
	<p>Document</p> <p>Technical Specification</p>	<p>version: vo.o8</p> <p>status:</p> <p>date: 2013-10-10</p>
<p>Reader-Host-Protocol – PUR-Extension</p>	<p>org</p>	

READER-HOST-PROTOCOL PUR - EXTENSION


<p>Reader-Host-Protocol-PUR-Extensions_o.o8_EN.docx</p>	<p>R&D iDTRONIC GmbH</p>	<p>page: 1 von 22</p>
---	------------------------------	-----------------------

	<div>Document</div> <div>Technical Specification</div>	<div>version: vo.o8</div> <div>status:</div> <div>date: 2013-10-10</div>
Reader-Host-Protocol – PUR-Extension		org

History of Change:


10.03.2010	R&D	Created document	v0.01
21.06.2011	R&D	Added Lock-Tag codes	v0.02
08.12.2011	R&D	Added Gen2-EPC-Size, Added description for Read-From-Tag, Added notification values	v0.03
12.12.2012	R&D	Added Cyclic-Inventory Start Bytes Added Gen2-Send-Handle Added Gen2-Send-PC	v0.04
09.01.2013	R&D	Added Commands: <ul style="list-style-type: none"> • Get-Handle-From-Tag • Read-From-Handle • Write-To-Handle 	v0.05
15.01.2013	R&D	Added new Inventory Mode Added new Tag-Handler	v0.06
02.05.2013	R&D	Added Gen2 Settings: <ul style="list-style-type: none"> • Gen2-Q-Setting • Gen2-Q-Method • Gen2-Session • Gen2-InventoryCount 	v0.07
10.10.2013	R&D	Added Custom-Tag-Command	v0.08

<p>Reader-Host-Protocol-PUR-Extensions_o.o8_EN.docx</p>	<p align="center">R&D iDTRONIC GmbH</p>	<p align="right">page: 2 von 22</p>
---	--	--


	<div>Document</div> <div>Technical Specification</div>	<div>version: vo.o8</div> <div>status:</div> <div>date: 2013-10-10</div>
Reader-Host-Protocol – PUR-Extension		org

1	Introduction.....	5
2	Parameter–Dictionary.....	6
2.1	Inventory Mode.....	7
2.2	Power Safe Setting	8
2.3	RSSI	8
2.4	Tag-Id-Behavior	9
2.5	Gen2-Link-Frequency	9
2.6	Gen2-Bit-Encoding.....	10
2.7	Gen2-Modulation Depth	10
2.8	Gen2-EPC-Size	11
2.9	Gen2-Send-Handle	11
2.10	Gen2-Send-PC.....	11
2.11	Gen2-Q-Setting.....	12
2.12	Gen2-Q-Method	12
2.13	Gen2-Session.....	12
2.14	Gen2-InventoryCount.....	13
3	Function Descriptions	14
3.1	Read-From-Tag (50-03)	14
3.2	Lock-Tag (50-05)	14
4	Custom–Tag–Commands	15
4.1	NXP-Set-ReadProtect (01)	15
4.2	NXP-Clear-ReadProtect (02)	16
4.3	Get-Handle-From-Tag (04)	17
4.4	Read-From-Handle (05).....	18
4.5	Write-To-Handle (06)	19
4.6	Custom-tag-Command-Handle (07)	20
5	Cyclic–Inventory Start Bytes.....	21

<p>Reader-Host-Protocol-PUR-Extensions_o.o8_EN.docx</p>	<p align="center">R&D iDTRONIC GmbH</p>	<p align="right">page: 3 von 22</p>
---	--	--

	<div>Document</div> <div>Technical Specification</div>	<div>version: vo.o8</div> <div>status:</div> <div>date: 2013-10-10</div>
Reader-Host-Protocol – PUR-Extension		org


6 Notifications 22

	<div>Document</div> <div>Technical Specification</div>	<div>version: vo.o8</div> <div>status:</div> <div>date: 2013-10-10</div>
Reader-Host-Protocol – PUR-Extension		org

1 Introduction

This document describes the extensions to the standard UHF-RFID Reader-Host-Protocol for the PUR.


