

SURFACE VEHICLE RECOMMENDED PRACTICE

SAE J2366-2

ISSUED
NOV2001

Issued 2001-11

Link Layer

TABLE OF CONTENTS

1.	Scope	4
2.	References	5
2.1	Applicable Publications	5
2.1.1	SAE Publications	5
2.2	Related Publications	5
2.2.1	SAE Publications	5
2.2.2	CAN Publication	5
2.2.3	ISO Publication	5
3.	Definitions	5
3.1	ATID	5
3.2	Bit	5
3.3	Clear	5
3.4	Collision	5
3.5	Controlling Layer	5
3.6	Dominant	5
3.7	Frame	5
3.8	FrameCollisionCounter Threshold, FCCThreshold	6
3.9	Halt State	6
3.10	HDLC	6
3.11	IDB and IDB-C	6
3.12	Insane Node	6
3.13	ITS	6
3.14	Key State	6
3.14.1	Key Off State	6
3.14.2	Key On State	6
3.14.3	Key Pulse	6
3.15	LLC (Logical Link Control)	6
3.16	MAC (Media Access Control)	6
3.17	Node	6
3.17.1	Active Node	7
3.17.2	Not Active Node	7
3.17.3	Not Present Node	7

SAE Technical Standards Board Rules provide that: "This report is published by SAE to advance the state of technical and engineering sciences. The use of this report is entirely voluntary, and its applicability and suitability for any particular use, including any patent infringement arising therefrom, is the sole responsibility of the user."

SAE reviews each technical report at least every five years at which time it may be reaffirmed, revised, or cancelled. SAE invites your written comments and suggestions.

TO PLACE A DOCUMENT ORDER: (724) 776-4970 FAX: (724) 776-0790
SAE WEB ADDRESS <http://www.sae.org>

Copyright 2000 Society of Automotive Engineers, Inc.
All rights reserved.

Documents
Document Sharing Hub
Printed in U.S.A.

3.17.4	Quiescent Node	7
3.18	Octet	7
3.19	OSI	7
3.20	PDU (Protocol Data Unit)	7
3.21	Power Mode State	7
3.22	Quiescent Network	7
3.23	Random	7
3.24	Recessive	7
3.25	Set	8
3.26	Time Intervals	8
3.26.1	Activity Timer, $T_{StartActivity}$	8
3.26.2	Wait Timer, T_{Wait}	8
3.26.3	Actual/Measured Token Hold Time, $T_{TokenHoldActual}$	8
3.26.4	Inter-Frame Maximum Delay, $T_{InterFrame}$	8
3.26.5	Token Hold Period Start Interval, $T_{StartTHP}$	8
3.26.6	My Token Hold Period Start Interval, $T_{MyStartTHP}$	8
3.26.7	Network Inactivity Time $T_{NoActivity}$	8
3.26.8	Lost Token Recovery Time $T_{LostToken}$	8
3.26.9	Maximum Token Hold Time, $T_{TokenHoldMax}$	9
3.26.10	Maximum Token Rotation Time, T_{MaxRot}	9
3.26.11	Node Inactivity Maximum, $T_{Inactive}$	9
3.26.12	Node Removal Timer, $T_{NodeRemove}$	9
3.26.13	Power Mode Status Change Time Interval, $T_{KeyOnOff}$	10
3.26.14	Bus Reset Time, $T_{ResetPeriod}$	10
3.26.15	Token Hold Time, $T_{TokenHold}$	10
3.26.16	Transmission Duration Time, $T_{TxDuration}$	10
3.26.17	Maximum Transmission Duration Time, $T_{MaxTxDuration}$	10
3.27	Token	10
3.28	Token Holder	10
3.29	<i>Token/Done</i> Frame	10
3.30	<i>Token/Start</i> Frame	10
4.	Environment	10
4.1	IDB-C Node Hardware Requirements	10
4.2	IDB-C without Gateway to OEM Data Bus	11
4.3	IDB-C with Gateway to OEM Data Bus	11
5.	Protocol Overview	12
5.1	Physical Bus Detection	13
5.2	Logical Network Architecture	13
5.2.1	Virtual Token Bus	13
5.2.2	Dynamic Address Allocation	13
5.2.3	Power Mode Detection and Generation	13
5.2.4	Frame Flow Control	13
5.2.5	Node Removal Detection	14
5.2.6	Lost Token Detection	14
5.2.7	Runaway/Insane Node Detection and Recovery	14
5.2.8	Recovery from bus contention following extended period of no activity	14
5.2.9	Recovery from bus contention when the $T_{NoActivity}$ timer has not expired	14
5.2.10	Enabling CAN Frame Filtering	15
5.2.11	Transmission Order of CAN Frames	15
5.2.12	Link Level Diagnostics	15

