BS ISO 11783-4:2011

BSI Standards Publication

# Tractors and machinery for agriculture and forestry — Serial control and communications data network

## Part 4: Network layer

bsi.

...making excellence a habit.™

**National foreword**

This British Standard is the UK implementation of ISO 11783-4:2011.

The UK participation in its preparation was entrusted to Technical Committee AGE/6, Agricultural tractors and forestry machinery.

A list of organizations represented on this committee can be obtained on request to its secretary.

This publication does not purport to include all the necessary provisions of a contract. Users are responsible for its correct application.



ISBN 978 0 580 66346 8

ICS 35.240.99; 65.060.01

**Compliance with a British Standard cannot confer immunity from legal obligations.**

This British Standard was published under the authority of the Standards Policy and Strategy Committee on 31 December 2011.

**Amendments issued since publication**

| Date | Text affected |
|---|---|

BS ISO 11783-4:2011

# INTERNATIONAL STANDARD

# ISO 11783-4

Second edition
2011-12-01

# Tractors and machinery for agriculture and forestry — Serial control and communications data network —

## Part 4: **Network layer**

*Tracteurs et matériels agricoles et forestiers — Réseaux de commande et de communication de données en série —*

*Partie 4: Couche réseau*

Reference number
ISO 11783-4:2011(E)

# 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.

ISO 11783-4 was prepared by Technical Committee ISO/TC 23, *Tractors and machinery for agriculture and forestry*, Subcommittee SC 19, *Agricultural electronics*.

This second edition cancels and replaces the first edition (ISO 11783-4:2001), which has been technically revised.

ISO 11783 consists of the following parts, under the general title *Tractors and machinery for agriculture and forestry — Serial control and communications data network*:

— *Part 1: General standard for mobile data communication*

— *Part 2: Physical layer*

— *Part 3: Data link layer*

— *Part 4: Network layer*

— *Part 5: Network management*

— *Part 6: Virtual terminal*

— *Part 7: Implement messages application layer*

— *Part 8: Power train messages*

— *Part 9: Tractor ECU*

— *Part 10: Task controller and management information system data interchange*

— *Part 11: Mobile data element dictionary*

— *Part 12: Diagnostics services*

— *Part 13: File server*

— *Part 14: Sequence control*

## Introduction

Parts 1 to 14 of ISO 11783 specify a communications system for agricultural equipment based on ISO 11898-1[1] and ISO 11898-2[2]. SAE J 1939[3] documents, on which parts of ISO 11783 are based, were developed jointly for use in truck and bus applications and for construction and agricultural applications. Joint documents were completed to allow electronic units that meet the truck and bus SAE J 1939 specifications to be used by agricultural and forestry equipment with minimal changes. This part of ISO 11783 is harmonized with SAE J 1939/31[4]. General information on ISO 11783 is to be found in ISO 11783-1.

The purpose of ISO 11783 is to provide an open interconnected system for on-board electronic systems. It is intended to enable electronic control units (ECUs) to communicate with each other, providing a standardized system.

The International Organization for Standardization (ISO) draws attention to the fact that it is claimed that compliance with this part of ISO 11783 may involve the use of a patent concerning the controller area network (CAN) protocol referred to throughout the document.

ISO takes no position concerning the evidence, validity and scope of this patent.

The holder of this patent right has assured ISO that he is willing to negotiate licences under reasonable and non-discriminatory terms and conditions with applicants throughout the world. In this respect, the statement of the holder of this patent right is registered with ISO. Information may be obtained from:

Robert Bosch GmbH
Wernerstrasse 51
Postfach 30 02 20
D-70442 Stuttgart-Feuerbach
Germany

Attention is drawn to the possibility that some of the elements of this part of ISO 11783 may be the subject of patent rights other than those identified above. ISO shall not be held responsible for identifying any or all such patent rights.

# Tractors and machinery for agriculture and forestry — Serial control and communications data network —

# Part 4:
# Network layer

## 1 Scope

ISO 11783 as a whole specifies a serial data network for control and communications on forestry or agricultural tractors and mounted, semi-mounted, towed or self-propelled implements. Its purpose is to standardize the method and format of transfer of data between sensor, actuators, control elements and information storage and display units, whether mounted on, or part of, the tractor or implement. This part of ISO 11783 describes the network layer, which defines the requirements and services needed for communication between control functions (CFs) in different segments of the ISO 11783 network. The various types of network interconnection units are defined in this part of ISO 11783.

## 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 11783-1, *Tractors and machinery for agriculture and forestry — Serial control and communications data network — Part 1: General standard for mobile data communication*

ISO 11783-2, *Tractors and machinery for agriculture and forestry — Serial control and communications data network — Part 2: Physical layer*

ISO 11783-3, *Tractors and machinery for agriculture and forestry — Serial control and communications data network — Part 3: Data link layer*

ISO 11783-5, *Tractors and machinery for agriculture and forestry — Serial control and communications data network — Part 5: Network management*

ISO 11783-7, *Tractors and machinery for agriculture and forestry — Serial control and communications data network — Part 7: Implement messages application layer*

ISO 11783-9, *Tractors and machinery for agriculture and forestry — Serial control and communications data network — Part 9: Tractor ECU*

 

# 3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO 11783-1 and the following apply.

**3.1**
**address space**
allowable range of addresses on a particular subnetwork

NOTE When an NIU separates network segments, the same address can be used by CFs on each side of the NIU.

**3.2**
**connection**
establishment of dynamic virtual addresses in an NIU (network interconnection unit) for sending and receiving messages between CFs on different network segments that have different address spaces

**3.3**
**network interconnection unit**
**NIU**
electronic control unit (ECU) used for interconnecting networks or network segments

**3.4**
**port**
network segment interface to an NIU

NOTE An NIU has two or more ports connected to different network segments.

**3.5**
**port pair**
two ports of an NIU indicating the direction of data flow from one segment to another segment

**3.6**
**transparent**
CF which provides services to another CF without it being aware of the source of these services

NOTE The CFs need not be aware there is an NIU connecting the CFs together.

**3.7**
**virtual CF**
apparent CF established by an NIU on one network segment using the same NAME of the actual CF on a different network segment

**3.8**
**actual CF**
CF established directly by an ECU on the network segment

# 4 Description

## 4.1 Role of the network interconnection unit (NIU)

### 4.1.1 Message transfer

#### 4.1.1.1 General

When multiple segments exist in a network, the NIU provides the means of transferring messages from one segment to another. The unit transfers individual message frames between two or more ports, of which there is one per segment.

 

#### 4.1.1.2 Message-transfer tasks

Depending on its type (see 4.2 and Clause 7), the NIU can perform one or more of the following message-transfer tasks:

— forwarding (6.1);

— filtering (6.2);

— address translation (6.3);

— repackaging (6.4).

#### 4.1.1.3 Main performance criteria

There are three main performance criteria for determining the suitability of an NIU for a given application:

a) *maximum number of messages guaranteed to be forwarded per second*: if this number is exceeded due to average or peak bus loads, messages can be lost;

b) *maximum number of messages guaranteed to be filtered per second*: if this number is exceeded due to the number of entries in the database, messages can be excessively delayed;

c) *maximum transit delay*: this is used to determine the worst-case latency for a message transmitted by one CF and received by another CF on another bus segment.

### 4.1.2 Database management

The NIU can also support bridge and database management (6.6), enabling access to, and configuration of, internal databases within the interconnection unit itself.

EXAMPLE Although a bridge separates two media segments and the message traffic on each, the network will still be considered a single network in terms of its address space and identifiers, as a result of the communication made possible by the interconnection unit.

### 4.1.3 Other network layer functions

Network interconnection units can perform other functions beyond those defined in this part of ISO 11783, as provided by the supplier or as dictated by the network configuration. ISO 11783-1 provides examples of these other functions.

## 4.2 Role of the network layer

The main role of the network layer is management of the transfer of messages between segments. The network layer includes a number of different types of network interconnection units which, depending on the functions required, can provide these services:

— the repeater forwards the messages (7.1);

— the bridge (7.2) filters messages and manages the message-filter database;

— the router (7.3) uses address translation to enable a network segment to appear as a single CF to other parts of the network;

— the gateway (7.4) repackages parameters into different messages for easier transfer, reception and interpretation by CFs;

— a special network interconnection unit, the tractor ECU, connects the implement and tractor buses on a tractor or self-propelled implement (see Figure 1, 5.1.3 and ISO 11783-9).

As well as these message-transfer functions, the network layer provides access to, and allows configuration of databases within, the NIU (4.1.2, 6.6 and ISO 11783-1).

NOTE The NIU can also participate in the address-claim procedure on behalf of CFs in a subnetwork (ISO 11783-5). However, because the use of a router or gateway for interfacing with a proprietary subnetwork is application-dependent, these NIUs are not defined in ISO 11783. Specific implementations can be developed by the component manufacturer, subsystem supplier or the OEM (original equipment manufacturer).

Figure 1 illustrates the topology of a typical network in agriculture and forestry that uses serial control and communications data NIUs. The maximum number of nodes per implement is specified in ISO 11783-2 and the maximum apparent number of CFs on a segment is limited by addressing as specified in ISO 11783-5.

