Agilent
J8115A LIN Tester
Error-injection in LIN Tester
Application Note
Introduction

This document outlines the usage cases for the error-injection capabilities of the LIN Tester. In order to test the robustness of a LIN implementation and the conformance to the LIN specification, it is essential that the LIN setup is tested with a controlled, programmable error injection to verify a correct node implementation.

All errors defined within the LIN specification document can be generated and detected using the Agilent LIN tester.

Agilent LIN Tester software

The Agilent LIN Tester software allows the user to control the J8115A LIN Tester in a basic and advanced emulation mode.

The advanced emulation capabilities of the Agilent LIN Tester make it possible to create real-time behavior in the LIN Tester’s node emulation. This includes modification of emulated signals, switching of schedule tables in real-time, and the introduction of errors in the LIN communication (e.g. incorrectly calculated checksum bytes). The user describes the functionality in a separate control file, the LIN Emulation Control file (LEC file).

This capability is critical to make sure that thorough conformance testing of a LIN network or node can be performed. This application note demonstrates how the J8115A LIN Tester can be used to insert different errors to test LIN networks for robustness.

Set transfer parameters

Using the advanced emulation capabilities of the LIN Tester it is possible to inject errors in a frame by setting the number of different transfer parameters in the LEC file.

The parameters listed in the section below describe the different errors that the LIN Tester can produce, and also how the LIN Tester detects the caused errors.

The intention of this reference is to understand which requirements can be tested by using error-injection in the LIN Tester.

Errors detected

The LIN Tester can detect the following errors:

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ECS</td>
<td>CheckSum Error</td>
</tr>
<tr>
<td></td>
<td>Wrong checksum</td>
</tr>
<tr>
<td>EFR</td>
<td>Byte Framing Error</td>
</tr>
<tr>
<td></td>
<td>Stop bit not detected</td>
</tr>
<tr>
<td>ENA</td>
<td>No Answer</td>
</tr>
<tr>
<td></td>
<td>No response on header of a frame</td>
</tr>
<tr>
<td>ESA</td>
<td>Short Answer</td>
</tr>
<tr>
<td></td>
<td>Less bytes received than expected</td>
</tr>
<tr>
<td>E55</td>
<td>Synchronization byte Error</td>
</tr>
<tr>
<td></td>
<td>Synch byte is not 0x55</td>
</tr>
<tr>
<td>EID</td>
<td>Frame ID Error</td>
</tr>
<tr>
<td></td>
<td>An ID that is not found in the LDF</td>
</tr>
<tr>
<td>ETB</td>
<td>Break High Time Error</td>
</tr>
<tr>
<td></td>
<td>Wrong Break delimiter time</td>
</tr>
<tr>
<td>ETH</td>
<td>Header Time Error</td>
</tr>
<tr>
<td></td>
<td>Header time too long</td>
</tr>
<tr>
<td>ETF</td>
<td>Frame Time Error</td>
</tr>
<tr>
<td></td>
<td>Frame too long</td>
</tr>
<tr>
<td>EUB</td>
<td>Unidentified Byte(s)</td>
</tr>
<tr>
<td></td>
<td>Byte received that was not expected</td>
</tr>
<tr>
<td>IDL</td>
<td>Bus Idle Time-out</td>
</tr>
<tr>
<td></td>
<td>No communication detected</td>
</tr>
<tr>
<td>BNZ</td>
<td>Bus Error</td>
</tr>
<tr>
<td></td>
<td>Physical bus error (e.g. shortcut to ground)</td>
</tr>
</tbody>
</table>

These errors are detected and can be used in the trigger of the LIN Tester software. These are shown as errors in the trace window in the error code column.
The LIN Frame consists of four key elements:

- **Synch Break** – The Synch Break is used to identify the beginning of a packet.

- **Synch Field** – The Synch Field is used for clock synchronization.

- **Identifier Field** – The Identifier Field is sent by the master node to all LIN nodes and contains one of 64 different values and 2 parity bits in the 8 bit data. The identifier is normally associated with a collection of signals that are subsequently transmitted on the LIN bus.

- **Data Field / Message response** – The Data Field response is either sent out by the master or by one of the slaves.

Figure 1 shows a sample LIN Frame taken by an oscilloscope with 4 Data Bytes and a checksum at the end. To illustrate the byte boundaries, different colors per byte are used.

