SIEMENS

SIMATIC Ident

Einleitung	1
Beschreibung	2
Bausteine parametrieren	3
Fehlermeldungen	4
Anhang	Α

RFID-Systeme Ident-Profil und Ident-Bausteine, Standardfunktion für Ident-Systeme

Funktionshandbuch

Rechtliche Hinweise

Warnhinweiskonzept

Dieses Handbuch enthält Hinweise, die Sie zu Ihrer persönlichen Sicherheit sowie zur Vermeidung von Sachschäden beachten müssen. Die Hinweise zu Ihrer persönlichen Sicherheit sind durch ein Warndreieck hervorgehoben, Hinweise zu alleinigen Sachschäden stehen ohne Warndreieck. Je nach Gefährdungsstufe werden die Warnhinweise in abnehmender Reihenfolge wie folgt dargestellt.

MGEFAHR

bedeutet, dass Tod oder schwere Körperverletzung eintreten **wird**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

/ WARNUNG

bedeutet, dass Tod oder schwere Körperverletzung eintreten **kann**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠VORSICHT

bedeutet, dass eine leichte Körperverletzung eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

ACHTUNG

bedeutet, dass Sachschaden eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

Beim Auftreten mehrerer Gefährdungsstufen wird immer der Warnhinweis zur jeweils höchsten Stufe verwendet. Wenn in einem Warnhinweis mit dem Warndreieck vor Personenschäden gewarnt wird, dann kann im selben Warnhinweis zusätzlich eine Warnung vor Sachschäden angefügt sein.

Qualifiziertes Personal

Das zu dieser Dokumentation zugehörige Produkt/System darf nur von für die jeweilige Aufgabenstellung qualifiziertem Personal gehandhabt werden unter Beachtung der für die jeweilige Aufgabenstellung zugehörigen Dokumentation, insbesondere der darin enthaltenen Sicherheits- und Warnhinweise. Qualifiziertes Personal ist auf Grund seiner Ausbildung und Erfahrung befähigt, im Umgang mit diesen Produkten/Systemen Risiken zu erkennen und mögliche Gefährdungen zu vermeiden.

Bestimmungsgemäßer Gebrauch von Siemens-Produkten

Beachten Sie Folgendes:

∕ WARNUNG

Siemens-Produkte dürfen nur für die im Katalog und in der zugehörigen technischen Dokumentation vorgesehenen Einsatzfälle verwendet werden. Falls Fremdprodukte und -komponenten zum Einsatz kommen, müssen diese von Siemens empfohlen bzw. zugelassen sein. Der einwandfreie und sichere Betrieb der Produkte setzt sachgemäßen Transport, sachgemäße Lagerung, Aufstellung, Montage, Installation, Inbetriebnahme, Bedienung und Instandhaltung voraus. Die zulässigen Umgebungsbedingungen müssen eingehalten werden. Hinweise in den zugehörigen Dokumentationen müssen beachtet werden.

Marken

Alle mit dem Schutzrechtsvermerk ® gekennzeichneten Bezeichnungen sind eingetragene Marken der Siemens AG. Die übrigen Bezeichnungen in dieser Schrift können Marken sein, deren Benutzung durch Dritte für deren Zwecke die Rechte der Inhaber verletzen kann.

Haftungsausschluss

Wir haben den Inhalt der Druckschrift auf Übereinstimmung mit der beschriebenen Hard- und Software geprüft. Dennoch können Abweichungen nicht ausgeschlossen werden, so dass wir für die vollständige Übereinstimmung keine Gewähr übernehmen. Die Angaben in dieser Druckschrift werden regelmäßig überprüft, notwendige Korrekturen sind in den nachfolgenden Auflagen enthalten.

Inhaltsverzeichnis

1	Einleitung	Einleitung				
2	Beschreib	oung	7			
	2.1	Anwendungsbereich und Merkmale	7			
	2.2	Bausteinfunktion	10			
	2.3	Abgrenzung zu anderen Programmbausteinen	11			
3	Bausteine	parametrieren	13			
	3.1	Übersicht der Ident-Bibliothek	13			
	3.2	Datentyp "IID_HW_CONNECT" parametrieren	15			
	3.3	Allgemeiner Aufbau der Funktionsbausteine	17			
	3.4	Ident-Bausteine programmieren	21			
	3.4.1	Basis-Bausteine				
	3.4.1.1	Read				
	3.4.1.2	Read_MV				
	3.4.1.3	Reset_Reader				
	3.4.1.4	Set_MV_Program				
	3.4.1.5	Write				
	3.4.2	Erweiterte Bausteine				
	3.4.2.1	Config_Upload/Download				
	3.4.2.2	Inventory				
	3.4.2.3	Read_EPC_Mem				
	3.4.2.4	Read_TID				
	3.4.2.5	Read_UID				
	3.4.2.6	Set Ant				
	3.4.2.7	Set_Param				
	3.4.2.8	Write_EPC_ID				
	3.4.2.9	Write_EPC_Mem				
	3.4.2.10	AdvancedCMD				
	3.4.3	Reset-Bausteine				
	3.4.3.1	Reset_MOBY_D				
	3.4.3.2	Reset_MOBY_U				
	3.4.3.3	Reset MV				
	3.4.3.4	Reset_RF200				
	3.4.3.5	Reset_RF300				
	3.4.3.6	Reset_RF600				
	3.4.3.7	Reset_Univ				
	3.4.4	Status-Bausteine				
	3.4.4.1	Reader_Status				
	3.4.4.2	Tag_Status				

	3.5	Ident-Profil programmieren	65
	3.5.1	Umstieg auf Ident-Bausteine/-Profil	65
	3.5.2	Aufbau des Ident-Profils	68
	3.5.3	Datenstruktur des Ident-Profils	72
	3.5.4	Befehle des Ident-Profils	74
	3.5.4.1	Befehlsstruktur	76
	3.5.4.2	Befehle	
	3.5.4.3	Erweiterte Befehle für Code-Lesesysteme (MV400)	
	3.5.4.4	Wirkung der Befehle	
	3.5.4.5	Befehlsbearbeitung	
	3.5.4.6	Neustart und Wiederanlauf parametrieren	
	3.5.4.7	Verkettung	
	3.5.4.8	Befehlswiederholung	101
	3.6	Transponder-Adressierung	107
4	Fehlerme	eldungen	115
	4.1	Aufbau des Ausgangsparameters "STATUS"	115
	4.2	STEP 7 - Fehlermeldungen	116
	4.3	Fehler vom Kommunikationsmodul/Reader	116
	4.4	Fehler vom Rückwandbus	124
	4.5	Warnungen	126
Α	Anhang		127
	A.1	Versteckte Statusparameter	127
	A.2	Service & Support	129

Einleitung

Zweck dieser Dokumentation

Als Schnittstelle zu den Kommunikationsdiensten stehen für Ihr Anwenderprogramm vorgefertigte Programmbausteine (FCs und FBs) zur Verfügung. Dieses Handbuch enthält Beschreibungen der Ident-Bausteine und des Ident-Profils, mit deren Hilfe Sie verschiedene Ident-Systeme inbetriebnehmen und parametrieren können.

Es richtet sich sowohl an Programmierer und Tester, als auch an Service- und Wartungstechniker.

Gültigkeitsbereich dieser Dokumentation

Diese Dokumentation ist gültig für das Ident-Profil bzw. die Ident-Bausteine und beschreibt den Lieferzustand ab Oktober 2014.

Einordnung in die Dokumentationslandschaft

Weitere Informationen zu den in diesem Handbuch genannten Bausteinen und Ident-Geräten, finden Sie im Internet

(<u>http://support.automation.siemens.com/WW/view/de/43532183/133300</u>) in den folgenden Handbüchern:

- FB 45
- FB 55
- SIMATIC RF620R/RF630R
- SIMATIC RF650R/RF680R/RF685R
- Anschaltmodul ASM 456
- Kommunikationsmodul RF120C
- Kommunikationsmodul RF170C
- SIMATIC RF180C
- SIMATIC MV420/MV440

Spezifikationen

Die in dem Handbuch enthaltenen Ident-Bausteine setzen auf dem Protokoll "Proxy Ident Function Block" auf. Die Spezifikation des "Proxy Ident Function Block" erhalten Sie über die PROFIBUS-Nuterzerorganisation.

Marken

SIMATIC ®, SIMATIC RF ®, MOBY ®, RF-MANAGER ® und SIMATIC Sensors ® sind eingetragene Marken der Siemens AG.

Abkürzungen und Namenskonventionen

Innerhalb dieser Dokumentation werden folgende Begriffe/Abkürzungen synonym verwendet:

Reader Schreib-/Lesegerät (SLG)

Transponder, Tag Mobiler Datenspeicher (MDS), Datenträger

Kommunikationsmodul (CM) Anschaltmodul (ASM)

Historie

Folgende Ausgabe(n) des Funktionshandbuchs wurden bisher veröffentlicht:

Ausgabe	Bemerkung
10/2014	Erstausgabe

Beschreibung

2.1 Anwendungsbereich und Merkmale

Die Ident-Bibliothek enthält STEP 7-Funktionen für Identifikatinssysteme. Die Bausteine bestehen aus Ident-Bausteinen und dem Ident-Profil. Das Ident-Profil kann in den SIMATIC-Steuerungen S7-300, S7-400, S7-1200 und S7-1500 für diverse Kommunikationsmodule, RFID-Reader und Code-Lesesysteme eingesetzt werden. Projektierbar ist es mit STEP 7 V5.5 und STEP 7 Basic / Professional V13. Die Ident-Bausteine setzen auf dem Ident-Profil auf und sind mit STEP 7 Basic / Professional ab V13 projektierbar.



Bild 2-1 Über die Ident-Bibliothek projektierbare Baugruppen: ASM 456, RF170C, RF180C, RF120C, RF680R/RF685R, MV440 und MV420

2.1 Anwendungsbereich und Merkmale

Das Ident-Profil und die Ident-Bausteine können gleichermaßen in unterschiedlichen Konfigurationen betrieben werden.

Tabelle 2- 1 Über die Ident-Bibliothek projektierbare Konfigurationen

Baugruppen	Bussysteme		Steuerungen		
	PROFIBUS	PROFINET	S7-300/-400	S7-1200/-1500	
ASM 456	✓		✓	✓	
RF170C	✓	✓	✓	✓	
RF180C		✓	✓	✓	
RF120C				√ 1)	
RF680R/RF685R		✓	✓	✓	
MV440/MV420	✓ ²⁾	1	√	✓	

¹⁾ nur S7-1200

Das Mischen dieser Konfigurationen und das Anschließen unterschiedlicher Kommunikationsmodule ist ebenso möglich.

Unterschied zwischen Ident-Bausteinen und Ident-Profil

Das Ident-Profil ist ein einziger komplexer Baustein, der alle Befehle und Funktionen für RFID- und Code-Lesesysteme enthält. Die Ident-Bausteine stellen eine vereinfachte Schnittstelle des Ident-Profils dar. Jeder Ident-Bausteine enthält einen einzelnen Befehl des Ident-Profils.

Grundsätzlich sollten Sie sich am Anfang entscheiden welche Bausteine Sie verwenden möchten - Ident-Profil oder Ident-Bausteine. Sie können allerdings nur eine von beiden Varianten nutzen. Das Mischen von Ident-Profil und Ident-Bausteinen pro Kanal ist nicht möglich!

²⁾ über Kommunikationsmodul

Die folgende Tabelle gibt Ihnen einen Überblick über die Unterschiede.

Tabelle 2- 2 Unterschiede zwischen Ident-Bausteine und Ident-Profil

Ident-Bausteine	Ident-Profil	
Pro Befehl ein Baustein	Voller Funktionsumfang in einem Baustein	
Einfache Programmierung	Komplexe Programmierung	
Systemspezifische Bausteine		
Unterstützter Befehlsumfang:	Unterstützter Befehlsumfang:	
Reader-Status	Alle im Reader implementierten Befehle wie z. B.	
Inventory	Inventory	
Tag-Status	Physical-Read	
Read	Get-Blacklist	
Write	Matchstring-Funktionen (nur bei MV)	
Set-Ant	•	
Write-ID		
Reset-Reader		
•		
Unterstützter Funktionsumfang:	Unterstützter Funktionsumfang:	
AdvancedCMD	Repeat	
für verkettete Befehlsstrukturen (das Aufset-	Verkettung	
zen der einzelnen Befehle einer Kette ist identisch mit dem Ident-Profil)		
uson mit dem ident-i folli)		

Ausführliche Informationen zu den Bausteinen finden Sie in den Kapiteln "Ident-Bausteine programmieren (Seite 21)" und "Ident-Profil programmieren (Seite 65)".

Empfehlung zur Auswahl der Bausteine

Wenn die Ident-Bausteine Ihre Funktionsanforderungen abdecken, nutzen Sie diese. Die Ident-Bausteine sind einfacher zu programmieren und können meist ohne weitere Dokumentation parametriert werden. Das Programm wird leichter lesbar und die Programmierung ist mit weniger Aufwand verbunden.

Das Ident-Profil ist ein komplexer Baustein. Wir empfehlen ausschließlich geschulten Anwendern oder bei speziellen Anforderungen das Ident-Profil zu verwenden.

2.2 Bausteinfunktion

Die Programmbausteine dienen als Kommunikationsschnittstelle zwischen einem Ident-Gerät (z. B. ASM 456) und dem Anwenderprogramm. Die Bausteine unterstützen dabei folgende Funktionen:

- Konfiguration
- Befehlsbearbeitung
- Lesen und Schreiben von Daten
- Diagnose

Das Ident-Profil ist ein einziger komplexer Baustein, der alle Befehle und Funktionen für RFID-Systeme und Code-Lesesysteme enthält. Die Ident-Bausteine stellen eine vereinfachte Schnittstelle des Ident-Profils dar. Jeder Ident-Baustein enthält einen einzelnen Befehl des Ident-Profils.

Die Größe der Datenpuffer "IDENT_DATA" (bei den Ident-Bausteinen), "TXREF" und "RXREF" (bei dem Ident-Profile) können variabel sein. Die Parameter sind unter S7-300 / S7-400 als "Any"-Pointer und unter S7-1200 / S7-1500 als "Variant" definiert.

Tabelle 2-3 Unterschied

S7-300/-400 ("Any"-Pointer)	S7-1200/-1500 ("Variant")
"IDENT_DATA", "TXREF", "RXREF": Es können unterschiedlich lange Arrays jeden Typs, mitgelieferte Status-UDTs und selbst definierte UDTs	"IDENT_DATA", "TXREF", "RXREF": Es können nur Arrays vom Typ Byte angelegt werden. Die Länge ist variabel.
angelegt werden.	Ausnahme: Bei den Ident-Bausteinen "Reader-Status" und "Tag-Status" können an "IDENT_DATA" auch die mitgelieferten Status-Datentypen angelegt werden.

2.3 Abgrenzung zu anderen Programmbausteinen

Von Programmbausteinen unterstütze Funktionen

Die folgende Tabelle zeigt eine Übersicht der von den Programmbausteinen unterstützten Funktionen.