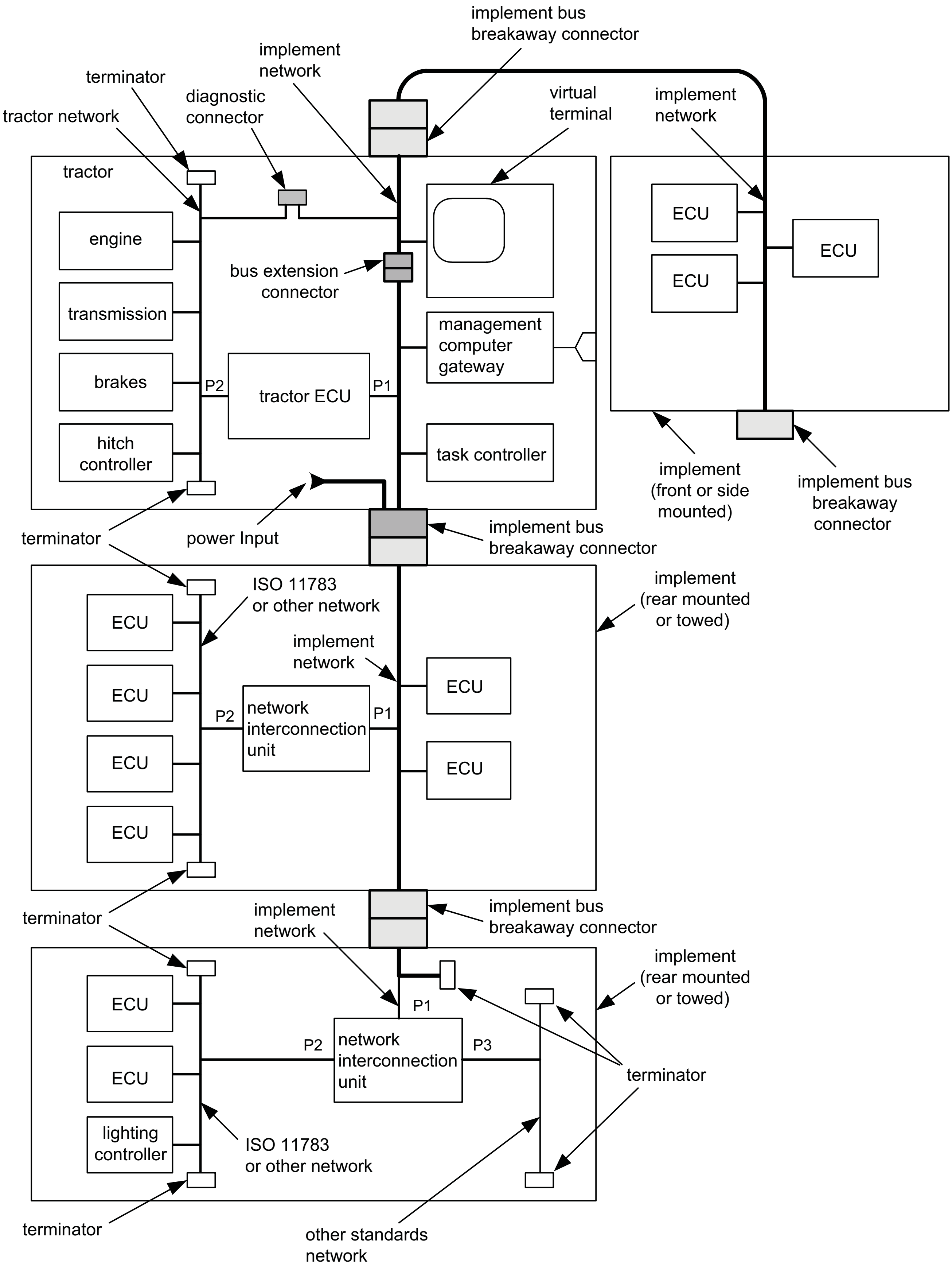


**Figure 1 — Typical ISO 11783 network**

# 5 Requirements

## 5.1 Network interconnection unit (NIU)

### 5.1.1 General requirements

The following apply.

a) The NIU shall provide guaranteed filtering and forwarding rates.

b) It shall not exceed the maximum transit-delay values.

c) In order to avoid excessive delays, the order in which a frame is received on one port and transmitted to another shall follow its given priority.

d) The network interconnection unit shall forward messages having a higher priority before forwarding those of a lower priority.

e) It shall forward the messages, according to their given priority, in the same order as they are received.

f) A simple first-in-first-out (FIFO) message queue shall not be used.

### 5.1.2 General recommendations

The following apply.

a) The NIU should provide the capability to read and modify the filter database.

b) The NIU should support database management by providing standard access for configuration of message forwarding, filtering, address translation and repackaging, as they pertain to bridge, router or gateway management, accordingly.

c) When in operation, the NIU should be transparent to any CF on the network.

### 5.1.3 Tractor ECU

There shall be a special type of NIU, the tractor ECU, located between the tractor's tractor network and implement segments, which shall provide isolation and protect the tractor network segment. Similar to a gateway, the tractor ECU represents the tractor to any other CF on the implement network (see Figure 1).

## 5.2 Network topology

The system network topology (6.7) shall be constructed so that there is only one path between CFs.

NOTE Although this part of ISO 11783 does not require that network loops be detected or duplicate messages be prevented from being generated or replicated indefinitely, it is the responsibility of the OEM to ensure there are no loops in the network. Redundant bus segments for fault tolerance can be used, but the provision of mechanisms for detecting, selecting and automatically reconfiguring the message routing path is the responsibility of the NIU supplier.

## 5.3 Network addressing

The data link layer of the network (ISO 11783-3) provides for 256 source addresses. The theoretical number of CF addresses permitted on the network is 254 when the null and global addresses are not used. The electrical loading on the bus from each ECU can restrict the number of nodes connected to the network (ISO 11783-2).

# 6 Network interconnection unit functions

## 6.1 Forwarding

An NIU transfers individual message frames between two or more ports (one port for each network segment). The order of frames received on one port and transmitted on another shall be preserved for a given priority level. An NIU shall forward all queued messages of a higher priority before those of a lower priority. Otherwise, all messages being forwarded to a specific port could be excessively delayed. A simple FIFO queue shall not be used to meet this requirement.

When an NIU forwards (7.1 Repeater and 7.2 Bridge) a message to another segment in the same address space, it uses an address identical to that of the originator of the message. Ordinarily, this does not cause arbitration problems, as the unit will not retransmit the message to the segment it originally came from and, moreover, the addresses are unique on a given ISO 11783 network.

The sole exception is when an address-claim message is forwarded to a segment in which another CF is simultaneously claiming the same address. In such a low-probability situation, the NIU should stop the automatic retransmission sequence within the CAN protocol chip. Otherwise, the NIU will experience multiple collisions and go “bus off”, thereby preventing other messages from being forwarded until the NIU is able to recover from the bus off condition.

An NIU may begin to forward messages from one segment to another before the NIU has claimed an address (i.e. it does not perform address translations) if it is simply acting as a repeater or bridge.

NOTE Until an NIU has completed a power up sequence and connected it to the network, the subnetwork and the CFs connected to it cannot receive other messages.

## 6.2 Filtering

### 6.2.1 General

For the filtering function, messages sent with Transport Protocol, Extended Transport Protocol, Fast Packet, or other message packeting mechanism, shall be handled according to the parameter group number (PGN) of the contained message. If the PGN of the contained message is defined for the filter, the protocol handling messages shall be processed according to the defined filter.

### 6.2.2 Block mode

In block filter mode (0), the NIU shall default to forwarding all messages (7.2). Bus utilization (traffic) can be higher on each bus segment, but if it is within acceptable limits, the message filtering algorithm will be non-existent. The filter database within the NIU can contain identifier entries (PGN values) for messages which shall not be forwarded (blocked). This can be used to reduce the overall bus traffic on a given segment, and is the preferred mode of operation for bridges conforming to ISO 11783. Filter database entries are typically made during assembly or initial configuration and retained in non-volatile memory.

### 6.2.3 Pass mode

In pass filter mode (1), the NIU shall default to not forwarding messages (7.2). Then, in order for a message to be forwarded, an entry shall exist with a specific identifier (PGN value) for that particular message. This mode is best for ports on NIUs that link subnetworks performing specific functions. It requires prior knowledge of the CFs and the functions of the whole network, or that the CFs be able to add entries to the filter database, in which case the NIU can require more memory and processing power if it is to accommodate a large filter database. Moreover, some entries within the database need to be permanent (i.e. configured to be always present), so that corresponding messages are always forwarded across the whole network. Typical applications are network management, diagnostics and global requests.

## 6.3 Address translation

An NIU can provide address translation for particular messages (7.3), permitting a single address to be used for referencing a particular segment (implement) even without knowledge of a particular function address (lighting) on the segment. For this, an address translation database shall exist in order to identify, through a "look-up" table, the corresponding source address or destination address. The NIU is required to have a valid claimed address before it can provide this service.

## 6.4 Message repackaging

An NIU can repackage messages (7.4) when transferring them from one segment to another. This provides a potential reduction in bus traffic by improving the amount of useful parameters per message, while reducing the number of different messages received by a particular CF. A message repackaging database or processing routine should exist in order to determine how the messages are to be repackaged.

## 6.5 Network message

### 6.5.1 Network message PGN

The PGN for the network message is shown in Table 1.

**Table 1 — Network message**

| Parameter group name | Network |
|---|---|
| Definition | Used to access NIU parametrics and databases |
| Transmission repetition rate | Per user requirements, shall not exceed 5 times per second |
| Data length | Variable |
| Extended data page | 0 |
| Data page | 0 |
| PDU format | 237 |
| PDU-specific field | Destination address |
| Default priority | 6 |
| PGN | 60672 ($00ED00_{16}$) |

The network message provides a means for

- accessing and configuring the database,
- accessing port addresses,
- accessing status and statistics within the network interconnection unit,
- opening and closing a connection between network segments.

When a request or command is made to a specific destination (i.e. it is not global), a response is required, even if only an acknowledgement indicating that the particular function code is not supported or could not be performed. After sending a request or command, the CFs shall wait for a response or the "no response" timeout before sending another request or command (see ISO 11783-3). In the case of multi-packet PGNs, several CAN data frames can occur as a result of a single request.

For variable length messages less than 8 bytes, fill unused bytes with $FF_{16}$. If more than 8 bytes are needed, transport protocol shall be used (see ISO 11783-3).

### 6.5.2 Message function

The first byte in the data field of the network message is used to identify the function required to be acted upon by the receiver of the message. Functions and their corresponding code are listed in Table 2.

**Table 2 — Message function summary**

| Subclause number | Definition | Direction | Function code |
|---|---|---|---|
| 6.6.2.3.1 | Request a copy of filter database | CF to NIU | $0_{10}$ |
| 6.6.2.3.2 | Response to a request for a copy of filter database | NIU to CF | $1_{10}$ |
| 6.6.2.3.3 | Add an entry to the filter database | CF to NIU | $2_{10}$ |
| 6.6.2.3.4 | Delete an entry from the filter database | CF to NIU | $3_{10}$ |
| 6.6.2.3.5 | Clear an entry from the filter database | CF to NIU | $4_{10}$ |
| | Obsolete, not to be used | N/A | $5_{10}$ |
| 6.6.2.3.6 | Create a filter database entry | CF to NIU | $6_{10}$ |
| 6.6.2.3.7 | Request to add NAME qualified filter database entries | CF to NIU | $7_{10}$ |
| | Reserved | N/A | $8_{10}$ - $63_{10}$ |
| 6.7.2.1 | Request a list of source addresses | CF to NIU | $64_{10}$ |
| 6.7.2.2 | Response to a request for a source address list | NIU to CF | $65_{10}$ |
| 6.7.2.3 | Request a source address and NAME list | CF to NIU | $66_{10}$ |
| 6.7.2.4 | Response to a source address list and NAME request | NIU to CF | $67_{10}$ |
| | Reserved | N/A | $68_{10}$ - $127_{10}$ |
| 6.8.3.1 | Request NIU general parametrics | CF to NIU | $128_{10}$ |
| 6.8.3.2 | Response to a request for NIU general parametrics | NIU to CF | $129_{10}$ |
| 6.8.3.3 | Command to reset general statistic parameters | CF to NIU | $130_{10}$ |
| 6.8.4.1 | Request NIU-specific parametrics | CF to NIU | $131_{10}$ |
| 6.8.4.2 | Response to a request for NIU-specific parametrics | NIU to CF | $132_{10}$ |
| 6.8.4.3 | Command to reset specific statistic parameters | CF to NIU | $133_{10}$ |
| | Reserved | N/A | $134_{10}$ - $191_{10}$ |
| 6.9.5.1 | Request to open connection | CF to NIU | $192_{10}$ |
| 6.9.5.2 | Response to request to open connection | NIU to CF | $193_{10}$ |
| 6.9.5.3 | Request to close connection | CF to NIU | $194_{10}$ |
| 6.9.5.4 | Response to request to close connection | NIU to CF | $195_{10}$ |
| | Reserved | | $196_{10}$ - $255_{10}$ |

### 6.5.3 Port numbers

#### 6.5.3.1 General

A port number is represented by a nibble for each numbered port, as shown in Table 3.

