**CAN E/F issue on Bus**

Issue :

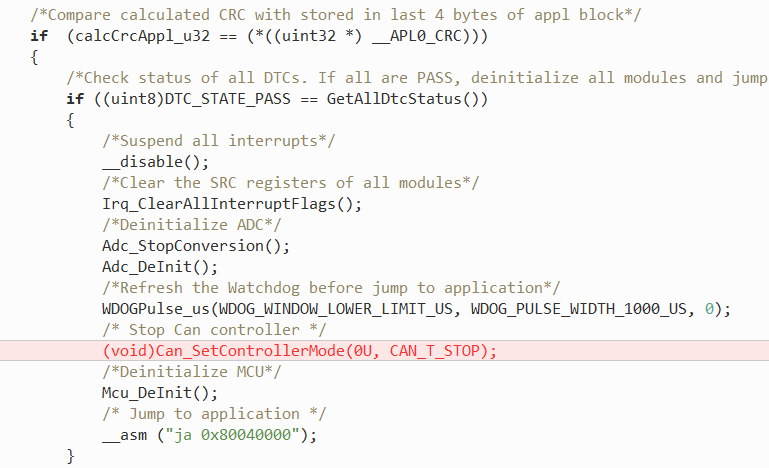
After restarting the FrCam(while other nodes on the BUS are active in transmission), CAN E/F are being transmitted for ~200ms from FrCam

Root cause:

CAN controller is not getting stopped in boot, before it get re-initialize again in Application. Because of this re-initialization

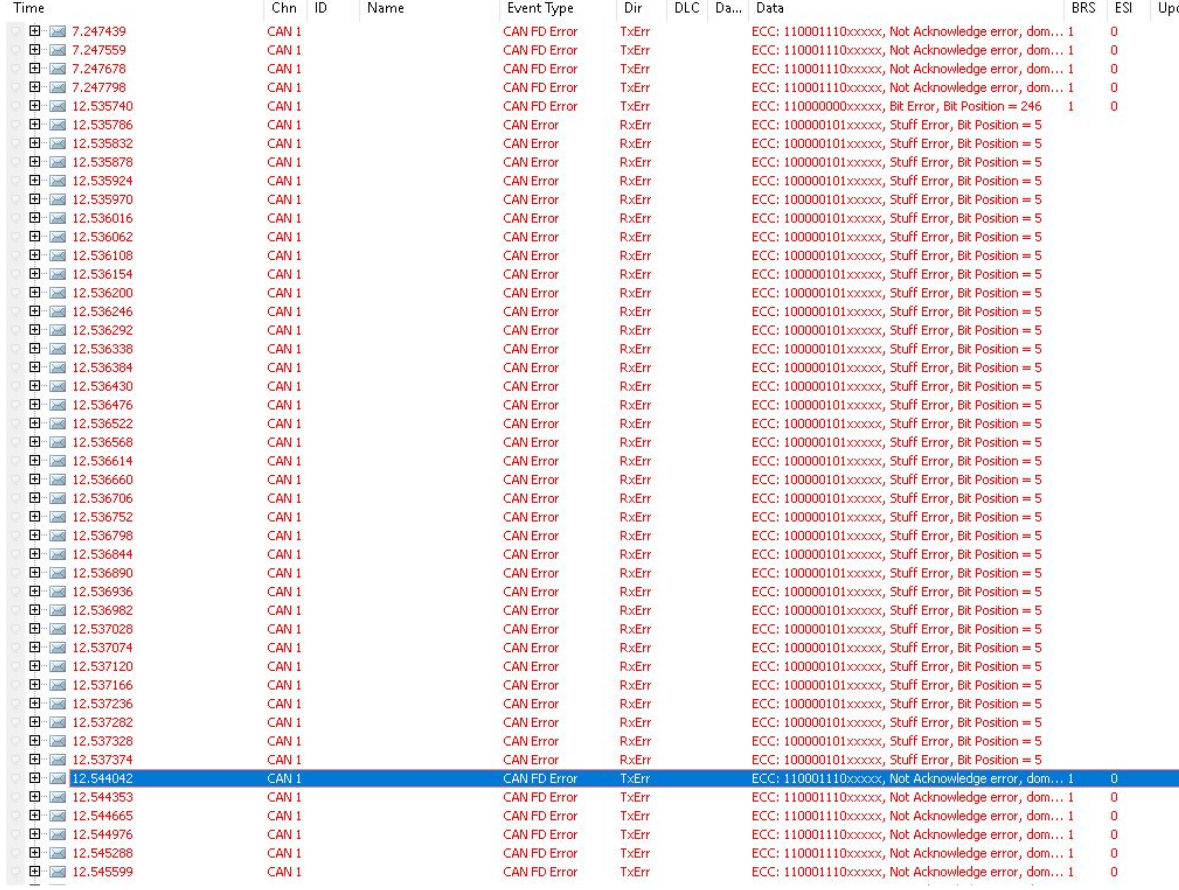
CAN Bus is getting disturbed and causing to Bit Error/Form Error and from there on to stuff error.

