INTERNATIONAL STANDARD

ISO 15765-3

First edition
2004-10-15

Road vehicles — Diagnostics on Controller Area Networks (CAN) —

Part 3:
Implementation of unified diagnostic services (UDS on CAN)

Véhicules routiers — Diagnostic sur gestionnaire de réseau de communication (CAN) —

Partie 3: Mise en œuvre des services de diagnostic unifiés (SDU sur CAN)

Reference number
ISO 15765-3:2004(E)

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

Contents

Page

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of technical committees is to prepare International Standards. Draft International Standards adopted by the technical committees are circulated to the member bodies for voting. Publication as an International Standard requires approval by at least 75 % of the member bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights.

ISO 15765-3 was prepared by Technical Committee ISO/TC 22, *Road vehicles*, Subcommittee SC 3, *Electrical and electronic equipment*.

ISO 15765 consists of the following parts, under the general title *Road vehicles — Diagnostics on Controller Area Networks (CAN)*:

— *Part 1: General information*

— *Part 2: Network layer services*

— *Part 3: Implementation of unified diagnostic services (UDS on CAN)*

— *Part 4: Requirements for emissions-related systems*

Introduction

This part of ISO 15765 has been established in order to enable the implementation of unified diagnostic services, as specified in ISO 14229-1, on controller area networks (UDS on CAN).

To achieve this, it is based on the Open Systems Interconnection (OSI) Basic Reference Model specified in ISO/IEC 7498 and ISO/IEC 10731, which structures communication systems into seven layers. When mapped on this model, the services specified by ISO 15765 are divided into

— unified diagnostic services (layer 7), specified in this part of ISO 15765,

— network layer services (layer 3), specified in ISO 15765-2,

— CAN services (layers 1 and 2), specified in ISO 11898,

in accordance with Table 1.

Table 1 — Enhanced and legislated OBD diagnostic specifications applicable to the OSI layers

Open Systems Interconnection (OSI) layers	Vehicle manufacturer enhanced diagnostics	Legislated on-board diagnostics (OBD)
Diagnostic application	User defined	ISO 15031-5
Application layer	ISO 15765-3	ISO 15031-5
Presentation layer	N/A	N/A
Session layer	ISO 15765-3	N/A
Transport layer	N/A	N/A
Network layer	ISO 15765-2	ISO 15765-4
Data link layer	ISO 11898-1	ISO 15765-4
Physical layer	User defined	ISO 15765-4

Road vehicles — Diagnostics on Controller Area Networks (CAN) —

Part 3: Implementation of unified diagnostic services (UDS on CAN)

1 Scope

This part of ISO 15765 specifies the implementation of a common set of unified diagnostic services (UDS), in accordance with ISO 14229-1, on controller area networks (CAN) in road vehicles as specified in ISO 11898. It gives the diagnostic services and server memory programming requirements for all in-vehicle servers connected to a CAN network and external test equipment. It does not specify any requirement for the in-vehicle CAN bus architecture.

2 Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO 14229-1, *Road vehicles — Unified diagnostic services (UDS) — Part 1: Specification and requirements*

ISO 11898-1, *Road vehicles — Controller area network (CAN) — Part 1: Data link layer and physical signalling*

ISO 11898-2, *Road vehicles — Controller area network (CAN) — Part 2: High-speed medium access unit*

ISO 11898-3, *Road vehicles — Controller area network (CAN) — Part 3: Low-speed, fault-tolerant, medium dependent interface*[1)]

ISO 15031-6, *Road vehicles — Communication between vehicle and external equipment for emissions-related diagnostics — Part 6: Diagnostic trouble code definitions*[1)]

ISO 15765-1, *Road vehicles — Diagnostics on controller area network (CAN) — Part 1: General information*

ISO 15765-2, *Road vehicles — Diagnostics on controller area network (CAN) — Part 2: Network layer service*[1)]

ISO 15765-4, *Road vehicles — Diagnostics on controller area network (CAN) — Part 4: Requirements for emissions-related systems*[1)]

SAE J1939-21, *Recommended practice for a serial control and communications vehicle network — Data link layer*[2)]

---

1) To be published.