6.	Link Layer Frame Format	16
6.1	CAN Frame Format, Objects, and Elements	17
6.1.1	Loss of Arbitration	17
6.1.2	Frame Acknowledgement	17
6.1.3	Remove Frames	17
6.1.4	Extended Frame (29 Bit Identifier) Element definition	17
6.2	Arbitration and CAN Error Handling	18
6.2.1	Transmitting Node Loses Arbitration	18
6.2.2	CAN BIT ERROR.....	18
6.2.3	CAN STUFF ERROR.....	18
6.2.4	CAN CRC ERROR	18
6.2.5	CAN FORM ERROR	18
6.2.6	ACKNOWLEDGMENT ERROR	18
6.3	Media Access Control (MAC) Sub-layer	18
6.3.1	MAC Field Definitions	19
6.3.2	MAC Frame Definitions.....	19
6.4	Logical Link Control (LLC) Sub-layer	27
6.4.1	Frame Field Definitions	27
6.4.2	LLC DATA Frame containing SAE J2366-4 Data	28
6.4.3	LLC DATA Frame containing Other Data	28
6.5	SAE J2366-2 Protocol Sequences	29
6.5.1	Initialization	29
6.5.2	Active State	31
6.5.3	Get Token State	32
6.5.4	Have Token State	34
6.5.5	SendFrame Function	35
6.5.6	ReceiveFrame Function.....	36
6.5.7	InboundFrame Function.....	36
	Appendix A Halt State Processing	38
Figure 1	Scope of SAE J2366-2	4
Figure 2	Example IDB-C without OEM Data Bus	11
Figure 3	Example IDB-C with OEM Data Bus	12
Figure 4	Mapping of SAE J2366-2 on CAN 2.0B	16
Figure 5	Link Layer Frame Format	17
Figure 6	MAC Sub-layer Frame Format	18
Figure 7	MAC BUSRESET Control Frame Format	20
Figure 8	MAC DEVRESET Control Frame Format	21
Figure 9	MAC JOIN Control Frame Format	22
Figure 10	MAC JNAK Control Frame Format	23
Figure 11	MAC LEAVE Control Frame Format	24
Figure 12	MAC STATUS Control Frame Format	25
Figure 13	MAC STATUSREQ Control Frame Format.....	26
Figure 14	LLC General Frame Format.....	27
Figure 15	LLC Other (non-SAE J2366-4) Data Frame Format	28
Figure 16	Initialization Sequence	30
Figure 17	Active State Sequence	32
Figure 18	Get Token State Sequence	33
Figure 19	Have Token State Sequence	34
Figure 20	SendFrame Function Sequence	35
Figure 21	ReceiveFrame Function Sequence	36
Figure 22	InboundFrame Function Sequence	37

Table 1	Physical Node ID Transmission Values.....	7
Table 2	Token Hold Time Maximums	9
Table 3	CAN Elements	17
Table 4	MAC Frame IDs Defined	19

1. **Scope**—This SAE Recommended Practice details the Link Layer of the Intelligent Transportation Systems (ITS) Data Bus, which is generally intended for in-vehicle use.

The ITS Data Bus (IDB) is a non-proprietary token passing bus, designed to allow disparate consumer, vehicle, and commercial electronic components to communicate and share information across a standard, open data bus. This first version of the IDB is called IDB-C.

This Recommended Practice describes the Link Layer of the IDB-C, as shown in Section 1 Below the Link Layer of the IDB-C is the CAN 2.0B Link Layer. The Physical Layer of the IDB-C incorporates the Physical Layer as specified by SAE J2366-1 and J2590.

