non-members) of the OpenTherm Association. The OpenTherm protocol was for the first time specified for the first time in the year 1996 by the company Honeywell. Later, this company sold the specification for one British Pound. The reason for creating this protocol was a need to have a standardized and simple-to-use communication between thermostats and boilers.

The non-profit OpenTherm Association has more than 40 member companies today (year 2008 – 42 companies), for example: Viessmann, Hager Controls SAS, Honeywell, Siemens Building Technologies...

#### Communication

The basic principle is that a thermostat or a room-unit calculates a heating demand and transmits it to a boiler or a heater. Then the boiler answers with his status and with other information for diagnostics. OpenTherm uses 2-wire non-polarity sensitive physical layer, where the cable is used not only for the communication between master and slave, but also for powering the master. The maximum length of the wiring is 50 meters. In communication, the master changes voltage level when transmitting data, and the slave changes current when answering to the master. Used levels of the current and the voltage are:

Current:

- high signal 17....23 mA
- low signal 5....9 mA

- Threshold 11.5....14.5 mA
- Voltage:
- high signal 15....18 V
  - low signal .....7 V max
  - Threshold 9.5....12.5 V
- Rise and fall times: 50  $\mu$ s max

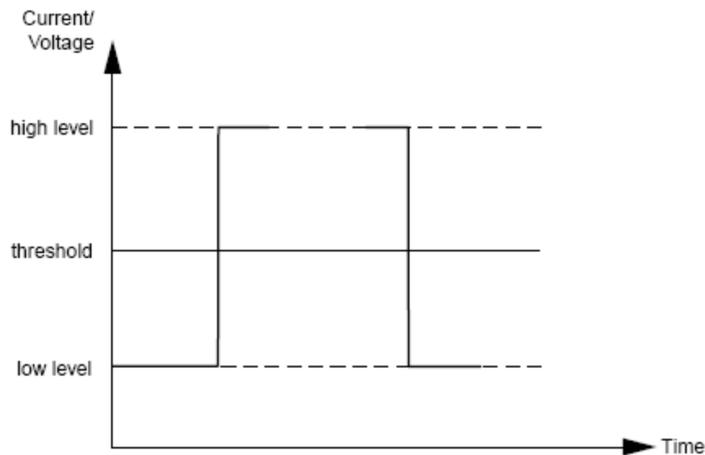


Fig. 3.2.5.2: Current/voltage levels in an OpenTherm communication (51)

OpenTherm is not exactly a bus system, because the communication can be only “point-to-point” or “multipoint-to-point”:

- a) An older type of the communication is “point-to-point” (Fig. 3.2.5.3) which means 1 master (controller/thermostat) communicating with 1 slave (boiler).

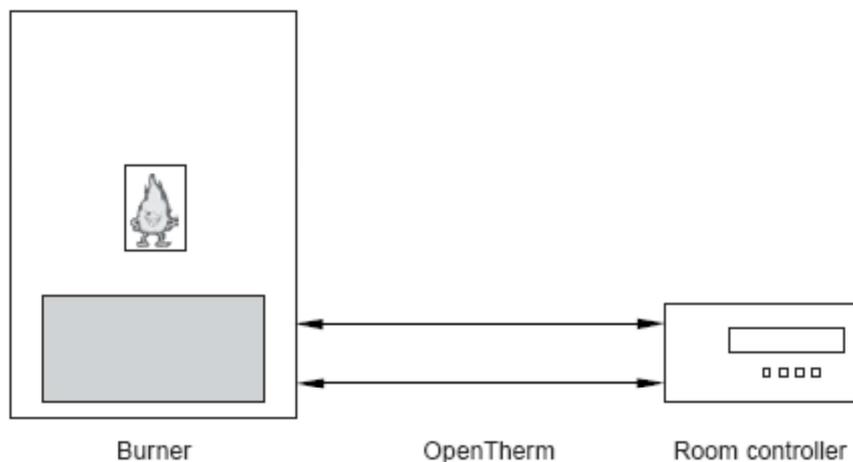


Fig. 3.2.5.3: Point-to-point communication (51)

- b) In June 2008, a new specification version 3.0 came, which brought two innovations: “Smart power” and “multi-point to point” (Fig. 3.2.5.4 and Fig. 3.2.5.5) communication. This means one slave can be controlled by one or more masters. The heart of this new feature is a special device called “Gateway”, which is placed between masters and a slave (Fig. 3.2.5.4).