**Table 3 — Port numbers**

| Port number | Definition |
|---|---|
| 0 | Local |
| 1 to 14 | Assignable |
| 15 | Global (all ports) |

Port number 0 (local) is used to enable a CF to direct a message to and from an NIU when the number of the connecting port is unknown. The message is directed to the “local” port which receives the message.

The port number 15 (global) is used to facilitate the directing of a message by a CF to an NIU without the CF being required to know the number of ports of the NIU.

#### 6.5.3.2 Port pair (from/to)

When required by the message function, the second byte of the network message is used to identify the direction of messages between ports. The lower nibble (bits 3–0) of the byte identifies the “To” port and the upper nibble (bits 7–4) identifies the “From” port.

If either of the port numbers within the port pair (from/to) is set to global, multiple responses from the NIU for each port pair can be provided.

## 6.6 Database management

### 6.6.1 General

A standard method should be provided for accessing and configuring the various databases within a network interconnection unit, including the unit's parametrics (status and statistics), and the network topology. All the functions concerned should use non-volatile memory to retain the data values through power loss. This is particularly important if a static filter database is to be maintained.

NOTE This part of ISO 11783 leaves undefined the provision for a separate, dynamic filter database, cleared upon power loss to permit easy reconfiguration as CFs are added and removed from the network.

### 6.6.2 Configuration of the network-interconnection-unit message filter database

#### 6.6.2.1 Methods for configuring a message filter database

The message filter database can be configured by the following means.

a) The supplier provides the NIU with a fixed filter database. Bridge design permits the OEM to pre-configure the filter database at the time of manufacture. This requires prior knowledge of the complete network, including the CFs and messages present, and cannot be capable of supporting additions or changes to the network over time without reconfiguration of the NIU during service.

b) The bridge is configured over the network using a diagnostic tool as part of a service procedure.

c) The NIU is configured at any time by any CF on the network. A separate security procedure to enable the modification of the database may be needed, and restrictions on access for reconfiguration will depend on the application.

A NAME shall be appended to each filter-database entry for those entries created using the N.MFDB_Create_Entry database management function (see Table 5). This NAME represents the CF that placed the entry and only the same NAME shall be able to remove it. This does not prevent CFs entering conflicting requests, but it avoids the unexpected deletion of entries. Nevertheless, provisions are to be made to enable the overriding of this requirement when diagnostic tools are to be used.

Each filter database entry identifies a PGN for filtering and marks it to be either passed or blocked (6.2.2 and 6.2.3). Also to be identified is the port pair (direction), necessary for restricting traffic to specific subnetworks while allowing particular messages to be forwarded from them.

When either of a pair (from/to) of port numbers is set to 15 (global), a message can be directed, even when the port number of the NIU is unknown, because of multiple responses from the NIU for every port pair.

The NIU should be capable of configuring the filter database using either local (0) or global (15) port identification methods.

EXAMPLE The tractor ECU filters out engine data to prevent it from going to the implement network segment, while allowing requests from the implement network segment to be forwarded to the tractor.

#### 6.6.2.2 Filter modes

The filter mode for PGNs listed in the filter database is identified by the value of the filter mode byte defined in Table 4.

**Table 4 — Filter mode**

| Value | Definition |
|---|---|
| $0_{10}$ | Block-specific PGNs (default = pass all) |
| $1_{10}$ | Pass-specific PGNs (default = block all) |
| $2_{10}$–$255_{10}$ | Reserved |

#### 6.6.2.3 Messages for configuring a message filter database

##### 6.6.2.3.1 Request a filter database copy (N.MFDB_Request)

The N.MFDB_Request allows a CF to request a copy of the message filter database to be forwarded to it by the NIU.

Transmission repetition rate: As required
Data length: 8 bytes
Parameter group number: 60672 ($00ED00_{16}$), CF to NIU, Destination-Specific

Byte 1 Message function = $0_{10}$ Request a copy of filter database
Byte 2 Port pair
Bits 3–0 "To" port
Bits 7–4 "From" port
Bytes 3–8 Reserved, transmit as $FF_{16}$

##### 6.6.2.3.2 Response to a filter database copy request (N.MFDB_Response)

The NIU response message to a request for a filter database copy includes the entries of the filtered PGN and its filter mode.

Transmission repetition rate: As required
Data length: Variable
Parameter group number: 60672 ($00ED00_{16}$), NIU to CF, Destination-Specific

Byte 1 Message function = $1_{10}$ Response to a request a copy of filter database

| | | | |
|---|---|---|---|
| Byte | 2 | Port pair | |
| | | Bits 3–0 | “To” port |
| | | Bits 7–4 | “From” port |
| Byte | 3 | Filter mode | |
| Bytes | 4–*n* | PGN entries | |

#### 6.6.2.3.3 Add filter database entries command (N.MFDB_Add)

This is the message for adding one or more entries into an NIU filter database. If the “To” port is set to global, the appropriate entry or entries shall be made in the filter database to effect the desired action on each port pair. Any CF using this message shall previously have knowledge of the filter mode of the filter database concerned before making an entry, because the filter mode is not included with this command and cannot be changed without clearing and rebuilding the database for that port pair. Acknowledgment of the command is provided with the Acknowledgment Message (PGN 59392).

| | |
|---|---|
| Transmission repetition rate: | As required |
| Data length: | Variable |
| Parameter group number: | 60672 ($00ED00_{16}$), CF to NIU, Destination-Specific |

| | | | | |
|---|---|---|---|---|
| Byte | 1 | Message function = $2_{10}$ | | Add an entry to the filter database |
| Byte | 2 | Port pair | | |
| | | Bits 3–0 | “To” port | |
| | | Bits 7–4 | “From” port | |
| Bytes | 3–*n* | PGN entries | | |

#### 6.6.2.3.4 Delete a filter database entry command (N.MFDB_Delete)

This is the command message for deleting one or more entries from an NIU filter database. Acknowledgment of the command is provided with the Acknowledgment Message (PGN 59392).

| | |
|---|---|
| Transmission repetition rate: | As required |
| Data length: | Variable |
| Parameter group number: | 60672 ($00ED00_{16}$), CF to NIU, Destination-Specific |

| | | | | |
|---|---|---|---|---|
| Byte | 1 | Message function = $3_{10}$ | | Delete an entry from the filter database |
| Byte | 2 | Port pair | | |
| | | Bits 3–0 | “To” port | |
| | | Bits 7–4 | “From” port | |
| Bytes | 3–*n* | PGN entries | | |

#### 6.6.2.3.5 Clear a filter database command (N.MFDB_Clear)

This is a command message for clearing one or more filter databases for the specified port pair and direction. Acknowledgment of the command is provided with the Acknowledgment Message (PGN 59392).

| | |
|---|---|
| Transmission repetition rate: | As required |
| Data length: | 8 bytes |
| Parameter group number: | 60672 ($00ED00_{16}$), CF to NIU, Destination-Specific |

| | | | | |
|---|---|---|---|---|
| Byte | 1 | Message function = $4_{10}$ | | Clear an entry from the filter database |
| Byte | 2 | Port pair | | |
| | | Bits 3–0 | “To” port | |
| | | Bits 7–4 | “From” port | |
| Bytes | 3–8 | Reserved, transmit as $FF_{16}$ | | |

#### 6.6.2.3.6 Create a filter database entry command (N.MFDB_Create_Entry)

This is the command message that creates one or more entries in a filter database. If the “To” port is set to global, multiple entries may be made in the filter database, one for each Port pair containing the “From” port. The filter mode is included with this command to indicate whether the new entry is for block or pass. The filter mode included with the message cannot change any existing filter modes for the port pair. It is recommended that this command return an error to the requester if the N.MFDB_Create_Entry command is attempted on a non-cleared database. The use of N.MFDB_Add command is sufficient, and (along with the obsolescence of the Set_Mode command) shall eliminate the possibility of requesting that entries be made with differing modes in a single database record. Acknowledgment of the command is provided with the Acknowledgment Message (PGN 59392).

| | |
|---|---|
| Transmission repetition rate: | As required |
| Data length: | Variable |
| Parameter group number: | 60672 ($00ED00_{16}$), CF to NIU, Destination-Specific |

| | | | | |
|---|---|---|---|---|
| Byte | 1 | Message function = $6_{10}$ | | Create a filter database entry command |
| Byte | 2 | Port pair | | |
| | | Bits 3–0 | “To” port | |
| | | Bits 7–4 | “From” port | |
| Byte | 3 | Filter mode | | |
| Bytes | 4–*n* | PGN entries | | |

#### 6.6.2.3.7 Request to Add NAME Qualified filter database entries command (N.MFDBNQ_Add)

This command is used to filter messages based upon the PGN of the message and the NAME of the source.

It can be applied to either a common address space or separate address spaces. When messages are forwarded from a segment with a different address space, if no open connection has been established (see 6.9) the NIU shall forward the messages with its own address as the source address. If an open connection (6.9.5.1) has been established with the sending CF, the SA of the virtual CF shall be used in the forwarded message.

PDU2 messages and PDU1 messages addressed to the global address can be filtered through a bridge. A CF or service tool can request the NIU to pass messages based on a match between PGN and identity of the source. The identity is established by the requester specifying the 64-bit NAME and a 64-bit qualification mask of the desired source. The NIU then associates this identity with a source address from its source address/NAME table.

The forwarded messages can optionally have a “data rate reduction” applied to them. This provides the CF setting up the filter the capability to specify the maximum update rate for the message regardless of the update rate on the original network. For forwarded messages, the most recent data shall be used.