Tabelle 2-4 Unterstütze Funktionen der Programmbausteine

Programm-	von Programmbaustein unterstütze Funktionen							
baustein	Singletag	Multitag	Normal- adressierung	File- handler	PROFIBUS	PROFINET	MV	
FB 45	✓		1		✓	✓	✓	Empfohlener Baustein für Singletag- Applikationen
FB 55	✓	✓	1		✓	✓		Empfohlener Baustein für Multitag- Applikationen
FC 56	✓	✓		✓	✓			Empfohlener Baustein für Filehandler
Normprofil V1.19	✓	✓	✓	✓	✓	✓	✓	Empfohlener Baustein für Fremdsteuerungen
Ident-Profil	√	√	✓		✓	✓	√	Baustein auf Grundla- ge der PNO- Spezifikation (Basiert auf Normprofil V1.19, jedoch ohne Filehandler)
Ident- Bausteine	1	✓	✓		✓	1	1	Auf dem Ident-Profil basierende Bausteine
FC 44	✓		1		1			Nur für RF160C
Applikations- bausteine RF160C	✓		1		1			Nur für RF160C
Applikations- bausteine IO-Link	✓		1		✓	1		Nur für RF200 IO-Link

2.3 Abgrenzung zu anderen Programmbausteinen

Kompatible Programmbausteine

Die folgende Tabelle zeigt die zu den Anschaltmodulen/Kommunikationsmodulen kompatiblen Programmbausteine.

Tabelle 2-5 Kompatible Programmbausteine

Ident-Systeme	kompatible Programmbausteine in Verbindung mit							
	S7-300 / S7-400 und STEP 7 Classic V5.5	S7-300 / S7-400 und STEP 7 Basic/Professional	S7-1200 / S7-1500 und STEP 7 Basic/Professional					
ASM 456	FB 45	FB 45	Ident-Profil					
	FB 55	FB 55	Ident-Bausteine					
	FC 56	FC 56	PIB_1200_UID_001KB					
	Normprofil V1.19	Ident-Profil	PIB_1200_UID_032KB					
	Ident-Profil							
ASM 475	FB 45	FB 45						
	FB 55	FB 55						
SIMATIC RF120C			Ident-Profil					
			Ident-Bausteine					
			PIB_1200_UID_001KB					
			PIB_1200_UID_032KB					
SIMATIC RF160C	FC 44	FC 44	Applikationsbausteine für					
	Applikationsbausteine für RF160C	Applikationsbausteine für RF160C	RF160C					
SIMATIC RF170C	FB 45	FB 45						
	FB 55	FB 55						
SIMATIC RF180C	FB 45	FB 45	Ident-Profil					
	FB 55	FB 55	Ident-Bausteine					
	Normprofil V1.19	Ident-Profil	PIB_1200_UID_001KB					
	Ident-Profil		PIB_1200_UID_032KB					
SIMATIC	Ident-Profil	Ident-Profil	Ident-Profil					
RF680R/RF685R		Ident-Bausteine	Ident-Bausteine					
SIMATIC	FB 79	FB 79	Ident-Profil					
MV420/MV440	Normprofil V1.19	Ident-Profil	PIB_1200_UID_001KB					
	Ident-Profil		PIB_1200_UID_032KB					

Bausteine parametrieren

3.1 Übersicht der Ident-Bibliothek

Bie Ident-Bibliothek mit dem Ident-Profil und den Ident-Bausteinen sind ab der Version V13 SP1 in STEP 7 integriert. Sie finden die Bausteine in dem Register "Anweisungen" unter "Optionale Pakete" > "SIMATIC Ident".

Die nachfolgende Tabelle gibt Ihnen einen Überblick über die aktuell vorhandenen Bausteine.

Tabelle 3- 1 Übersicht der Ident-Bibliothek

Einordnung	Einordnung		Symbolischer Name	Beschreibung
Anweisungen/	Ident-	Basis-	Read	Mit Hilfe dieser Bausteine Können Sie die
Bausteine	Bausteine	Bausteine	Write	Kommunikation mit den Ident-Systemen ein-
	I Reset Reader I ' '	fach programmieren.		
			Reader_Status	Die Basis-Bausteine beinhalten all die Bausteine, die sehr häufig zur Anwendung kom-
			Tag_Status	men.
			Read_MV	
			Set_MV_Program	
		Erweiterte	Config_Download	Mit Hilfe dieser Bausteine Können Sie die
		Bausteine	Config_Upload	Kommunikation mit den Ident-Systemen ein-
			Inventory	fach programmieren. Die erweiterten Bausteine stellen weitere.
			Read_EPC_Mem	seltener benötigte Funktionen für die Bedie-
			Read_TID	nung des Ident-Systems zur Verfügung.
			Read_UID	
			Set_ANT_RF300	
			Set_ANT_RF600	
			Set_Param	
			Write_EPC_ID	
			Write_EPC_Mem	
			AdvancedCmd	Erweiterter Befehlsvorrat. Mit dem Baustein "AdvancedCmd" ist es möglich, auf weitere Befehle aus dem Ident-Befehlsvorrat zuzugreifen und verkettete Befehle auszuführen.
		Reset-	Reset_MOBY_D	Mit Hilfe dieser Bausteine Können Sie die
		Bausteine	Reset_MOBY_U	Kommunikation mit den Ident-Systemen ein-
			Reset_MV	fach programmieren. Die Reset-Bausteine dienen zur einfachen
			Reset_RF200	Initialisierung der Ident-Systeme, wenn der
			Reset_RF300	Baustein "Reset_Reader" nicht von dem Ident-
			Reset_RF600	System unterstützt wird.
			Reset_Univ	

3.1 Übersicht der Ident-Bibliothek

Einordnung		Symbolischer Name	Beschreibung
	Ident-Profil	Ident_Profile	Für Experten stehen diese Bausteine zur Verfügung, um komplexe Befehlsstrukturen optimal in den eigenen Programmablauf einzubinden. Zusätzlich ist es möglich Repeat-Kommandos und Verkettung zu nutzen.
PLC-Datentypen	Systemdatentypen	IID_HW_CONNECT	Datentyp für alle Bausteine zur physikalischen Adressierung von Kommunikationsmodul und Reader und zum Synchronisieren der Funktionsbausteine die pro Reader verwendet werden.
		IID_CMD_STRUCT	Datentyp für das Ident-Profil, zum Parametrieren der Befehle.
	Statusdatentypen	IID_READER_STATUS_ 84_MOBY_U	Datentypen für das Ergebnis von "Reader_Status" mit dem jeweiligen Attribut.
		IID_READER_STATUS_ 81_RF200_300_U	Die Datentypen helfen dabei, die vom Reader empfangenen Daten, komfortabel zu interpre-
		IID_READER_STATUS_ 86_RF300	tieren und direkt weiterzubearbeiten ohne Datentyp-Konvertierungen vorzunehmen.
		IID_READER_STATUS_ A0_A1_RF600	
		IID_READER_STATUS_ 87_RF600	
		IID_READER_STATUS_ 88_RF600	
		IID_READER_STATUS_ 89_RF68xR	
		IID_TAG_STATUS_ 83_ISO	Datentypen für das Ergebnis von "Tag_Status" mit dem jeweiligen Attribut.
		IID_TAG_STATUS_ 80_MOBY_U	Die Datentypen helfen dabei, die vom Reader empfangenen Daten, komfortabel zu interpre-
		IID_TAG_STATUS_ 04_RF300	tieren und direkt weiterzubearbeiten ohne Datentyp-Konvertierungen vorzunehmen.
		IID_TAG_STATUS_ 82_RF300	
		IID_TAG_STATUS_ 84_RF600	
		IID_TAG_STATUS_ 85_RF600	
IID_INVENTORY_ 00_MOBY_U IID_INVENTORY_ A0_A1_RF600		Datentypen für das Ergebnis von "Inventory" mit dem jeweiligen Attribut.	
			Die Datentypen helfen dabei, die vom Reader empfangenen Daten, komfortabel zu interpre-
		IID_INVENTORY_ 82_83_RF600	tieren und direkt weiterzubearbeiten ohne Datentyp-Konvertierungen vorzunehmen.
		IID_INVENTORY_ 85_RF600	
		IID_INVENTORY_ 8x_9x_RF6_MD	

3.2 Datentyp "IID HW CONNECT" parametrieren

Bevor Sie mit dem Parametrieren der Bausteine beginnen können, müssen Sie zuerst eine Variable des PLC-Datentyps "IID_HW_CONNECT" anlegen. Mit Hilfe des PLC-Datentyps "IID_HW_CONNECT" wird das Ident-System bzw. ein Kanal des Ident-Systems adressiert.

Adressierung der Ident-Geräte

Bei der Arbeit mit allen Anweisungen/Bausteinen, benötigen Sie den Datentyp "IID_HW_CONNECT" zur Reader-Adressierung. Die Befehlsparametrierung am Ident-Profil wird durch die Ident-Bausteine übernommen. Das Ident-Profil sowie der Baustein "AdvancedCMD" benötigen zusätzlich den Datentyp "IID_CMD_STRUCT" zur Parametrierung der einzelnen Befehle. Abhängig davon, ob Sie mit dem Ident-Profil oder den Ident-Bausteinen arbeiten, müssen Sie diese Datentypen wie in den folgenden Kapiteln beschrieben einbinden und parametrieren.

Parametrieren des Datentyps "IID_HW_CONNECT"

Gehen Sie folgendermaßen vor, um den Datentyp "IID_HW_CONNECT" für einen Kanal zu parametrieren:

- 1. Doppelklicken Sie in der Projektnavigation in dem Ordner "Programmbaustein" auf den Eintrag "Neuen Baustein anlegen".
- 2. Klicken Sie auf die Schaltfläche "Datenbaustein" und weisen Sie dem Baustein einen Namen zu.
- Bestätigen Sie die Eingabe mit "OK".
 - Der Datenbaustein wird geöffnet.
- 4. Legen Sie eine neue Variable an, indem Sie in der Spalte "Name" einen Variablennamen eintragen.
- 5. Wählen Sie in der Spalte "Datentyp" den Datentyp "IID_HW_CONNECT" aus.



Bild 3-1 Datenbaustein anlegen

- 6. Geben Sie die Adress-Daten des Ident-Gerätes bzw. des Kanals an.
 - HW ID: Hardware-Kennung der Baugruppe (nur bei S7-1200 und S7-1500)
 - CM_CHANNEL: Kanal des Reader am CM bzw. der Antenne
 - LADDR: E/A-Adresse der Baugruppe

Die Werte der Parameter "HW_ID" und "LADDR" können Sie in der Gerätekonfiguration in den Eigenschaften des Kommunikationsmoduls/Readers auslesen. Tragen Sie die ausgelesenen Parameterwerte in der Spalte "Startwert" der zugehörigen Parameter ein. Das Auslesen der Parameterwerte wird im Folgenden beschrieben.

Gehen Sie folgendermaßen vor, um die Parameterwerte "HW_ID" und "LADDR" für einen Kanal auszulesen (nur bei RF180C und ASM 456):

- 1. Öffnen Sie die Gerätesicht.
- Doppelklicken Sie auf das Kommunikationsmodul.
 Das Eigenschaftenfenster des CMs wird geöffnet.
- Markieren Sie in dem Register "Geräteübersicht" das Modul "2x RS422 Kanäle RFID_1".
 Die in dem Register angezeigte E/A-Adresse, entspricht der "LADDR".
 Beachten Sie, dass die Eingangs- und Ausgangsadresse den gleichen Wert haben müssen.
- 4. In dem Register "Eigenschaften" > "Allgemein" > "HW-Kennung" finden Sie die HW-Kennung, welche der "HW_ID" entspricht.

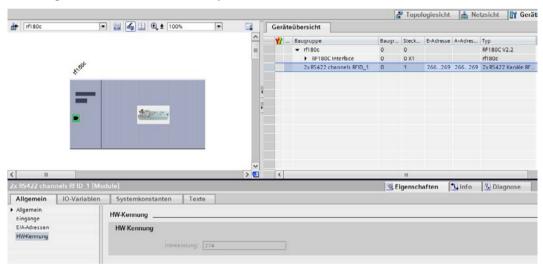


Bild 3-2 Der Parameter "HW-Kennung"

5. Übertragen Sie die Werte von "LADDR" und "HW_ID" in den PLC-Datentyp "IID_HW_CONNECT" des zu parametrierenden Readers.

Hinweis

User Mode einstellen

Beachten Sie, dass Sie in den Eigenschaften des Kommunikationsmoduls, dem Parameter "User Mode" den Wert "RFID-Normprofil" zuweisen und den passenden MOBY Mode auswählen.

Bei allen anderen Kommunikationsmodulen/Readern finden Sie die beiden Parameter direkt in den Eigenschaften der Baugruppe.

Der Datentyp "IID_HW_CONNECT" ist nun für einen Kanal angelegt und adressiert. Für jeden weiteren Reader/Kanal wiederholen Sie diesen Vorgang. Wenn Sie einen anderen Kanal des Readers/CMs benutzen wollen, stellen Sie dies über den Parameter "CM_CHANNEL" ein. Die Parameter "HW_ID" und "LADDR" bleiben bei allen Kanälen/Readern/Antennen gleich.

Die Bibliothek ist nun eingebunden und die notwendigen Bausteine und Datentypen sind in Ihrem Projekt angelegt. Der Datentyp "IID_HW_CONNECT" ist ebenfalls angelegt und adressiert. Nun können Sie mit dem Programmieren der Bausteine beginnen.

Hinweis

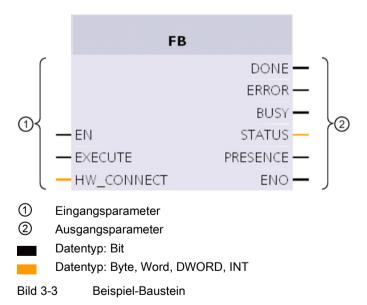
"IID_CMD_STRUCT" projektieren

Wenn Sie mit dem Ident-Profil oder mit dem Bausteine "AdvancedCmd" arbeiten, müssen Sie zusätzlich in dem bereits angelegten Datenbaustein ein weiteres Element mit dem Datentyp "IID_CMD_STRUCT" (Array [1...10]) anlegen.

3.3 Allgemeiner Aufbau der Funktionsbausteine

Aufbau der Bausteine anhand des Beispiel-Bausteins "FB"

Die nachfolgende Grafik zeigt einen Beispiel-Baustein mit Eingangs- und Ausgangsparametern wie sie bei allen Bausteinen in der gleichen Weise vorhanden sind.



3.3 Allgemeiner Aufbau der Funktionsbausteine

Eingangsparameter

EN

Freigabeeingang

EXECUTE

An diesen Eingang muss eine positive Flanke angelegt werden, damit der Baustein den Befehl ausführt.