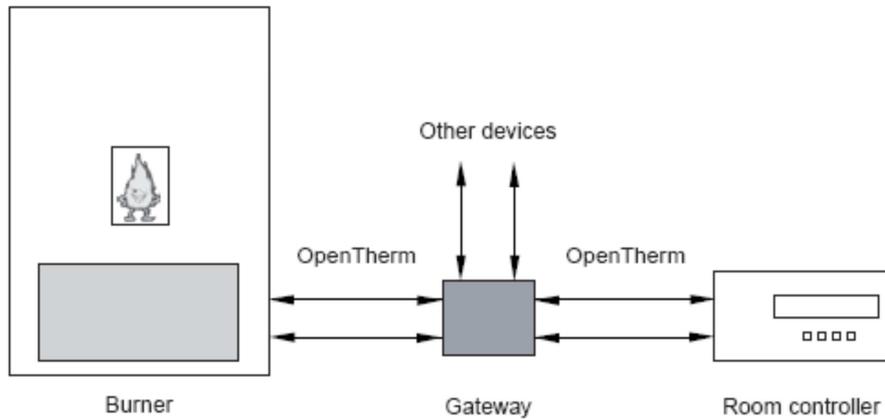


Fig. 3.2.5.4: Multipoint-to-point communication (51)



Fig. 3.2.5.5: Multi-point to point OpenTherm communication (52)

A master (a thermostat) is in an OpenTherm system powered from a slave (a boiler). With the new feature from the specification 3.0 called “Smart power”, power consumption can be smartly managed by asking the slave for providing more or less power according to the needs of the master (e.g. backlight on). The minimal available power is 35 mW. The other two power levels available to the master are: 136 mW (medium power) and 255 mW (high power).

**There are two implementations of the OpenTherm protocol:**

- a) An older analog implementation called **OpenTherm/light (OT/-)**, which offers a very limited functionality. A master generates a PWM signal which is representing the desired temperature of the boiler water (a set point). In the case of short circuiting of the OpenTherm connection, the boiler starts to heat the water. The current signal from the boiler indicates its status: “error” and “no error”. OT/- is not a much usable implementation and due to its restrictions and limited possibilities is not used often.
- b) The full implementation of the protocol is called **OpenTherm/plus (OT/+)**. OT/+ is, in contrast to the analog OT/-, a digital implementation. That means that no PWM signal is generated and communication is realized by packet sending. Devices which are supporting both implementations detect automatically after reset which protocol is used by a partner-side. The OT/+ protocol uses Manchester encoded

(synchronization can be done every bit) unidirectional serial communication with a transmission rate of 1.0 kHz and data frame (Fig. 3.2.5.6) consisting of 34 bits (a start-bit, 32 data bits, a stop-bit). The data bits consist of:

- 1 parity-bit (even)
- 3 message type bits = used for an identification of the content and meaning of the frame (read-data, write-data, data-invalid...)
- 4 spare bits
- 8 data-ID bits, which identify uniquely the transmitted values. These bits are called OpenTherm-ID. There are 256 OT-ID available: 128 of them are reserved for OEM use, and the other 90 are functionally specified.
- 16 data-Value bits

In normal operation, a master sends a frame every second. Answer from a slave is expected in 20ms – 800 ms. After a slave answers, there is a 100 ms pause.

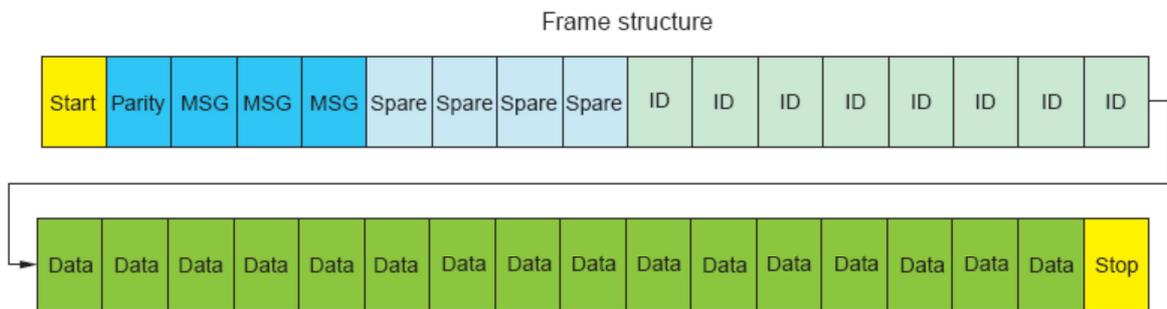


Fig. 3.2.5.6: OpenTherm frame structure (51)

