A lightweight communication bus based on CAN FD for data exchange with small monolithic actuators and sensors

Fred Rennig, STMicroelectronics Application GmbH

An increasing number of small actuator and sensor devices require a lightweight communication protocol based on and compatible to CAN FD that is integrated in monolithic silicon devices and works without the need of costly external components like crystals. The controller of these actuators uses a subset of the standard CAN FD protocol for communication.

With the increasing computation power of automotive microcontrollers and the upcoming of high data rate networks, data processing and controlling can be combined in single electronic control units (ECU), that move closer to the area where actuators and sensors are placed in the vehicle. A consequence is for example the move towards zone controllers which are placed in various areas (zones) of the car and are connected to the central ECUs by high bandwidth Ethernet networks. These zone controllers are capable to collect sensor data and drive actuators that work with high resolution and agility. The data processing of these sensors and actuators can therefore be moved into the zone controllers and do not need to be placed on the same circuit board. Since the network connection between the actuators and sensors on one side and the controllers on the other is not restricted to a dedicated component, but is routed through the vehicle, a robust, reliable and trusted network is required.

Rear LED topology

An application in which this scheme has been implemented is the animated rear light. Figure 1 shows the communication structure of a modern rear light system. Besides the basic functions like turn and brake indicator and taillight it contains additional features like the ability to inform and warn the traffic behind the car as well as animations for design and branding. These functions and animations are realized by hundreds of individual light sources that need to be dynamically activated with a high resolution of individual intensity. Additionally, diagnosis data must be obtained cyclically to ensure short reaction times on failures.

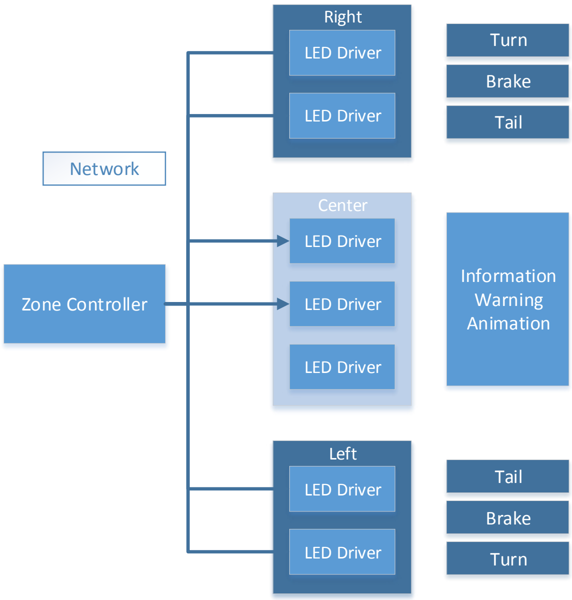

Figure 1 Rear Light Structure

In the shown example the LED drivers are both actuators for the light source and sensors for the diagnosis of component malfunctions.

Communication requirements

The example in Figure 1 uses many individual driver circuits on several printed circuit boards (PCB). The LED drivers are monolithic integrated devices. For efficiency the communication protocol is

also implemented in the same monolithic circuit, additional external components must be avoided, especially costly frequency setting elements like crystals. Due to the integration in a device without microcontroller core higher layer functions that deal with the received data frames need to be implemented in the device as well. These are memory read / write functions and safety features like watchdog treatment and diagnosis data read. Also, error sensing features are needed like detecting device or controller disconnect and bus blocks. Wake-up and sleep management is essential too.
On the other hand, the zone controller implements a CAN FD protocol controller in hardware with an accurate time base and software to control the entire lamp functions. So, using these protocol controllers makes the implementation on the controller side efficient. Additionally, the CAN FD protocol is very well established and has proven its reliability and robustness in vehicle networks for years. With the wide usage of CAN FD in automotive comes a huge experience in all areas of the automotive industry also in manufacturing including the tools and knowledge needed for high quality and efficient manufacturing.

