MOST ECL for Diagnosis

Ring break diagnosis or other diagnosis through the electronic control line

To detect faults in a MOST system, one possibility is to run tests through an Electronic Control Line (ECL). The optional specification designed by the MOST Cooperation provides mechanisms to trigger and collect the results of different tests.

By Ben Sahl

Due to the fact that MOST is based on a ring topology, there are several possibilities of faults. For the optical transmission, typical ones are the Sudden Signal Off (SSO), the Critical Unlock (CU), and the ring break. In case of a ring break, no communication is possible anymore and SSO and CU can seriously disturb the signal transmission.

If no reliable MOST communication is possible, tests like e.g. Ring Break Diagnosis (RBD) need to be triggered and the results have to be collected. With the first generation of MOST25, three general concepts for RBD have already emerged as possible solutions to analyze a broken ring:

- 1. Concept: Run RBD on every start-up. In this case every device in the ring starts in RBD mode on every start-up of the system. Advantage: For this solution no additional hardware is required, only the software has to be configured. As a result, this solution does not require a lot of effort. Disadvantages: The whole system starts in RBD mode every time. The result can only provide one error at the time. If there are more defects in the system, the complete test procedure has to be repeated after fixing the detected error, until communication is stable again. In addition, this solution provides no information whether a device is not working at all, e.g. through an electrical defect, or if the optical fiber between two devices is damaged.
- 2. Concept: Run RBD on first switch-on. For this solution the hardware has to be able to detect whether the power supply has been removed for more than 30 seconds. Only if this is the case, the system starts in RBD mode like in the first solution. Advantage: No loss of time on regular start-up compared to solution 1. Disadvantages: Like in solution 1, only one defect is discoverable on each test-run. Again a conclusion about the electrical functionality of a device is not possible. Additional logic needs to be integrated into hardware, to detect first switch-on. To trigger the test, the power supply has to be removed and reattached for all MOST devices at the same time, e.g. by disconnecting and reconnecting the battery.
- 3. Concept: Run RBD through an additional electrical line. The RBD is triggered through a signal on a so-called ECL. In addition to triggering the test it is possible to collect the results on this line. Advantages: Availability of the test result for every single device on one test run. In addition to the optical test result, information about the electrical functionality is also received. Beyond that, this line can be used for waking up the system or other tests and diagnosis as well. Disadvantages: The additional hardware and software can be quite complex according to the requirements and due to restrictive timing requirements. Also, testing and integrating the whole system with all devices is an additional challenge.

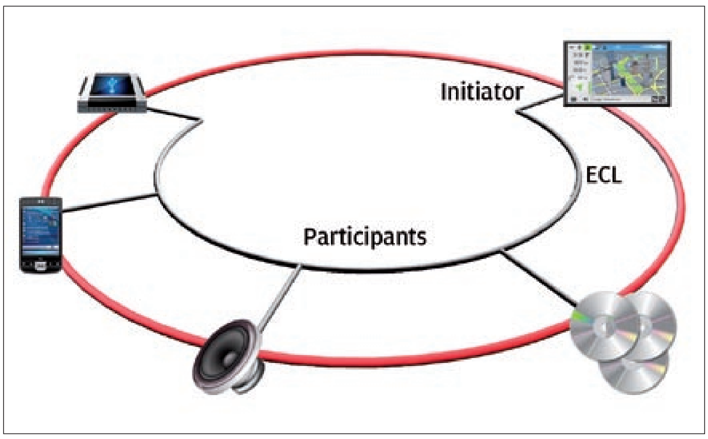

Figure 1. Overview of a typical MOST ECL system.

Diagnosis when communication isn't possible

The ECL provides the ability to run highly sophisticated diagnostics in a MOST ring, even when normal MOST communication is not possible due to a defect in the system. In the past the ECL concept was realized by OEMs in different ways. For MOST 3Vx the MOST Cooperation WG Diagnosis Adaptation has released an optional specification describing one solution [1]. The basic idea is to have one initiator in the system to trigger a test on all connected devices through the ECL (**figure 1**). If a device is in sleep mode, it has to be woken up, as the optical signal may be interrupted. After running the test, every „participant" has a defined timeslot to send its result. The specification provides the hardware requirements and the proprietary protocol, and already contains a rough software design.

The ECL itself is a one wire line. The physical requirements are defined in a way that makes it possible to realize it as a discrete circuit or to use a LIN-like transceiver instead. If using a LIN-like transceiver it is important to be aware of the duration of low-level signals within the protocol. Some LIN-like transceivers have a dominant time-out function, which avoids pulling down the signal for a longer period. As this article focuses on the concept and the software point of view, the hardware aspects will not be examined in depth. In general, the system integrator can provide the information which LIN-like transceivers need to fulfil the requirements for ECL. Even though LIN-like transceivers might be used, the LIN protocol is not used at all; MOST Cooperation has specified a simple and fully proprietary protocol.

