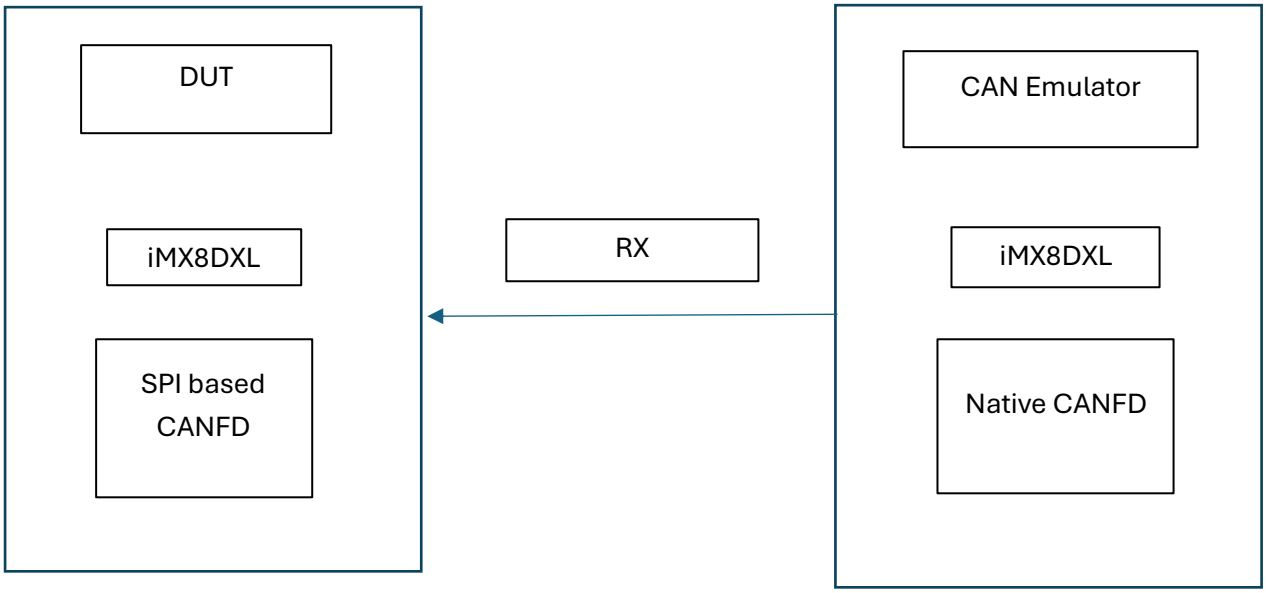
Issue Observed:

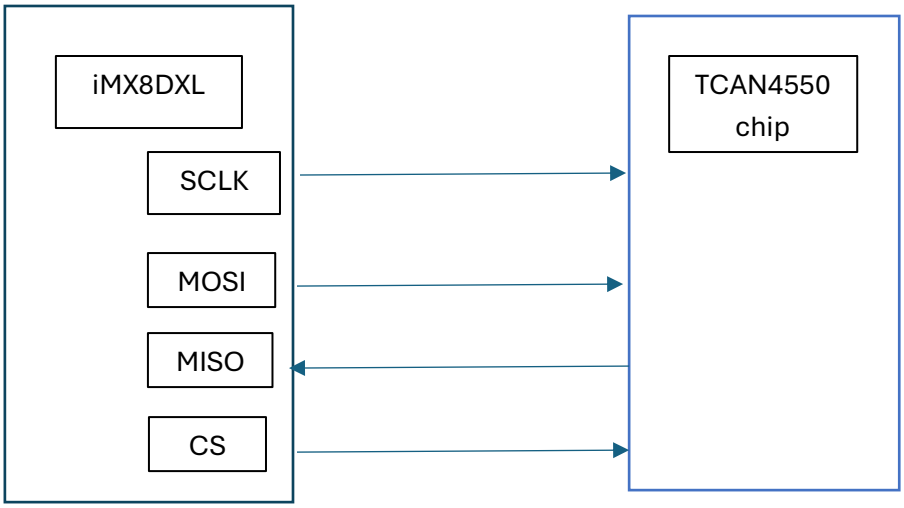
We are currently monitoring SPI-based communication for receiving (RX) CAN FD messages between two devices:

1. **CAN emulator:** having iMX8DXL processor with native CAN FD.
2. **DUT:** with iMX8DXL processor using SPI to receive messages.