2) Society of Automotive Engineers standard.

3 Terms, definitions and abbreviated terms

For the purposes of this document, the terms and definitions given in ISO 14229-1, ISO 15765-1 and ISO 15765-2 and the following abbreviated terms apply.

DA destination address

ID identifier

DLC data length code

GW gateway

LSB least significant bit

MSB most significant bit

NA network address

SA source address

SM subnet mask

TOS type of service

4 Conventions

This part of ISO 15765 is based on conventions defined in ISO 14229-1, which are guided by OSI Service Conventions (see ISO/TR 8509) as they apply for diagnostic services.

5 Unified diagnostic services (UDS) applicability to OSI model

See Figure 1.

6 Application and session layers

6.1 Application layer services

This part of ISO 15765 uses the application layer services as defined in ISO 14229-1 for client-server based systems to perform functions such as test, inspection, monitoring, diagnosis or programming of on-board vehicle servers.

6.2 Application layer protocol

This part of ISO 15765 uses the application layer protocol as defined in ISO 14229-1.

6.3 Application layer and diagnostic session management timing

IMPORTANT — Any N_USData.indication with <N_Result> not equal to N_OK that is generated in the server shall not result in a response message from the server application.

6.3.1 General

The following specifies the application layer and session layer timing parameters and how they are handled for the client and the server.

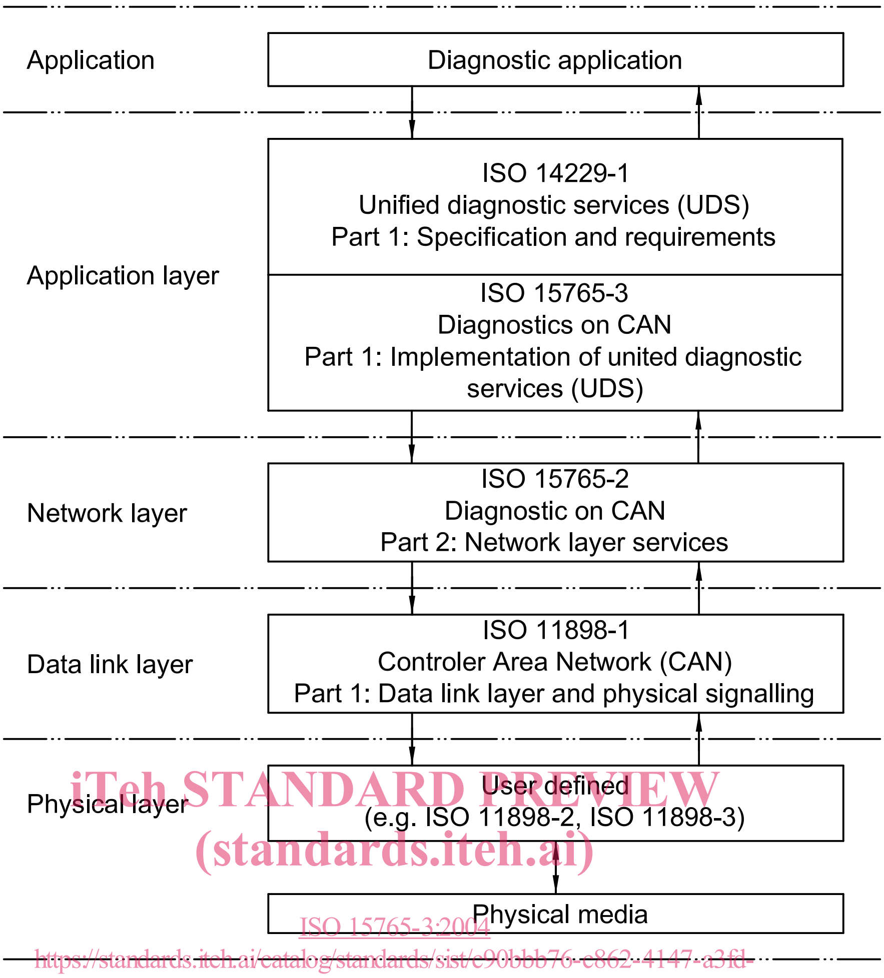


Figure 1 — Implementation of UDS on CAN in OSI model

The following communication scenarios shall be distinguished from one another:

a) physical communication during

1) default session, and

2) non-default session — session handling required;

b) functional communication during

1) default session, and

2) non-default session — session handling required.