<p>Reader-Host-Protocol-PUR-Extensions_o.o8_EN.docx</p>	<p align="center">R&D iDTRONIC GmbH</p>	<p align="right">page: 5 von 22</p>
---	--	--

	Document Technical Specification	version: vo.o8 status: date: 2013-10-10
Reader-Host-Protocol – PUR-Extension		org

2 Parameter-Dictionary

The Parameters that can be set and read with the commands Get-Param and Set-Param are shown in the following table:

Name	Address	Size	Description	Version
Inventory Mode	0x0000	1 Byte	The used inventory mode	
Power Safe Settings	0x0001	3 Bytes	The settings for the power safe	
RSSI	0x0002	1 Byte	Defines if the RSSI Value is sent to the host	
Tag-Id-Behavior-Mode	0x0003	1 Byte	Defines how the reader should behave if a tag is detected.	
Gen2-Link-Frequency	0x0020	1 Byte	The used link frequency	
Gen2-Bit-Encoding	0x0021	1 Byte	The used bit encoding	
Gen2-Modulation-Depth	0x0022	1 Byte	The used modulation depth	
Gen2-EPC-Size	0x0023	1 Byte	The expected EPC size	V1.07
Gen2-Send-Handle	0x0024	1 Byte	Send Handle with EPC	V1.17
Gen2-Send-PC	0x0025	1 Byte	Send PC with EPC	V1.17

	<div>Document</div> <div>Technical Specification</div>	<div>version: vo.o8</div> <div>status:</div> <div>date: 2013-10-10</div>
Reader-Host-Protocol – PUR-Extension		org

2.1 Inventory Mode

The set inventory mode defines how the reader does an inventory. This setting affects for example the used slot count and the used anti collision procedure.

Data structure:

1 byte Inventory-Mode-Enum


Possible settings:

Name	Value	Description
Gen2 – Fast Multi Tag	0x00	Inventory Mode that does not take the tag to the Opened but to the Acknowledged State. This inventory mode is not as secure as the standard mode, but is faster.
Gen2 – Fast Single Tag	0x01	The same inventory mode like the Fast Multi Tag, but with the slot count of 1. This has the effect that no anti-collision procedure is performed, but if there is only one tag in the field, it is detected much more faster
Gen2 – Standard Inventory	0x02	Inventory mode like defined in the standard
Gen2 – NXP-ReadProtect Inventory	0x03	Inventory that only searches for read protected tags. The EPC of these tags will always be composed of zeros.
Gen 2 – Standard Single Tag	0x04	The same inventory mode like the Standard Multi Tag, but with the slot count of 1. This has the effect that no anti-collision procedure is performed, but if there is only one tag in the field, it is detected much more faster.

Default:

Inventory-Mode-Enum **0x02** Gen2 – Standard Inventory

<p>Reader-Host-Protocol-PUR-Extensions_o.o8_EN.docx</p>	<p align="center">R&D iDTRONIC GmbH</p>	<p align="right">page: 7 von 22</p>
---	--	--

	<div>Document</div> <div>Technical Specification</div>	<div>version: vo.o8</div> <div>status:</div> <div>date: 2013-10-10</div>
Reader-Host-Protocol – PUR-Extension		org

2.2 Power Safe Setting

The power safe setting can be used to pulse the reader and so save power. If turned on, the reader switches the field on and performs an inventory after which he switches off the field and sleeps for the specified time. After this time this process is restarted.

Data Structure:

1 byte	Switch (ON (0x01) / OFF (0x00))
2 byte	Sleep Time in milliseconds

Default:

Switch	0x00	OFF
Sleep Time	0x00FA	250 ms

2.3 RSSI


If the RSSI Setting is enabled, the RSSI Value of the detected tags is appended to every Inventory-Cyclic-Interrupt.

Data Structure:

1 byte	Switch (ON (0x01) / OFF (0x00))
--------	---------------------------------

Default:

Switch	0x00	OFF
--------	------	-----

	<div>Document</div> <div>Technical Specification</div>	<div>version: vo.o8</div> <div>status:</div> <div>date: 2013-10-10</div>
Reader-Host-Protocol – PUR-Extension		org

2.4 Tag-Id-Behavior

With this option the reader be configured how, it should react if it detects a tag.

