An open approach for connecting control devices of public transport vehicles

* Gerd Waizmann, OnTime Engineering GmbH
Frank Steinert, OnTime Engineering GmbH

We will describe a system, that uses CAN as a medium to connect different control devices on board of vehicles of the public transport system. Among the connected devices, there are the main board computer, graphical display panels for the information of the driver and a wireless data communication to a local traffic control center. Possible future extensions are the integration of a high speed infrared communication or measuring the position with a GPS receiver.

We will give a technical overview on this system. Especially the implementation of an open system approach, based on the results of the CiA working group CAN in mobile applications will be described. There will be also a discussion of the advantages using an open approach, and possible reasons why not using an open approach. The authors are involved in industrial projects concerning these topics. They are also active members of the CiA working group: „CAN in mobile applications“

Dieses Papier beschreibt ein System, bei dem CAN zur Verbindung der verschiedenen Steuergeräte an Bord eines Fahrzeuges des Öffentlichen Personennahverkehrs verwendet wird. Beispiele für diese Steuergeräte sind Bordrechner, grafische Anzeige- und Bedieneinheiten und eine drahtlose Datenkommunikation zu einer lokalen Verkehrssteuerzentrale.

Hiermit wird ein technischer Überblick über ein solches System geben. Dabei wird besonders auf die Nutzung eines offenen Systemansatzes auf der Basis der CiA Arbeitsgruppe: „CAN in mobile applications“ eingegangen. Darüber hinaus werden die Vorteile eines offenen Systemansatzes dargelegt und auch mögliche Gründe gegen einen solchen offenen Ansatz diskutiert. Die Autoren sind in Industrieprojekten und Standardisierungsbemühungen zu diesen Themen aktiv beteiligt.

Gerd Waizmann is Managing Director of OnTime Engineering and active member of the CiA working group: 'CAN in mobile applications'.
Frank Steinert is employee of OnTime Engineering GmbH and is working on projects concerning the application of CAN in vehicles.

OnTime Engineering, Gesellschaft für Informationstechnik mbH
Rosenheimer Str. 3 D-83083 Riedering Tel: +49-8036-4645 FAX: +49-8036-4757
email: 100414.1302@compuserve.com

Vehicles of public transport systems - today

More than ten years ago, the german 'Verband öffentlicher Verkehrsbetriebe' (VÖV) defined a serial communication system for use on board of vehicles of the public transport system. The members of the VÖV are manufacturers of equipment and operators of urban transportation systems. The communication system was called IBIS (integrated board information system) and is widely used on board of vehicles like bus, tram or trolley bus. The IBIS system consists of driver and passenger information systems and a link to the traffic control center for fleet management purposes.

Equipment with an IBIS interface might be a userinterfacefor the driver, ticket canceller, destination displays, automatic announcement of the next stop, and a lot of other equipment. The communication among all these devices was realized by a so-called: 'Wagenbus'. If there was the need to communicate between wagons, a second system, the so-called: 'Zugbus', was used. The communication between the vehicles and the dispatch center was realized by a special data radio link.

