

Handbuch

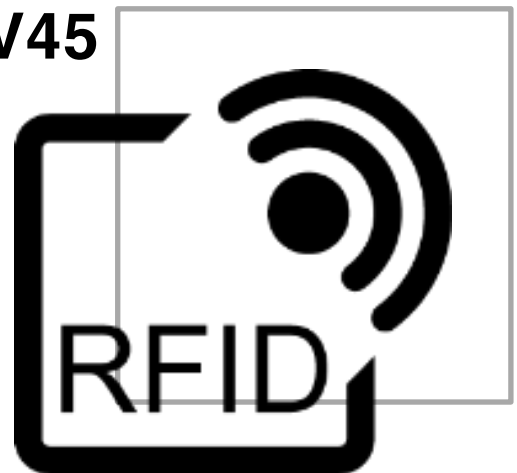
**Inbetriebnahme**

**IDENTControl IC-KP-B12-V45**

**und IC-KP-B17-AIDA1 mit**

**Profinet Protokoll an**

**Siemens S7 Steuerung**



Es gelten die Allgemeinen Lieferbedingungen für Erzeugnisse und Leistungen der Elektroindustrie, herausgegeben vom Zentralverband Elektroindustrie (ZVEI) e.V. in ihrer neusten Fassung sowie die Ergänzungsklausel: "Erweiterter Eigentumsvorbehalt".

<b>1</b>	<b>Einstellung Kommunikationsparameter über Webinterface ....</b>	<b>5</b>
<b>2</b>	<b>Installation GSDML-Datei.....</b>	<b>8</b>
<b>3</b>	<b>Installation Anwenderprogramm .....</b>	<b>9</b>
<b>4</b>	<b>Hardwarekonfiguration .....</b>	<b>10</b>
<b>5</b>	<b>Einstellung Geräteparameter .....</b>	<b>13</b>
<b>6</b>	<b>Funktionsbaustein „FB190_IUHParm“ .....</b>	<b>14</b>
6.1	Reset to Default: Kanal 1.....	17
6.2	PowerTransmit - PT: Kanal 1 .....	18
6.3	ChannelDenseReaderMode - CD: Kanal 1.....	19
<b>7</b>	<b>Funktionsbaustein FB32 „Multiframe“ .....</b>	<b>22</b>
7.1	Filterkonfiguration .....	29
<b>8</b>	<b>Fehler- bzw. Gerätediagnose.....</b>	<b>31</b>
<b>9</b>	<b>Beispiele Befehlsausführungen .....</b>	<b>32</b>
9.1	Initialisierung: (mit Datenträgertyp IUC72).....	32
9.2	Single Read Fixcode: (Kopf 1).....	33
9.3	Enhanced Read Fixcode: (Kopf 1) .....	34
9.4	Single Read Words: (Kopf 1; 15 Blöcke ab Adresse 0) .....	35
9.5	Enhanced Read Words: (Kopf 1; 15 Blöcke ab Adresse 0) .....	36
9.6	Single Read SpecialFixcode: (Kopf 1).....	37
9.7	Enhanced Read SpecialFixcode: (Kopf 1) .....	38
9.8	Single Write Words: (Kopf 1; 15 Blöcke ab Adresse 0) .....	39
9.9	Enhanced Write Words: (Kopf 1; 15 Blöcke ab Adresse 0).....	40
9.10	Single Write Fixcode: (Kopf 1; IPC11) .....	41
9.11	Single Write SpecialFixcode: (Kopf 1).....	42
9.12	Error Handling: (Kopf 1).....	43



9.13	Befehlsliste (Prefetch): (Kopf 1) .....	43
10	Tabelle Datenträger .....	47
11	Tabelle Statuswerte .....	49
12	Tabelle Versionsmeldung .....	50

## 1 **Einstellung Kommunikationsparameter über Webinterface**

Im Auslieferungszustand der IC-KP-B12-V45 sind folgende Kommunikationsparameter vordefiniert:

IP-Adresse:	172.16.177.0
Gateway-Adresse:	172.16.11.222
Subnetz-Maske:	255.255.0.0
Profinetname:	PF-IDENTControl

Im Auslieferungszustand der IC-KP-B17-AIDA1 sind folgende Kommunikationsparameter vordefiniert:

IP-Adresse:	169.254.10.12
Gateway-Adresse:	169.254.254.1
Subnetz-Maske:	255.255.0.0
Profinetname:	IDENTControl

Die Kommunikationsparameter können durch Direktbedienung am Gerät verändert werden. Hierzu müssen folgende Menüpunkte durchlaufen werden: IDENTControl → IDENT Gateway → Einstellung Netzwerk → IP-Adresse ( Subnet-Mask; Standart-Gateway)

Zur Aktivierung der neu eingestellten Parameter zu übernehmen, ist ein Neustart durchzuführen.

Des Weiteren besteht die Möglichkeit die Kommunikationsparameter über die geräteinterne Webseite einzustellen. Die Webseite kann über einen Internetbrowser und der voreingestellten IP-Adresse aufgerufen werden.

PEPPERL+FUCHS  
SENSING YOUR NEEDS

IC-KP-B17-AIDA1

Home

Contents

- Home
- Network
- Email
- Security
- Send Command
- Data Logging
- Documentation
- Contact

Device	Information
Communication Module	Version: © P+F IDENT IC-KP-B17-AIDA1 #213244 1831785 3.04.13
RFID CH1	Version: IUH-F190-V1-EU #230471 1831818 31.07.13 Operation: NOT ACTIVE Connect: CONNECTED TAG-Type: 80
RFID CH2	Version: Operation: NOT ACTIVE Connect: NOT CONNECTED TAG-Type: 80
RFID CH3	Version: Operation: NOT ACTIVE Connect: NOT CONNECTED TAG-Type: 80
RFID CH4	Version: Operation: NOT ACTIVE Connect: NOT CONNECTED

Unter der Auswahl „Network“ können die Kommunikationsparameter eingestellt werden.

PEPPERL+FUCHS  
SENSING YOUR NEEDS

IC-KP-B17-AIDA1

Network

Home → Network

Contents

- Home
- Network
- Email
- Security
- Send Command
- Data Logging
- Documentation
- Contact

MAC address: 00:0D:81:01:79:87

use DHCP:

IP address: 172.24.55.164

subnet mask: 255.255.255.192

gateway address: 172.24.55.190

duplex mode: auto detect

datahold time: 10ms x 2

profinet devicename: identcontrol

ethernet/ip instance: output:100 / input:150

save & reset cancel

Legal Notice | © 2009 All Rights Reserved.

Die Parametrierung wird durch „Save & Reset“ in das Gerät übernommen. Anschließend führt das Gerät automatisch einen Restart durch.

Der Profinetname kann mit Hilfe der Steuerung vergeben werden. Dazu innerhalb der Hardwarekonfiguration folgendes anwählen: Zielsystem → Ethernet → Ethernet-Teilnehmer bearbeiten

The screenshot shows a dialog box titled "Ethernet-Teilnehmer bearbeiten". It contains the following sections and controls:

- Ethernet Teilnehmer:** A "MAC-Adresse:" text box followed by a "Durchsuchen..." button. A label "Online erreichbare Teilnehmer" is positioned to the right.
- IP-Konfiguration einstellen:** A radio button labeled "IP-Parameter verwenden" is selected. To its right is a "Netzübergang" section with two radio buttons: "Keinen Router verwenden" (selected) and "Router verwenden". Below "Router verwenden" is an "Adresse:" text box.
- IP-Adresse von einem DHCP-Server beziehen:** A radio button labeled "IP-Adresse von einem DHCP-Server beziehen" is unselected. Below it is the text "identifiziert über" followed by three radio buttons: "Client-ID" (selected), "MAC-Adresse", and "Gerätename". A "Client-ID:" text box is located below these options.
- Gerätename vergeben:** A "Gerätename:" text box followed by a "Name zuweisen" button.
- Rücksetzen auf Werkseinstellungen:** A "Zurücksetzen" button.
- Buttons:** "Schließen" and "Hilfe" buttons are located at the bottom of the dialog.

Hier kann zunächst mit „Durchsuchen“ im Profinet-IO System nach angeschlossenen Teilnehmern gesucht werden. Wenn mehrere Profinetteilnehmer vorhanden sind, können diese durch die Funktion „Blinken“ identifiziert werden. Anschließend ist die IP-Konfiguration einzustellen. Dabei kann festgelegt werden, ob die IP-Adresse fest vergeben wird, oder ob die IP-Adresse dynamisch von einem DHCP-Server bezogen wird. Abschließend ist ein eindeutiger Profinetname für das Gerät zuzuweisen.

## 2 Installation GSDML-Datei

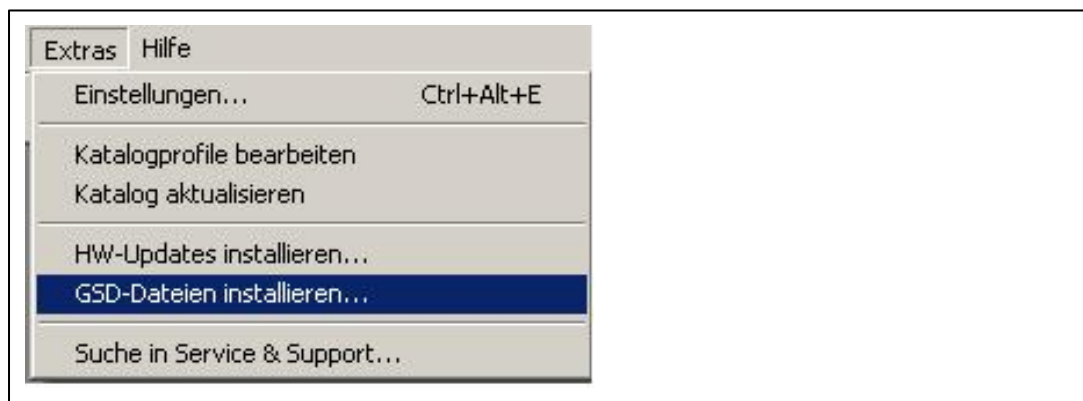
Vor der ersten Inbetriebnahme des Identifikationssystems IDENTControl ist zunächst die GSDML-Datei zu installieren. Die GSDML-Datei kann der dem Produkt beiliegenden CD „Identifikationssysteme“ entnommen werden. Alternativ kann die Datei via Internet heruntergeladen werden.

[www.pepperl-fuchs.com](http://www.pepperl-fuchs.com)

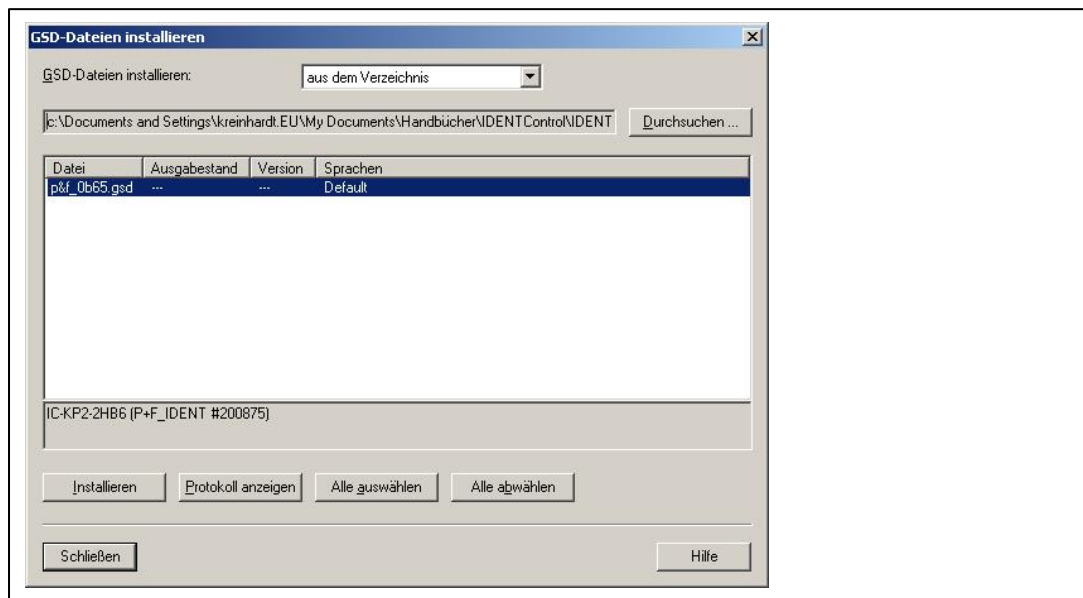
(Produktsuche → IC-KP-B12-V45 → technische Dokumente → 1830920.zip)

(Produktsuche → IC-KP-B17-AIDA1 → technische Dokumente → 1831801.zip)

Dazu muss innerhalb der Simatic Hardwarekonfiguration der Menüpunkt „Extras“ → „GSD-Datei installieren...“ aufgerufen werden.



Anschließend die GSDML-Datei aus dem Quellverzeichnis auswählen.

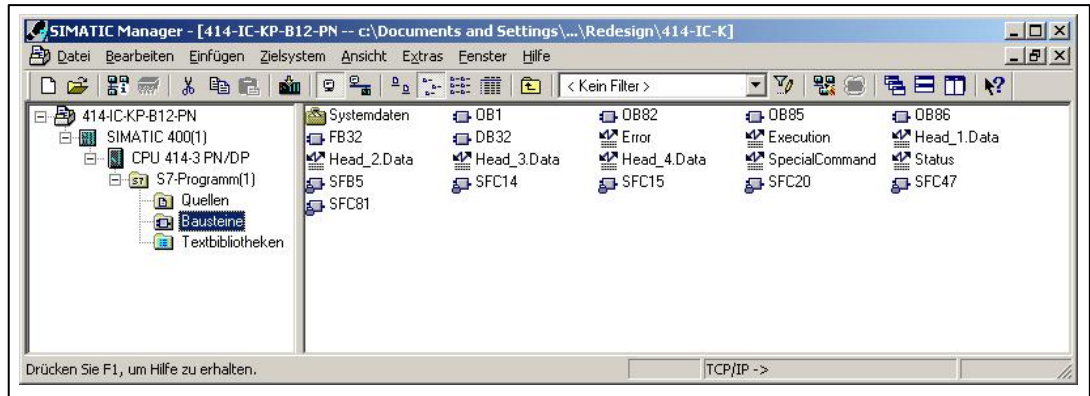


Die GSDML-Datei wird durch den Menüpunkt „Extras“ → „Katalog aktualisieren“ in den Hardwarekatalog übernommen.



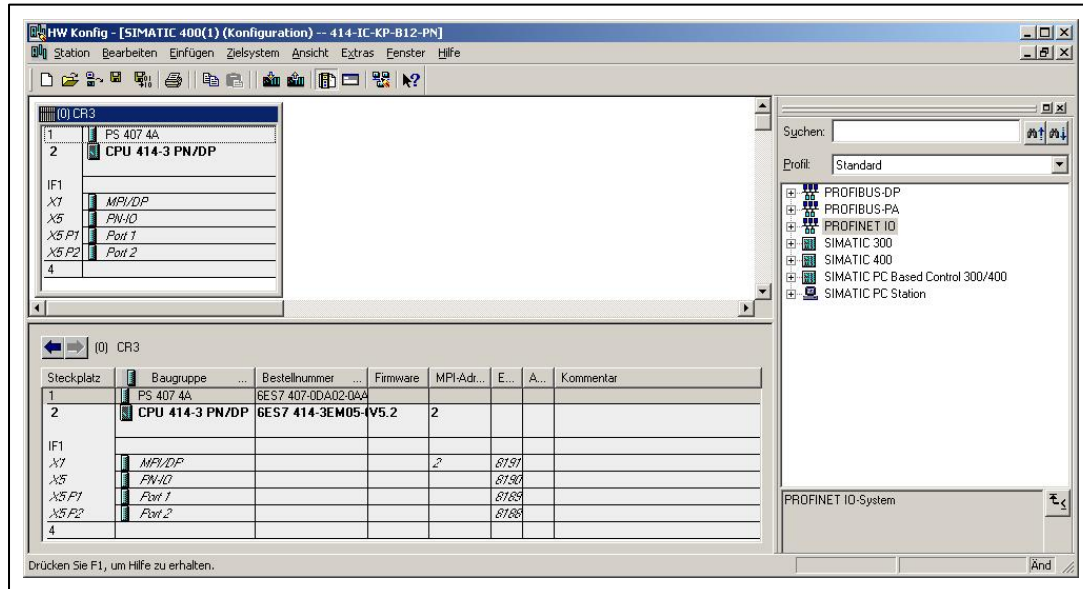
### 3 Installation Anwenderprogramm

Für die Installation des Anwenderprogramms muss zunächst die Datei „IC-KP-B12\_PN.zip“ entpackt werden. Dazu innerhalb des SIMATIC Managers den Menüpunkt „Datei“ → „Dearchivieren...“ anwählen. Anschließend die Datei markieren und über „Öffnen“ bestätigen sowie den zugehörigen Ablagepfad bestimmen. Nach erfolgreich durchgeführter Installation erscheint das Anwenderprogramm innerhalb des SIMATIC Managers.

