Implementation Guide for the CAN Protocol

(Addendum to the protocol specification)

The Controller Area Network protocol specification document describes the function of the network on the whole. Additionally, Bosch provides a Reference CAN Model to the CAN licensees, supporting the protocol's implementation into the licensees' CAN controller nodes.

This Reference CAN Model is in some cases, where the reaction to certain conditions was left open, more restricted than the protocol specification. The specific reaction to those conditions defined by the Reference CAN Model can be regarded as a de facto standard, simplifying the implementation's verification. The verification is done by the comparison of the functions of an implementation to the functions of the Reference Model when applying a set of test conditions. All existing CAN implementations comply to this de facto standard, including 82526 and 82C200, which were designed before the existence of the Reference CAN Model.

In this paper, the label "Reference CAN Model" stands for both versions, the "C Reference CAN Model" and the "VHDL Reference CAN Model"; their functions are identical.

The additional restrictions of the Reference CAN Model apply in the cases of the reception of a Data Length Code > 8 (1), the reception of a dominant SRR bit in an Extended Frame (2), the reception of a dominant bit as last bit of End Of Frame (3), the increment of the Receive Error Count when it has reached the Error Passive level (4), and the condition for Hard Synchronization (5).

These cases are explained in the following, with references to the CAN Specification Revision 2.0, Part B:

(1) According to the CAN Specification, no transmitter may send a frame with DLC > 8. The case of DLC > 8 is not covered by any of the error types defined in chapter 7.1 "Error Detection". It is neither a Bit Error, nor a Stuff Error, nor a CRC Error, nor an Acknowledge Error. It could be regarded as a Form Error, but the DLC belongs to the stuffed Control Field and the Form Error is only defined for the fixed-form bit fields (see chapter 6 "Bit Stream Coding"). So no condition for Error Signalling (see chapter 7.2) is fulfilled, the reaction of a receiver to a DLC > 8 is not defined. The Reference CAN Model defines as de-facto standard the assumption [if received DLC > 8 then DLC := 8], expecting to receive 8 data bytes even when the received Data Length Code exceeds its upper limit of 8.

(2) The CAN specification requires the SRR bit to be sent as recessive . The receiver's reaction to a SRR bit sampled as dominant is not defined. It is

obviously neither a Bit Error, nor a Stuff Error, nor a CRC Error, nor an Acknowledge Error (see chapter 7.1 "Error Detection"). And, since the SRR bit is located in a stuffed bit field, a SRR bit received as dominant is not a Form Error. The Reference CAN Model defines as de-facto standard that the SRR bit is treated like the Reserved Bits, which have to be sent as dominant, but whose actual value is ignored by receivers. So no transmitter may send a dominant SRR bit in an Extended Frame while a receiver ignores the value of the SRR bit (but the value is not ignored for bit stuffing and arbitration).
Since the SRR bit is received before the IDE bit, a receiver cannot decide instantly whether it receives a RTR or a SRR bit. That means only the IDE bit decides whether the frame is a Standard Frame or an Extended Frame.

(3) According to chapter 5 "Message Validation", a message is valid for receivers, even when the last bit of End of Frame is received as dominant. Therefore, this dominant bit is not regarded as an error. On the other hand, the fixed-form bit field End of Frame contains an illegal bit and the receiver of the dominant bit may have lost synchronization, which requires a reaction. The Reference CAN Model follows the example of chapter 3.2.4 "Overload Flag", condition 3, where the reception of a dominant bit as the last bit or Error Delimiter of Overload Delimiter is responded with an Overload Frame.

(4) Theoretically, the Fault Confinement Rules could increment the Receive Error Count's value over all limits, when an Error Passive receiver detects additional errors without receiving any error free message. This cannot be implemented in hardware, the counter's value is limited by its actual number of digits. In the Reference CAN Model, the Receive Error Count has a resolution of 8 bits, which is sufficient for all purposes of fault confinement, because once the Receive Error Count has reached its Error Passive level (128), it is irrelevant how much this level is exceeded. So the Receive Error Count needs not to be incremented above the Error Passive level.
In the Reference CAN Model, the Receive Error Counter is used to count the 128 sequences needed for the Busoff Recovery Sequence (see Fault Confinement Rule 12). This technique is not intended as an example for hardware implementations of CAN protocol controllers, the CAN licensee is free to use other solutions best suited for the individual implementation.

(5) Synchronization Rule 4 requires the Hard Synchronization to be performed at every edge from recessive to dominant during Bus Idle. Additionally, chapter 3.2.1 "Data Frame – Start of Frame" requires the Hard Synchronization for each received Start of Frame. A Start of Frame can be received not only during Bus Idle, but also during Suspend Transmission and at the end of Intermission. Therefore, the Reference CAN Model enables the Hard Synchronization not only for Bus Idle state, but also for Suspend state and for the end of the Intermission

State. Any node disables Hard Synchronization when it samples an edge from recessive to dominant or when it starts to send the dominant Start of Frame bit. Since the synchronization on edges from dominant to recessive has become obsolete with the upgrade from CAN protocol version 1.1 to version 1.2 (see CAN Specification Revision 2.0, Part A, chapter 9.1 section [4]) the Reference CAN Model does not support this kind of synchronization.