There exist only few standard interfaces for interconnection of the OpenTherm system with BACnet IP, LON, Modbus, WWW (Fig. 3.2.5.7) or KNX (Fig. 3.2.5.8).

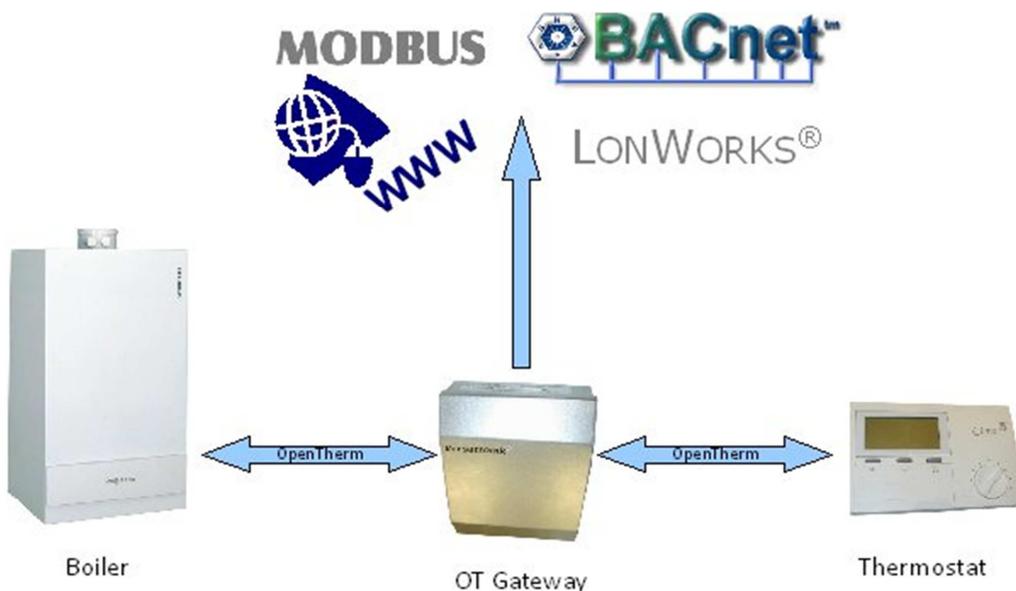


Fig. 3.2.5.7: A gateway to other systems (53)



Fig. 3.2.5.8: KNX and OpenTherm (54)

#### Advantages of OpenTherm:

- OpenTherm can use existing 2-wires cables, which were used for example for an old thermostat
- No polarity sensitive wiring
- No use of batteries – a master powered from a slave
- An open standard – devices from different companies can work together
- Offers comfortable controlling of a HVAC application (e.g. shows more details about state of the boiler on a display of the thermostat).

#### Further reading:

(50) (51) (52) (53) (54) (55) (56) (57)

### 3.2.6. SMI (Standard Motor Interface)



Fig. 3.2.6.1: SMI (58)

Standard Motor Interface (SMI) is one of the newest specialized fieldbuses for home and building automation. The history of this protocol started in the year 2001, when the SMI Working Group was founded. The reason for developing this bus was dissatisfaction with a standard way of controlling drives in a building:

- No intelligent operating (not possible to move to an exact position...)
- No feedback from drives about their position and about their state

- Very complex cabling
- SMI tries to find solutions for these problems.