Data structure:

1 byte Tag-Id-Behavior-Enum

Possible settings:

Name	Value	Description
Send-Tag-Id-Immediately	0x00	Every time the reader detects a tag, it is immediately forwarded to the host.
Send-Tag-Id-Once	0x01	If the reader detects a tag, it forwards the tag-Id to the host and stores it into a temporary buffer. If the same tag is detected again in the same inventory round, it is no more forwarded to the host. The buffer is cleared at the start of each cyclic inventory.
Send-Tag-Id-Immediately And Stop	0x02	The same Tag-Id behavior like Send-Tag-Id-Immediately but when the first tag is detected, the Cyclic-Inventory is stopped.

Default:

Tag-Id-Behavior-Enum **0x00** Send-Tag-Id-Immediately

2.5 Gen2-Link-Frequency

This option sets the used Link Frequency for the Gen2 protocol.

Data Structure:

1 byte Link-Frequency


Possible settings:

Value	Description
0x00	40 kHz
0x01	80 kHz
0x02	160 kHz
0x03	213 kHz
0x04	256 kHz
0x05	320 kHz

Default:

Link-Frequency **0x02** 160 kHz

<p>Reader-Host-Protocol-PUR-Extensions_o.o8_EN.docx</p>	<p align="center">R&D iDTRONIC GmbH</p>	<p align="right">page: 9 von 22</p>
---	--	--

	<div>Document</div> <div>Technical Specification</div>	<div>version: vo.o8</div> <div>status:</div> <div>date: 2013-10-10</div>
Reader-Host-Protocol – PUR-Extension		org

2.6 Gen2-Bit-Encoding

This option sets the used Bit-Encoding for the Gen2 protocol.

Data Structure:

1 byte Bit-Encoding

Possible settings:

Value	Description
0x00	FM0
0x01	Miller 2
0x02	Miller 4
0x03	Miller 8

Default:

Bit-Encoding **0x01** Miller 2

2.7 Gen2-Modulation Depth

This option sets the used Modulation Depth for the Gen2 protocol.


Data Structure:

1 byte Modulation-Depth in %

Default:

Modulation-Depth **0x64** 100%

<p>Reader-Host-Protocol-PUR-Extensions_o.o8_EN.docx</p>	<p align="center">R&D iDTRONIC GmbH</p>	<p align="right">page: 10 von 22</p>
---	--	---

	<div>Document</div> <div>Technical Specification</div>	<div>version: vo.o8</div> <div>status:</div> <div>date: 2013-10-10</div>
Reader-Host-Protocol – PUR-Extension		org

2.8 Gen2-EPC-Size

This option sets the expected EPC size of the tags that should be scanned. If the EPC size is set to a constant value, only tags with this EPC size are detected. If the EPC size is set to 0, all tags with different sizes are detected. For constant sizes only multiples of 2 are allowed.

Data Structure:

1 byte EPC Size

Possible settings:

Value	Description
0	Dynamic
2	2 Byte EPC
4	4 Byte EPC
...	...
0x01	12 Byte EPC
...	...
18	18 Byte EPC

Default:

Bit-Encoding **12** Constant 12 Bytes Size

2.9 Gen2-Send-Handle

If the Send-Handle is enabled, the Handle Value of a detected tag is appended to every Inventory-Cyclic-Interrupt.

Data Structure:

1 byte Switch (ON (0x01) / OFF (0x00))

Default:

Switch 0x00 OFF

2.10 Gen2-Send-PC

If the Send-PC is enabled, the PC Value of a detected tag is appended to every Inventory-Cyclic-Interrupt.


Data Structure:

1 byte Switch (ON (0x01) / OFF (0x00))

Default:

Switch 0x00 OFF

<p>Reader-Host-Protocol-PUR-Extensions_o.o8_EN.docx</p>	<p align="center">R&D iDTRONIC GmbH</p>	<p align="right">page: 11 von 22</p>
---	--	---

	<p align="center">Document</p> <p align="center">Technical Specification</p>	<p>version: vo.o8</p> <p>status:</p> <p>date: 2013-10-10</p>
Reader-Host-Protocol – PUR-Extension		org

2.11 Gen2-Q-Setting

This field specifies the used Q value for Gen2 inventory. The Q defines how much slots are used for an inventory. The slot count is 2Q.

