



Protocol Interface Manual

## **NetIdent Protocol**

Hilscher Gesellschaft für Systemautomation mbH  
Rheinstraße 15  
D-65795 Hattersheim  
Germany

Tel. +49 (6190) 9907 - 0  
Fax. +49 (6190) 9907 - 50

Sales: +49 (6190) 9907 - 0  
Hotline and Support: +49 (6190) 9907 - 99

Sales Email: [sales@hilscher.com](mailto:sales@hilscher.com)  
Hotline and Support Email: [hotline@hilscher.com](mailto:hotline@hilscher.com)

Web: <http://www.hilscher.com>

---

Index	Date	Version	Chapter	Revision
1	18.03.02	1.000	all	Drawn up

Although this program has been developed with great care and intensively tested, Hilscher Gesellschaft für Systemautomation mbH cannot guarantee the suitability of this program for any purpose not confirmed by us in writing.

Guarantee claims shall be limited to the right to require rectification. Liability for any damages which may have arisen from the use of this program or its documentation shall be limited to cases of intent.

We reserve the right to modify our products and their specifications at any time in as far as this contributes to technical progress. The version of the manual supplied with the program applies.

---

<b>1</b>	<b>INTRODUCTION .....</b>	<b>4</b>
1.1	Terms and Abbreviations .....	4
1.2	Requirements .....	4
<b>2</b>	<b>SPECIFICATION .....</b>	<b>5</b>
2.1	Algorithm .....	5
2.2	Transport Protocol .....	5
2.3	Telegrams .....	5
2.3.1	Format of Telegrams .....	6
2.3.1.1	Magic Cookie- Identifier .....	6
2.3.1.2	Version Number .....	7
2.3.1.3	Op Code – Telegram Type .....	7
2.3.1.4	Transaction Id – Transaction number .....	7
2.3.1.5	Flags – Bit Oriented Field .....	7
2.3.1.6	Error Code .....	7
2.3.1.7	Master IP Address – IP Address of the Protocol Master .....	7
2.3.1.8	Master Port Number – Port Number of Protocol Master .....	7
2.3.1.9	IP Address – IP Address of Device .....	8
2.3.1.10	Ethernet Address- Ethernet Address of the Device .....	8
2.3.1.11	Serial Number – Serial Number of Device .....	8
2.3.1.12	Device Type – Device Type of Device .....	8
2.3.1.13	Device Name – Device Name of Device .....	8
2.3.2	Telegram types .....	8
2.3.2.1	Identify Request .....	8
2.3.2.2	Identify Reply .....	9
2.3.2.3	Set IP Address Request .....	9
2.3.2.4	Set IP Address Reply .....	9
<b>3</b>	<b>RESTRICTIONS .....</b>	<b>10</b>

# 1 Introduction

The configuration of devices with Ethernet interfaces over TCP/IP requires a device with a preconfigured IP address. Also the user of the configuration tool has to know the configured IP address to get a connection to the device with the tool.

Every device needs an alternative way to get configured, if the device has no actual IP address or the user doesn't know the IP address.

This document describes an algorithm and the underlying protocol to get able to identify Hilscher devices in the local Ethernet without any knowledge of the configured IP address. It is also possible to assign a new IP address to the device with this protocol.

## 1.1 Terms and Abbreviations

IP	Internet Protocol
TCP	Transmission Control Protocol
UDP	User Datagram Protocol

## 1.2 Requirements

This description is restricted to Hilscher devices with Ethernet interfaces and running TCP/UDP IP stack. The TCP/UDP IP stack needs a NetIdent extension. Devices with older TCP/UDP IP stack do not support this algorithm.

---

## **2 Specification**

### **2.1 Algorithm**

The identification algorithm is based on the Master-Slave mechanism. A host in the network, e.g. a PC with a running configuration tool, is the active part and sends commands or requests to the network. The connected Ethernet devices are answering to this requests.

With different services defined on the NetIdent protocol it is possible to activate predefined actions on the slaves.

### **2.2 Transport Protocol**

The NetIdent protocol uses UDP for communication between master and slave. The transfer of the telegrams is done via Unicast, if the host wants to address one device and its IP address is well known and is valid in the local subnet. Otherwise it is done via limited broadcasts (IP address 255.255.255.255). Through this procedure the generated network load is minimized, and it is reached, that an device could be reached also then, if it possesses no IP address or no for the local subnet valid IP address.