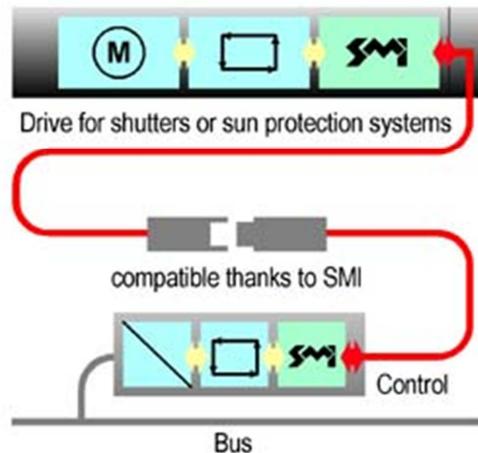


Fig. 3.2.6.2: A basic SMI concept (58)

Standard Motor Interface is a typical monomaster-multislave system where one master is able to control up to 16 slaves (drives). Slaves can be addressed individually or together (group addressing or broadcast messages) – the allocation of one address per one drive is possible, but is not necessary if all the drives are to be controlled together.

There are two types of commands transferred through SMI – standard commands and manufacturer-specific commands. Commands offer, besides basic actions (e.g. move up, move down), more complicated and complex actions, for example precise movement to intermediate position or request for diagnostic/feedback messages (possible responses: a motor is rotating, direction of the rotation, a motor defect, an actual position). Diagnostic response is also possible to be read during the drive rotation.

### SMI Applications

SMI is specialized only in the intelligent digital operation of drives in buildings. The typical applications are:

- roller shutters
- sun protection systems
- venetian blind drives
- awning
- solar-operated systems
- swimming pool covers
- winter gardens
- smoke and fire protection
- ...

SMI drives can be operated by control units, which can be completely stand-alone (Fig. 3.2.6.3) or are connected to higher building automation systems like KNX or LON for example (Fig. 3.2.6.4).

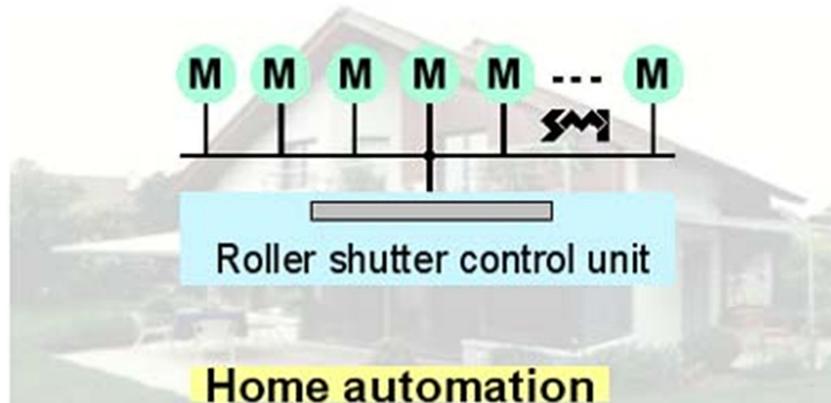


Fig. 3.2.6.3: Home automation and SMI (58)



Fig. 3.2.6.4: SMI and BMS (58)

### Communication

SMI uses a 5-wire cable (Fig. 3.2.6.5) for the interconnection of the units: three wires are for the power supply (L, N, PE) and the last two wires are reserved for the data transfer. Maximal length of the cable is up to 350 m with free topology. Data transfer is bi-directional with a speed 2400 Bit/s.

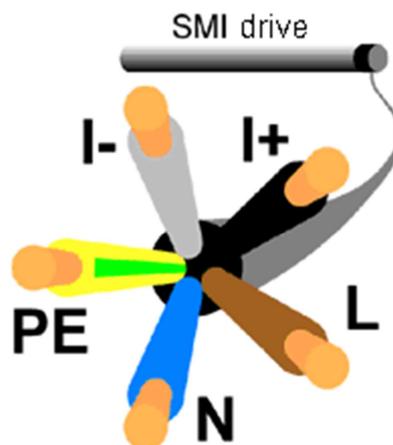


Fig. 3.2.6.5: SMI cable (58)