HW CONNECT

Globaler Parameter vom Typ "IID_HW_CONNECT" um den Kanal/Reader zu adressieren und die Bausteine zu synchronisieren. Dieser Parameter muss einmal für jeden Kanal/Reader erstellt und adressiert werden. "HW_CONNECT" muss immer an die Bausteine übergegeben werden, um den entsprechenden Kanal/Reader zu adressieren.

Ausgangsparameter

DONE (BOOL)

Der Auftrag wurde abgearbeitet. Bei einem positiven Ergebnis wird dieser Parameter gesetzt.

• ERROR (BOOL)

Der Auftrag wurde mit einem Fehler beendet. Der Fehlercode wird in Status angezeigt.

• BUSY (BOOL)

Der Auftrag wird gerade ausgeführt.

STATUS (DWORD)

Anzeige der Fehlermeldung, wenn das "ERROR"-Bit gesetzt wurde.

PRESENCE (BOOL)

Über dieses Bit wird die Anwesenheit eines Transponders angezeigt. Der angezeigte Wert wird bei jedem Aufruf des Bausteins aktualisiert.

Dieser Parameter kommt nicht bei Code-Lesesystem spezifischen Bausteinen vor.

ENO

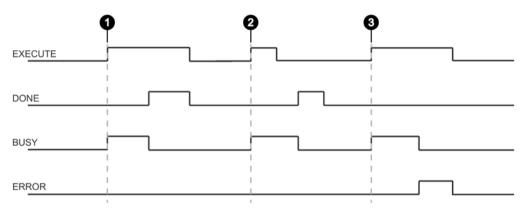
Freigabeausgang

Allgemeiner Ablauf beim Aufruf der Bausteine

Hinweis

Unterschiedliche Abläufe bei Ident-Profil und Normprofil V1.19

Beachten Sie, dass der Ablauf des Ident-Profils nicht dem des Normprofils V1.19 entspricht.



- Fall ① Durch Setzen von EXECUTE (EXECUTE = 1) wird die Funktion/Anweisung gestartet. Wurde der Auftrag erfolgreich beendet (DONE = 1) müssen Sie EXECUTE wieder zurücksetzen. DONE wird zeitgleich zurückgesetzt.
- Fall ② EXECUTE wird nur für einen Zyklus gesetzt. Sobald BUSY gesetzt ist können Sie EXECUTE wieder zurücksetzen. Wurde der Auftrag erfolgreich beendet, wird DONE für einen Zyklus gesetzt.
- Fall ③ Handhabung wie Fall 1, jedoch mit Fehlerausgabe. Sobald ERROR gesetzt ist, steht in der Ausgabe STATUS der genaue Fehlercode. ERROR und STATUS behalten ihren Wert solange bei, wie EXECUTE gesetzt ist bzw. für einen Zyklus, wenn EXECUTE vor Beendigung des Bausteins zurückgesetzt wurde.
- Bild 3-4 Allgemeiner Ablauf beim Aufruf der Bausteine

Funktionsweise der Bausteine:

Sie können immer nur einen Befehl an den Reader bzw. das Kommunikationsmodul versenden. Sie können jedoch zwei oder mehrere Bausteine gleichzeitig starten. Die Bausteine werden dann in der Reihenfolge des Aufrufs abgearbeitet.

Dies gilt nicht bei den Reset-Bausteinen. Wird ein Reset-Befehl ausgeführt, wird der zu diesem Zeitpunkt aktive Befehl abgebrochen.

3.3 Allgemeiner Aufbau der Funktionsbausteine

Baustein anlegen

Voraussetzung

Der Datentyp "IID_HW_CONNECT" wurde parametriert.

Gehen Sie folgendermaßen vor, um einen Baustein einzubinden und den Aufruf zu parametrieren:

- 1. Öffnen Sie den von Ihnen erstellten Programmbaustein, durch einen Doppelklick im Register "Projektnavigation" > "Programmbausteine".
- 2. Ziehen Sie den gewünschten Baustein aus dem Bibliotheksregister in den Programmbaustein.
- 3. Geben Sie am Eingangsparameter "HW_CONNECT" die Variable an, die Sie zuvor angelegt haben.

Der Baustein wird aufgerufen und mit dem entsprechenden Kanal verbunden.

Hinweis

Arbeit mit mehreren Kanälen

Wenn Sie mit mehreren Kanälen arbeiten, müssen sie darauf achten, dass für jeden Kanal der Baustein mit einem eigenen Instanz-DB aufgerufen wird.

Hinweis

Arbeiten mit dem Ident-Profil oder mit dem Baustein "AdvancedCmd"

Wenn Sie mit dem Ident-Profil oder mit dem Baustein "AdvancedCmd" arbeiten, müssen Sie zusätzlich den Eingangsparameter "CMDREF" mit einer Variablen des Datentyp "IID_CMD_STRUCT" (Array [1...10]) verbinden.

3.4.1 Basis-Bausteine

3.4.1.1 Read

Der Baustein "Read" liest die Anwenderdaten vom Transponder aus und stellt diese im Puffer "IDENT_DATA" bereit. Die physikalische Adresse und die Länge der Daten, wird über die Parameter "ADR_TAG" und "LEN_DATA" übergeben. Bei den Readern RF68xR liest der Baustein die Daten aus der Memory-Bank 3 (USER-Bereich) aus. Der eindeutige Zugriff auf einen bestimmten Transponder erfolgt über die "EPCID_UID" und "LEN_ID".

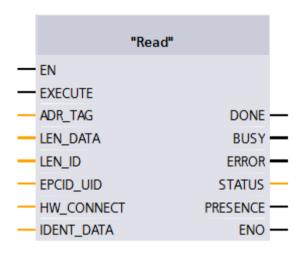


Bild 3-5 Baustein "Read"

Tabelle 3-2 Erläuterung zum Baustein "Read"

Parameter	Datentyp	Default-Werte	Beschreibung
ADR_TAG	DWord	DW#16#0	Physikalische Adresse auf dem Transponder ab der gelesen wird. Weitere Informationen zur Adressie- rung finden Sie im Kapitel "Transponder-Adressierung (Sei- te 107)".
			Bei MV: Ab Adresse "0" steht die Länge des gelesenen Codes (2 Byte). Ab Adresse "2" steht der gelesene Code. 1)
LEN_DATA	Word	W#16#0	Länge der zu lesenden Daten
LEN_ID	Byte	B#16#0	Länge der EPC-ID/UID
			Default-Wert: 0x00 ≜ unspezifizierter Singletag-Zugriff (RF680R, RF685R)
EPCID_UID	Array[162] of Byte	0	Puffer für bis zu 62 Byte EPC-ID, 8 Byte UID oder 4 Byte Handle-ID.
			2-62-Byte EPC-ID wird an Anfang des Puffer eingetragen (Länge wird durch "LEN_ID" beschrieben)
			8-Byte UID wird an Anfang des Puffers eingetragen ("LEN_ID = 8")
			4-Byte Handle-ID muss im Array- Element [5]-[8] eingetragen wer- den ("LEN_ID = 8")
			Default-Wert: 0x00 ≜ unspezifizierter Singletag-Zugriff (RF620R, RF630R, RF640R)
IDENT_DATA	Any / Variat	0	Datenpuffer in dem die gelesenen Daten abgelegt werden.
			Hinweis: Bei Variant ist derzeit nur ein "Ar- ray_of_Byte" mit variabler Länge an- legbar. Bei Any können zusätzlich auch andere Datentypen/UDTs ange- legt werden.

Weitere Informationen zur Arbeit mit Code-Lesesystemen, finden Sie in der Betriebsanleitung "SIMATIC MV420 / SIMATIC MV440".

3.4.1.2 Read_MV

Der Baustein "Read_MV" liest das Leseergebnis einer Kamera aus. Die Länge der zu lesenden Daten ermittelt die Kamera automatisch aus der Länge des angelegten Empfangpuffers. Die tatsächliche Länge des Leseergebnisses wird in dem Ausgangsparameter "LEN_DATA" ausgegeben. Die Daten werden im Datentpuffer "IDENT_DATA" abgelegt. Ist der Puffer zu klein, erscheint die Fehlermeldung "0xE7FE0400" und die zu erwartende Länge wird an "LEN_DATA" ausgegeben.

Um eine optimale Geschwindigkeit zu erreichen, empfehlen wir Ihnen, die Länge des Datentyps "IDENT_DATA" so anzupassen, dass diese möglichst nah der maximal erwarteten Länge des Leseergebnisses (2 Byte Codelänge + gelesener Code) entspricht.

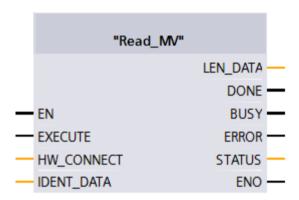


Bild 3-6 Baustein "Read_MV"

Tabelle 3-3 Erläuterung zum Baustein "Read_MV"

Parameter	Datentyp	Default-Werte	Beschreibung
IDENT_DATA	Any / Variant	0	Leseergebnis
			In Byte 0 und 1 steht die Länge des gelesenen Codes.
LEN_DATA	Word	W#16#0	Länge des Leseergebnis ≙ 2 Byte Codelänge + gelesener Code
			Hinweis: Bei Variant ist ein "Array_of_Byte" mit variabler Länge anlegbar. Bei Any können zusätzlich auch andere Datentypen/UDTs angelegt werden.

3.4.1.3 Reset_Reader

Der Baustein "Reset_Reader" ist aktuell nur in Verbindung mit den Readern RF680R und RF685R oder dem Kommunikationsmodul RF120C und angeschlossenem Reader verwendbar.

Mit Hilfe des Bausteins "Reset_Reader" können Sie alle Reader-Typen der Siemens RFID-Systeme zurücksetzen. Dabei werden alle Reader auf die Einstellungen zurückgesetzt, die in der Gerätekonfiguration des RF120C hinterlegt sind bzw. im Reader RF68x via WBM konfiguriert wurden. Der Baustein "Reset_Reader" besitzt keine gerätespezifischen Parameter und wird über den Parameter "EXECUTE" ausgeführt.

Beschreibungen weiterer Reset-Bausteine für den Betrieb mit den Kommunikationsmodulen RF180C und ASM 456 oder Code-Lesesystemen finden sie im Kapitel "Reset-Bausteine (Seite 48)".

Mit dem Baustein "Reset_Reader" und den anderen Reset-Bausteinen können Sie jederzeit jeden laufenden Ident-Baustein unterbrechen. Diese Bausteine werden dann mit "DONE = true" und "ERROR = false" beendet.

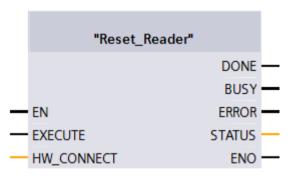


Bild 3-7 Baustein "Reset_Reader"

3.4.1.4 Set_MV_Program

Mit Hilfe des Bausteins "Set_MV_Program" können Sie das Programm in einer Kamera wechseln. Über den Parameter "PROGRAM" wird die gewünschte Programmnummer übergeben.

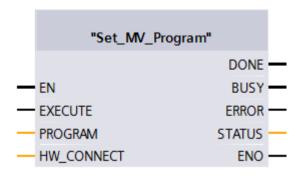


Bild 3-8 Baustein "Set_MV_Program"

Tabelle 3-4 Erläuterung zum Baustein "Set_MV_Program"

Parameter	Datentyp	Default-Werte	Beschreibung
PROGRAM	Byte	B#16#1	Programmnummer
			Wertebereich: 0x01 0x0F

3.4.1.5 Write

Der Baustein "Write" schreibt die Anwenderdaten aus dem Puffer "IDENT_DATA" auf den Transponder. Die physikalische Adresse und die Länge der Daten, wird über die Parameter "ADR_TAG" und "LEN_DATA" übergeben. Bei den Readern RF68xR schreibt der Baustein die Daten in die Memory-Bank 3 (USER-Bereich). Der eindeutige Zugriff auf einen bestimmten Transponder erfolgt über die "EPCID UID" und "LEN ID".

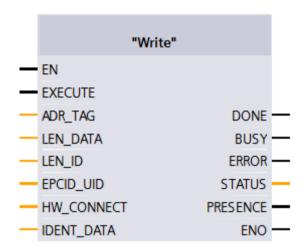


Bild 3-9 Baustein "Write"

Tabelle 3-5 Erläuterung zum Baustein "Write"

Parameter	Datentyp	Default-Werte	Beschreibung
ADR_TAG	DWord	DW#16#0	Physikalische Adresse auf dem Transponder ab der geschrieben wird. Weitere Informationen zur Adressie- rung finden Sie im Kapitel "Transponder-Adressierung (Sei- te 107)".
			Bei MV: Adresse ist immer 0. 1)
LEN_DATA	Word	W#16#0	Länge der zu schreibenden Daten
LEN_ID	Byte	B#16#0	Länge der EPC-ID/UID
			Default-Wert: 0x00 ≙ unspezifizierter Singletag-Zugriff (RF680R, RF685R)

Parameter	Datentyp	Default-Werte	Beschreibung
EPCID_UID	Array[162] of Byte	0	Puffer für bis zu 62 Byte EPC-ID, 8 Byte UID oder 4 Byte Handle-ID.
			2-62-Byte EPC-ID wird an Anfang des Puffer eingetragen (Länge wird durch "LEN_ID" beschrieben)
			8-Byte UID wird an Anfang des Puffers eingetragen ("LEN_ID = 8")
			4-Byte Handle-ID muss im Array- Element [5]-[8] eingetragen wer- den ("LEN_ID = 8)
			Default-Wert: 0x00 ≜ unspezifizierter Singletag-Zugriff (RF620R, RF630R, RF640R)
IDENT_DATA	Any / Variant	0	Datenpuffer mit den zu schreibenden Daten.
			Bei MV: Erstes Byte codiert den ent- sprechenden MV-Befehl. ¹⁾
			Hinweis: Bei Variant ist derzeit nur ein "Ar- ray_of_Byte" mit variabler Länge an- legbar. Bei Any können zusätzlich auch andere Datentypen/UDTs ange- legt werden.

Weitere Informationen zur Arbeit mit Code-Lesesystemen, finden Sie in der Betriebsanleitung "SIMATIC MV420 / SIMATIC MV440".

3.4.2 Erweiterte Bausteine

3.4.2.1 Config_Upload/-_Download

Mit Hilfe der Bausteine "Config_Upload" und "Config_Download" können Sie über das Steuerungsprogramm die Konfiguration der Reader RF680R/RF685R auslesen ("Config_Upload") oder schreiben ("Config_Download").