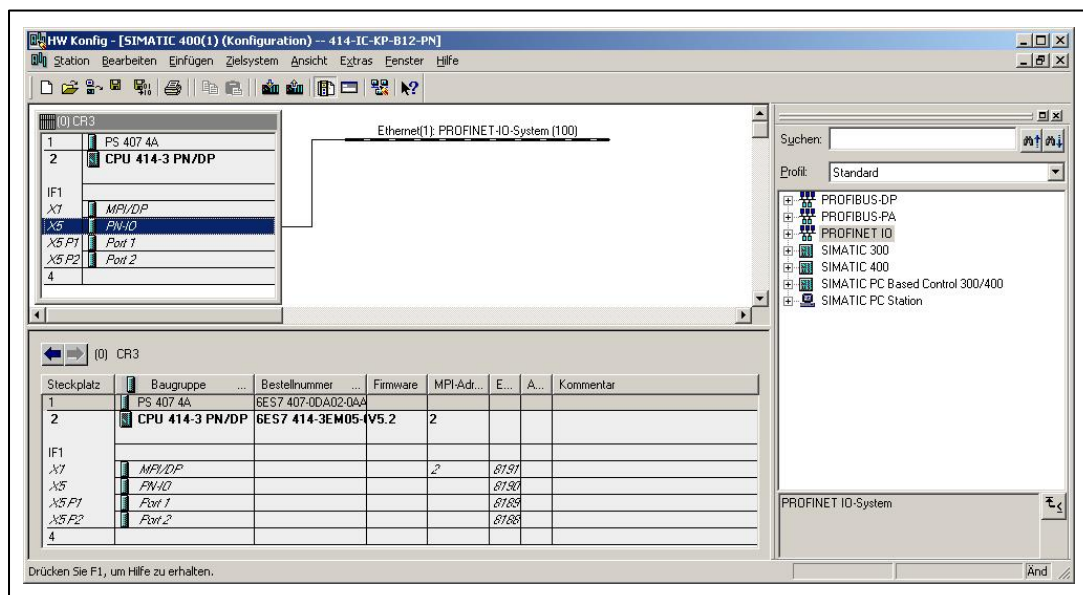


## 4 Hardwarekonfiguration

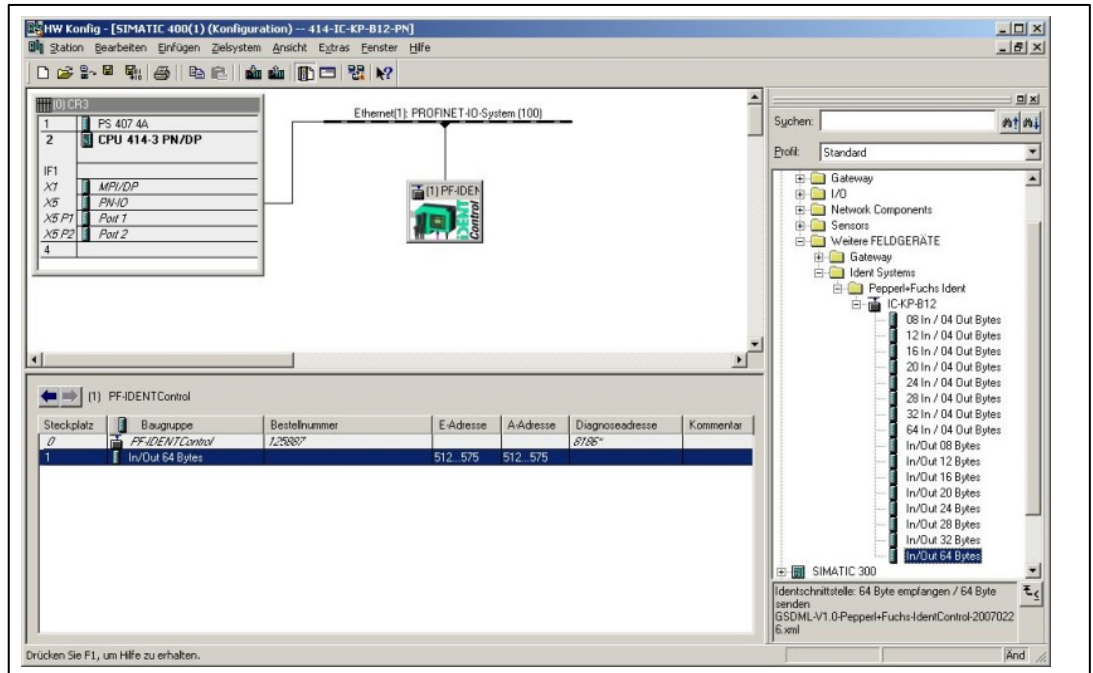
Innerhalb der Hardwarekonfiguration müssen die vorhandenen Baugruppen parametrieren werden.



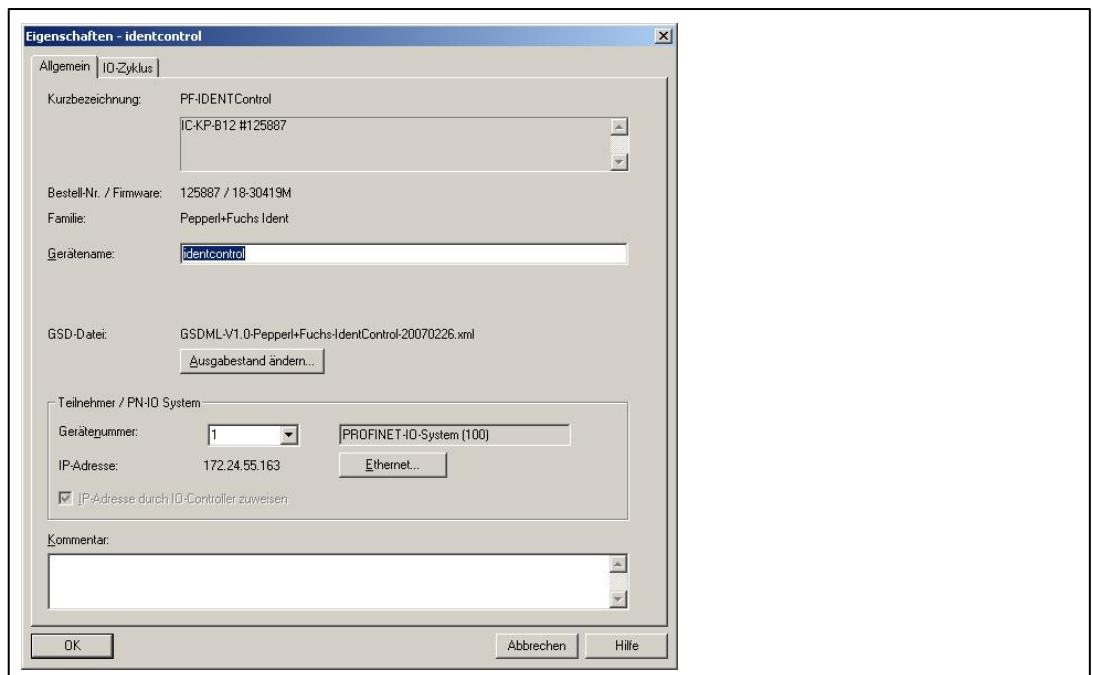
Anschließend wird durch Rechtsklick auf den Profinetport ein Profinet-IO System eingefügt werden.



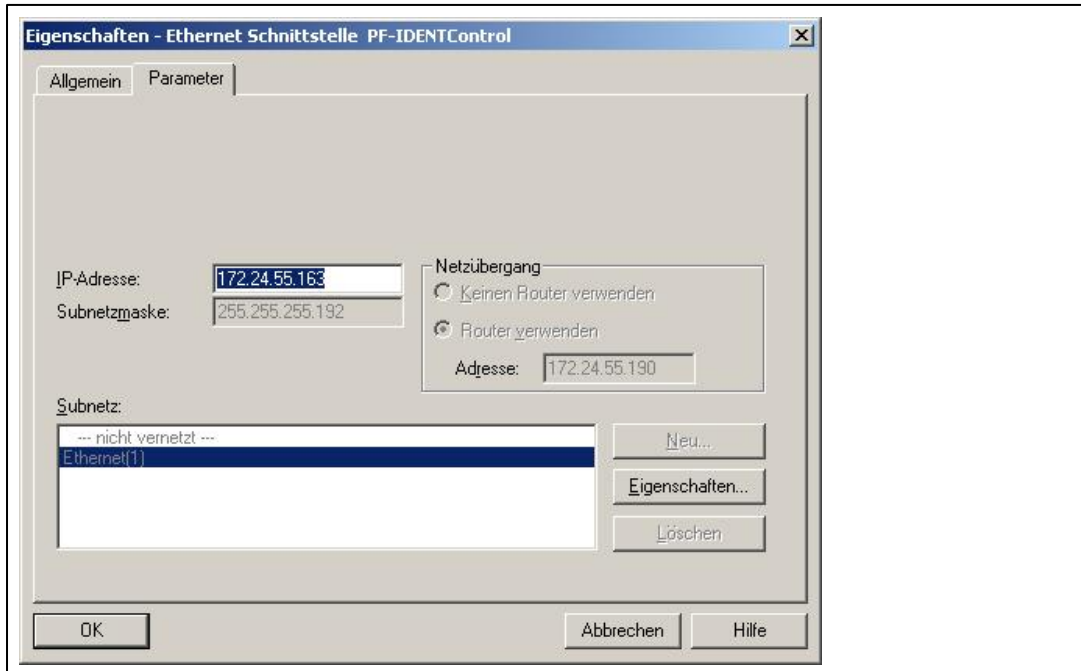
Anschließend erfolgt die Anbindung der IDENTControl an das Profinet-IO System. Dazu muss das Symbol „IC-KP-B12“ (bzw. „IC-KP-B17“) aus den Hardwarekatalog via „Drag n´Drop“ auf das Mastersystem gezogen werden. Anschließend wird die Telegrammlänge durch die Kommunikationsmodule (z.B. In/Out 64 Byte) definiert. Das Modul muss einen Slot zugewiesen werden.



Durch Doppelklick auf das Symbol der IDENTControl wird ein Fenster mit den Eigenschaften des Profinetteilnehmers aufgerufen. Hier kann ggf. der Name des Profinetteilnehmers (hier: identcontrol) verändert werden.



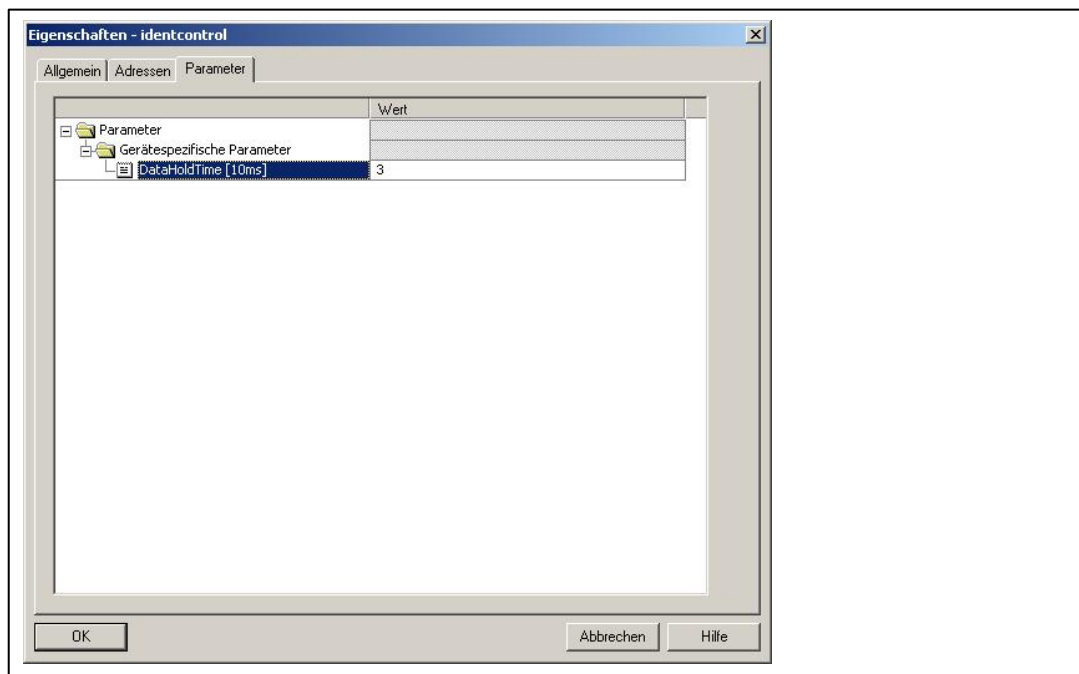
Mit Hilfe der Auswahl „Ethernet“ kann die IP-Adresse des Teilnehmers verändert werden.



Sollte der Funktionsbaustein in ein anderes Applikationsprogramm übertragen werden, so empfiehlt es sich die Symboltabelle ebenfalls in das neue Programm zu kopieren. Dadurch wird die Arbeit mit den Symbolinformationen unterstützt.

## 5 Einstellung Geräteparameter

Die Geräteparameter können durch einen Doppelclick auf den Steckplatz 0 der E/A-Tabelle aufgerufen werden.



Der gerätespezifische Parameter „Data Hold Time“ (DHT) beschreibt die Datenhaltezeit innerhalb des Ausgangsdatenfeldes der IDENTControl. Die DHT sollte den doppelten Wert der Zykluszeit der Steuerung aufweisen. Hierbei gilt zu beachten, dass der Wert der DHT ein Vielfaches von 10ms darstellt. Beträgt die Zykluszeit innerhalb der Steuerung rund 20ms, so sollte ein Wert der DHT von mindestens 4 gewählt werden.

Anmerkung: Der Wert der DHT bei der erstmaligen Einbindung (d.h. Default-Wert) ist bei der IC-KP-B12-V45 = „0“. Für das Gerät IC-KP-B17-AIDA1 ist die Grundeinstellung der Wert „2“.

## 6 Funktionsbaustein „FB190\_IUHParam“

Der FB190 dient dem Parametrieren der Leseköpfe IUH-F190 und IUH-F117. Dabei erfolgt der Aufruf des Bausteins und des zugehörigen Instanzdatenbausteins durch:

Call „FB190\_IUHParam“, „DB190\_IUHParam“ (Symbolische Darstellung)

Bzw.

Call FB190, DB190

Zum Aktivieren des Funktionsbausteins muss „ParameterEnable“ aktiv sein. Ist „NormalEnable“ aktiv muss dieser zunächst zurückgesetzt werden.

Dieser Baustein unterstützt Auswerteeinheiten mit bis zu zwei Kanälen.

Nachfolgendes Bild zeigt den Aufruf der Funktion und die zu parametrierenden Variablen.

```
CALL "FB190_IUHParam" , "DB190_IUHParam"
HeadNumber          := "HeadxNumber"
ReadWriteParameter  := "ReadWriteParameter"
PowerTransmit       := "Power Transmit"
TriesAllowed        := "TriesAllowed"
ChannelDenseReaderMode := "Channel Dense Reader"
ProtocolMode        := "Protocol Mode"
Information         := "More Information"
QValue             := "Q-Value"
NumberOfTags       := "Number of Tags"
SensingMode        := "Sensing Mode"
MemoryBank         := "Memory Bank"
MeasureReflection   := "Measure Reflection"
AdditionalInformation := FALSE //Input not supported in multiframe-mode
ResetToDefault     := "Reset to Default"
FilterList         := "Filter List"
EnhancedStatus5    := "Enhanced Status 5"
AntennaPolarisation := "Antenna Polarisation"
ParamFinished      := "ParamFinished"
ParamError         := "ParamError"
ParamTypeError     := "ParamTypeError"
ParamBusy         := "ParamBusy"
ParamStart        := "ParamStart"
```

Name	Deklaration	Datentyp	Beschreibung
HeadNumber	Input	BYTE	Kopf/Kanalnummer an den Parameter gelesen/geschrieben werden sollen
ReadWriteParameter	Input	BOOL	Ausführung von 0:= lesen 1:=schreiben der Parameter
PowerTransmit	Input	BOOL	Zugriff auf Sendeleistung des Lesekopfes
TriesAllowed	Input	BOOL	Zugriff auf Schreib-/Leseversuche
ChannelDenseReaderMode	Input	BOOL	Zugriff auf Reihenfolge der im Dense Reader Mode (= DRM) erlaubten Sendekanäle
ProtocolMode	Input	BOOL	Zugriff auf Ausgabeprotokoll (Singleframe und Multiframe)
Information	Input	BOOL	Zugriff auf Ausgabe zusätzlicher Informationen, sofern die Lesung erfolgreich war und das Multiframe-Protokoll gesetzt ist
QValue	Input	BOOL	Zugriff auf Q-Wert zur Festlegung der genutzten Zeitschlitze (2Q) für Antikollision (Slotted-Aloha)

NumberofTags	Input	BOOL	Zugriff auf Anzahl an Transpondern im Erfassungsbereich, die der Schreib-/Lesekopf sucht
SensingMode	Input	BOOL	Zugriff auf Pausenzeit in ms, nach Gesamtheit aller Durchläufe
MemoryBank	Input	BOOL	Zugriff auf Bank, auf den die Schreib-/Lesebefehle SR,ER, SW und EW zugreifen
MeasureReflection	Input	BOOL	Zugriff auf gemessene reflektierte Sendeleistung (nur lesend)
AdditionalInformation	Input	BOOL	Zugriff auf Ausgabe zusätzlicher Informationen, sofern die Lesung erfolgreich war und das Singleframe-Protokoll gesetzt ist (nur lesbar) –nicht im Multiframe-Betrieb!
Reset to Default	Input	BOOL	Setzt alle Einstellungen des Schreib-/Lesekopfs auf seine Defaultkonfiguration zurück
Filterlist	Input	BOOL	Schreibt die in OB1 definierte Filtermaske für Zugriff auf die im Feld befindlichen Transponder
Antenna Polarisation	Input	BOOL	Schaltet die Polarisation auf linear bzw. zirkular
Enhanced Status 5	Input	BOOL	Setzt die Anzahl der erfolglosen Schreib-/Leseversuche, bis bei einem enhanced-Befehl ein Status 5 ausgegeben wird
ParamStart	Input	BOOL	Ausführung Parameter-Befehl (Flanke positiv)
ParamBusy	Output	BOOL	Einzelner Parameterbefehl wird zurzeit ausgeführt
ParamFinished	Output	BOOL	Parameterzugriff beendet
ParamError	Output	BOOL	Fehler ist aufgetreten
ParamTypeError	Output	WORD	Rückgabe des fehlerhaften Parametertyps (HEX)

Es können sowohl einzelne oder mehrere/alle Parameter auf einmal ausgelesen oder geschrieben werden.

Ausgelesene Parameter werden im Instanz-Datenbaustein DB190 in dafür vorgesehene Strukturen gespeichert. Am Beispiel des Parameters „Power Transmit“ (PT) soll dies verdeutlicht werden.