Depending on the role as master or slave a different list of errors can be inserted. The master always issues the synch break, the synch byte and the ID. The following data bytes can either be inserted from a slave on the bus (that is responsible to fulfill this ID) or by the slave node within the master itself.

The master also controls the schedule and possible alternative schedules. All schedule related tests have to be performed by the master node.
Master only errors

The following list of error codes demonstrates the capability of the J8115A LIN Tester to insert errors and test the robustness of a LIN communication network.

**SET_BREAK_DOMINANT_TIME**

With this command, you can set the break dominant time (Break Low) through the LEC.

```
Set_Transfer_Param (Frame_name, SET_BREAK_DOMINANT_TIME, );    //time in ms (d.dd)
```

Use the pop-up menu in the LEC Editor Nodes-Frames-Signals section by right click on the frame, which behavior you would like to affect, and choose Set Transfer Param (Header) Set Break Dominant Time and the line above will be inserted with the appropriate frame name to the text. Write the time (in ms) you would like to set, as the last parameter (after the last comma).

The break dominant time will be rounded to 10 microseconds resolution. If the set parameter is too low, you will see the EUB Error Code in the trace window, if too high, the ETH and EUB or if more high the E55 error code.

The results are shown as an oscilloscope capture along with corresponding trace window display from LIN Tester. This continues throughout the application note.

**SET_BREAK_DELIMITER_TIME**

With this command, you can set the break delimiter time (Break High) through the LEC.

```
Set_Transfer_Param (Frame_name, SET_BREAK_DELIMITER_TIME, );    //time in ms (d.dd)
```

Use the pop-up menu in the LEC Editor Nodes-Frames-Signals section by right click on the frame, which behavior you would like to affect, and select Set Transfer Param (Header) Set Break Delimiter Time and the line above will be inserted with the appropriate frame name to the text. Write the time (in ms) you would like to set, as the last parameter (after the last comma).

The break delimiter time will be rounded to bit time resolution according to the actual Baud rate.
Master only errors

**SET_HEADER_INTERBYTE_TIME**

With this command, you can set the header interbyte time through the LEC.

```
Set_Transfer_Param(Frame_name, SET_HEADER_INTERBYTE_TIME, );  //time in ms (d.dd)
```

Use the pop-up menu in the LEC Editor Nodes-Frames-Signals section by right click on the frame, which behavior you would like to affect, and choose Set Transfer Param (Header) Set Header Interbyte Time and the line above will be inserted with the appropriate frame name to the text. Write the time (in ms) you would like to set, as the last parameter (after the last comma).

The header interbyte time will be rounded to bit time resolution according to the actual Baud rate.

If the set parameter is too high, you will see the ETH and EUB or if more high the EID Error Code in the Trace window.

**CHANGE_SYNCHRON_BYTE**

With this command, you can set the synchron byte (Sync Field) value through the LEC.

```
Set_Transfer_Param(Frame Name, CHANGE_SYNCHRON_BYTE, );  //new value of byte
```

Use the pop-up menu in the LEC Editor Nodes-Frames-Signals section by right click on the frame, which behavior you would like to affect, and choose Set Transfer Param (Header) Change Synchron Byte and the line above will be inserted with the appropriate frame name to the text. Write the new synchron byte value you would like to set, as the last parameter (after the last comma).

If the Synchron byte is not equal with 55, you will be see the E55 and EUB Error Code in the Trace window.
Master only errors

FRAMING_ERROR_SYNCHRON

This command will cause a framing error, by sending a wrong stop bit in the synchron byte.

Set_Transfer_Param(Frame_name, FRAMING_ERROR_SYNCHRON, 0); //no parameter

Use the pop-up menu in the LEC Editor Nodes-Frames-Signals section by right click on the frame, which behavior you would like to affect, and choose Set Transfer Param (Header) Framing Error Synchron Byte and the line above will be inserted with the appropriate frame name to the text.

You will see the EFR and EUB Error Codes in the Trace window.

CHANGE_ID_BYTE

With this command, you can set the ID byte (Ident Field) value through the LEC.

Set_Transfer_Param(Frame_name, CHANGE_ID_BYTE, ); //new value of byte

Use the pop-up menu in the LEC Editor Nodes-Frames-Signals section by right click on the frame, which behavior you would like to affect, and choose Set Transfer Param (Header) Change ID Byte. The line above will be inserted with the appropriate frame name to the text. Write the new ID byte value, you would like to set, as the last parameter (after the last comma).