Solution:



Results:

Before Fix:



After Fix:



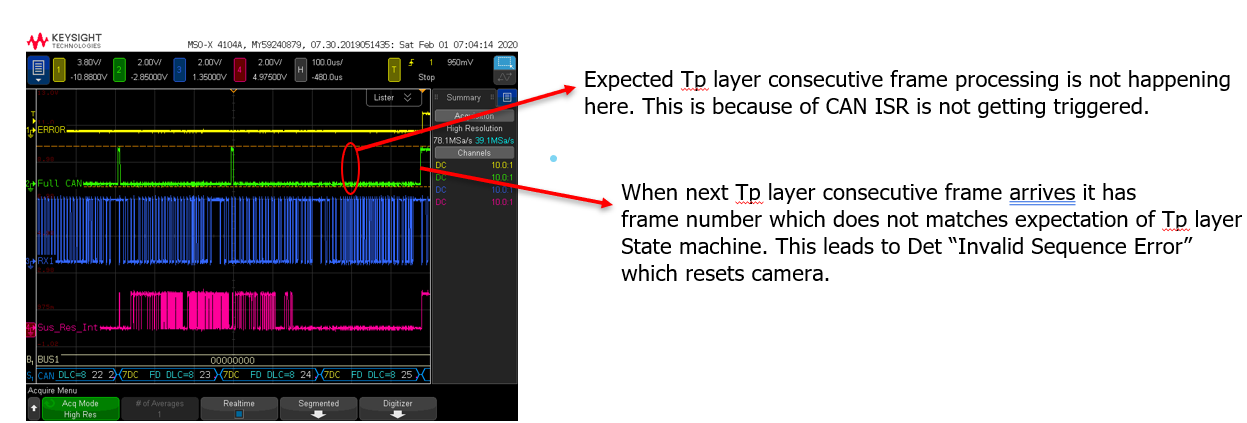
**Consecutive TP frame missing Investigation**

**Issue description**:

OEM reported that camera is getting reset during writing of DID 2E4024.

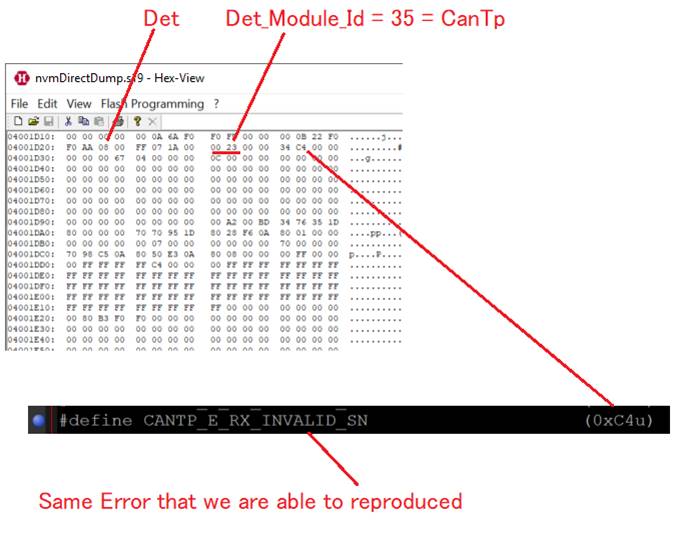
**Issue analysis**:

One Tp consecutive frame is missing in between leading to Det “Invalid Sequence Error”. This Det is causing reset