This command message creates a filter database entry that is qualified by NAME field as well as PGN. A maximum transfer rate can optionally be specified.

| | |
|---|---|
| Transmission repetition rate: | As required |
| Data length: | Variable |
| Parameter group number: | 60672 ($00ED00_{16}$), CF to NIU, Destination-Specific |

| | | | | |
|---|---|---|---|---|
| Byte | 1 | Message function = $7_{10}$ | | Request to Add NAME Qualified filter database entries |
| Byte | 2 | Port pair | | |
| | | Bits 3–0 | “To” port | |
| | | Bits 7–4 | “From” port | |
| Byte | 3 | Filter mode | | |
| Byte | 4 | Reserved | | |
| Byte (*t*) | 5 | Maximum transfer rate | | |
| | | 0 | = Reserved | |
| | | 1–250 | = messages per second for this PGN | |

251–254 = ISO use
255 = No rate reduction specified

Byte (*p*) 6–8 PGN entry

Byte (*d*) 9–16 Desired source NAME — NAME of CF from which to accept the PGN

Byte (*n*) 17–24 NAME qualifier — 64 bit mask used to qualify source NAME
1 in any bit position indicates that the bit in the NAME field of source must match the same bit in the desired source NAME.
0 in any bit position indicates that the bit in the NAME field of source is “don't care”, no matching required

Additional bytes *t*, *p*, *d* and *n* are repeated as required for multiple entries.

EXAMPLE The messages in Table 5 are transmitted to obtain the entries of the filter database within the tractor ECU (SA = 240). The destination-specific request is initiated from an off-board diagnostic tool (SA = 249). The request is for a list of only those PGNs being filtered when going to an implement (port 1 to port 2). The destination-specific response indicates that the only blocked message (Filter mode = 0) is engine configuration ($00FEE3_{16}$).

**Table 5 — Example of message filter database access**

| Function | Identifier | | | | | | Data | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | PRI | EDP | DP | PF | DA | SA | Control code | Port pair | Filter mode | PGN | Reserved bytes |
| N.MFDB_Request | 110 | 0 | 0 | 237 | 240 | 249 | $0_{10}$ | $12_{16}$ | $FF_{16}$ | $FFFFFF_{16}$ | $FFFF_{16}$ |
| N.MFDB_Response | 110 | 0 | 0 | 237 | 249 | 240 | $1_{10}$ | $12_{16}$ | $00_{16}$ | $00FEE3_{16}$ | $FFFF_{16}$ |

## 6.7 Network topology information

### 6.7.1 General

All network interconnection units conforming to this part of ISO 11783 shall be transparent to other CFs on the network. Knowledge of the topology of a network can be necessary to properly set up the databases. The messages defined in 6.7.2 provide the capability to obtain this missing information.

The port number, contained in the lower nibble of the data byte, is used to identify the addresses associated with NIU ports. Where there is more than one NIU on a network, it can only identify the port where a source address is located. A particular source address can reside on a remote bus segment, and responses from each of the NIUs shall be compared to determine which local bus segment contains the given source address. The unit shall first perform a request-for-address claim and then construct a list of source addresses associated with each port.

NOTE 300 ms (one network-interconnection-unit delay plus CF response time) is used as the timeout for an address request.

### 6.7.2 Network topology messages

#### 6.7.2.1 Request a source address list (N.NT_Request)

This message is not recommended for new designs. See 6.7.2.3.

The N.NT_Request allows a CF to request the list of source addresses on an NIU port. This network topology message request can be used when the port pair shares the same address space when the requester is trying to determine the physical layout or topology of the ports with respect to the NIU. In the case of a bridge, all address claims appear on both sides of the NIU so the requestor could use this message request to determine which ports are on which segments of the network.

 

Transmission repetition rate: As required
Data length: 8 bytes
Parameter group number: 60672 ($00ED00_{16}$), CF to NIU, Destination-Specific

| | | | |
|---|---|---|---|
| Byte 1 | Message function = $64_{10}$ | | Request a list of source addresses |
| Byte 2 | Port pair | | |
| | Bits 3–0 | Port number | |
| | Bits 7–4 | Set to $F_{16}$ | |
| Bytes 3–8 | Reserved, transmit as $FF_{16}$ | | |

#### 6.7.2.2 Response to a source address list request (N.NT_Response)

This message is not recommended for new designs. See 6.7.2.4

The NIU response message to a request for a source address list includes the entries of source addresses. This network topology message response can be used when the port pair shares the same address space when the requester is trying to determine the physical layout or topology of the ports with respect to the NIU. In the case of a bridge, all address claims appear on both sides of the NIU so the NIU could use this message response to report which ports are on which segments of the network.

Transmission repetition rate: As required
Data length: Variable
Parameter group number: 60672 ($00ED00_{16}$), NIU to CF, Destination-Specific

| | | | |
|---|---|---|---|
| Byte 1 | Message function = $65_{10}$ | | Response of a request for a source address list |
| Byte 2 | Port pair | | |
| | Bits 3–0 | Port number | |
| | Bits 7–4 | Set to $F_{16}$ | |
| Bytes 3–$n$ | Source address entries | | |

#### 6.7.2.3 Request a source address and NAME list (N.NTX_Request)

The Request a source address list and NAME message allows a CF to request the list of source addresses and associated NAMEs on an NIU port. This network topology message request is used when the network segment topology being requested is not part of the same address space as the requestor, i.e. the NIU is acting as a router for this port pair. Thus the requestor will not have seen the address claims of the subject network segment. The request is not allowed for the global port number.

Transmission repetition rate: As required
Data length: 8 bytes
Parameter group number: 60672 ($00ED00_{16}$), CF to NIU, Destination-Specific

| | | | |
|---|---|---|---|
| Byte 1 | Message function = $66_{10}$ | | Request a list of source addresses and NAMEs |
| Byte 2 | Port pair | | |
| | Bits 3–0 | Requested port number (0-14) | |
| | Bits 7–4 | Set to $F_{16}$ | |
| Bytes 3–8 | Reserved, transmit as $FF_{16}$ | | |

#### 6.7.2.4 Response to a source address list and NAME request (N.NTX_Response)

The NIU response message to a request for a source address and NAME list includes all of the entries of source addresses and associated NAMEs on the network segment associated with the defined port including the address and NAME claimed by the NIU.

Transmission repetition rate: As required
Data length: Variable
Parameter group number: 60672 ($00ED00_{16}$), NIU to CF, Destination-Specific

| | | | |
|---|---|---|---|
| Byte | 1 | Message function = $67_{10}$ | Response to a request for a source address and NAME list |
| Byte | 2 | Port pair<br>Bits 3–0 Requested port number (0–14)<br>Bits 7–4 Set to $F_{16}$ | |
| Byte | 3 | Number of source address/NAME pairs (Port_Num) being reported for current port | |
| Byte | 4 | First source address | |
| Bytes | 5–12 | NAME associated with first source address | |
| Byte | 13 | Second source address | |
| Bytes | 14–21 | NAME associated with second source address… | Repeat until a 9-byte source address and NAME pair has been transmitted for the Port_Num ports being reported |

## 6.8 Network interconnection unit parametrics

### 6.8.1 General

There are two sets of message functions for accessing the NIU parametrics (status and statistics). These parametrics are defined in Table 6. General parametrics that are applicable for an NIU are accessed with the general parametric messages. Those parametrics applicable for a given NIU port pair are accessed with the specific parametric messages. Their valid value ranges are in accordance with ISO 11783-7 with all bit values set to “1” to indicate “not available”, and each parameter number 1 byte in length.

Parametrics within a response list are in the same order as the parametric identifiers in the request list (no delimiters are required since all parametrics and parametric identifiers are of fixed length). Although a response to a request message shall be received before another request can be sent to a given address, this request can be for multiple parametrics.

It is recommended that the request always be for Parameter 0, which returns the entire list in numerical order.

**Table 6 — NIU parametrics (status and statistics)**

| Able to be reset? | Parametric identifier | Number of bytes | Definition |
|---|---|---|---|
| | 0 | N/A | Used to request all parameters in numerical order (not used in response message) |
| — | 1 | 2 | Buffer size (bytes) |
| — | 2 | 2 | Maximum filter database size (bytes) |
| — | 3 | 2 | Number of filter database entries |
| — | 4 | 2 | Maximum number of messages received per second |
| — | 5 | 2 | Maximum number of messages forwarded per second |
| — | 6 | 2 | Maximum number of messages filtered per second |
| — | 7 | 2 | Maximum transit delay time (ms) |
| Yes | 8 | 2 | Average transit delay time (ms) |
| Yes | 9 | 2 | Number of messages lost due to buffer overflow |
| Yes | 10 | 2 | Number of messages with excess transit delay time |
| Yes | 11 | 2 | Average number of messages received per second |
| Yes | 12 | 2 | Average number of messages forwarded per second |

**Table 6** (*continued*)

| Able to be reset? | Parametric identifier | Number of bytes | Definition |
|---|---|---|---|
| Yes | 13 | 2 | Average number of messages filtered per second |
| — | 14[a] | 4 | Uptime since last power on reset (s) |
| — | 15 | 1 | Number of ports |
| — | 16 | 1 | Network Interconnection Unit Type |
| | 17–255 | N/A | Reserved for future assignment by ISO |

[a] The parametric identifier "14" has changed from the previous edition of ISO 11783-4. This parameter is now harmonized with SAE J1939-31[4].

### 6.8.2 Status and statistics parameter definitions

#### 6.8.2.1 General

The status and statistics parameter definitions shall be used when the data is reported in the NIU parametric responses such as N.GP_Response or N.SP_Response. Parameters with a data length greater than 1 byte shall conform to the ISO 11783 byte order convention described in ISO 11783-7. The definition of the parameter ranges shall support the parameter ranges convention described in ISO 11783-7, unless explicitly noted otherwise in the parameter definition. Some of these parameters may be applicable for a given NIU, for a specific port pair, or for both.

#### 6.8.2.2 Buffer Size (Parametric 1)

This parameter reports the NIU buffer size in bytes.

Data Length: 2 bytes
Resolution: 1 byte/bit, 0 offset
Data Range: 0 to 64 255 bytes
Data Type: Measured

#### 6.8.2.3 Maximum Filter Database Size (Parametric 2)

This parameter reports the maximum available size for the Filter Database.

Data Length: 2 bytes
Resolution: 1 byte/bit, 0 offset
Data Range: 0 to 64 255 bytes
Data Type: Measured

#### 6.8.2.4 Number of Filter Database Entries (Parametric 3)

This parameter reports the number of entries in the NIU filter database.