If the ID is not appropriate, you will be see the EID and EUB Error Codes in the Trace window.
Master only errors

**FRAMING_ERROR_ID**

This command causes a framing error; in fact send a wrong stop bit in the ID byte.

```plaintext
Set_Transfer_Param (Frame_name, FRAMING_ERROR_ID, 0);   //no parameter
```

Use the pop-up menu in the LEC Editor Nodes-Frames-Signals section by right click on the frame, which behavior you would like to affect, and choose Set Transfer Param (Header) Framing Error ID. The line above will be inserted with the appropriate frame name to the text.

You will see the EFR and EUB Error Codes in the Trace window.

![Image of Trace Window](image1)

![Image of Oscilloscope](image2)
Master and Slave errors

**SEND_NO_ANSWER**

This command sends no answer with the frame.

```
Set_Transfer_Param(Frame_name, SEND_NO_ANSWER, 0);
//no parameter
```

Use the pop-up menu in the LEC Editor Nodes-Frames-Signals section by right click on the frame, which behavior you would like to affect, and choose Set Transfer Param (Response) Send No Answer and the line above will be inserted with the appropriate frame name to the text.

You will see the ENA Error Code in the Trace window.

**SEND_NO_CHECKSUM**

This command sends no checksum with the frame.

```
Set_Transfer_Param(Frame_name, SEND_NO_CHKSUM, 0);
//no parameter
```

Use the pop-up menu in the LEC Editor Nodes-Frames-Signals section by right click on the frame, which behavior you would like to affect, and choose Set Transfer Param (Response) Send No Checksum and the line above will be inserted with the appropriate Checksum to the text.

You will see the ESA Error Code in the Trace window.
**Master and Slave errors**

**SEND_WRONG_CHKSUM**

This command sends wrong checksum with the frame.

```
Set_Transfer_Param(Frame_name, SEND_WRONG_CHKSUM, 0);    //no parameter
```

Use the pop-up menu in the LEC Editor Nodes-Frames-Signals section by right click on the frame, which behavior you would like to affect, and choose Set Transfer Param (Response) Send Wrong Checksum and the line above will be inserted with the appropriate frame name to the text.

You will see the ECS Error Code in the Trace window.

**SEND_EXTRA_BYTE**

This command sends extra byte with the frame.

```
Set_Transfer_Param(Frame_name, SEND_EXTRA_BYTE, );    // value of byte
```

Use the pop-up menu in the LEC Editor Nodes-Frames-Signals section by right click on the frame, which behavior you would like to affect, and choose Set Transfer Param (Response) Send Extra Byte and the line above will be inserted with the appropriate frame name to the text.

Write the new byte value you would like to set, as the last parameter (after the last comma).

You will see the EUB Error Code in the Trace window.
Master and Slave errors

**SET_INFRAME_RESPONSE_TIME**

With this command, you can influence the Inframe response time (Inframe Response Space) time through the LEC.

\[\text{Set\_Transfer\_Param(\text{Frame\_name}, \text{SET\_INFRAME\_RESPONSE\_TIME},);} \quad \text{//time in ms (d.dd)}\]

Use the pop-up menu in the LEC Editor Nodes-Frames-Signals section by right click on the frame, which behavior you would like to affect, and choose Set Transfer Param (Response) Set Inframe Response Time and the line above will be inserted with the appropriate frame name to the text. Write the time (in ms) you would like to set, as the last parameter (after the last comma).

If the set time is too high you will see one of the following Error Codes in the Trace window: ETF, ESA, ENA, depending on the length of the Inframe response time.

**SET_SLAVE_INTERBYTE_TIME**

With this command, you can set the response interbyte time through the LEC.

\[\text{Set\_Transfer\_Param(\text{Frame\_name}, \text{SET\_SLAVE\_INTERBYTE\_TIME},);} \quad \text{//time in ms (d.dd)}\]

Use the pop-up menu in the LEC Editor Nodes-Frames-Signals section by right click on the frame, which behavior you would like to affect, and choose Set Transfer Param (Response) Set Slave Interbyte Time and the line above will be inserted with the appropriate frame name to the text. Write the time (in ms) you would like to set, as the last parameter (after the last comma).

