

Application of Adaptive Modular Micro Devices (AMMD) in Home Automation

Dipl.-Ing. Thomas A. Theissen, tth@wikon.de
Product Management
WIKON Kommunikationstechnik GmbH
Sauerwiesen 2
67661 Kaiserslautern, Germany

The project has partially been funded by :

Bundesministerium für Bildung und Forschung (BMBF)
Heinemannstr. 2
53175 Bonn, Germany

The project has been carried out in cooperation with :

Technische Universität München
Lehrstuhl für Messsystem- und Sensortechnik
Univ.-Prof. Dr.-Ing. F. Schneider, Dipl.-Ing. Stjepan Skrtic
Theresienstr. 90 / N5
D-80333 München, Germany

Binder Elektronik GmbH
Wolfgang Binder, Dipl.-Ing. (FH) Matthias Geiger
Hauptstraße 142
D-74889 Sinsheim, Germany

Abstract

The paper will describe a new approach to interacting with EIB/KONNEX networks in a home automation network. Basically, the combination of three new features form an innovative concept for a graphical user interface display :

A new **hardware** has been developed to host a graphical touch display for a standard flush mount device. Three microcontrollers are used to accomplish the demanding computational power. Stacked PCB's in Multi-Chip-Module (MCM) technology are used to achieve a maximum integration in the given mechanical dimensions. Three layers are stacked with dedicated tasks : An upper layer provides the user interface with display, touch field, microcontroller and USB connector for programming user settings. An intermediate layer provides a microcontroller as well as additional memory for all logical tasks. The bottom layer finally holds the bus interface with a microcontroller using the TPUART.

A **firmware** has been implemented in all of the three

microcontrollers. The bottom controller basically holds the EIB communication stack. The front controller handles all outputs to the graphical display as well as inputs through the touch sensitive area. Also, it handles communication to a local PC for parameter download. The main controller coordinates all user interactions and generates/receives bus communication telegrams. It also holds instructions for data logging, graphics, PLC functionality and scenes. Communication between these controllers is achieved by a serial protocol.

A new **software** concept has been developed for easy system configuration by the end user. As opposed to products already on the market, this display would be easy programmable by any user. Anybody will be enabled to design an individual concept of scenes, loggers and even PLC functionality. All editing can be done in a web-based XML environment. With this, the user will interactively create a system behaviour model based on UML. This can then be downloaded to the target display via a standard USB link. Minor changes, however, can be done directly at the display. This includes - but is not limited to - connection of digital or analogue sensors to any appropriate actor.

As of now, the project is in a pre-market status. Several different prototypes have been developed and are currently tested in an end user environment.

Table of content

Conceptual outline	p. 2
Project setup	p. 2
Data conversion to AMMD/XML format	p. 2
Data editor	p. 2
Data download	p. 2
Creating new functionalities	p. 2
Hardware implementation	p. 2
Hardware, Level 1	p. 3
Hardware, Level 2	p. 4
Hardware, Level 3	p. 4
Hardware, Assembly	p. 5
Firmware, Level 1	p. 5
Firmware, Level 2	p. 5
Firmware, Level 3	p. 5
Software	p. 6
PHP parser for ETS/XML data translation	p. 6
Web interface for XML generation	p. 6
PHP parser for XML/AMMD data translation	p. 6
Download software (RS232/USB)	p. 6
Perspective	p. 7

Conceptual Outline

As of now, KNX technology basically relies on a trained technician for installation and maintenance. So far, no software or device is provided for the end user to modify his installations, create new scene settings or implement PLC or data logger functionality. Anybody who wants to go into details needs to work with special software and/or make major investments into this technology.

At the same time, conventional installations can not face the demands from today's customers : A versatile yet easy to understand home automation system with a user interface, simple enough to be understood at the first look.

After initial research, a concept for a new communication controller has been designed and first prototypes have been implemented.

With AMMD, any end user will be empowered to modify his KNX home network by himself. At the same time, he will get a fancy and powerful display for easy access and visualization of all of his KNX devices. The following scenarios have been created :

Project Setup

As the standard procedure, a technician will set up the KNX network using the ETS tool. The generated data base is the foundation for all further steps. The AMMD unit itself will not be part of this project. It will only be installed later on.

Data conversion to AMMD / XML format

The ETS generated data will now be transformed into an XML file.