Every MOST device has to be connected to the ECL and can be woken up by this line. The ECL works with an active low logic. All information of the specified protocol is based on timing (**figure 2**). The complete system test, is split up in three phases: start sequence, parameter sequence and result sequence. The ECL System consists of the initiator, which triggers the sequence, and the participants, which have to react to the data sent by the initiator. There is only one initiator allowed. During start and parameter sequence, the initiator acts as the sender and the participants are not allowed to pull down the signal line. During the result sequence, roles are changed and each participant acts as a sender during its specific timeslot and the initiator as the receiver.

The start sequence begins with one or more start impulses. Optional retries are available in case a participant cannot recognize the first impulse due to sleep mode or other restrictions. E.g. a device can wake up on the rising edge of the first start impulse due to hardware restrictions. To be sure that the software can recognize a valid start sequence, the system integrator may decide to use two or more impulses. These pulses are followed by a timeout, which has to be defined by the system integrator. During this time all devices have to wake up and be able to recognize the following sequence.

The parameter sequence, which the initiator is sending, starts with an initial low period for synchronization. This is followed by six timeslots containing the bit combination (parameters P1 – P5) and a synchronization bit (always low), informing every participant, which test has to be executed. Depending on these parameters, a variable pause follows to run the test during this time. For synchronization, the parameter sequence is closed by a defined low level period.

In the latest specification, four different tests are defined. They can be classified in two groups:

- Long run tests: In the case of a long run, there is a slot of several seconds reserved for test execution. Currently, long run tests execute the RBD of the network service and provide a result for an optical test.
- Short run tests: For a short run, the sequence continues after a shorter timeout. The short tests check for electrical functionality or SSO/CU.

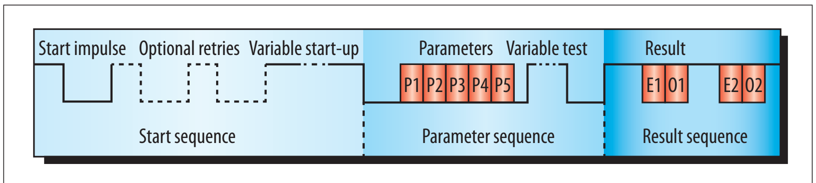

Figure 2. Definition of the ECL protocol.

System integrators have the ability to define further tests for their systems.

During the result sequence, all participants send their answers to the initiator. The system integrator has to define one ECL node class for every MOST node. Depending on this class, each participant has to send its test result in a corresponding timeslot. The answer contains two bits. The first bit (En) gives the initiator information about the electrical functionality of the device. If the participant is alive, it has to pull down the signal. The second bit (On) provides the answer to the test, which was triggered by the initiator in the parameter sequence.

Software architecture

Based on the Electrical Control Line Specification released by the MOST Cooperation, EB (Elektrobit) has implemented a software solution, which is easy to adapt to the requirements of a device, both for initiators and participants. The specification already provides a software design, which gives a good idea of the requirements. The most important aspect is the timing behaviour, due to fact that the protocol is running on a timing base. Even though there are defined tolerances in the specification, it can be challenging depending on the software architecture regarding the timers in the device. The solution provided by EB covers all important aspects and reduces the integration effort to a minimum. For both initiator and participant only the corresponding interfaces for application, test application, timers and ECL hardware driver have to be implemented (**figure 3**). The state machines covering the ECL protocol are completely enclosed inside the initiator and participant core. *sj*

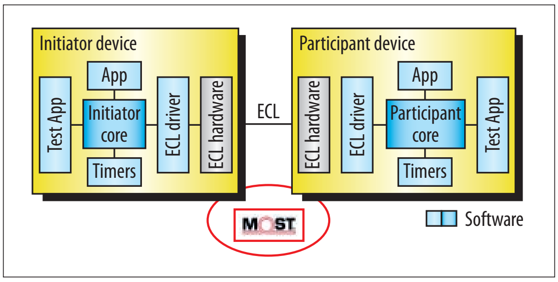

Figure 3. Architectural view of ECL.

Dipl.-Inf. (FH) Ben Sahl
is Team Manager for Infotainment Embedded OS and Connectivity at Elektrobit Automotive GmbH in Erlangen (Germany). In 2005 he was certified as SMSC MOST Consultant (SMC).