Communication Bus for Automotive Applications

Stefan Poledna
TTTech Computertechnik AG
Vienna, Austria
poledna@tttech.com

Wolfgang Ettlmayr
TTChip GmbH
Vienna, Austria
ettlmayr@ttchip.com

Markus Novak
TTTech Computertechnik AG
Vienna, Austria
novak@tttech.com

Abstract

Integrated solutions and interconnectivity impose a demanding challenge to automotive industry including requirements, such as safety, high inherent reliability, and fault-tolerance. Drive-by-wire applications imply not only the replacement of mechanical and hydraulic components by electronic systems but it also introduce new functionality, such as collision avoiding cars, enhanced vehicle handling and driving behaviour, and improved vehicle durability.

The Time-Triggered Architecture (TTA) provides a systematic solution to realize these objectives. The primary emphasis in the development of TTP® (Time-Triggered Protocol) was safety at all times which was driven by the experience of aerospace industry and fly-by-wire systems. As an open standard, TTP has been rigorously reviewed and scrutinized by research and science institutions worldwide. This technology in conjunction with the availability of hardware and development software tools has led to the first serial applications of TTP in aerospace, automotive, and railway system applications.

1. Introduction

The number of electronic control units in cars has increased remarkably in recent years. Future x-by-wire systems like brake-by-wire, steer by wire, and the electronic vehicle dynamics control demand networking of components with increased mutual dependency of system functions. This step to an integrated electronics architecture leads to new requirements in the areas of safety, availability, and fault tolerance. The testability, safety of system integration and component updates, as well as diagnoseability in the field have to improve - even in the presence of complexity increases - due to high cost sensitivity.

A communication architecture, which exceeds the limitations of existing event-triggered technology, is essential for these innovations [11][12]. Requirements for new protocols for the next generation of automotive applications include [4]:

- Fault-tolerance to replace mechanical and hydraulic components by computer control
- Provision of a global time base for distributed control algorithms
- Determinism provided by the time-triggered paradigm, i.e., guaranteed latency with minimal jitter
- Bandwidth up to 25 Mbit/sec.
- Uniform bus system within vehicles and support of different physical layers (fiber-optics, electrical)
- Support of scaleable redundancy
- Prompt error detection and error reporting
- Flexibility, expandability and easy configuration in automotive applications.

2. Time-Triggered Architecture

In accordance with these requirements Time-Triggered Architecture (TTA) has been developed during a period of 20 years at Vienna University of Technology [1] together with industrial partners and other leading research institutions. Corner stone of TTA is the Time-Triggered Protocol TTP® [2]. The variant TTP/C refers to safety-critical, fault-tolerant high-speed networks; TTP/A is a low cost derivative for smart sensor and actuator networks [5]. (Fig. 1)

TDMA (time division multiple access) bus access strategy acts as the basic functional principle of TTP [6]. An a priori defined time schedule controls all activities of this communication system. All members of the communication system know their assigned sending slots. A distributed algorithm establishes the global time base with steady clock synchronization [8]. To prevent the so-called babbling idiot problem and slightly-out-of-specification (SOS) failures, TTP/C provides an independent bus-guardian that guarantees exclusive access to the bus.

Precisely defined interfaces in value and time domain ensure the composability of TTP based systems. In consequence components can be developed and tested individually since functionalities established and tested at the subsystem level are guaranteed to work at the system level. This dramatically reduces test and validation efforts [6]. The TTP/C protocol contains a membership service to promptly detect node failures and to detect inconsistent states in a distributed control system [7][10]. It allows implementing an efficient never-give-up (NGU) strategy. TTP/C supports different physical

interconnection structures, a bus configuration and a star or multi-star configuration that can be applied on fiber-optical as well as on electrical physical layers. Based on the a priori known communication schedule, the communication overhead is reduced to a minimum.