For all cases, the possibility of requesting an enhanced response-timing window by the server via a negative response message, including a response code 78 hex, shall be considered.

The network layer services as defined in ISO 15765-2 are used to perform the application layer and diagnostic session management timing in the client and the server.

6.3.2 Application layer timing parameter definitions

The application layer timing parameter values for the default diagnostic session shall be in accordance with Table 2.

Table 2 — Application layer timing parameter definitions for the defaultSession

Timing parameter	Description	Type	Min.	Max.
$P2_{CAN\_Client}$	Timeout for the client to wait after the successful transmission of a request message (indicated via N_USData.con) for the start of incoming response messages (N_USDataFirstFrame.ind of a multi-frame message or N_USData.ind of a SingleFrame message).	Timer reload value	$P2_{CAN\_Server\_max}$ + $\Delta P2_{CAN}$	N/A [a]
$P2^{*}_{CAN\_Client}$	Enhanced timeout for the client to wait after the reception of a negative response message with response code 78 hex (indicated via N_USData.ind) for the start of incoming response messages (N_USDataFirstFrame.ind of a multi-frame message or N_USData.ind of a SingleFrame message).	Timer reload value	$P2^{*}_{CAN\_Server\_max}$ + $\Delta P2_{CAN\_rsp}$	N/A [b]
$P2_{CAN\_Server}$	Performance requirement for the server to start with the response message after the reception of a request message (indicated via N_USData.ind).	Performance requirement	0	50 ms
$P2^{*}_{CAN\_Server}$	Performance requirement for the server to start with the response message after the transmission of a negative response message (indicated via N_USData.con) with response code 78 hex (enhanced response timing).	Performance requirement	0 [c]	5000 ms
$P3_{CAN\_Client\_Phys}$	Minimum time for the client to wait after the successful transmission of a physically addressed request message (indicated via N_USData.con) with no response required before it can transmit the next physically addressed request message (see 6.3.5.3).	Timer reload value	$P2_{CAN\_Server\_max}$	N/A [d]
$P3_{CAN\_Client\_Func}$	Minimum time for the client to wait after the successful transmission of a functionally addressed request message (indicated via N_USData.con) before it can transmit the next functionally addressed request message in case no response is required or the requested data is only supported by a subset of the functionally addressed servers (see 6.3.5.3).	Timer reload value	$P2_{CAN\_Server\_max}$	N/A [d]

[a] The maximum time a client waits for a response message to start is at the discretion of the client, provided that $P2_{CAN\_Client}$ is greater than the specified minimum value of $P2_{CAN\_Client}$.

[b] The value that a client uses for $P2^{*}_{CAN\_Client}$ is at the discretion of the client, provided it is greater than the specified minimum value of $P2^{*}_{CAN\_Client}$.

[c] During the enhanced response timing, the minimum time between the transmission of consecutive negative messages, each with response code 78 hex, shall be ½ $P2^{*}_{CAN\_Server\_max}$, with a maximum tolerance of ± 20% of $P2^{*}_{CAN\_Server\_max}$.

[d] The maximum time a client waits until it transmits the next request message is at the discretion of the client, provided that for non-default sessions the $S3_{Server}$ timing is kept active in the server(s).

The parameter $\Delta P2_{CAN}$ considers any system network design-dependent delays such as delays introduced by gateways and bus bandwidth plus a safety margin (e.g. 50 % of worst case). The worst-case scenario (transmission time necessary for one “round trip” from client to server and back from server to client), based on system design, is impacted by

a) the number of gateways involved,

b) CAN frame transmission time (baud rate),

c) CAN bus utilization, and

d) the CAN device driver implementation method (polling vs interrupt) and processing time of the network layer.