Die Konfigurationsdaten sind nicht interpretierbare Daten. Speichern Sie die Daten in der Steuerung, um im Falle eines Gerätetauschs diese wieder auf den Reader zu schreiben. Die Bytes 6-9 (siehe nachfolgende Tabelle) beinhalten eine Konfigurations-ID mit einer eindeutigen Versionskennung. Mit der Konfigurations-ID können Sie überprüfen, ob bei einem "Config-Upload" die gelesenen Konfigurationsdaten mit den in der Steuerung gespeicherten Konfigurationsdaten übereinstimmen. Die Konfigurationsdaten sind wie folgt aufgebaut:

Tabelle 3-6 Aufbau der Konfigurationsdaten

Byte	Name	
0	Strukturkennung (2 Byte)	
2	Längenangabe (4 Byte)	
	Länge von Versionskennung und Parameterblock	
6 Versionskennung (4 Byte)		
	Mit Hilfe der Kennung können Sie die Konfiguration eindeutig identifizieren. Es handelt sich dabei um einen Zeitstempel im Linux-Format.	
	Der Zeitstempel gibt an, wieviele Sekunden seit dem 1. Januar 1979, 00:00 Uhr vergangen sind. Die Kennung wird beim Erzeugen einer Konfiguration vergeben.	
10 Ende "DATA"	Parameterblock	

Der "Config_Upload/Config_Download" kann an jedem Kanal des RF680R/RF685R durchgeführt werden. Es werden immer dieselben Konfigurationsdaten übertragen.

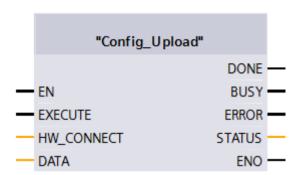


Bild 3-10 Baustein "Config_Upload"

Tabelle 3-7 Erläuterung zum Baustein "Config_Upload"

Parameter	Datentyp	Beschreibung
DATA	Any / Variant	Datenpuffer für Konfigurationsdaten.
		Die reale Länge der Daten hängt von der Komplexität der Projektierung und dem Firmware-Ausgabestand des Readers ab. Bei einer Standardprojektierung des Readers RF680R/RF685R empfehlen wir eine Speichergröße von 4 KB. Verwenden Sie erweitere Reader-Projektierungen (Filterungen) oder möchten Sie in Zukunft die Projektierung ändern ohne die Speichergröße "DATA" anpassen zu müssen, empfehlen wir Ihnen eine Speichergröße von 8-16 KB.
		Hinweis: Bei Variant ist derzeit nur ein "Array_of_Byte" mit variabler Länge anlegbar. Bei Any können zusätzlich auch andere Datentypen/UDTs ange- legt werden.

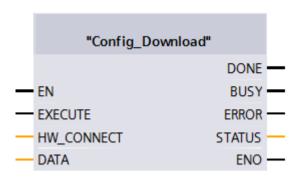


Bild 3-11 Baustein "Config_Download"

Tabelle 3-8 Erläuterung zum Baustein "Config_Download"

Parameter	Datentyp	Beschreibung
DATA	Any / Variant	Datenpuffer für Konfigurationsdaten.
		Die reale Länge der Daten hängt von der Komplexität der Projektierung und dem Firmware-Ausgabestand des Readers ab. Bei einer Standardprojektierung des Readers RF680R/RF685R empfehlen wir eine Speichergröße von 4 KB. Verwenden Sie erweitere Reader-Projektierungen (Filterungen) oder möchten Sie in Zukunft die Projektierung ändern ohne die Speichergröße "DATA" anpassen zu müssen, empfehlen wir Ihnen eine Speichergröße von 8-16 KB.
		Hinweis: Bei Variant ist derzeit nur ein "Array_of_Byte" mit variabler Länge anlegbar. Bei Any können zusätzlich auch andere Datentypen/UDTs ange- legt werden.

3.4.2.2 Inventory

Der Baustein "Inventory" aktiviert die Ausführung von Inventories. Bei den Readern RF620R und RF630R werden immer Inventories durchgeführt, sobald die Antenne eingeschaltet ist.

An dem Ausgangsparameter "NUMBER_TAGS" wird die Anzahl der erkannten Transponder ausgegeben.

Besonderheit der Reader RF680R, RF685R

Beachten Sie, dass die Länge des Datenpuffers ("IDENT_DATA") mindestens der Länge der maximal zu erwartenden Daten entsprechen muss. Werden mehr Transponder erkannt und Daten ausgelesen, als in der zugewiesenen Pufferlänge von "IDENT_DATA" Platz haben, gehen die Daten dieser Transponder verloren. Dieses Verhalten wird durch den Fehler "E7FE0400h" (Pufferüberlauf) angezeigt.

Für die Reader RF680R und RF685R stehen zusätzlich die Parameter "DURATION" und "DUR_UNIT" zur Verfügung. Mit Hilfe der Paramater können Sie die Dauer der Inventories festlegen.

Es gibt bei den Readern RF680R/RF685R vier verschiedene Modi, welche Sie über den Parameter "ATTRIBUTE" auswählen können.

- Beim Start wird eine bestimmte Dauer/Anzahl (Zeitdauer, Anzahl Inventories, Anzahl "Observed"-Events bzw. erkannte Transponder) angegeben. Dabei wird zwischen folgenden drei Optionen unterschieden:
 - Dauer

Inventories über eine festgelegte Zeitdauer durchführen

Anzahl Inventories

Eine festgelegte Anzahl an Inventories durchführen

Anzahl "Observed"-Events

Solange Inventories durchführen, bis eine festgelegte Anzahl Transponder gleichzeitig erkannt wurde.

Für diese Dauer/Anzahl werden dann Inventories vom Reader durchgeführt. Ist die angegebene Zeit/Anzahl erreicht, wird der Baustein beendet und er liefert alle erkannten Transponder in "IDENT_DATA" zurück. D.h. Weitere Befehle können erst dann ausgeführt werden, wenn alle Inventories vollständig durchgeführt wurden. Die Einheit (Zeit oder Anzahl) wird über "DUR_UNIT" und der Wert (Zeitwert oder Anzahl) wird über "DURATION" angegeben. Dieser Modus kann über die Attribute "0x80" und "0x81" ausgeführt werden. Bei dem jeweiligen Attribut werden mehr oder weniger Daten über die erkannten Transponder geliefert.

Mit den Attributen "0x86" ("Presence_Mode" starten) und "0x87" ("Presence_Mode" beenden) k\u00f6nnen Inventories dauerhaft ausgef\u00fchrt werden. Die Anwesenheit eines Transponders kann dann immer \u00fcber "PRESENCE" abgefragt werden, ohne dass der Baustein mit "EXECUTE" gestartet werden muss. Es werden keine Informationen \u00fcber die erkannten Transponder bei Ausf\u00fchrung des Befehls zur\u00fcckgeliefert!

Um Informationen über die erkannten Transponder zu erhalten, führen Sie einen der ersten beiden oben genannten Aufrufe durch (mit Zeit / Anzahl Inventories = 0).

Wenn dieser Modus aktiv ist, werden Transponder betreffende Befehle nicht sofort ausgeführt, sondern erst, wenn ein Transponder erfasst wird. Dadurch werden kürzere

Reaktionszeiten ermöglicht, da der Befehl bereits ansteht wenn der Transponder ins Antennenfeld kommt.

Der "Presence_Mode" ist sinnvoll im Rahmen der "Repeat Command"-Funktion.

An dem Ausgangsparameter "NUMBER_TAGS" wird die Anzahl der erkannten Transponder ausgegeben.

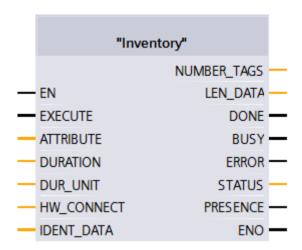


Bild 3-12 Baustein "Inventory"

Tabelle 3-9 Erläuterung zum Baustein "Inventory"

Parameter	Datentyp	Default-Werte	Beschreibung
ATTRIBUTE	Byte	B#16#0	Auswahl des Status-Modus:
			• RF300, MOBY U: 0x00
			 RF620R, RF630R: 0x82 (Folgedatensatz auslesen), 0x83, 0x85, 0x90, 0x91, 0x92
			• RF680R, RF685R: 0x80, 0x81, 0x86, 0x87
DURATION	Word	W#16#0	RF680R, RF685R: Zeitdauer abhängig von "DUR_UNIT"
			Zeitdauer oder Anzahl Inventories oder Anzahl der "Observed"-Events
			Bsp.:
			0x00 ≜ kein Inventory
			0x01 ≙ ein Inventory
DUR_UNIT	Word	W#16#0	RF680R, RF685R: Einheit für "DURATION"
			0x00
			0x01 ≙ Inventories
			0x02 ≙ Anzahl der "Observed"- Events

Parameter	Datentyp	Default-Werte	Beschreibung
IDENT_DATA	Any / Variant	0	Datenpuffer für Inventory-Daten Hinweis: Bei Variant sind ein "Array_of_Byte" mit variabler Länge und die vorhandenen Status-UDTs anlegbar. Bei Any können zusätzlich auch andere Daten-
			typen/UDTs angelegt werden.
NUMBER_TAGS	Int	0	Anzahl der Transponder im Antennenfeld
LEN_DATA	Word	W#16#0	Länge der gültigen Daten

Ergebnisse für MOBY U

Tabelle 3- 10 ATTRIBUTE "0x00" (Datentyp "IID_INVENTORY_00_MOBY_U")

Na	ıme	Тур	Kommentar
nui	mber MDS	WORD	Number of MDS
UI	D length	WORD	length of UID
UI	D	ARRAY[112] of IID IN I 8BYTE	
	UID[1]	IID_IN_I_8BYTE	
	UID	ARRAY[18] of BYT	E
	UID[1]	BYTE	
	UID[2]	BYTE	
	UID[3]	BYTE	
	UID[4]	BYTE	
	UID[5]	BYTE	
	UID[6]	BYTE	
	UID[7]	BYTE	
	UID[8]	BYTE	
	UID[2]	"IID_IN_I_8BYTE"	
	UID[3]	"IID_IN_I_8BYTE"	
	UID[4]	"IID_IN_I_8BYTE"	
	UID[5]	"IID_IN_I_8BYTE"	
	UID[6]	"IID_IN_I_8BYTE"	
	UID[7]	"IID_IN_I_8BYTE"	
	UID[8]	"IID_IN_I_8BYTE"	
	UID[9]	"IID_IN_I_8BYTE"	
	UID[10]	"IID_IN_I_8BYTE"	
	UID[11]	"IID_IN_I_8BYTE"	
	UID[12]	"IID_IN_I_8BYTE"	

Ergebnisse für RF620R, RF630R

Tabelle 3- 11 ATTRIBUTE "0x83" (Datentyp "IID_INVENTORY_82_83_RF600") für RF620R, RF630R mit EPC-ID/UID

Na	me	Тур	Kommentar
res	served0	BYTE	
nun	nber MDS	BYTE	Number of MDS
EPC		ARRAY[119] of "IID IN I 12BYT	E"
E	EPC[1]	"IID_IN_I_12BYT	
	ID	ARRAY[112] of BYTE	
	ID[1]	BYTE	
	ID[2]	BYTE	
	ID[3]	BYTE	
	ID[4]	BYTE	
	ID[5]	BYTE	
	ID[6]	BYTE	
	ID[7]	BYTE	
	ID[8]	BYTE	
	ID[9]	BYTE	
	ID[10]	BYT	
	ID[11]	BYTE	
	ID[12]	BYTE	
E	EPC[2]	"IID_IN_I_12BYT	Е"
E	EPC[3]	"IID_IN_I_12BYT	Е"
F	EPC [4]	"IID_IN_I_12BYT	Е"
F	EPC[5]	"IID_IN_I_12BYT	Е"
E	EPC[6]	"IID_IN_I_12BYT	Е"
F	EPC[7]	"IID_IN_I_12BYT	Е"
F	EPC[8]	"IID_IN_I_12BYT	Е"
E	EPC[9]	"IID_IN_I_12BYT	Е"
E	EPC[10]	"IID_IN_I_12BYT	Е"
F	EPC[11]	"IID_IN_I_12BYT	Е"
E	EPC [12]	"IID_IN_I_12BYT	Е"
E	EPC[13]	"IID_IN_I_12BYT	Е"
E	EPC [14]	"IID_IN_I_12BYT	Е"
E	EPC [15]	"IID_IN_I_12BYT	Е"
E	EPC[16]	"IID_IN_I_12BYT	Е"
E	EPC [17]	"IID_IN_I_12BYT	Е"
E	EPC[18]	"IID_IN_I_12BYT	Е"
E	EPC[19]	"IID_IN_I_12BYT	Е"
		1	· · ·

Hinweis

Anzahl der EPC-IDs

"number_MDS" gibt die Anzahl der mit dem "INVENTORY"-Baustein übertragenen EPC-IDs (1 bis 19) an. Um die Handle-IDs von allen im Antennenfeld befindlichen Transpondern zu erhalten, müssen Sie ggf. den "INVENTORY"-Baustein mit ATTRIBUTE "0x82" wiederholt aufsetzen.

Tabelle 3- 12 ATTRIBUTE "0x83", "0x90", "0x91" und "0x92" (Datentyp "IID_INVENTORY_8x_9x_RF6_MD") für RF620R, RF630R mit Handle-ID

N	lame	Тур	Kommentar
r	eserved	BYTE	
	umber MDS	BYTE	Number of MDS
U	ID	ARRAY[129] of "IID IN I 8BYTE"	
	UID[1]	"IID_IN_I_8BYTE"	
	UID	ARRAY[18] of BYT	E
	UID[1]	BYTE	
	UID[2]	BYTE	
	UID[3]	BYTE	
	UID[4]	BYTE	
	UID[5]	BYTE	
	UID[6]	BYTE	
	UID[7]	BYTE	
	UID[8]	BYTE	
	UID[2]	"IID_IN_I_8BYTE"	
	UID[3]	"IID_IN_I_8BYTE"	
	UID[4]	"IID_IN_I_8BYTE"	
	UID[5]	"IID_IN_I_8BYTE"	
	UID[6]	"IID_IN_I_8BYTE"	
	UID[7]	"IID_IN_I_8BYTE"	
	UID[8]	"IID_IN_I_8BYTE"	
	UID[9]	"IID_IN_I_8BYTE"	
	UID[10]	"IID_IN_I_8BYTE"	
	UID[11]	"IID_IN_I_8BYTE"	
	UID[12]	"IID_IN_I_8BYTE"	
	UID[13]	"IID_IN_I_8BYTE"	
	UID[14]	"IID_IN_I_8BYTE"	
	UID[15]	"IID_IN_I_8BYTE"	
	UID[16]	"IID_IN_I_8BYTE"	
	UID[17]	"IID_IN_I_8BYTE"	
	UID[18]	"IID_IN_I_8BYTE"	

Name	Тур	Kommentar
UID[19]	"IID_IN_I_8BYTE"	
UID[20]	"IID_IN_I_8BYTE"	
UID[21]	"IID_IN_I_8BYTE"	
UID[22]	"IID_IN_I_8BYTE"	
UID[23]	"IID_IN_I_8BYTE"	
UID[24]	"IID_IN_I_8BYTE"	
UID[25]	"IID_IN_I_8BYTE"	
UID[26]	"IID_IN_I_8BYTE"	
UID[27]	"IID_IN_I_8BYTE"	
UID[28]	"IID_IN_I_8BYTE"	
UID[29]	"IID_IN_I_8BYTE"	
reserved1	DWORD	
Data	ARRAY[1222] of BYTE	

Hinweis

Anzahl der Handle-IDs

"number_MDS" gibt die Anzahl der mit dem "INVENTORY"-Baustein übertragenen Handle-IDs (1 bis 29) an. Um die Handle-IDs von allen im Antennenfeld befindlichen Transpondern zu erhalten, müssen Sie ggf. den "INVENTORY"-Baustein mit ATTRIBUTE "0x82" wiederholt aufsetzen.