Challenges

From the previously shown arguments CAN FD seems to be the ideal candidate as an interface between zone controllers and monolithic integrated devices. Unfortunately, bus arbitration and data rate switching between arbitration and data segment data rate requires an accurate clock. This clock accuracy made the use of an external clock frequency determining device like a crystal or a ceramic resonator necessary. And the cost of such an element can be in the range of the cost of a small sensor or actuator. While the data rate switching can be avoided by restricting the protocol to use the same data rate for the arbitration and data phase, arbitration is an integral part of the CAN protocol.

Communication peculiarities

This disadvantage can be circumvented when considering the specific system architecture in which these devices are used. Typically for a CAN system is that all bus participants are granted access to the bus while the access priority is determined by the participants' CAN identifiers. Although this makes the network very flexible it is not needed in the specific topology where one central controller and many actuators / sensors are used. The master of the communication is the zone controller. He controls the communication, sends to and requests data from the devices around him. Therefore, the communication master has the control over the bus and assigns bus access to the other devices. With such a master – slave communication scheme arbitration is not needed, and the clock accuracy requirements related to it can be relaxed thanks to the CAN inherited bit stuffing feature that guarantees a minimum edge density.

Protocol

Based on the above a reduced protocol can be used, that consists of the standard identifier, a fixed data rate for the entire frame, CAN FD mode only, the data length code with the fitting data field plus stuff bit count and its parity and resulting cyclic redundancy check field. The remaining bits in the control field are therefore fixed, which results in an arbitration and control field as shown in Figure 2.

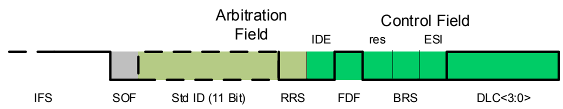

Figure 2 Arbitration and control fields

Since the CAN identifier does not anymore determine the bus access priority it can be used for device addressing and labeling the frame type. In the example of the rear light system the brightness of a high number of light sources must be set within a short time frame for many devices. With the advantage of a bus network and a master controlling the communication the master can send broadcast frames that are received by all bus participants on the bus programming the light sources of several devices using one CAN FD frame with 64-byte payload and low overhead. The master can get diagnosis data as memory content by requesting data from only one device with unicast frames addressed to it. The differences of the frame types and addresses are coded into the frame identifier. While broadcast frames are addressed to a group of devices (“chains”) unicast frames are addressed to only one (“slave- “) device.

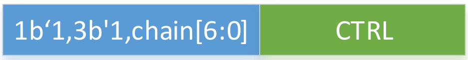

Figure 3 Broadcast identifier

Figure 3 shows the identifier of a broadcast frame with the leading frame type tags and the chain number to which the frame is addressed. Every slave knows the chain it belongs to and the data bytes it must pick from the data field. The chains are assigned during the communication initialization.

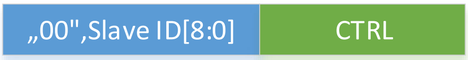

Figure 4 Unicast request identifier

A unicast request identifier is depicted in Figure 4 with the leading bits for the frame type and the address of the slave to which the request is sent

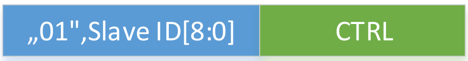

Figure 5 Unicast response identifier

while Figure 5 shows the identifier of the answering slave. Embedded in the unicast data is the memory address to be written or to be read. The response contains the read memory content. With these mechanism actuators can be programmed in an efficient way and specific sensor data can be obtained reliably.

Error handling

The master – slave structure also simplifies the error handling of the communication, because erroneous frames can just be dropped. An implemented watchdog will not be triggered so the slaves react on the loss of communication by e.g. entering a fail-safe state. The unicast request – response scheme detects a disconnected slave or even a disconnected master. And data written by broadcast commands can be read back within time frames satisfying the functional safety fault reaction time requirements. Therefore, error frames are not needed to be sent by the slaves and do not need to be treated by them when sent by the master. This further simplifies the protocol implementation in the monolithic integrated devices.