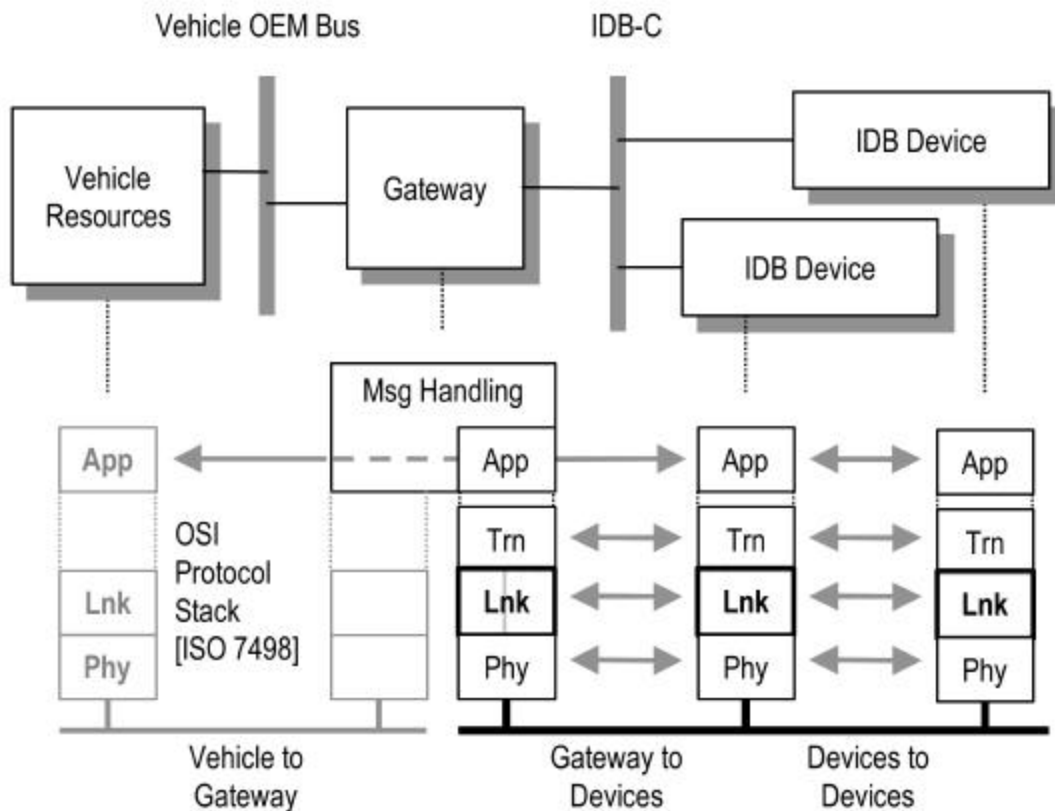


FIGURE 1—SCOPE OF SAE J2366-2

The IDB-C Link Layer is logically divided into two functional sub-layers, viz., Logical Link Control (LLC) and Media Access Control (MAC). However, these are logical groupings that re-use some of the same bits within the CAN 2.0B frame for efficiency.

The MAC sub-layer handles all media access functions, such as virtual token handling, addressing, and notifications of IDB-C configuration changes to other MAC layers on the IDB-C.

The LLC sub-layer provides the mechanisms to send and receive point-to-point or broadcast frames to other LLC sub-layers on the IDB-C. In addition, the LLC provides an Alternate Transport Identifier (ATID) allowing more than one set of higher layer protocols to be used on a single Node.

2. References

2.1 Applicable Publications—The following publications form a part of this specification to the extent specified herein. Unless otherwise indicated, the latest issue of SAE publications shall apply.

2.1.1 SAE PUBLICATIONS—Available from SAE, 400 Commonwealth Drive, Warrendale, PA 15096-0001, <http://www.sae.org>.

SAE J2366-1—ITS Data Bus—Physical Layer
SAE J2590—PMODE for In-Vehicle Networks

2.2 Related Publications—The following publications are provided for information purposes only and are not a required part of this document.

2.2.1 SAE PUBLICATIONS—Available from SAE, 400 Commonwealth Drive, Warrendale, PA 15096-0001, <http://www.sae.org>.

SAE J2355—ITS Data Bus—Architecture Reference Model Information Report
SAE J2366-4—ITS Data Bus—Thin Transport Layer
SAE J2366-7—ITS Data Bus—Application Message Layer

2.2.2 CAN SPECIFICATION—Available from Robert Bosch GmbH, Postfach 50, D-7000 Stuttgart 1, Germany.

CAN Specification, Version 2.0B

2.2.3 ISO PUBLICATION—Available from ANSI, 25 West 43rd Street, New York, NY 10036-8002.

ISO 7498:1990—Information processing systems—Open system interconnect—Basic reference model

3. Definitions

3.1 ATID—Alternate Transport Identifier. This identifies which protocol is to be used to decode the protocol data packet portion of the frame.

3.2 Bit—A signaling unit or unit of data transmission of minimal length. It can hold either the value '0'B or '1'B.

3.3 Clear—The word “clear” is used in referring to one or more bits to indicate that they each have a value of '0'B. This is also referred to as “dominant” in CAN 2.0B terminology.

3.4 Collision—A collision occurs when two or more Nodes attempt to transmit data on the network simultaneously. This results in the transmissions interfering with each other. On a CAN bus, one of the colliding Nodes “wins”, and its data appears on the bus. Other Nodes back off and retry their transmission.

3.5 Controlling Layer—The Controlling Layer is that hardware, software application, or protocol layer responsible for determining if and when the Node should join or leave the network.