Tabelle 3- 13 ATTRIBUTE "0x85" (Datentyp "IID_INVENTORY_85_RF600")

Name		1	Тур	Kommentar
reserved		ved	BYTE	
number MDS		r MDS	STRUCT	Number of MDS
ID			BYTE	
	ID[1]		BYTE	
	H	Mandle	ARRAY[18] of BYTE	
		Handle[1]	BYTE	
		Handle[2]	BYTE	
		Handle[3]	BYTE	
		Handle[4]	BYTE	
		Handle[5]	BYTE	
		Handle[6]	BYTE	
		Handle[7]	BYTE	
		Handle[8]	BYTE	
	F	IPC	ARRAY[112] of BYTE	
		EPC[1]	BYTE	
		EPC[2]	BYTE	

Name		Тур	Kommentar
	EPC[3]	BYTE	
	EPC[4]	BYTE	
	EPC[5]	BYTE	
	EPC[6]	BYTE	
	EPC[7]	BYTE	
	EPC[8]	BYTE	
	EPC[9]	BYTE	
	EPC[10]	BYTE	
	EPC[11]	BYTE	
	EPC[12]	BYTE	
II	D[2]	"IID_IN_I_20Byte"	
II	D[3]	"IID_IN_I_20Byte"	
II	D[4]	"IID_IN_I_20Byte"	
II	D[5]	"IID_IN_I_20Byte"	
II	D[6]	"IID_IN_I_20Byte"	
II	D[7]	"IID_IN_I_20Byte"	
II	D[8]	"IID_IN_I_20Byte"	
II	D[9]	"IID_IN_I_20Byte"	
II	D[10]	"IID_IN_I_20Byte"	
II	D[11]	"IID_IN_I_20Byte"	

Hinweis

Anzahl der übertragenen IDs

"number_MDS" gibt die Anzahl der mit dem "INVENTORY"-Baustein übertragenen IDs (1 bis 11; Handle-IDs und EPC-IDs) an. Um die IDs von allen im Antennenfeld befindlichen Transpondern zu erhalten, müssen Sie ggf. den "INVENTORY"-Baustein mit ATTRIBUTE "0x82" wiederholt aufsetzen.

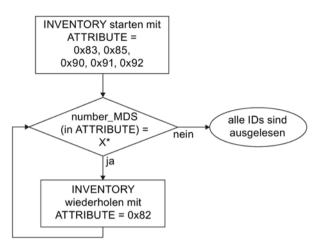
Weitere Details zu den einzelnen Status-Modi entnehmen Sie bitte den zu den Modi passenden Handbüchern "FB 45", "FB55" und "SIMATIC RF620R/RF630R".

Die hier angegeben Kennungen der Status-Modi entsprechen folgenden Kennungen in den anderen Handbüchern:

0x82	≙	0x02
0x83		0x03
0x85	≙	0x05
0x90		0x10
0x91		0x11
0x92	≙	0x12

Programmierung des ATTRIBUTE "0x82"

Wenn die Anzahl der im Antennenfeld befindlichen Transponder unbekannt ist, wiederholen Sie den "INVENTORY"-Baustein mit Hilfe des ATTRIBUTE = "0x82".



^{*} Die Anzahl der zurückgelieferten IDs "X" ist abhängig vom verwendeten "ATTRIBUTE".

Bild 3-13 Programmablauf des ATTRIBUTE "0x82" bei unbekannten Transponder-Populationen

Ergebnisse für RF680R, RF685R

Die Anzahl der Elemente "TAG_DATA[x]" der Datentypen der ATTRIBUTE "0x80" und "0x81" ist abhängig von der Anzahl der zu erwartenden Transponder. Aus diesem Grund müssen Sie sich den Empfangspuffer selbst zusammenstellen. Beachten Sie bei der Erstellung des Empfangspuffer "IDENT_DATA"/Datentyps folgenden Aufbau:

- Das erste Element "NUM MDS" ist immer vom Typ "WORD".
- Das darauf folgende Element "TAG_DATA" ist immer vom Typ "ARRAY". In dem "ARRAY" müssen Sie die Anzahl der zu erwartenden Transponder eintragen ("n").

Die nachfolgenden Tabellen zeigen beispielhaft den Aufbau des Empfangspuffer "IDENT_DATA"/Datentyp für die ATTRIBUTE "0x80" und "0x81".

Tabelle 3- 14 ATTRIBUTE "0x80"

N	lame	Тур	Kommentar
N	IUM MDS	WORD	Number of MDS
Т	'AG_DATA	ARRAY[1n]of IID IN I 80	Length of EPC ID
	TAG_DATA[1]	IID_IN_I_80	
	Reserved	BYTE	
	ID_Len	BYTE	Length of EPC ID
	EPC_ID	ARRAY[162] of BYTE	EPC-ID
	tagPC	WORD	
	TAG_DATA[2]	IID_IN_I_80	
	• • •		
	TAG_DATA[n]	IID_IN_I_80	

Tabelle 3- 15 ATTRIBUTE "0x81"

Na	ame	Тур	Kommentar	
NU	M MDS	WORD	Number of MDS	
TA	G_DATA	ARRAY[1n] of IID IN I 81		
	TAG_DATA[1]	IID_IN_1_81		
	reserved	BYTE		
	ID_LEN	BYTE	EPC length	
	EPC_ID	ARRAY[162]of BYTE	EPC-ID	
	tagPC	WORD		
	RSSI	BYTE	RSSI value	
	MaxRSSI	BYTE	highest RSSI value	
	MinRSSI	BYTE	lowest RSSI value	
	channel	BYTE	channel; 115_ESTI; 153:FCC	
	antenna	BYTE	antenna; bit coded; Bit 0=antenna 1; Bit 1=antenna 2	
	polarization	BYTE	polarizatuin of antenna; 0=undefined; 1=circular	
	time	Time_OF_Day	S7 time	
	power	BYTE	power in dBm	
	filterDataAvailable	BYTE	0=false; 1=true	
	Inventoried	WORD	1)	
	TAG_DATA[2] IID_IN_1_81			
	•••			
П	TAG_DATA[n] IID_IN_1_81			

¹⁾ Angabe, wie oft der Transponder über die Luftschnittstelle erkannt wurde, bis er in den Zustand "Observed" wechselt.

3.4.2.3 Read_EPC_Mem

Der Baustein "Read_EPC_Mem" liest Daten aus dem EPC-Speicher des RF600-Transponders aus. Über den Parameter "LEN_DATA" wird die auszulesende Länge des EPC-Speichers angegeben.

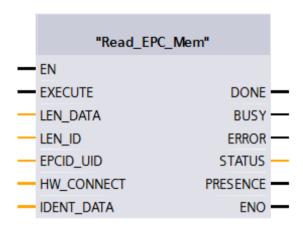


Bild 3-14 Baustein "Read_EPC_Mem"

Tabelle 3- 16 Erläuterung zum Baustein "Read_EPC_Mem"

Parameter	Datentyp	Beschreibung
LEN_DATA	Word	Länge des auszulesenden EPC-Speichers (1 62 Byte)
LEN_ID	Byte	Länge der EPC-ID/UID
		Default-Wert: 0x00 ≜ unspezifizierter Singletag- Zugriff (RF680R, RF685R)
EPCID_UID	Array[162] of Byte	Puffer für bis zu 62 Byte EPC-ID, 8 Byte UID oder 4 Byte Handle-ID.
		 2-62-Byte EPC-ID wird an Anfang des Puffer eingetragen (Länge wird durch "LEN_ID" be- schrieben)
		 8-Byte UID wird an Anfang des Puffers ein- getragen ("LEN_ID = 8")
		4-Byte Handle-ID muss im Array-Element [5]-[8] eingetragen werden ("LEN_ID = 8") Default-Wert: 0x00 ≜ unspezifizierter Singletag- Zugriff (RF620R, RF630R, RF640R)
IDENT_DATA	Any / Variant	Datenpuffer in den die gelesenen EPC- Speicherdaten abgelegt werden.
		Hinweis: Bei Variant ist derzeit nur ein "Array_of_Byte" mit variabler Länge anlegbar. Bei Any können zusätzlich auch andere Datentypen/UDTs ange- legt werden.

3.4.2.4 Read_TID

Der Baustein "Read_TID" liest Daten aus dem TID-Speicherbereich (Tag Identification Memory Bank) des RF600-Transponders aus. Über den Parameter "LEN_DATA" wird die zu lesende Länge der TID angegeben. Die Länge der TID variiert abhängig vom Transponder und ist dem dazugehörigen Datenblatt zu entnehmen.

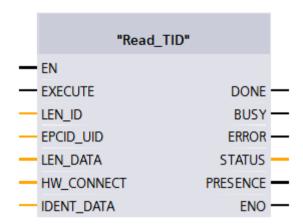


Bild 3-15 Baustein "Read_TID"

Tabelle 3- 17 Erläuterung zum Baustein "Read_TID"

Parameter	Datentyp	Default-Werte	Beschreibung
LEN_ID	Byte	B#16#0	Länge der EPC-ID/UID
EPCID_UID	Array[162] of Byte	0	Länge der EPC-ID/UID
	Буце		Default-Wert: 0x00 ≙ unspezifizierter Singletag-Zugriff (RF680R, RF685R)
LEN_DATA	Word	W#16#4	Puffer für bis zu 62 Byte EPC-ID, 8 Byte UID oder 4 Byte Handle-ID.
			2-62-Byte EPC-ID wird an Anfang des Puffer eingetragen (Länge wird durch "LEN_ID" beschrieben)
			8-Byte UID wird an Anfang des Puffers eingetragen ("LEN_ID = 8")
			4-Byte Handle-ID muss im Array- Element [5]-[8] eingetragen wer- den ("LEN_ID = 8")
			Default-Wert: 0x00 ≙ unspezifizierter Singletag-Zugriff (RF620R, RF630R, RF640R)
IDENT_DATA	Any / Variant	0	Gelesene TID
			Hinweis: Bei Variant ist derzeit nur ein "Ar- ray_of_Byte" mit variabler Länge an- legbar. Bei Any können zusätzlich auch andere Datentypen/UDTs ange- legt werden.

3.4.2.5 Read_UID

Der Baustein "Read_UID" liest die UID eines HF-Transponders aus. Die UID hat immer eine feste Länge von 8 Byte.

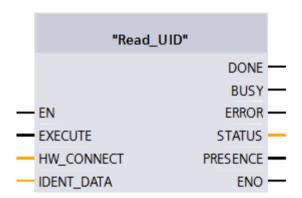


Bild 3-16 Baustein "Read_UID"

Tabelle 3- 18 Erläuterung zum Baustein "Read_UID"

Parameter	Datentyp	Beschreibung
IDENT_DATA	Any / Variant	UID
		Hinweis: Bei Variant ist derzeit nur ein "Array_of_Byte" mit variabler Länge anlegbar. Bei Any können zusätzlich auch andere Datentypen/UDTs ange- legt werden.

3.4.2.6 Set Ant

Mit Hilfe des Bausteins "Set_Ant" können Sie Antennen aus- oder einschalten. Es gibt unterschiedliche Bausteine für RF300 und RF600. Der Baustein "Set_Ant_RF300" kann auch für RF200, MOBY D und MOBY U verwendet werden. Der Baustein "Set_Ant_RF600" bezieht sich ausschließlich auf die Reader RF620R und RF630R.

Set_Ant_RF300

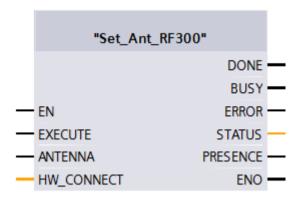


Bild 3-17 Baustein "Set_Ant_RF300"

Tabelle 3- 19 Erläuterung zum Baustein "Set_Ant_RF300"

Parameter	Datentyp	Beschreibung	
ANTENNA	Bool	0 = Antenne ausschalten	
		1 = Antenne einschalten	

Set_Ant_RF600

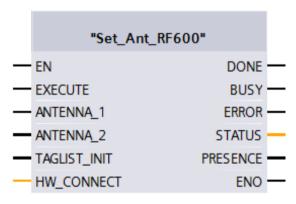


Bild 3-18 Baustein "Set_Ant_RF600"

Tabelle 3- 20 Erläuterung zum Baustein "Set_Ant_RF600"

Parameter	Datentyp	Beschreibung
ANTENNA_1	Bool	0 = Antenne 1 ausschalten 1 = Antenne 1 einschalten
ANTENNA_2	Bool	0 = Antenne 2 ausschalten 1 = Antenne 2 einschalten
TAGLIST_INIT	Bool	0 = TagList wird zurückgesetzt 1 = Mit vorhandener TagList wird weitergearbeitet

3.4.2.7 Set_Param

Mit Hilfe des Bausteins "Set_Param" können Sie UHF-Parameter an einem RF680R/RF685R während der Laufzeit ändern (z. B. die Antennenleistung).

Hinweis

Einstellungen nur flüchtig gespeichert

Beachten Sie, dass die im Baustein "Set_Param" hinterlegten Parameter nur flüchtig gespeichert werden. Wird die Spannung des Readers unterbrochen, gehen die hinterlegten Werte verloren und müssen erneut gesetzt werden.