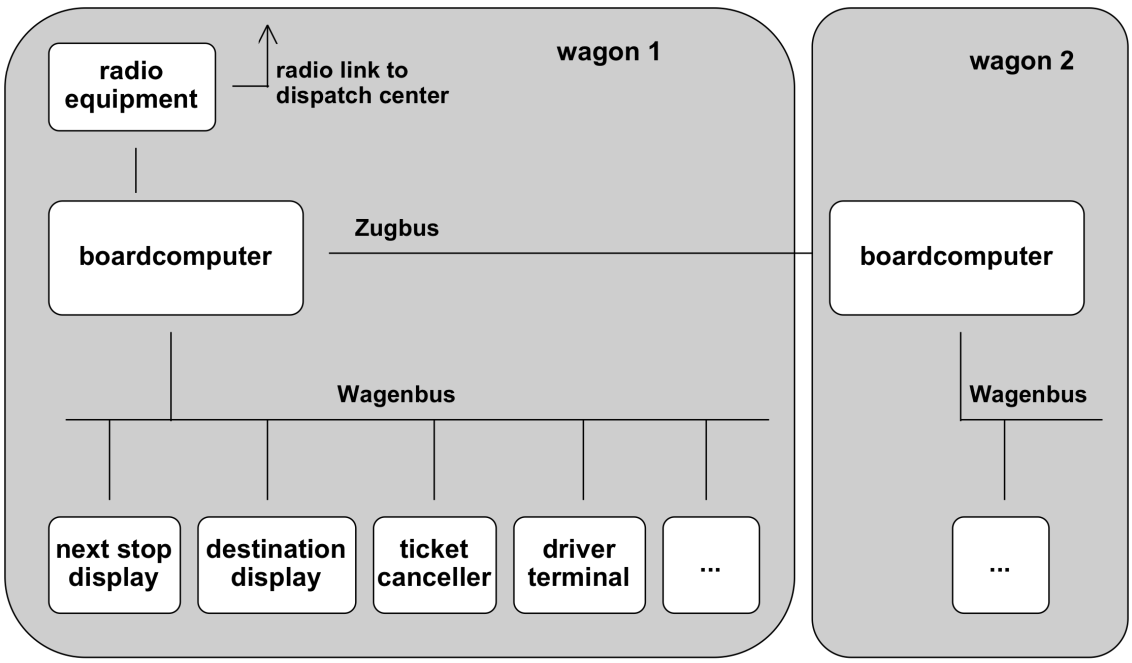

Abb1: IBIS on board of public transport vehicles

The communication among the devices, connected to the zug- or wagenbus was realized by a simple protocol. The physical link was a 4 wire cabling (each two for send and receive). The security of the protocol was achieved by a simple parity. The maximum data transfer rate was very restricted by a maximum of 1200 Baud. Due to the ASCII based coding of the telegrams, the transmission of free text was restricted. The architecture of the system was based on a Master-Slave polling scheme with a cycle time of 1 second and more and a lot of telegram overhead.

When defining the IBIS ten yeas ago, there were two main aims:

- '...the standardization of the equipment from different kind, different manufacturers, used in different kinds of vehicles...'
- '... a modular, easy to expand system, with a low basic axpenditure..'

Vehicles of public transport systems - new approach

Today, IBIS is widely used on board of german transportation vehicles. But in the time between the definition of IBIS and today, a lot of new requirements and technical possibilities came up. New technological possibilities, like digital radio links (for example via GSM) or doing position fixes using the satellite based global positioning system (GPS) came up, and have to be integrated. But all this is no longer possible with the limited bandwidth and the technical basics of the zug- and wagenbus.

But any new system architecture has to consider the wide base of installed and available equipment for the IBIS. So, any new system architecture has to integrate the new approach with the older one.

Requirements

- it still has to be an open system
- enough bandwidth, even for download of program and data
- easy to handle and to implement.
- low overhead for short data telegrams
- easy to expand for future purposes

Why using CAN?

One possible successor of the wagenbus is the CAN. The CAN was designed for using on board of vehicles and is also proved by using in the factory automation as a fieldbus. Based on CAN it is possible to implement a new system, that meets the prementioned requirements.

The special reasons for using the CAN had been:

- performance
  As CAN based systems are multimaster systems, an up-to-date architecture with distributed intelligence is possible. The layer 2 is already implemented in silicon, thus eases the software and unloads the CPU from tasks like detection and correction of transmission errors.

• reliability
CAN has a hamming-distance of 6, outperforming most of its competitors.

• multiple sources
a lot of chip vendors offer silicon support for CAN. Also more and more vehicles of the new generation are already using the CAN.