3.6 Dominant—A term defined in CAN 2.0B to indicate a data bit with a value of '0'B.

3.7 Frame—An ordered sequence of octets, used to communicate information between the Link Layers of IDB-C Nodes.

3.8 FrameCollisionCounter Threshold, FCCThreshold—Maximum number of non-JOIN, JNAK, STATUSREQ frames the Token Holder shall accept within its own token hold period from any given Node. If the Token Holder receives more such frames from a given Node, it shall queue a DEVRESET frame to that Node.

FCCThreshold is set to 1. The FrameCollisionCounter is incremented whether $T_{NoActivity}$ expired or not. If $T_{NoActivity}$ expired, reception of one non-JOIN, JNAK or STATUSREQ frame is normal, because of network contention. But if further frames are received from any Node, this means that the sending Node did not recognize the proper Token Holder. It is therefore requested to reset immediately to avoid further transmission and get a new chance to properly identify the Token Holder.

3.9 Halt State—The Halt State is a state in which an initialized Node remains, when the Controlling Layer (or other mechanism) has specified that the Node should not be participating on the IDB-C. In this state, the Node is a Not Present Node from the point of view of other Nodes.

3.10 HDLC—High Level Data Link Control – one of the protocols from which some features of the IDB-C Link Layer protocol were adapted.

3.11 IDB and IDB-C—The ITS Data Bus. This acronym describes a family of networks operating at different data rates and over different physical layers. This document describes the Link Layer protocol for the IDB-C, a low speed network based on CAN 2.0B.

3.12 Insane Node—This term is used to define a condition under which a Node is interfering with communication on the network, and is not adhering to protocol specifications.

3.13 ITS—Intelligent Transportation Systems

3.14 Key State—The Key State (Off, On, or Pulse) is implemented by the SAE J2590 PMODE specification. The SAE J2366-2 Link Layer may need an out of band mechanism for querying or setting the Key State.

3.14.1 KEY OFF STATE—The term “Key Off State” as used in this document has the same meaning as the PMODE OFF state as defined in SAE J2590. In this state, the network operates in a manner that requires devices to conserve power, at the expense of some protocol efficiency.

3.14.2 KEY ON STATE—The term “Key On State” as used in this document has the same meaning as the PMODE ON state as defined in SAE J2590. The state is set by the Vehicle Services Interface, typically based upon the state of the vehicle key switch.

The Key On State is the default state of the network in the absence of a Vehicle Services Interface providing power mode information. This simplifies cabling and operation outside a vehicle.

3.14.3 KEY PULSE—The term “Key Pulse” is used in this document to describe the mechanism to wake up a network, as defined in SAE J2590. This signal activates the IDB-C while it is in the Key Off State. Only devices configured to operate in the Key Off State respond to a Key Pulse.

3.15 LLC (Logical Link Control)—This is a sub-layer of the Data Link Layer and forms the interface between the MAC and next higher layer in the protocol stack.

3.16 MAC (Media Access Control)—This sub-layer of the Data Link Layer (DLL) is responsible for control within the DLL.

3.17 Node—A Node is a device on the network. Each Node shall have at least one physical Node ID. A Link Layer address of '0'H is the logically lowest address and 'F'H is the highest. The physical Node ID values are inverted when sent on the wire, both in the *FromAddr* and *DestAddr* header elements, as shown in Table 1, r = Recessive, d = Dominant.

TABLE 1—PHYSICAL NODE ID TRANSMISSION VALUES

Node ID	Transmit	Node ID	Transmit	Node ID	Transmit	Node ID	Transmit
0	rrrr	4	rdr	8	drr	C	ddr
1	rrrd	5	rdrd	9	drrd	D	ddrd
2	rrdr	6	rddr	A	drdr	E	dddr
3	rrdd	7	rddd	B	drdd	F	dddd