The value of $\Delta P2_{CAN}$ is divided into the time to transmit the request to the addressed server and the time to transmit the response to the client:

$$\Delta P2_{CAN} = \Delta P2_{CAN\_Req} + \Delta P2_{CAN\_Rsp}$$

Figure 2 provides an example of how $\Delta P2_{CAN}$ can be composed.

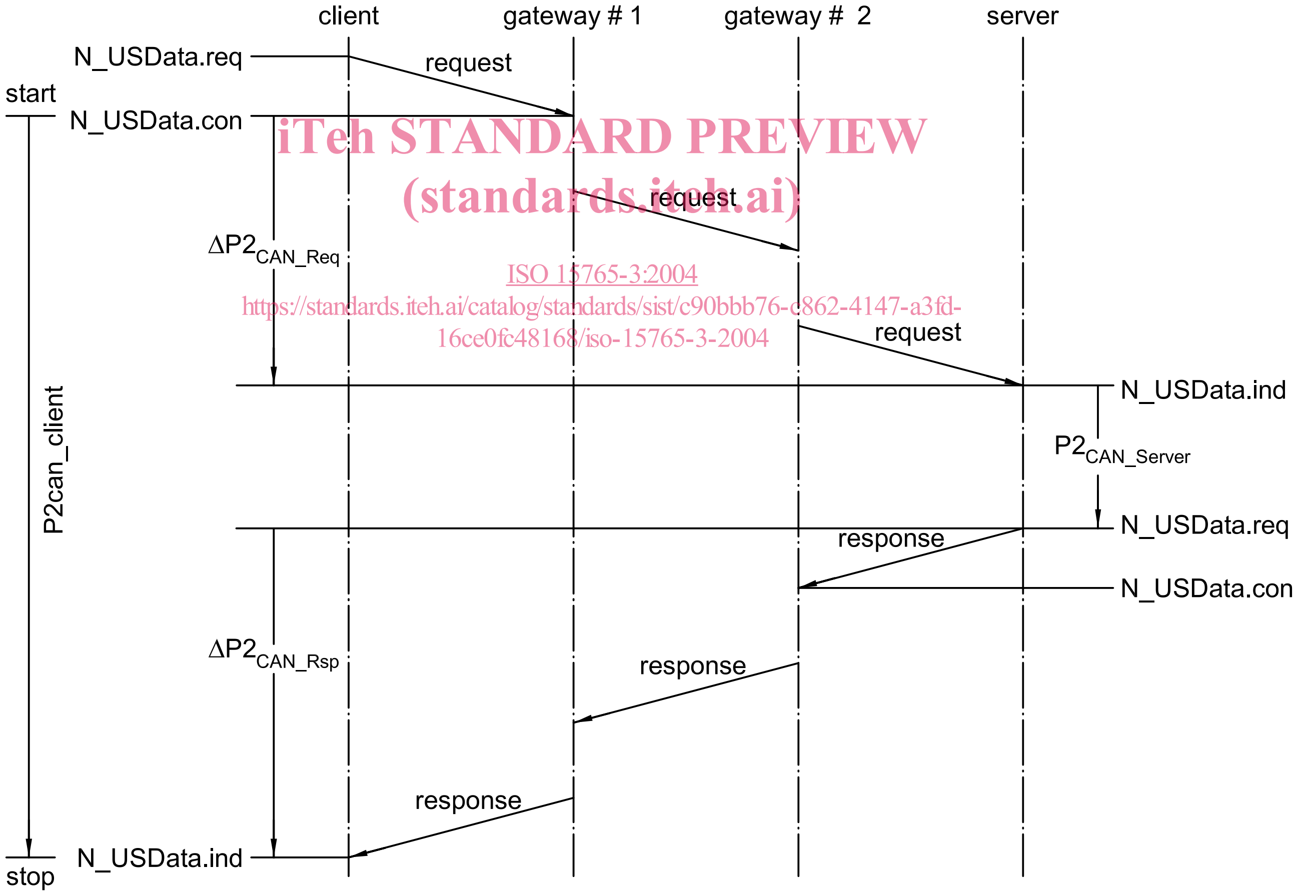


Figure 2 — Example for $\Delta P2_{CAN}$ — SingleFrame request and response message

