Using IUT-B40… RFID Readers in Allen-Bradley PLC

Reader Setup and Add-On Instructions for Use with Easy and Expert Mode

Table of Contents

Introduction 2
Importing EDS File 2
Assembly Selection 2
Configuring Parameters 4
Operational Parameters 5
UHF Parameters 6
Add-On Instructions 7
Initialization 7
Easy Mode 8
PF_RFID_Easy_Long 8
PF_RFID_Easy_Short 11
Easy Mode Timing Diagrams 12
Expert Mode 13
PF_RFID_Expert_Long 13
Expert Mode Timing Diagrams 19
Example 20

PEPPERL+FUCHS

Introduction

This document will show you how to use your IUT-F190-B40-2V1D-… reader in an Allen-Bradley PLC using Studio 5000 Add-On instructions. Easy and Expert mode will both be described here. Use the selection guide to choose the correct add-on for you.

	Easy Mode	Expert Mode
Digital IO in Process Data	✓	
One Bit Read/Write Control	✓	
All Read/Write Commands Possible		✓
Write EPC Codes		✓
Tag Present Bit	✓	
Filtering		✓
UHF Parameterization	Config	Config + PLC
Function Blocks Required	No	Yes

Importing EDS File

The EDS file will tell the PLC how to communicate to the B40 reader. The EDS file is imported using the EDS Hardware Installation tool (See figure 1), and the EDS file can be found on the product page on the internet.

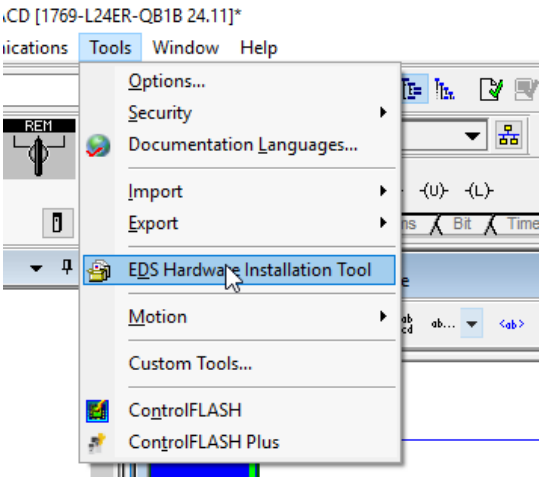


Figure 1: EDS file installation method in Studio 5000 Logix Designer

Assembly Selection

The B40 reader is added to the project by right-clicking on the Ethernet network and selecting New Module (See Figure 2). Select the B40 reader from the device catalog and click Create (See Figure 3).

PEPPERL+FUCHS

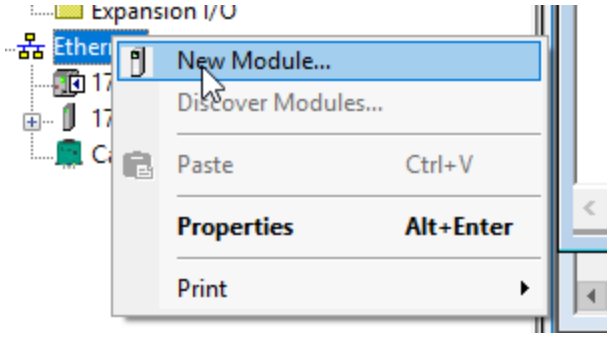


Figure 2: Adding B40 reader to the project

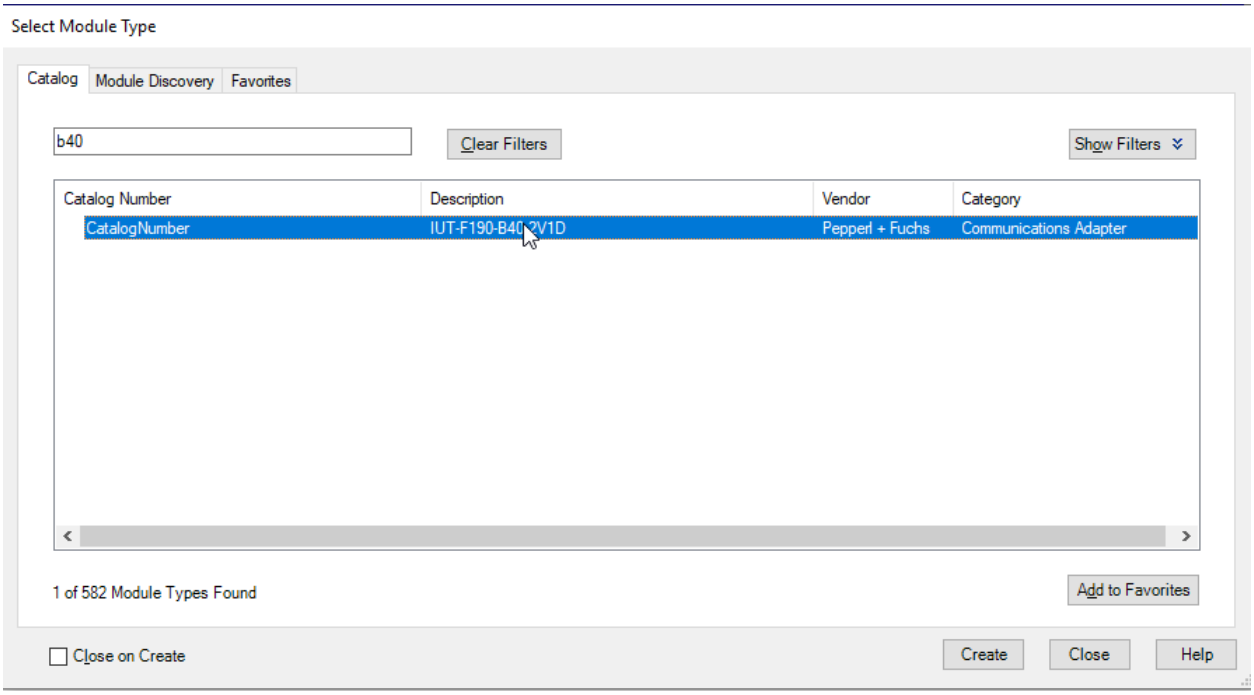


Figure 3: Creating module after selection in product catalog

There are four assemblies to choose from (See figure 4). The length is configured as 32 bytes or 128 bytes and UHF parameters are configured if selected. All assemblies allow for changing of the operating mode and Easy mode configuration parameters.