- 3.17.1 **ACTIVE NODE**—A Node is Active or in the Active state when it has a Node ID and is re-asserting its presence periodically, as defined in this document. An ACTIVE Node is fully participating in IDB-C activity.
- 3.17.2 **NOT ACTIVE NODE**—A Node is Not Active or is in the Not Active state when it has been assigned a Node ID, but fails (in the Key On State) to transmit any message within $T_{Inactive}$. After two consecutive such $T_{Inactive}$ periods have elapsed, the Node is considered a Not Present Node. In the Key Off State, a Node does not become a Not Present Node for this reason.
- 3.17.3 **NOT PRESENT NODE**—A Node is said to be Not Present when it is not connected to the network, otherwise has no Node ID, or has not transmitted any messages on the network within two $T_{Inactive}$ periods. A Node that is Not Present may be in this state because it has chosen to leave the IDB-C, because it has just detected power applied to the network interface, or for other reasons. A Node may choose to leave the network at any time.
- 3.17.4 **QUIESCENT NODE**—A Quiescent Node is a Node that is designed to operate in the Key Off State. In a network all Nodes transition to Quiescent state or Not Active state simultaneously, based upon the transition to the Key Off State. A Quiescent Node retains its Node ID and node map even after it stops actively transmitting on the network. A Quiescent Node becomes Active, retaining its Node ID, upon receipt of a Key Pulse indication.
- 3.18 **Octet**—An octet is a unit of data transmission that consists of eight bits. On most commonly used microprocessors, such an eight-bit quantity is commonly referred to as a “byte”. The bits within an octet may be subdivided and/or grouped, to provide specific meanings.
- 3.19 **OSI**—Officially known as the “ISO Reference Model of Open System Interconnection”. It refers to a model, or way of thinking about communication protocols as being divided into (usually) seven distinct layers.
- 3.20 **PDU (Protocol Data Unit)**—Message packets exchanged between two protocol entities.
- 3.21 **Power Mode State**—J2366-1 and J2590 define a signal, PMODE, which indicates the power mode status of the IDB-C. This line indicates a PMODE bus state of ON (also called “Key On” in this document) or it indicates a PMODE bus state of OFF (also called “Key Off” in this document). When pulsed to the Key On state, this line indicates to Nodes that a device wishes to communicate with other Nodes on the network. There is no meaning associated with a pulse to the Key Off state.
- 3.22 **Quiescent Network**—The network is said to be quiescent when in the Key Off State and no Nodes are Active on the IDB-C.
- 3.23 **Random**—Good random values are those that decrease the probability of multiple Nodes selecting the same Node ID or selecting the same value for a variable time interval. An implementation's random number generation shall consider that there may be more than one device of the same type, manufacturer, model, and manufacturing lot on any given network.
- 3.24 **Recessive**—A term defined in CAN 2.0B to indicate a data bit with a value of '1'B.

3.25 Set—The word “set” is used in referring to one or more bits to indicate that they each have a value of '1'B. This is also referred to as “recessive” in CAN 2.0B terminology.

3.26 Time Intervals—Several critical time intervals are specified for the Link Layer Protocol. Unless otherwise noted, time intervals are significant to the nearest 0.1 mS. Each is defined below:

- 3.26.1 **ACTIVITY TIMER, $T_{\text{STARTACTIVITY}}$** —This is the amount of time a device spends at power up looking for responses to its initial High Priority STATUSREQ frame, and other valid frames on the network. The duration of this timer is $2 \times T_{\text{MaxRot}} + 10 \text{ mS}$.
- 3.26.2 **WAIT TIMER, T_{WAIT}** —This is the amount of time a device spends after sending a JOIN frame, looking for a JNAK response. The duration of this timer is $2 \times T_{\text{MaxRot}} + 10 \text{ mS}$.
- 3.26.3 **ACTUAL/MEASURED TOKEN HOLD TIME, $T_{\text{TOKENHOLDACTUAL}}$** —*Token Hold Time*, $T_{\text{TokenHoldActual}}$ is the measured interval that starts with the transmission of the first frame by a Node on the network, as indicated by the *Token and Start/Done* bits, and ends when the Node has completely clocked out a frame with the *Token and Start/Done* bits dominant.
- 3.26.4 **INTER-FRAME MAXIMUM DELAY, $T_{\text{INTERFRAME}}$** —This interval is the maximum amount of time allowed between the ends of transmitted frames within a token hold period. This time is defined as 1.0 mS.
- 3.26.5 **TOKEN HOLD PERIOD START INTERVAL, T_{STARTTHP}** —This is the maximum amount of time that a Node may take to completely transmit its first frame (all bits clocked out). This time is defined as 1.0 mS.
- 3.26.6 **MY TOKEN HOLD PERIOD START INTERVAL, $T_{\text{MYSTARTTHP}}$** —This is the amount of time that must elapse after the receipt of a frame with dominant *Token and Start/Done* bits before a given Node's Token Hold Period begins. The time is defined as $T_{\text{StartTHP}} \times N$, where N is the number of Active Nodes between the one that just completed its Token Hold Period and this Node. If there is bus activity during this period, this Node shall cancel this timer and revert to looking for a frame with dominant *Token and Start/Done* bits.
- 3.26.7 **NETWORK INACTIVITY TIME $T_{\text{NOACTIVITY}}$** —This is the amount of time of no network activity (valid or invalid frames or CAN errors related to bus activity) following transmission of a frame with the *Start/Done* bit dominant, after which any Node may begin transmitting. It is defined as $T_{\text{StartTHP}} \times N$, where N is the number of Active Nodes.
- 3.26.8 **LOST TOKEN RECOVERY TIME $T_{\text{LOSTTOKEN}}$** —This is an amount of time after a Node receives a frame with dominant *Token* and recessive *Start/Done* bits (indicating the start of that Node's token hold period) without receiving another frame with a dominant *Token* bit. After this time period, a Node may presume that there is no bus activity and attempt to acquire the *Token*. This interval is defined as $T_{\text{Inactive}} + T_{\text{MaxRot}}$.