NOTE For the sake of simplicity in describing the timing parameters, in all the figures that follow it is assumed that the client and the server are located on the same network. All descriptions and figures are presented in a time-related sequential order.

6.3.3 Session layer timing parameter definitions

When a diagnostic session other than the defaultSession is started, then a session handling is required which is achieved via the session layer timing parameter given in Table 3.

Table 3 — Session layer timing parameter definitions

Timing parameter	Description	Type	Recommended timeout ms	Timeout ms
$S3_{Client}$	Time between functionally addressed TesterPresent (3E hex) request messages transmitted by the client to keep a diagnostic session other than the defaultSession active in multiple servers (functional communication) or maximum time between physically transmitted request messages to a single server (physical communication).	Timer reload value	2 000 ms	4 000 ms
$S3_{Server}$	Time for the server to keep a diagnostic session other than the defaultSession active while not receiving any diagnostic request message.	Timer reload value	N/A	5 000 ms

Furthermore, the server might change its application layer timings $P2_{CAN\_Server}$ and $P2^*_{CAN\_Server}$ when transitioning into a non-default session in order to achieve a certain performance or to compensate restrictions which might apply during a non-default diagnostic session. The applicable timing parameters for a non-default diagnostic session are reported in the DiagnosticSessionControl positive response message in the case where a response is required to be transmitted (see service description in 9.2.1) or have to be known in advance by the client in case no response is required to be transmitted. When the client starts a non-default session functionally, then it shall adapt to the timing parameters of the responding servers.

Table 4 defines the conditions for the client and the server to start/restart its $S3_{Client}$/$S3_{Server}$ timer. For the client a periodically transmitted functionally addressed TesterPresent (3E hex) request message shall be distinguished from a sequentially transmitted physically addressed TesterPresent (3E hex) request message, which is only transmitted in case of the absence of any other diagnostic request message. For the server there is no need to distinguish between that kind of TesterPresent (3E hex) handling. Furthermore, Table 4 shows that the $S3_{Server}$ timer handling is based on the network layer service primitives, which means that the $S3_{Server}$ timer is also restarted upon the reception of a diagnostic request message that is not supported by the server.

6.3.4 Client and server timer resource requirements

The timer resource required for the client and the server to fulfil the above given timing requirements during the default session and any non-default session shall be in accordance with Tables 5 and 6 list. During a non-default session, the additional timer resource requirements given in Table 6 shall apply for the client and the server.

Table 4 — Session layer timing start/stop conditions for the client and the server