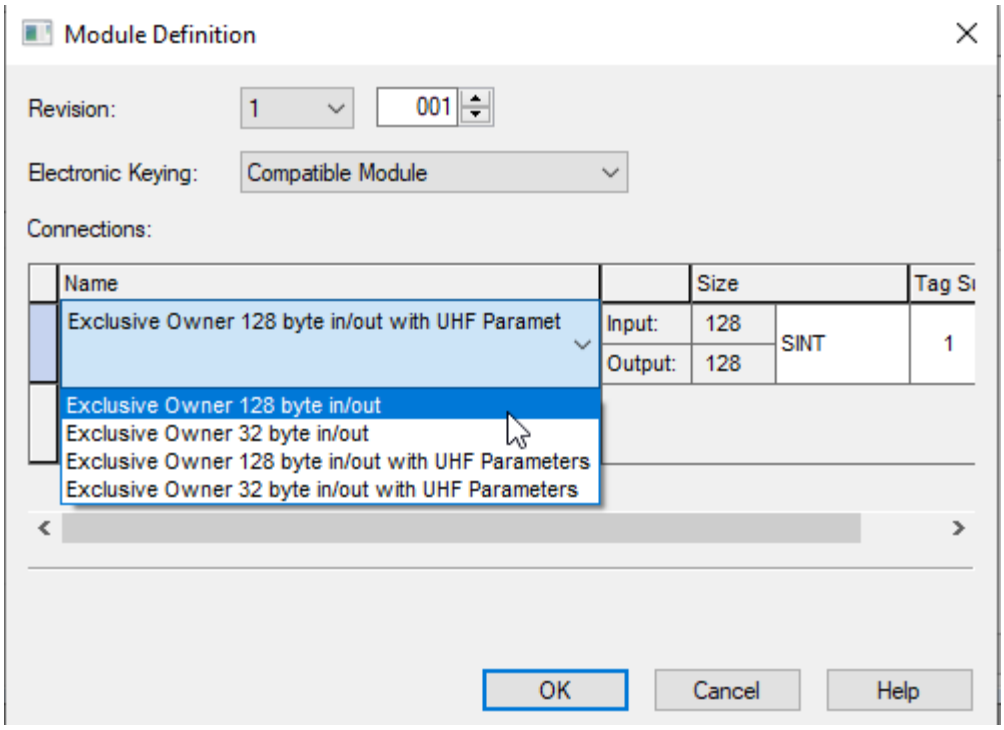


Figure 4: Assembly selection

Configuring Parameters

All parameters can be set in the configuration assembly. The parameters are sent to the controller on Forward Open of EtherNet/IP protocol. In order to configure the reader again you must power cycle the controller, download the program or inhibit/uninhibit the B40 reader (See figure 5).

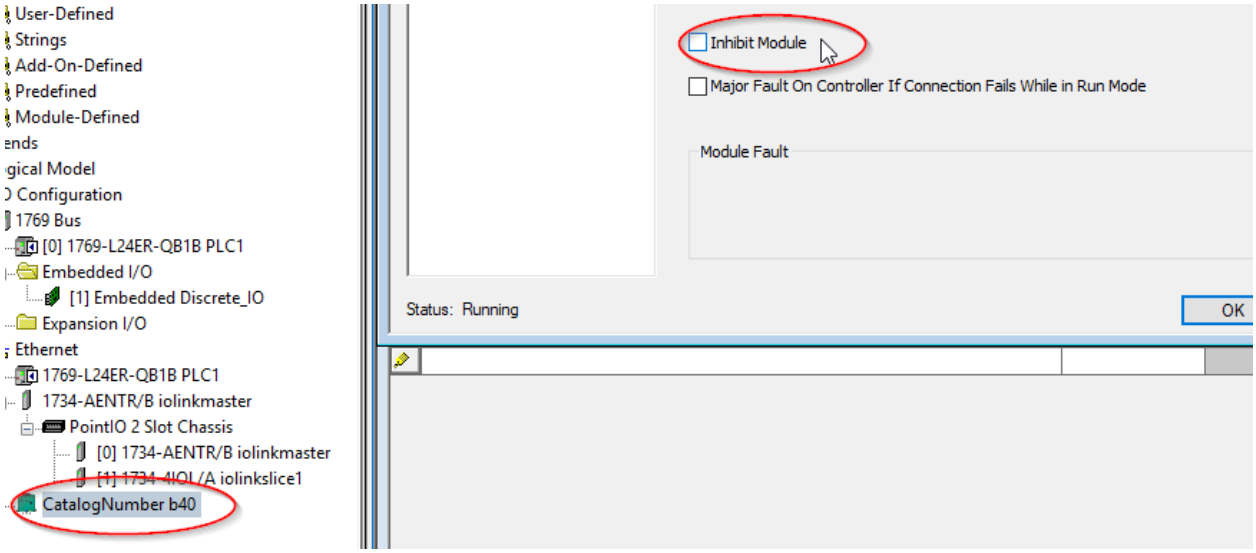


Figure 5: Inhibiting controller to force the forward-open and parameter change

Operational Parameters

The first set of parameters are operation parameters. They tell the B40 reader how to work properly. They are available in all assemblies (See figure 6).

b40:C	{...}		_0039:IUT_F190_B40_2V1...
b40:C.Operation_Parameter_Operation_Mode	128	Decimal	INT
b40:C.Read_Task_Memory_Area	0	Decimal	INT
b40:C.Read_Task_Number_of_bytes	8	Decimal	INT
b40:C.Read_Task_Start_address	0	Decimal	DINT
b40:C.Read_Task_Autostart	0	Decimal	INT
b40:C.Write_Task_Memory_Area	0	Decimal	INT
b40:C.Write_Task_Number_of_bytes	8	Decimal	INT
b40:C.Write_Task_Start_address	0	Decimal	DINT
b40:C.Input_Representation	0	Decimal	INT

Figure 6: Operational Parameters

Operation_Parameter_Operating_Mode – Read/Write mode. **Make sure the mode is set correctly for your application**

128 = Easy mode (default)
0 = Expert mode

Read_Task_Memory_Area – The type of data read in Easy mode

0 = UII/EPC + User Memory (default)
64 = UII/EPC only
128 = UII/EPC + TID

Read_Task_Number_of_bytes – The number of bytes to read in Easy mode. Must be in multiples of 4. Make sure the assembly can support the quantity.

4 = four bytes to read
8 = eight bytes to read (default)
12 = twelve bytes to read (largest size for 32-byte assembly with 96 bit EPC Long input representation, only 10 bytes returned)
…
64 = sixty-four bytes (largest size for most UHF tags)
…
124 = 124 bytes to read

Read_Task_Start_Address – The start location to read data in Easy mode

Read_Task_AutoStart – If enabled makes the reader start reading on power-up. If enabled you have no control over starting/stopping of the read command. No writing is allowed.

0 = off (default)
128 = Enabled

Write_Task_Memory_Area – The area of memory to perform the write operation

0 = User memory (default)

Write_Task_Number_of_bytes - The number of bytes to write in Easy mode. Must be in multiples of 4. Make sure the assembly can support the quantity.

4 = four bytes to write
8 = eight bytes to write (default)
12 = twelve bytes to write
…
28 = twenty-eight bytes to write (largest size using 32 byte assembly)
…
64 = sixty-four bytes to write (largest size for most UHF tags)
…
124 = 124 bytes to write

Write_Task_Start_Address - The start location to write data in Easy mode

Input_Representation – The format of the data in response.