With an additional addressing mechanism within the telegram, the service is activated on the right device, also if the command was sent as broadcast. The protocol master uses UDP port 25384 for communication. Slave devices are waiting for incoming commands on UDP port 25385.

### **2.3 Telegrams**

The NetIdent protocol uses telegrams with a fixed structure. There are no optional fields or fields with a variable length within the telegram. Extensions are possible by adding further fields to the existing telegram.

### 2.3.1 Format of Telegrams

Every telegram in NetIdent protocol contains fields according to the following structure written in C. All data is transferred in network byte order.

```
typedef struct {
    unsigned long    ulMagicCookie;           /* Magic Cookie      */
    unsigned long    ulVersion;               /* Version Number    */
    unsigned long    ulOpCode;                /* Op Code           */
    unsigned long    ulXid;                   /* Transaction Id    */
    unsigned long    ulFlags;                 /* Flags             */
    unsigned long    ulErrorCode;              /* Error code        */
    unsigned long    ulMasterIpAddress;        /* Master IP address */
    unsigned long    ulMasterPortNumber;       /* Master port number */
    unsigned long    ulIpAddress;              /* IP Address        */
    unsigned char    abEthernetAddress[6];     /* Ethernet Address  */
    unsigned long    ulSerialNumber;           /* Serial Number     */
    unsigned long    ulDeviceType;             /* Device type       */
    char             szDeviceName[64];         /* Name of device    */
}
```

Name	Length (Bytes)	Value	Description
Magic Cookie	4	"HINI"	Identifier
Version Number	4	1000	Version 1.000
Op Code	4	1 2 3 4	Telegram type Identify Request Identify Reply Set IP Address Request Set IP Address Reply
Transaction Id	4		Transaction Number
Flags	4		Bit oriented field, defining which fields are valid
Error Code	4		Error Code
Master IP Address	4		IP address of NetIdent protocol master
Master Port Number	4		Port number for responses to NetIdent protocol master
IP Address	4		IP address of the device
Ethernet Address	6		Ethernet address of the device
Serial Number	4		Serial number of device
Device Type	4		Type of device
Device Name	64		Null terminated string containing name of the device

#### 2.3.1.1 Magic Cookie- Identifier

The fixed Identifier "HINI" (**H**ilscher **N**etIdent) is required in every telegram to identify a valid NetIdent telegram.

### 2.3.1.2 Version Number

The version number is used to characterize the format of the telegram. The binary version number is interpreted as a fixed point value. The value 1000 is interpreted as 1.000.

### 2.3.1.3 Op Code – Telegram Type

The Op Code field contains information about the command type. Further details are explained in Chapter 2.3.2.

### 2.3.1.4 Transaction Id – Transaction number

The transaction number is used to identify an reply telegram unambiguous. All replies are sent with the same transaction number as the corresponding request.

### 2.3.1.5 Flags – Bit Oriented Field

Every single Bit in this field gives information which field in the telegram is valid. The following flags are defined:

Flag	Value	Meaning
NID_FLAG_ERROR_CODE	00000080h	The field ulErrorCode is valid
NID_FLAG_MASTER_IP_ADDRESS	00000040h	The field ulMasterIpAddress is valid
NID_FLAG_MASTER_PORT_NUMBER	00000020h	The field ulMasterPortNumber is valid
NID_FLAG_IP_ADDRESS	00000010h	The field ulIpAddress is valid
NID_FLAG_ETHERNET_ADDRESS	00000008h	The field abEthernetAddress is valid
NID_FLAG_SERIAL_NUMBER	00000004h	The field ulSerialNumber is valid
NID_FLAG_DEVICE_TYPE	00000002h	The field ulDeviceType is valid
NID_FLAG_DEVICE_NAME	00000001h	The field szDeviceName is valid

### 2.3.1.6 Error Code

This field contains information about the result of the requested service. The master has to analyze this field, if the flag NID\_FLAG\_ERROR\_CODE is set. This field has no consequence in request telegrams.

### 2.3.1.7 Master IP Address – IP Address of the Protocol Master

In requests the protocol master has to insert his IP address in this field, and has to mark the field valid with the corresponding flag. If the field is marked valid, responding slaves will send responses via unicast to this address. Otherwise they will send the answer as a limited broadcast to the IP address 255.255.255.255.

In responses the slave has to insert the Master IP address of the request, that is regardless of whether the request was sent via unicast or broadcast.

