
CAN Power Management Layer Specification

Network Standby-/Power Reduction Capabilities

CiA Draft Standard 150

Version 1.1

May 1997

1. Scope

This document specifies facilities and services of an Power Management Layer protocol entity on the CAN bus allowing significant reduction of power consumption in CAN networks by the introduction of the so-called network standby capability. The power reduction facilities of current CAN controller hardware designs by using the sleep mode, which can be recognized as a de-facto standard; are supported for usage under all higher layer protocols during an internal Power Management Layer IDLE state.

Although all hardware CAN controller devices on the market offer power reduction mechanisms implemented in a similar way (de-facto standard), this feature is not covered by /1/. The corresponding services will be introduced in chapter 3.1 and summarized in chapter 4. for further reference.

The purpose of an ISO/OSI Power Management Layer specification is to build up reliable communication, what is trivial in CAN networks where CAN controllers may not enter the sleep mode. If one or more CAN controllers in a network are switched to the sleep mode, this might result in message loss. The described protocol solves this problem by managing the network-wide consistency of the CAN-controllers operation modes in an intelligent manner. The multi-master architecture of the network is thereby fully maintained.

WARNING: This CAN Power Management Layer deals with reliable communication on CAN networks with power reduction capability. There is very little relationship to the ISO/OSI Power Management Layer otherwise.

The main features of the CAN Power Management Layer are

- standby capability supports CAN sleep mode for usage under all CAN higher-layer protocols, i.e. functional LLC interface remains virtually the same although time behaviour may change
- full compatibility with CAN multi-master decentralized network architecture
- easy hardware implementation/integration into CAN peripheral controllers
- coexistence of nodes with and without Power Management Layer support is possible (performance may suffer but can be optimized, see interoperability rules in chapter 5.5)

2. References

- /1/ ISO 11898, "Road vehicles - Interchange of digital Information - Controller Area Network (CAN) for high-speed communications". November 1993.
- /2/ ISO 7498, "OSI Basic Reference Model".
- /3/ CiA DS-201 "CAN in OSI Reference Model". Version 1.1. February 1996.
- /4/ CiA DS-205-1A "Addendum to CiA/DS-205-1 CAL LMT Service Specification". May 1997.
- /5/ CiA DS-205-2A "Addendum to CiA/DS-205-2 CAL LMT Protocol Specification". May 1997

3. General Description

3.1 CAN ISO/OSI Layer-2 Hardware Power Reduction Capabilities

Within a CAN network, consisting of several nodes, power reduction can be achieved during idle-times without bus traffic by switching the nodes' local CAN controller device into a special mode, the sleep mode. In combination with the sleep mode, additional local power reduction mechanisms may be used, which allow a considerable saving of power. Lowest power consumption can be achieved in CMOS-designs, if the nodes processor is stopped.

The CAN standard /1/ specifies that the CAN data link layer's (layer-2) subentity LLC (Logical Link Control) owns a Node_Status and informs higher layer entities via this status, whether the Fault Confinement Logic has entered the bus-off state. For all existing CAN controllers, this Node_Status is extended by the above-mentioned sleep mode for power reduction purposes. Therefore, all existing CAN peripheral controllers feature an additional LLC-, „Power Reduction Capability“, which is outside the scope of /1/.

CAN controllers with LLC power reduction capability behave as follows (see extended LLC power reduction capability state diagram in **Fig.3.1**):

CAN Power Management Layer Specification

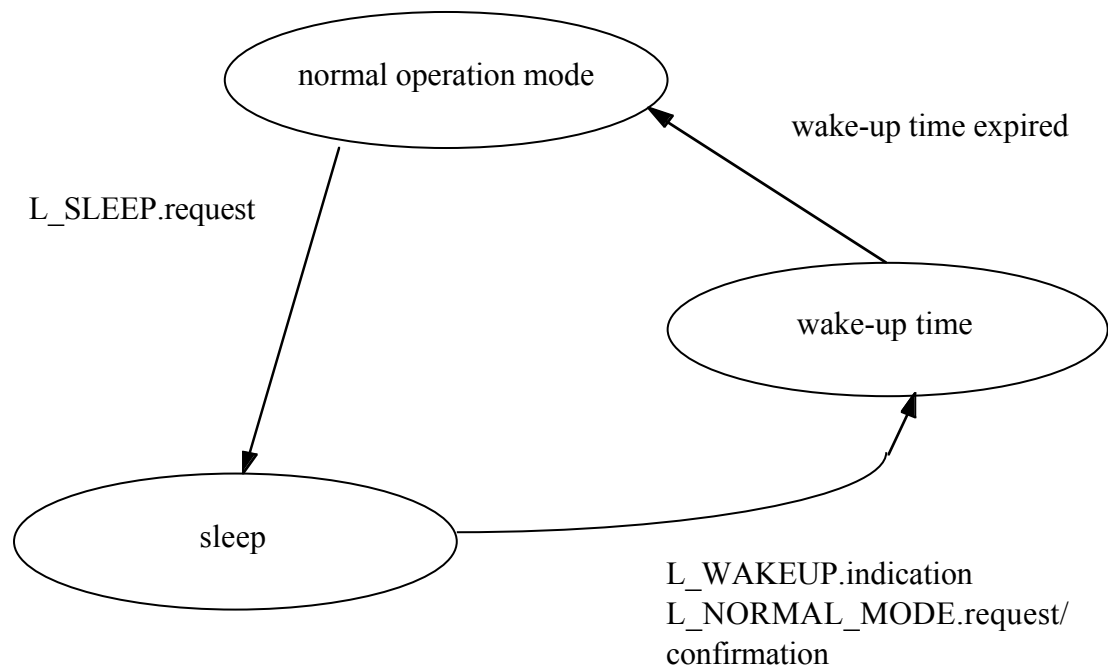


Fig. 3.1 State diagram of the CAN node with de-facto LLC power reduction capabilities

A node's CPU may set its CAN controller from normal operation mode into sleep mode, when the bus is in idle state. The transition from normal operation mode to sleep mode is called going asleep and is requested by the LLC client by the `L_SLEEP.request` and performed by LLC without indication.