0 = EPC included with all responses (default)
128 = EPC is not included with response

UHF Parameters

UHF parameters can only be modified if the correct assembly is chosen. If used, then all UHF parameters of configuration tags will overwrite the current parameters in the head. Parameter changes on the web interface can still be used for testing purposes, but these parameters will take effect on the next EtherNet/IP Forward Open (See figure 7). For a detailed list of allowed values and functionality of each parameter see the B40 interface manual. UHF parameters could be different for each region code.

b40:C.Power_Transmission_1_PT	125	Decimal	INT
b40:C.Power_Transmission_2_PT	0	Decimal	INT
b40:C.Power_Transmission_3_PT	0	Decimal	INT
b40:C.Power_Transmission_4_PT	0	Decimal	INT
b40:C.Power_Transmission_5_PT	0	Decimal	INT
b40:C.Power_Transmission_6_PT	0	Decimal	INT
b40:C.Power_Transmission_7_PT	0	Decimal	INT
b40:C.Power_Transmission_8_PT	0	Decimal	INT
b40:C.Power_Transmission_9_PT	0	Decimal	INT
b40:C.Power_Transmission_10_PT	0	Decimal	INT
b40:C.Number_Of_Tags_To_Find_NT	255	Decimal	INT
b40:C.Tries_Allowed_TA	2	Decimal	INT
b40:C.Q_Value_QW	2	Decimal	SINT
b40:C.Enhanced_Status_5_E5	5	Decimal	INT
b40:C.Antenna_Polarization_AP	67	Decimal	SINT
b40:C.Trigger_Input_1_Mode	255	Decimal	INT
b40:C.Trigger_Input_1_Action	255	Decimal	INT
b40:C.Trigger_Input_1_Execution_Type	128	Decimal	INT
b40:C.Trigger_Input_1_Memory_Area	0	Decimal	INT
b40:C.Trigger_Input_1_Memory_Block_Address	0	Decimal	DINT
b40:C.Trigger_Input_1_Byte_Length	4	Decimal	INT
b40:C.Trigger_Input_2_Mode	255	Decimal	INT
b40:C.Trigger_Input_2_Action	255	Decimal	INT
b40:C.Trigger_Input_2_Execution_Type	128	Decimal	INT
b40:C.Trigger_Input_2_Memory_Area	0	Decimal	INT
b40:C.Trigger_Input_2_Memory_Block_Address	0	Decimal	DINT
b40:C.Trigger_Input_2_Byte_Length	4	Decimal	INT
b40:C.Trigger_Output_Mode	255	Decimal	INT
b40:C.Trigger_Output_Event	255	Decimal	INT

Figure 7: UHF Parameters for FR2-02 readers.

Add-On Instructions

Add-on Instructions are easy-to-use function blocks for use in the Rockwell Automation Studio 5000 software. Easy mode does not require the use of function blocks but they will make integration of the reader in to your PLC program easier.

Initialization

In the Off-To-On transition of the **Read** or the **Write** bits the *ResponseData* array is cleared, *ErrorInfo* and the *ResponseCounter* is reset to 0.

Instruction	Description

PF_RFID_Easy_Long	Easy triggering of read and write commands when the B40 reader is configured for Easy mode, Input_Representation must also be set to Long (Data includes the EPC code and multiple tags in field are allowed)
PF_RFID_Easy_Short	Easy triggering of read and write commands when the B40 reader is configured for Easy mode, Input_Representation must also be set to Short (Data does not include EPC code and multiple tags in field are forbidden)
PF_RFID_Expert_Long	Used to trigger of read and write commands, UHF configuration and filtering when the B40 reader is configured for Expert Mode. Input_Representation must also be set to Long
PF_RF_Exp_Rcv	Internal Add-on that takes care of receiving the Expert Mode fragmentation protocol
PF_RF_Exp_Snd	Internal Add-on that takes care of sending the Expert Mode fragmentation protocol

Easy Mode

PF_RFID_Easy_Long

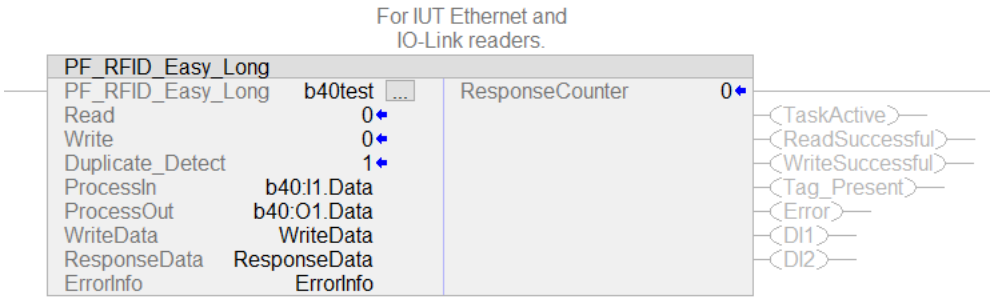
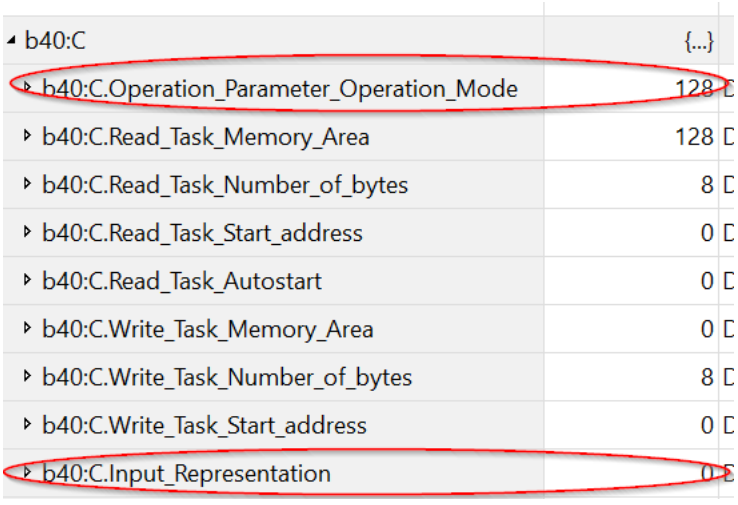


Figure 8: Easy Mode Function Block

To use this Add-On instruction the mode must be set to Easy Mode and the Input Representation must be set to Long. The value of these parameters is 128 and 0 respectively. See the Operational Parameter section for a detail explanation of each parameter.



Additionally make sure to adjust the other operating parameters so the correct amount and location of the data are read. Also adjust the Autostart parameter as well. If this parameter is enabled then the reader will read all the time and the triggers will not do anything.

Name	Value	Style	Data Type
b40:C	{...}		_0039:IUT_
b40:C.Operation_Parameter_Operation_Mode	128	Decimal	INT
b40:C.Read_Task_Memory_Area	128	Decimal	INT
b40:C.Read_Task_Number_of_bytes	8	Decimal	INT
b40:C.Read_Task_Start_address	0	Decimal	DINT
b40:C.Read_Task_Autostart	0	Decimal	INT
b40:C.Write_Task_Memory_Area	0	Decimal	INT
b40:C.Write_Task_Number_of_bytes	8	Decimal	INT
b40:C.Write_Task_Start_address	0	Decimal	DINT
b40:C.Input_Representation	0	Decimal	INT

Input Parameters

Read – Triggers the reading of tags.

0 = No reading
1 = Read all the time

Write – Triggers the writing of tags.