**Issue confirmation**:

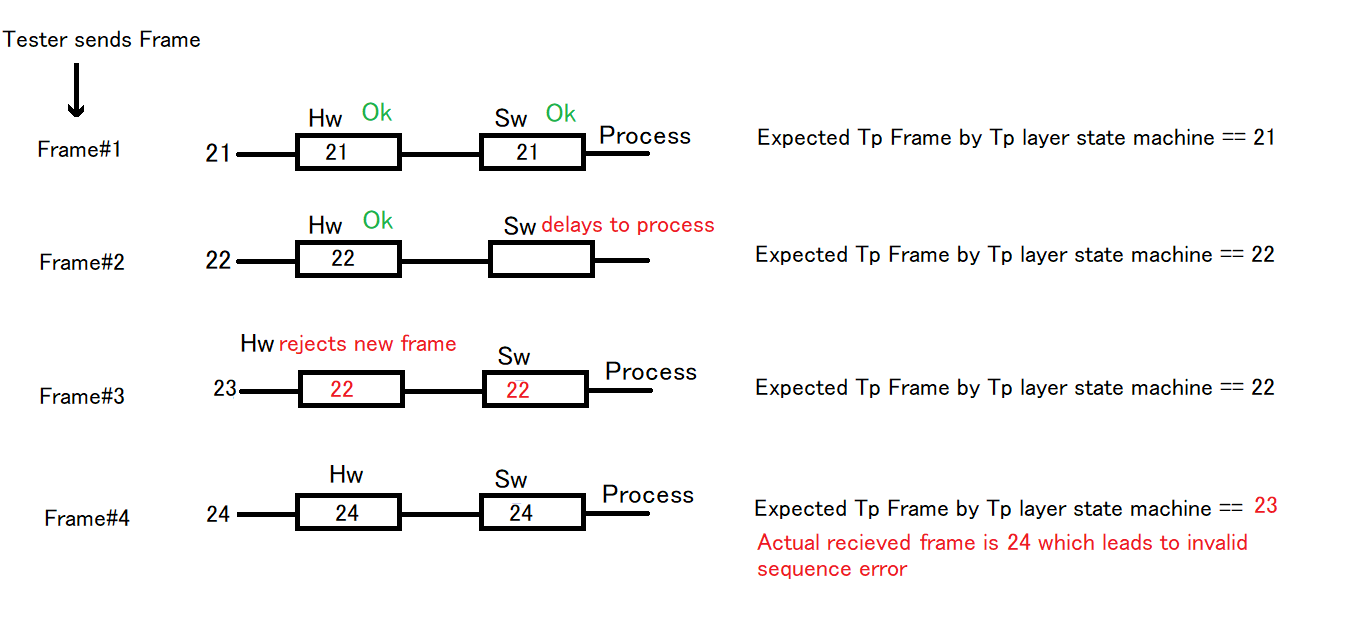
This issue is confirmed by NvM dump from Nissan module as shown in following picture



**Root cause:**

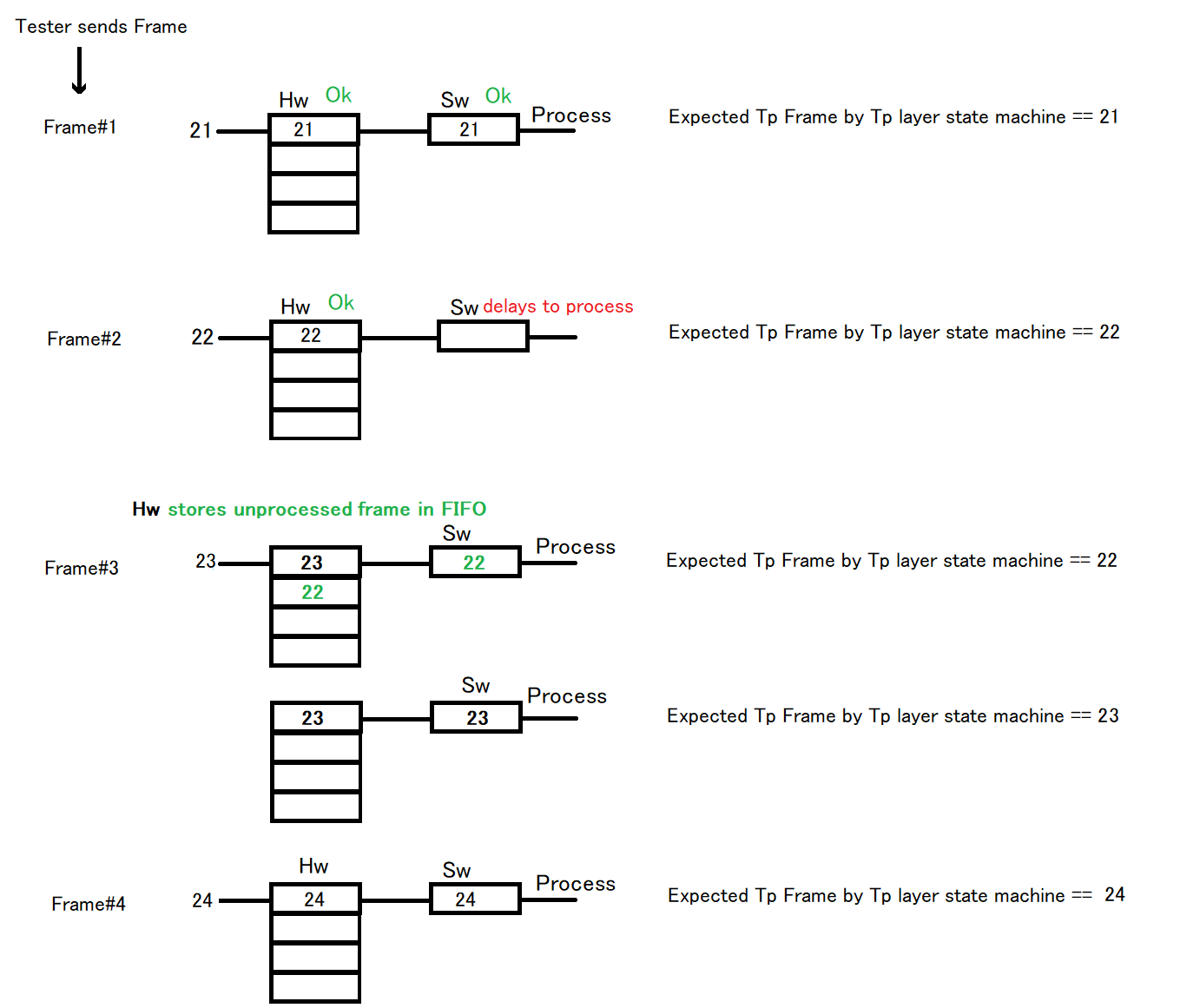
With BS = 0, observed Stmin was as low as 125us. This duration is very short. If back to back frame comes within such short duration, then there are chances that software may not get enough time to process it. This may occur rarely.