Data Structure:

1 byte	Initial Q
1 byte	Minimal Q
1 byte	Maximal Q

Default:

Initial Q	4	Slot count: 16
Minimal Q	2	Slot count: 4
Maximal Q	15	Slot count: 32768

2.12 Gen2-Q-Method

This field specifies the method which is used to adjust the Q value between different inventories.

Data Structure:

1 byte	Method ID
--------	-----------

Possible settings:

Value	Description
0x00	Static (Q stays at initial value)
0x01	RFE Dynamic Adjust Method

Default:

Method ID	0x01	RFE Dynamic Adjust Method
-----------	------	---------------------------

2.13 Gen2-Session

This field specifies the method used session for inventories.

Data Structure:

1 byte	Session
--------	---------


Possible settings:

Value	Description
0x00	Session 0
0x01	Session 1
0x02	Session 2
0x03	Session 3

Default:

Session	0x01	Session 1
---------	------	-----------

Reader-Host-Protocol-PUR-Extensions_o.o8_EN.docx	R&D iDTRONIC GmbH	page: 12 von 22
--	------------------------------	------------------------

	<div>Document</div> <div>Technical Specification</div>	<div>version: vo.o8</div> <div>status:</div> <div>date: 2013-10-10</div>
Reader-Host-Protocol – PUR-Extension		org

2.14 Gen2-InventoryCount


This field specifies how much inventories are executed until a new “Select” command is issued.

Data Structure:

1 byte Count

Default:

CountQ 0x01 1 inventory per Select cmd.

	<div>Document</div> <div>Technical Specification</div>	<div>version: vo.o8</div> <div>status:</div> <div>date: 2013-10-10</div>
Reader-Host-Protocol – PUR-Extension		org

3 Function Descriptions

3.1 Read-From-Tag (50-03)

If the byte size 0 is selected, the reader reads until the tag responses with a “Memory Overrun” error. But with using tags with bigger User Memory banks, the reader uses another stop condition. It reads at a maximum of 200 bytes, and then returns the 200 bytes. So if bigger amounts of data should be read and the maximum size is not known, the Read-From-Tag function should be called with incremented addresses as long as the read byte count is lower than 200 bytes or a TMI_MEM_OVERRUN error is returned.

3.2 Lock-Tag (50-05)


The lock tag command needs to types of codes to lock a tag. For a Gen2 tag these codes are:

```
typedef enum{
    UNLOCK                = 0x00,
    LOCK                  = 0x01,
    PERMALOCK             = 0x02,
    LOCK_AND_PERMALOCK    = 0x03,
} eRFE_LOCK_MODE;

typedef enum{
    KILL_PASSWORD         = 0x00,
    ACCESS_PASSWORD       = 0x01,
    EPC                   = 0x02,
    TID                   = 0x03,
    USER                  = 0x04,
} eRFE_LOCK_MEMORY_SPACE;
```

These codes are directly connected to the Gen2 standard. Further information about these codes and there meaning can be found there.

<p>Reader-Host-Protocol-PUR-Extensions_o.o8_EN.docx</p>	<p align="center">R&D iDTRONIC GmbH</p>	<p align="right">page: 14 von 22</p>
---	--	---

	<p align="center">Document</p> <p align="center">Technical Specification</p>	<p>version: vo.o8</p> <p>status:</p> <p>date: 2013-10-10</p>
<p>Reader-Host-Protocol – PUR-Extension</p>	<p>org</p>	

4 Custom-Tag-Commands

In this section the available custom tag commands are documented. These commands can be used by calling the command 05-10 of the Reader-Host-Protocol.

The available commands are:

Command	Value
NXP-Set-ReadProtect	0x01
NXP-Clear-ReadProtect	0x02
Get-Handle-From-Tag	0x04
Read-From-Handle	0x05
Write-To-Handle	0x06

4.1 NXP-Set-ReadProtect (01)

This function transfers a NXP tag into the ReadProtect Mode. In this mode the tag only returns zeroes instead of its actual EPC.