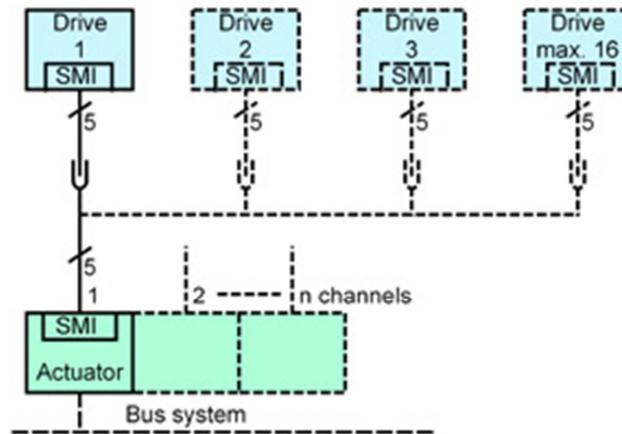


Fig. 3.2.6.6: SMI cabling and basic concept (58)

### SMI Group

SMI Group is today (autumn 2009) much smaller than for example the OpenTherm Association, DALI Group or EnOcean Alliance, but is still growing. Group consists of 7 base members (Becker Antriebe GmbH, elero GmbH, Griesser Electronic AG, SELVE GmbH & Co. KG and Alcatel-Lucent Deutschland AG), which own the brand and finance the SMI development. Then there are 11 SMI Partners (license holders – for example ABB, WAGO, Insta...), which are producing and selling SMI compatible drives and control units. The last part of the SMI Group is SMI Supporters, which are supporting the development of the SMI (4 companies).

One of interesting applications of the SMI protocol is a sun protection system in the new opera building in Oslo based on SMI drives.

### Further reading:

(58) (59)

## 4. Other buses and systems used in home automation

There is a lot of other buses, protocols and systems that are used in the building automation and in BMS but they are not much wide spread or their usage in buildings is not a typical application – some examples include:

- Well known open standardized protocol **CAN** (Control Area Network). CAN has a lot of applications in car or train automation, industrial automation and also some in the building automation (*Moniko Tower in Zürich*)....
- BMS **Local Control Network** (LCN) - <http://www.lcn-america.com/>
- **Zumbotel Luxmate** – a system for light management, which is using DSI (Digital Signal Interface) and DALI [www.luxmate.com](http://www.luxmate.com)
- **MP-Bus** – a bus designed by the company Belimo for a master/slave communication with valves
- **I<sup>2</sup>C** – protocol used typically for sensors in robotics
- **ZigBee** – universal wireless mesh network
- **CEBus** - Consumer Electronics Bus
- **C-Bus**
- **oBIX** - Open Building Information Exchange is a standard for Web Services-based interfaces to building control systems (60)
- **DyNet**
- **UPB** – Universal Powerline Bus
- **X10** – a open standard for a communication of home automation devices
- **INSTEON**
- ...

## 5. Bibliography

1. **Žabka, Miroslav.** ABB: Inteligentní elektroinstalace Ego-n. *Elektrika.cz*. [Online] [Cited: February 18, 2011.] <http://elektrika.cz/data/clanky/abb-inteligentni-elektroinstalace-ego-nae/view>.
2. **ABB s.r.o., Elektro-Praga.** Ego-n. ABB. [Online] [Cited: February 18, 2011.] <http://www117.abb.com/index.asp?thema=9745>.
3. **Elko EP.** iNels. *iNels*. [Online] [Cited: February 18, 2011.] [www.inels.cz](http://www.inels.cz).
4. **Bothe, Robert.** Inteligentní elektroinstalace budov - systém Nikobus, Uživatelský manuál v.1.0. *Eaton elektrotechnika*. [Online] [Cited: February 18, 2011.] <http://www.eatonelektrotechnika.cz/pdf/manual%20nikobus.pdf>.
5. **TES.** Chytrý dům. *Testování energetických systémů*. [Online] [Cited: April 04, 2011.] <http://www.tesnet.cz/cs/produkty-chytrydum.php>.
6. **Eaton.** Xcomfort. *Xcomfort*. [Online] [Cited: February 21, 2011.] [www.xcomfort.cz](http://www.xcomfort.cz), [www.xcomfort.com](http://www.xcomfort.com).
7. **NEWMAN, H. MICHAEL.** BACnet Goes to Europe . . . and Beyond. *BACnet.org*. [Online] 1998. [Cited: February 24, 2011.] <http://www.bacnet.org/Bibliography/HPAC-10-98.html>.
8. **BACnet Interest Group Europe.** BACnet Basics. *BACnet Interest Group Europe*. [Online] [Cited: February 23, 2011.] <http://www.big-eu.org/eng/bacnet/basics.php>.
9. **Wolfgang Granzer, Wolfgang Kastner, Christian Reinisch.** *Gateway-free Integration of BACnet and KNX using Multi-Protocol Devices*. Vienna : Vienna University of Technology, Automation Systems Group, 2008.
10. **Hermann Merz, Thomas Hansemann, Christof Hübner.** *Automatizované systémy budov*. Praha : Grada Publishing, a.s., 2009. 978-80-247-2367-9.
11. **ICT TU Wien.** About BACnet. *BACnet*. [Online] [Cited: February 25, 2011.] <http://www.ict.tuwien.ac.at/bacnet/>.
12. **Swan, Bill.** The Language of BACnet-Objects, Properties and Services. *BACnet.org*. [Online] [Cited: February 25, 2011.] <http://www.bacnet.org/Bibliography/ES-7-96/ES-7-96.htm>.
13. **KNX Association.** KNX. *KNX*. [Online] [Cited: February 22, 2011.]
14. **KNX.** Central Control of the Public Lighting System with KNX. *KNX.org*. [Online] [Cited: February 23, 2011.] <http://www.knx.org/fileadmin/downloads/02%20-%20What%20is%20KNX/03%20-%20KNX%20Award%20Projects/KNX%20Award%20Projects%202008/S%20-%20KNX%20Award%202008%20-%20Special%20EN.pdf>.
15. —. Cruise Ship MS Belle de l'Adriatique. *KNX.org*. [Online] [Cited: February 23, 2011.] <http://www.knx.org/fileadmin/downloads/02%20-%20What%20is%20KNX/03%20-%20>