Data editing

The user will have a choice of modifying these XML data from the beginning. He can create scenarios, PLC functionality (e.g. If-Then) and data loggers trough a web interface software.

Data download

Connecting his computer to the AMMD device, the user will now download his new created connections to the device. This can be achieved either trough the EIB network or via direct connection to the AMMD device. This would then not require a KNX interface and be the preferred method.

Creating new functionalities

The user will now be able to create new functionalities on the AMMD device itself in EDITOR mode. Accessible are

- Devices (e.g. binary or analogue sensors and actors)
- Rules (e.g. If-then)
- Tasks (e.g. Data logger)

In a menu driven software interface, he will first select the device which he wants to access (e.g. Lamp actuator). He can then select from a variety of appropriate user interfaces (e.g. rocker switch) and connect these directly or perform another task.

The display would then be set to USER mode and the new implemented functionalities would be freely available.

Hardware implementation

Another aim of AMMD was to keep the device small enough to fit into a standard flush mount enclosure. However, the dimensions of a perspective end product would greatly depend on the used display technology and size.

Hardware

Stacked Multi-Chip-Modules (MCM) are the only way to design complex electronic devices for flush mount applications. Thus, AMMD is based on this technology and consists of the following levels:

Level 1 : Display carrier

The top level holds all components required for user interaction :

- LCD or OLED Display
- Illumination for the LCD
- Touch Field
- Passive Infrared (PIR) Detector
- Programming Interface (Serial RS232 or USB)
- Buzzer

In addition, this level is equipped with the following :

- Microcontroller MSP430
- Power supply for the display
- Serial link to level 2 module