<table>
<tr><th>Timing Parameter</th><th>Action</th><th>Physical and functional communication, using functionally addressed, periodically transmitted TesterPresent request message</th><th>Physical communication only, using a physically addressed, sequentially transmitted TesterPresent request message</th></tr>
<tr><td rowspan="5">$S3_{Client}$</td><td rowspan="2">Initial start</td><td rowspan="2">N_USData.con that indicates the completion of the DiagnosticSessionControl (10 hex) request message. This is only true for if the session type is a non-default session.</td><td>N_USData.con that indicates the completion of the DiagnosticSessionControl (10 hex) request message in case no response is required.</td></tr>
<tr><td>N_USData.ind that indicates the reception of the DiagnosticSessionControl (10 hex) response message in case a response is required.</td></tr>
<tr><td rowspan="3">Subsequent start</td><td rowspan="3">N_USData.con that indicates the completion of the functionally addressed TesterPresent (3E hex) request message, which is transmitted each time the $S3_{Client}$ timer times out.</td><td>N_USData.con that indicates the completion of any request message in case no response is required.</td></tr>
<tr><td>N_USData.ind that indicates the reception of any response message in case a response is required.</td></tr>
<tr><td>N_USData.ind that indicates an error during the reception of a multi-frame response message.</td></tr>
<tr><td rowspan="7">$S3_{Server}$</td><td rowspan="2">Initial start</td><td colspan="2">N_USData.con that indicates the completion of the transmission of a DiagnosticSessionControl positive response message for a transition from the default session to a non-default session, in case a response message is required.</td></tr>
<tr><td colspan="2">Successful completion of the requested action of the service DiagnosticSessionControl (10 hex) for a transition from the default session to a non-default session, in case no response message is required/allowed.</td></tr>
<tr><td>Subsequent stop</td><td colspan="2">N_USDataFirstFrame.ind that indicates the start of a multi-frame request message or N_USData.ind that indicates the reception of any SingleFrame request message. If the defaultSession is active, the $S3_{Server}$ timer is disabled.</td></tr>
<tr><td rowspan="4">Subsequent start</td><td colspan="2">N_USData.con that indicates the completion of any response message that concludes a service execution (final response message) in case a response message is required/allowed to be transmitted (this includes positive and negative response messages). A negative response with response code 78 hex does not restart the $S3_{Server}$ timer.</td></tr>
<tr><td colspan="2">Completion of the requested action (service conclusion) in case no response message (positive and negative) is required/allowed.</td></tr>
<tr><td colspan="2">N_USData.ind that indicates an error during the reception of a multi-frame request message.</td></tr>
<tr><td colspan="2">See 6.3.5.4 for further details regarding the $S3_{Server}$ handling in the server when the server is requested to transmit unsolicited response message such as periodic data or responses based on an event.</td></tr>
</table>

Table 5 — Timer resources requirements during defaultSession

Timing parameter	Client	Server
$P2_{CAN\_Client}$	A single timer is required for each logical communication channel (physical and functional communication), e.g. each point-to-point communication requires a separate communication channel.	N/A
$P2_{CAN\_Server}$	N/A	An optional timer might be required for the enhanced response timing in order to ensure that subsequent negative response messages with response code 78 hex are transmitted prior to the expiration of $P2^{*}_{CAN\_Server}$.
$P3_{CAN\_Physical}$	A single timer is required per logical physical communication channel.	N/A
$P3_{CAN\_Functional}$	A single timer is required per logical functional communication channel.	N/A