Operand	Symbol	Anzeigeformat	Statuswert	Steuerwert
1	M 13.0 "ParameterEnable"	BOOL	true	true
2	M 13.1 "NormalEnable"	BOOL	false	false
3				
4	M 9.0 "Start"	BOOL	false	true
5	M 11.0 "Finished"	BOOL	true	
6	M 11.2 "ErrorParam"	BOOL	false	
7	MW 14 "ErrorCMD"	HEX	W#16#0000	
8				
9	MB 12 "HeadxNumber"	HEX	B#16#01	B#16#01
10	M 9.1 "ReadWriteParameter"	BOOL	false	
11				
12	M 9.2 "Power Transmit"	BOOL	true	true
13	M 9.3 "TriesAllowed"	BOOL	false	
14	M 9.4 "Channel Dense Reader"	BOOL	false	
15	M 9.5 "Protocol Mode"	BOOL	false	
16	M 9.6 "More Information"	BOOL	false	
17	M 9.7 "Q-Value"	BOOL	false	
18	M 10.0 "Number of Tags"	BOOL	false	
19	M 10.1 "Sensing Mode"	BOOL	false	
20	M 10.2 "Memory Bank"	BOOL	false	
21	M 10.3 "Measure Reflection"	BOOL	false	
22	M 10.5 "Reset to Default"	BOOL	false	
23	M 10.6 "Filter List"	BOOL	false	
24				

Ausgelesene Daten in DB190:

PT.Read.Length := W#16#4

PT.Read.PT1 := W#16#14 (1. Sendeleistung)

PT.Read.PT2 := W#16#64 (2. Sendeleistung)

20	10.0	stat	PT.Read.Length	WORD	W#16#0	W#16#0004
21	12.0	stat	PT.Read.PT1	WORD	W#16#0	W#16#0014
22	14.0	stat	PT.Read.PT2	WORD	W#16#0	W#16#0064
23	16.0	stat	PT.Read.PT3	WORD	W#16#0	W#16#0000
24	18.0	stat	PT.Read.PT4	WORD	W#16#0	W#16#0000
25	20.0	stat	PT.Read.PT5	WORD	W#16#0	W#16#0000
26	22.0	stat	PT.Read.PT6	WORD	W#16#0	W#16#0000
27	24.0	stat	PT.Read.PT7	WORD	W#16#0	W#16#0000
28	26.0	stat	PT.Read.PT8	WORD	W#16#0	W#16#0000
29	28.0	stat	PT.Read.PT9	WORD	W#16#0	W#16#0000
30	30.0	stat	PT.Read.PT10	WORD	W#16#0	W#16#0000

Die Struktur setzt sich zusammen aus: Abkürzung des Parametertyps + lesen/schreiben + Parameter

Zum Schreiben der Parameter existiert in OB1 das Netzwerk 2. In diesem sind die gewünschten Werte in den Code zu schreiben. Je nach gesetzten Eingängen werden nur die ausgewählten Parameter geschrieben.

**Die korrekte Notation und passenden Wertebereiche sind dem beigefügten Netzwerkkommentar zu entnehmen (Netzwerk 2: Init Parameter).**

Ist ein Parameter falsch konfiguriert, wird dessen Parametertyp nach der Übertragung angezeigt.



	Operand	Symbol	Anzeigeformat	Statuswert	Steuerwert
1	M 13.0	"ParameterEnable"	BOOL	true	true
2	M 13.1	"NormalEnable"	BOOL	false	false
3					
4	M 9.0	"Start"	BOOL	false	true
5	M 11.0	"Finished"	BOOL	true	
6	M 11.2	"ErrorParam"	BOOL	false	
7	MW 14	"ErrorCMD"	HEX	W#16#5441	
8					
9	MB 12	"HeadxNumber"	HEX	B#16#01	
10	M 9.1	"ReadWriteParameter"	BOOL	true	
11					
12	M 9.2	"Power Transmit"	BOOL	true	
13	M 9.3	"TriesAllowed"	BOOL	true	
14	M 9.4	"Channel Dense Reader"	BOOL	false	
15	M 9.5	"Protocol Mode"	BOOL	true	
16	M 9.6	"More Information"	BOOL	false	
17	M 9.7	"Q-Value"	BOOL	false	
18	M 10.0	"Number of Tags"	BOOL	true	
19	M 10.1	"Sensing Mode"	BOOL	false	
20	M 10.2	"Memory Bank"	BOOL	false	
21	M 10.3	"Measure Reflection"	BOOL	false	
22					
23					

Die Abbildung zeigt das Schreiben der Parameter: „Power Transmit“ (PT), „Tries Allowed“ (TA), „Protocol Mode“ (QV) und „Number of Tags“ (NT) an Kanal 1. Es liegt eine Fehlkonfiguration des Parameters „Tries Allowed“ vor. Erkennbar am Rückgabewert von „ErrorCMD“ := W#16#5441. Der Ausgang enthält den hexadezimalen Parametertyp des fehlerhaften Parameters (5441h = TAASCII).

Sollen Parameter geschrieben werden und am Baustein sind zudem noch nur zu lesende Parametertypen gesetzt (siehe Handbuch F190), ignoriert der Funktionsbaustein diese Eingänge. Ebenso verhält es sich beim Lesen.

Im Auslieferungszustand des Bausteins sind alle Parameter in OB1 mit Default-Werten belegt.

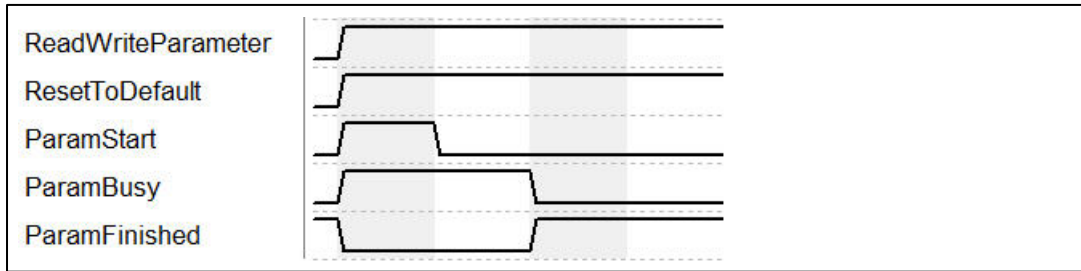
### Beispiele Befehlsausführungen:

#### 6.1 Reset to Default: Kanal 1

Festlegung der Parameter:  
HeadNumber := B#16#01  
ReadWriteParameter := 1  
ResetToDefault := 1

Start der Ausführung:  
Start := 1 (positive Flanke)  
ParamBusy := 1

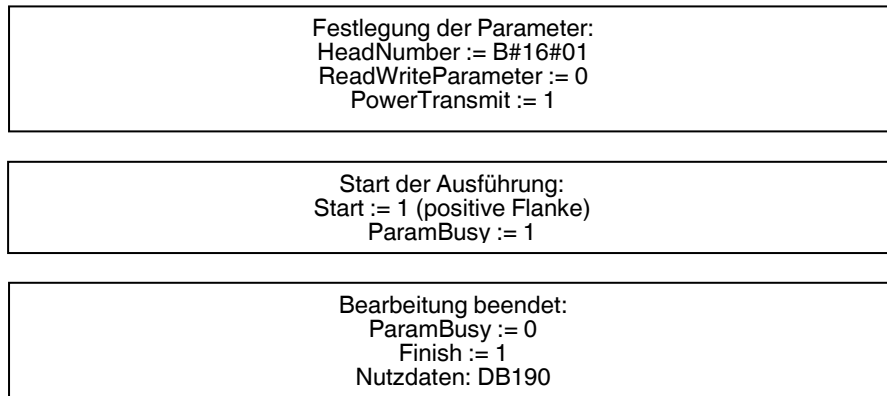
Bearbeitung beendet:  
ParamBusy := 0  
Finish := 1



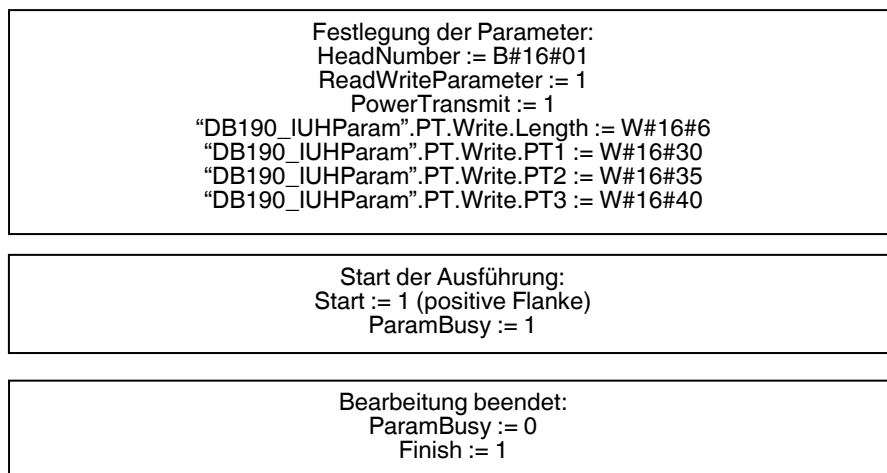
Nachdem der Befehl mit „ParamStart“ angestoßen wurde signalisiert „ParamBusy“, dass die Befehlsabarbeitung im Gange ist. „ParamFinished“ wechselt während dieser Zeit auf FALSE. Nach dem der Befehl beendet ist geht „ParamFinished“ wieder auf TRUE.

## 6.2 PowerTransmit - PT: Kanal 1

Lesen:



Schreiben: 3 Sendeleistungen parametrieren



### 6.3 ChannelDenseReaderMode - CD: Kanal 1

Lesen:

Festlegung der Parameter:  
HeadNumber := B#16#01  
ReadWriteParameter := 0  
ChannelDenseReaderMode := 1

Start der Ausführung:  
Start := 1 (positive Flanke)  
ParamBusy := 1

Bearbeitung beendet:  
ParamBusy := 0  
Finish := 1  
Nutzdaten: DB190

Schreiben: Kanal A & Kanal B

Festlegung der Parameter:  
HeadNumber := B#16#01  
ReadWriteParameter := 1  
ChannelDenseReaderMode := 1  
"DB190\_IUHParam".CD.Write.Length := B#16#2  
"DB190\_IUHParam".CD.Write.ChannelA := B#16#D  
"DB190\_IUHParam".CD.Write.ChannelB := B#16#7

Start der Ausführung:  
Start := 1 (positive Flanke)

Bearbeitung beendet:  
Finish := 1

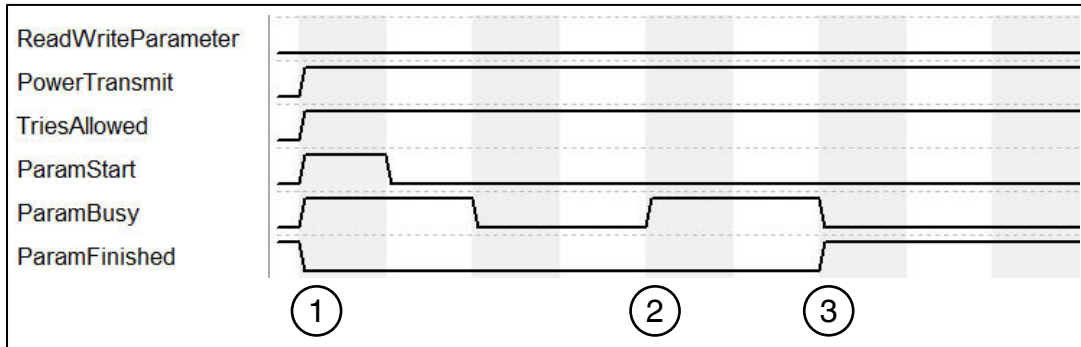
Mehrere Parameter lesen/schreiben: Kanal 1

Lesen:

Festlegung der Parameter:  
HeadNumber := B#16#01  
ReadWriteParameter := 0  
PowerTransmit := 1  
TriesAllowed := 1

Start der Ausführung:  
Start := 1 (positive Flanke)  
ParamBusy := 1

Bearbeitung beendet:  
Finish := 1  
ParamBusy := 0  
Nutzdaten: DB190



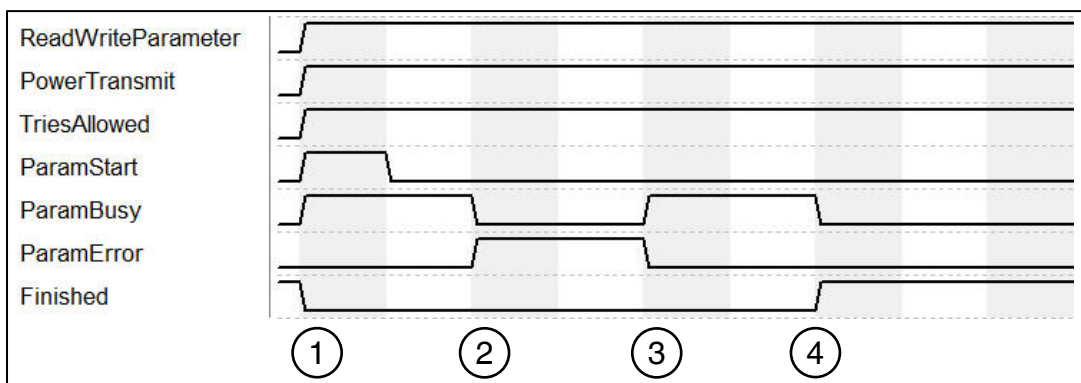
1. Befehlsausführung (Lesen) wird mit „ParamStart“ begonnen. „ParamFinished“ wechselt auf FALSE.
2. „ParamBusy“ signalisiert Befehlsabarbeitung von „Power Transmit“ und „Tries Allowed“.
3. Nachdem beide Parameter beendet sind wechselt „ParamFinished“ wieder auf TRUE.

Schreiben:

Festlegung der Parameter:  
 HeadNumber := B#16#01  
 ReadWriteParameter := 1  
 PowerTransmit := 1  
 ChannelDenseReaderMode := 1  
 NumberofTags := 1  
 "DB190\_IUHParam".PT.Write.Length := W#16#2  
 "DB190\_IUHParam".PT.Write.PT1 := W#16#30  
 "DB190\_IUHParam".CD.Write.Length := B#16#2  
 "DB190\_IUHParam".CD.Write.ChannelA := B#16#D  
 "DB190\_IUHParam".CD.Write.ChannelB := B#16#7  
 "DB190\_IUHParam".NT.Write.NT := B#16#FF

Start der Ausführung:  
 Start := 1 (positive Flanke)  
 ParamBusy := 1

Bearbeitung beendet:  
 ParamBusy := 0  
 Finish := 1



1. Befehlsausführung (Schreiben) wird mit „ParamStart“ begonnen. „ParamFinished“ wechselt auf FALSE.

2. „ParamBusy“ signalisiert Befehlsabarbeitung von „Power Transmit“ und „Tries Allowed“. Fehlkonfiguration von Parameter „PowerTransmit“. „ParamError“ geht auf TRUE.
3. „TriesAllowed“ wird ausgeführt. „ParamError“ wird zurückgesetzt. Ausgang „ParamtypeError“ zeigt fehlerhaften Parameter an
4. Alle Befehle bearbeitet. „ParamFinished“ wechselt wieder auf TRUE.

## 7 Funktionsbaustein FB32 „Multiframe“

Der Aufruf des Funktionsbausteins und des zugehörigen Instanzdatenbausteins erfolgt durch:

Call „IDENTControl“, „InstDB“ (Symbolische Darstellung)

Bzw.

Call FB32, DB32

Zum Aktivieren muss „NormalEnable“ aktiv sein.

Dieser Baustein unterstützt Auswerteeinheiten mit bis zu zwei Kanälen.

Nachfolgendes Bild zeigt den Aufruf der Funktion und die zu parametrierenden Variablen

```
CALL "IDENTControl" , "InstDB"  
IC_INPUT_Address :=W#16#200  
IC_OUTPUT_Address :=W#16#200  
Length_IN :=64  
Length_OUT :=64  
Timeout :=T#3S  
Head1DataFixcode :="Head1DataFixcode"  
Head2DataFixcode :="Head2DataFixcode"  
Head3DataFixcode :="Head3DataFixcode"  
Head4DataFixcode :="Head4DataFixcode"  
Head1SingleEnhanced:= "Head1SingleEnhanced"  
Head2SingleEnhanced:= "Head2SingleEnhanced"  
Head3SingleEnhanced:= "Head3SingleEnhanced"  
Head4SingleEnhanced:= "Head4SingleEnhanced"  
Head1SpecialCommand:= "Head1SpecialCommand"  
Head2SpecialCommand:= "Head2SpecialCommand"  
Head3SpecialCommand:= "Head3SpecialCommand"  
Head4SpecialCommand:= "Head4SpecialCommand"  
Head1Read := "Head1Read"  
Head2Read := "Head2Read"  
Head3Read := "Head3Read"  
Head4Read := "Head4Read"  
Head1Write := "Head1Write"  
Head2Write := "Head2Write"  
Head3Write := "Head3Write"  
Head4Write := "Head4Write"  
Head1Quit := "Head1Quit"  
Head2Quit := "Head2Quit"  
Head3Quit := "Head3Quit"  
Head4Quit := "Head4Quit"  
QuitErrorHead1 := "QuitErrorHead1"  
QuitErrorHead2 := "QuitErrorHead2"  
QuitErrorHead3 := "QuitErrorHead3"  
QuitErrorHead4 := "QuitErrorHead4"  
IC_Command_on_Head1:= "IC_Command"  
Head1WordAddress :=W#16#0  
Head2WordAddress :=W#16#0  
Head3WordAddress :=W#16#0  
Head4WordAddress :=W#16#0  
Head1TagType :=W#16#3830  
Head2TagType :=W#16#3830  
Head3TagType :=W#16#3830  
Head4TagType :=W#16#3830  
Head1SpecialFixcode:= "Head1SpecialFixcode"  
Head2SpecialFixcode:= "Head2SpecialFixcode"  
Head3SpecialFixcode:= "Head3SpecialFixcode"  
Head4SpecialFixcode:= "Head4SpecialFixcode"  
Head1CacheSize :=10  
Head2CacheSize :=10  
Head3CacheSize :=10  
Head4CacheSize :=10  
Head1SetFilter := "Head1SetFilter"  
Head2SetFilter := "Head2SetFilter"  
Head3SetFilter := "Head3SetFilter"  
Head4SetFilter := "Head4SetFilter"  
Head1FilterON := "Head1FilterON"  
Head2FilterON := "Head2FilterON"  
Head3FilterON := "Head3FilterON"  
Head4FilterON := "Head4FilterON"  
Head1Done := "Head1Done"  
Head2Done := "Head2Done"  
Head3Done := "Head3Done"  
Head4Done := "Head4Done"  
Head1NoDataCarrier := "Head1NoDataCarrier"  
Head2NoDataCarrier := "Head2NoDataCarrier"  
Head3NoDataCarrier := "Head3NoDataCarrier"  
Head4NoDataCarrier := "Head4NoDataCarrier"  
Head1Error := "Head1Error"  
Head2Error := "Head2Error"  
Head3Error := "Head3Error"  
Head4Error := "Head4Error"  
Head1Busy := "Head1Busy"  
Head2Busy := "Head2Busy"  
Head3Busy := "Head3Busy"  
Head4Busy := "Head4Busy"  
Head1Status := "Head1Status"  
Head2Status := "Head2Status"  
Head3Status := "Head3Status"  
Head4Status := "Head4Status"  
Head1ReplyCounter := "Head1ReplyCounter"  
Head2ReplyCounter := "Head2ReplyCounter"  
Head3ReplyCounter := "Head3ReplyCounter"  
Head4ReplyCounter := "Head4ReplyCounter"  
Head1CacheFull := "Head1CacheFull"  
Head2CacheFull := "Head2CacheFull"  
Head3CacheFull := "Head3CacheFull"  
Head4CacheFull := "Head4CacheFull"  
InitFinish := "InitFinish"  
SetRestart := "SetRestart"  
Head1NewData := "Head1NewData"  
Head2NewData := "Head2NewData"  
Head3NewData := "Head3NewData"  
Head4NewData := "Head4NewData"
```