NOTE— If a Node assumes there is a Token Holder, it will ignore *Token/Start* or *Token/Done* frames from any Node other than the Token Holder. If the *Token/Done* frame is not received (for whatever reason), then the receiving Node will not change the Token Holder identity. If the Node which released the Token stops sending for T_{Inactive} , the receiving Node will assume that there is no bus activity after $T_{\text{LostToken}}$ has elapsed, and will attempt to acquire the Token, even if there is a legitimate Token Holder and activity on the bus. This is avoided if $T_{\text{LostToken}} > T_{\text{Inactive}}$, because then the Node which released the Token will, at least, sent a STATUS frame, with the *Token* and *Start/Done* bits dominant, which is the condition the other Node was waiting for. From then on, the receiving Node sees the Token circulating again.

3.26.9 MAXIMUM TOKEN HOLD TIME, $T_{\text{TokenHoldMax}}$ —This interval specifies the maximum $T_{\text{TokenHoldActual}}$ for any Node. This interval is based on the number of Active Nodes at the time when this Node calculates its $T_{\text{TokenHoldMax}}$, and does not change during this Node's token hold period. The interval is expressed in the formula listed as follows. Table 2 lists the valid values for this time based on the number of Active Nodes.

If a Node properly regains the Token less than T_{MaxRot} since the start of its previous Token Hold Period, it may expand its $T_{\text{TokenHoldMax}}$ for the current Token Hold Period to all remaining unused T_{MaxRot} time in the current Token Rotation. A Node shall expand its $T_{\text{TokenHoldMax}}$ in no more than five consecutive Token Rotations.

$$T_{\text{TokenHoldMax}} = (T_{\text{MaxRot}} - [N_{\text{ActiveNodes}} \times T_{\text{StartTHP}}]) / N_{\text{ActiveNodes}} \quad (\text{Eq. 1})$$

where:

$N_{\text{ActiveNodes}}$ = Number of Active Nodes on the network

T_{MaxRot} = Maximum virtual token rotation (50.0 mS)

T_{StartTHP} = 1.0 mS period per Active Node

Solving for the worst case (16 Active Nodes):

$$\begin{aligned} T_{\text{TokenHoldMax}} &= (50.0 \text{ mS} - [16 \times 1.0 \text{ mS}]) / 16 \\ &= (50.0 \text{ mS} - 16.0 \text{ mS}) / 16 \\ &= 2.125 \text{ mS per Node} \end{aligned} \quad (\text{Eq. 2})$$

TABLE 2—TOKEN HOLD TIME MAXIMUMS

Num Nodes	$T_{\text{TokenHoldMax}}$ (mS)	Num Nodes	$T_{\text{TokenHoldMax}}$ (mS)
1	N/A	9	4.5
2	24.0	10	4.0
3	15.6	11	3.5
4	11.5	12	3.1
5	9.0	13	2.8
6	7.3	14	2.5
7	6.1	15	2.3
8	5.2	16	2.1

3.26.10 MAXIMUM TOKEN ROTATION TIME, T_{MaxRot} —The maximum time for any single token rotation shall be equal to 50.0 mS.

3.26.11 NODE INACTIVITY MAXIMUM, T_{Inactive} —This is the amount of time within which a Node shall transmit at least one frame with the *Token* bit dominant while operating in the Key On State. A device with no data to send shall use a STATUS frame for this purpose. This time period is defined as 500.0 mS.

3.26.12 NODE REMOVAL TIMER, $T_{\text{NodeRemove}}$ —This is the amount of time after which a Node shall be removed from an Active Node's internal node status if it fails to transmit at least one frame with the *Token* bit dominant while operating in the Key On State. This time period is defined as $(2 \times T_{\text{Inactive}}) + 100.0 \text{ mS}$.

NOTE— For simplicity, the state diagrams in this document imply that there is a single timer, used for all Pending Nodes. While a single physical timer may be used, individual timeout values must be maintained for each Active Node.