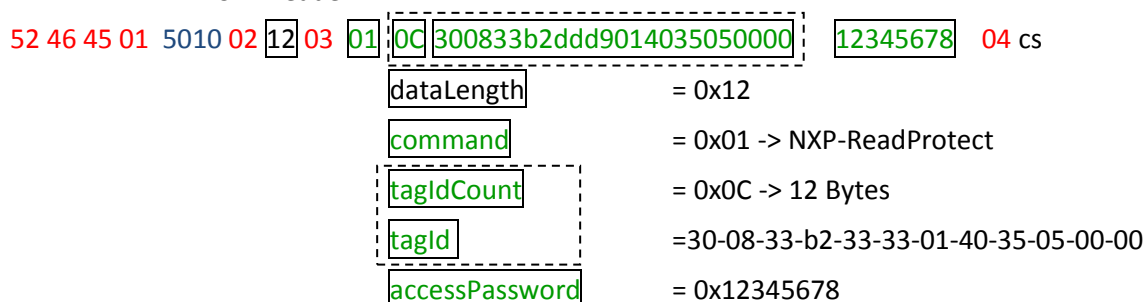
Parameters: unsigned char **command**, unsigned char **tagIdCount**,
unsigned char **tagId**[tagIdCount], unsigned long **accessPassword**,

Return Values: RFE_RET_VALUE **status**

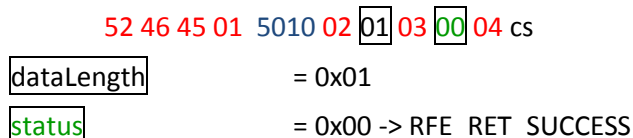
Status Values: RFE_RET_SUCCESS, RFE_RET_ERR_ON_EXEC_OP, RFE_RET_ERR_COULD_NOT_WRITE,
RFE_RET_ERR_WRONG_PARAM_COUNT, RFE_RET_ERR_WRONG_PARAM, Every TMI
Return Code


Example: Set the tag 30-08-33-b2-dd-d9-01-40-35-05-00-00 to ReadProtect:

PC -> Reader



Reader -> PC



	<p align="center">Document</p> <p align="center">Technical Specification</p>	<p>version: vo.o8</p> <p>status:</p> <p>date: 2013-10-10</p>
<p>Reader-Host-Protocol – PUR-Extension</p>	<p>org</p>	

4.2 NXP-Clear-ReadProtect (02)

This function retrieves a NXP tag from the ReadProtect Mode.

Parameters: unsigned char **command**, unsigned char **tagIdCount**,
unsigned char **tagId**[tagIdCount], unsigned long **accessPassword**,

Return Values: RFE_RET_VALUE **status**

Status Values: RFE_RET_SUCCESS, RFE_RET_ERR_ON_EXEC_OP, RFE_RET_ERR_COULD_NOT_WRITE,
RFE_RET_ERR_WRONG_PARAM_COUNT, RFE_RET_ERR_WRONG_PARAM, Every TMI
Return Code

Example: Clear the ReadProtect of the tag 00-00-00-00-00-00-00-00-00-00-00-00:

PC -> Reader


52 46 45 01 5010 02 12 03 02 0C 00000000000000000000000000000000 12345678 04 cs

dataLength	= 0x12
command	= 0x02 -> NXP-Clear-ReadProtect
tagIdCount	= 0x0C -> 12 Bytes
tagId	= 00-00-00-00-00-00-00-00-00-00-00-00
accessPassword	= 0x12345678

Reader -> PC

52 46 45 01 5010 02 01 03 00 04 cs

dataLength	= 0x01
status	= 0x00 -> RFE_RET_SUCCESS

	<p align="center">Document</p> <p align="center">Technical Specification</p>	<p>version: vo.o8</p> <p>status:</p> <p>date: 2013-10-10</p>
<p>Reader-Host-Protocol – PUR-Extension</p>	<p>org</p>	

4.3 Get-Handle-From-Tag (04)

This function tries to get the handle of a specified tag.

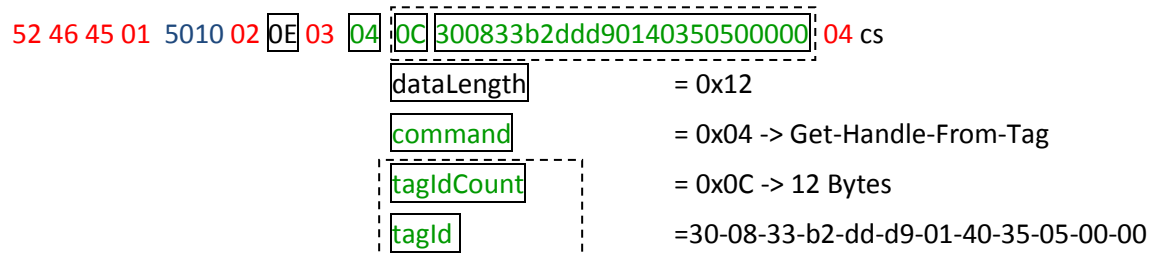
Parameters: unsigned char **command**, unsigned char **tagIdCount**,
unsigned char **tagId**[tagIdCount]