Once put into sleep mode, the CAN controller observes the bus lines, to detect the end of the bus idle state. It leaves this mode after the bus becomes busy, i.e. after a dominant bit is monitored (start of frame bit). Usually this will be the transmission of any frame through the network. Note that this frame will not be received properly by sleeping nodes but will cause them to wake-up. It is not possible to selectively wake-up a subset of the sleeping nodes via the CAN-bus.

For leaving the sleep mode (called wake-up), a CAN controller or respectively a CAN node requires an implementation specific time (called wake-up time) to restore the normal operation mode (i.e. for resynchronisation). During this time period for wakup, frames can not be received or sent correctly by the controller.

Note that if only one node is in normal operation mode and several sleeping nodes are attached to a network, a frame transmitted by the normal operation node will not be acknowledged by the other nodes as long as they have not reentered normal operation mode and therefore will be retransmitted until the „fastest“ node has terminated wake-up and has received properly.

The wake up process may also be initiated by the LLC client entity by the `L_NORMAL_MODE.request`.

CAN Power Management Layer Specification

As the CAN subsystem behaves different in sleep mode (no proper immediate reception), higher layer protocols usually don't refer to it and expect the LLC remaining in normal operation mode.

3.2 Power Management Layer Conception

The current CiA DS-201 specification "CAN in OSI Reference Model" /3/ defines a three layered architecture for open CAN bus systems. This model defines a trivial functionality for the Power Management Layer, i.e. the functionality of the Data Link Layer is referenced and offered to the higher layers directly.

In order to take advantage of the power reduction capabilities, the Power Management Layer standby capability introduces a state machine with an IDLE state (in which switching the CAN controller to the sleep mode is allowed) and in which the loss of the first incoming CAN frame and a delay to restore normal operation mode is foreseen by the communication partner. When a frame should be sent from a standby-capable node to another one and the nodes are in IDLE state, this frame has to be preceded by two frames, with the purpose to wake up the CAN controllers (wakeup-unqualified) and to repeat this wakeup (wakeup-qualified). Two frames are necessary to distinguish this wakeup from possible noise or unqualified disturbance by non-standby-capable nodes. Between the unqualified and the qualified wake-up-service an appropriate wake-up time delay will allow resynchronisation of the sleeping CAN controllers.

All nodes with Power Management Layer standby capability enter the IDLE state synchronously. For this purpose, all nodes in the network are equipped with a special Window Timer which is reset after reception or transmission of any valid CAN frame over the bus. Until a special Minimum Active Time expires, setting the CAN controller into sleep mode is not allowed. No wakeup CAN frame is required within the Minimum Active Time period to continue the communication; the protocol overhead is kept at a minimum.

NOTE:

For the implementation of the CAN Power Management Layer, the nodes' CAN peripheral controller must be able to detect **all** CAN frames on the bus (although the content of the frames are not relevant for the SLSU). Usually this can be performed only if message filtering is disabled and **all** L_DATA.indications, L_REMOTE_indications and all L_DATA and L_REMOTE.confirms are notified to the Power Management Layer Service Unit SLSU.

Due to deviations in the node's timebases and to the possible coexistence of non-standby capable nodes (no Power Management Layer, immediate transmission of frames), the implementation of the mechanism is more complicated than described here. More details can be found in chapter 5.

The mechanism is completely independent from the client i.e. higher layer instance, which is requesting the transmission of a particular frame. Therefore, a convenient solution for the implementation of the above mentioned mechanism is to specify a Power Management Layer, which encapsulates and translates all the service calls to the data link layer entity. Thus all higher layer service specifications remain unaffected. Now any higher layer instance sends its

CAN Power Management Layer Specification

transmit-request to the Power Management Layer (instead of directly to the data link layer) which retranslates this service call to two calls of the data link layer transmit-service if necessary.

Consequently, the CAN Power Management Layer specification implements:

1. A set of Power Management Layer services for access from higher levels to replace the data link layer services with exact the same functionality. Depending on the layer configuration the implementation may differ now, i.e. one call to the Power Management Layer services S_DATA.request or S_REMOTE.request may cause several calls to the data link layer service L_DATA.request or L_REMOTE.request and may use extended Layer-2 services (proposed in /2/) to control the power reduction capability.
2. An additional set of Power Management Layer configuration services grants interoperability with nodes who do not support the extended Power Management Layer functionality. These services may be used by level 7 implementations in the context of layer management /4/ /5/.

The main process underlying the Power Management Layer can be understood as a state machine (see **Fig. 5.2**) and is suitable for hardware integration in future controller generations.

Fig. 3.2 shows the OSI layer diagram for the CAN bus after introduction of the Power Management Layer.