0 = No write
1 = Write all the time

Duplicate_Detect – Searches for duplicate tags in *ResponseData* array. Not used when writing

0 = Adds tag to *ResponseData* array at location [0] and shifts all other tags down in array
1 = Searches the *ResponseData* array first. If the tag has not been read before then adds tag to *ResponseData* array at location [0] and shifts all other tags down in array (default).

In/Out Parameters

ProcessIn – The input data of the B40 reader

ProcessOut – The output data of the B40 reader

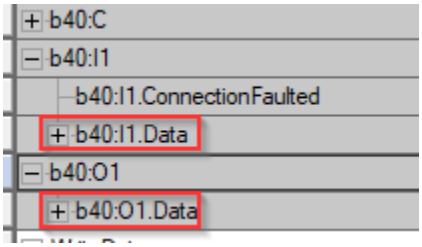


Figure 9: Input and output variables used in ProcessIn and ProcessOut variables

WriteData – The string of data that is used when writing to tags. This data only takes affect with the Off-to-On transition of the **Write** bit. The datatype is a STRING variable of type PF_String192.

ResponseData – This is the location where all read data and all responses to write commands are placed. The array size will determine how many tags are stored. The array can be as small as 1 or as large as the PLC memory allows (See Figure 10 for an example for storing 10 tags).

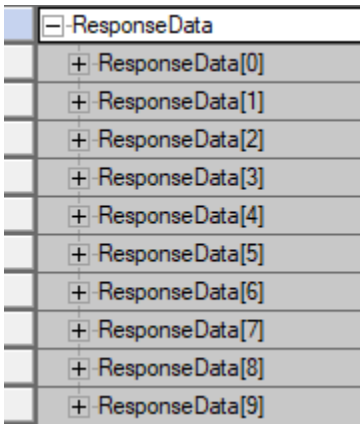


Figure 10: Array size set to 10 to store up to 10 tags

If the array size is more than 1 the newest data is placed in the [0] position and all older data is moved down in the array (See Figure 11).

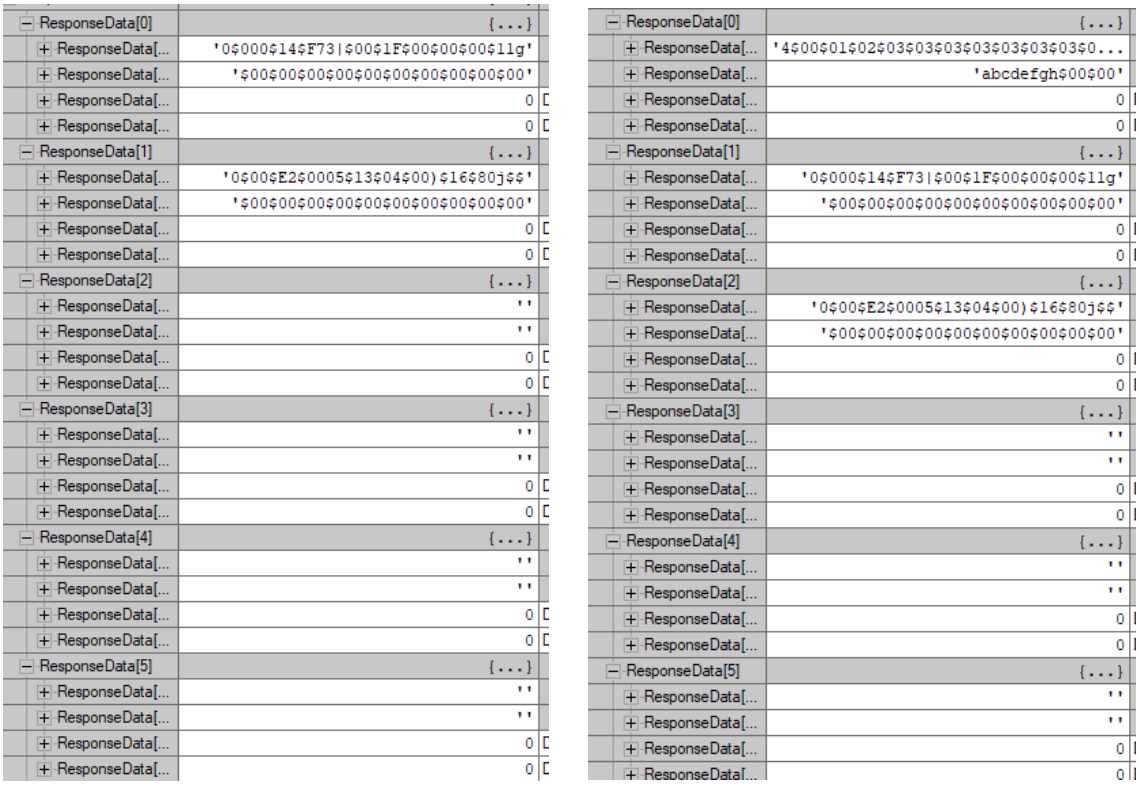


Figure 11: New tag added to position [0] and all older tags moved one index higher

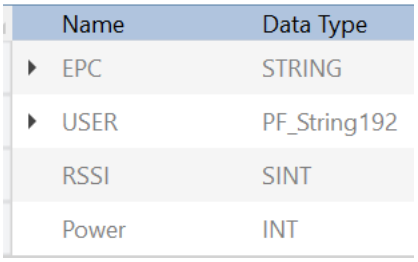
The data for the tag contains up to 4 pieces of information including EPC, USER, RSSI and Power (See Figure 12).

EPC – The UII/EPC code, the EPC code will also include the PC (Protocol Control Word). This is the value searched when checking for duplicate tags. No checking is done when writing tags.

USER – The second set of data read from the tag. It could be the user data or the TID code. Not used when writing tags.

RSSI – The RSSI value of tag

Power – The power setting of the reader when the tag was read or written



Name	Data Type
EPC	STRING
USER	PF_String192
RSSI	SINT
Power	INT

Figure 12: UDT for responses to reading and writing of tags

ErrorInfo – The error code and error string that is returned when the **Error** bit goes on

Output parameters

TaskActive – A reading or writing command is currently running in the head

ReadSuccess – A tag was read successfully

WriteSuccess – A tag was written successfully

Tag_Present – A tag is currently in front of the head

Error – An error occurred, See ErrorInfo for error details

ResponseCounter – Counts up with every successful read or write attempt. No count up on reading if Duplicate Detect is enabled

PF_RFID_Easy_Short

This Add-On is exactly the same as the PF_RFID_Easy_Long except for the follow information. The Short input representation will respond without the EPC code. This additionally requires that only one tag ever be present in front of the reader. Multiple tags in field will generate an error code 10 “multiple tags in field”.