%20KNX%20Award%20Projects/KNX%20Award%20Projects%202008/S%20-%20KNX%20Award%202008%20-%20Special%20EN.pdf.

16. —. Terminal 5 Heathrow in London. *KNX.org*. [Online] [Cited: February 23, 2011.] [http://www.knx.org/fileadmin/downloads/02%20-%20What%20is%20KNX/03%20-%20KNX%20Award%20Projects/KNX%20Award%20Projects%202006/english\\_commercial\\_Andromeda%20Telematics.PDF](http://www.knx.org/fileadmin/downloads/02%20-%20What%20is%20KNX/03%20-%20KNX%20Award%20Projects/KNX%20Award%20Projects%202006/english_commercial_Andromeda%20Telematics.PDF).
17. **Wikipedia EN**. KNX. *Wikipedia*. [Online] [Cited: february 22, 2011.] [http://en.wikipedia.org/wiki/KNX\\_\(standard\)](http://en.wikipedia.org/wiki/KNX_(standard)).
18. **Echelon**. LonWorks Technology Overview. *Echelon*. [Online] [Cited: February 23, 2011.] <http://www.echelon.com/communities/developers/lonworks/>.
19. **Wikipedia EN**. LonWorks. *Wikipedia EN*. [Online] [Cited: February 23, 2011.] <http://en.wikipedia.org/wiki/LonWorks>.
20. **Galbavý, Martin**. *Vizualizace a vzdálené řízení v síti LonWorks*. Bachelor thesis. Prague : FEE CTU in Prague, 2006.
21. **DALI AG**. DALI. [Online] [Cited: June 17, 2009.] <http://www.dali-ag.org/>.
22. **Advance**. The ABC's of DALI. [Online] [Cited: June 17, 2009.] [http://www.advancetransformer.com/uploads/resources/CO-7110-R01\\_ABCofDALI.pdf](http://www.advancetransformer.com/uploads/resources/CO-7110-R01_ABCofDALI.pdf).
23. **Wikipedia EN**. DALI. [Online] [Cited: June 17, 2009.] [http://en.wikipedia.org/wiki/Digital\\_Addressable\\_Lighting\\_Interface](http://en.wikipedia.org/wiki/Digital_Addressable_Lighting_Interface).
24. **Resi**. User manual RESI-DALI-MODBUS Converter. [Online] [Cited: June 17, 2009.] [http://www.marcomweb.it/Documenti/DALI\\_MODBUS.pdf](http://www.marcomweb.it/Documenti/DALI_MODBUS.pdf).
25. **Baldwin, Tom**. 5 pin XLR. [Online] [Cited: June 17, 2009.] [freespace.virgin.net/tom.baldwin/pinout-5xlr.html](http://freespace.virgin.net/tom.baldwin/pinout-5xlr.html).
26. **Baldwin, Tom**. 3 pin XLR. [Online] [Cited: June 17, 2009.] <http://freespace.virgin.net/tom.baldwin/pinout-3xlr.html>.
27. **Doug Fleenor Design Inc**. DMX 512 A Simple Explanation. [Online] [Cited: June 17, 2009.] [www.dfd.com/dmxbasic.html](http://www.dfd.com/dmxbasic.html).
28. **Rol, Erwin**. DMX512. [Online] [Cited: June 17, 2009.] [www.erwinrol.com/index.php?stagecraft/dmx.php](http://www.erwinrol.com/index.php?stagecraft/dmx.php).
29. **Wikipedia EN**. RDM. [Online] [Cited: June 17, 2009.] [http://en.wikipedia.org/wiki/RDM\\_\(lighting\)](http://en.wikipedia.org/wiki/RDM_(lighting)).
30. **Rol, Erwin**. Art-Net. [Online] [Cited: June 17, 2009.] <http://www.erwinrol.com/index.php?stagecraft/artnet.php>.
31. **USITT**. DMX512. [Online] [Cited: June 17, 2009.] <http://www.usitt.org/standards/DMX512.html>.