Table 6 — Additional timer resources requirements during non-defaultSession

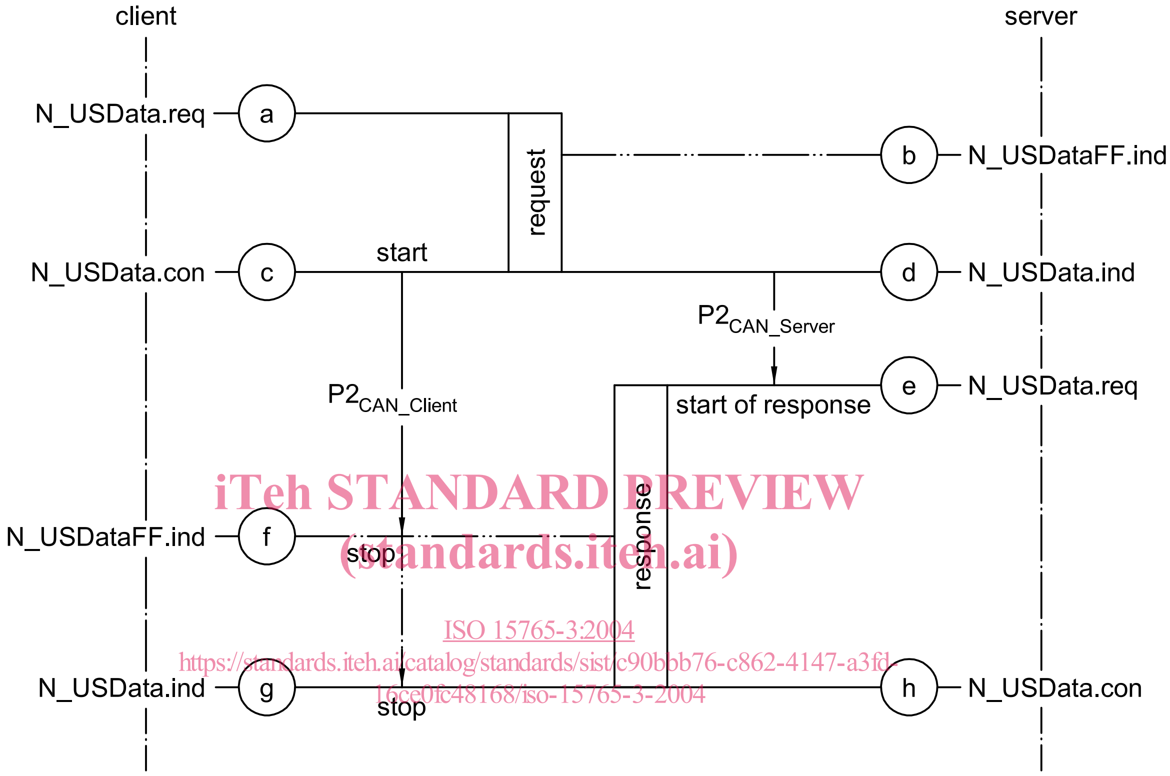
Timing Parameter	Client	Server
$S3_{Client}$	A single timer is required when using a periodically transmitted, functionally addressed TesterPresent (3E) hex request message to keep the servers in a non-defaultSession. There is no need for additional timers per activated diagnostic sessions.	N/A
	A single timer is required for each point-to-point communication channel when using a sequentially transmitted, physically addressed TesterPresent (3E) hex request message to keep a single server in a non-defaultSession in case of the absence of another diagnostic request message then.	
$S3_{Server}$	N/A	A single timer is required in the server, because only a single diagnostic session can be active at a time in a single server.

6.3.5 Detailed timing parameter descriptions

6.3.5.1 Physical communication

6.3.5.1.1 Physical communication during defaultSession

Figure 3 graphically depicts the timing handling in the client and the server for a physically addressed request message during the default session.



[a] The diagnostic application of the client starts the transmission of the request message by issuing a N_USData.req to its network layer. The network layer transmits the request message to the server. The request message can either be a single-frame message or a multi-frame message.

[b] In the case of a multi-frame message, the start of the request is indicated in the server via N_USDataFF.ind that is issued by its network layer.

[c] The completion of the request message is indicated in the client via N_USData.con. When receiving the N_USData.con the client starts its $P2_{CAN\_Client}$ timer, using the default reload value $P2_{CAN\_Client}$. The value of the $P2_{CAN\_Client}$ timer shall consider any latency that is involved based on the vehicle network design (communication over gateways, bus bandwidth, etc.). For simplicity, the figure assumes that the client and the server are located on the same network.

[d] The completion of the request message is indicated in the server via the N_USData.ind.

[e] The server is required to start with its response message within $P2_{CAN\_Server}$ after the reception of N_USData.ind. This means that, in the case of a multi-frame response message, the FirstFrame shall be sent within $P2_{CAN\_Server}$ and, for single-frame response messages, that the SingleFrame shall be sent within $P2_{CAN\_Server}$.

[f] In the case of a multi-frame response message, the reception of the FirstFrame is indicated in the client via the N_USDataFF.ind of its network layer. When receiving the FirstFrame indication, the client stops its $P2_{CAN\_Client}$ timer.

[g] The network layer will generate a final N_USData.ind in case the complete message is received or an error occurred during the reception. In case of a single-frame response message, the reception of the SingleFrame is indicated in the client via a single N_USData.ind. When receiving this single frame indication, the client stops its $P2_{CAN\_Client}$ timer.

[h] The completion of the response message is indicated in the server via N_USData.con.

Figure 3 — Physical communication during default session