- 3.26.13 POWER MODE STATUS CHANGE TIME INTERVAL, T_{KEYONOFF} — T_{KEYONOFF} defines the length of time the PMODE must remain in the same state to indicate a change between Key On and Key Off power states. This time period is 100.0 mS.
- 3.26.14 BUS RESET TIMER, $T_{\text{RESETPERIOD}}$ —This is used to ensure that Nodes have correctly recognized the BUSRESET frame. If not, the device that sent the BUSRESET frame shall reinitialize its CAN controller. This time period is defined as 500.0 mS.
- 3.26.15 TOKEN HOLD TIME, $T_{\text{TOKENHOLD}}$ —This is the amount of time a Node has already spent in its current Token Hold Period.
- 3.26.16 TRANSMISSION DURATION TIME, $T_{\text{TXDURATION}}$ —This is the amount of time a Node spends trying to send a frame. If this time exceeds $T_{\text{TokenHoldMax}}$, transmission is aborted.
- 3.26.17 MAXIMUM TRANSMISSION DURATION TIME, $T_{\text{MAXTXDURATION}}$ —This is the maximum amount of time a Node is allowed for transmission of a frame.
 In the Get Token State, $T_{\text{MaxTxDuration}} = T_{\text{StartTHP}}$.
 In the Have Token State, $T_{\text{MaxTxDuration}} = T_{\text{TokenHoldMax}} - T_{\text{TokenHold}}$.
- 3.27 Token**—The Token in IDB-C is a “virtual token”. A Node holding the Token has the right to transmit information on the network. The Token is based upon a relative time slot after the last transmission. The Token is rotated based on the numerical ascendancy of the (Active) physical node identifiers, with a wrap around of numerically highest to lowest (Active) identifier.
- 3.28 Token Holder**—The Token Holder is the Node currently holding the Token. It is of major importance that all Nodes on the network see the same Node as the Token Holder at any given time. When a Node releases the Token, there is no Token Holder until another Node acquires the Token.
- 3.29 Token/Done Frame**—This is used to describe a frame that has its *Token and Start/Done* bits ‘dominant.’
- 3.30 Token/Start Frame**—This is used to describe a frame that has its *Token* bit dominant and its *Start/Done* bit ‘recessive.’
- 4. Environment**—The primary target environment for the IDB-C is the consumer vehicle market—namely, automobiles. However, use in other vehicles is also expected, including trucks, public transit, and emergency vehicles. The applications running on the IDB-C will be operating under less than ideal conditions. Both the physical devices themselves and their applications shall be capable of withstanding a rather harsh environment. This environment is expected to include unexpected power failures, voltage spikes, and unpredictable shutdowns.
- Each device shall power up and shut down gracefully and shall not violate the timing and utilization restraints specified by the SAE J2366 series of standards and SAE J2590. When possible, Nodes shutting down should remove themselves from the network by sending a LEAVE frame.
- SAE J2366-1 and J2590 define a hardware mechanism by which Nodes are informed of the “vehicle power mode.” Nodes use this information to determine when to shut down and when to restart. Nodes also use this mechanism (via a Key Pulse) to wake up the network.
- 4.1 IDB-C Node Hardware Requirements**—A Node shall conform to the SAE J2366-1 and J2590 specifications. A Node capable of being a Quiescent Node (see 3.17.4) shall be capable of generating a Key Pulse (see 3.14 and 3.14.2).

4.2 IDB-C without Gateway to OEM Data Bus—The basic configuration of the IDB-C (example in Figure 2) is, at minimum, a single bus architecture with only power being supplied by the vehicle. In this configuration, there is no communications or application support between Nodes on the IDB-C and the vehicle's private OEM data bus (if any). Nodes that depend on the vehicle information will not be able to perform their functions in this configuration.

At least a minimal gateway, providing “vehicle power mode” information shall be required to assert the Key On/ Off State. Without the power mode information, the IDB-C operates in normal power mode. This uses more power than operation in the Key Off State. The minimal gateway is not required to be a Node. When a gateway exists on the IDB-C, it may provide this “vehicle power mode” information to the other IDB-C Nodes.