32. **MCUserver.** Anatomie DMX152. [Online] [Cited: June 17, 2009.]  
<http://www.mcu.cz/print.php?news.81>.
33. **Kar, Ujjal.** DMX512 EQUIPMENT. [Online] [Cited: June 17, 2009.] <http://www.dmx512-online.com/equpt.html>.
34. **Kar, Ujjal.** The DMX512 Packet. [Online] [Cited: June 17, 2009.] <http://www.dmx512-online.com/packt.html>.
35. **Pangolin Laser Systems.** About DMX512. [Online] [Cited: June 17, 2009.]  
<http://www.pangolin.com/LD2000/dmx-about.htm>.
36. **Wikipedia CS.** DMX512. [Online] [Cited: June 17, 2009.] <http://cs.wikipedia.org/wiki/DMX512>.
37. **Nuši, Jaroslav.** Protokol DMX512. [Online] [Cited: June 17, 2009.]  
<http://dmx512.svetla.org/theory.htm>.
38. **EnOcean.** EnOcean - Self-powered Wireless Sensors. [Online] [Cited: May 12, 2009.]  
<http://www.enocean.com/>.
39. **Wikipedia EN.** *EnOcean*. [Online] [Cited: May 12, 2009.] <http://en.wikipedia.org/wiki/EnOcean>.
40. **TDC.** EnOcean Voting and Quiz Scoring System. *TDC*. [Online] [Cited: May 12, 2009.]  
<http://www.tdc.co.uk/index.php?key=voting>.
41. **EnOcean.** Case Studies. [Online] [Cited: May 12, 2009.] <http://www.enocean.com/en/case-studies/>.
42. **Technik.ihned.cz.** I nepatrná energie se dá efektivně využít. *Odpady - Odpadové hospodářství, ekonomika životního prostředí*. [Online] 3 25, 2005. [Cited: May 4, 2009.]  
[http://odpady.ihned.cz/?secpart=\\_clanek\\_fchec\\_ih\\_](http://odpady.ihned.cz/?secpart=_clanek_fchec_ih_).
43. **M-Bus Usergroup.** M-Bus Usergroup. [Online] [Cited: August 27, 2009.] [www.m-bus.com](http://www.m-bus.com).
44. **HWgroup.** HWg-PWR: M-Bus IP měřič energie (SNMP, WEB). *HWgroup*. [Online] [Cited: March 28, 2011.] [http://www.hw-group.com/products/HWg-PWR/hwg-pwr\\_cz.html](http://www.hw-group.com/products/HWg-PWR/hwg-pwr_cz.html).
45. **Vojáček, Antonín.** Sběrnice Wireless M-BUS - jde to i bezdrátově... *Automatizace.HW.cz*. [Online] [Cited: March 28, 2011.] <http://automatizace.hw.cz/sbernice-wireless-mbus-jde-i-bezdratove>.
46. **Kocourek, Petr and Novák, Jiří.** M- Bus: Popis funkce. [Online] [Cited: August 26, 2009.]  
<http://fieldbus.feld.cvut.cz/mbus/mbus.html>.
47. **Wikipedia EN.** Meter-Bus. *Wikipedia*. [Online] [Cited: March 28, 2011.]  
<http://en.wikipedia.org/wiki/Meter-Bus>.
48. **Wikipedia CZ.** M-Bus. *Wikipedia*. [Online] [Cited: March 28, 2011.]  
<http://cs.wikipedia.org/wiki/M-Bus>.