To use this Add-On instruction the mode must be set to Easy Mode and the Input Representation must be set to Short. The value of these parameters is 128 and 128 respectively. See the Operational Parameter section for a detail explanation of each parameter.

b40:C	{...}		_0039:I
b40:C.Operation_Parameter_Operation_Mode	128	Decimal	INT
b40:C.Read_Task_Memory_Area	128	Decimal	INT
b40:C.Read_Task_Number_of_bytes	8	Decimal	INT
b40:C.Read_Task_Start_address	0	Decimal	DINT
b40:C.Read_Task_Autostart	0	Decimal	INT
b40:C.Write_Task_Memory_Area	0	Decimal	INT
b40:C.Write_Task_Number_of_bytes	8	Decimal	INT
b40:C.Write_Task_Start_address	0	Decimal	DINT
b40:C.Input_Representation	128	Decimal	INT

The data for the tag contains up to 3 pieces of information including USER, RSSI and Power (See Figure 13).

Name	Data Type
Data	PF_String192
RSSI	SINT
Power	INT

Figure 13: UDT for responses to reading and writing of tags in Short input representation

Easy Mode Timing Diagrams

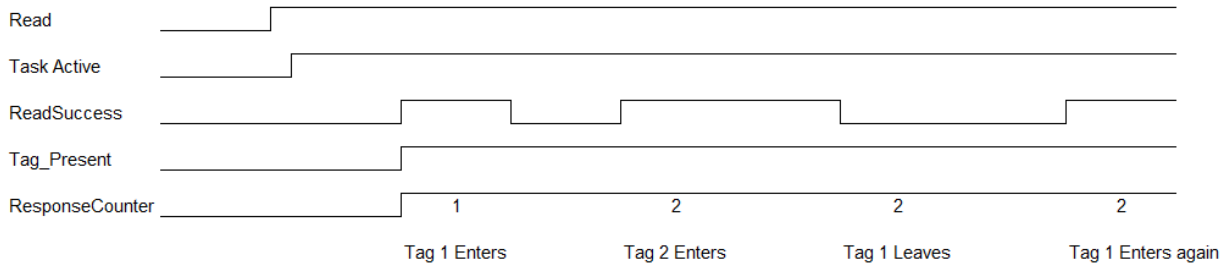


Figure 14: Timing Diagram for reading when Duplicate_Detect is enabled and Long input representation

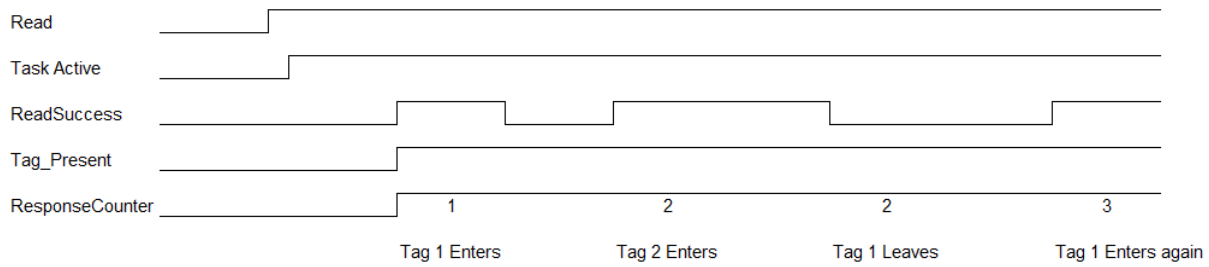


Figure 15: Timing Diagram for reading when Duplicate_Detect is disabled Long input representation

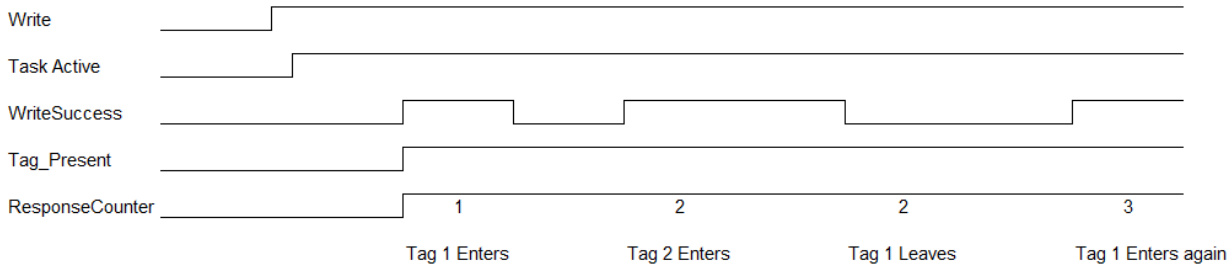


Figure 16: Timing Diagram for writing

Expert Mode

PF_RFID_Expert_Long

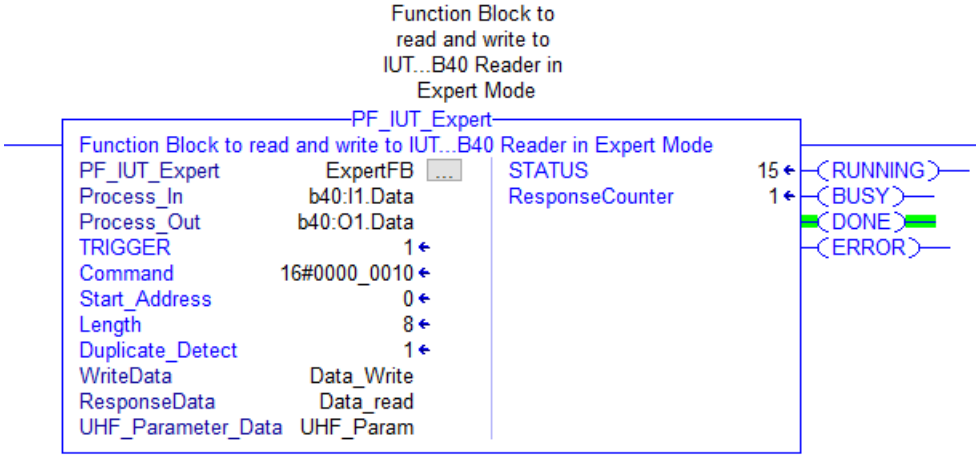


Figure 17: Expert Mode Function Block

Input Parameters

TRIGGER – In the Off-To-On transition a new command that is located in *Command* is executed

Command – The Command to be executed. See below table for options.