### 2.3.1.8 Master Port Number – Port Number of Protocol Master

In request telegrams the protocol master fills in the port number on which the master is waiting for replies and marks the field valid. Slaves are sending their answer to this port number, if the field is marked valid. Otherwise slaves have to send their responses to port number 25384.

---

### **2.3.1.9 IP Address – IP Address of Device**

This field has only meaning in the Set IP Address request. The protocol master inserts the new IP address for the slave in this field and marks it valid in the flags. In all other requests the master has to mark this field invalid and the slave has to ignore this field. In a response the slave has to insert its actual IP address.

### **2.3.1.10 Ethernet Address- Ethernet Address of the Device**

Slaves have to insert their Ethernet address in this field and have to mark it valid in the flags. In requests this field is not valid and slaves have to ignore this data.

### **2.3.1.11 Serial Number – Serial Number of Device**

In request telegrams the slave has to evaluate this field as additional address information, if the field is marked valid. Only if the serial number in the request corresponds with the serial number of the device, the device will treat the request. All other slaves have to ignore the telegram. In response telegrams the slaves inserts its serial number.

### **2.3.1.12 Device Type – Device Type of Device**

In request telegrams the slave has to evaluate this field as additional address information, if the field is marked valid. Only if the device type in the request corresponds with the device type of the device, the device will treat the request. All other slaves have to ignore the telegram. In response telegrams the slaves inserts its device type

### **2.3.1.13 Device Name – Device Name of Device.**

This field has no meaning in request telegrams. In response telegrams the slave sends its device name as null terminated ASCII string and marks the field as valid. The device name has only additional information character.

## **2.3.2 Telegram types**

### **2.3.2.1 Identify Request**

The protocol master asks one or more devices to answer with an Identify Reply telegram. If the fields `ulDeviceType` or `ulSerialNumber` are valid, the slaves have to use fields as additional addressing information. This means, only devices with the given device type or serial number have to answer to this request. The protocol master has to fill the fields `ulMasterIpAddress` and `ulMasterPortNumber`.

To reduce network traffic the master should send this request via Unicast to the device directly. If the master wants to reach more than one or unknown devices it has to send the request via Limited Broadcast to the address (255.255.255.255).



### 2.3.2.2 Identify Reply

As soon as a device receives an Identify Request telegram, it has to answer with an Identify Reply telegram. The slave has to set the fields `ulDeviceType`, `ulSerialNumber`, `abEthernetAddress`, `ulIpAddress`, `ulMasterIpAddress` and `ulMasterPortNumber` with valid values, if possible. The reply has to be sent via unicast to the master IP address and to the master port number, if the slave is possible to do this. If one information is missing the slave has to use standard values (IP address 255.255.255.255, Port 25384) instead.

### 2.3.2.3 Set IP Address Request

The protocol master asks a device to change its IP address to the address given in the telegram parameter `ulIpAddress`. This field has to be marked valid. The master has to set valid values to the parameters `ulMasterIpAddress` and `ulMasterPortNumber`. The requested slave only proceeds, if the fields `ulDeviceType` and `ulSerialNumber` are valid. In this case the slave has to change its IP address and has to send a Set IP Address Reply to the master. Otherwise the slave has to ignore the telegram.

If the slave is configured with a valid IP address, the slave will only accept the Set Ip Address Request within the first 3 minutes after power on or reset. This is done to prevent unintentional setting of the IP address while the device is in normal operating mode. After that time the slave will answer with an error.

### 2.3.2.4 Set IP Address Reply

If a device has received a valid Set IP Address Request, it has to send a Set IP Address Reply after setting the new IP address. In the response the parameters `ulDeviceType`, `szDeviceName`, `ulSerialNumber`, `abEthernetAddress`, `ulIpAddress`, `ulMasterIpAddress` and `ulMasterPortNumber` has to be set to valid values and the corresponding flags have to be marked valid. The response is send via Unicast to the specified master IP address and master port number of the request. If this is not possible, because one or more information is missing, it is sent to the standard values (IP address 255.255.255.255, Port 25384).

---

### **3 Restrictions**

The described procedure is only for devices which are able to answer to the requests from a NetIdent protocol master, so only for devices from Hilscher GmbH with the NetIdent extension. Because of the use of Limited Broadcasts a protocol master can reach only devices within the local Ethernet. Connected routers do not route this requests.