Name	Deklaration	Datentyp	Beschreibung
IC_INPUT_Address	Input	WORD	Startadresse der Auswerteeinheit im Prozessabbild der Eingänge (E-Adresse)
IC_OUTPUT_Address	Input	WORD	Startadresse der Auswerteeinheit im Prozessabbild der Ausgänge (A-Adresse)
Length_IN	Input	INT	Länge des Eingangsabbildes (Länge des empfangenen Profibustelegramms)
Length_OUT	Input	INT	Länge des Ausgangsabbildes (Länge des gesendeten Profibustelegramms)
Timeout	Input	TIME	Timer zur Überwachung der Antwortzeit der Auswerteeinheit
Head1/2/3/4DataFixcode	Input	BOOL	Zugriff Kopf 1/2/3/4 auf 0:=Fixcode 1:=Datenbereich
Head1/2/3/4SingleEnhanced	Input	BOOL	Ausführung an Kopf 1/2/3/4 von 0:=Single 1:=Enhanced Befehl
Head1/2/3/4SpecialCommand	Input	BOOL	Ausführung SpecialCommand an Kopf 1/2/3/4 (Flanke positiv); Befehlsparameter vorher in Struktur Head_X.SpecialCommand festlegen; empfangene Nutzdaten befinden sich in der Struktur Head_X.InData
Head1/2/3/4Read	Input	BOOL	Ausführung Lese Befehl an Kopf 1/2/3/4 (Flanke positiv); Befehlsparameter HeadXWordNum und HeadXWordaddress festlegen; empfangene Nutzdaten befinden sich in der Struktur Head_X.InData
Head1/2/3/4Write	Input	BOOL	Ausführung Schreib Befehl an Kopf 1/2/3/4 (Flanke positiv); Befehlsparameter HeadXWordNum und HeadXWordaddress festlegen; zuschreibende Nutzdaten vorher in der Struktur Head_X.OutData.DW1...15 festlegen
Head1/2/3/4Quit	Input	BOOL	Ausführung Quit Befehl an Kopf 1/2/3/4 zum Abbruch eines Enhanced Befehls (Flanke positiv)
QuitErrorHead1/2/3/4	Input	BOOL	Ausführung Fehleroutine an Kopf 1/2/3/4 (Flanke positiv)
IC_Command_Head1	Input	BOOL	Ausführung eines SpecialCommands an die Auswerteeinheit (Flanke positiv); Befehl wird an Kanal 0 gesendet; Befehlsparameter vorher in Struktur Head_X.SpecialCommand festlegen; empfangene Nutzdaten befinden sich in der Struktur Head_X.InData
Head1/2/3/4WordNum	Input	INT	Anzahl der an Kopf 1/2/3/4 angesprochenen Datenblöcke
Head1/2/3/4WordAddress	Input	WORD	Startadresse des an Kanal 1/2/3/4 angesprochenen Speicherbereichs auf Datenträger
Head1/2/3/4TagType	Input	WORD	Datenträgertyp Kanal 1/2/3/4 (Tabelle Datenträgertypen)
Head1/2/3/4SpecialFixcode	Input	BOOL	Zugriff Kopf 1/2/3/4 auf 1:=SpecialFixcode (EPC)
Head1/2/3/4CacheSize	Input	INT	Anzahl der an Kopf 1/2/3/4 zwischengespeicherten Telegramme
Head1/2/3/4FilterSettings	Input	BOOL	Initialisiert an Kopf 1/2/3/4 die in OB1 definierten



			Filterparameter →Head1Write
Head1/2/3/4FilterON	Input	BOOL	Aktiviert den/die gesetzten Filter an Kopf 1/2/3/4 (Flanke positiv) Deaktiviert den/die gesetzten Filter an Kopf 1/2/3/4 (Flanke negativ)
Head1/2/3/4Done	Output	BOOL	Neue Daten vorhanden (Enhanced) bzw. Befehl beendet (Single) an Kanal 1/2/3/4 (Flanke positiv)
Head1/2/3/4NoDataCarrier	Output	BOOL	Kein Datenträger zur Ausführungszeit eines Befehls an Kanal 1/2/3/4 vorhanden
Head1/2/3/4Error	Output	BOOL	Fehler ist aufgetreten an Kanal 1/2/3/4 (positive Flanke)
Head1/2/3/4Busy	Output	BOOL	Befehl wird an Kanal 1/2/3/4 bearbeitet
Head1/2/3/4Status	Output	BYTE	Statuswert Kanal 1/2/3/4
Head1/2/3/4ReplyCounter	Output	BYTE	Wert Antwortzähler Kanal 1/2/3/4
Head1/2/3/4CacheFull	Output	BOOL	Telegrammspeicher (DB) für Kopf 1/2/3/4 ist voll
InitFinish	In/Output	BOOL	Initialisierung beendet (Flanke positiv)
SetRestart	In/Output	BOOL	Ausführung Initialisierung Auswerteeinheit (Flanke positiv)
Head1/2/3/4NewData	In/Output	BOOL	Neue Daten im Telegrammspeicher (DB) vorhanden
Head1/2/3/4SetFilter	In/Output	BOOL	Initialisiert an Kopf 1/2/3/4 die in OB1 definierten Filterparameter →Head1/2/3/4Write
Head1/2/3/4FilterON	In/Output	BOOL	Aktiviert den/die gesetzten Filter an Kopf 1/2/3/4 (Flanke positiv) Deaktiviert den/die gesetzten Filter an Kopf 1/2/3/4 (Flanke negativ)

Beispiel für die Einstellung der Kommunikationsparameter:

In der Hardwarekonfiguration wurde das Kommunikationsmodul „64In / 64 Out Bytes“ ausgewählt. Das Prozessabbild der Eingänge (E-Adresse) beginnt bei Adresse 512, endet nach der Adresse 575 und hat eine Länge von 64 Byte. Das Prozessabbild der Ausgänge (A-Adresse) beginnt bei der Adresse 512 und endet nach der Adresse 575. Dadurch ergibt sich folgende Parametrierung für den Funktionsbaustein:

```

IC_INPUT_Address      :=W#16#200
IC_OUTPUT_Address     :=W#16#200
Length_IN             :=64
Length_OUT            :=64
    
```

Anmerkung:

Bei der Auswahl eines Kommunikationsmoduls für den reinen Lesebetrieb (z.B. „64 In / 4 Out Bytes“) muss die Ausgangsadresse („IC\_OUTPUT\_Address“) vollständig innerhalb des Prozessabbildes liegen. Bei CPUs der Baureihe S7-300 hat der A-Bereich eine Länge von 0 bis 256. Die Baureihe S7-400 hat ein Prozessabbild der Ausgänge mit einer Länge von 0 bis 512.

Bitte beachten Sie die maximal mögliche Telegrammlänge der verwendeten CPU. Die CPU S7-315-2DP ist für eine maximale Telegrammlänge von 32 Bytes ausgelegt. Die CPUs der Baureihe S7-400 sind für eine Telegrammlänge von 64 Bytes ausgelegt.

**Wichtig:**

Zum Speichern der eintreffenden Telegramme im Multiframe-Modus besitzt das Projekt vier zusätzliche Datenbausteine pro Kanal. Diese setzen sich aus den UDT'S 700-702 zusammen.

Folgende Datenbausteine und ihre Funktion:

- DB700: Speichert gelesene Words (Nutzerdaten), Fixcode (TID) und Special Fixcode (EPC). Zeigt nach einem Single-Befehl die Anzahl der gelesenen Tags an
- DB701: Bei eingeschaltetem IF-Parameter werden hier die zusätzlichen Informationen jedes Telegramms angezeigt
- DB703: Zeigt während dem Enhanced SpecialFixcode Befehl Tags mit zugehöriger EPC an, die den Erfassungsbereich verlassen haben

Sind Daten in einem Datenbaustein vorhanden wird das Bit „HeadXNewData“ gesetzt.

Dabei dient jeder Datenbaustein wie ein Puffer und speichert bis zu 10 Telegramme zwischen. Dieser Puffer kann durch Copy&Paste beliebig erweitert oder verkleinert werden. Es wird empfohlen die Erweiterung/Kürzung bei allen vier Datenbausteinen durchzuführen, um eine korrekte Konsistenz der Daten zueinander zu gewährleisten.

Erweiterung:

DB700:

Adresse	Name	Typ	Anfangswert	Kommentar
0.0		STRUCT		
+0.0	TransmissionFinished	STRUCT		
+0.0	NumberTagsInField	DWORD	DW#16#0	Only available for single commands
=4.0		END_STRUCT		
+4.0	Telegram1	"UDT700_Frame"		Telegram #1
+68.0	Telegram2	"UDT700_Frame"		Telegram #2
+132.0	Telegram3	"UDT700_Frame"		Telegram #3
+196.0	Telegram4	"UDT700_Frame"		Telegram #4
+260.0	Telegram5	"UDT700_Frame"		Telegram #5
+324.0	Telegram6	"UDT700_Frame"		Telegram #6
+388.0	Telegram7	"UDT700_Frame"		Telegram #7
+452.0	Telegram8	"UDT700_Frame"		Telegram #8
+516.0	Telegram9	"UDT700_Frame"		Telegram #9
+580.0	Telegram10	"UDT700_Frame"		Telegram #10
+644.0	Telegramm11	"UDT700_Frame"		Telegramm #11
=708.0		END_STRUCT		

DB701:

Adresse	Name	Typ	Anfangswert	Kommentar
0.0		STRUCT		
+0.0	Information1	"UDT701_FrameIF"		Additional Information Telegram #1
+6.0	Information2	"UDT701_FrameIF"		Additional Information Telegram #2
+12.0	Information3	"UDT701_FrameIF"		Additional Information Telegram #3
+18.0	Information4	"UDT701_FrameIF"		Additional Information Telegram #4
+24.0	Information5	"UDT701_FrameIF"		Additional Information Telegram #5
+30.0	Information6	"UDT701_FrameIF"		Additional Information Telegram #6
+36.0	Information7	"UDT701_FrameIF"		Additional Information Telegram #7
+42.0	Information8	"UDT701_FrameIF"		Additional Information Telegram #8
+48.0	Information9	"UDT701_FrameIF"		Additional Information Telegram #9
+54.0	Information10	"UDT701_FrameIF"		Additional Information Telegram #10
+60.0	Information11	"UDT701_FrameIF"		Additional Information Telegram #11
+66.0		END_STRUCT		

DB703:

Adresse	Name	Typ	Anfangswert	Kommentar
0.0		STRUCT		
+0.0	GoneTag1	"UDT702_EPC"		
+64.0	GoneTag2	"UDT702_EPC"		
+128.0	GoneTag3	"UDT702_EPC"		
+192.0	GoneTag4	"UDT702_EPC"		
+256.0	GoneTag5	"UDT702_EPC"		
+320.0	GoneTag6	"UDT702_EPC"		
+384.0	GoneTag7	"UDT702_EPC"		
+448.0	GoneTag8	"UDT702_EPC"		
+512.0	GoneTag9	"UDT702_EPC"		
+576.0	GoneTag10	"UDT702_EPC"		
+640.0	GoneTag11	"UDT702_EPC"		
+704.0		END_STRUCT		

Damit der Ausgang „HeadXCacheFull“ korrekt schaltet muss die Anzahl der Elemente zum Speichern der Telegramme an den Baustein übergeben werden.

In diesem Fall:

HeadXCache := 11

„HeadXCacheFull“ signalisiert, dass der Datenbaustein seine maximale Speicherkapazität erreicht hat und die nun folgenden Telegramme nicht mehr gespeichert werden. Zum Leeren des Puffers und Zurücksetzen des CachFull-Statusbits muss „HeadXNewData“ mit „0“ quittiert werden. Dies hat eine Löschung und Rücksetzung des Puffers zur Folge. Um Datenverluste zu vermeiden stellen Sie sicher, dass die benötigten Daten im Datenbaustein rechtzeitig rauskopiert und anschließend „HeadXNewData“ zurückgesetzt wird.

„SetRestart“ löscht ebenfalls den kompletten Inhalt der Datenbausteine.

Für Kanal 2 arbeiten analog DB710 bis DB713.

Ist eine andere Nummerierung der Datenbausteine erwünscht, so muss diese im OB1 im Netzwerk 4 kenntlich gemacht werden.

```
Netzwerk 4: Configure DB for incoming telegrams
When your DB-Number differs from default enlist here

//Channel 1
FB32: L 700
      T "InstDB".DBChan1_Multiframe      DB32.DBW908
      L 701
      T "InstDB".DBChan1_Information    DB32.DBW910
      L 703
      T "InstDB".DBChan1_GoneTag        DB32.DBW912
//Channel 2
      L 710
      T "InstDB".DBChan2_Multiframe      DB32.DBW914
      L 711
      T "InstDB".DBChan2_Information    DB32.DBW916
      L 713
      T "InstDB".DBChan2_GoneTag        DB32.DBW918
//Channel 3
      L 720
      T "InstDB".DBChan3_Multiframe      DB32.DBW920
      L 721
      T "InstDB".DBChan3_Information    DB32.DBW922
      L 723
      T "InstDB".DBChan3_GoneTag        DB32.DBW924
//Channel 4
      L 730
      T "InstDB".DBChan4_Multiframe      DB32.DBW926
      L 731
      T "InstDB".DBChan4_Information    DB32.DBW928
      L 733
      T "InstDB".DBChan4_GoneTag        DB32.DBW930
```

Beispiel für Kanal 1:

OB1 - Netzwerk 4:

L 1200

T „InstDB“.DBChan1\_Multiframe

L 1201

T „InstDB“.DBChan1\_Information

L 1203

T „InstDB“.DBChan1\_GoneTag

- „InstDB“.DBChan1\_Multiframe speichert Words (Nutzdaten), Fixcode oder Special Fixcode
- „InstDB“.DBChan1\_Information speichert zusätzliche Informationen , wenn eingeschalteter IF-Parameter
- „InstDB“.DBChan1\_GoneTag zeigt im Enhanced SpecialFixcode – Betrieb die EPC der Tags an, welche den Erfassungsbereich während der Lesung verlassen haben

## 7.1 Filterkonfiguration

Der Funktionsbaustein FB32 erlaubt das Erstellen und Setzen von Filtermasken um auf bestimmte im Erfassungsbereich befindliche Datenträger zuzugreifen. Zur Konfiguration des Filters müssen im OB1 (Netzwerk 5: Configure Filter) die gewünschte Einstellungen getroffen werden. Es können bis zu 3 Filter erstellt werden (Filternummer: 1- 3).

### Wertebereiche:

Filternummer	1, 2, 3
Memory Bank	1 (EPC/UII), 2 (TID), 3 (User Memory)
Negation	0 (nicht negiert), 1 (negiert)
LogicOP	0 (ODER-Verknüpfung), 1 (AND-Verknüpfung)
Truncation	Wert immer 0
MaskLength	Maskenlänge, 40h = 64dez (64Bit bzw. 8 Byte)
MaskData (UserData.DW2)	DW#16#30001122 (Beispiel)
MaskData (UserData.DW3)	DW#16#33445566 (Beispiel)

```

▣ Netzwerk 5: Configure Filter
Set up filter configuration in this network.
Range of values:

Filternummer: 1, 2, 3
Memory Bank: 1 (EPC/UII), 2 (TID), 3 (User Memory)
Negation: R (not active), S (active)
LogicOP: R (OR), S (AND)
Truncation: *not available yet*
MaskLength: e.g. 16#40 = 64 (64Bit/8Byte)
MaskData: 11.22.33.44.55.66.77.88

//Set up general Filter settings
O "Head1SetFilter"
O "Head2SetFilter"
O "Head3SetFilter"
O "Head4SetFilter"
SPBN efil

L 1
T "InstDB".Filternummer

L 1
T "InstDB".MemoryBank

R "InstDB".Negation
R "InstDB".LogicOP
// R "InstDB".Truncation

L B#16#40
T "InstDB".MaskLength

// Head 1 filtermask:
U "Head1SetFilter"
SPBN FIL2
L DW#16#30001122
T "InstDB".Head_1.OutData.UserData.DW2
L DW#16#33445566
T "InstDB".Head_1.OutData.UserData.DW3
    
```



Die Abbildung zeigt eine Filterkonfiguration für Kanal 1. Mit Festlegung der Filternummer 1, Filterung auf den EPC mit der Wertzuweisung 1 für die Memory Bank. Es erfolgt keine Negation auf die Maske und eine ODER-Verknüpfung der Filter (nur wirksam, wenn mehrere Filter aktiv). Die Maskenlänge beträgt 64Bit. Die Filtermaske, bestehend aus folgenden, 8 Bytes langem, EPC: 30.00.11.22.33.44.55.66

Beispiel: Filterparametrierung (Filter Nr.2)

Es wird auf Datenträger gefiltert, dessen Daten im „User Memory“ nicht der 5 Byte langen Maske (AA.BB.CC.DD.EE) entsprechen.



**Achtung!**

Achten Sie auf die korrekte Notation der Maskendaten.

Festlegung Filternummer:  
„InstDB“.Filternumber := 2  
Festlegung Speichersegment:  
„InstDB“.MemoryBank := 3 (User Memory)

Weitere Konfiguration:  
Negation := 1  
LogicOP := 0  
MaskLength := B#16#28 (5 Byte)  
MaskData := DW#16#AABBCCDD  
MaskData :=DW#16#EE000000 (Notation beachten!)