Single Commands – Executes one time on Off-To-On transition

PEPPERL+FUCHS

Enhanced Commands – Continuously executes

Command Code(hex)	Function
Read/Write Commands	
01	Single Read EPC + TID
1D	Continuous Read EPC + TID
0A	Single Read EPC
71	Continuous Read EPC
CE	Single Write EPC
10	Single Read EPC + User Data (Byte length must be multiple of 4)
49	Single Read EPC + User Data (Byte length must be multiple of 2)
19	Enhanced Read EPC + User Data (Byte length must be multiple of 4)
4B	Enhanced Read EPC + User Data (Byte length must be multiple of 2)
40	Single Write User Data (Byte length must be multiple of 4)
4A	Single Write User Data (Byte length must be multiple of 2)
1A	Enhanced Write User Data (Byte length must be multiple of 4)
4C	Enhanced Write User Data (Byte length must be multiple of 2)
System Commands	
02	Quit Enhanced command
03	Read Version
5244BF	RD, Reset all UHF Parameters to Default
Read UHF Parameters	
1BE	RC, Read Region Code
2BE	AP, Read Polarization
2BF	AP, Write Polarization (‘**C**’, ‘H’ or ‘V’)
3BE	PT, Read Power
3BF	PT, Write Power (3, 4, 5, 6, 8, 10, 13, 15, 20, 25, 30, 40, 50, 60, 80, 100, **125**, 200, 250, 300, 400, 500, 600, 800, 1000, 1250) mW
4BE	CD, Read Channels
4BF	CD, Write Channels for (FR1-01 only) 4, 7, 10, 13
5BE	NC, Read Number of Channels
5BF	NC, Write Number of Channels (1, 2, 3, **4**…50) depends on Model
6BE	TA, Read Number of Attempts
6BF	TA, Write Number of Attempts (1, **2**, 3…255)
7BE	E5, Read Enhanced Status 5
7BF	E5, Write Enhanced Status 5 (1,2,3,**4**,5…255)
8BE	MB, Read Memory Bank to use with Read/Write commands
9BF	MB, Write Memory Bank to use with Read/Write commands (1, 2, **3**)
ABE	QW, Read Q Value
ABF	QW, Write Q Value
BBE	NT, Read Cancelation Criteria
BBF	NT, Write Cancellation Criteria (1,2…**255**)
CBE	SM, Read Transmission Pauses
CBF	SM, Write Transmission Pauses (**0**…65535) (FR1-01 default=100)
DBE	MF, Read Measured Reflections

Filter Commands	
CB	Write Filter Mode (**0**,1,2)
10BE	Read Filter String 1
0CA	Write Filter String 1
11BE	Read Filter String 2
1CA	Write Filter String 2
12BE	Read Filter String 2
2CA	Write Filter String 2

Start_Address – The first byte to start reading or writing. Only used for Read/Write commands

0 = First byte (default)

x = Must be a multiple of two or four bytes depending on the command

Length – The Number of bytes to read or write. Only used for Read/Write commands.

x = Must be a multiple of two or four bytes depending on the command (default 8)

Duplicate_Detect – Searches for duplicate tags in the ***ResponseData*** array. Not used for Write commands.

0 = Adds RFID tag to the ***ResponseData*** array at location [0] and shifts all other tags down in array

1 = Searches the ***ResponseData*** array first. If the RFID tag has not been read before it will be added in the ***ResponseData*** array at location [0] and shifts all other tags down in array (default).

In/Out Parameters

ProcessIn – The input data of the B40 reader

ProcessOut – The output data of the B40 reader

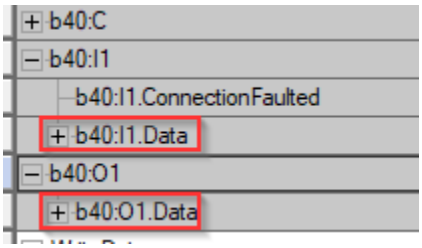


Figure 18: Input and output variables used in ProcessIn and ProcessOut variables

WriteData – The string of data that is used when writing to tags. The datatype is a STRING variable of the type PF_String192.

ResponseData – This is the location where all read data and all responses to Write commands are placed. The array size will determine how many tags are stored. The array can be as small as 1 or as large as the PLC memory allows (See Figure 19 for an example for storing 10 tags).

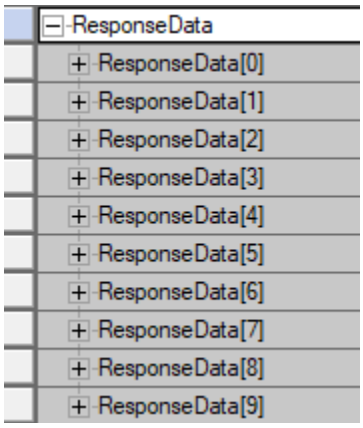


Figure 19: Array size set to 10 to store up to 10 tags

If the array size is more than 1 the newest data is placed in the [0] position and all older data is moved down in the array (See Figure 20).

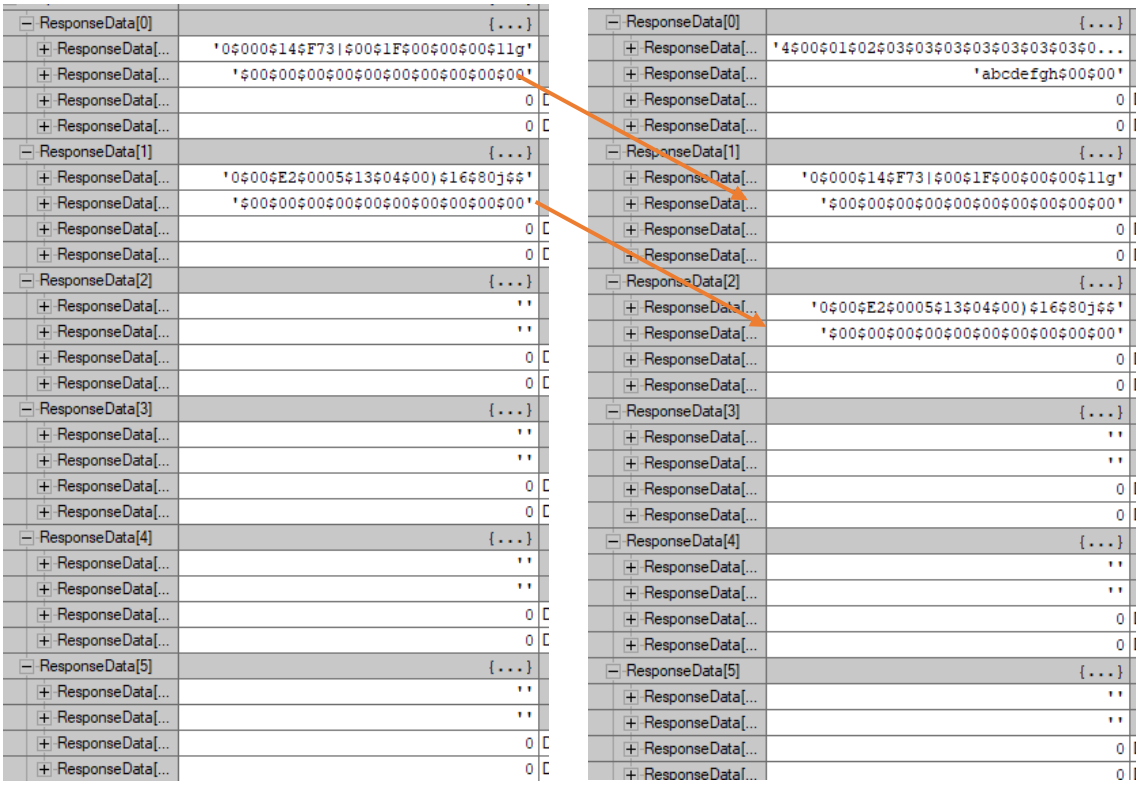


Figure 20: New tag added to position [0] and all older tags moved one index higher

The data for the tag contains up to four pieces of information including EPC, USER, RSSI and Power (See Figure 21).

EPC – The UII/EPC code, the EPC code will also include the PC (Protocol Control Word). This is the value searched when checking for duplicate tags. No checking is done when writing tags.