49. **Vojáček, Antonín.** M-BUS (Meter-Bus) - základní popis komunikačního protokolu. *Automatizace.hw.cz*. [Online] [Cited: March 28, 2011.] <http://automatizace.hw.cz/mbus-meterbus-zakladni-popis-komunikacniho-modelu>.
50. **OpenTherm Association.** OpenTherm Association. *OpenTherm*. [Online] [Cited: May 19, 2009.] <http://www.opentherm.eu>.
51. **NEC.** 78K0 Series - OpenTherm Data Link Layer Implementation. [Online] [Cited: June 17, 2009.] [http://www.eu.necel.com/\\_pdf/U17475EE1V0AN00.PDF](http://www.eu.necel.com/_pdf/U17475EE1V0AN00.PDF).
52. **OpenTherm Association.** OpenTherm Newsletter 04/2009. [Online] [Cited: June 17, 2009.] <http://www.opentherm.eu/pdf/OTnewsletter0409.pdf>.
53. **KWE Technologies Group.** Versatronik® 505 OT - OpenTherm Gateway. [Online] [Cited: June 17, 2009.] <http://www.kwe-tech.com/index.php?section=Products&subs=Comm%20Gateways&page=OpenTherm.html>.
54. **HKL-Info.** Bus-Systeme in der Heizungstechnik. [Online] [Cited: June 17, 2009.] <http://hkl-info.de/bussystemehkl.html>.
55. **Wikipedia EN.** OpenTherm. [Online] [Cited: June 17, 2009.] <http://en.wikipedia.org/wiki/OpenTherm>.
56. **O'Hara, Martin.** An Introduction to OpenTherm. [Online] [Cited: June 17, 2009.] [http://www.beama.org.uk/UserFiles/file/downloads/smarthouse/OpenTherm\\_SmartHomesJuly07.pdf](http://www.beama.org.uk/UserFiles/file/downloads/smarthouse/OpenTherm_SmartHomesJuly07.pdf).
57. **Honeywell.** Data-Id Map. [Online] [Cited: June 17, 2009.] <http://www.hccp.nl/opentherm%20pages/Data-Id%20Map.otm>.
58. **SMI Group.** SMI. [Online] [Cited: August 20, 2009.] <http://www.smi-group.com/>.
59. **Jüngst, Christoph.** *SMI - Drive Technology for Roller Shutters and Sun Protection*. [Online] [Cited: August 20, 2009.] [http://www.knx.org/fileadmin/downloads/05%20-%20KNX%20Partners/02%20-%20Becoming%20a%20certified%20KNX%20Training%20Centre/2008%20Conference/15b\\_Jungst\\_Becker\\_en.pdf](http://www.knx.org/fileadmin/downloads/05%20-%20KNX%20Partners/02%20-%20Becoming%20a%20certified%20KNX%20Training%20Centre/2008%20Conference/15b_Jungst_Becker_en.pdf).
60. **Wikipedia.** oBIX. *Wikipedia*. [Online] [Cited: March 25, 2011.] <http://en.wikipedia.org/wiki/OBIX>.
61. **Nývlt, Ondřej.** Automatický návrh řízení pro domovní automatizaci. *Diplomová práce*. Praha : FEL ČVUT, 2008.
62. **Chaloupek, Jindřich.** Sběrnice pro řízení inteligentních budov. *Semestrální práce*. Praha : FEL ČVUT, 2009.
63. **KNX.** Connecting M-Bus meters to the KNX World. *KNX.org*. [Online] [Cited: August 26, 2009.] <http://www.knx.org/knx/knx-applications/smart-metering/connecting-m-bus/>.