Filter setzen:  
Head1SetFilter := 1  
Head1Write := 1

Filter aktivieren:  
Head1FilterON := 1

Filter deaktivieren:  
Head1FilterON := 0

## 8 Fehler- bzw. Gerätediagnose

Bei der Inbetriebnahme des Bausteins (Initialisierung) treten die Mehrzahl der Fehler auf. Wird nach dem Start der Initialisierung (Flanke positiv auf SetRestart) das Bit InitFinish nicht automatisch gesetzt, so ist bei der Durchführung der Initialisierung ein Fehler aufgetreten. Hauptursache ist eine unterschiedliche Parametrierung der Hardwarekonfiguration innerhalb des Simatic Hardwaremanagers und der Parametrierung des E/A-Bereichs bzw. der Telegrammlänge am Funktionsbaustein. Es ist ebenfalls der eingestellte Datenträgertyp (Head1(2)TagType) zu überprüfen.

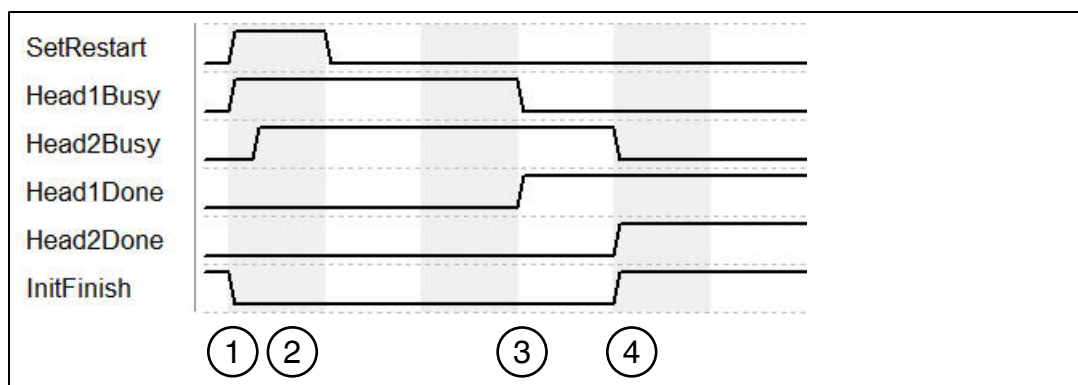
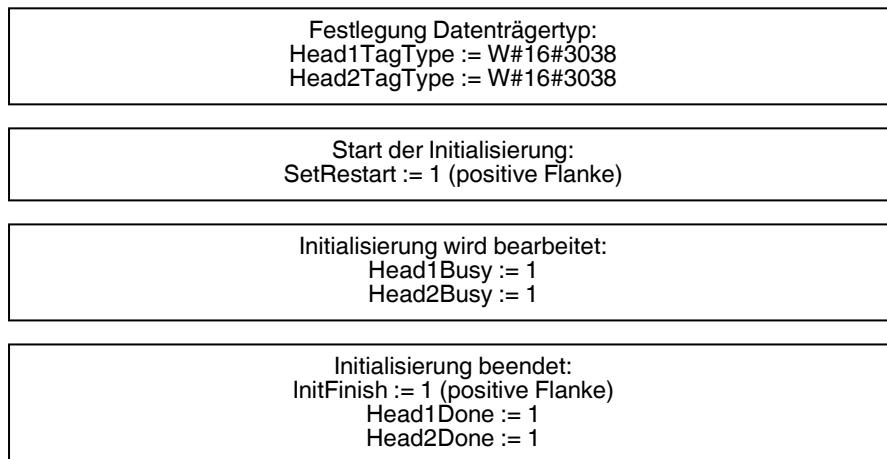
Eine weitere Fehlermöglichkeit besteht bei der Ausführung von Befehlen. Sobald hier ein Fehler auftritt wird das Bit Head1(2)Error gesetzt. Anschließend kann mit Hilfe der aufgelisteten Fehlertabelle eine Diagnose durchgeführt werden. Ursachen für Fehler können bei falschen Befehlsparametern (besonderst bei SpecialCommand) oder falsch eingestellten Datenträgertyp liegen. Ein falscher Befehlsparameter wird durch den Statuswert (Head1(2)Status) 0x04 signalisiert.

Name	Datentyp	Beschreibung	Behebung
Head1/2/3/4Error	BOOL	Fehler an Kopf 1/2/3/4	
Head_1/2/3/4InvalidResponse	BOOL	Gesendetes und empfangenes Befehlstelegramm sind nicht identisch	Überprüfung der Data Hold Time. Überprüfung der E/A Konfiguration.
Head_1/2/3/4.TimeoutOccured	BOOL	Timer zur Kommunikationsüberwachung abgelaufen. Slave hat im Zeitfenster keine Antwort gesendet.	Vergrößerung der Variablen Timeout bei großen Zykluszeiten.
Head_1/2/3/4.Error_SFC_14	BOOL	Fehler beim Einlesen der Daten vom Prozessabbild	Überprüfung der Variablen Head_1/2/3/4.Ret_Val_SFC14. Überprüfung der parametrierten E/A-Adresse und Telegrammlänge
Head_1/2/3/4.Error_SFC_15	BOOL	Fehler beim Schreiben der Daten in das Prozessabbild	Überprüfung der Variablen Head_1/2/3/4.Ret_Val_SFC15. Überprüfung der parametrierten E/A-Adresse und Telegrammlänge.
Head_1/2/3/4.Ret_Val_SFC14	WORD	Fehlercode bei der Ausführung der SFC14	W#16#8090 W#16#80B1 Überprüfung der parametrierten E/A-Adresse und Telegrammlänge (weitere Informationen über Systemhilfe der SFC14)
Head_1/2/3/4.Ret_Val_SFC15	INT	Fehlercode bei der Ausführung der SFC15 (Umwandlung in HEX Format erforderlich)	W#16#8090 W#16#80B1 Überprüfung der parametrierten E/A-Adresse und Telegrammlänge (weitere Informationen über Systemhilfe der SFC15)
Head1/2/3/4Status	BYTE	Status der an Kanal 1/2/3/4 zuletzt eingelesenen Daten	Siehe Tabelle Statuswerte
Memory.InData.Status	BYTE	Status des zuletzt eingelesenen Datentelegramms	Siehe Tabelle Statuswerte

2014-02

## 9 Beispiele Befehlsausführungen

### 9.1 Initialisierung: (mit Datenträgertyp IUC72)

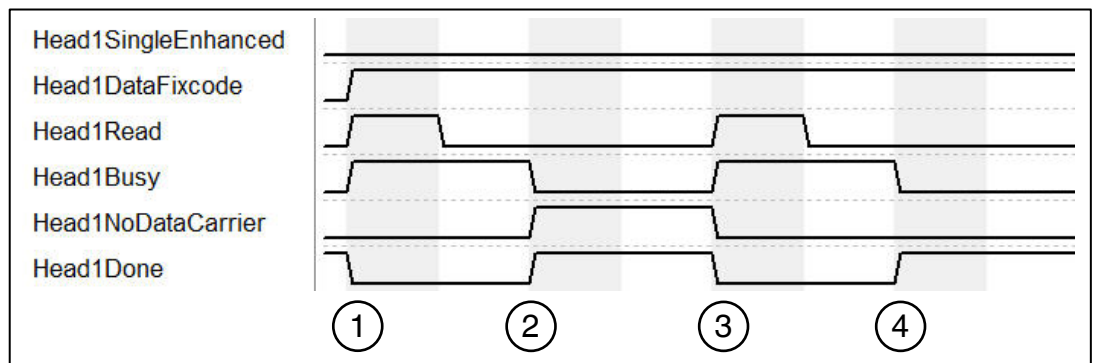
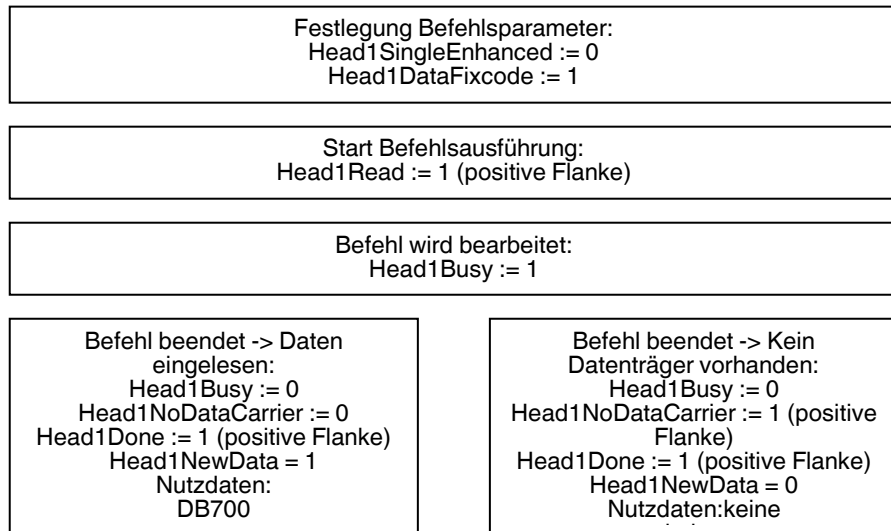


1. „SetRestart“ löst Initialisierungsroutine aus. Initialisierung erfolgt an Kopf 1 „Head1Busy“.
2. Kopf 2 wird unmittelbar initialisiert „Head2Busy“, nachdem Kopf 1 Bestätigung geschickt hat, dass eine Initialisierung an ihm durchgeführt werden soll.
3. Initialisierung an Kopf 1 beendet „Head1Done“.
4. „Head1Done“ und „Head2Done“ auf TRUE. Initialisierung beendet „InitFinish“ geht auf TRUE.

\*Zur besseren Übersicht ist die Restartroutine nur an Kanal 1 und Kanal 2 dargestellt. Analog dazu verhält es sich mit den Kanälen 3 und 4.

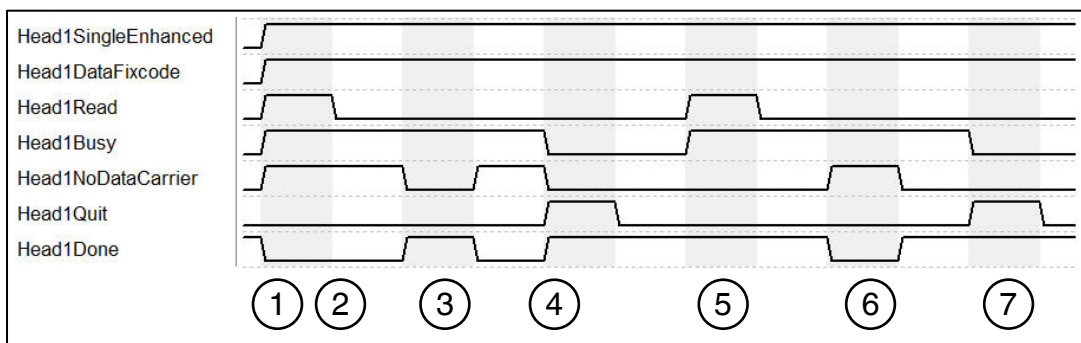
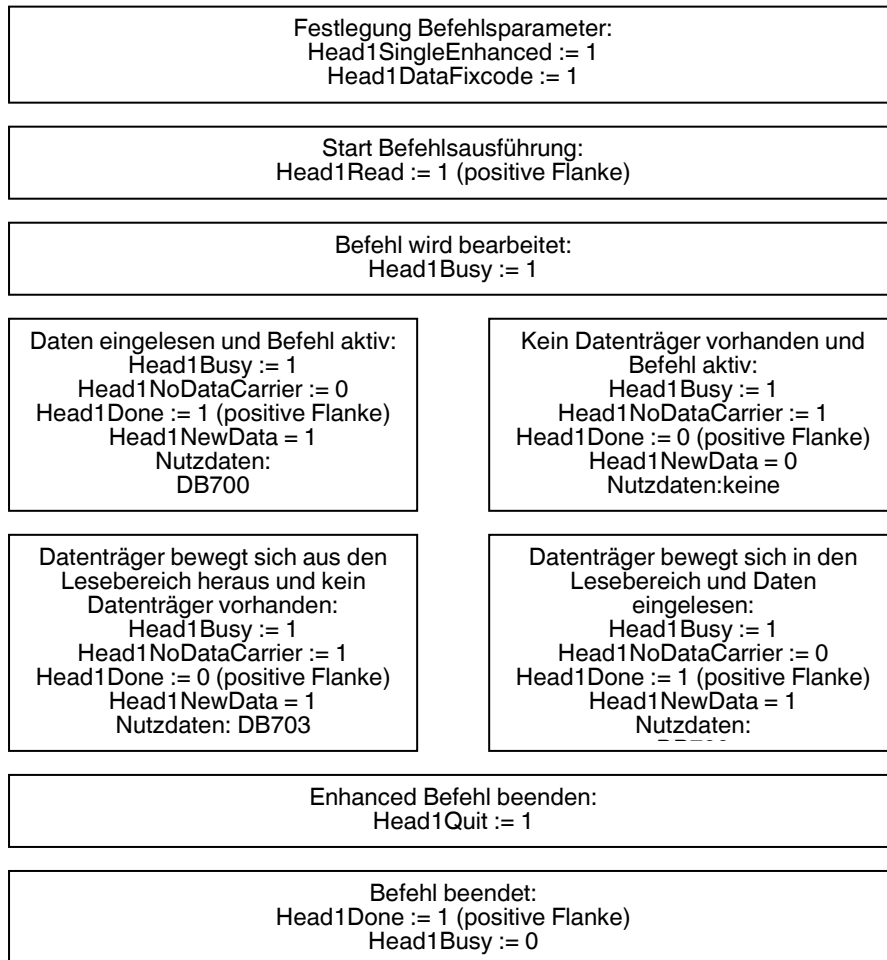


## 9.2 Single Read Fixcode: (Kopf 1)



1. „Head1SingleEnhanced“ ist nullaktiv, um einen Single-Befehl zu initialisieren. „Head1DataFixcode“ ist TRUE, um auf den Fixcode/TID zuzugreifen. Single Read Fixcode wird gestartet, in dem „Head1Read“ auf TRUE wechselt. „Head1Busy“ wechselt auf TRUE, zeitgleich geht „Head1Done“ auf FALSE.
2. Nach der Befehlsausführung ist „Head1Busy“ wieder nullaktiv und „Head1Done“ wechselt zurück auf TRUE. Zum Zeitpunkt der Lesung hat sich kein Datenträger im Erfassungsbereich befunden und „Head1NoDataCarrier“ geht auf TRUE.
3. Single Read Fixcode wird gestartet, in dem „Head1Read“ auf TRUE wechselt. „Head1Busy“ wechselt auf TRUE, zeitgleich geht „Head1Done“ auf FALSE. „Head1NoDataCarrier“ wird FALSE.
4. Nach der Befehlsausführung ist „Head1Busy“ wieder nullaktiv und „Head1Done“ wechselt zurück auf TRUE. Zum Zeitpunkt der Lesung hat sich ein Datenträger im Erfassungsbereich befunden und „Head1NoDataCarrier“ bleibt weiterhin nullaktiv.

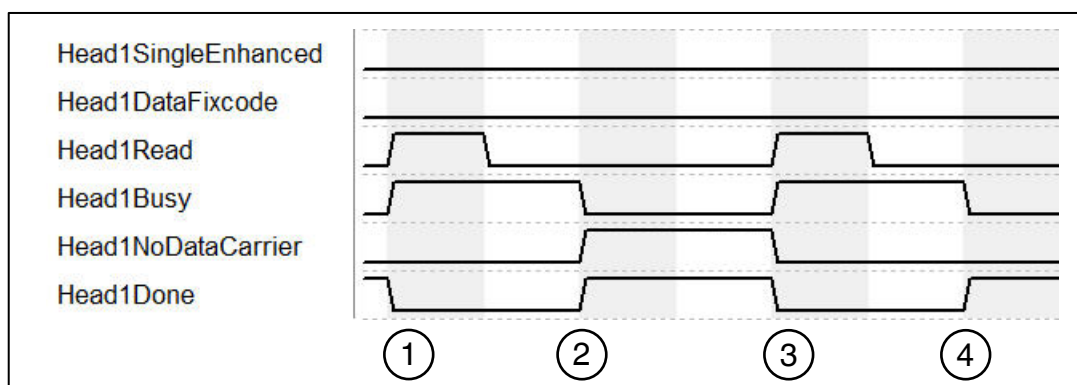
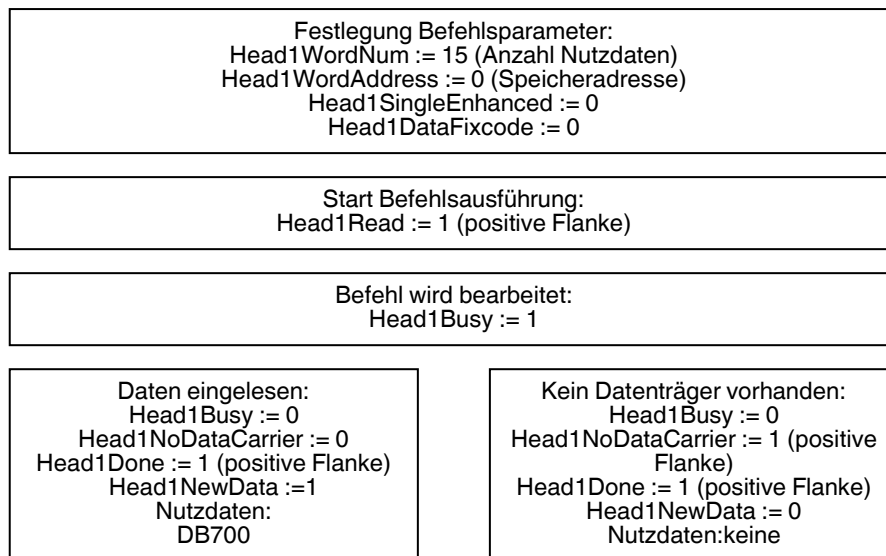
### 9.3 Enhanced Read Fixcode: (Kopf 1)



1. „Head1SingleEnhanced“ ist TRUE, um einen Enhanced-Befehl zu initialisieren. „Head1DataFixcode“ ist TRUE, um auf den Fixcode/TID zuzugreifen. Enhanced Read Fixcode wird gestartet „Head1Read“ ist TRUE. „Head1Busy“ wechselt auf TRUE, zeitgleich geht „Head1Done“ auf FALSE.
2. „Head1NoDataCarrier“ ist TRUE, da sich zu diesem Zeitpunkt kein Datenträger im Feld befindet.
3. Datenträger tritt in Erfassungsbereich. „Head1NoDataCarrier“ geht auf FALSE, „Head1Done“ wechselt auf TRUE. Kurz darauf verlässt der Datenträger wieder das Feld.