CAN Power Management Layer Specification

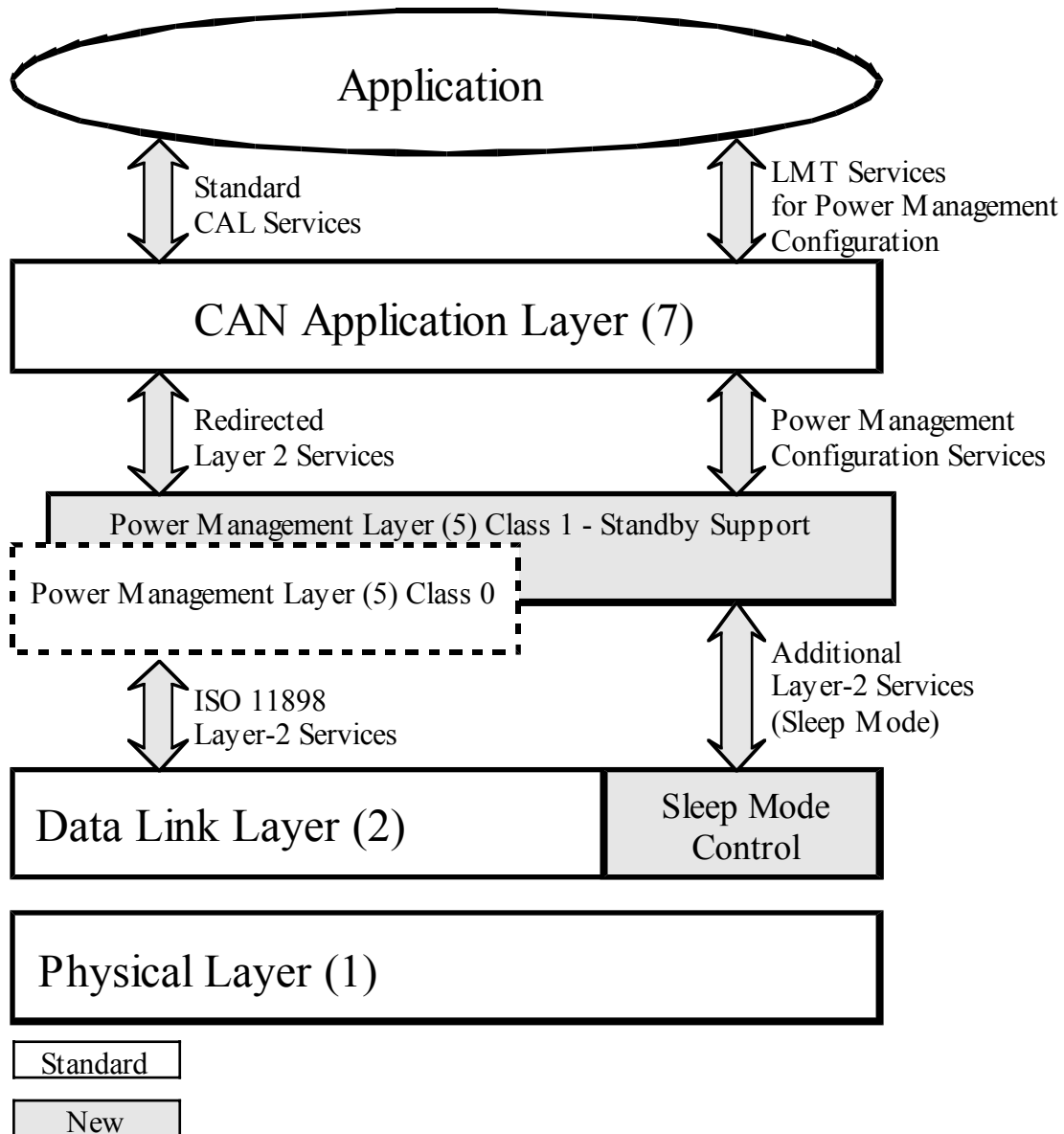


Fig. 3.1: ISO/OSI-Layer Diagram for the CAN Bus with CAN Power Management Layer

3.3 Logical Link Control (LLC) via Power Management Layer Service Unit (SLSU) Capabilities

This Power Management Layer document deals with capabilities and states of LLC and SLSU. In order to keep them apart, they are summarized and ordered in the following table.

CAN Power Management Layer Specification

LLC	SLSU
Power Reduction Capability	Standby Capability
States: <ul style="list-style-type: none"> • normal operational mode • sleep mode • wake-up time (intermediate) 	States: <ul style="list-style-type: none"> • ACTIVE • IDLE • LISTEN / PRE-IDLE / PENDING (intermediate)

4. Additional LLC Services

4.1 Additional LLC Service Primitive Overview

In addition to the existing data link layer services, the following services related to the sleep mode of the CAN controller are defined in accordance with the de facto standard of current hardware implementations:

L_SLEEP.request	Request to enter the sleep mode
L_NORMAL_MODE.request	Request to leave the sleep mode
L_NORMAL_MODE.confir m	Confirmation that the sleep mode was left
L_WAKEUP.indication	Indication that the sleep mode was left due to an activity (dominant state) on the bus lines

These are local services, not causing any communication via the CAN bus.

4.2 Additional LLC Service Primitive Specification

4.2.1 L_SLEEP.request

a) Function

The L_SLEEP.request primitive is passed from the LLC user to the LLC sublayer to request that a CAN node's peripheral controller is switched into the sleep mode.

b) Semantics of the L_SLEEP.request primitive

The primitive has no parameters.

c) Effect on receipt

Receipt of the primitive causes the LLC sublayer to set the CAN controller to sleep mode if possible.

Note: Typical implementation in existing hardware designs: A special bit in one of the controller registers has to be set.

4.2.2 L_NORMAL_MODE.request

a) Function

The L_SLEEP.request primitive is passed from the LLC user to the LLC sublayer to request that a CAN node's peripheral controller should return from sleep mode.

b) Semantics of the L_NORMAL_MODE.request primitive

The primitive has no parameters.

c) Effect on receipt

Receipt of the primitive causes the LLC sublayer to switch the CAN controller back to normal operation.

Note: Typical implementation in existing hardware designs: A special bit in one of the controller registers has to be set or any hardware register has to be accessed.

4.2.3 L_NORMAL_MODE.confirm

a) Function

The L_NORMAL_MODE.confirm primitive is passed from the LLC sublayer to the LLC user to convey the results of the previous L_NORMAL_MODE.request primitive.

b) Semantics of the L_NORMAL_MODE.confirm primitive

The primitive shall provide the following parameters:

L_NORMAL_MODE.confirm (WAKEUP_STATUS)

The WAKEUP_STATUS is used to indicate the completion of leaving the sleep mode, caused by a previous L_NORMAL_MODE.request.

WAKEUP_STATUS: [OK, FAIL]

c) Effect on receipt

The effect on receipt of this primitive by the LLC user is unspecified.

Notes: Typical implementation in existing hardware designs: A special bit in one of the controller registers (status register) is set.

The L_NORMAL_MODE.confirmation informs the user that the sleep mode is left, but does not guarantee that the normal operation mode of the CAN node is restored. The CAN controller may require an implementation specific time (wake-up time) to restore the normal operation mode.

4.2.4 L_WAKEUP.indication

a) Function

The L_WAKEUP.indication primitive is passed from the LLC sublayer to the LLC user to indicate that an activity in the network was detected and that the sleep mode was left automatically.

b) Semantics of the L_WAKEUP.indication primitive

The primitive has no parameters:

c) Effect on receipt

The effect on receipt of this primitive by the LLC user is unspecified.

