KNX, Ebus, Canbus

KNX

“KNX is the world’s only open standard for building control and automation and can make major energy savings of up to 60%, significantly reducing the carbon footprint of a building. This is truly green sustainable technology that can be applied to small and large buildings alike.”

KNX Basics

- Separates power supply and device control into two separate networks
- Devices made by different manufactures can be interconnected
- Actors are used between the devices and the power supply. They are connected to the bus

KNX Device Categories

- **A-mode:** Automatic mode devices which can configure themselves and are able to be installed by the end user
- **E-mode:** Easy mode devices require basic training to install.
- **S-mode:** System mode devices are the most sophisticated one.

KNX Architecture

- Sensors: Gather information and send it on the bus
  - Push buttons, thermostats...
- Actuators: Receive commands from the bus which are converted into actions
  - dimming units, heating valves, displays...
- Controllers and other logic functions
  - room temperature controllers...
- System devices and components
  - e.g. line couplers

Ebus

Ebus Basics

- To control heating & solar components
- Developed & used mostly by German companies
- Can provide power to participants (5V/18mA)
- Up to 25 masters and 228 slaves

Ebus Physical Layer

- 2 Wires, twisted pair
- serial data transmission in 8 bit UART mode
- Fix data rate of 2400 Baud
- Start- and single stop bits
- Logical 0 = 9-12V, Logical 1 = 15-24V, power over bus

Ebus Data-Link Layer

- Master-Slave-communication
- Only masters can initiate commands
- 254 primary commands and 254 secondary commands

Packet Structure

- 8-bit source address (lower 4 bits: priority, higher 4 bits: bus master address)
- 8-bit destination address (254 unicast destinations, 0xff = broadcast)
- 16-bit command code (8 bit primary command, 8 bit secondary command)
- 8-bit data length indicator: 0–16 (byte stuffing bytes not counted)
- 0–16 data bytes
- 8-bit CRC

Ebus Data-Link Layer

1. master starts by sending command
2. slaves reply by sending ACK, length, 0-16 data bytes and CRC
3. master sends ACK byte back
4. master sends SYN byte (0xaa) to signal a free bus

- for master and slaves:
  send 0x00 (correctly received)
  send 0xff (incorrectly received)

Ebus Application layer

- Different generic protocol packages on application level
- Some manufacturers have proprietary expansion packs
- Compatibility with other implementations is mostly maintained on physical and data-link-layer, not on the application layers

Ebus summary

- Used by german manufacturers
- used for heating infrastructure
- every manufacturer uses own extensions to the standard
- therefore the standard is not so important
- documentation not easy to find

Canbus

Canbus Basics

Purpose: reducing the length of electrical wiring

Automotive Industry, also used in other automation areas

Can is provided in two different standards with different application domains

Can be used as diagnostic interface too.

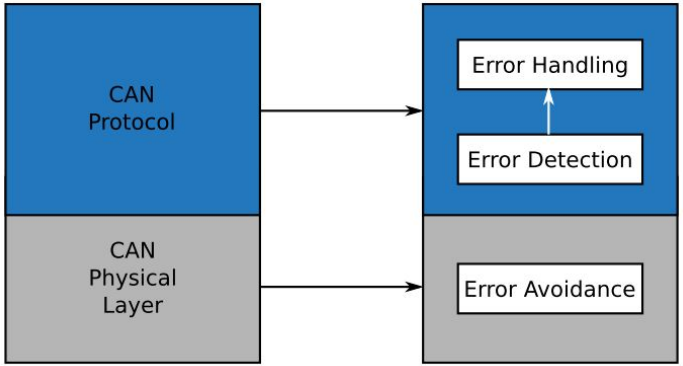
Supports different Bus structures

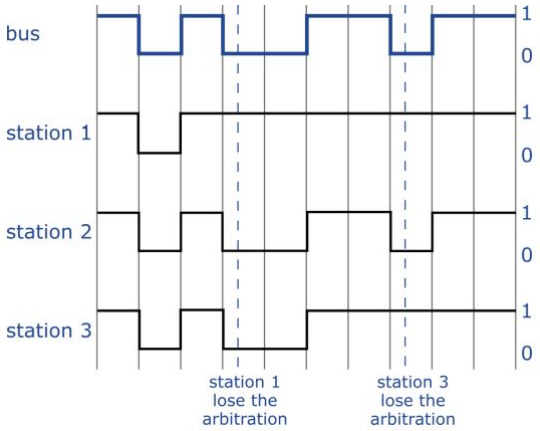
Can bus

Supports every transmission media with dominant and recessive state

Is organized into messages with multi master

Highest priority wins





CAN Bus node

µ-Controller (µC)

- executes application

CAN-Controller

- does transimission and receiving

CAN-transceiver

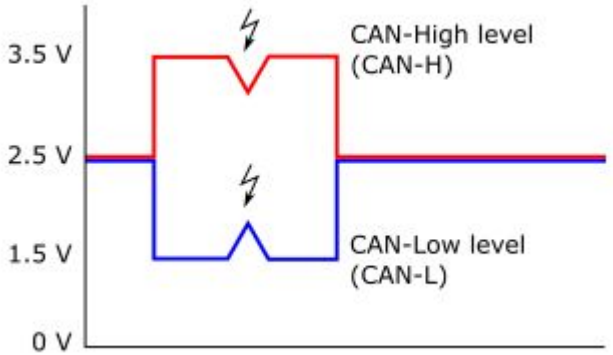
- physical signal amplifying

Can High

Transmits data at a speed between 125 kBit/s to 1 MBit/s

Is normally used for safety- and time critical stuff.

uses two cables with differential signals
to detect and remove disturbances.



Can Low

Transmits data between 5 and 125 kBit/s

used in the comfort domain

used for simple actors like switches or control knobs.

uses a single wire and a share mass and is only capable to work over small distances.