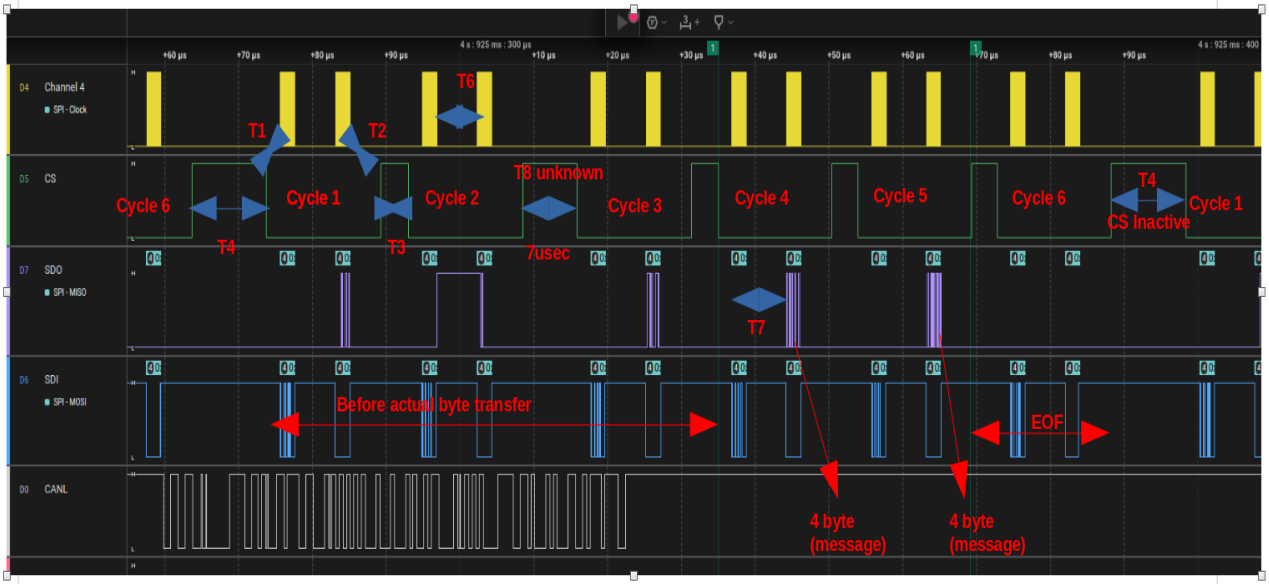
Test Setup:



The SPI frequency is set to **16 MHz**, and the **watermark level** for the m_can driver is configured to **26**, which triggers an interrupt after 26 messages are received. Upon interrupt, the SPI transfer for receiving messages starts.



While the SPI transfer is functioning, we have observed certain delays that are not expected. Below is a detailed breakdown of these delays and their theoretical vs. observed values.


D4 Channel 4
SPI - Clock
D5 CS
D7 SDO
SPI - MISO
D6 SDI
SPI - MOSI
D0 CANL
T1
T2
T6
T8 unknown
7usec
T3
T4
T7
Cycle 6
Cycle 1
Cycle 2
Cycle 3
Cycle 4
Cycle 5
Cycle 6
Cycle 1
CS Inactive
Before actual byte transfer
EOF
4 byte (message)
4 byte (message)

T	Description	Value – Theoretical	Value - Observed
T1	PCSSCLK: CS to CLK	120 nsec	2usec
T2	SCLKPCS: CLK to CS	60 nsec	10usec
T3	DBT: CS inactive after transfer of 4bytes	400 nsec	3usec
T4	Chip select inactive	400nsec (assumed)	20usec
T5	INT to CS	180nsec	30usec
T6	Time between the 2 clock patterns	400nsec (assumed)	5.5usec
T7	Time from first clock edge to start of 4 byte transfer T6 + 4byte transfer time (1.92usec)		7.42usec
T8	Chip select inactive after first 2 cycles of transfer **7usec**		7usec
Cycle 1 = Cycle 3 = Cycle 4 = Cycle 5 = Cycle 6 where Cycle 4 and 5 are actual data cycle	T1 + 4byte data transfer + T6 + 4byte data transfer + T2 + T3	120nsec + 1.92usec + 400nsec + 1.92usec + 60nsec + 400nsec = 4.82usec	(2 + 1.92 + 5.5 + 1.92 + 10 + 3 usec) 24.34usec (Values vary from 23 usec to 27usec)
Cycle 2	T1 + 4byte data transfer + T6 + 4byte data transfer + T2 + T8	120nsec + 1.92usec + 400nsec + 1.92usec + 60nsec + 400nsec = 4.82usec (Theoretically it should be same as of other cycles i.e.,400nsec(T3))	(2 + 1.92 + 5.5 + 1.92 + 10 + 7 usec) 28.34usec
1 message starting to another message start (1 Frame)	Cycle 1+Cycle 2+Cycle 3+Cycle 4+Cycle 5+Cycle 6 + T4	6 *4.82 + 400nsec= 29.32usec	(5 * 24.34 + 28.34 + 20) 170.04usec (Values vary from 165 usec to 175 usec)
Time for 26 messages	26 * 1 frame time + T5	26 * 29.32usec + 180nsec = 762.5usec = 0.7msec	26 * 170.04+ 30 = 4.5 (Values vary from 4msec to 5.5msec)

Number of msg/sec	26000/Time for 26 msg	26000 / 0.7= 37142	26000/4.5 = 5700

Summary:

We are observing throughput as around 15% of the theoretical value and during these transfers the CPU load will also be high and during Tx+Rx the cpu load goes above 100%.

NOTE:

- For 4byte transfer it take 32 clock cycles and takes 1.92usec for 16MHz spi clock frequency.
- Cycles 1,3,4,5,6 are same in term of timings and Cycle 2 has larger DBT time (T8).
- Cycle 4 and 5 are data cycles.

Questions:

We are currently in process to optimize these delays and increase our throughput. So, we wanted to know:

- What is your test setup while checking this TCAN chip?
- What about the delays involved? And can you provide the similar timing diagram
- What was the throughput you have seen for 8byte and 64byte message for 1Mb and 5Mb data bitrates?