Notes: Typical implementation in existing hardware designs: A special bit in one of the controller registers (status register) is set, an interrupt request is generated.

The L_WAKEUP.indication informs the user that the sleep mode is left, but does not guarantee that the normal operation mode of the CAN node is restored. The CAN controller may require an implementation specific wake-up time to restore the normal operation state.

5. Power Management Layer

5.1 Power Management Layer Classes

As mentioned in chapter 3.2, the power reduction is the main purpose of the Power Management Layer specification. Two Power Management Layer classes are specified:

Class 0 - no standby support. All Power Management Layer services are trivial. A node with this Power Management Layer class fully complies with the existing CAN specification.

Class 1 - standby support. A node with this Power Management Layer class has a special nontrivial Power Management Layer protocol entity to establish reliable communication and possesses the standby capability.

In order to support interoperability with existing class 0 designs and to be able to operate the class 1 nodes in applications, where real-time performance is more important than power reduction, the behavior of a class 1 system can be altered to be the same as for a class 0 system by special configuration services.

As the class 0 Power Management Layer is trivial, this document **specifies the class 1 Power Management Layer only.**

5.2 Attributes of the Power Management Layer Service Unit (SLSU)

The SLSU has the following attributes:

- **State**

The current state of the SLSU. May have the values [ACTIVE, PRE_IDLE, IDLE, PENDING, LISTEN] (See Fig.2). Conditions for state transitions and state descriptions can be found in chapter 5.3.

- **Standby Support Flag**

May have the values [ON, OFF]. ON allows standby support, as described in this document. When this attribute is OFF, standby support is disabled and real-time performance is slightly enhanced. The SLSU then behaves as if it would be class 0.

Note:

If a node's standby support flag is OFF, a proper communication under all circumstances can only be granted with a class 0 node or another node the standby support flag of which is OFF.

- **HWSleep Support Flag**

May have the values [ON, OFF]. ON allows the SLSU to use the LLC power reduction capabilities of the CAN controller (sleep mode) during the SLSU IDLE state. When HWSleep Support Flag is OFF, during IDLE, PRE_IDLE, LISTEN and PENDING states (see 5.3) the SLSU will receive **all** level 2 services properly, i.e. L_DATA indications will be provided as S_DATA.indications to the SLSU user.

If the HWSleep Support Flag is ON, however, in SLSU IDLE and LISTEN states L_DATA and L_REMOTE indications which belong to „unqualified“ L_DATA and L_REMOTE requests (issued by Power Management Layer class 0 nodes at that time) may be lost but instead reported as L_WAKEUP.indications to the SLSU but not to the SLSU user. (see chapter 5.5)

Note:

A CAN frame sent from a class 0 node can be received under all possible conditions by a class 1 node with the HWSleep Support Flag OFF.

- **Minimum Active Time**

This attribute defines the time interval, in which the SLSU **minimally** remains in ACTIVE state, even if there are no CAN frame on the bus. It is therefore the **maximum** time to issue L_DATA or L_REMOTE requests directly without passing through the PENDING state (see 5.3). The Minimum Active Time is counted by the Window Timer, which is reset in ACTIVE state by every valid CAN frame on the bus, and thereby prolonged.

- **Pre-Idle Time**

CAN Power Management Layer Specification

The synchronisation of the nodes' states might be lost due to slight deviations of the nodes' timebases. Pre-Idle Time in connection with PRE-IDLE state allows resynchronisation of the nodes' states. This attribute defines the time interval during which the SLSU remains in the PRE-IDLE state before entering IDLE state.

- **Listen Time**

This attribute defines a timeout interval, during which the SLSU remains in the LISTEN state before falling back into IDLE state. This timeout is useful to handle erroneous wakeups by noise properly. Listen Time is prolonged by every valid CAN frame on the bus (Window Timer set to zero). Listen time must be longer than pending time.

- **Pending Time**

This attribute defines a time interval value, during which the SLSU is in the PENDING state. This interval state assures, that previously IDLE nodes can recover from sleep mode and move to LISTEN state. Pending Time is **not** prolonged by any valid CAN frame on the bus.

- **Window Timer**

This timer attribute counts up the time, which is spent by SLSU in one of the states ACTIVE, PENDING, PRE-IDLE or LISTEN in order to detect the expiring of the related time intervals. The Window Timer is reset to zero at every state transition (with the exception of entering IDLE state because this state has no time limit). In LISTEN and ACTIVE states, the Window Timer is reset as well by **every** valid CAN frame transmission in the network. In these cases, the Timer counts the time after the last valid CAN frame in the network.

- **Pending Queue**

This attribute represents a FIFO queue, which keeps all data of S_SEND.requests, which should be sent (with delay) in an appropriate state of the SLSU. In PRE_IDLE, IDLE, LISTEN and PENDING state, all S_SEND.requests are accumulated in this queue.

5.3 State Transition Diagram

The Power Management Layer service unit follows a state transition diagram (fig. 5.3). The state transitions in the diagram are caused by SLSU user requests to the Power Management Layer, by expiring of the Window Timer or by indications from the data link layer.