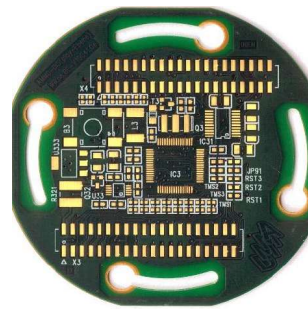


Fig. 3 : PCB – Top

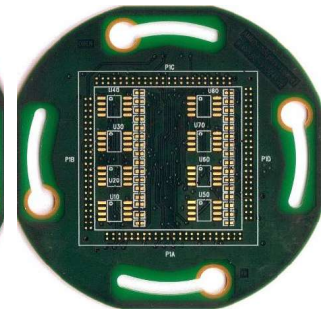


Fig. 4 : PCB – Bottom

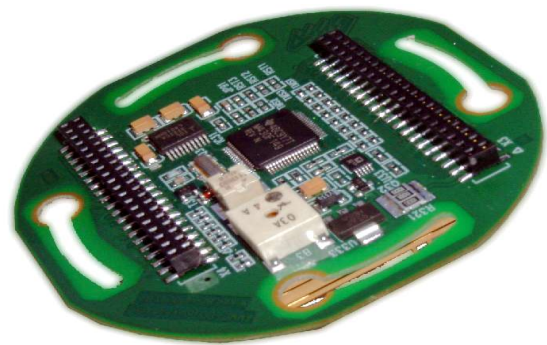


Fig. 5 : Display carrier, top

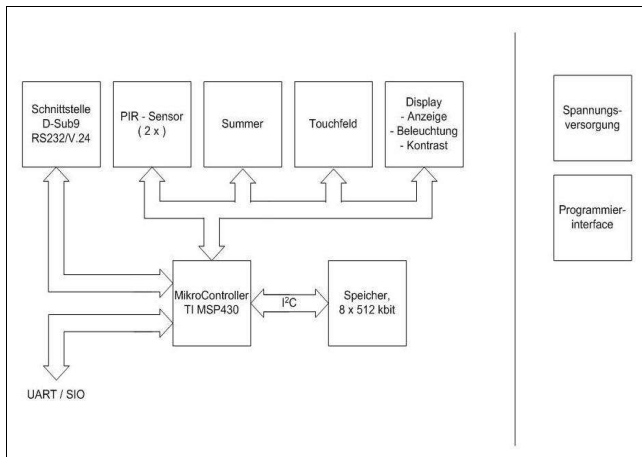


Fig. 1 : Display carrier, basic design

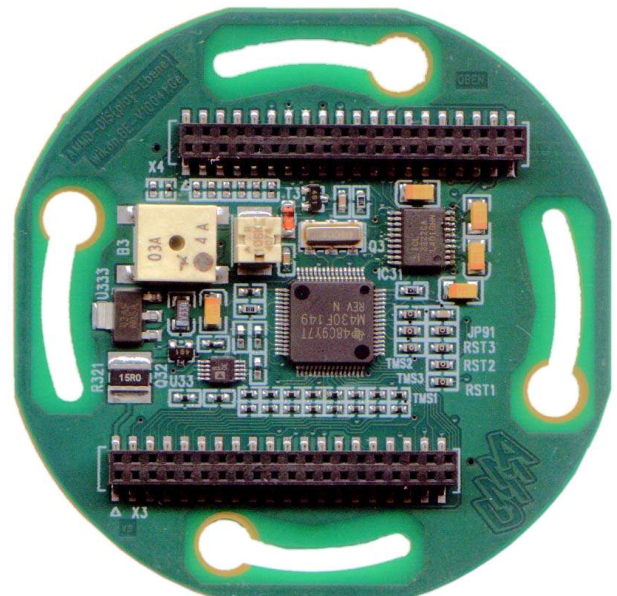


Fig. 6 : Top assembly

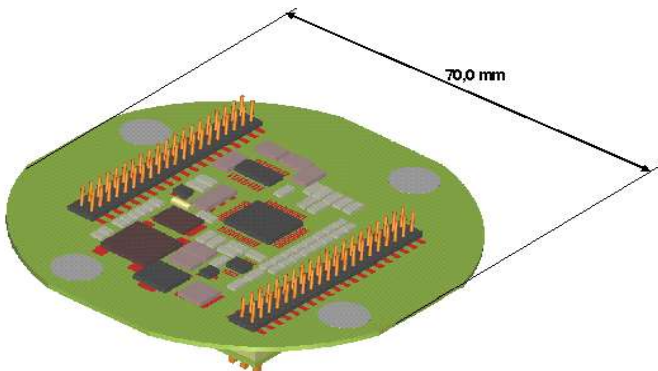


Fig. 2 : Display Carrier, dimensions

Level 2 : 'Intelligence' Level

The centre ('intelligence') level basically coordinates all communication. Thus, additional memory and two serial ports are required for this microcontroller :

- Microcontroller MSP430
- Additional, non-volatile memory
- Two serial ports to level 1 and level 3

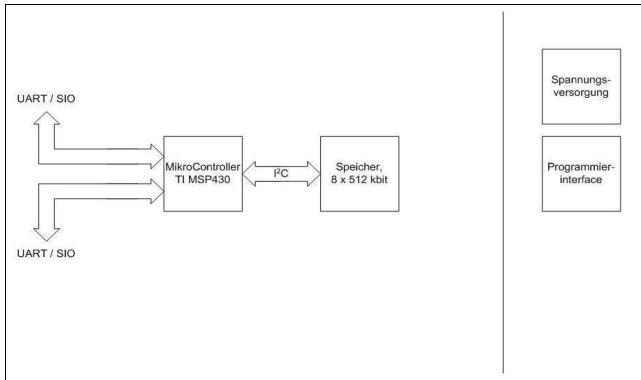


Fig. 7 : 'Intelligence' Level, basic design

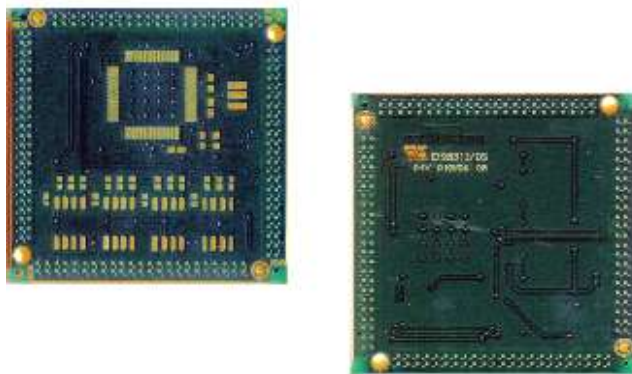


Fig. 8 : 'Intelligence' Level, PCB layout

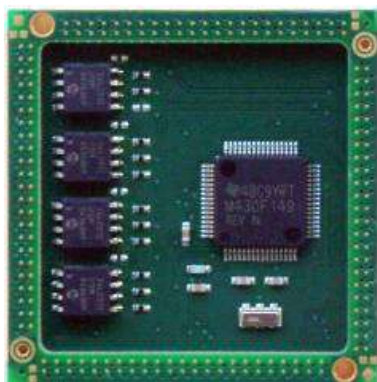


Fig. 9 : 'Intelligence' Level, assembly

Level 3 : Bus Coupler

The 3rd level realizes communication with the bus system. In AMMD, two bus levels have been implemented : KNX and Ethernet. This paper however, will discuss the KNX units only.