USER – The second set of data read from the tag. It could be the user data or the TID code. Not used when writing tags.

RSSI – The RSSI value of tag

100 – Tag is very close
0 – Tag is far away

Power – The power setting of the reader when the tag was read or written

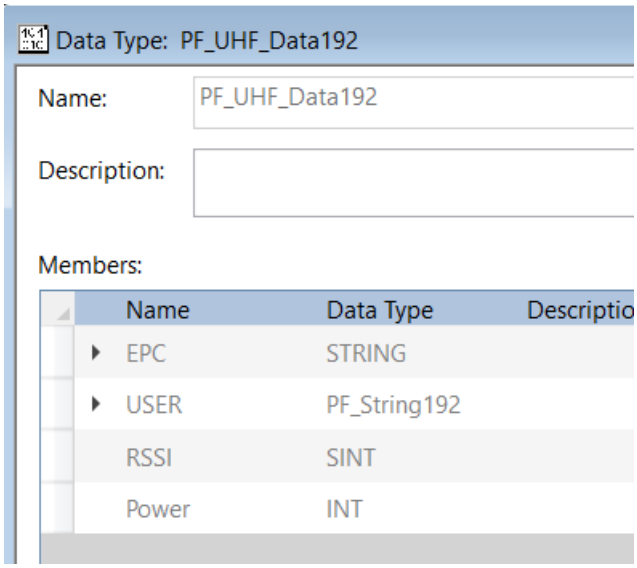


Figure 21: UDT for responses to reading and writing of tags

UHF_Parameter_Data – All data used when reading and writing parameters. Default parameters are not loaded automatically. It is suggested that you read the parameter to be changed, adjust it as required for the application and write the new parameter to the reader. If you want the parameter to be permanent and you also mapped UHF parameters in the EtherNet/IP configuration then it is suggested that you write the parameter there in parallel. See the UHF_Parameter section above, and **Figure 22**below for all parameter options.

UHF_Param	{...}		UHF_Parameter...	
UHF_Param.RC_Region_Code_01	16#8002	Hex	INT	1BE/xxx, Region where the reader is approved, see Manual for o...
UHF_Param.AP_Polarization_02	'$00'	ASCII	SINT	2BE/2BF, C=Combined H=Horizontal V=Vertical
UHF_Param.PT_Transmit_Power_03	{...}	Decimal	INT[10]	3BE/3BF, Sets the transmitted power in mW. Cycles through up to...
UHF_Param.PT_Transmit_Power_03[0]	125	Decimal	INT	3BE/3BF, Sets the transmitted power in mW. Cycles through up to...
UHF_Param.PT_Transmit_Power_03[1]	0	Decimal	INT	3BE/3BF, Sets the transmitted power in mW. Cycles through up to...
UHF_Param.PT_Transmit_Power_03[2]	0	Decimal	INT	3BE/3BF, Sets the transmitted power in mW. Cycles through up to...
UHF_Param.PT_Transmit_Power_03[3]	0	Decimal	INT	3BE/3BF, Sets the transmitted power in mW. Cycles through up to...
UHF_Param.PT_Transmit_Power_03[4]	0	Decimal	INT	3BE/3BF, Sets the transmitted power in mW. Cycles through up to...
UHF_Param.PT_Transmit_Power_03[5]	0	Decimal	INT	3BE/3BF, Sets the transmitted power in mW. Cycles through up to...
UHF_Param.PT_Transmit_Power_03[6]	0	Decimal	INT	3BE/3BF, Sets the transmitted power in mW. Cycles through up to...
UHF_Param.PT_Transmit_Power_03[7]	0	Decimal	INT	3BE/3BF, Sets the transmitted power in mW. Cycles through up to...
UHF_Param.PT_Transmit_Power_03[8]	0	Decimal	INT	3BE/3BF, Sets the transmitted power in mW. Cycles through up to...
UHF_Param.PT_Transmit_Power_03[9]	0	Decimal	INT	3BE/3BF, Sets the transmitted power in mW. Cycles through up to...
UHF_Param.CD_Transmission_Channels_04	{...}	Decimal	SINT[4]	4BE/4BF, Transmission Channels Used only for the -01 Readers
UHF_Param.NC_Number_of_Channels_05	4	Decimal	SINT	5BE/5BF, The number of channels per read attempt
UHF_Param.TA_Number_of_Attempts_06	0	Decimal	SINT	6BE/6BF, The number of read or write attempts to perform
UHF_Param.E5_Enhanced_Status5_07	5	Decimal	SINT	7BE/7BF, This is how many unsuccessfull
UHF_Param.MB_Memory_Bank_08	0	Decimal	SINT	8BE/8BF, The memory bank used when the Read/Write User me...
UHF_Param.QW_Q_Value_0A	0	Decimal	SINT	ABE/ABF, Used in ALOHA principal for anti-collision 2^Q 0=1 1=2...
UHF_Param.NT_Cancellation_Criteria_for_Search_0B	255	Decimal	INT	BBE/BBF, 255=not used 1-x number of tags to look for during a si...
UHF_Param.SM_Transmission_Pause_0C	0	Decimal	DINT	CBE/CBF, 0=no pause x=the pause time between cycles where n...
UHF_Param.MF_Measure_Reflection_0D	{...}	Decimal	SINT[52]	DBE/xxx, The measure of the reflected transmitted power in the e...
UHF_Param.Filter_Mode	0	Decimal	SINT	xxx/CB, 0=Off, 1=Mode 1, 2=Mode 2
UHF_Param.Filter1	{...}		PF_UHF_Filter	10BE/0CA
UHF_Param.Filter1.MemoryBank	1	Decimal	SINT	1=EPC, 2=TID, 3=User
UHF_Param.Filter1.Negation	0	Decimal	BOOL	0=Transfer tags that match, 1=Transfer tags that do not match
UHF_Param.Filter1.Logical_Operation	0	Decimal	BOOL	0=Link all Filter Masks with OR, 1=Link all Filter Masks with AND (...
UHF_Param.Filter1.Truncation	0	Decimal	BOOL	0=All Filter Data is transmitted, 1=Only Data following the Filter is ...
UHF_Param.Filter1.FilterAddress	32	Decimal	INT	First Bit you will filter to
UHF_Param.Filter1.FilterLength	32	Decimal	INT	Number of bits you will filter
UHF_Param.Filter1.Mask	{...}	Decimal	SINT[32]	10BE/0CA
UHF_Param.Filter2	{...}		PF_UHF_Filter	11BE/1CA
UHF_Param.Filter2.MemoryBank	1	Decimal	SINT	1=EPC, 2=TID, 3=User
UHF_Param.Filter2.Negation	1	Decimal	BOOL	0=Transfer tags that match, 1=Transfer tags that do not match
UHF_Param.Filter2.Logical_Operation	1	Decimal	BOOL	0=Link all Filter Masks with OR, 1=Link all Filter Masks with AND (...
UHF_Param.Filter2.Truncation	0	Decimal	BOOL	0=All Filter Data is transmitted, 1=Only Data following the Filter is ...
UHF_Param.Filter2.FilterAddress	32	Decimal	INT	First Bit you will filter to
UHF_Param.Filter2.FilterLength	32	Decimal	INT	Number of bits you will filter
UHF_Param.Filter2.Mask	{...}	Decimal	SINT[32]	11BE/1CA
UHF_Param.Filter3	{...}		PF_UHF_Filter	12BE/2CA
UHF_Param.Filter3.MemoryBank	1	Decimal	SINT	1=EPC, 2=TID, 3=User
UHF_Param.Filter3.Negation	0	Decimal	BOOL	0=Transfer tags that match, 1=Transfer tags that do not match
UHF_Param.Filter3.Logical_Operation	0	Decimal	BOOL	0=Link all Filter Masks with OR, 1=Link all Filter Masks with AND (...
UHF_Param.Filter3.Truncation	0	Decimal	BOOL	0=All Filter Data is transmitted, 1=Only Data following the Filter is ...
UHF_Param.Filter3.FilterAddress	32	Decimal	INT	First Bit you will filter to
UHF_Param.Filter3.FilterLength	32	Decimal	INT	Number of bits you will filter
UHF_Param.Filter3.Mask	{...}	Decimal	SINT[32]	12BE/2CA
UHF_Param.Version	' (C) P...		STRING	3/xxx, Version