Data Length: 2 bytes
Resolution: 1/bit, 0 offset
Data Range: 0 to 64 255 bytes
Data Type: Measured

#### 6.8.2.5 Maximum Messages Received per second (Parametric 4)

This parameter reports the maximum number of messages that the NIU is capable of receiving per second.

Data Length: 2 bytes
Resolution: 1 message/second/bit, 0 offset
Data Range: 0 to 64 255 messages/second
Data Type: Measured

#### 6.8.2.6 Maximum Messages Forwarded per second (Parametric 5)

This parameter reports the maximum number of messages that the NIU is capable of forwarding per second.

Data Length: 2 bytes
Resolution: 1 message/second/bit, 0 offset
Data Range: 0 to 64 255 messages/second
Data Type: Measured

#### 6.8.2.7 Maximum Messages Filtered per second (Parametric 6)

This parameter reports the maximum number of messages that the NIU is capable of filtering per second.

Data Length: 2 bytes
Resolution: 1 message/second/bit, 0 offset
Data Range: 0 to 64 255 messages/second
Data Type: Measured

#### 6.8.2.8 Maximum Transit Delay Time (Parametric 7)

This parameter reports the performance capability for the maximum transit delay time for a message moved through the NIU.

Data Length: 2 bytes
Resolution: 1 millisecond/bit, 0 offset
Data Range: 0 to 64 255 milliseconds
Data Type: Measured

#### 6.8.2.9 Average Transit Delay Time (Parametric 8)

This parameter reports the average of the actual transit delay time for messages moved through the NIU since this data was last reset.

Data Length: 2 bytes
Resolution: 1 millisecond/bit, 0 offset
Data Range: 0 to 64 255 milliseconds
Data Type: Measured

#### 6.8.2.10 Number of Messages lost due to Buffer Overflow (Parametric 9)

This parameter reports the number of messages lost by the NIU due to buffer overflow since this data was last reset.

Data Length: 2 bytes
Resolution: 1 message/bit, 0 offset
Data Range: 0 to 64 255 messages
Data Type: Measured

#### 6.8.2.11 Number of Messages with Excess Transit Delay Time (Parametric 10)

This parameter reports the number of messages with excess transit delay since this data was last reset.

Data Length: 2 bytes
Resolution: 1 message/bit, 0 offset
Data Range: 0 to 64 255 messages
Data Type: Measured

#### 6.8.2.12 Average Messages Received per second (Parametric 11)

This parameter reports the average number of messages received per second since this data was last reset.

Data Length: 2 bytes
Resolution: 1 message/second/bit, 0 offset
Data Range: 0 to 64 255 messages/second
Data Type: Measured

#### 6.8.2.13 Average Messages Forwarded per second (Parametric 12)

This parameter reports the average number of messages forwarded per second since this data was last reset.

Data Length: 2 bytes
Resolution: 1 message/second/bit, 0 offset
Data Range: 0 to 64 255 messages/second
Data Type: Measured

#### 6.8.2.14 Average Messages Filtered per second (Parametric 13)

This parameter reports the average number of messages filtered per second since this data was last reset.

Data Length: 2 bytes
Resolution: 1 message/second/bit, 0 offset
Data Range: 0 to 64 255 messages/second
Data Type: Measured

#### 6.8.2.15 Uptime since last power on reset (Parametric 14)

This parameter reports the total uptime for the NIU since the last power on reset of the device.

Data Length: 4 bytes
Resolution: 1 second/bit, 0 offset
Data Range: 0 to 4 211 081 215 seconds
Data Type: Measured

#### 6.8.2.16 Number of Ports (Parametric 15)

This parameter reports the number of ports on the NIU. All value states are used for valid data for this parameter, except for $F_{16}$ which is reserved for “Not Available” or the parameter data has no meaning.

Data Length: 1 byte
Resolution: 1 port/bit, 0 offset
Data Range: 2 to 14 ports
Data Type: Measured

#### 6.8.2.17 Network Interconnection Unit Type (Parametric 16)

This parameter reports the declaration of the type of network interconnection device. It is possible for the NIU to indicate different NIU types for each port pair. The Network Interconnection Unit Type value assignments are shown in Table 7.

Data Length: 1 byte
Resolution: 256 states/8 bits, 0 offset
Data Range: 0 to 250 bytes
Data Type: Measured

**Table 7 — Network interconnection Unit Types**

| Value | Definition |
|---|---|
| 0 | None (No interaction for the given port pair) |
| 1 | Repeater |
| 2 | Bridge |
| 3 | Router |
| 4 | Gateway |
| 5 | Tractor ECU |
| 6–249 | Reserved |
| 250 | Multiple (Used with general parametric if NIU contains multiple port pair with differing types) |
| 251–255 | Reserved |
| NOTE The value for Repeater or Bridge can only be used if a repeater claims an address to communicate on the network. These types can be transparent and their presence on the network may not be known to other CFs. | |

### 6.8.3 General parametric messages (GP)

#### 6.8.3.1 Request NIU parametrics (N.GP_Request)

The N.GP_Request allows a CF to request one or more NIU parametrics. A request of parametric 0 returns the full list of parametrics. Transport protocol is used whenever a message length is greater than 8 bytes. If fewer than 7 parameters are requested, then the unused data bytes are set to $FF_{16}$. The parametric identifiers shall be listed in ascending numerical order.

Transmission repetition rate: As required
Data length: Variable
Parameter group number: 60672 ($00ED00_{16}$), CF to NIU, Destination-Specific

Byte 1 Message function = $128_{10}$ Request NIU parametrics
Bytes 2–$n$ Parametric identifiers, 1 per byte

#### 6.8.3.2 Response to NIU parametrics request (N.GP_Response)

The NIU response to the request NIU parametrics message contains one or more of its parametrics. The response to a request for parameter zero contains all of the NIU parametrics, in numerical order. If the returned list of parameters is shorter than expected, it indicates that the NIU has no knowledge of additions to the list and stopped with the last known parameter. Transport protocol is used whenever a message length is greater than 8 bytes. If the message length is less than 8 bytes, the unused data bytes are set to $FF_{16}$.

| | |
|---|---|
| Transmission repetition rate: | As required |
| Data length: | Variable |
| Parameter group number: | 60672 ($00ED00_{16}$), NIU to CF, Destination-Specific |

| | | | |
|---|---|---|---|
| Byte | 1 | Message function = $129_{10}$ | Response to a request for NIU parametrics |
| Bytes | 2,3 | First requested parametric | |
| Bytes | 4–$n$ | Remaining requested parametrics | |

#### 6.8.3.3 Reset statistics parametrics command (N.GP_Reset_Statistics)

This is the command message to clear all of the statistical parameters which are resetable, noted by “Yes” in Table 6. Acknowledgment of the command is provided with the Acknowledgment Message (PGN 59392).

| | |
|---|---|
| Transmission repetition rate: | As required |
| Data length: | 8 bytes |
| Parameter group number: | 60672 ($00ED00_{16}$), CF to NIU, Destination-Specific |

| | | | |
|---|---|---|---|
| Byte | 1 | Message function = $130_{10}$ | Command to reset statistic parameters |
| Bytes | 2–8 | Reserved, transmit as $FF_{16}$ | |

### 6.8.4 Specific parametrics (SP)

#### 6.8.4.1 Request NIU-specific parametrics (N.SP_Request)

The N.SP_Request allows a CF to request one or more NIU parametrics. A request of parametric 0 returns the full list of parametrics. Transport protocol is used whenever a message length is greater than 8 bytes. If fewer than 6 parameters are requested, then the unused data bytes are set to $FF_{16}$. The parametric identifiers shall be listed in increasing numerical order.

| | |
|---|---|
| Transmission repetition rate: | As required |
| Data length: | 8 bytes |
| Parameter group number: | 60672 ($00ED00_{16}$), CF to NIU, Destination-Specific |

| | | | |
|---|---|---|---|
| Byte | 1 | Message function = $131_{10}$ | Request NIU-specific parametrics |
| Byte | 2 | Port pair | |
| | | Bits 3–0 | “To” port |
| | | Bits 7–4 | “From” port |
| Bytes | 3–$n$ | Parametric identifiers, 1 per byte | |

#### 6.8.4.2 Response to NIU-specific parametrics request (N.SP_Response)

The NIU response to the request NIU-specific parametrics message contains one or more of its parametrics. The response to a request for parameter zero contains all of the NIU parametrics, in numerical order. If the returned list of parameters is shorter than expected, it indicates that the NIU has no knowledge of additions to the list and stopped with the last known parameter. Transport protocol is used whenever a message length is greater than 8 bytes. If the message length is less than 8 bytes, then the unused data bytes are set to $FF_{16}$.

Transmission repetition rate: As required
Data length: Variable
Parameter group number: 60672 ($00ED00_{16}$), NIU to CF, Destination-Specific

| | | | | |
|---|---|---|---|---|
| Byte | 1 | Message function = $132_{10}$ | | Response to a request for NIU-specific parametrics |
| Byte | 2 | Port pair | | |
| | | Bits 3–0 | "To" port | |
| | | Bits 7–4 | "From" port | |
| Byte | 3,4 | First requested parametric | | |
| Bytes | 5–*n* | Remaining requested parametrics | | |

#### 6.8.4.3 Reset specific statistics parametrics command (N.SP_Reset_Statistics)

This is the command message to clear all of the specific statistical parameters which are resetable, noted by "Yes" in Table 6. Acknowledgment of the command is provided with the Acknowledgment Message (PGN 59392).

Transmission repetition rate: As required
Data length: 8 bytes
Parameter group number: 60672 ($00ED00_{16}$), CF to NIU, Destination-Specific

| | | | | |
|---|---|---|---|---|
| Byte | 1 | Message function = $133_{10}$ | | Command to reset specific statistic parameters |
| Byte | 2 | Port pair | | |
| | | Bits 3–0 | "To" port | |
| | | Bits 7–4 | "From" port | |
| Bytes | 3–8 | Reserved, transmit as $FF_{16}$ | | |

## 6.9 Forwarding destination-specific messages by establishing a connection

### 6.9.1 General

The network in Figure 1 illustrates a network topology using NIUs, which can be gateways or routers, to connect implement subnetworks to the ISO 11783 implement network. Network segments (21) each have their own address spaces which are separate from the implement network. These segments can include a number of CFs comprised of sensors, actuators and low capability controllers which have a lesser amount of communications with CFs in the main network segments.

A typical router (see 7.3) has the ability to forward or route messages from CFs on one network segment to CFs on another network segment that has a different address space. An address translation database in the router is used to determine the address to which to forward a message. This capability allows service tools to perform diagnostics on multiple network segments from a single connection. Downloading software or calibrations can also be facilitated by such a capability. A further example is an implement with multiple network segments and a CF that requires communication with a VT (virtual terminal) which is part of the main implement network.