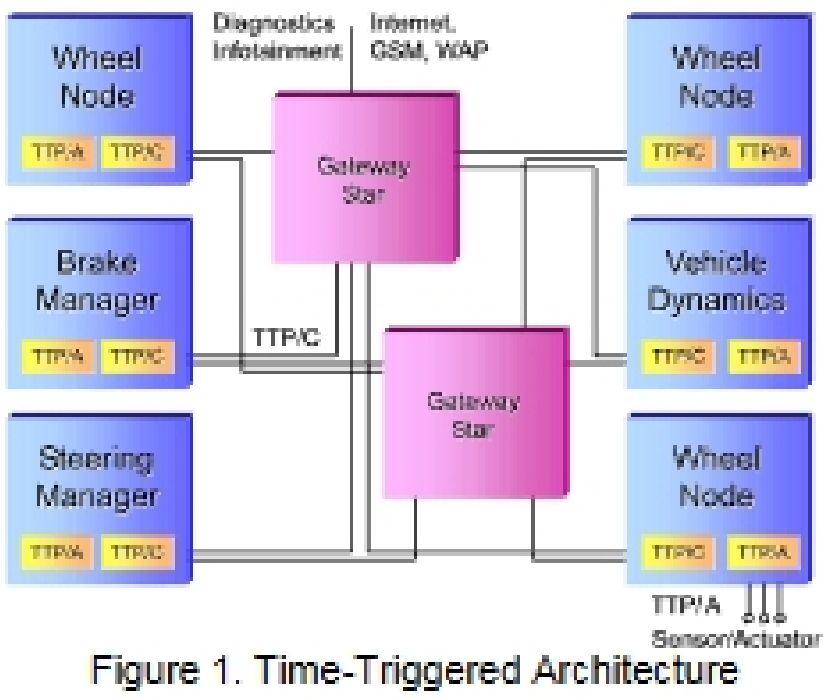

Figure 1. Time-Triggered Architecture

3. TTP Communication

3.1 Time synchronization

The network consists of n nodes. Each node sends at its predefined timeslot and listens to all other nodes.

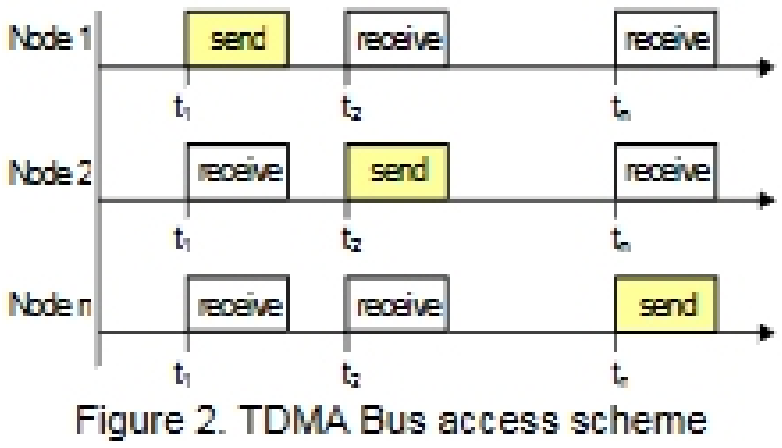

Figure 2. TDMA Bus access scheme

To be able to send at the right time all nodes have to be well synchronized to a global time. Since there is no bus master there is a distributed algorithm for time synchronization. One requirement for this algorithm to work is that every node starts communication at the beginning of its sending slot. Each node reads the schedule from its MEDL (Message description list), so every node knows exactly, when it's time for the other nodes to send.

Node A compares the scheduled time for node B to the actual time, when the message of node B is received and corrects the value by the system known propagation delay from node A to node B. Node A stores this value and does the same for node C. When it has a set of four values (which need not be the last four nodes, it can be any four nodes in the network) it throws away the biggest and the smallest one and corrects its own time by the mean of the remaining two values. This means the nodes are not fully synchronized, but within a specified precision.