- TPUART for bus communication
- User interface for installation : LED, push button
- Microcontroller MSP430
- Magnetic iCoupler for galvanic isolation
- Serial link to level 2 module
- Power supply

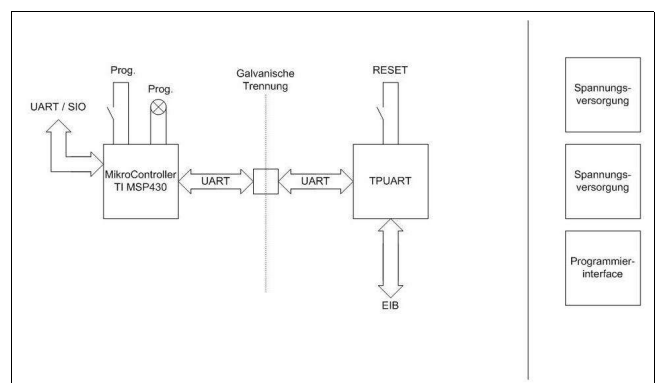


Fig. 10 : KNX level, basic design

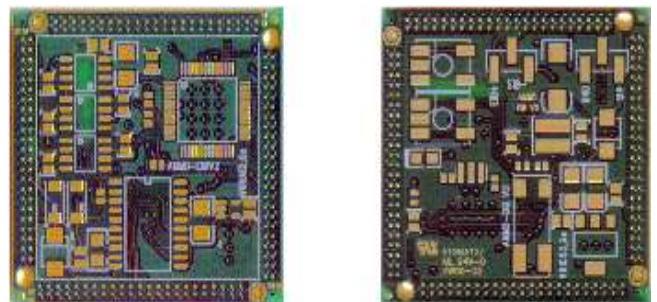


Fig. 11 : KNX Level, PCB layout



Fig. 12 : KNX level, assembly

Assembly

All modules are assembled by means of a FR4 frames :

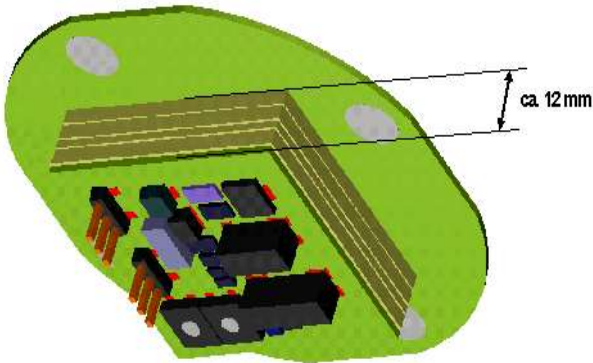


Fig. 13 : Assembly, bottom view

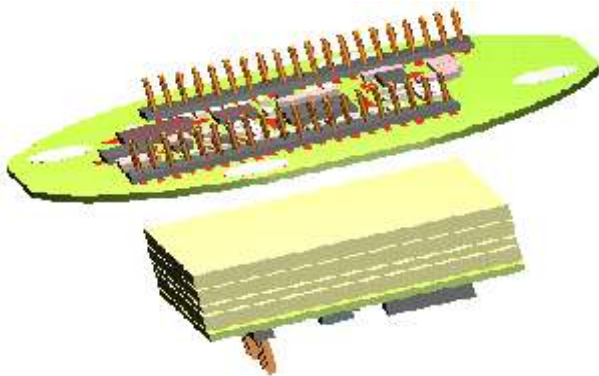


Fig. 14 : Assembly, principle

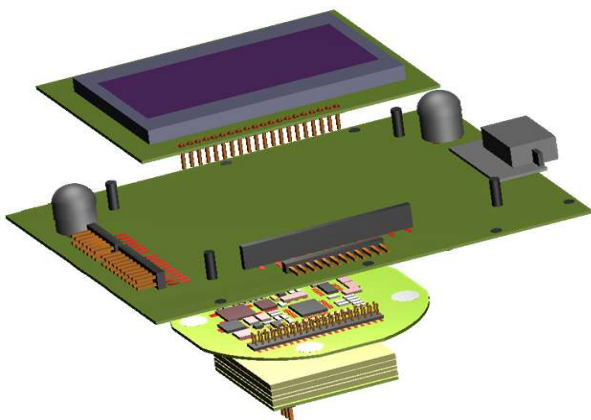


Fig. 15 : Experimental LCD carrier assembly

Firmware

Major parts of the AMMD firmware has been implemented based on the MicroC/OS-II Real Time Kernel operating system by Jean Labrosse.