CAN Power Management Layer Specification

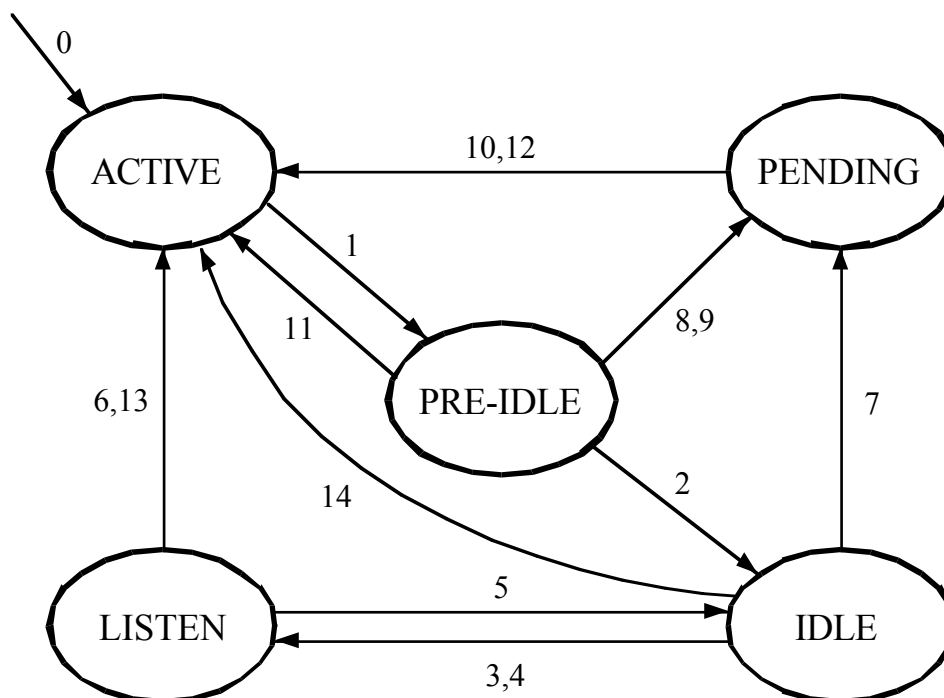


Fig 5. 2: State Diagram of the SLSU

The states of the SLSU have the following meaning:

- **ACTIVE:**

The LLC is in normal operation mode and CAN transmissions are performed without any protocol overhead. S_SEND.requests which are received from the SLSU user or which are accumulated in the Pending Queue are performed immediately. L_REMOTE.indications and L_DATA.indications are indicated directly to the SLSU user as S_REMOTE.indication and S_DATA indication. For trivial class 0 Power Management Layer support or if the standby support flag is 0, this state is never left, otherwise it is left if no valid CAN frame was transceived on the bus (L_REMOTE.indication, L_DATA.indication, L_REMOTE.confirm and L_DATA.confirm) for an interval longer than the Minimum Active Time. It is important for synchronous operation of all nodes, that the active state is entered and left with low inter-node time jitter at all nodes.

NOTE: For a software implementation of SLSU, the CAN controller must support the possibility to recognize **all** valid CAN frames on the bus (not only these which are filtered and of interest at a node). The SLSU requires a notification at these events to reset the Window Timer.

- **PRE-IDLE**

CAN Power Management Layer Specification

The LLC is still in normal operation mode although Minimum Active Time has expired.

L_REMOTE.indications and L_DATA.indications are indicated directly to the SLSU user as S_REMOTE.indication and S_DATA.indication and any valid CAN frame on the bus cause a state transition to PENDING. If L_DATA or L_REMOTE_requests appear, it is stored in Pending Queue, an unqualified wakeup service is performed and PENDING is entered, too. Otherwise, after Pre-Idle Time has expired, the state changes to IDLE.

This state was introduced to compensate for time jitter: if pre-idle is not **left** totally synchronously by all nodes, „unqualified“ bus transmission by either Power Management Layer class 0 nodes or by nodes performing a unqualified wakeup service could occur at a time when several nodes (group 1) are already in IDLE state, while others might still remain PRE-IDLE (group 2). The state transition rules grant, that group 1 nodes switch from IDLE to LISTEN and that group 2 nodes switch to PENDING. These nodes will generate the WAKEUP_QUALIFIED - Service after Pending Time in order to recover ACTIVE state synchronously for both group 1 and group 2 nodes.

- **IDLE**

Depending on the HW Sleep Flag, the CAN controller may be set to sleep mode or normal operational mode during this Power Management Layer state. The SLSUs at all attached nodes „know“, that this state must be left before they can perform L_DATA.requests. The SLSU Window Timer needs not to run during this standby state, so energy consumption can be reduced significantly.

Nodes whose CAN controller is in sleep mode during IDLE should be capable to detect any dominant state on the bus, so that they perform wakeup from the bus both for CAN frames and noise. If this wakeup is caused by noise or by unqualified frames from Power Management Layer class 0 nodes, this leads to a contemporary state change to LISTEN and a return to IDLE afterwards, because no qualified wakeup service will be received within Listen Time.

Nodes whose CAN controller is **not** in LLC sleep mode during IDLE should detect any valid CAN frame on the bus (L_REMOTE.indication, L_DATA.indication) and switch to LISTEN. (They will not come to LISTEN due to noise, but this plays no significant role)

Note, that L_REMOTE.indications and L_DATA.indications (besides the L_DATA.indication for the unqualified wakeup service) have to be issued as S_REMOTE and S_DATA.indications to SLSU user in this state, too! This is necessary due to the fact, that if the HWSleep Flag was set to off, because the user expected that valid CAN frames might arrive from nodes without Power Management Layer standby capability even during IDLE (see chapter 5.5)

If an S_DATA request is issued by the SLSU user on one of the nodes, this request is stored in Pending Queue, the CAN controller LLC normal operation mode is restored (requiring some

CAN Power Management Layer Specification

wake-up time), the unqualified wakeup service is performed and the state changes to PENDING. This causes other nodes in IDLE to switch to LISTEN.

- **PENDING**

During PENDING state, the CAN controller is in normal operation mode and the SLSU awaits the end of Pending Time which gives other nodes in LISTEN state the possibility to recover from LLC sleep mode to CAN controller normal operation mode.

If an S_DATA request is issued by the SLSU user on one of the nodes, this request is stored in Pending Queue. The Windows Timer is **not** reset to zero by CAN frames on the bus (L_REMOTE.indications and L_DATA.indications) although the indications are issued as S_REMOTE.indications and S_DATA.indications to the SLSU user.

After Pending Time has elapsed, a qualified wakeup service is issued and all standby capability nodes with standby capability are switched to ACTIVE state.

- **LISTEN**

During LISTEN state, the CAN controller recovers normal operation mode (it is usually in LLC wakeup at the start of LISTEN state) and the SLSU awaits either a qualified wakeup service or - as a deadline - the end of Listen Time.