3.2 Start-up Communication

At start-up there is no global time and the nodes of the TTP/C network do not wake up at the same time. But every node has a different timeout for sending and receiving by design. So at the beginning there is silence on the bus, everybody is listening. Then the first receive timeout occurs and this node starts communication by sending an I-frame (Initialisation-frame). An I-frame consists of frame header the C-state and a CRC. The C-state contains information of the Global Time, the MEDL position, the current mode, pending mode changes and the membership vector. As soon as the first I-frame was received every node is able to join the communication by sending an I-frame at its scheduled time. After start-up time one node suggests to go to another mode, for instance the application mode. At the end of the current cluster cycle the mode change is executed.

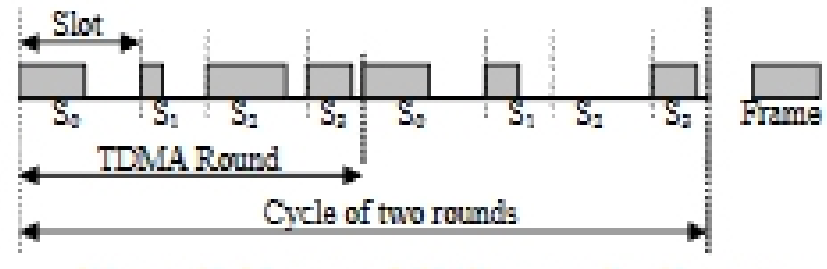

Figure 3. Frames, TDMA rounds, Cycles

The start-up period is the only one where collisions on the bus can happen. If for example node A and node D decide to send at the same time the CRC of that I-frame would be incorrect and therefore the I-frame is invalid. Different timeouts on different nodes ensure that a collision can only take place once at start-up.

3.3 Implicit membership service

In safety critical systems it is important to know which nodes are "alive" and which are not. TTP/C therefore offers an implicit membership service to detect node failures. Every node stores a membership vector and the global time in its so-called C-state. The membership vector has a bit position for each node, up to 64 nodes per cluster. With each frame a CRC is transmitted, which is calculated from the user data and the C-state of the current node. If for example node A has a different C-state than node B, then node A would realize the CRC of B is not correct and therefore reset the membership-bit of B. Node B also would realize that the CRC of A is incorrect and reset the membership-bit of A. If a node detects more nodes with wrong CRCs than correct CRCs during a TDMA round it withdraws itself from the bus and waits for the next I-frame to resynchronise. This is a democratic decision that

prevents cluster building within the system. In case of byzantine faults (frames get disturbed during transmission and not all nodes receive the same data) or slightly-out-of-specification failures (leading to the same problem, that not all nodes receive the same data) at least two clusters are formed. The cluster containing more members will win the competition, all other nodes have to resynchronise. This ensures that parts of the system will continue working even in case of failure.

3.4 Implicit acknowledgement

For self-diagnosis it is important for a node to get feedback from the other nodes. To learn if node B for instance thinks the transmission of node A was ok, node A calculates the CRC for the frame of node B twice: once with the own membership vector bit set to 1 (B thinks A is ok) and second with the own membership vector bit set to 0 (B thinks A is not ok., or B is not ok.) So node A can easily see what the other nodes think of the last transmission of A and take consequences if necessary.

4. System configuration

Independent of the physical layer there are two possible system configurations:

4.1 Bus configuration

In the Bus configuration every node has its own independent bus-guardian, which allows write-access to the shared bus only at scheduled timeslots. This guarantees a collision free communication on the bus.

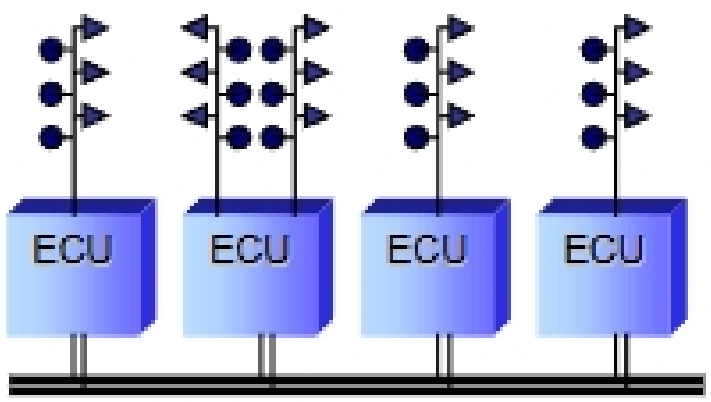

Figure 4. Bus Configuration

4.2 Star configuration

In the Star configuration a central bus-guardian (for each of the two channels) connects the nodes. It allows only one node to speak at a time, according to the MEDL. The star configuration also allows a reshaping of the bus signal for distant nodes.