All system software developed by WIKON has been designed using UML (Unified Modelling Language) activity diagrams. This method has been found superior for the design of new applications as well as for all end customer communication.

Level 1 : Display carrier

The obvious and main task of this level is the complete user interface handling. Versatile drivers have been implemented, to accomplish usage of different display technologies and interfaces. Black-and-white displays can be used as well as colour devices. A resistive touch field is used for all user data input. Audible feedback can be given to the user by means of the buzzer, and presence detection is accomplished by a PIR sensor.

Other than this, the display level – as of now – handles all PLC functionality as well : The user tasks are stored and performed in non-volatile memory and executed on request. The Real-Time-Clock is maintained and the data logger readings are stored.

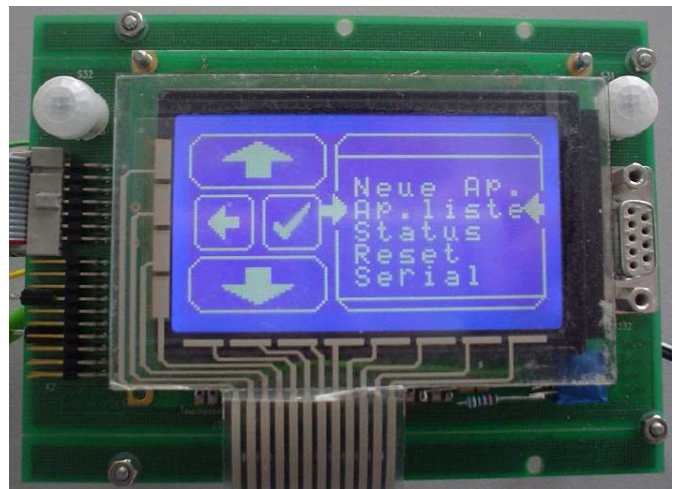


Fig. 16 : Experimental display (Example view)

Level 2 : 'Intelligence' Level

This intermediate level interfaces between the bus coupler and the display level. It manages all configuration and relation data. All communication data to the next higher level is handled in a well-specified AMMD object format. Thus, different bus couplers can be used without modifying this level's firmware.

Level 3 : Bus Coupler

The firmware for the bus level units is mainly based on an EIB/KNX stack implementation. It handles incoming communication telegrams from the bus and generates the

appropriate messages to the next level. In the same manner, bus telegrams requested by the 'intelligence' level will be transmitted. All data is translated between the bus specific format (e.g. group address) and an AMMD object format.

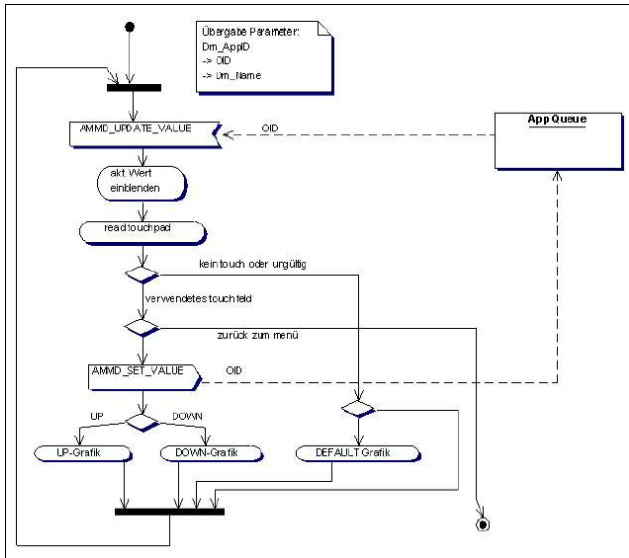


Fig 16 : UML for application "Dimmer"