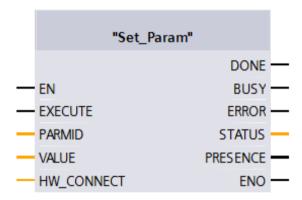


Bild 3-19 Baustein "Set_Param"

Tabelle 3- 21 Erläuterung zum Baustein "Set_Param"

Parameter	Datentyp	Default-Werte	Beschreibung
PARMID	DWORD	0x00	Parameterkennung
VALUE	DWORD	0x00	Parameterwert

Tabelle 3-22 Parameterwerte

PARMID (hex)	PARMID (ASCII)	Parameter	VALUE
0x41315057	A1PW	Strahlungsleistung Antenne 01	Wertebereich 0, 5 33
0x41325057	A2PW	Strahlungsleistung Antenne 02	Schrittweite 0,25
0x41335057	A3PW	Strahlungsleistung Antenne 03	Strahlungsleistung der Antenne in
0x41345057	A4PW	Strahlungsleistung Antenne 04	dBm. Byte 1 und 2 sind unbelegt, Byte 3 stellt die Ganzzahl dar und Byte 4 die Nachkommastelle. Beispiel: Eine Strahlungsleistung
			von 10,25 dBm entspricht einem "VALUE" von "0x0A19".
0x41315452	A1TR	RSSI-Schwellwert Antenne 01	Wertebereich 0 255
0x41325452	A2TR	RSSI-Schwellwert Antenne 02	Schwellwert für RSSI. Transponder
0x41335452	A3TR	RSSI-Schwellwert Antenne 03	mit niedrigeren Werten werden verworfen. Einheitsloser Wert, ohne
0x41345452	A4TR	RSSI-Schwellwert Antenne 04	direkten Bezug zur Strahlungsleistung.
0x5331444C	S1DL	RSSI Delta Lesestelle 1	Wertebereich 0 255
0x5332444C	S2DL	RSSI Delta Lesestelle 2	Differenz für RSSI-Werte. Trans-
0x5333444C	S3DL	RSSI Delta Lesestelle 3	ponder mit niedrigeren Werten bezogen auf den Transponder mit
0x5334444C	S4DL	RSSI Delta Lesestelle 4	dem höchsten RSSI-Wert werden verworfen. Einheitsloser Wert, ohne direkten Bezug zur Strahlungsleistung.
0x4131504F	A1PO	Polarisation Antenne 01	Wertebereich 0, 1, 2, 4
0x4132504F	A2PO	Polarisation Antenne 02	Polarisation der Antenne (für intelli-
0x4133504F	A3PO	Polarisation Antenne 03	gente Antennen, z. B. interne Antenne RF685R)
0x4134504F	A4PO	Polarisation Antenne 04	0: default, undefiniert
			1: zirkular
			2: vertikal linear
			4: horizontal linear
			Eingabe ist bitkodiert. Kombinationen sind möglich (Werte addieren).

3.4.2.8 Write_EPC_ID

Der Baustein "Write_EPC_ID" überschreibt die EPC-ID des RF600-Transponders und passt die Länge der EPC-ID im Speicher des Transponders an. Über den Parameter "LEN_ID_NEW" wird die zu schreibende neue EPC-ID-Länge und über die Parameter "LEN_ID" und "EPCID_UID" wird die bisherige EPC-ID angegeben.

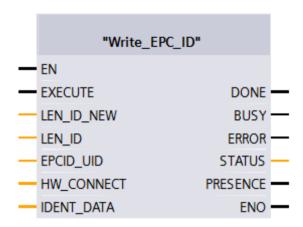


Bild 3-20 Baustein "Write_EPC_ID"

Tabelle 3- 23 Erläuterung zum Baustein "Write_EPC_ID"

Parameter	Datentyp	Default-Werte	Beschreibung
LEN_ID_NEW	Byte	W#16#0C	Länge der aktuellen EPC-ID
LEN_ID	Byte	B#16#0	Länge der bisherigen EPC-ID
EPCID_UID	Array[162] of Byte	0	Bisherige EPC-ID
IDENT_DATA	Any / Variant	0	Aktuelle EPC-ID Hinweis: Bei Variant ist derzeit nur ein "Ar- ray_of_Byte" mit variabler Länge an- legbar. Bei Any können zusätzlich auch andere Datentypen/UDTs ange- legt werden.

3.4.2.9 Write_EPC_Mem

Der Baustein "Write_EPC_Mem" überschreibt den EPC-Speicher des RF600-Transponders. Über den Parameter "LEN_DATA" wird die zu überschreibende Länge des EPC-Speichers angegeben.

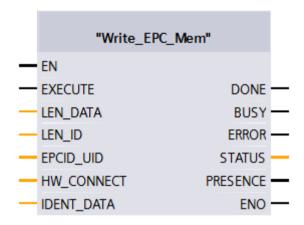


Bild 3-21 Baustein "Write_EPC_Mem"

Tabelle 3- 24 Erläuterung zum Baustein "Write_EPC_Mem"

Parameter	Datentyp	Beschreibung
LEN_DATA	Word	Länge des zu überschreibenden EPC-Speichers (1 62 Byte)
LEN_ID	Byte	Länge der EPC-ID/UID
		Default-Wert: 0x00 ≙ unspezifizierter Singletag- Zugriff (RF680R, RF685R)
EPCID_UID	Array[162] of Byte	Puffer für bis zu 62 Byte EPC-ID, 8 Byte UID oder 4 Byte Handle-ID.
		2-62-Byte EPC-ID wird an Anfang des Puffer eingetragen (Länge wird durch "LEN_ID" be- schrieben)
		8-Byte UID wird an Anfang des Puffers ein- getragen ("LEN_ID = 8")
		4-Byte Handle-ID muss im Array-Element [5]-[8] eingetragen werden ("LEN_ID = 8") Default-Wert: 0x00 ≜ unspezifizierter Singletag- Zugriff (RF620R, RF630R, RF640R)
IDENT_DATA	Any / Variant	Datenpuffer mit den zu überschreibenden EPC- Speicherdaten.
		Hinweis: Bei Variant ist derzeit nur ein "Array_of_Byte" mit variabler Länge anlegbar. Bei Any können zusätzlich auch andere Datentypen/UDTs ange- legt werden.

3.4.2.10 AdvancedCMD

Mit dem Baustein "AdvancedCmd" kann jeder Befehl ausgeführt werden, auch Befehle die durch die anderen Bausteine nicht dargestellt werden. Dieser allgemeine Aufbau kann für alle Befehle verwendet werden und ist ausschließlich für geschulte Anwender konzipiert.

Dieser Baustein ermöglicht es Ihnen verkettete Befehle zu versenden. Dafür stellt der Baustein einen CMD-Puffer für 10 Befehle zur Verfügung. Alle verketteten Befehle müssen ab dem ersten Platz im Puffer eingetragen werden. Zusätzlich muss für jeden verketteten Befehl das "Chained-Bit" in der CMD-Struktur gesetzt werden. Ausschließlich bei dem letzten Befehl in der Kette wird das "Chained-Bit" nicht gesetzt. Weitere Informationen zum "Chained-Bit" finden Sie im Kapitel "Verkettung (Seite 99)".

Im Eingangsparameter "CMD" muss die komplette Befehlsstruktur angegeben werden. Die Struktur für den Parameter "CMD" müssen Sie in einem Datenbaustein anlegen.

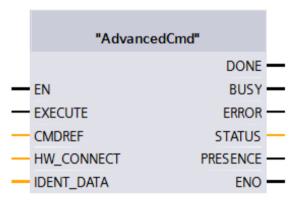


Bild 3-22 Baustein "AdvancedCmd"

Tabelle 3- 25 Erläuterung zum Baustein "AdvancedCMD"

Parameter	Datentyp	Default-Werte	Beschreibung
CMDREF	IID_CMD_STRUCT		Eine ausführliche Beschreibung des Parameters finden Sie in den Kapiteln:
			"Befehle des Ident-Profils (Seite 74)"
			"Befehlsstruktur (Seite 76)"
IDENT_DATA	Any / Variant	0	Puffer für zu schreibende bzw. gelesene Daten.
			Hinweis: Bei Variant ist derzeit nur ein "Ar- ray_of_Byte" mit variabler Länge an- legbar. Bei Any können zusätzlich auch andere Datentypen/UDTs ange- legt werden.

3.4.3 Reset-Bausteine

Die in diesem Kapitel beschriebenen Reset-Bausteine benötigen Sie, wenn Sie die Code-Lesesysteme MV420, MV440 oder die Kommunikationsmodule RF180C, ASM 456 an einer Steuerung SIMATIC S7-1200/-1500 betreiben möchten. Alternativ können Sie diese Bausteine auch für die RF120C verwenden, wenn Sie in der Gerätekonfiguration die entsprechende Einstellung parametriert haben.

Diese Reset-Bausteine haben im System die gleiche Funktion wie der weiter vorne beschriebene Baustein "Reset_Reader". Allerdings müssen Sie bei den hier beschriebenen Bausteinen Reader-abhängige Parameter einstellen.

Beachten Sie, dass bei den Parametern automatisch der angegebene Default-Wert verwendet wird, wenn Sie keinen Wert manuell auswählen.

3.4.3.1 Reset_MOBY_D

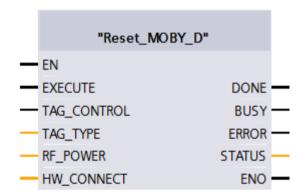


Bild 3-23 Baustein "Reset_MOBY_D"

Tabelle 3- 26 Erläuterung zum Baustein "Reset_MOBY_D"

Parameter	Datentyp	Default-Werte	Beschreibung
TAG_CONTROL	Bool	True	Anwesenheitskontrolle
TAG_TYPE	Byte	1	Transponder-Typ:
			1 = jeder ISO-Transponder
RF_POWER	Byte	0	Ausgangsleistung
			HF-Leistung von 0,5 W bis 10 W in 0,25 W-Schritten (Wertebereich: 0x02 0x28)

3.4.3.2 Reset_MOBY_U

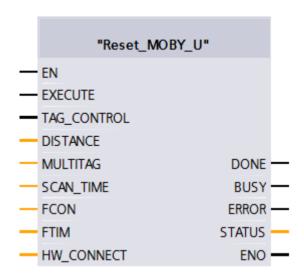


Bild 3-24 Baustein "Reset_MOBY_U"

Tabelle 3- 27 Erläuterung zum Baustein "Reset_MOBY_U"

Parameter	Datentyp	Default-Werte	Beschreibung
TAG_CONTROL	Bool	True	Anwesenheitskontrolle
DISTANCE	Byte	23h	Reichweitenbegrenzung (Wertebereich: 0x02 0x23 oder 0x82 0xA3 für verminderte Sende- leistung)
MULTITAG	Byte	1	Maximale Anzahl der parallel im Antennenfeld bearbeitbaren Transponder. (Wertebereich: 0x01 0x12)
SCAN_TIME	Byte	0	Scanning_Time: Stand-by-Zeit des Transponders (Wertebereich: 0x00 0xC8)
FCON	Byte	0	field_ON_control: Bero-Betriebsart (Wertebereich: 0x00 0x03)
FTIM	Byte	0	field_ON_time: Zeit für Bero- Betriebsart (Wertebereich: 0x00 0xFF)

3.4.3.3 Reset_MV

Um Kameras der Code-Lesesysteme zurückzusetzen, rufen Sie den Baustein auf und führen Sie den Parameter "EXECUTE" aus.

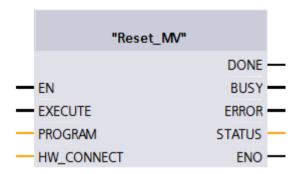


Bild 3-25 Baustein "Reset_MV"

Tabelle 3- 28 Erläuterung zum Baustein "Reset_MV"

Parameter	Datentyp	Beschreibung
PROGRAM	Byte	Programmanwahl
		B#16#0: Reset ohne Programmanwahl bzw. im Falle einer Diagnose wird der Fehlercode für "IN_OP = 0" am Ausgangsparameter "STATUS" angezeigt.
		B#16#1 B#16#15: Nummer des zu startenden Programms ⇒ Reset mit Programmanwahl (ab FW V5.1 der MV4x0)

3.4.3.4 Reset_RF200

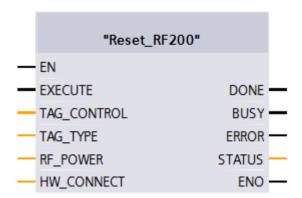


Bild 3-26 Baustein "Reset_RF200"

Tabelle 3- 29 Erläuterung zum Baustein "Reset_RF200"

Parameter	Datentyp	Default-Werte	Beschreibung	
TAG_CONTROL	Byte	1	Anwesenheitskontrolle	
TAG_TYPE	Byte	1	Transponder-Typ:	
			1 = jeder ISO-Transponder	
			• 3 = MDS D3xx - Optimierung	
RF_POWER	Byte	4	Ausgangsleistung; nur für RF290R relevant	
			HF-Leistung von 0,5 W bis 5 W in 0,25 W-Schritten (Wertebereich: 0x02 0x14). Default-Wert 0x04 ≙ 1 W.	

3.4.3.5 Reset_RF300

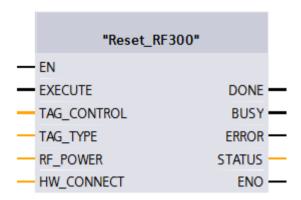


Bild 3-27 Baustein "Reset_RF300"

Tabelle 3- 30 Erläuterung zum Baustein "Reset_RF300"

Parameter	Datentyp	Default-Werte	Beschreibung
TAG_CONTROL	Byte	1	Anwesenheitskontrolle
			• 0 = Aus
			• 1 = Ein
			4 = Anwesenheit (Antenne ist aus. Antenne wird nur eingeschaltet, wenn ein Read- oder Write-Befehl geschickt wird.)
TAG_TYPE	Byte	0	Transponder-Typ:
			1 = jeder ISO-Transponder
			0 = RF300-Transponder
RF_POWER	Byte	0	Ausgangsleistung; nur für RF380R relevant
			HF-Leistung von 0,5 W bis 2 W in 0,25 W-Schritten (Wertebereich: 0x02 0x08). Default-Wert 0x00 ≙ 1,25 W.

3.4.3.6 Reset_RF600

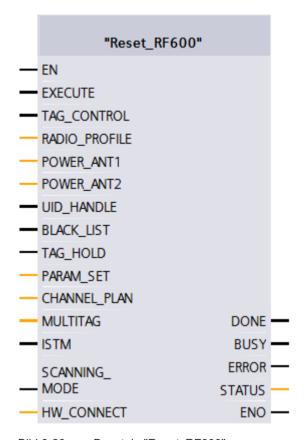


Bild 3-28 Baustein "Reset_RF600"

Tabelle 3- 31 Erläuterung zum Baustein "Reset_RF600"

Parameter	Datentyp	Default-Werte	Beschreibung
TAG_CONTROL	Bool	True	Anwesenheitskontrolle
RADIO_PROFILE	Byte	1	Scanning_Time: Funkprofil nach EPC-Global (Wertebereich: 0x01 0x09, abhängig von der Reader-Variante)
POWER_ANT1	Byte	0	Sendeleistung für Antenne 1 bzw. interne Antenne (Wertebereich: 0x00 0x0F)
POWER_ANT2	Byte	0	Sendeleistung für Antenne 2 bzw. externe Antenne (Wertebereich: 0x00 0x0F)
UID_HANDLE	Bool	False	Bedeutung der UID im Befehl:
			True = Handle-ID, nur die niederwertigsten 4 Bytes der UID werden ausgewertet; False = UID-/EPC-ID mit 8 Byte Länge
BLACK_LIST	Bool	False	True = Black List aktivieren

Parameter	Datentyp	Default-Werte	Beschreibung
TAG_HOLD	Bool	False	True = Tag Hold aktivieren
PARAM_SET	Byte	0	Field_ON_Control (0 = fast; Wertebereich: 0x00, 0x02)
CHANNEL_PLAN	Byte	OF	Field_ON_Time (Wertebereich: 0x00 0x0F; nur bei ETSI)
MULTITAG	Byte	1	Maximale Anzahl der parallel im Antennenfeld bearbeitbaren Transponder. (Wertebereich: 0x01 0x50)
ISTM	Bool	False	True = Intelligenten Singletag-Modus aktivieren
SCANNING_MOD E	Bool	False	True = Scanning Mode aktivieren 1)

¹⁾ Ist aktuell mit den Ident-Bausteinen nicht möglich.