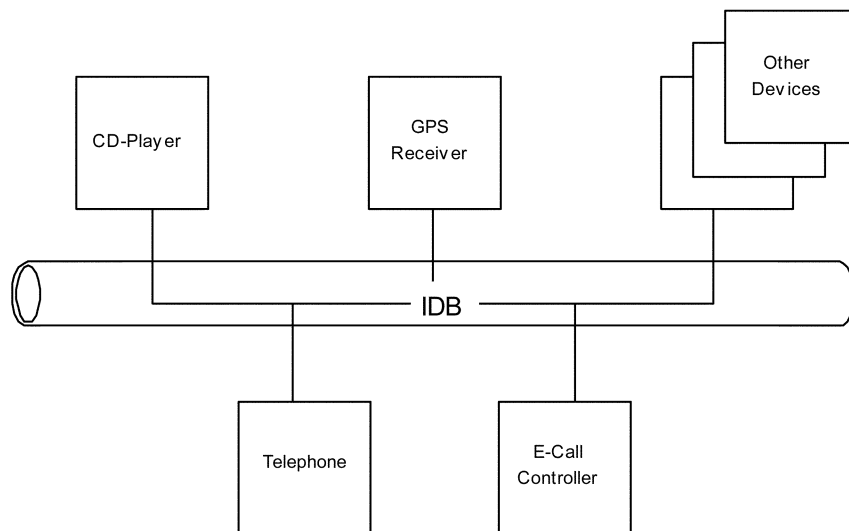


FIGURE 2—EXAMPLE IDB-C WITHOUT OEM DATA BUS

4.3 IDB-C with Gateway to OEM Data Bus—With the addition of a gateway the IDB-C can communicate with various devices on the existing OEM Bus through the Gateway. Figure 3 illustrates this configuration. By adding additional gateways to the IDB-C, it should be possible to communicate to a wide variety of systems with multiple protocols.

NOTE— Although the vehicle is shown as having an “OEM Bus” in this diagram, the IDB-C does not require that the vehicle or Gateway be implemented in that manner.

At least a minimal gateway, providing “vehicle power mode” information shall be required to assert the Key On/ Off State. Without the power mode information, the IDB-C operates in normal power mode. This uses more power than operation in the Key Off State. The minimal gateway is not required to be an IDB-C Node. When an IDB-C Gateway exists on the network, it shall provide this “vehicle power mode” information to the other Nodes.

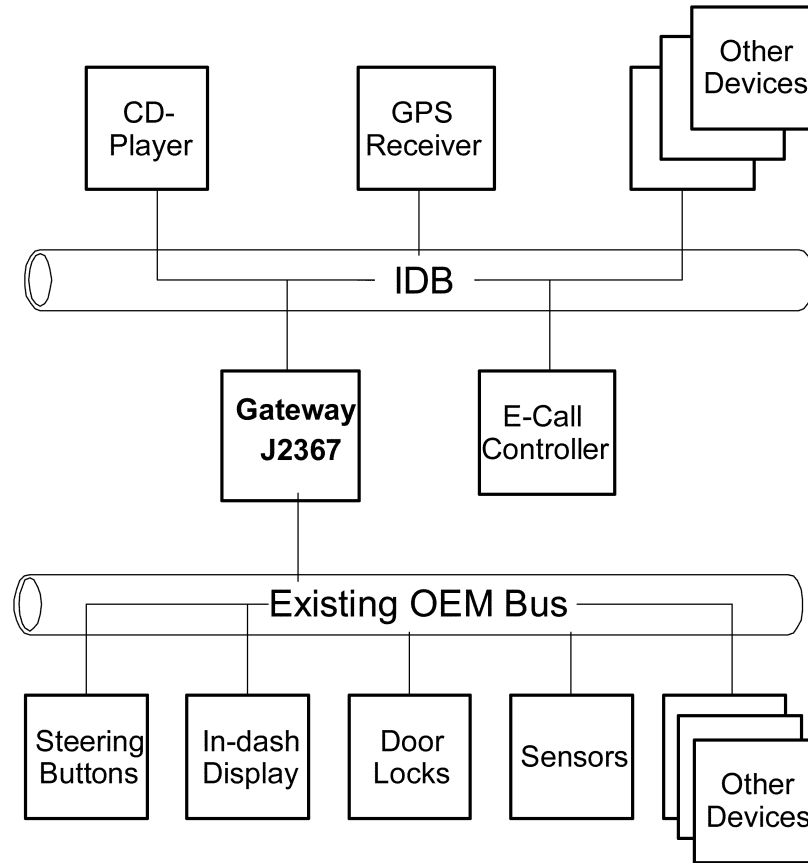


FIGURE 3—EXAMPLE IDB-C WITH OEM DATA BUS

5. **Protocol Overview**—This protocol is designed to meet the varied and challenging requirements for in-vehicle data communications. Some noteworthy features of the protocol:
- It provides deterministic frame latency. For the IDB-C this means that a Node is guaranteed to receive the Token within T_{MaxRot} .
 - The protocol provides for Node self-discovery and addressing.
 - Devices with lower demands can register on the bus but do not need to continually participate in bus activity, except for transmitting the relatively infrequent STATUS frame. This provides for greater data efficiency.
 - The protocol follows the ISO OSI protocol layering concepts.
 - The IDB-C protocol provides MAC/LLC and HDLC-like features.
 - The IDB-C provides dynamic frame flow control to allow graceful recovery from dropped frames due to slow processing and/or scarce memory resources.

All field values or bits that are not explicitly defined by this document, or explicitly referenced to other documents are reserved for future definition by a later version of SAE J2366-2. They are **not** available for arbitrary use by specific implementations.