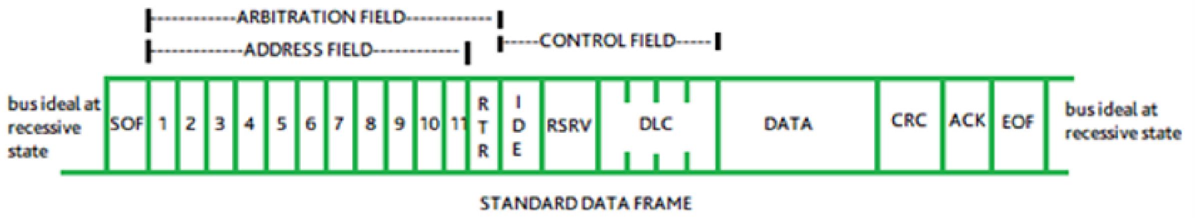
Can Errors: Basics

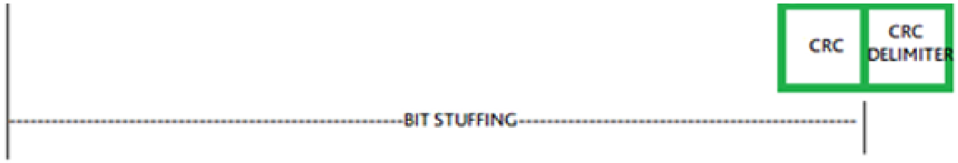
Now when we have studied CAN bus protocol and its functioning, it would be easy to understand how it faces errors and deals with them. Of course, no system is an ideal system. Errors are always bound to happen, but a sound system will always know how to detect the Error, omit it and resend the rectified data. CAN bus in similar manner experiences such errors but fights them effectively.

Before we begin let's understand the Data frame of the CAN bus:



A more detailed version of the Standard CAN data frame helps one understand how error bits are located and worked upon. Here the placement of Delimiter bits has been specified.

Questions?



Delimiter Bit: these are recessive bits that usually serve to provide time/space for completing a particular action. These bits ensure that there are bit transitions in the fields that do not have bit-stuffing applied. The bit transitions are necessary to recover timing synchronization that might not be otherwise available due to NRZ encoding. Other than providing the time for synchronization, the delimiter serves a specific purpose in Error Detection. The delimiter bits must come at a predefined place so that the form of the CAN frame is maintained.

CRC Delimiter: a CRC delimiter bit gives time or space to the ECU to calculate the CRC.

ACK Delimiter: once the data is received, the receiving end sends an acknowledgement to the transmitting node, and it requires some time, and hence ACK delimiter is used.

Bit Stuffing: after transmitting five consecutive bits of the same level by a node, it adds a sixth bit of the opposite level to the outgoing bitstream. The receiver removes this extra bit. This not only avoids excessive DC components on the Bus but also enables receivers to detect errors. Since this is a general rule, the receiver doesn't need extra information about the location of the stuffing bits to do the de-stuffing. Bit stuffing does not ensure that transmission errors do not corrupt the data; it ensures that the transmission starts and ends at the correct places.

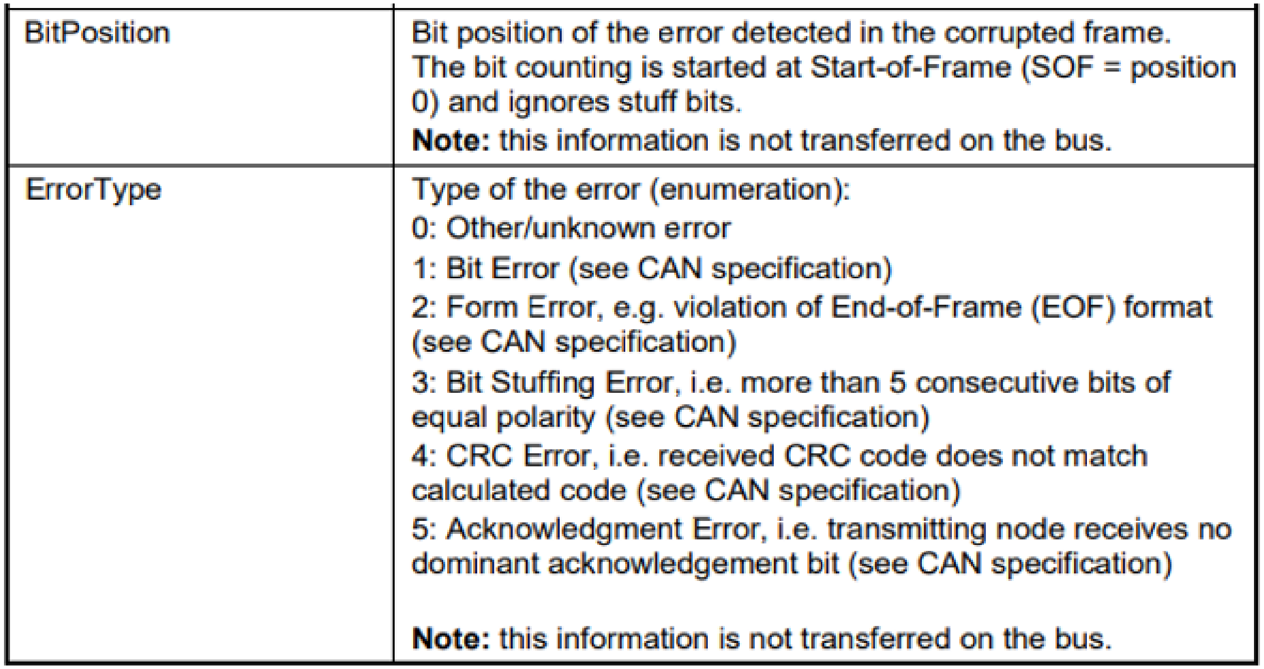
the Bus, a "Stuff Error" is signalled.