• open standards
CAN is standardized as ISO/DIS 11898 and available to the user without any royalty fees. Furthermore the CiA defined the CAN application layer (CAL) as a standard for the layer 7 and a profile for mobile applications.

new architecture

The new architecture of the system is based on a distributed, multi-master, client-server architecture based on CAN. It consists of:

• boardcomputer
for general control tasks. It stores the schedule of the vehicle, and manages all the tasks needed on board of the vehicle. It also acts as a gateway between the new, CAN based, system, and the old-style, Zug- and Wagenbus based,

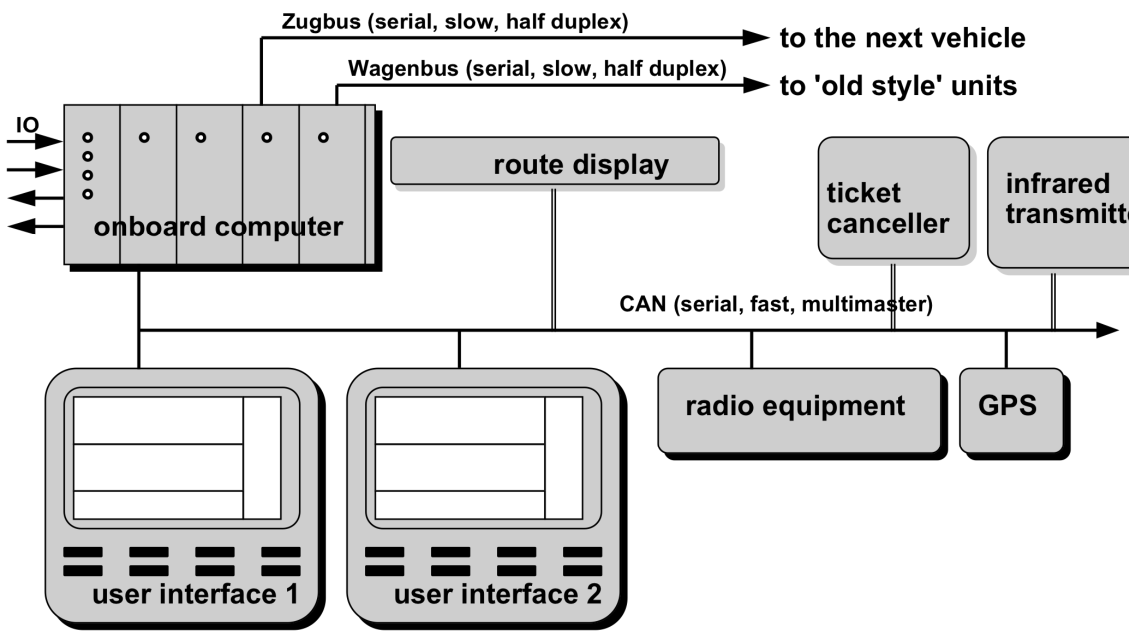

Abb 2: Public transport vehicles - new approach

components.

- user interface
  graphical LCD interface with pushbuttons for the interaction with the busdriver.

- radio equipment
  sends and receives data telegrams to and from the urban dispatch center.

At the first implementation step, these components are linked with a common CAN. The baudrate is set to 250 kBit/s. All other equipment is still connected with the oldstyle Wagenbus. But it is intended, that more and more equipment will use the CAN for communication. For example, the task of doing position fixes could be delegated to a special GPS Module with a CAN interface, that delivers on the CAN every second the actual position of the vehicle.

Why an open approach? Advantages - disadvantages

When starting a new system design, there is always the question of using an open approach, or using a propriety approach. As this question is independent from he type of system we will discuss it more in detail. First of all:

what is an open approach?

The main feature of an open approach are, but not limited to, three mayor points:

- uses a predefined standard
  You don't have to invent the wheel twice!

- everyone can implement
  Even you!

- different implementations are interoperable
  otherwise, it's not a real standard!

Based on this definition, why should you use an open approach?

why using an open approach?

For using an open approach there are a lot of reasons:

- use off the shelf parts
  you can use soft- and hardware components, already designed and tested by others. You can buy instead of make a lot of components for your system. So, you can get features for your system, you never can afford to develop yourself; and the best: all these features are already tested by others. You will save a lot of debug and testing time.

- build a multi vendor net
  If you want, of have to implement a system, based of components from different vendors, you almost have no other option than using an open approach. And

applications on board of public transport vehicles are typically multi vendor systems. Usually the manufacturer of the driver information system is not the same as the manufacturer of a ticket canceller.

