Communication Mechanisms in Automotive Systems

Matthias Stümpfle

Institute of Communications Switching and Data Technics
University of Stuttgart
Seidentraße 36, W-7000 Stuttgart 1
Phone: +49-711-121-2485 Fax: +49-711-121-2477
email: stuempfle@nvdv.e-technik.uni-stuttgart.dbp.de

Extended Abstract

Introduction

Automotive systems as cars, buses, etc. are becoming increasingly complex. Especially the desire for more intelligent communication between electrical and electronical devices is steadily increasing. Cable lengths up to two km for current standard solutions raise costs, the vehicle's weight and its property for easy maintenance. One first step to reduce cost and complexity in nowadays car communication systems is the use of serial buses. Using serial bus systems leads to new problems in designing the electronic system of a vehicle; designers now must be able to construct clean network and layered software architectures.

Classification of communication types

Certain functions of the electronic system of a vehicle are grouped into so called "electronic control units" (ECU). These functions - realized as processes running on processors or microcontrollers- either belong logically or topologically together. A process is defined as a logical sequence of statements to perform a certain functionality. ECUs are physically connected by a serial bus system.

Inside ECUs and between them information must be shared and exchanged to achieve a maximum of performance and usability. Therefore, we distinguish between three kinds of communication forms (see figure 1):

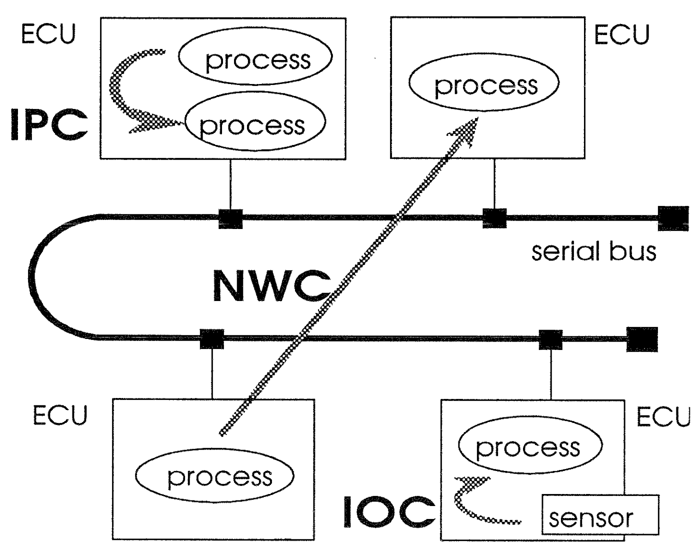

Figure 1: Types of communication

- Interprocess-Communication (IPC) which represents an exchange of information between processes that are located in the same ECU. Processes may either be realized on one processor or on several processors in the ECU.
- Network-Communication (NWC) between related processes located in different ECUs using any sort of underlying serial communication network.
- The third kind is communication between processes and I/O-facilities (IOC), such as sensors and actuators.

Transport system

Application processes running in an ECU may use either sort of the proposed communication facilities to exchange signals. Signals are logical parts of information that are represented by a name to which an actual value may be assigned, similar to the treatment of variables in a programming language. To perform their desired functionality processes may, e.g., send calculated data to other processes or receive information from sensors. The problem now is to easily handle these different kinds of communication facilities. An appropriate solution to this problem is to hide from the application how the information is transported. This idea represents a realization of an OSI transport layer protocol (layer 4). Application processes in so called "end systems" use a provided "transport system" as a facility to send or receive information (see figure 2). From the point of view of the application the transport system is transparent.

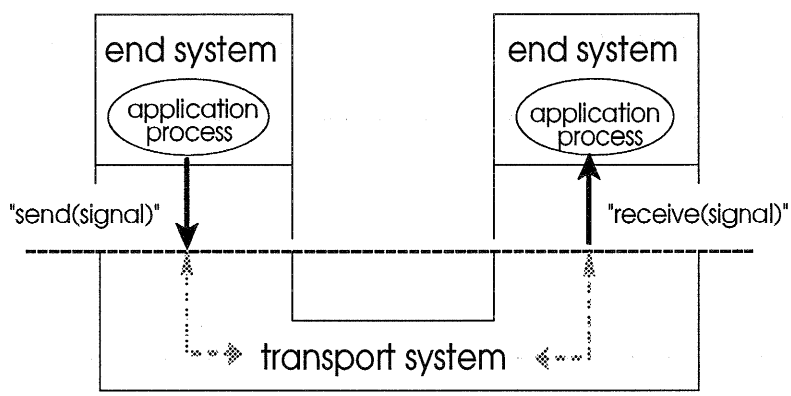

Figure 2: Transparent transport system

Applications must only deal with a certain limited amount of function calls which do not vary upon the chosen transport-system. Simple data transmission calls as "receive" or "send" and some control calls to evaluate or set and change system parameters are sufficient to perform the desired communication between applications and their environment.

Network management

Running a network and providing communication over the network demands any form of network management functions, e.g., connection establishing, switching to other modes, performing fault tolerance functions, etc. To fulfill the requirements of a sophisticated network with determined response times overhead resulting from network management during runtime must be avoided. We consider it most effective to integrate as much management power as possible into the design phase and into the startup phase of the network.

Therefore, the network design phase will be supported by a special description language to specify and establish communication relations and other system parameters. Resulting descriptions then will be evaluated by a specialized compiler. This will also help to save dynamic memory space which is limited.

Only high priority fault tolerance management services, like finding an alternate path or replacing the functionality of a broken ECU by another ECU, should be provided during runtime of the network.

References

HERRTWICH, R.G., HOMMEL, G.: "Kooperation und Konkurrenz"; Springer Verlag, 1989

ISO/OSI-Referenzmodell für offene Systeme, ISO Standard 7498

Kocher, H.: "Intra-car Communication", Proceedings of the 5th Prometheus Workshop, Munich, 1991

STÜMPFLE, M., SCHMID, G.: "Mechanismen der Interprozeß-Kommunikation"; IND, 1992