- Form Error: this refers to the fixed form of the field. Form check checks the CAN frame sent/received for a standard format. Violation of fixed bit format results in a Form Error. E.g. CRCD, ACKD, EOF have to be recessive bits, and the presence of any dominant bit will automatically be treated as a Form Error. Also, delimiter bits must come at a predefined place so that the form of the CAN frame is maintained; if the receiver fails to find delimiter bits at a proper place, this also generates a Form Error Frame. In this case, the fixed forms of the fields will have to be retransmitted.

- CRC Error: The transmitting node send a 15 bit CRC value onto the Bus. The receiving node then calculates a 15-bit number based on the payload it receives on its own. Then it compares it to the CRC delimiter it received as part of the message. If the received CRC does not match with the calculated code, the receiver knows that the 8 bytes of the payload were corrupted or modified during transmission, known as CRC Error.

- ACK Error: The ACK bit is a defined recessive "1" (transmitted by the transmitter node), and the reply from the receiver is a defined dominant "0". But, if that is not the case and the transmitter does not receive a dominant acknowledgement bit in reply, it is termed ACK Error.

Below is the ASAM standard table for CAN error:



BitPosition	Bit position of the error detected in the corrupted frame. The bit counting is started at Start-of-Frame (SOF = position 0) and ignores stuff bits. **Note:** this information is not transferred on the bus.
ErrorType	Type of the error (enumeration): 0: Other/unknown error 1: Bit Error (see CAN specification) 2: Form Error, e.g. violation of End-of-Frame (EOF) format (see CAN specification) 3: Bit Stuffing Error, i.e. more than 5 consecutive bits of equal polarity (see CAN specification) 4: CRC Error, i.e. received CRC code does not match calculated code (see CAN specification) 5: Acknowledgment Error, i.e. transmitting node receives no dominant acknowledgement bit (see CAN specification) **Note:** this information is not transferred on the bus.

Error Frame Format

When a node detects an error in a message, a special message known as an "error frame" is transmitted. This special message violates the formatting rules of a CAN message and causes all other nodes (connected on the network) to send an error frame as well. This intentional violation of the CAN standard (i.e. the sending of 6 dominant bits) guarantees the destruction of a faulty data or remote frame. Thus, enabling the original transmitter to retransmit the message automatically.

transmits it. It is transmitted by the node/s that detect/s any communication error, resulting in an immediate termination of transmission, and is retransmitted later. Again, this is all controlled by a controller, not application software.

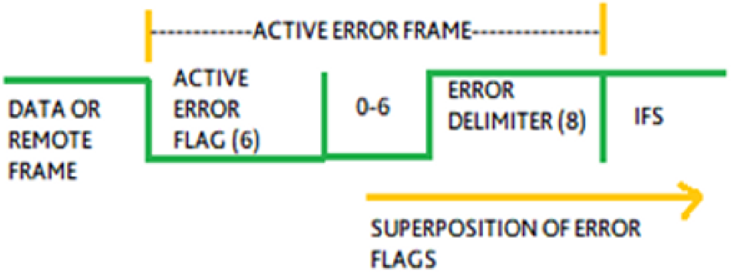
Error Flag: Every CAN controller along a bus tries to detect errors within a message as error handling is built into the CAN protocol. Every CAN controller along a bus will try to detect errors within a message. If an error is found, the discovering node transmits an Error Flag, thus destroying the bus traffic. Using the error counters, a CAN node can detect faults and perform error confinement.

Error States of CAN Bus

Depending on the specific error count, a CAN controller handles the switching of the error state.

Error Active

A node starts in Error Active mode. CAN controller assumes the normal state Error Active. In this state, the CAN controller sends six dominant bits after detecting an error. When the limit is below TEC[NM1] (Transmit Error Counter) <127 and REC (Receive Error Counter) <127, an Error Active node will transmit Active Error Flags when it detects errors.



Error Passive

The Error Passive state indicates a detected error by sending six homogeneous recessive bits only. This prevents the error-detecting receivers from globalizing detected errors. When any one of the two Error

Bus Off

When the Transmit Error Counter raises above 255(TEC>255 and REC>255), the node will enter the Bus Off state. It happens when the CAN controller fails or at times of extreme accumulations of errors. In this case, the CAN controller disconnects itself from the CAN bus. As a result, that particular CAN node will get switched off from the CAN network; Resulting in no communication at all.

"Influx data loggers' supports CAN error Logging in ASAM standard format." It makes sure that your data log is checked for these errors and quality data is delivered. Your data is essential. The CAN error logging functionality on Influx Devices is tested and benchmarked with other industry-leading CAN devices. For more information about our products, Click here.

Was this article helpful?

sales_india@influxtechnology.com

Do not use the Logger's USB port for data logging purpose, as it may result in damaging the USB port and this won't be covered under warranty.

Warranty and Limitation of Liability