In such scenario FULL CAN(single buffer) Hw will discard new frame. Discarded frame will not be sent again by tester. When Tester sends new frame, it will not match Tp layers expectation which leads to Det Invalid Sequence Error. This is pictorially shown below:



**Solution:**

Suggest by vector, Hw FIFO change has solved this issue. Hw FIFO is an array of Hw buffer which can retain frames in case of software delays it in processing. In rare situation when Sw is delaying processing of frame, Hw can retain it in FIFO. When software will get chance it can process it. Thus FIFO prevents frame rejection which are being sent at high rate. This is pictorially shown below.



Above issue discussion with vector

With the available debug information (as we do not have info of local variables),  it seems like the  InitTxBasicCan might be overwriting the RAM area at two different points. Once for initialization on channel 0 at 0xF0200384 and once for channel 1 at 0xF0200004. This might be because the function is called without any Tx BasicCans present in the configuration, which is fine. However, "0" Tx BasicCan is considered a part of the Multiple Tx BasicCan feature, which indicates that we have a Tx BasicCan amount different from "1". This is enforced by the generator, however, according to the generated data, it seems like this has been overwritten by a user-define:

 \* [Warning] CAN02002 - An invalid value is configured

 \* - [Reduced Severity due to User-Defined Parameter] CanMultipleBasicCANTxObjects is not active but multiple TX BasicCANs used on some controller.

 \* Erroneous configuration elements:

 \* /ActiveEcuC/Can/CanGeneral[0:CanMultipleBasicCANTxObjects](value=false) (DefRef: /MICROSAR/Can\_Mpc5700Mcan/Can/CanGeneral/CanMultipleBasicCANTxObjects)

As the warning tries to tell, this might have unpredictable behavior, as would be the case in this situation:

 \* WARNING: This code has been generated with reduced-severity errors.

 \* The created output files contain errors that have been ignored. Usage of the created files can lead to unpredictable behavior of the embedded code.

 \* Usage of the created files happens at own risk!

So, please remove the user define and enable the Multiple Tx BasicCan feature. You might see an error from the CanIf when you do this. If that is the case, the issue has been reported in attached ESCAN00094355 report. Use the workaround from the attached Escan report and enable the feature in CanIf by user-define.

The issue occurred because when there are no-BasicCan, the source code expected and relies on the switch Multiple Tx BasicCan

If that switch is not active, then we "know" that are is exactly "1" BasicCan configured. However, the this switch was not active although there are no BasicCans. We then try to init this BasicCan (which does not exist). Since there are no Tx messages at all, the macro for Tx Access points to the start of the message RAM. Which was the receive filter for the particular FullCan

When we swapped the ID of the message, the list is re-sorted and there should be 'another message' that lost it's filter

I am closing this CASE as your query is resolved but you can always contact us on our support mail id if you have further questions or can call us on our below given support number.

All current service packs and drivers can be found in our Download-Center: <https://vector.com/vi_downloadcenter_de.html>

Please visit our KnowledgeBase: <https://vector.com/kbp/> for the most commonly asked questions.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*