- try resource sharing
  If you want to share some physical resources, let's say an position sensor, for different applications, for example locating the vehicle for the dispatch center and information of passengers with a 'next stop' display, you will need an open approach. If these two tasks, located in devices of different vendors, want to use the same information (the position of the vehicle) you have to use an open approach for your system.

why not?

But even despite of these reasons, there might be some reasons for not using an open approach. One of the most commonly mentioned is the amount of resources needed for implementing.

- resources
  You need more bytes, more microseconds or more CPU power to implement an open approach, than for implementing a customized protocol.
  OK, you may be right! But why not combining the benefits of both, by not customizing a full blown standard by using only the features you need?

- complexity
  The standard has so many features and possibilities! I can't understand all that stuff!
  OK, standards may be difficult to understand! But why not getting help from others, already familiar with it?

- security
  I have to build a secure system, my net is my castle and nobody shall disturb it!
  OK, there may be some systems that need real security (for example the motor management) but if you only want to protect your system from competitors: Why not using the advantages of an open system and protect it (for example by crypting your secrets)?

advantages

As a result of all of these reasons for using and against using, what are the real advantages you can benefit by using an open approach?

- effectivity
  Your cost per feature will drop, because these costs are shared with others!

- time
  You get more features in less time, because they are already predefined and prebuild!

- flexibility
  You can get solutions from different vendors, you have the choice and can use the vendor, that fits most. And you have second sources!

- safety
  You have the safety of using components already used and tested by others!

Standards for the open approach

The layer 1 and 2 of the CAN protocol are defined by the ISO/DIS 11898. Based on this standard, the CiA (CAN in automation) defined a standard for the layer 7 of the OSI model: the CAN application layer (CAL). Furthermore, a CiA working group defined a profile of the CAL for using in mobile applications. The focus of this working group is the use of CAN on board of bus and trucks. Members of this working group are manufacturers of equipment together with truck manufacturer. The aim of this working group is an open standard, with the possibility of a communication of control devices from different vendors.

physical layer

The first thing to standardize is the physical medium to connect the nodes. For the CAN this is done by the ISO/DIS 11898 for high speed applications and by the CiA/DS 102-1. On the market, there are chip solutions for this layer available from different vendors.

data link layer

The next step to standardize is the data link layer. This layer is also specified by the ISO/DIS 11898. The task to implement this specification is already done by a lot of chip suppliers. They, and the licensee of the CAN protocol, guarantee that all CAN chips available can be connected together and that they are compatible to the CAN specification.

application layer

The more difficult part to standardize is the application layer because there is not yet an international standard, nor this layer is specified by the CAN specification. Because of this dissatisfying situation the association CAN in Automation (CiA) specified and recommended an open and powerful application layer for CAN, the CAN application layer (CAL).

CAL defines a language for the description of services for distributed applications. CAL also includes services for the network management like initializing or starting nodes and services for the assignment of the CAN identifiers.

application profile

The CiA working group: 'CAN in mobile applications' defined, based on the pre mentioned standards, an application profile for mobile applications. This profile defines special recommendations for the usage of CAN as an application platform onboard of vehicles.

- bus line and bit rates
- selection of CAL objects and services
- definition of standard variables
- module definition

Furthermore a cold start procedure and a gateway to a CAN network based on J1939 will be defined.

what is already realized in projects

OnTime Engineering is currently involved in projects for a customer, developing a new generation of systems for public traffic vehicles, based on these standards. This means, we already have implemented the CAL based subset of the CAN in mobile applications working group.

Aim: interoperability of different manufacturers

As a result and as the aim of this speech you should remember:

Equipment manufacturers for public transport systems have to provide systems that are at least interoperable. So they have to use at least a common protocol.

To use a common protocol means two possibilities:

1. adapt a standard protocol rather than invent your own
2. use the propriety protocol of a competitor.

If there is enough interest and response for the first possibility, the adaptation of a standard protocol may be defined by an own CiA working group, let’s name it:

‘CAN in public traffic’

You’re welcome to join!