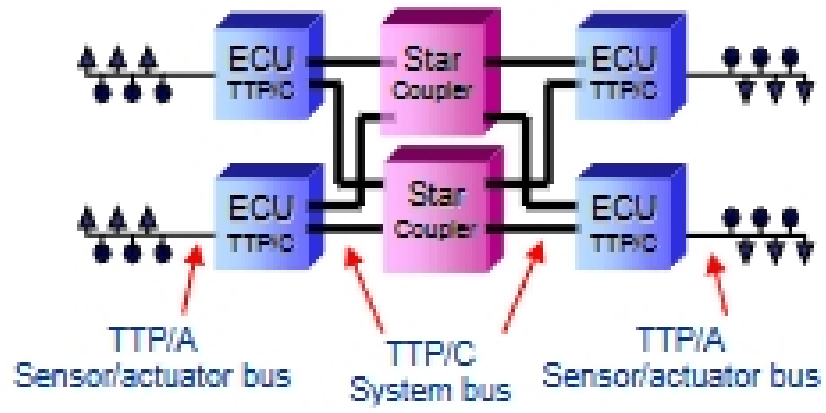

Figure 5. Star Configuration

5. The Host Interface (CNI)

In a distributed system not all nodes necessarily need to work synchronized, but the communication, in case of TTP, has to. This results in at least two clock domains in one node.

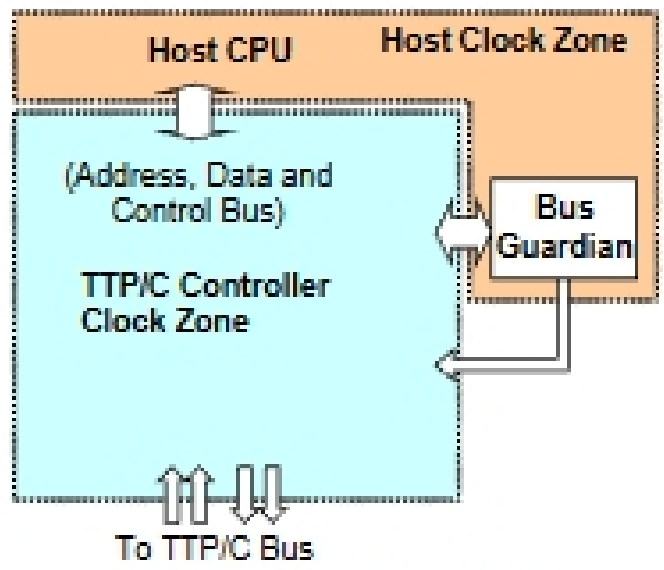

Figure 6. Clock domains

These clock domains are decoupled by temporal firewalls. A temporal firewall works like a like a letter box:

- Unidirectional: you can only 'write' to a letter box, not 'read' messages from it.
- A priory known access scheme: the letter box is emptied at regular, known times (defined by a 'global time') by the 'communication subsystem'.
- State information and validity span: the information content of a letter, valid at the time of writing the letter, stays valid for some amount of time.
- A failed update by the 'writer' is immediately visible: no letter in the letter box.
- By posting a letter (i.e., writing to the firewall), the communication system's timing is not influenced at all.
- BUT: a real letter box can contain several letters. A temporal firewall can not - any update overwrites the previous content of the firewall
- AND: a letter has exactly one receiver. Temporal Firewall data is independent of the receiver and is delivered simultaneously to all receivers in a broadcast system.