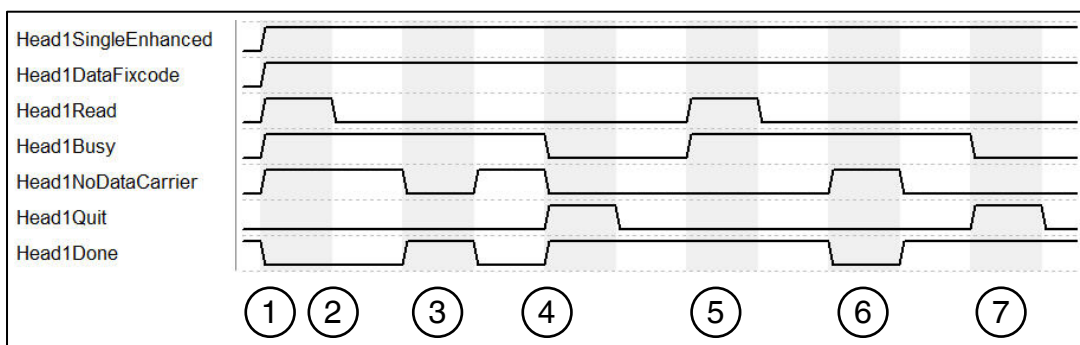
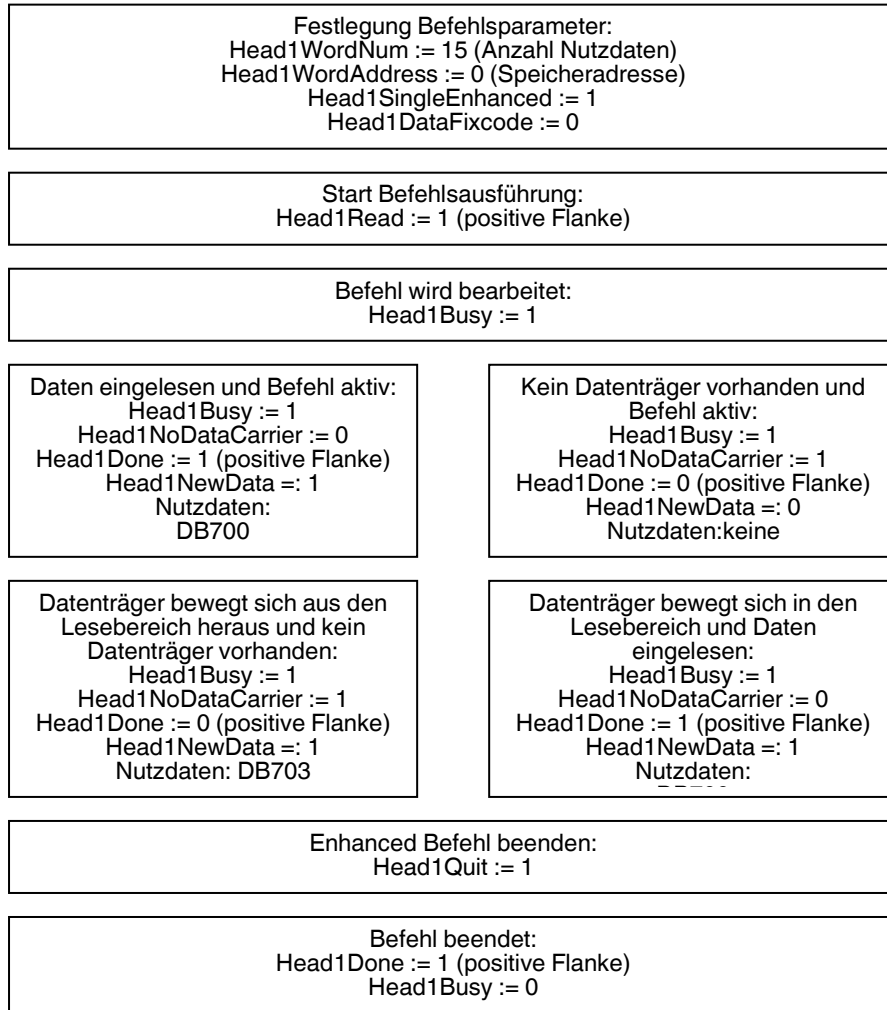
4. „Head1Quit“ bricht die kontinuierliche Lesung ab. „Head1Busy“ geht auf FALSE, „Head1Done“ ist jetzt TRUE.
5. Neuer Enhanced Read Fixcode Befehl gestartet. Datenträger wird erkannt “Head1NoDataCarrier” ist FALSE, “Head1Done” ist aktiv.
6. „Head1Done“ ist FALSE, da kein Datenträger im Feld „Head1NoDataCarrier“ ist TRUE. Kurz darauf tritt Datenträger wieder in den Erfassungsbereich.
7. Befehlsabbruch mit „Head1Quit“

## 9.4 Single Read Words: (Kopf 1; 15 Blöcke ab Adresse 0)



1. “Head1SingleEnhanced” ist nullaktiv, um einen Single-Befehl zu initialisieren. „Head1DataFixcode“ ist FALSE, um auf Words/User-Data zuzugreifen. Single Read Words wird gestartet, in dem „Head1Read“ auf TRUE wechselt. „Head1Busy“ wechselt auf TRUE, zeitgleich geht „Head1Done“ auf FALSE.
2. Nach der Befehlsausführung ist „Head1Busy“ wieder nullaktiv und „Head1Done“ wechselt zurück auf TRUE. Zum Zeitpunkt der Lesung hat sich kein Datenträger im Erfassungsbereich befunden und „Head1NoDataCarrier“ geht auf TRUE.
3. Single Read Words wird gestartet, in dem „Head1Read“ auf TRUE wechselt. „Head1Busy“ wechselt auf TRUE, zeitgleich geht „Head1Done“ auf FALSE. „Head1NoDataCarrier“ wird FALSE.
4. Nach der Befehlsausführung ist „Head1Busy“ wieder nullaktiv und „Head1Done“ wechselt zurück auf TRUE. Zum Zeitpunkt der Lesung hat sich ein Datenträger im Erfassungsbereich befunden und „Head1NoDataCarrier“ bleibt weiterhin nullaktiv.

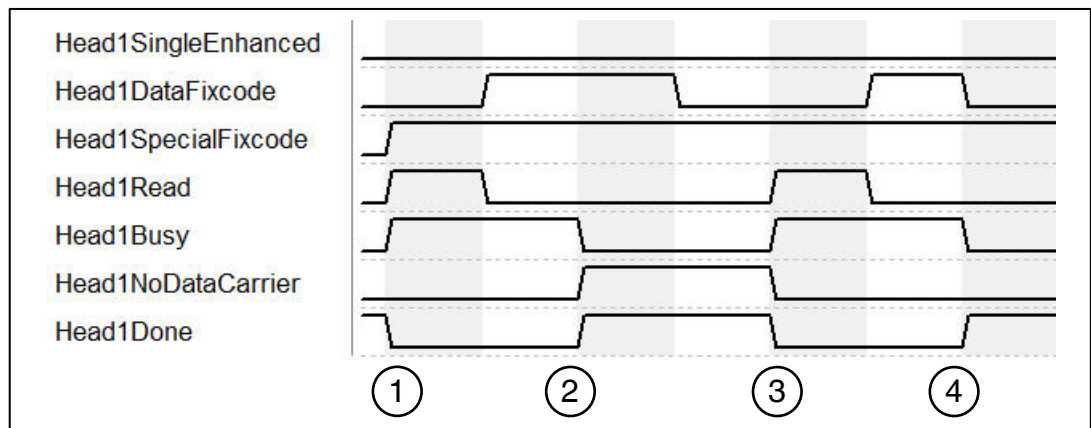
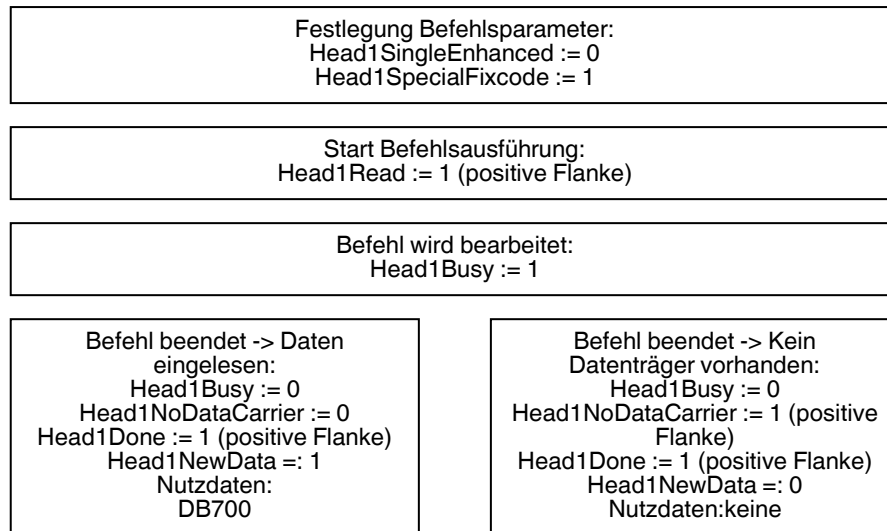
## 9.5 Enhanced Read Words: (Kopf 1; 15 Blöcke ab Adresse 0)



1. Enhanced Read Words wird gestartet „Head1Read“ ist TRUE. „Head1Busy“ wechselt auf TRUE, zeitgleich geht „Head1Done“ auf FALSE.
2. „Head1NoDataCarrier“ ist TRUE, da sich zu diesem Zeitpunkt kein Datenträger im Feld befindet.
3. Datenträger tritt in Erfassungsbereich. „Head1NoDataCarrier“ geht auf FALSE, „Head1Done“ wechselt auf TRUE. Kurz darauf verlässt der Datenträger wieder das Feld.

4. „Head1Quit“ bricht die kontinuierliche Lesung ab. „Head1Busy“ geht auf FALSE, „Head1Done“ ist jetzt TRUE.
5. Neuer Enhanced Read Words Befehl gestartet. Datenträger wird erkannt “Head1NoDataCarrier” ist FALSE, “Head1Done” ist aktiv.
6. „Head1Done“ ist FALSE, da kein Datenträger im Feld „Head1NoDataCarrier“ ist TRUE. Kurz darauf tritt Datenträger wieder in den Erfassungsbereich.
7. Befehlsabbruch mit „Head1Quit“

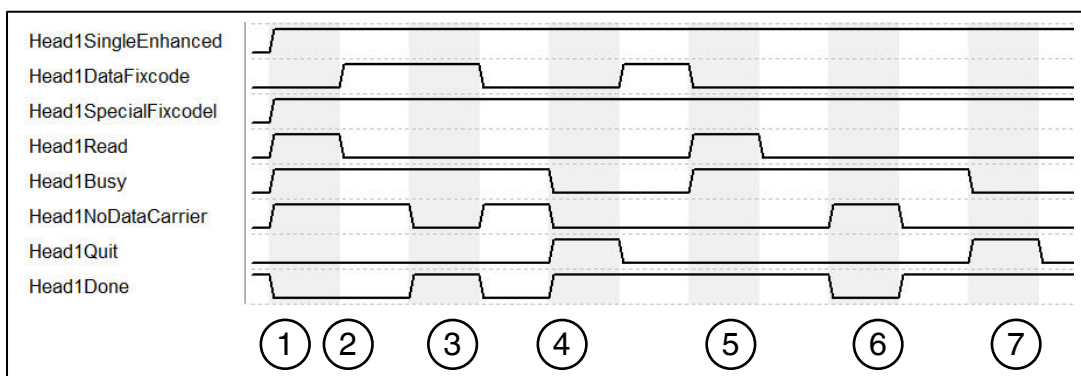
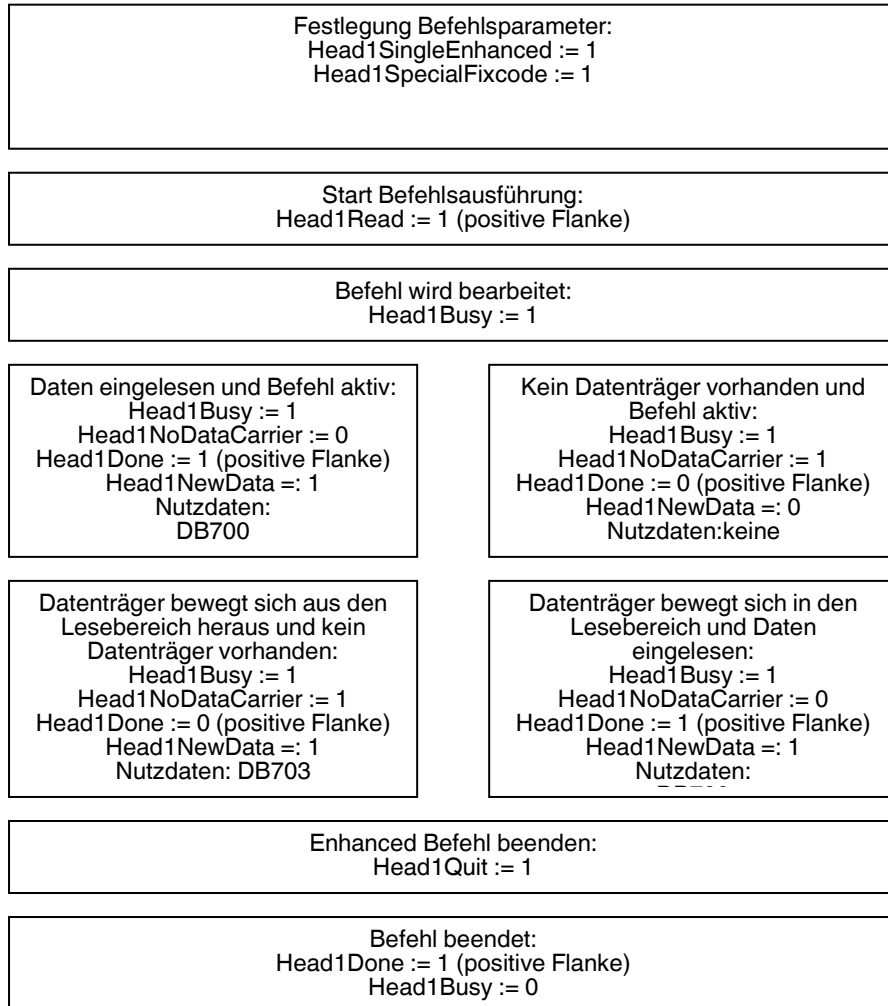
## 9.6 Single Read SpecialFixcode: (Kopf 1)



1. „Head1SingleEnhanced“ ist nullaktiv, um einen Single-Befehl zu initialisieren. „Head1DataFixcode“ ist irrelevant, da „Head1SpecialFixcode“ höher priorisiert ist als Words oder Fixcode (d.h. sobald „Head1SpecialFixcode“ TRUE ist, ist „Head1DataFixcode“ hinfällig). „Head1SpecialFixcode“ ist TRUE, um auf den SpecialFixcode/EPC zuzugreifen. Single Read SpecialFixcode wird gestartet, in dem „Head1Read“ auf TRUE wechselt. „Head1Busy“ wechselt auf TRUE, zeitgleich geht „Head1Done“ auf FALSE.
2. Nach der Befehlsausführung ist „Head1Busy“ wieder nullaktiv und „Head1Done“ wechselt zurück auf TRUE. Zum Zeitpunkt der Lesung hat sich kein Datenträger im Erfassungsbereich befunden und „Head1NoDataCarrier“ geht auf TRUE.
3. Single Read SpecialFixcode wird gestartet, in dem „Head1Read“ auf TRUE wechselt. „Head1Busy“ wechselt auf TRUE, zeitgleich geht „Head1Done“ auf FALSE. „Head1NoDataCarrier“ wird FALSE.

- Nach der Befehlsausführung ist „Head1Busy“ wieder nullaktiv und „Head1Done“ wechselt zurück auf TRUE. Zum Zeitpunkt der Lesung hat sich ein Datenträger im Erfassungsbereich befunden und „Head1NoDataCarrier“ bleibt weiterhin nullaktiv.