If the set parameter is too high, you will see the ETF or ESA Error Code in the Trace window, depends on the length of the response interbyte time.
Master and Slave errors

**FRAMING_ERROR_FIRSTBYTE**

This command causes a framing error in the first response byte; in fact send a wrong stop bit in the first byte.

```
Set_Transfer_Param(Frame_name, FRAMING_ERROR_FIRSTBYTE, 0);  //no parameter
```

Use the pop-up menu in the LEC Editor Nodes-Frames-Signals section by right click on the frame, which behavior you would like to affect, and choose Set Transfer Param (Response) Framing Error First Response Time and the line above will be inserted with the appropriate frame name to the text.

You will see the EFR Error Code in the Trace window.

**FRAMING_ERROR_LASTBYTE**

This command causes a framing error in the last response byte.

The LIN Tester reports the EFR error code. [LIN Protocol Specification revision 2.0: 4.2.2 Frame processor] and [LIN Protocol Specification revision 2.0: 6 Status Management]
Master and Slave errors

**CHANGE_TRANSFER_SPEED**

With this command, you can set the speed of the response through the LEC.

```
Set_Transfer_Param(Frame_name, CHANGE_TRANSFER_SPEED, );    //new speed in kbps (dd.ddd)
```

Use the pop-up menu in the LEC Editor Nodes-Frames-Signals section by right click on the frame, which behavior you would like to affect, and choose Set Transfer Param (Response) Change Transfer Speed and the line above will be inserted with the appropriate frame name to the text.

Write the speed (in kbps) you would like to set, as the last parameter (after the last comma).

The minimum speed is 5 kbps and the maximum is 20.161 kbps.

You will see errors in the Trace window, if the difference between the set transfer speed and the baud rate is more than approximately 5%.

**SEND_ANSWER**

This command sends an answer to the frame (defined by Frame_name). This command has importance, if the slave node, which has to respond to the frame, is not emulated.

```
Set_Transfer_Param(Frame_name, SEND_ANSWER, 0);    //no parameter
```

Use the pop-up menu in the LEC Editor Nodes-Frames-Signals section by right click on the frame, which behavior you would like to affect, and choose Set Transfer Param (Response) Send Answer and the line above will be inserted with the appropriate frame name to the text.
Master and Slave errors

**SET_FRAME_DELAY**

With this command, you can set the Frame Delay time through the LEC.

```c
Set_Transfer_Param(Frame_name, SET_FRAME_DELAY, 20); // delay in timebase
```

Use the pop-up menu in the LEC Editor Nodes-Frames-Signals section by right click on the frame, which behavior you would like to affect, and choose Set Transfer Param (Header) Set Delay and the line above will be inserted with the appropriate frame name to the text. Write the time (in timebase, defined in the LDF file, under the Node definition in the Master sub-class) you would like to set, as the last parameter (after the last comma).

If the frame becomes too short because of the set parameter, you will see the ESA Error Code in the Trace window, if it is too long, and the Bus IdleTimeOut is elapsed, the IDL error code appears in the Trace window.

Summary

This extensive and detailed list shows how the user can use the LIN Tester to check the robustness of his LIN network / node implementation against various failures.

The error injection capabilities are very powerful and allow controlling most of the relevant parameters of the LIN protocol.

**Related literature**

<table>
<thead>
<tr>
<th>Publication title</th>
<th>Publication type</th>
<th>Publication number</th>
</tr>
</thead>
<tbody>
<tr>
<td>J8120A VPT 501</td>
<td>Data Sheet</td>
<td>5989-6818EN</td>
</tr>
<tr>
<td>J8115A LIN Tester</td>
<td>Data Sheet</td>
<td>5989-6817EN</td>
</tr>
</tbody>
</table>

**Product Web site**

For the most up-to-date and complete application and product information, please visit our product Web site at: [www.agilent.com/find/automotive-network](http://www.agilent.com/find/automotive-network)
Remove all doubt

Our repair and calibration services will get your equipment back to you, performing like new, when promised. You will get full value out of your Agilent equipment throughout its lifetime. Your equipment will be serviced by Agilent-trained technicians using the latest factory calibration procedures, automated repair diagnostics and genuine parts. You will always have the utmost confidence in your measurements.

Agilent offers a wide range of additional expert test and measurement services for your equipment, including initial start-up assistance onsite education and training, as well as design, system integration, and project management.

For more information on repair and calibration services, go to www.agilent.com/find/removealldoubt