If Listen Time expires, the IDLE state is reentered and depending on the HW Sleep Flag, the CAN controller may be set to sleep mode

The Window Timer is reset to zero by any CAN frame on the bus (L_REMOTE.indications and L_DATA.indications), although these indications are issued as S_REMOTE.indications and S_DATA.indications to the SLSU user. However, if the L_DATA.indication of a qualified wakeup service is received, the node changes to ACTIVE state without passing an S_DATA.indication to the SLSU user.

If an S_DATA request is issued by the SLSU user on one of the nodes, this request is stored in Pending Queue. (If LISTEN is left due to the end of the Listen Time with an S_DATA.request stored in the Pending Queue, the IDLE state is immediately left afterwards again in direction to the PENDING state!)

The conditions for each state transition and the actions to be performed during state transitions are defined in Table 5.3.

Table 5.3

Transition Number	Condition(s)	Action performed
0	Initial transition after system restart	Window Timer set to zero
1	Window Timer value exceeds Minimum Active Time	Window Timer set to zero
2	Pre-Idle Time has expired	L_SLEEP.request sent to LLC

CAN Power Management Layer Specification

		(switch CAN controller into sleep mode)
3	L_DATA(L_REMOTE).indication has been received from the LLC (any valid CAN frame on the bus)	Window Timer set to zero
4	L_WAKEUP.indication has been received from the LLC (any dominant bit on the bus)	Window Timer set to zero
5	Listen Time has expired	L_SLEEP.request sent to LLC (switch CAN controller into sleep mode)
6	L_DATA.indication for the qualified wakeup service has been received (Nodes in the network have performed the SEND_WAKEUP_QUALIFIED protocol).	Window Timer set to zero
7	Pending queue is not empty (user has requested data transmission)	Send L_NORMAL_MODE.request to LLC Perform SEND_WAKEUP_UNQUALIFIED protocol Window timer set to zero
8	Pending queue is not empty (user has requested data transmission)	Perform SEND_WAKEUP_UNQUALIFIED protocol Window Timer set to zero
9	L_DATA(L_REMOTE).indication has been received from the LLC (any valid CAN frame on the bus)	Window Timer set to zero
10	Pending Time has expired	Perform SEND_WAKEUP_QUALIFIED protocol Set Window Timer to zero
11	S_SET_STANDBY_SUPPORT.request (OFF)	Set Window Timer to zero
12	S_SET_STANDBY_SUPPORT.request (OFF)	Set Window Timer to zero
13	S_SET_STANDBY_SUPPORT.request (OFF)	Set Window Timer to zero
14	S_SET_STANDBY_SUPPORT.request (OFF)	Send L_NORMAL_MODE.request to LLC Window Timer set to zero

The action listed in the greyed cells is performed only in case the HWSleep flag is set to ON.

5.4 SEND_WAKEUP protocols description

5.4.1 SEND_WAKEUP_UNQUALIFIED protocols description

The SEND_WAKEUP_UNQUALIFIED protocol is intended to switch (wake up) all CAN controllers which are in IDLE to LISTEN (leave sleep mode). It is not accessible to the SLSU user (no corresponding service request) but initiated by the SLSU internally during state transitions 7 and 8. In order to wake up the communication partner(s), a special stimulation CAN frame is sent using the following LLC request:

L_DATA.request (IDENTIFIER, DLC, DATA)

where IDENTIFIER has the value 2027 (now reserved by the CiA in CAL for module self test), DLC is set to one and the parameter DATA has the value 00H.

The L_DATA.indication on the other nodes for this request (if not processed as L_WAKEUP.indication) should **not** be passed as S_DATA.indication to the SLSU user.

5.4.2 SEND_WAKEUP_QUALIFIED protocols description

The SEND_WAKEUP_QUALIFIED protocol is intended to switch all nodes from LISTEN or PENDING state into SLSU ACTIVE state. It is not accessible to the SLSU user (no corresponding service request) but initiated by the SLSU internally during state transitions 10. In order to activate all communication partners, a special stimulation CAN frame is sent using the following LLC request:

L_DATA.request (IDENTIFIER, DLC, DATA)

where IDENTIFIER has the value 2027 (now reserved by the CiA in CAL for module self test), DLC is set to one and the parameter DATA has the value 0FFH.

The L_DATA.indication on the other nodes for this request should **not** be passed as S_DATA.indication to the SLSU user.

5.5 Rules for Interoperability and Performance Optimisation

There is a set of rules for the system integrator to set up the support flags in order to grant interoperability and optimal system performance.

- i The Standby support flag must be ON to allow communication relationships with other class 1 nodes that have their standby support flag set to ON. Standby support causes some protocol overhead (reduced system response time capabilities because dummy wakeup CAN frames are sent in advance to any frame when the Minimum Active Time has

CAN Power Management Layer Specification

expired), but it is necessary (together with the HWSleep support flag = ON) to allow switching a node into power reduction mode.

- ï Standby support = ON and HWSleep support = OFF is most compatible but is less efficient in real-time performance and does not support the power reduction functionality of the CAN controller.
- ï Standby support = OFF (same behavior as empty Power Management Layer or class 0) is best for real-time performance, if no power reduction is necessary.
- ï Standby support = ON and HWSleep support = ON can be set for all nodes of a network without class 0 nodes and minimizes power consumption.

NOTE:

CAN higher layer protocols like CAL and CANopen include a network master who can manage the state of related slave. Even for processes like the network setup, the identifier distribution process and the node guarding process, no special precautions for usage of the Power Management Layer have to be taken:

1. If all nodes feature Power Management Layer class 1 and their Standby support flag is ON (**including** the masters), all protocol CAN frames - if necessary - are encapsulated by wakeup- frames and all nodes are allowed to have their HWSleep Support Flag ON.
2. If Power Management Layer class 0 nodes are connected to the network too, the masters' Standby support flag should be ON and HWSleep support should be OFF. This setup allows, that class 1 nodes, which communicate with other standby nodes (including the master) **only**, can still use power reduction capabilities!