Return Values: RFE_RET_VALUE **status**, unsigned char **handle**[2]

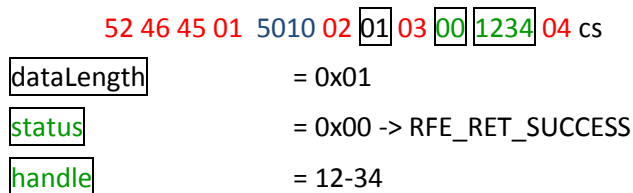
Status Values: RFE_RET_SUCCESS, RFE_RET_ERR_ON_EXEC_OP, RFE_RET_ERR_COULD_NOT_WRITE,
RFE_RET_ERR_WRONG_PARAM_COUNT, RFE_RET_ERR_WRONG_PARAM, Every TMI
Return Code


Example: Get the handle from the tag 30-08-33-b2-dd-d9-01-40-35-05-00-00:

PC -> Reader



Reader -> PC



	<p align="center">Document</p> <p align="center">Technical Specification</p>	<p>version: vo.o8</p> <p>status:</p> <p>date: 2013-10-10</p>
<p>Reader-Host-Protocol – PUR-Extension</p>	<p>org</p>	

4.4 Read-From-Handle (05)

With this function data can be read from the memory of a tag via the specified handle.

Parameters: unsigned char **handle**[2], unsigned char **memoryBank**, unsigned short **startAddress**, unsigned long **accessPassword**, unsigned char **byteCount**

Return Values: RFE_RET_VALUE **status**, unsigned char **byteCount**, unsigned char **data**[**byteCount**]

Status Values: RFE_RET_SUCCESS, RFE_RET_RESULT_PENDING, RFE_RET_ERR_ON_EXEC_OP, RFE_RET_ERR_WRONG_PARAM_COUNT, RFE_RET_ERR_WRONG_PARAM, Every TMI Return Code

Example: Read 5 byte from the tag with the handle 12-34 at the memory bank 1 and the start address 0x12:

PC -> Reader


52 46 45 01 5010 02 0B 03 05 1234 01 0000 00000000 06 04 cs

dataLength	= 0x0B
command	= 0x05 -> Read-From-Handle
handle	= 12-34
memoryBank	= 0x01 -> second bank
startAddress	= 0x0000
accessPassword	= 0x00000000
bytesCount	= 0x06 -> 6 Bytes

Reader -> PC

52 46 45 01 5010 02 08 03 00 06 020023A4884C 04 cs

dataLength	= 0x08
status	= 0x00 -> RFE_RET_SUCCESS
bytesCount	= 0x06 -> 6 Bytes
data	= 0x020023A4884C

	<p align="center">Document</p> <p align="center">Technical Specification</p>	<p>version: vo.o8</p> <p>status:</p> <p>date: 2013-10-10</p>
<p>Reader-Host-Protocol – PUR-Extension</p>	<p>org</p>	

4.5 Write-To-Handle (06)

With this function data can be written to the memory of a tag via the specified handle.