Instead of using address translation, another type of router can be dynamically configured by CFs to make connections based on the "discover" topology of the network and specific functions that CFs perform as indicated in their NAMEs. This implementation allows message transfers to be processed by the NIU once a connection has been established. The CFs participating in the communications do so with no change to existing software implementations for either single or multi-packet destination-specific messages beyond the initial topology discovery and opening of a connection process.

### 6.9.2 Establishing connections

Figure 2 illustrates the message flow diagram for CF A on network segment 1 that requires communication with CF C on network segment 2. CF A first determines the NAMEs of CFs connected on network 2 by using the N.NTX Request function of the network message directed to the NIU B. NIU B replies with the N.NTX

Response function and the list of NAMEs and addresses on network 2. Once CF A determines from this list that the target CF C is on network 2, it proceeds to request NIU B to establish a connection, i.e. open a connection between the CFs. The NIU B records the connection information for its message forwarding function and then creates the virtual C, or C^, on network 1 by claiming an address for C^ on CF A's network using CF C's same NAME. The address claimed is not necessarily the same as the address CF C claims on network segment 2 but is one that is available on network 1 from the self-configurable address range.

The NIU also creates a virtual A^ on network 2 by claiming an address for A^ on network 2 using CF A's same NAME and an available address.

When CF A receives a successful address claim for C^ on the network, it can then start sending destination-specific messages to C^ just as it would to any other claimed address on its network segment. When NIU B receives a message with the destination address of C^, it determines from the message forwarding information for that virtual CFs address and retransmits the message on network 2 using the address of A^ as the source address in the message header and CF C as the destination address.

CF C receives a message from A^ and responds to the message in the appropriate manner for that message, including but not limited to assessing the security access rights, the actions that its hardware is designed for and/or responding with the required messages, acknowledgements or NACKs (negative acknowledgements).

Connections can also be closed by the CF that established the connection if the connection is not held open by other CFs. If the same connection has been opened multiple times by a single CF, this is considered only a single connection and can be closed with a single close command.

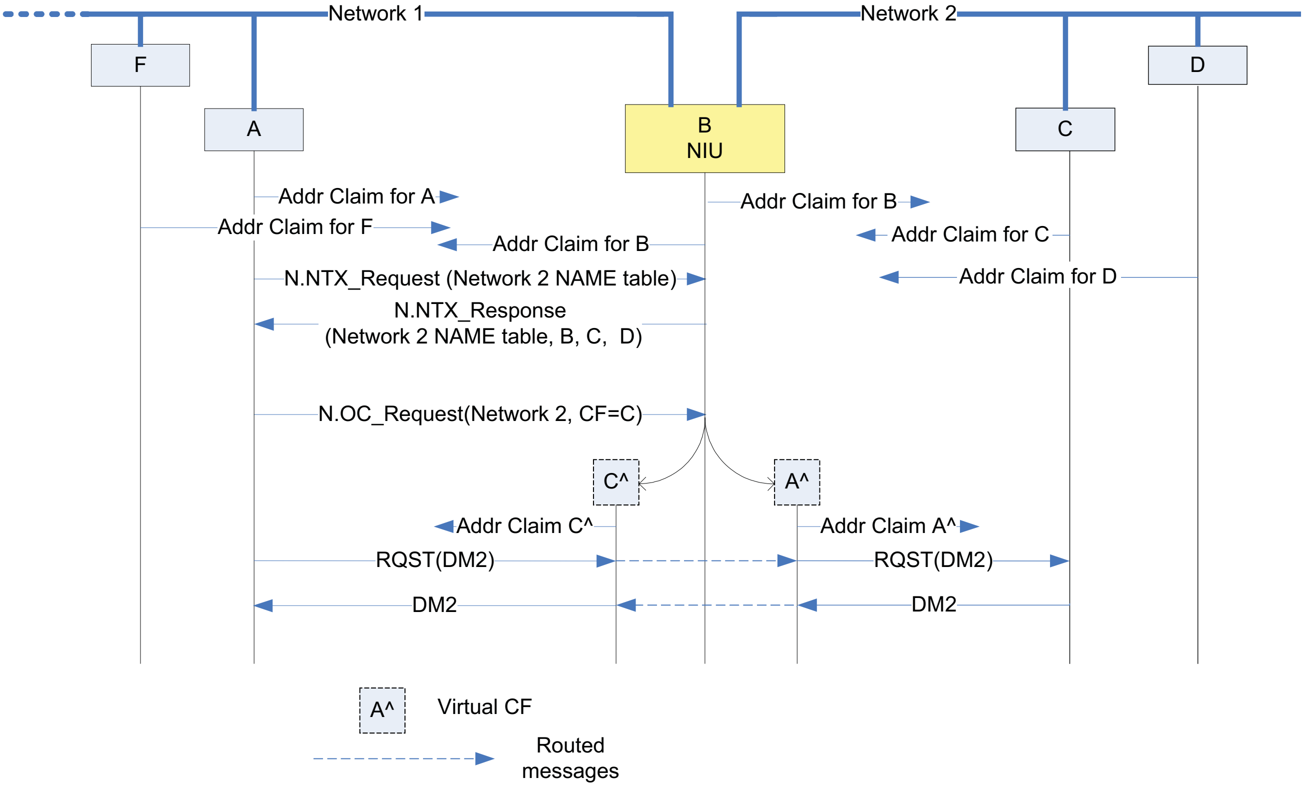


**Figure 2 — Establishing an NIU connection for message transfer**

### 6.9.3 Multiple connections

Figure 3 illustrates CF A on network 1, first connecting to CF C, as shown in Figure 2, and then to a second CF D on network 2. Because A^ already exists on network 2, it is used to make the connection. The address for D^ is then also claimed on network 1. In this case, since the first connection was not closed, A^ is forwarding messages to both C^ and D^.

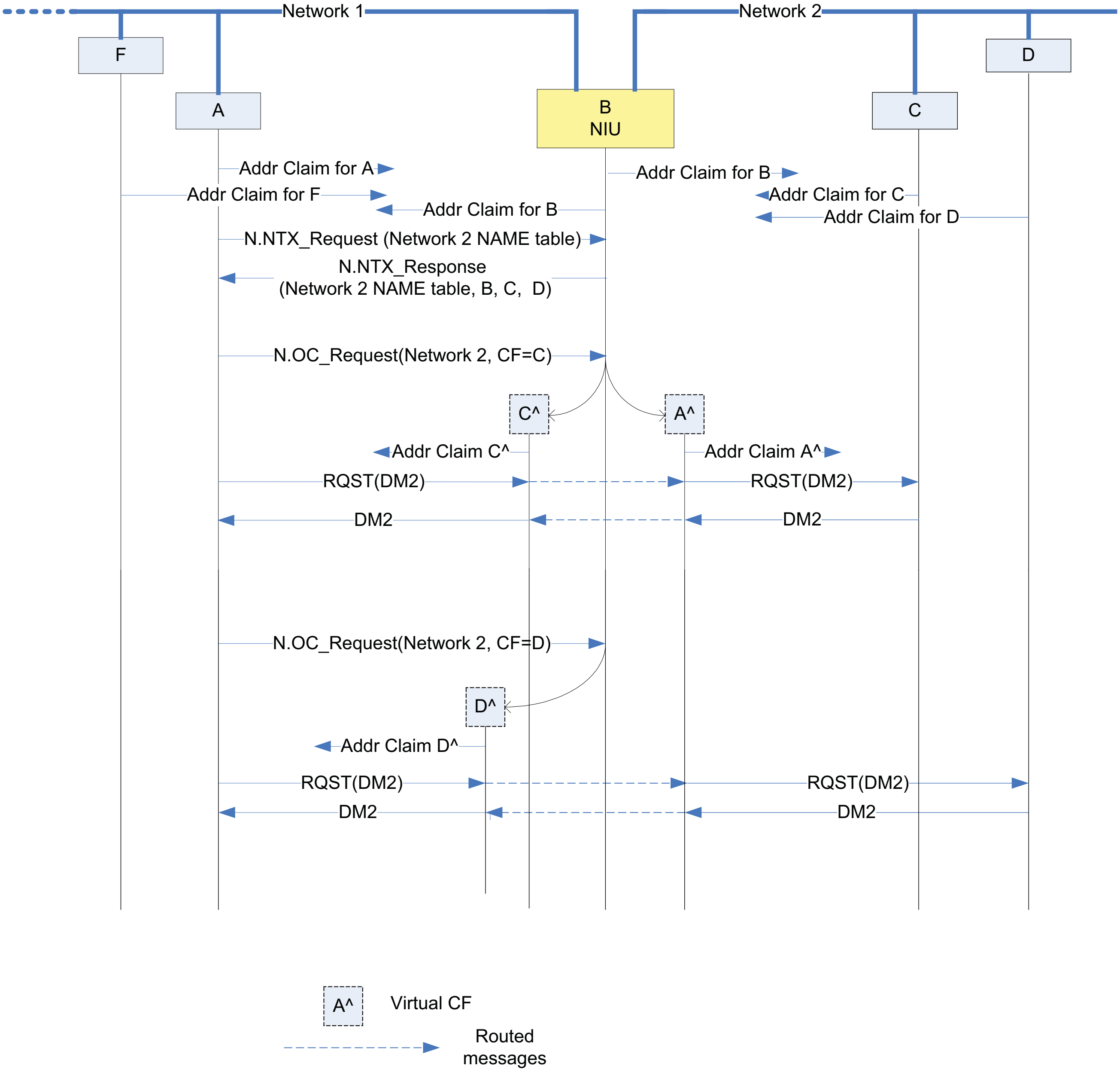


**Figure 3 — Establishing an NIU connection to second CF on network 2**

Another example, shown in Figure 4, is of a second CF F on network 1 connecting to the same CF C on network 2. CF F sends a message to C^ but is acknowledged as defined in ISO 11783-3 with an Access Denied control byte because it does not have a registered open connection with CF C. CF F has to request to open connection with CF C. F^ is then set up on network 2 and C^ on network 1 is reused. CF F can then send a message to C^ which is forwarded to CF C on network 2.

If the address has already been claimed on a network segment, additional claims are not required for the case of opening an additional connection.