***Figure 22**: Parameters Read and Written to UHF head*

Output parameters

Running – On while a Read or Write command is executing

Busy – A command is in the process of executing, and first answer is not back yet.

Done – A command has completed executing. Not used for Enhanced commands.

Error – An error occurred; see **Status** for details

Status – Detail on the execution of a command; see **Figure 23** below

Status	Description	Cause
-1	Busy executing a command	-
-2	Timeout	Not in Expert Mode or reader not connected
0	Command executed correctly	-
11	Tag present, additional info	-
15	Single command complete	-
4	Bad command	Check parameters are within acceptable range
10	Multiple tags in field with same EPC	Reprogram EPC codes to be unique
14	Buffer overflow	Slow down reading or writing, or too many tags in field

***Figure 23**: Possible Status Codes*

ResponseCounter – Counts up with every successful read or write attempt. Will not count duplicate tags if **Duplicate_Detect** is enabled. Only resets if a read/write command or **Write Parameter** command is sent. **Read Parameter** commands can be executed while a read command is running so that the counter remains unchanged.

Expert Mode Timing Diagrams

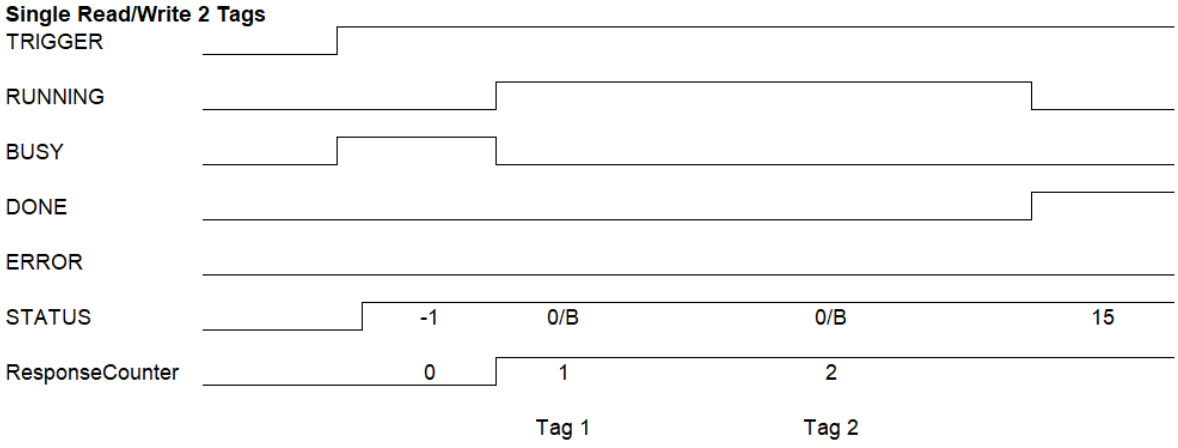


*Figure24: Timing Diagram for **Single Read/Write 2 Tags***

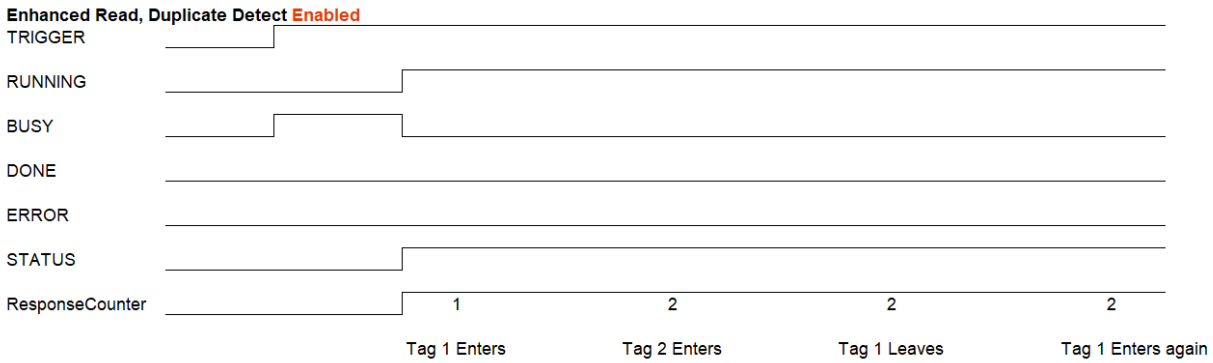


*Figure 25: Timing Diagram for **Enhanced Read** with **Duplicate Detect Enabled***

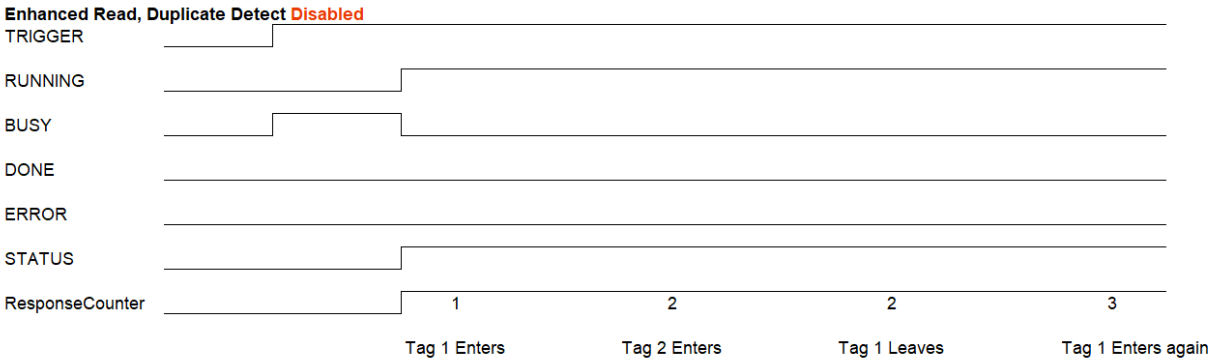


*Figure 26: Timing Diagram for **Enhanced Read** with **Duplicate Detect Disabled***

Example

See video that is included