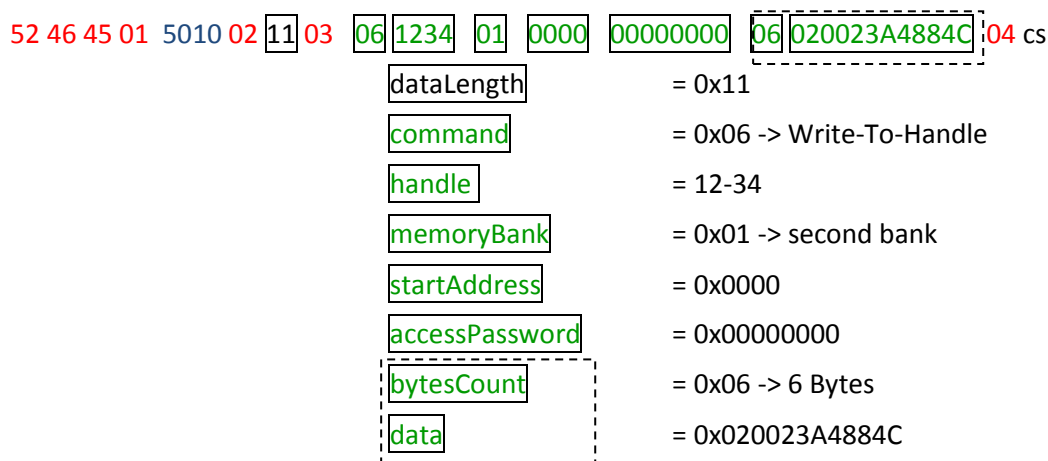
Parameters: unsigned char **handle**[2], unsigned char **memoryBank**, unsigned short **startAddress**, unsigned long **accessPassword**, unsigned char **byteCount**[byteCount]

Return Values: RFE_RET_VALUE **status**

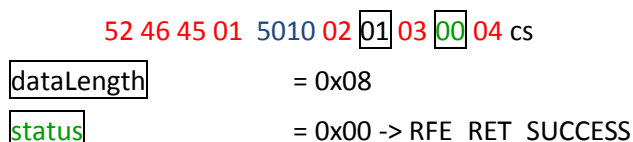
Status Values: RFE_RET_SUCCESS, RFE_RET_RESULT_PENDING, RFE_RET_ERR_ON_EXEC_OP, RFE_RET_ERR_COULD_NOT_WRITE, RFE_RET_ERR_WRONG_PARAM_COUNT, RFE_RET_ERR_WRONG_PARAM, Every TMI Return Code


Example: Write 5 byte to the tag with the handle 12-34 at the memory bank 1 and the start address 0x12:

PC -> Reader



Reader -> PC



	<p align="center">Document</p> <p align="center">Technical Specification</p>	<p>version: vo.o8</p> <p>status:</p> <p>date: 2013-10-10</p>
<p>Reader-Host-Protocol – PUR-Extension</p>	<p>org</p>	

4.6 Custom-tag-Command-Handle (07)

This function enables the reader to send custom Gen2 commands to the tag.

If the password is not zero, the reader tries to access the tag. The reader always appends the 16-bit handle at the end of the command.

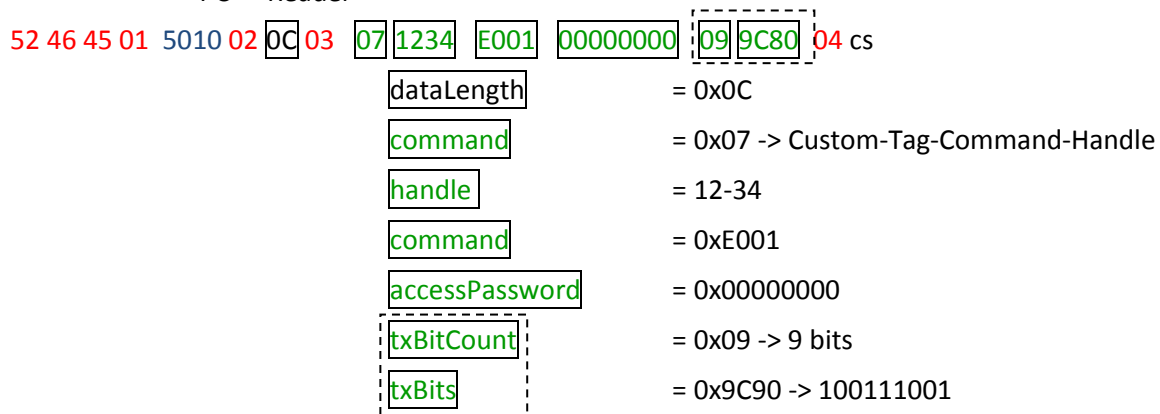
Parameters: unsigned char **handle**[2], unsigned char **command**[2],
unsigned long **accessPassword**, unsigned char **txBitCount**,
unsigned long **txBits** [bitCount*8]