6. Power Management Layer Service

This document uses the same format for service description as used in /1/. The prefix "S" for the names of the services is used to remind that the services are part of the Power Management Layer. **As far as the service names are the same as in /1/, they directly replace existing layer-2 services for usage in higher layers.**

The Power Management Layer encapsulates all references to the additional LLC services related to the power reduction capabilities and implements the more general standby support. On the other hand configuration services are offered for usage by the layer management of the higher layers.

S_DATA and S_REMOTE primitives are referenced by higher layers instead of direct L_DATA and L_REMOTE service primitives and are redirected to these primitives through the Power Management Layer entity depending from the Power Management Layer state and configuration.

CAN Power Management Layer Specification

6.1 Power Management Layer Service Primitive Overview

This chapter summarizes the service primitives of the Power Management Layer.

S_RESET.request	Request to reset the Power Management Layer service provider to a predefined state
S_DATA.request	Request to send a CAN data frame
S_DATA.indication	Indication of reception of a data frame
S_DATA.confirm	Confirmation that a CAN data frame was sent
S_REMOTE.request	Request to send a CAN remote frame
S_REMOTE.indication	Indication of reception of a remote frame
S_REMOTE.confirm	Confirmation that a CAN remote frame was sent
S_SET_STANDBY_SUPPORT.request	Request to enable/disable the Power Management Layer standby support
S_SET_HWSLEEP_SUPPORT.request	Request to enable/disable the Power Management Layer hardware sleep support
S_SET_MINIMUM_ACTIVE_TIME.request	Request to set the value for the Minimum Active Time duration when standby support flag is ON
S_SET_MINIMUM_ACTIVE_TIME.confirm	Confirmation for the corresponding request
S_SET_PRE-IDLE_TIME.request	Request to set the value for the Pre-Idle Time duration when standby support flag is ON
S_SET_PRE-IDLE_TIME.confirm	Confirmation for the corresponding request
S_SET_PENDING_TIME.request	Request to set the value for the Pending Time duration when standby support flag is ON
S_SET_PENDING_TIME.confirm	Confirmation for the corresponding request
S_SET_LISTEN_TIME.request	Request to set the value for the Listen Time duration when standby support flag is ON
S_SET_LISTEN_TIME.confirm	Confirmation for the corresponding request

6.2 Power Management Layer Service Specification

6.2.1 S_RESET.request

a) Function

The S_RESET.request is passed from the Power Management Layer user to the SLSU to request that the SLSU is set into a default setup.

b) Semantics

This service has no parameters.

c) Effect on receipt

The SLSU attributes are set to the following values:

State = ACTIVE

Standby Support Flag = OFF

HWSleep Support Flag = ON

Window Timer = 0

Pending Queue: Empty (queued messages discarded)

6.2.2 S_DATA.request

a) Function

The S_DATA.request is passed from the Power Management Layer user to request that the SLSU initiates a data frame transmission over the bus.

b) Semantics

S_DATA.request(IDENTIFIER, DLC, DATA)

IDENTIFIER 2027 is not allowed (reserved for wakeup-services).

c) Effect on receipt

Receipt of this primitive causes the SLSU to put the request into the pending queue. Depending on the SLSU configuration and state, the request is fetched immediately from the queue and the corresponding data link layer service L_DATA.request is called.

6.2.3 S_DATA indication

a) Function

The S_DATA.indication is passed from the SLSU to the Power Management Layer user to indicate the reception of a data frame. Depending from its state, the SLSU will reset the Window Timer. With the exception of the L_DATA.indication of the WAKEUP services, all L_DATA.indications are provided as S_DATA.indications to the SLSU user. Note that this may occur in LISTEN and IDLE states too, if the HWSleep Support Flag is not set.

b) Semantics of the S_DATA.indication

S_DATA.indication(
IDENTIFIER

CAN Power Management Layer Specification

```

    DLC
    DATA
)

```

The parameter DATA is insignificant, if the parameter DLC is zero.

c) Effect on receipt

The effect on receipt is unspecified.

6.2.4 S_DATA.confirm

a) Function

The S_DATA.confirm is passed from the SLSU to the Power Management Layer user to convey the results of the previous S_DATA.request primitive. Depending from its state, the SLSU will reset the Window Timer if an L_DATA.confirm indicates the completion of a transfer. All L_DATA.confirms are provided as S_DATA.confirms to the SLSU user.

b) Semantics of the S_DATA.indication

```

S_DATA.confirm(
    TRANSFER_STATUS
)

```

The parameter TRANSFER_STATUS is used to indicate the completion of the the transmission initiated by the previous S_DATA.request primitive [COMPLETE, NOT_COMPLETE].

c) Effect on receipt

The effect on receipt is unspecified.

6.2.5 S_REMOTE.request

a) Function

The S_REMOTE.request is passed from the Power Management Layer user to request that the SLSU initiates a remote frame transmission over the bus.

b) Semantics

```

S_REMOTE.request(IDENTIFIER, DLC)

```

c) Effect on receipt

Receipt of this primitive causes the SLSU to put the request into the pending queue. Depending on the SLSU configuration and state, the request is fetched immediately from the queue and the corresponding data link layer service L_REMOTE.request is called.

CAN Power Management Layer Specification

6.2.6 S_REMOTE indication**a) Function**

The S_REMOTE.indication is passed from the SLSU to the Power Management Layer user to indicate the reception of a remote frame. Depending from its state, the SLSU will reset the Window Timer. All L_DATA.indications are provided as S_DATA.indications to the SLSU user. Note that this may occur in LISTEN and IDLE states too, if the HWSleep Support Flag is not set.

b) Semantics of the S_REMOTE.indication

```
S_REMOTE.indication(
    IDENTIFIER
    DLC
)
```

c) Effect on receipt

The effect on receipt is unspecified.

6.2.7 S_REMOTE confirm**a) Function**

The S_REMOTE.confirm is passed from the SLSU to the Power Management Layer user to convey the results of the previous S_REMOTE.request primitive. Depending from its state, the SLSU will reset the Window Timer if an L_REMOTE.confirm indicates the completion of a transfer. All L_REMOTE.confirms are provided as S_REMOTE.confirms to the SLSU user.

b) Semantics of the S_REMOTE.confirm

```
S_REMOTE.confirm(
    TRANSFER_STATUS
)
```

The parameter TRANSFER_STATUS is used to indicate the completion of the the transmission initiated by the previous S_REMOTE.request primitive [COMPLETE, NOT_COMPLETE].

c) Effect on receipt

The effect on receipt is unspecified.