3.4.3.7 Reset_Univ

Der Baustein "Reset_Univ" ist ein universeller Reset-Baustein, mit dem Identifikationssysteme zurückgesetzt werden können. Verwenden Sie diesen Baustein nur in Absprache mit dem Support.

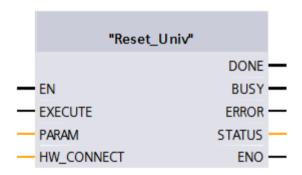


Bild 3-29 Baustein "Reset_Univ"

Tabelle 3- 32 Erläuterung zum Baustein "Reset_Univ"

Parameter	Datentyp	Beschreibung
PARAM	Array [116] of Byte	Daten für Reset-Telegramm
		Die hier einzustellenden Daten können für Spezialeinstellungen bei Bedarf vom Support bereitgestellt werden.

Tabelle 3-33 Aufbau des Parameters "PARAM"

Byte	1	25	6	78	9	10	11	12	1314	15	16
Wert	04h	0	0Ah	0	scan- ning_ time	param	option_ 1	dis- tance_ limiting	Anzahl der Trans- ponder	field_ on_ control	field_ on_ time

3.4.4 Status-Bausteine

3.4.4.1 Reader_Status

Der Baustein "Reader_Status" liest Statusinformationen aus dem Reader aus. Für die verschiedenen Reader-Familien gibt es unterschiedliche Status-Modi, die Sie über den Parameter "ATTRIBUTE" auswählen können.

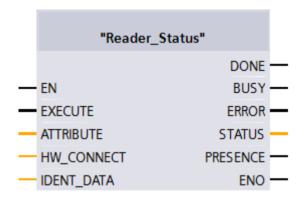


Bild 3-30 Baustein "Reader_Status"

Tabelle 3- 34 Erläuterung zum Baustein "Reader_Status"

Parameter	Datentyp	Default-Werte	Beschreibung
ATTRIBUTE	Byte	B#16#81	Kennung der Status-Modi / Mögliche Eingaben:
			• RF200: 0x81
			• RF300: 0x81, 0x86
			• RF620R, RF630R: 0x87, 0x88, 0xA0, 0xA1
			• RF680R, RF685R: 0x89
			• MOBY U: 0x81, 0x84, 0x85
			• MOBY D: 0x81
IDENT_DATA	Any / Variant	0	Ergebniswerte je nach Attribute
			Hinweis: Bei Variant sind ein "Array_of_Byte" mit variabler Länge und die vorhandenen Status-UDTs anlegbar. Bei Any können zusätzlich auch andere Datentypen/UDTs angelegt werden.

Ergebnisse

Legen Sie den jeweils richtigen Datentyp, der dem ATTRIBUTE-Wert zugeordnet ist, am Eingang "IDENT_DATA" des Bausteins an, damit die Daten richtig interpretiert werden können.

Tabelle 3- 35 ATTRIBUTE "0x81" (Datentyp "IID_READER_STATUS_81_RF200_300_U")

Name	Тур	Kommentar
status info	BYTE	SLG status mode
hardware	CHAR	Type of hardware
hardware version	WORD	Version of hardware
loader version	WORD	Version of loader
firmware	CHAR	Type of firmware
firmware version HB	BYTE	Version of firmware
firmware_version_LB	BYTE	
driver	CHAR	Type of driver
driver version	WORD	Version of driver
interface	BYTE	Type of interface (RS 232/RS 422)
baud	BYTE	Baudrate
reserved1	BYTE	Reserved
reserved2	BYTE	Reserved
reserved3	BYTE	Reserved
distance limiting SLG	BYTE	Distance limiting of SLG
multitag SLG	BYTE	Multitag SLG
field ON control SLG	BYTE	Field ON control
field ON time SLG	BYTE	Field On time
sync SLG	BYTE	Synchronization with SLG
status ant	BYTE	Status of antenne
stand by	BYTE	Time of standby after command
MDS control	BYTE	Presence mode

Tabelle 3- 36 ATTRIBUTE "0x84" (Datenytp "IID_READER_STATUS_84_MOBY_U")

Name	Тур	Kommentar
status info	BYTE	SLG status mode
number MDS	BYTE	Range 124
UID	ARRAY [124] of DWord	

Tabelle 3- 37 ATTRIBUTE "0x86" (Datentyp "IID_READER_STATUS_86_RF300")

Name	Тур	Kommentar	
status info	BYTE	SLG status mode	
FZP	BYTE	Error counter passive: distortion without communication	
ABZ	BYTE	Dropout counter	
CFZ	BYTE	Code error counter	
SFZ	BYTE	Signature error counter	
CRCFZ	BYTE	CRC-error counter	
BSTAT	BYTE	Status of last command	
ASMFZ	BYTE	Error counter for host interface (ASM)	
reserved0	ARRAY [120]		

Tabelle 3- 38 ATTRIBUTE "0x87" (Datentyp "IID_READER_STATUS_RF600")

Name	Тур	Kommentar	
status info	BYTE	SLG status mode	
hardware	CHAR	Type of hardware	
hardware version	WORD	Version of hardware	
reserved0	WORD	verbion of naraware	
firmware	CHAR	Type of firmware	
firmware version HB	BYTE	Version of firmware highbyte	
firmware version LB	BYTE	Version of firmware lowbyte	
driver	CHAR	Type of driver	
current_time_hour	BYTE	Hours 1)	
current time min	BYTE	Minutes	
current time sec	BYTE	Seconds	
reserved1	BYTE		
SLG version	BYTE	SLG version	
baud	BYTE	Baudrate	
reserved2	BYTE		
distance limiting SLG	BYTE	Selected transmit power	
multitag SLG	BYTE	Multitag SLG	
field ON control SLG	BYTE	Selected comunication typ	
field ON time SLG	BYTE	Selected channel	
expert mode	BYTE	Expert mode	
status_ant	BYTE	Status of antenna ²⁾	
scanning_time_SLG	BYTE	Radio communication profile (country specific radio standart)	
MDS control	BYTE	Presence mode	

¹⁾ Es wird der readerinterne Zeitstempel ausgegeben, welcher sich auf dieses Ereignis bezieht. Der readerinterne Zeitstempel wird nicht mit UTC synchronisiert.

²⁾ Der Antennen-Status bezieht sich auf den "ATTRIBUTE" (Bit 0 und 1) des zuletzt ausgeführten "SET-ANT" oder auf den durch "init_run" voreingestellten Wert. Beim "init_run" des RF620R ist der voreingestellte Wert "1" (int. Antenne an), beim RF630R "3" (Antennen 1 und 2 an).

Tabelle 3- 39 ATTRIBUTE "0x88" (Datentyp "IID_READER_STATUS_88_RF600")

Name	Тур	Kommentar	
status info	BYTE	SLG-Status mode (Subcommand)	
hardware	CHAR	Type of hardware	
hardware version	WORD	Version of hardware	
reserved word1	WORD	Reserved	
firmware	CHAR	Type of firmware	
firmware version HB	BYTE	Version of firmware (High-Byte)	
firmware version LB	BYTE	Version of firmware (Low-Byte)	
driver	CHAR	Type of driver	
current_time_hour	BYTE	Hours 1)	
current_time_minute	BYTE	Minutes 1)	
current_time_sec	BYTE	Seconds 1)	
current_time_reservByte	BYTE		
SLG version	BYTE	SLG-Version	
baud	BYTE	Baudrate	
reserved byte1	BYTE	Reserved	
distance limiting SLG	BYTE	Selected transmit power	
multitag SLG	BYTE	Multitag SLG	
field ON control SLG	BYTE	Selected communication type	
field ON time SLG	BYTE	Selected channel	
expert mode	BYTE	Expert mode	
status ant	BYTE	Status of antenna 2)	
scanning_time_SLG	BYTE	Radio communication profile (country specific radio standart)	
MDS control	BYTE	Presence mode	
blink pattern	BYTE	Blink Pattern	
act algor Single Tag	Bool	Single Tag [1]	
act algor ITF Phase2	Bool	ITF Phase2 [2]	
act algor ITF Phase1	Bool	ITF Phase1 [3]	
act algor Smoothing	Bool	Smoothing [4]	
act algor Blacklist	Bool	Blacklist [5]	
act algor RSSI Threshold	Bool	RSSI Threshold [6]	
act algor Power Ramp	Bool	Power Ramp [7]	
act algor Power Gap	Bool	Power Gap [8]	
Reserved1	Bool	Reserved1 [1]	
Reserved2	Bool	Reserved2 [2]	
Reserved3	Bool	Reserved3 [3]	
Reserved4	Bool	Reserved4 [4]	
act algor EPC MemBankFilter	Bool	EPC MemBankFilteres [5]	
act algor Tag Holg	Bool	Tag Hold [6]	
act algor Multi Tag	Bool	Multi Tag [7]	
act algor ISTM	Bool	ISTM [8]	
reserved word2	WORD	Reserved	
reserved word3	WORD	Reserved	
reserved word4	WORD	Reserved	
filtered max rssi	BYTE	Maximum RSSI value of a tag, of all filtered tags	
reserved byte2	BYTE	Reserved	
filtered tags rssi	BYTE	Number of tags, filtered out by the RSSI threshold	
reserved byte3	BYTE	Reserved	
filtered tags black list	WORD	Number of tags, filtered out via Black-List	
filtered tags epc data	WORD	Number of tags, filtered out via EPC Data Filter	
filtered tags smoothing	WORD	Number of tags in Tag List of status Not-Observed	
itf ph1 max detect	WORD	Number of reads of a Tag, filtered out via ITF-phase 1	
itf ph1 tags detect	WORD	Number of tags, filtered out via ITF-phase 1	
itf ph2 max detect	WORD	Number of reads of a Tag, filtered out via ITF-phase 2	
itf ph2 tags detect	WORD	Number of tags, filtered out via ITF-phase 2	

Name	Тур	Kommentar	
filtered_istm_min_dist	WORD	$\begin{array}{c} {\tt Minimum\ distance\ of\ tags\ according\ to\ sorting\ criterion} \\ {\tt of\ ISTM} \end{array}$	
filtered istm tags	WORD	Number of tags, filtered out via ISTM algorithm	
last error	BYTE	error code of the last occuring error (last command)	
reserved byte4	BYTE	Reserved	
error command1	WORD	Last command (has lead to error code) "last error"	
error command2	WORD	Last command (has lead to error code) "last error"	
error command3	WORD	Last command (has lead to error code) "last error"	
reserved word5	WORD	Reserved	
reserved_array_byte	ARRAY[130] of Byte		

Tabelle 3- 40 ATTRIBUTE "0x89" (Datentyp "IID_READER_STATUS_89_RF68xR")

Name	Тур	Kommentar
status info	BYTE	SLG-Status mode (Subcommand)
hardware version	BYTE	Version of hardware
firmware_version	ARRAY[14] of CHAR	Version of firmware
config ID	DWORD	Unix timestamp
inventory_status	WORD	0=inventory not active; 1=inventory active; 2=presence mode active
sum of filtered tags	WORD	All filtered Tags
filtered smoothing	WORD	Filtered Tags trough Smoothing
filtered blacklist	WORD	Filtered Tags trough Blacklist
filtered data-filter	WORD	Filtered Tags trough Data-Filter
filtered RSSI threshold	WORD	Filtered Tags trough RSSI Threshold
filtered RSSI delta	WORD	Filtered Tags trough RSSI Delta

Tabelle 3- 41 ATTRIBUTE "0xA0" und "0xA1" (Datenytp "IID_READER_STATUS_A0_A1_RF600")

Name		Тур	Kommentar
re	served	BYTE	
St	atus info	BYTE	Status-Info, SLG-Status SubCommand 20/21
nu	mber tags frame	BYTE	Number of Tags in this frame
nu	mber tags next frames	BYTE	Number of Tags in the next frames
re	served bytel	BYTE	Reserved
re	served byte2	BYTE	Reserved
re	served byte3	BYTE	Reserved
re	served byte4	BYTE	Reserved
re	served byte5	BYTE	Reserved
re	served byte6	BYTE	Reserved
Bl	ack_List_ID	ARRAY[113] of "IID_IN_Blacklist"	EPC-ID Length
	Black_List_ID[1]	"IID_IN_Blacklist"	
	EPC_Length	BYTE	EPC-ID Length
	Antenna	BYTE	Antenna = Default 3
	Filtered_Tag	WORD	Number of times - EPC-ID filtered out via Black- List
	EPC	ARRAY[112] of Byte	EPC-ID
Bl	ack_List_ID[2]	"IID_IN_Blacklist"	
Bl	ack_List_ID[3]	"IID_IN_Blacklist"	
Bl	ack_List_ID[4]	"IID_IN_Blacklist"	
Bl	ack_List_ID[5]	"IID_IN_Blacklist"	

Name	Тур	Kommentar
Black_List_ID[6]	"IID_IN_Blacklist"	
Black_List_ID[7]	"IID_IN_Blacklist"	
Black_List_ID[8]	"IID_IN_Blacklist"	
Black_List_ID[9]	"IID_IN_Blacklist"	
Black_List_ID[10]	"IID_IN_Blacklist"	
Black_List_ID[11]	"IID_IN_Blacklist"	
Black_List_ID[12]	"IID_IN_Blacklist"	
Black_List_ID[13]	"IID_IN_Blacklist"	

Weitere Details zu den einzelnen Status-Modi entnehmen Sie bitte den zu den Modi passenden Handbüchern "FB 45", "FB55" und "SIMATIC RF620R/RF630R".

Die hier angegeben Kennungen der Status-Modi entsprechen folgenden Kennungen in den anderen Handbüchern:

0x81	≙	0x01
0x82	≙	0x02
0x83	≙	0x03
0x85		0x05
0x87	≙	0x07
0x88	≙	0x08
0x90	≙	0x10
0x91	≙	0x11
0x92	≙	0x12
0xA0	≙	0x20
0xA1	≙	0x21

3.4.4.2 Tag_Status

Der Baustein "Tag_Status" liest die Statusinformationen des Transponders aus. Für die verschiedenen Transponder-Typen und Reader-Familien gibt es unterschiedliche Status-Modi, die Sie über den Parameter "ATTRIBUTE" auswählen können.