Return Values: RFE_RET_VALUE **status**, bool **headerBit**,
unsigned char **rxByteCount**, unsigned char **rxBytes** [rxByteCount]

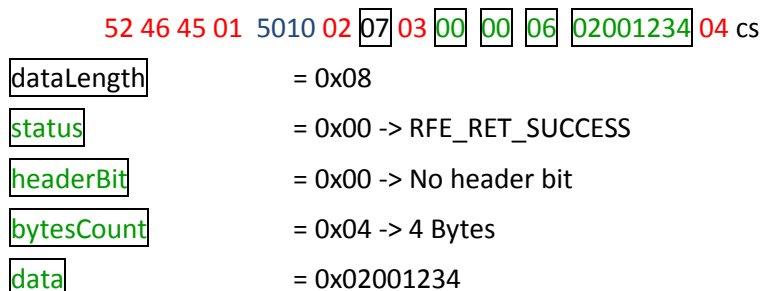
Status Values: RFE_RET_SUCCESS, RFE_RET_RESULT_PENDING, RFE_RET_ERR_ON_EXEC_OP,
RFE_RET_ERR_COULD_NOT_WRITE, RFE_RET_ERR_WRONG_PARAM_COUNT,
RFE_RET_ERR_WRONG_PARAM, Every TMI Return Code


Example: Send the custom command E0-01 with 9 bits payload (100111001) to the tag:

PC -> Reader



Reader -> PC



	<p align="center">Document</p> <p align="center">Technical Specification</p>	<p>version: vo.o8</p> <p>status:</p> <p>date: 2013-10-10</p>
<p>Reader-Host-Protocol – PUR-Extension</p>		<p>org</p>

5 Cyclic-Inventory Start Bytes

The standard start bytes for cyclic inventory information are extended with this type of reader. This is the complete list of start bytes.


Name	Byte	Size	Description
RFE_TAG_ID_START_BYTE	0x01	Variable	The ID of the detected Tag. The variable size of the id is sent in the first byte of the ID.
RFE_RSSI_START_BYTE	0x02	2 Bytes	The RSSI value of the detected tag.
RFE_ANTENNA_ID_START_BYTE	0x05	1 Byte	The antenna at which the tag was detected.
RFE_READ_FREQU_START_BYTE	0x06	3 Bytes	The frequency at which the tag was detected.
RFE_GEN2_HANDLE_START_BYTE	0x07	2 Bytes	The handle of the detected Gen2 tag.
RFE_GEN2_PC_START_BYTE	0x0A	2 Bytes	The PC of the detected Gen2 tag.

Example:

52 46 45 01 9002 02 1b 03 01 0c 000000000000000000000004 02 081e 06 0d37fc 07 5eac 0a 3000 04 cs

Part	Field	Description
01 0c 000000000000000000000004	ID	ID(12 Byte): 00-00-00-00-00-00-00-00-00-00-00-04
02 081e	RSSI	Signal Q: 8 dB Signal I: 30 dB
06 0d37fc	FREQU	Frequency: 866300 MHz
07 5eac	HANDLE	Gen2 Handle: 0x5eac
0a 3000	PC	Gen2 PC: 0x3000

<p>Reader-Host-Protocol-PUR-Extensions_o.o8_EN.docx</p>	<p align="center">R&D iDTRONIC GmbH</p>	<p align="right">page: 21 von 22</p>
---	--	---

	<div>Document</div> <div>Technical Specification</div>	<div>version: vo.o8</div> <div>status:</div> <div>date: 2013-10-10</div>
Reader-Host-Protocol – PUR-Extension		org

6 Notifications

The available notifications are:

ID	Name	Value	Description
0	Antenna-Power-Changed	1 byte = (bool) on	The notification is sent every time the antenna power changes.
1	Frequency-Changed	3 byte = (long) frequency	The notification is sent every time the frequency changes.
2	Inventory-Round-Ended	0 byte	The notification is sent every time an inventory round ended.
3	LBT-RSSI-Value-Measured	2 byte = (short) value	The notification is sent every time the LBT implementations measured a new RSSI value.

<p>Reader-Host-Protocol-PUR-Extensions_o.o8_EN.docx</p>	<p align="center">R&D iDTRONIC GmbH</p>	<p align="right">page: 22 von 22</p>
---	--	---