Implementation

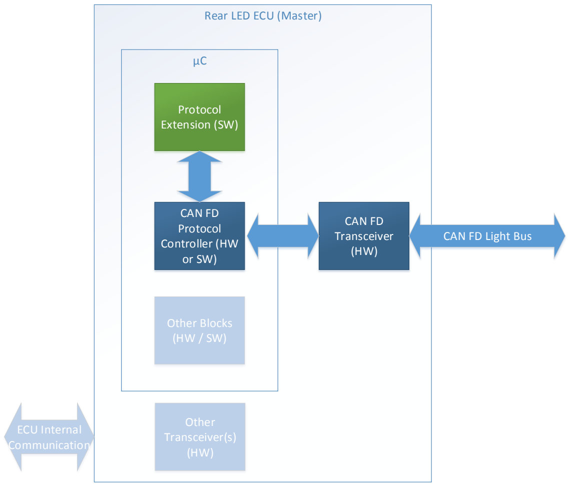

Figure 6 Master Implementation

Figure 6 shows the implementation in the master control unit. It uses the existing CAN FD protocol controller that is usually realized in hardware and an extension for the additional functions like generating the various frame types. Since CAN FD is supported by AUTOSAR the lightweight protocol can be implemented in AUTOSAR. A standard CAN FD transceiver transmits and receives the signals on the communication bus.

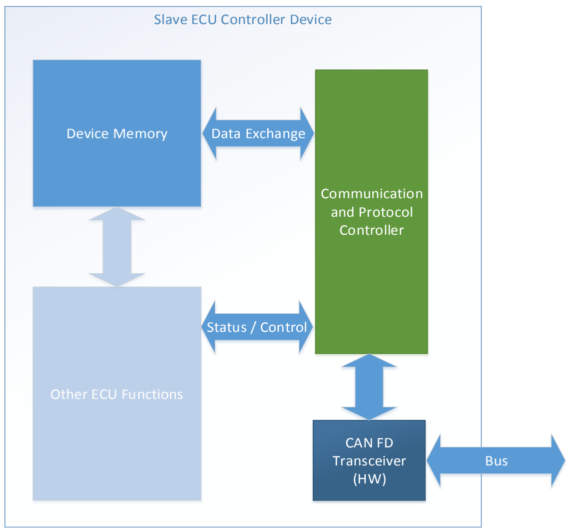

Figure 7 Slave implementation

The communication interface is on the slave side entirely implemented inside of the device including the CAN FD transceiver. Received data is stored in the device memory from which content can be requested by the master via unicast frames. Device functions access these memory locations. In the example of the light driver LED brightness information is stored in dedicated memory addresses and diagnosis data is obtained from this memory.

Functional safety considerations

The unique slave addresses and the frame tags allow to clearly identify the sender and the receiver of each data frame. While the broadcast frames enhance the bus utilization the unicast request and response access each slave from the master individually. By triggering a watchdog in each device communication interrupts can be identified, and the slave may take the appropriate action like entering a fail-safe mode. The frames are protected according to the CAN FD standard with a cyclic redundancy check and the expectation of a response from the slave ensures the detectability of a communication failure. Diagnosis data and memory content written by broadcast frames can be used to verify the correct function and memory content of the slave unit. Based on these capabilities ASIL safety requirements can be fulfilled.

Conclusion

A lightweight CAN FD based communication interface has been presented that can be implemented into single monolithic integrated devices without the need of costly external components like crystals. This is possible due to the controller actuator/sensor interaction that exist in zone-based architectures. With CAN FD as physical layer and the CAN FD protocol as basis a robust automotive grade communication network is available which can use the existing infrastructure like knowledge, experience, measurement equipment and other tools. And it can also fulfill ASIL functional safety requirements. A working application example in which this communication scheme is already implemented is an animated rear light system for cars. By using a CAN FD data rate of 1 Mb/s and 64-byte payload it is possible to drive 4000 individual light sources with an accuracy of 8 bit per pixel at a refresh rate of 50 ms plus diagnosis messages.

Fred Rennig
STMicroelectronics Application GmbH
Bahnhofstr. 18
DE-85609 Aschheim-Dornach
www.st.com

References

[1] INTERNATIONAL STANDARD ISO 11898-1 (second Edition, 2015-12-15)