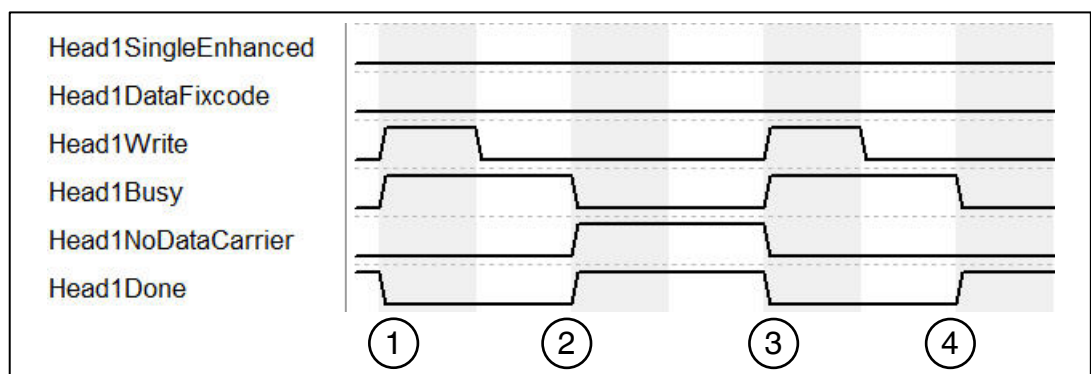
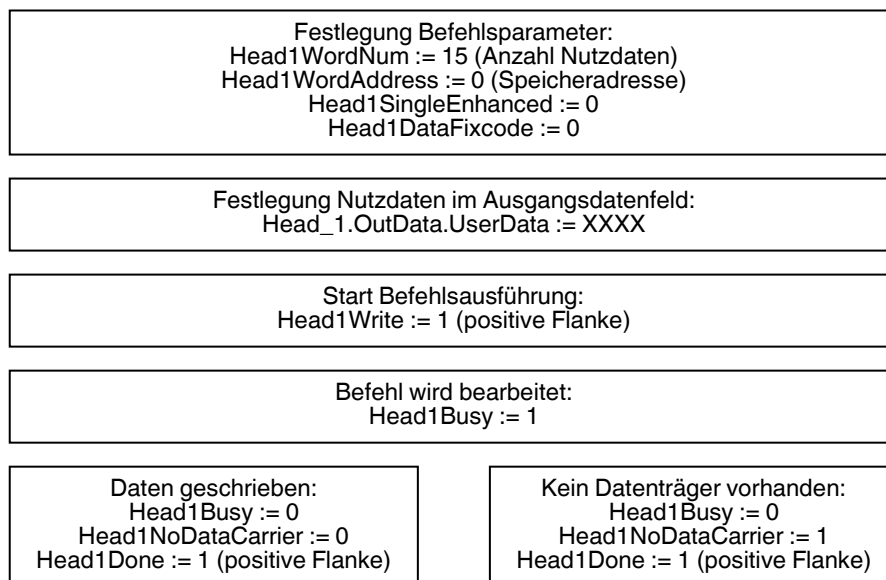
## 9.7 Enhanced Read SpecialFixcode: (Kopf 1)



- „Head1SingleEnhanced“ ist TRUE, um einen Enhanced-Befehl zu initialisieren. „Head1DataFixcode“ ist irrelevant, da „Head1SpecialFixcode“ höher priorisiert ist als Words oder Fixcode (d.h. sobald „Head1SpecialFixcode“ TRUE ist, ist „Head1DataFixcode“ hinfällig). „Head1SpecialFixcode“ ist TRUE, um auf den SpecialFixcode/EPC zuzugreifen Enhanced Read SpecialFixcode wird gestartet

- „Head1Read“ ist TRUE. „Head1Busy“ wechselt auf TRUE, zeitgleich geht „Head1Done“ auf FALSE.
- 2. „Head1NoDataCarrier“ ist TRUE, da sich zu diesem Zeitpunkt kein Datenträger im Feld befindet.
- 3. Datenträger tritt in Erfassungsbereich. „Head1NoDataCarrier“ geht auf FALSE, „Head1Done“ wechselt auf TRUE. Kurz darauf verlässt der Datenträger wieder das Feld.
- 4. „Head1Quit“ bricht die kontinuierliche Lesung ab. „Head1Busy“ geht auf FALSE, „Head1Done“ ist jetzt TRUE.
- 5. Neuer Enhanced Read SpecialFixcode Befehl gestartet. Datenträger wird erkannt „Head1NoDataCarrier“ ist FALSE, „Head1Done“ ist aktiv.
- 6. „Head1Done“ ist FALSE, da kein Datenträger im Feld „Head1NoDataCarrier“ ist TRUE. Kurz darauf tritt Datenträger wieder in den Erfassungsbereich.
- 7. Befehlsabbruch mit „Head1Quit“

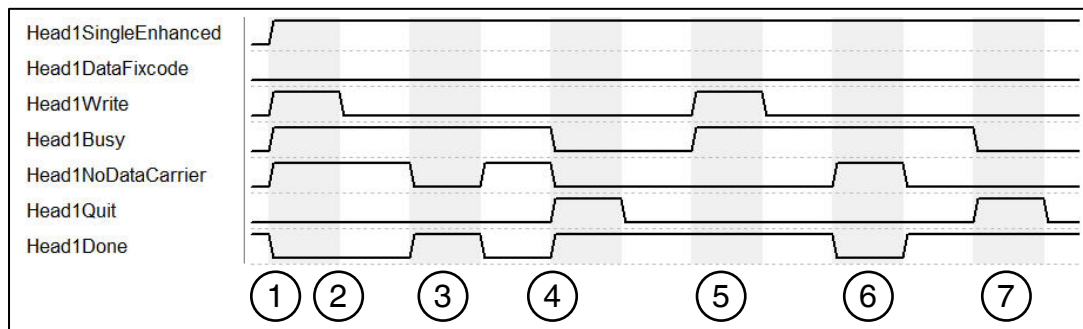
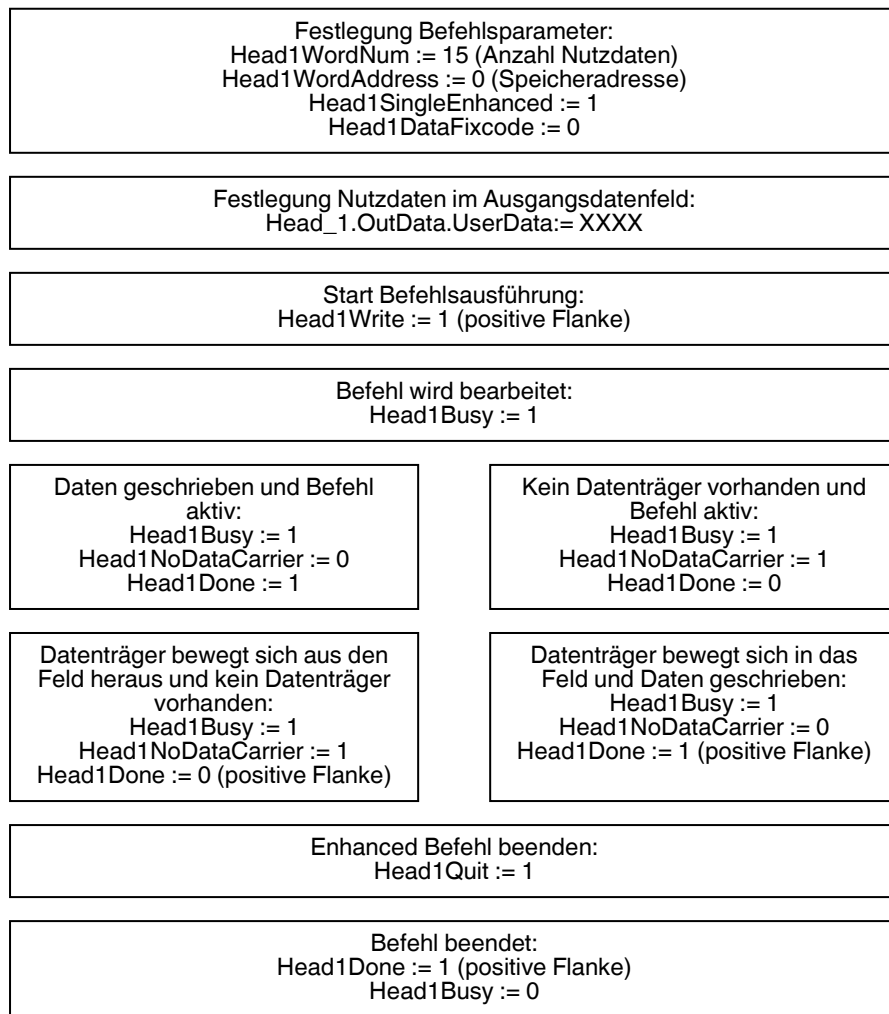
### 9.8 Single Write Words: (Kopf 1; 15 Blöcke ab Adresse 0)



1. „Head1SingleEnhanced“ ist nullaktiv, um einen Single-Befehl zu initialisieren. „Head1DataFixcode“ ist FALSE, um auf Words/User-Data zuzugreifen. Single Write Words wird gestartet, in dem „Head1Write“ auf TRUE wechselt. „Head1Busy“ wechselt auf TRUE, zeitgleich geht „Head1Done“ auf FALSE.

2. Nach der Befehlsausführung ist „Head1Busy“ wieder nullaktiv und „Head1Done“ wechselt zurück auf TRUE. Zum Zeitpunkt des Schreibens hat sich kein Datenträger im Erfassungsbereich befunden und „Head1NoDataCarrier“ geht auf TRUE.
3. Single Write Words wird gestartet, in dem „Head1Write“ auf TRUE wechselt. „Head1Busy“ wechselt auf TRUE, zeitgleich geht „Head1Done“ auf FALSE. „Head1NoDataCarrier“ wird FALSE.
4. Nach der Befehlsausführung ist „Head1Busy“ wieder nullaktiv und „Head1Done“ wechselt zurück auf TRUE. Zum Zeitpunkt des Schreibens hat sich ein Datenträger im Erfassungsbereich befunden und „Head1NoDataCarrier“ bleibt weiterhin nullaktiv.

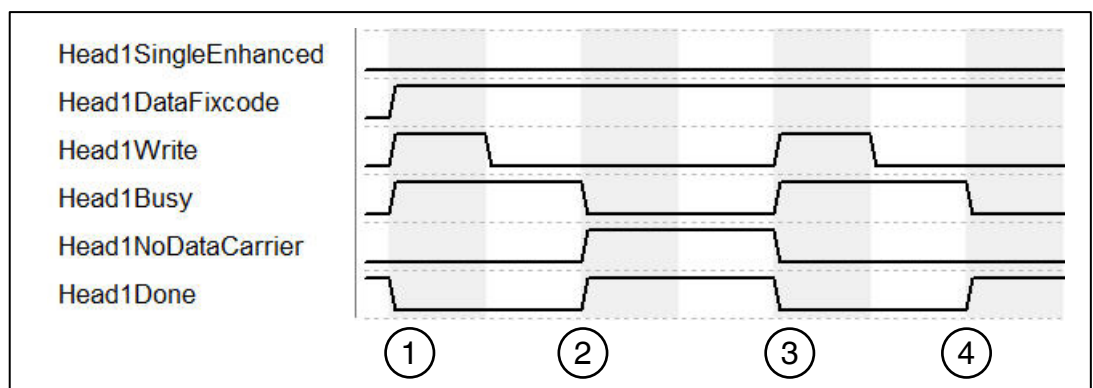
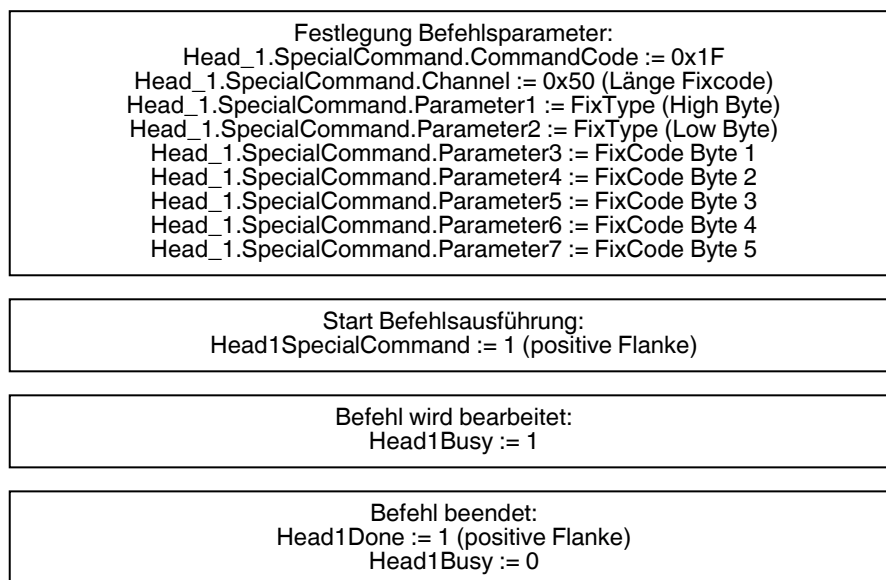
## 9.9 Enhanced Write Words: (Kopf 1; 15 Blöcke ab Adresse 0)





1. „Head1SingleEnhanced“ ist TRUE, um einen Enhanced-Befehl zu initialisieren. „Head1DataFixcode“ ist False, um auf Words/User-Data zuzugreifen. Enhanced Write Words wird gestartet „Head1Write“ ist TRUE. „Head1Busy“ wechselt auf TRUE, zeitgleich geht „Head1Done“ auf FALSE.
2. „Head1NoDataCarrier“ ist TRUE, da sich zu diesem Zeitpunkt kein Datenträger im Feld befindet.
3. Datenträger tritt in Erfassungsbereich. „Head1NoDataCarrier“ geht auf FALSE, „Head1Done“ wechselt auf TRUE. Kurz darauf verlässt der Datenträger wieder das Feld.
4. „Head1Quit“ bricht das kontinuierliche Schreiben ab. „Head1Busy“ geht auf FALSE, „Head1Done“ ist jetzt TRUE.
5. Neuer Enhanced Write Words Befehl gestartet. Datenträger wird erkannt „Head1NoDataCarrier“ ist FALSE, „Head1Done“ ist aktiv.
6. „Head1Done“ ist FALSE, da kein Datenträger im Feld „Head1NoDataCarrier“ ist TRUE. Kurz darauf tritt Datenträger wieder in den Erfassungsbereich.
7. Befehlsabbruch mit „Head1Quit“

## 9.10 Single Write Fixcode: (Kopf 1; IPC11)

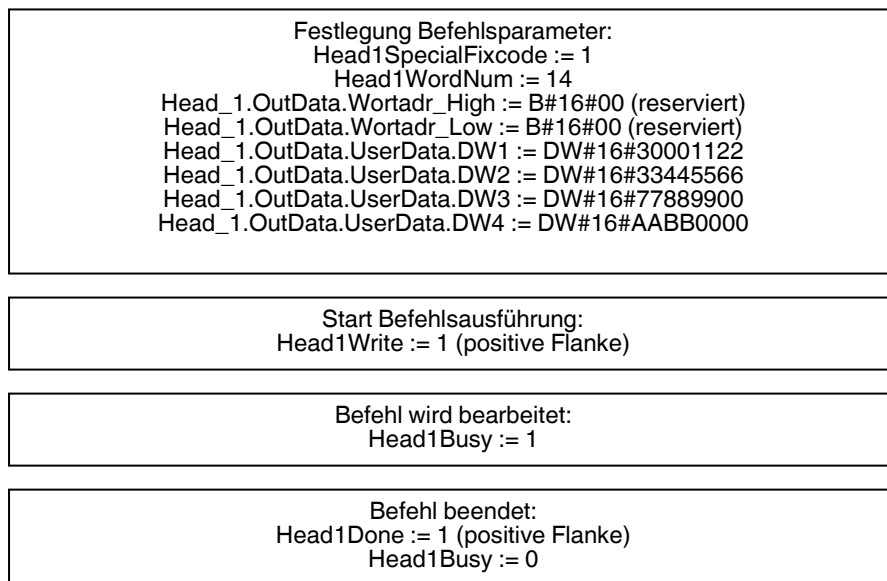


1. „Head1SingleEnhanced“ ist nullaktiv, um einen Single-Befehl zu initialisieren. „Head1DataFixcode“ ist TRUE, um auf den Fixcode/TID zuzugreifen. Single Write

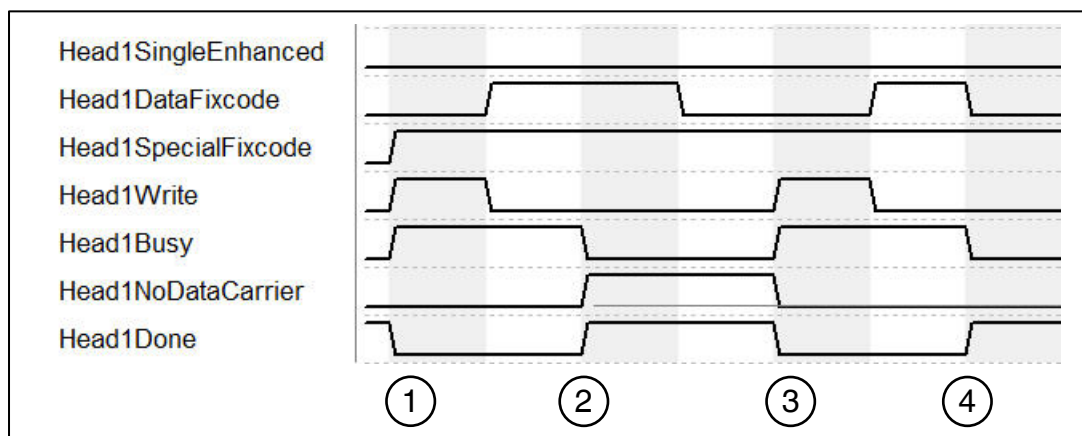
Fixcode wird gestartet, in dem „Head1Write“ auf TRUE wechselt. „Head1Busy“ wechselt auf TRUE, zeitgleich geht „Head1Done“ auf FALSE.

2. Nach der Befehlsausführung ist „Head1Busy“ wieder nullaktiv und „Head1Done“ wechselt zurück auf TRUE. Zum Zeitpunkt des Schreibens hat sich kein Datenträger im Erfassungsbereich befunden und „Head1NoDataCarrier“ geht auf TRUE.
3. Single Write Fixcode wird gestartet, in dem „Head1Write“ auf TRUE wechselt. „Head1Busy“ wechselt auf TRUE, zeitgleich geht „Head1Done“ auf FALSE. „Head1NoDataCarrier“ wird FALSE.
4. Nach der Befehlsausführung ist „Head1Busy“ wieder nullaktiv und „Head1Done“ wechselt zurück auf TRUE. Zum Zeitpunkt des Schreibens hat sich ein Datenträger im Erfassungsbereich befunden und „Head1NoDataCarrier“ bleibt weiterhin nullaktiv.

## 9.11 Single Write SpecialFixcode: (Kopf 1)



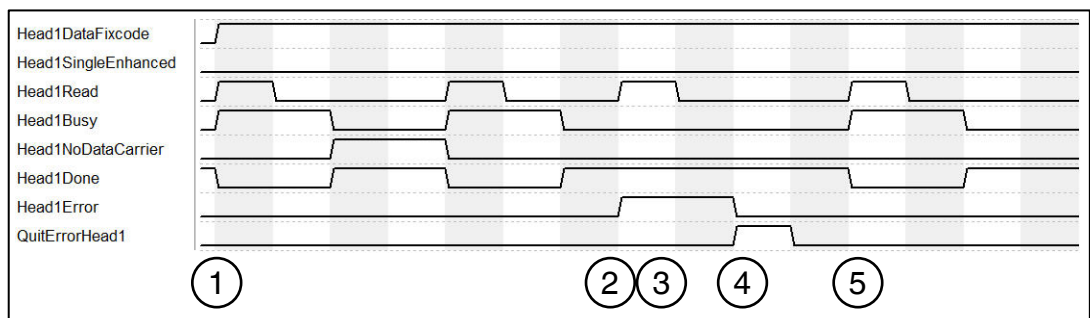
Korrekte Notation des PC beachten (gemäß EPC Gen 2 (ISO/IEC 18000-63))! Kann bei falscher Konfiguration zur Unbrauchbarkeit des Datenträgers führen!