6.2.8 S_SET_STANDBY_SUPPORT request**a) Function**

The S_SET_STANDBY_SUPPORT.request is passed from the Power Management Layer user to the Power Management Layer to set the value of the Standby Support Flag of the SLSU.

b) Semantics of the S_SET_STANDBY_SUPPORT.request

```
S_SET_STANDBY_SUPPORT.request (VALUE)
```

CAN Power Management Layer Specification

The parameter VALUE may be [ON, OFF].

c) Effect on receipt

Receipt of this request causes the SLSU to set the requested value of the Standby Support Flag and causes a SLSU state transitions..

6.2.9 S_SET_HWSLEEP_SUPPORT request**a) Function.**

The S_SET_HWSLEEP_SUPPORT.request is passed from the Power Management Layer user to the Power Management Layer to set the value of the HWSleep Support Flag of the SLSU.

b) Semantics of the S_SET_HWSLEEP_SUPPORT.request

S_SET_HWSLEEP_SUPPORT.request(VALUE)

The parameter VALUE may be [ON, OFF].

c) Effect on receipt

Receipt of this request causes the SLSU to set the requested value to the HWSleep Support Flag of the SLSU.

6.2.10 S_SET_MINIMUM_ACTIVE_TIME.request**a) Function**

The S_SET_MINIMUM_ACTIVE_TIME request is passed from the Power Management Layer user to the Power Management Layer to set the value of the Minimum Active Time.

b) Semantics of the S_SET_MINIMUM_ACTIVE_TIME.request

S_SET_MINIMUM_ACTIVE_TIME.request (WINDOW)

The parameter WINDOW represents the requested value for the Minimum Active Time.

c) Effect on receipt

Receipt of this request causes SLSU to set the requested value for the Minimum Active Time.

6.2.11 S_SET_MINIMUM_ACTIVE_TIME confirm**a) Function**

The S_SET_MINIMUM_ACTIVE_TIME.confirm is passed from the Power Management Layer to the Power Management Layer user to convey the results of the previous S_SET_MINIMUM_ACTIVE_TIME.request.

b) Semantics of the S_SET_MINIMUM_ACTIVE_TIME.confirm

S_SET_MINIMUM_ACTIVE_TIME.confirm (STATUS)

The parameter STATUS is used to indicate results of the S_SET_WINDOW.request.

STATUS: [SUCCESS, FAIL].

c) Effect on receipt

The effect on receipt is unspecified.

6.2.12 S_SET_PRE-IDLE_TIME request

a) Function

The S_SET_PRE-IDLE_TIME.request is passed from the Power Management Layer user to the Power Management Layer to set the value of the Pre-Idle Time of the SLSU.

b) Semantics of the S_SET_PRE-IDLE_TIME.request

S_SET_PRE-IDLE_TIME.request(PRE-IDLE)

The parameter PRE-IDLE represents the requested value for the Pre-Idle Time.

c) Effect on receipt

Receipt of the request causes SLSU to set the requested value for the Pre-Idle Time.

6.2.13 S_SET_PRE-IDLE_TIME confirm

a) Function

The S_SET_PRE-IDLE_TIME.confirm is passed from the Power Management Layer to the Power Management Layer user to convey the results of the previous S_SET_PRE-IDLE_TIME.request.

b) Semantics of the S_SET_PRE-IDLE_TIME.confirm

S_SET_PRE-IDLE_TIME.confirm(STATUS)

The parameter STATUS is used to indicate results of the S_SET_PRE-IDLE_TIME.request. STATUS: [SUCCESS, FAIL].

c) Effect on receipt

The effect on receipt is unspecified.

6.2.14 S_SET_PENDING_TIME request

a) Function

The S_SET_PENDING_TIME.request is passed from the Power Management Layer user to the Power Management Layer to set the value of the Pending Time of the SLSU.

b) Semantics of the S_SET_PENDING_TIME.request

S_SET_PENDING_TIME.request(PENDING)

The parameter PENDING represents the requested value for the Pending Time.

c) Effect on receipt

Receipt of the request causes SLSU to set the requested value for the Pending Time.

6.2.15 S_SET_PENDING_TIME confirm

a) Function

The S_SET_PENDING_TIME.confirm is passed from the Power Management Layer to the Power Management Layer user to convey the results of the previous S_SET_PENDING_TIME.request.

CAN Power Management Layer Specification

b) Semantics of the S_SET_PENDING_TIME.confirm

S_SET_PENDING_TIME.confirm(STATUS)

The parameter STATUS is used to indicate results of the S_SET_PENDING_TIME.request.
STATUS: [SUCCESS, FAIL].

c) Effect on receipt

The effect on receipt is unspecified.

6.2.16 S_SET_LISTEN_TIME request**a) Function**

The S_SET_LISTEN_TIME.request is passed from the Power Management Layer user to the Power Management Layer to set the value of the Listen Time of the SLSU.

b) Semantics of the S_SET_LISTEN_TIME.request

S_SET_LISTEN_TIME.request (LISTEN)

The parameter LISTEN represents the requested value for the Listen Time.

c) Effect on receipt

Receipt of the request causes SLSU to set the requested value for the Listen Time.

6.2.17 S_SET_LISTEN_TIME confirm**a) Function**

The S_SET_LISTEN_TIME.confirm is passed from the Power Management Layer to the Power Management Layer user to convey the results of the previous S_SET_LISTEN_TIME.request.

b) Semantics of the S_SET_PENDING_TIME.confirm

S_SET_LISTEN_TIME.confirm(STATUS)

The parameter LISTEN is used to indicate the results of the S_SET_LISTEN_TIME.request.
STATUS: [SUCCESS, FAIL].

c) Effect on receipt

The effect on receipt is unspecified.