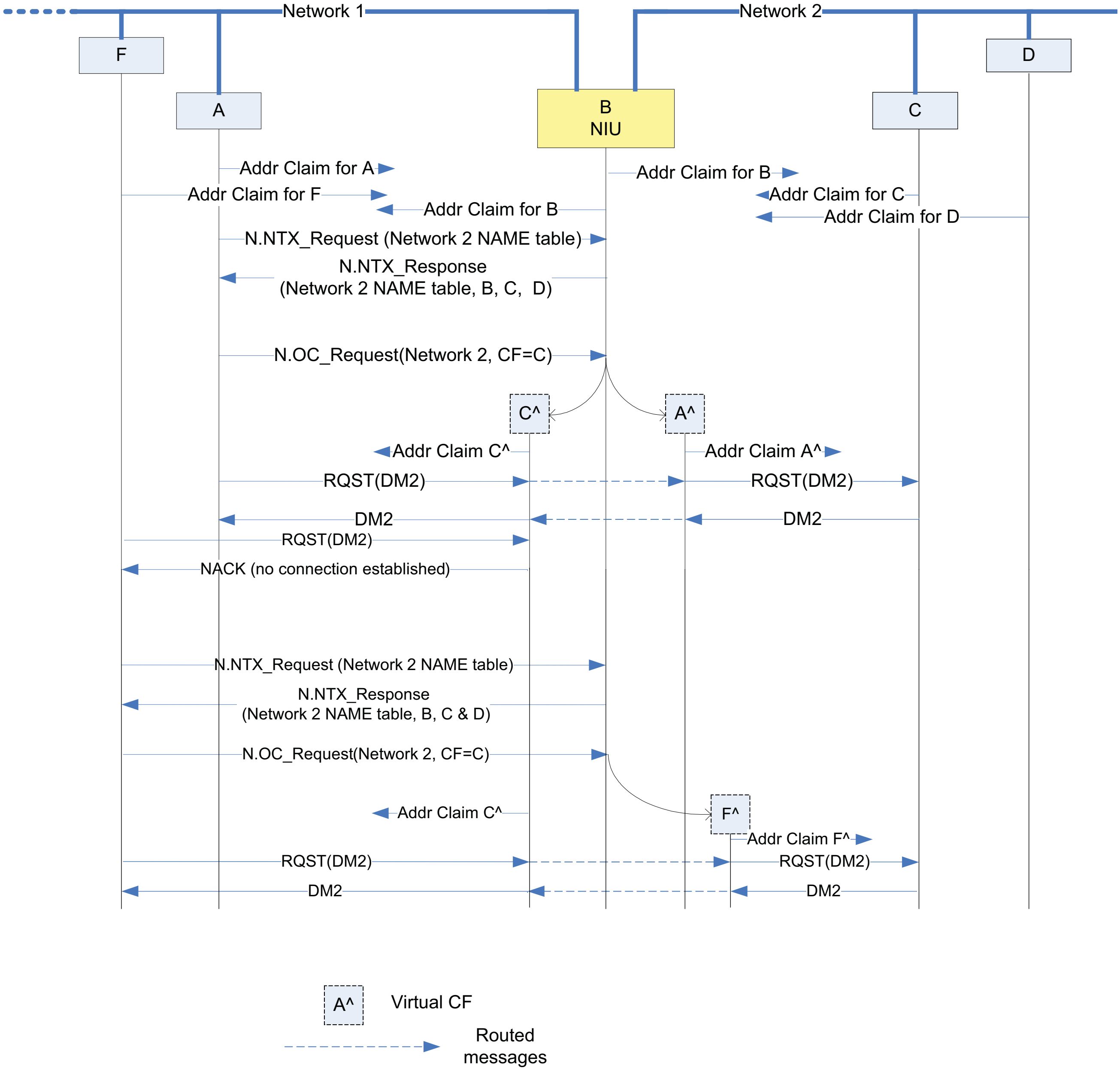


**Figure 4 — Establishing an NIU connection to second CF on network 1**

The procedure in the examples above can be repeated to learn of control functions on ports of another NIU on network segment 2 and so on until the furthest segments of the entire network are found. See the example in Figure 5.

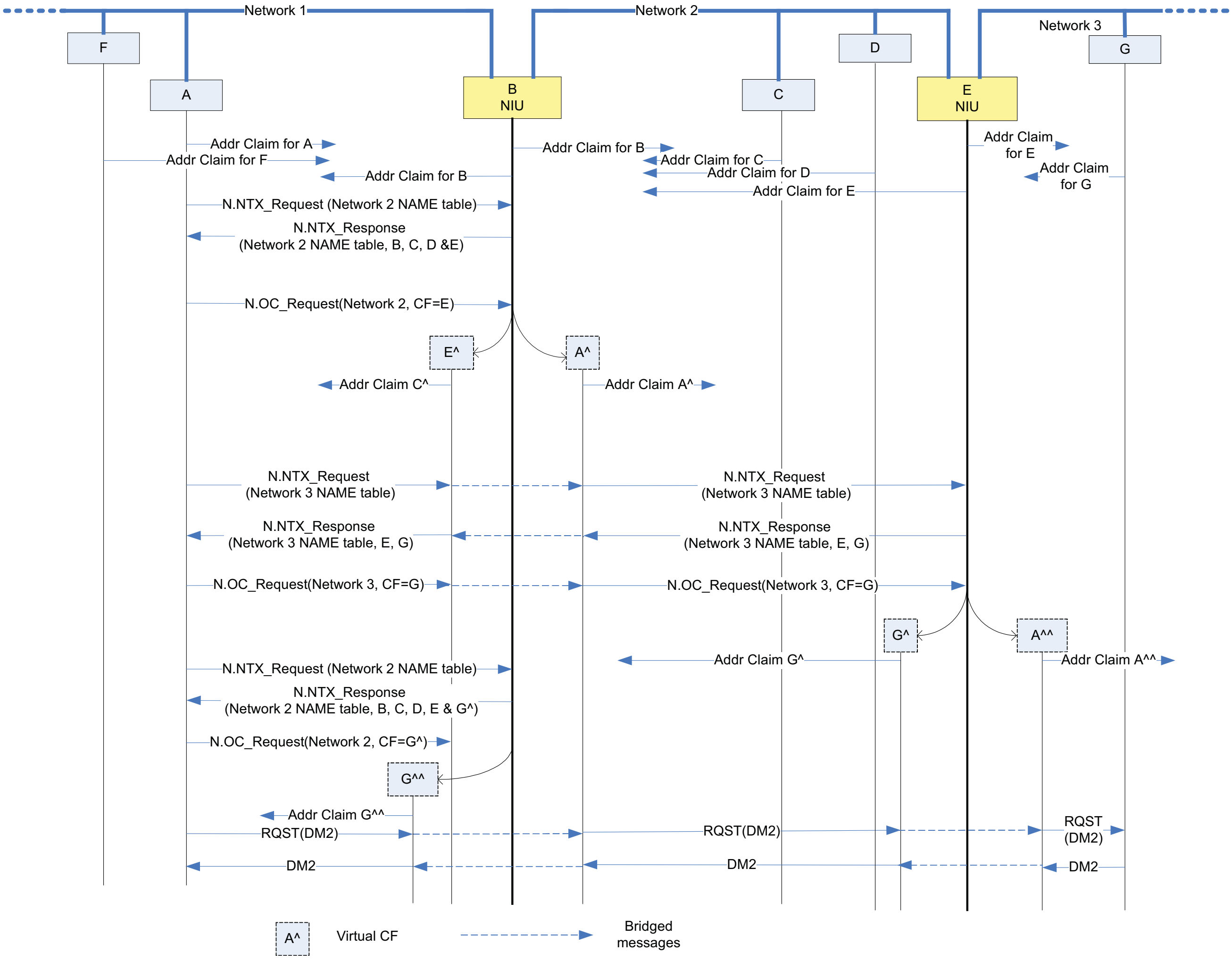


**Figure 5 — Establishing multi-segment NIU connections**

### 6.9.4 Connection requirements

The following requirements shall be completed by an NIU establishing a connection between two network segments.

1) Once connections have been opened between CFs, messages can continue to be sent in this manner between the two CFs for the duration of the current power cycle or until the connection is closed by the original CF that opened it.

2) Messages requiring transport protocol can also be sent. The NIU shall retransmit them with a small enough delay to maintain the transport session timing requirements.

3) Any number of virtual connections between CFs on one network and CFs on other network segments can be established, limited only by the NIU's capability to maintain the connections and the normal limits of 253 total CF addresses per network segment. The self-configurable addresses available are only 128 through 247.

4) Virtual CFs with the same NAME as their original CF can have the same or different source address on the network upon which it appears.

5) If a CF involved in an open connection changes address for any reason, the NIU shall track this address change and update the connection information.

6) If a CF involved in an open connection changes any part of its NAME, it shall first close any of its open connections. If a CF involved in an open connection changes any part of its NAME without first closing any open connections, it can cause unpredictable behaviour of the NIU. A new connection shall be established using the new NAME and address.

7) If a CF that is part of an open connection resends the address claim message for any reason, the NIU shall also send the address claim on the virtual CF's subnetwork with its appropriate address.

8) If a CF that is part of an open connection cannot claim an address and sends CANNOT claim, the NIU shall close the virtual connection and send the CANNOT claim on the virtual CF's subnetwork.

9) There shall only be one virtual representation of a CF on a network segment. It can have associations to multiple connections.

10) Closing an open connection does not remove the virtual CF from the network. A virtual CF stays connected until all connections involving that virtual CF have been closed.

11) When an NIU reports the NAME and address table for another network segment (port pair), it shall also report its own NAME and address on that network segment.

Because virtual CFs appear as any other CF to the other CFs on a network, actual CFs can establish communication with them without having opened a connection with them. The NIU shall NACK messages received by a virtual CF from a CF not currently configured in an open connection for that virtual CF. The Acknowledge message shall contain the address of the virtual CF in the Address Acknowledged (ADD_NACK) parameter. The control byte shall be set to “access denied” (2).

### 6.9.5 Connection messages

#### 6.9.5.1 Request to Open a Connection to another network segment (N.OC_Request)

The NIU request message to establish a connection includes the port pair identity of the destination network segment and the NAME of the desired CF on the destination network segment. This message is from the requesting CF and sent to the NIU. This form of the message uses Transport Protocol.

| | |
|---|---|
| Transmission repetition rate: | As required |
| Data length: | 10 bytes |
| Parameter group number: | 60672 (00ED00$_{16}$), CF to NIU, Destination-Specific |

| | | | | |
|---|---|---|---|---|
| Byte | 1 | Message function = 192$_{10}$ | | Request to Open Connection |
| Byte | 2 | Port pair | | |
| | | Bits 3–0 | “To” port. | The port of the NIU connected to the network segment of the CF targeted for connection. |
| | | Bits 7–4 | “From” port. | The port on the NIU connected to the requester's network segment. This should be the local port, port 0. (See 6.5.3) |
| Bytes | 3–10 | NAME of CF to be connected to on the “to” port. | | |

The requesting CF should use the “request source address and NAME list” (6.7.2.3) function of the network message to obtain the identities of the CFs on the destination network segment and select the desired CF. If a request to open a connection is an exact match with a currently active connection, the response is sent as usual and there is no change to the status of the connection.