1. „Head1SingleEnhanced“ ist nullaktiv, um einen Single-Befehl zu initialisieren. „Head1DataFixcode“ ist irrelevant, da „Head1SpecialFixcode“ höher priorisiert ist als Words oder Fixcode (d.h. sobald „Head1SpecialFixcode“ TRUE ist, ist „Head1DataFixcode“ hinfällig). „Head1SpecialFixcode“ ist TRUE, um auf den SpecialFixcode/EPC zuzugreifen. Single Write SpecialFixcode wird gestartet, in dem

- „Head1Write“ auf TRUE wechselt. „Head1Busy“ wechselt auf TRUE, zeitgleich geht „Head1Done“ auf FALSE.
2. Nach der Befehlsausführung ist „Head1Busy“ wieder nullaktiv und „Head1Done“ wechselt zurück auf TRUE. Zum Zeitpunkt des Schreibens hat sich kein Datenträger im Erfassungsbereich befunden und „Head1NoDataCarrier“ geht auf TRUE.
3. Single Write SpecialFixcode wird gestartet, in dem „Head1Write“ auf TRUE wechselt. „Head1Busy“ wechselt auf TRUE, zeitgleich geht „Head1Done“ auf FALSE. „Head1NoDataCarrier“ wird FALSE.
4. Nach der Befehlsausführung ist „Head1Busy“ wieder nullaktiv und „Head1Done“ wechselt zurück auf TRUE. Zum Zeitpunkt des Schreibens hat sich ein Datenträger im Erfassungsbereich befunden und „Head1NoDataCarrier“ bleibt weiterhin nullaktiv.

## 9.12 Error Handling: (Kopf 1)



1. Befehlsausführung gestartet.
2. Nach zweitem Lesebefehl ist ein Fehler aufgetreten „Head1Error“ ist TRUE.
3. Fehler verriegeln den kompletten Baustein. Dieser ignoriert nachfolgende Befehle. „Head1Read“ hat keine Auswirkung, so lange „Head1Error“ TRUE ist.
4. „QuitErrorHead1“ quittiert den Fehler und hebt Verriegelung des Bausteins auf.
5. Normale Befehlsausführung wieder möglich.

## 9.13 Befehlsliste (Prefetch): (Kopf 1)

Mit Hilfe der Befehlsliste können mehrere Befehle nacheinander abgearbeitet werden. Zunächst muss die Befehlsliste geöffnet werden. Anschließend werden die auszuführenden Befehle in die Liste übertragen und nach Beendigung wird die Befehlsliste wieder geschlossen. Die Liste wird abgearbeitet, sobald diese im Single oder Enhanced Mode aktiviert wird. Die Befehlslisten werden flüchtig gespeichert. Das Öffnen, Schließen und Aktivieren der Befehlslisten, sowie die Übertragung der auszuführenden Befehle erfolgt über den SpecialCommand.

Öffnen der Befehlsliste:

Festlegung Befehlsparameter: Öffnen der Befehlsliste  
Head\_1.SpecialCommand.CommandCode := 0xAC  
Head\_1.SpecialCommand.Channel := 0x00  
Head\_1.SpecialCommand.Parameter1 := 0x00 (ListNo)  
Head\_1.SpecialCommand.Parameter2 := 0x01 (ListModus)

Start Befehlsausführung: Befehlsübertragung  
Head1SpecialCommand := 1 (positive Flanke)

Befehl wird bearbeitet: Übertragung läuft  
Head1Busy := 1

Befehl beendet: Übertragung beendet  
Head1Done := 1 (positive Flanke)  
Head1Busy := 0

Nach dem Öffnen der Befehlsliste werden die auszuführenden Befehle durch den SpecialCommand an die Auswerteeinheit übertragen. Nachfolgend werden zunächst der Befehl Enhanced Read Fixcode und anschließend der Befehl Enhanced Read Words parametrisiert.

Befehl 1: Enhanced Read Fixcode Kopf 1

Festlegung Befehlsparameter: Enhance Read Fixcode  
Head\_1.SpecialCommand.CommandCode := 0x01  
Head\_1.SpecialCommand.Channel := 0x00

Start Befehlsausführung: Befehlsübertragung  
Head1SpecialCommand := 1 (positive Flanke)

Befehl wird bearbeitet: Übertragung läuft  
Head1Busy := 1

Befehl beendet: Übertragung beendet  
Head1Done := 1 (positive Flanke)  
Head1Busy := 0

## Befehl 2: Enhanced Read Words

Festlegung Befehlsparameter: Enhanced Read Words  
Head\_1.SpecialCommand.CommandCode := 0x19  
Head\_1.SpecialCommand.Channel := 0xF0 (Anzahl Datenblöcke)  
Head\_1.SpecialCommand.Parameter1 := 0x00 (WordAddress High Byte)  
Head\_1.SpecialCommand.Parameter2 := 0x00 (WordAddress Low Byte)

Start Befehlsausführung: Befehlsübertragung  
Head1SpecialCommand := 1 (positive Flanke)

Befehl wird bearbeitet: Übertragung läuft  
Head1Busy := 1

Befehl beendet: Übertragung beendet  
Head1Done := 1 (positive Flanke)  
Head1Busy := 0

Die Anzahl der parametrisierten Befehle kann auf maximal 10 erweitert werden. Die Parametrierung wird durch das Schließen der Befehlsliste abgeschlossen.

Schließen der Befehlsliste:

Festlegung Befehlsparameter: Schließen der Befehlsliste  
Head\_1.SpecialCommand.CommandCode := 0xAC  
Head\_1.SpecialCommand.Channel := 0x00  
Head\_1.SpecialCommand.Parameter1 := 0x00 (ListNo)  
Head\_1.SpecialCommand.Parameter2 := 0x00 (ListModus)

Start Befehlsausführung: Befehlsübertragung  
Head1SpecialCommand := 1 (positive Flanke)

Befehl wird bearbeitet: Übertragung läuft  
Head1Busy := 1

Befehl beendet: Übertragung beendet  
Head1Done := 1 (positive Flanke)  
Head1Busy := 0

Die Befehlsliste wird ausgeführt, sobald diese aktiviert wird. Die Aktivierung kann im Single oder Enhanced Modus erfolgen.

Aktivierung im Single Modus:

```
Festlegung Befehlsparameter: Aktivierung Single Mode
Head_1.SpecialCommand.CommandCode := 0xAC
Head_1.SpecialCommand.Channel := 0x00
Head_1.SpecialCommand.Parameter1 := 0x00 (ListNo)
Head_1.SpecialCommand.Parameter2 := 0x02 (ListModus)
```

```
Start Befehlsausführung: Befehlsübertragung
Head1SpecialCommand := 1 (positive Flanke)
```

```
Befehl wird bearbeitet: Übertragung läuft
Head1Busy := 1
```

```
Befehl beendet: Übertragung beendet
Head1Done := 1 (positive Flanke)
Head1Busy := 0
```

Nach der Aktivierung der Befehlsliste, werden die Befehle nacheinander abgearbeitet. Die eingelesenen Nutzdaten befinden sich innerhalb der Datenstruktur Head\_1.InData.UserData. Befindet sich ein Datenträger innerhalb des Erfassungsbereichs, so hat der Status (Head1Status) aller Antworttelegramme den Wert 0x00. Für eine Unterscheidung der verschiedenen aufeinanderfolgenden Antworttelegramme kann der Ausführungszähler (Head1ReplyCounter) verwendet werden. Bei Erhalt eines neuen Antworttelegramms hat sich der Wert dieser Variable zum vorhergehenden Telegramm verändert. Nach der Durchführung eines Durchlaufs der Befehlsliste erhält man ein Antworttelegramm mit dem Status (Head1Status) 0x0F. Bei Aktivierung der Befehlsliste im Single Mode wird die Befehlsliste einmalig durchlaufen. Nach Erhalt des letzten Telegramms (Head1Status = 0x0F) wird die Ausführung automatisch gestoppt. Bei der Aktivierung im Enhanced Mode beginnt nach Erhalt des letzten Telegramms die Ausführung der Befehlsliste von vorn.

## 10 Tabelle Datenträger

Name	TagType	Zugriff	Fixcode	Daten	WordAddress	Frequenz
IPC02	W#16#3032	Read Fixcode	5 Byte	-	-	125kHz
IPC03	W#16#3033	Read Fixcode Read Words Write Words	4 Byte	116 Byte	0000...001C	125kHz
IPC11	W#16#3131	Read Fixcode Write Fixcode	5 Byte	-	-	125kHz
IPC12	W#16#3132	Read Fixcode Read Words Write Words	4 Byte	8192 Byte	0000...07FF	125kHz
IDC	W#16#3530	Read Words Write Words Read Fixcode Write Fixcode Read Special Fixcode Write Special Fixcode	7 Byte Fixcode  6 Byte Special Fixcode	128 Byte	0000...001F	250kHz
ICC	W#16#3532	Read Fixcode	7 Byte	-	-	250kHz
IUC72	W#16#3830	Read Fixcode Write Fixcode Read Word Write Words Read Special Fixcode Write Special Fixcode	8 Byte	64 Byte		865 ... 868 MHz
IUC73	W#16#3830	Read Fixcode Read Special Fixcode	8 Byte	-	-	868 MHz
IUC74	W#16#3830	Read Fixcode Write Fixcode Read Words Write Words Read Special Fixcode Write Special Fixcode	8 Byte	28 Byte		865 ... 871 MHz
IUC75	W#16#3830		8 Byte	-	-	865 ... 928 MHz
IQC21	W#16#3231	Read Fixcode Read Words Write Words	8 Byte	112 Byte	0000...001B	13,56MHz
IQC22	W#16#3232	Read Fixcode Read Words Write Words	8 Byte	256 Byte	0000...003F	13,56MHz
IQC23	W#16#3233	Read Fixcode Read Words Write Words	8 Byte	224 Byte	0000...0037	13,56MHz
IQC24	W#16#3234	Read Fixcode Read Words Write Words	8 Byte	928 Byte	0000...00E7	13,56MHz
IQC31	W#16#3331	Read Fixcode Read Words Write Words	8 Byte	32 Byte	0000...0007	13,56MHz
IQC33	W#16#3333	Read Fixcode Read Words	8 Byte	2000 Byte	0000...00F9	13,56MHz

2014-02



		Write Words				
IQC35	W#16#3335	Read Fixcode Read Words Write Words	8 Byte	256 Byte	0000...003F	13,56MHz
MVC	W#16#3630	Read Fixcode Read Words Write Words	8 Byte	7552 Byte	0000...075F	2,45GHz



## 11 Tabelle Statuswerte

HeadX Status	Bedeutung	Aktion
0x00	Befehl wurde fehlerfrei ausgeführt	Keine; ein neuer Befehl wurde an diesen Kanal gesendet
0xFF	Befehl in Bearbeitung	Befehl wird gerade an diesen Kanal bearbeitet; ein Befehl kann an einen weiteren Kanal gesendet werden (nicht den gleichen)
0x01	Niedriger Batteriestand (gilt nur für MVC Datenträger)	Nutzdaten sind im selben Telegramm enthalten; Wechsel der Batterie oder das ganzen Datenträgers empfohlen;
0x02	Einschaltmeldung; Reset Befehl wurde ausgeführt	Auswerteeinheit ist für Befehlsausführung bereit
0x04	Falscher bzw. unvollständiger Befehl; Befehlsparameter nicht im gültigen Bereich	Überprüfung der Befehlsparameter und des Datenträgertyps (IQC33 hat nur geradzahlige Vielfache der WordNum); Überprüfung der Installation des Kopfes (ist Kopf geerdet; wurde abgeschirmtes Kabel verwendet)
0x05	Kein Datenträger befindet sich im Erfassungsbereich	Überprüfung des Abstandes zwischen Datenträger und Kopf; Überprüfung der Installation des Kopfes (ist Kopf geerdet; wurde abgeschirmtes Kabel verwendet)
0x06	Hardwarefehler; kein Schreib-/Lesekopf an diesen Kanal angebunden; Schreib-/Lesekopf ist defekt	Überprüfung des Lesekopfkabels (abgeschirmtes Kabel V1-G-XM-PUR ABG-V1-W); Überprüfung der LED des Kopfes (ausgeschaltet: Kopf defekt; blinkend: Ausführung der Initialisierung erforderlich; konstant: Kopf ist OK)
0x07	Interner Gerätefehler	Überlauf des internen Speichers (Verkürzung der Data Hold Time)
0x09	Parametriertes Datenträgertyp passt nicht zum angeschlossenen Lesekopf	Überprüfung des festgelegten Datenträgertyps
0x0A	Es befinden sich mehrere Transponder mit gleichem EPC im Erfassungsbereich	Sicherstellen, dass jeder Transponder einen einzigartigen EPC aufweist.
0x0B	Zusätzliches Telegramm bei eingeschaltetem IF-Parameter	Keine; Telegramm mit Zusatzinformationen wurde gesendet
0x0F	Letztes Telegramm im Multiframe-Modus	Keine; Nach einem Single-Befehl Anzahl der gelesenen Transponder im Feld in ASCII
0x20	Einschaltmeldung; Reset Befehl wurde ausgeführt	Fehler wurde durch das Businterface gesendet; Auswerteeinheit ist für Befehlsausführung bereit
0x40	Falscher bzw. unvollständiger Befehl; Befehlsparameter nicht im gültigen Bereich	Fehler wurde durch das Businterface gesendet; Überprüfung der Befehlsparameter und des Datenträgertyps (IQC33 hat nur geradzahlige Vielfache der WordNum); Überprüfung der Installation des Kopfes (ist Kopf geerdet; wurde abgeschirmtes Kabel verwendet)
0x60	Hardwarefehler; kein Schreib-/Lesekopf an diesen Kanal angebunden; Schreib-/Lesekopf ist defekt	Fehler wurde durch das Businterface gesendet; Überprüfung des Lesekopfkabels (abgeschirmtes Kabel V1-G-XM-PUR ABG-V1-W); Überprüfung der LED des Kopfes (ausgeschaltet: Kopf defekt; blinkend: Ausführung der Initialisierung erforderlich; konstant: Kopf ist OK)
0x70	Interner Gerätefehler	Überlauf des internen Speichers (Verkürzung der Data Hold Time)

2014-02

## 12 Tabelle Versionsmeldung

Version	Datum	Anderung Funktionsbaustein	Anderung Dokumentation
2.0	18.11.2008	Anpassung von 4-Kanal Version auf 2-Kanal Version Ersetzen der IN-Variable „IDENTControlAddress“ durch die IN-Variablen INPUT/OUTPUT_Address und INPUT/OUTPUT_Length Ausgangsdatenlänge von 4 Bytes durch Implementation der SFC81 möglich Einführung der OUT-Variablen Done; NoDataCarrier; Busy und Error	Ersterstellung
2.1	28.02.2009	Variable Memory.Error_SFC_14 wird mit den Variablen Head1/2Error verknüpft Änderung der Überprüfung der Variablen Head_1/2.ExistTC und Head_1/2.NotExist bei der Parametrierung der Befehle -> Triggermode Einführung der Statusüberprüfung auf den Wert 0x0F -> letztes Telegramm der Befehlsliste Änderung der Gleichheitsüberprüfung der Eingangs- und Ausgangsdatenfelder -> bei Ausführung der Command List sind Aus- und Eingangsdatenfeld ungleich Rücksetzen der Variable Head_1/2.QuitError in der Restart und QuitError Routine Einführung neuer Symbolnamen für die Nutzdatenfelder der Ein- und Ausgangsdatenfelder -> Head_1/2.InData/OutData.UserData Einführung einer Wertüberprüfung des Parameters Head1/2WordNum -> Wert > 15 wird Head1/2Error gesetzt Einführung der Ausgangsparameter Head1/2Status und Head1/2ReplyCounter -> Batteriestatus und Befehlsliste	Einführung neues Bild für Übersicht der Variablen des Funktionsbausteins (Seite 4) Einführung der neuen Variablen in die Tabelle zur Funktionsbeschreibung (Seite 4-5) Hinweis auf die maximale Telegrammlänge verschiedener CPU Baureihen (Seite 6) Änderung der Variablenbenennung des Statuswertes (Seite 6-7)
2.2	24.04.2009	Einfügen einer Textbibliothek zur Zuordnung der Stausmeldungen	keine
2.8	02.08.2013	Implementierung des Multiframe-Modus Parametrierung der IUH-Leseköpfe nun möglich SpecialFixcode (EPC) les-/schreibbar Datenbausteine zum Verarbeiten der unterschiedlichen Telegramme an Kanal1/2 eingefügt (DB70X & DB71X) SetRestart löscht Inhalt der Datenbausteine In/Out HeadXNewData zeigt neue Daten in Datenbausteinen an HeadXCacheFull signalisiert überlaufenden Datenbaustein	Parametriermöglichkeiten mit FB190 erklärt (Seite 6) Erklärung des neuen Multiframe-Bausteins FB32 (Seite 10) Update diverser Statuswerte
2.9	04.09.2013	Filterfunktionen dem FB32 hinzugefügt FB190 zeigt fehlerhaft konfigurierte Parameter an	Filterfunktionen erklärt (Seite 17) Fehleranalyse bei Parameterbefehlen (Seite 8)
3.0	25.10..2013	Ein DB zum Speichern aller Nutztelegramme (User/Fixcode/Special Fixcode) Verhalten von HeadXDone für Single- und Enhanced-Befehle angepasst Schreibbefehle führen nicht länger zu einer HeadXCacheFull anzeige Handling von Status 0x0A an Multiframe angepasst FB190: interne Abläufe optimiert, Parameterfehler und ParametertypError werden nun korrekt angezeigt, Parameter: E5 und AP hinzugefügt	Zustandsdiagramme für alle Befehle hinzugefügt Kennzeichnung der Ein- und Ausgänge am Baustein Handbuch an Changelog angepasst



# FABRIKAUTOMATION – SENSING YOUR NEEDS



## Zentrale weltweit

Pepperl+Fuchs GmbH  
68307 Mannheim · Deutschland  
Tel. +49 621 776-0  
E-Mail: [info@de.pepperl-fuchs.com](mailto:info@de.pepperl-fuchs.com)

## Zentrale USA

Pepperl+Fuchs Inc.  
Twinsburg, Ohio 44087 · USA  
Tel. +1 330 4253555  
E-Mail: [sales@us.pepperl-fuchs.com](mailto:sales@us.pepperl-fuchs.com)

## Zentrale Asien

Pepperl+Fuchs Pte Ltd.  
Singapur 139942  
Tel. +65 67799091  
E-Mail: [sales@sg.pepperl-fuchs.com](mailto:sales@sg.pepperl-fuchs.com)

[www.pepperl-fuchs.com](http://www.pepperl-fuchs.com)

 **PEPPERL+FUCHS**  
*SENSING YOUR NEEDS*