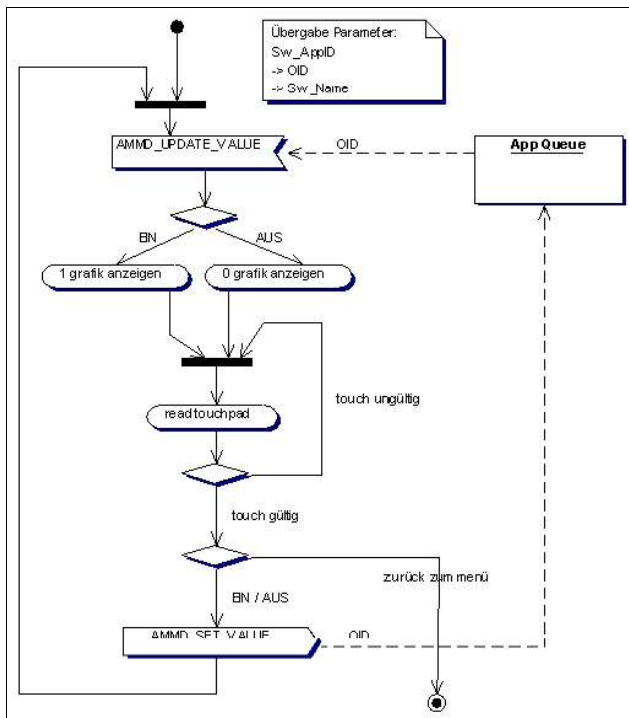


Fig 17 : UML for application "Switch"

Software

Four software packages have been developed for Microsoft Windows 2000/XP. As of now, the following programs are required for complete AMMD usage :

PHP parser for ETS/XML data translation

A parser – based on PHP (Hypertext Preprocessor) – has been developed for translating ETS generated data to AMMD XML (Extensible Markup Language) format.

As the standard tool in the KNX environment, most installations are planned and installed using the ETS tool. Thus, almost all projects can be accessed through an ETS generated data file.

Since AMMD data is independent of the underlying bus system, these data – devices and their relations - need to be translated to AMMD XML object data. A PHP based program has been developed to perform this.

Web interface for XML generation

The common XML language is used to completely describe the devices behaviour. XML easily allows implementation of PLC, timer and data logger functionality. Also, XML files can be generated in two ways :

- Any advanced user will be able to generate and edit an XML file with a standard editor.
- Users not familiar with the XML language will work with an interactive web interface. The output of this will then be an XML file, representing the requested operation.

The new AMMD XML DTD (Document Type Definition) supports standard applications, If-Then rules and all real and virtual AMMD devices. This is an example :

```
<ammd>
<infrastruktur>
<geraet oid="1" name="lampe1" ammd_typ="A_BINARY" />
<geraet oid="2" name="lampe2" ammd_typ="A_BINARY" />
<geraet oid="3" name="Markise" ammd_typ="A_RELATIVE" />
<geraet oid="4" name="LichtSensor" ammd_typ="A_RELATIVE"
aktor="nein" />
</infrastruktur>

<funktionalitaeten>
<rule r_name="MarkiseZU" oid_s="4" oid_a="3" von="0"
bis="30" wert="0" op_id="AKT" op_rel="ABS" />
<rule r_name="MarkiseAUF" oid_s="4" oid_a="3" von="70"
bis="100" wert="100" op_id="AKT" op_rel="ABS" />
<app titel="AlleLampen" OID1="1" PollOID2="2"
AppTyp="doppelschalter" />
</funktionalitaeten>
</ammd>
```

PHP parser for XML/AMMD data translation

The XML file will then be translated into the AMMD object language.

Download software (RS232/USB)

After translation, the new configuration will be downloaded into the AMMD module. A menu-driven software can accomplish this either through a serial or a USB port.

Perspective

The developed device has made user access to any KNX network easier and more user friendly. The general idea of giving anybody access to basic as well as advanced functionalities on the home network seems to have worked. However, some restrictions apply :

- Protection
As of now, any user has the same access to the network. For further research as well as products, a password and/or user access limitation needs to be evaluated.
- Power supply
As of now, the device is designed for an additional 24V power supply. Using low power display technology (smart paper), even a unit without these additional wirings could be envisioned.
- Centre Level
Eventually, the units could work without the centre level. This would be possible, once the new processors of the MSP430 family are powerful enough to handle the additional tasks in the remaining μ C's.
- KNX certification
So far, no KNX certification has been applied for.