### 6.9.5.2 Request to Close a Connection to another network segment (N.CC_Request)

The NIU request message to close a connection includes the port pair identity of the destination network segment and the NAME of the desired CF on the destination network segment. This message is from the requesting CF and sent to the NIU. The requester shall be the same CF that originally opened the connection. This form of the message uses Transport Protocol.

| | |
|---|---|
| Transmission repetition rate: | As required |
| Data length: | 10 bytes |
| Parameter group number: | 60672 (00ED00$_{16}$), CF to NIU, Destination-Specific |

| | | | | |
|---|---|---|---|---|
| Byte | 1 | Message function = 193$_{10}$ | | Request to Close Connection |
| Byte | 2 | Port pair | | |
| | | Bits 3–0 | “To” port. | The port of the NIU connected to the network segment of the CF targeted for connection closure. |
| | | Bits 7–4 | “From” port. | The port on the NIU connected to the requester's network segment. This should be the local port, port 0. (See 6.5.3). |
| Bytes | 3–10 | NAME of CF to be disconnected from remote network. | | |

The requesting CF has to be the same one that opened the connection and is using the same NAME for the CF on the destination network segment that it used to open the connection.

### 6.9.5.3 Response to Open Connection Request (N.OC_Response)

The NIU response message to establish a connection includes the success or failure indication and failure reason if appropriate. This message is from the NIU and sent to the requesting CF. Byte 3 is the success or fail indication and byte 4 is the failure reason indicator.

| | |
|---|---|
| Transmission repetition rate: | As required |
| Data length: | 8 bytes |
| Parameter group number: | 60672 (00ED00$_{16}$), NIU to CF, Destination-Specific |

| | | | | |
|---|---|---|---|---|
| Byte | 1 | Message function = 194$_{10}$ | | Response to Request to open connection |
| Byte | 2 | Port pair | | |
| | | Bits 3–0 | “To” port. | The port of the NIU connected to the requester's network segment. This should be the local port, port 0. |
| | | Bits 7–4 | “From” port. | The port on the NIU connected to the network segment of the CF targeted for connection. |
| Byte | 3 | Status | | |
| | | Bits 1, 2 | Success/Failure | |
| | | 0x01 = success | | |
| | | 0x00 = failure | | |
| | | Bits 3–8 | Reserved | |
| Byte | 4 | Failure Reason Code | | |
| | | 0 | Cannot find CF with NAME | |
| | | 1 | Number of connections to NAME exceeded | |
| | | 2 | Number of connections in NIU exceeded | |
| | | 3 | Busy | |
| | | 4 | Request type not supported | |
| | | 5–254 | Reserved | |
| | | 255 | Not available | |
| Bytes | 5–8 | Reserved (FF$_{16}$) | | |

#### 6.9.5.4 Response to Close Connection Request (N.CC_Response)

The NIU response message to close a connection includes the success or failure indication and failure reason if appropriate. This message is from the NIU and sent to the requesting CF. Byte 3 is the success or fail indication and byte 4 is the failure reason indicator.

| | |
|---|---|
| Transmission repetition rate: | As required |
| Data length: | 8 bytes |
| Parameter group number: | 60672 ($00ED00_{16}$), NIU to CF, Destination-Specific |

| | | | | |
|---|---|---|---|---|
| Byte | 1 | Message function = $195_{10}$ | | Response to Request to close connection |
| Byte | 2 | Port pair | | |
| | | Bits 3–0 | “To” port. | The port of the NIU connected to the requester's network segment. This should be the local port, port 0. |
| | | Bits 7–4 | “From” port. | The port on the NIU connected to the network segment of the CF targeted for connection closure. |
| Byte | 3 | Status | | |
| | | Bits 1,2 | Success/Failure (0x01 equals success, 0x00 equals failure) | |
| | | Bits 3–8 | Reserved | |
| Byte | 4 | Failure Reason Code (see 6.9.5.3) | | |
| Bytes | 5–8 | Reserved ($FF_{16}$) | | |

# 7 Types of network interconnection units

## 7.1 Repeater

The repeater is essentially transparent to any CF on the network.

The repeater forwards messages between bus segments running at the same data rate by regenerating the signal from one segment to another at sub-bit-time intervals and with anti-loopback/lockout. As the repeater is unable to filter the messages, it forwards all of them. Because it does this with only a sub-bit-time delay, the maximum transit delay should be less than 10 % of bit time (400 ns at 250 kilobits/s). This allows bit-wise arbitration to occur correctly across the repeater, while still permitting reasonable propagation delay (cable distance).

If fault isolation is provided, the repeater can disable one or more of its transmitters when a bus fault on a segment is detected.

NOTE 1 No NIU data management function is defined for a repeater.

NOTE 2 Some physical layers cannot support repeaters because of transit delays that affect CAN arbitration.

## 7.2 Bridge

The bridge is essentially transparent to any CF on the network.

The bridge both forwards and filters messages between bus segments. In fulfilling these tasks, it also stores messages. By filtering, it can effectively reduce the amount of bus traffic present on each segment of the network. A bridge can filter any, all, or none of the messages it receives, depending on the application.

There is some transit delay through the bridge, the maximum of which is application dependent. However, the maximum transit delay recommended by this part of ISO 11783 is 10 ms.

## 7.3 Router

### 7.3.1 General

Address claim messages do not cross through a router.

Once operational, the router should be essentially transparent to any CF on the network.

### 7.3.2 Address translation

In addition to forward and filter functions, a router may remap addresses from one port (bus segment) to another port (bus segment).

### 7.3.3 Router management

Although not required, the data management function should be supported in order to provide a standard access to configure the forward, filter and address-translation databases.

In addition to forwarding and filtering, the router performs address remapping, or message routing, from one port, or bus segment, to another, thus enabling a subsystem to appear as a single address to another part of the network. The message filter database is typically configured for pass mode (1), and all messages, except those with a specific entry, are blocked. Because CFs do not then need special knowledge of other individual CFs (addresses) on the system, this may simplify CF development. However, address-claim messages cannot cross through the router, a “look-up” table shall exist to provide the address-translation map, and there is some translation and forwarding delay.

## 7.4 Gateway

### 7.4.1 General

The principal operation performed by a gateway, in addition to those carried out by a router as described in 7.3, is message repackaging. This can simplify the development of CFs, which do not then require specific knowledge of other individual CFs (addresses) on the subsystem.

Once operational, the gateway should be essentially transparent to any CF on the network.

### 7.4.2 Message repackaging

The gateway can take parameters from one or more messages and repackage them into one or more “new” messages. This permits parameters to be grouped for easier transfer, reception and interpretation by another CF.

### 7.4.3 Gateway management

Although not required, the database function should be supported in order to provide standard access for the configuration of the forward, filter, address-translation and message-repackaging databases.

In addition to the tasks performed by the router (7.3), the gateway can repackage parameters from one or more messages into a “new” one. This permits parameters from several CFs to be grouped for easier transfer, reception and interpretation by another CF. As with the router, because CFs do not then need special knowledge of other individual CFs (addresses) on the system, this can simplify CF development. Similarly, the message filter database is typically configured for pass mode (1), and all messages, except those with a specific entry, are blocked. There is some translation, repackaging, and forwarding delay through the gateway, and there needs to be a message-building function containing a database for repackaging.

### 7.5 Tractor ECU

The tractor ECU is a special NIU used to connect the implement bus and the tractor bus located on a tractor or self-propelled implement.

See 5.1.3, and ISO 11783-1 and ISO 11783-9.

# Bibliography

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

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

[3] SAE J 1939, *Recommended Practice for a Serial Control and Communications Vehicle Network*

[4] SAE J 1939-31, *Network Layer*

**ICS 35.240.99; 65.060.01**

Price based on 32 pages

*This page deliberately left blank*

# British Standards Institution (BSI)

BSI is the national body responsible for preparing British Standards and other standards-related publications, information and services.

BSI is incorporated by Royal Charter. British Standards and other standardization products are published by BSI Standards Limited.

## About us

We bring together business, industry, government, consumers, innovators and others to shape their combined experience and expertise into standards -based solutions.

The knowledge embodied in our standards has been carefully assembled in a dependable format and refined through our open consultation process. Organizations of all sizes and across all sectors choose standards to help them achieve their goals.

## Information on standards

We can provide you with the knowledge that your organization needs to succeed. Find out more about British Standards by visiting our website at bsigroup.com/standards or contacting our Customer Services team or Knowledge Centre.

## Buying standards

You can buy and download PDF versions of BSI publications, including British and adopted European and international standards, through our website at bsigroup.com/shop, where hard copies can also be purchased.

If you need international and foreign standards from other Standards Development Organizations, hard copies can be ordered from our Customer Services team.

## Subscriptions

Our range of subscription services are designed to make using standards easier for you. For further information on our subscription products go to bsigroup.com/subscriptions.

With **British Standards Online (BSOL)** you'll have instant access to over 55,000 British and adopted European and international standards from your desktop. It's available 24/7 and is refreshed daily so you'll always be up to date.

You can keep in touch with standards developments and receive substantial discounts on the purchase price of standards, both in single copy and subscription format, by becoming a **BSI Subscribing Member**.

**PLUS** is an updating service exclusive to BSI Subscribing Members. You will automatically receive the latest hard copy of your standards when they're revised or replaced.

To find out more about becoming a BSI Subscribing Member and the benefits of membership, please visit bsigroup.com/shop.

With a **Multi-User Network Licence (MUNL)** you are able to host standards publications on your intranet. Licences can cover as few or as many users as you wish. With updates supplied as soon as they're available, you can be sure your documentation is current. For further information, email bsmusales@bsigroup.com.

## Revisions

Our British Standards and other publications are updated by amendment or revision.

We continually improve the quality of our products and services to benefit your business. If you find an inaccuracy or ambiguity within a British Standard or other BSI publication please inform the Knowledge Centre.

## Copyright



## Useful Contacts:

**Customer Services**
**Tel:** +44 845 086 9001
**Email (orders):** orders@bsigroup.com
**Email (enquiries):** cservices@bsigroup.com

**Subscriptions**
**Tel:** +44 845 086 9001
**Email:** subscriptions@bsigroup.com

**Knowledge Centre**
**Tel:** +44 20 8996 7004
**Email:** knowledgecentre@bsigroup.com

**Copyright & Licensing**
**Tel:** +44 20 8996 7070
**Email:** copyright@bsigroup.com

**BSI Group Headquarters**

389 Chiswick High Road London W4 4AL UK

bsi.

...making excellence a habit.™