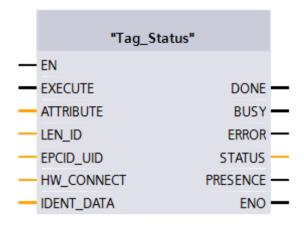


Bild 3-31 Baustein "Tag_Status"

Tabelle 3- 42 Erläuterung zum Baustein "Tag_Status"

Parameter	Datentyp	Default-Werte	Beschreibung
ATTRIBUTE	Byte	B#16#0	Kennung der Status-Modi / Mögliche Eingaben:
			• RF200: 0x83
			• RF300: 0x04, 0x82, 0x83 (nur ISO-Transponder)
			• RF620R, RF630R: 0x84, 0x85
			• MOBY D: 0x83 1)
			• MOBY U: 0x80
LEN_ID	Byte	B#16#0	Länge der EPC-ID/UID
			Default-Wert: 0x00 ≜ unspezifizierter Singletag-Zugriff (RF680R, RF685R)

Parameter	Datentyp	Default-Werte	Beschreibung
EPCID_UID	Array[162] of Byte	0	Puffer für bis zu 62 Byte EPC-ID, 8 Byte UID oder 4 Byte Handle-ID.
			2-62-Byte EPC-ID wird an Anfang des Puffer eingetragen (Länge wird durch "LEN_ID" beschrieben)
			8-Byte UID wird an Anfang des Puffers eingetragen ("LEN_ID = 8")
			4-Byte Handle-ID muss im Array- Element [5]-[8] eingetragen wer- den ("LEN_ID = 8")
			Default-Wert: 0x00 ≜ unspezifizierter Singletag-Zugriff (RF620R, RF630R, RF640R)
IDENT_DATA	Any / Variant	0	Ergebniswerte je nach Attribute
			Hinweis: Bei Variant sind ein "Array_of_Byte" mit variabler Länge und die vorhandenen Status-UDTs anlegbar. Bei Any können zusätzlich auch andere Datentypen/UDTs angelegt werden.

¹⁾ nur SLG D10S

Ergebnisse

Tabelle 3- 43 ATTRIBUTE "0x04" (Datentyp "IID_TAG_STATUS_04_RF300")

Name	Тур	Kommentar
reserved	BYTE	
status info	BYTE	MDS status mode
UID	ARRAY [18] of BYTE	
MDS type	BYTE	Type of MDS
Lock state	BYTE	Write Protection Status EEPROM
Reserved1	ARRAY[16] of BYTE	

Tabelle 3- 44 ATTRIBUTE "0x80" (Datentyp "IID_TAG_STATUS_80_MOBY_U")

Name	Тур	Kommentar	
UID	ARRAY [14] of BYTE	Unique indentifier (MDS-Number)	
MDS type	BYTE	Trype of MDS	
sum subframe access	DInt	Sum of subframe access	
sum searchmode access	INT	Sum of search mode access	
ST date Week	BYTE	Date of last sleep-time change (week of year)	
ST date Year	BYTE	Date of last sleep-time change (year)	
battery left	INT	Battery power left (percent)	
ST	BYTE	Actual sleep-time on MDS	

Tabelle 3- 45 ATTRIBUTE "0x82" (Datentyp "IID_TAG_STATUS_82_RF300")

Name	Тур	Kommentar
reserved	BYTE	
status info	BYTE	MDS status mode
UID	ARRAY [18] of BYTE	
LFD	BYTE	Magnetic flux density: correlation between limit-value
FZP	BYTE	Error counter passive: distortion without communication
FZA	BYTE	Error counter active: distortion during communication
ANWZ	BYTE	Presence counter: measure value for presence time
reserved1	ARRAY [13] of BYTE	

Tabelle 3-46 ATTRIBUTE "0x83" (Datentyp "IID_TAG_STATUS_83_ISO")

Name	Тур	Kommentar
reserved	BYTE	
status info	BYTE	MDS status mode
UID	ARRAY [18] of BYTE	
MDS Type	BYTE	Type of MDS
IC version	BYTE	Chip version
size HB	BYTE	Size of Memory (high Byte)
size LB	BYTE	Size of memory (low Byte)
lock state	BYTE	Write protection status EEPROM
block size	BYTE	Size of a block in addressable memory
number of block	BYTE	Number of blocks in addressable memory

Tabelle 3- 47 ATTRIBUTE "0x84" (Datentyp "IID_TAG_STATUS_84_RF600")

Name	Тур	Kommentar
reserved	BYTE	
status info	BYTE	MDS status mode
UID	ARRAY [18] of BYTE	
antenna	BYTE	Antenna which has observed the MDS
RSSI	BYTE	RSSI value
last observed hour	BYTE	Last observed time hour
last observed min	BYTE	Last observed time minute
last observed sec	BYTE	Last observed time seconds
last observed channel	BYTE	Last observed time channel
EPC length	BYTE	EPC-Length
reserved1	BYTE	

¹⁾ Es wird der interne Zeitstempel der readerinternen Uhr ausgegeben, welche sich auf dieses Ereignis bezieht. Die readerinterne Uhr wird nicht mit UTC synchronisiert.

Tabelle 3- 48 ATTRIBUTE "0x85" (Datentyp "IID_TAG_STATUS_85_RF600")

Name	Тур	Kommentar
status info	BYTE	MDS status mode
antenna	BYTE	Antenna which has observed the MDS
channel	BYTE	Channel

Name	Тур	Kommentar	
UID	ARRAY [18] of BYTE		
DT_glimpsed_1	BYTE	Time elasped between acknowledgement and first read in [ms]1 Highbyte	
DT_glimpsed_2	BYTE	Time elasped between acknowledgement and first read in [ms]2	
DT_glimpsed_3	BYTE	Time elasped between acknowledgement and first read in [ms]3	
DT_glimpsed_4	BYTE	Time elasped between acknowledgement and first read in [ms]4 Low-Byte	
reserved1	BYTE		
reserved2	BYTE		
reserved3	BYTE		
reserved4	BYTE		
last observed hour	BYTE	Last observed time hour	
last_observed_min	BYTE	Last observed time minutes 1)	
last_observed_sec	BYTE	Last observed time seconds 1)	
last observed EPC length	BYTE	Last observed time EPC length	
EPC_ID_Byte	ARRAY [162] of BYTE	EPC-ID	
reads HB	BYTE	Number of Reads of MDS in Inventory (1 - 65535)	
reads LB	BYTE	Number of Reads of MDS in Inventory (1 - 65535)	
RSSI	BYTE	Current RSSI value of MDS ²⁾	
mean RSSI	BYTE	Mean RSSI value of MDS	
max RSSI	BYTE	Max RSSI value of MDS	
min RSSI	BYTE	Min RSSI value of MDS	
min POWER	BYTE	Min Power value of MDS	
current_POWER	BYTE	Current Power value of MDS ³⁾	
reserved5	ARRAY[1137] of BYTE		

- 1) Es wird der readerinterne Zeitstempel ausgegeben, welcher sich auf dieses Ereignis bezieht. Der readerinterne Zeitstempel wird nicht mit UTC synchronisiert.
- Der Wert "Reads" gibt unabhängig von den eingestellten Smoothing Parametern die Summe der Transpondererkennungen (Inventories) an. Im Grenzfall kann der Zähler "Reads" auf diese Weise recht hohe Werte annehmen ohne dass der Transponder jemals den Zustand "Observed" erzielt hat.
- 3) Der "current_Power"-Wert wird als Sendeleistung in 0,25 dBm-Schritten (ERP/EIRP) angegeben. Ein "current_Power"-Wert von "72" (0x48) entspricht demnach 18 dBm (ERP/EIRP).

Weitere Details zu den einzelnen Status-Modi entnehmen Sie bitte den zu den Modi passenden Handbüchern "FB 45", "FB55" und "SIMATIC RF620R/RF630R".

Die hier angegeben Kennungen der Status-Modi entsprechen folgenden Kennungen in den anderen Handbüchern:

0x04	≙	0x01
0x82	≙	0x02
0x83		0x03
0x84	≙	0x04
0x85	Δ	0x05

3.5 Ident-Profil programmieren

3.5.1 Umstieg auf Ident-Bausteine/-Profil

Die Ident-Bausteine bzw. das Ident-Profil ersetzen den "PIB_1200_UID_001KB" und dessen Bausteine. Neben der Namensänderungen wurden auch funktionale Änderungen an dem Baustein vorgenommen. Beachten Sie folgende Punkte, wenn Sie ein vorhandenes Projekt mit PIB-Bausteinen/"PIB_1200_UID_001KB" aus der Bibliotheksversion V1.04 auf die Ident-Bausteine bzw. das Ident-Profil aus der Bibliothek V2.0 hochrüsten wollen:

- Löschen Sie alle bisherigen Bausteine aus dem Programm.
- Passen Sie jede Verwendungsstellung an den Aufruf der neuen Instruktion an.
- Ändern Sie den Datentyp der folgenden Variablen:
 - "HW CONNECT VAR" → "IID HW CONNECT"
 - "CMD STRUCT" → "IID CMD STRUCT"

Beispiel: Umstieg ohne Multiinstanz

Um von einem Baustein ohne Multiinstanz auf Ident-Bausteine/-Profil umzusteigen, gehen Sie wie folgt vor:

- 1. Löschen Sie alle bisherigen Bausteine ("PIB_1200_UID_001KB", "Read", "Write", usw.) und deren Instanz-DBs aus dem Ordner "Programmbausteine" der Projektnavigation.
- 2. Löschen Sie die bisherigen Datentypen "HW_CONNECT_VAR" und "CMD_STRUCT" aus dem Ordner "PLC-Datentypen" der Projektnavigation.

3.5 Ident-Profil programmieren

3. Ziehen Sie den gewünschten Ident-Baustein aus dem Bibliotheksregister in den geöffneten Baustein.

Achten Sie darauf den Namen des alten Bausteinaufrufs bei dem neuen Bausteinaufruf wieder zu verwenden (z. B. "Reset RF300 DB").

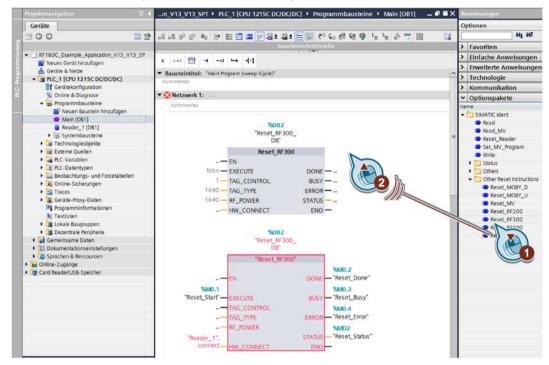


Bild 3-32 Ident-Baustein einfügen

- 4. Kopieren Sie die Variablen des alten Bausteinaufrufs in den neuen Bausteinaufruf (z. B. per Drag & Drop).
- 5. Löschen Sie den alten Bausteinaufruf.
- 6. Öffnen Sie den Datenbaustein in dem die Variable "HW CONNECT VAR" angelegt ist.
- 7. Notieren Sie sich die Adressparameter der Variablen.
- 8. Ändern Sie den Datentyp von "HW_CONNECT_VAR" in "IID_HW_CONNECT" und geben Sie die Adressparameter ein.
 - Wenn Sie eine Variable vom Typ "CMD_STRUCT" angelegt haben, ändern Sie den Datentyp in "IID_CMD_STRUCT".
- 9. Wiederholen Sie die Schritte 3 bis 5 für jeden angelegten Baustein.
- 10. Wiederholen Sie die Schritte 6 bis 8 für jeden Kanal/Reader.

Beispiel: Umstieg mit Multiinstanz

Um von einem Baustein mit Multiinstanz auf Ident-Bausteine/-Profil umzusteigen, gehen Sie wie folgt vor:

- 1. Löschen Sie alle bisherigen Bausteine ("PIB_1200_UID_001KB", "Read", "Write", usw.) und deren Instanz-DBs aus dem Ordner "Programmbausteine" der Projektnavigation.
- 2. Löschen Sie die bisherigen Datentypen "HW_CONNECT_VAR" und "CMD_STRUCT" aus dem Ordner "PLC-Datentypen" der Projektnavigation.
- 3. Öffnen Sie den Datenbaustein, in dem Sie einen Baustein als Multiinstanz verwenden.
- 4. Ändern Sie den Datentyp der Multiinstanz:
 - Bei einem PIB-Baustein (z. B. "Read")

Löschen Sie die Anführungszeichen des Datentyps.



Bild 3-33 Datentyp ändern

- Bei "PIB 1200 UID 001KB"
 - Ändern Sie den Datentyp von "PIB_1200_UID_001KB" in "Ident_Profile".
- 5. Öffnen Sie den Datenbaustein in dem die Variable "HW_CONNECT_VAR" angelegt ist.
- 6. Notieren Sie sich die Adressparameter der Variablen.
- 7. Ändern Sie den Datentyp von "HW_CONNECT_VAR" in "IID_HW_CONNECT" und geben Sie die Adressparameter ein.
 - Wenn Sie eine Variable vom Typ "CMD_STRUCT" angelegt haben, ändern Sie den Datentyp in "IID_CMD_STRUCT".
- 8. Wiederholen Sie die Schritte 3 bis 5 für jeden angelegten Baustein.
- 9. Wiederholen Sie die Schritte 6 bis 8 für jeden Kanal/Reader.

3.5 Ident-Profil programmieren

Bei beiden Varianten entfällt die Variablentabelle "PIB_CONSTANTS". Diese ist in der Bibliothek nicht mehr vorhanden und muss ebenfalls aus dem Anwenderprogramm gelöscht werden.

Hinweis

Inhalte der Variablentabelle "PIB_CONSTANTS" vor dem Löschen prüfen

Bevor Sie die Variablentabelle "PIB_CONSTANTS" löschen, überprüfen Sie, dass Sie keine eigenen Variablen in dieser Variablentabelle definiert haben. Haben Sie eigene Variaben angelegt, prüfen Sie, ob diese in eine andere Variablentabelle verschoben werden müssen.

3.5.2 Aufbau des Ident-Profils

Hinweis

Parallelbetrieb von Ident-Bausteinen und Ident-Profil ist nicht möglich

Beachten Sie, dass das CM bzw. der Reader nicht zeitgleich über die Ident-Bausteine und über das Ident-Profil betrieben werden kann.

Die in dem Kapitel "Ident-Bausteine programmieren (Seite 21)" beschriebenen Bausteine stellen eine vereinfachte Schnittstelle des Ident-Profils dar. Sollten die Funktionalitäten der Bausteine für Ihre Anwendung nicht ausreichen, können Sie alternativ das Ident-Profil verwenden. Mithilfe des Ident-Profil können Sie komplexe Befehlsstrukturen programmieren und mit Befehlswiederholung arbeiten. Die nachfolgende Grafik zeigt das Ident-Profil, inklusive der damit umsetzbaren Befehle.

Hinweis

Ident-Profil für geschulte Anwender

Das Ident-Profil ist ein komplexer Baustein und beinhaltet alle Funktionalitäten der Ident-Bausteine. Das Ident-Profil wurde speziell für geschulte Baustein-Anwender konzipiert, die komplexe Funktionen mit Hilfe eines einzigen Bausteins projektieren wollen. Ungeschulten Anwendern, empfehlen wir die Verwendung der Ident-Bausteine.

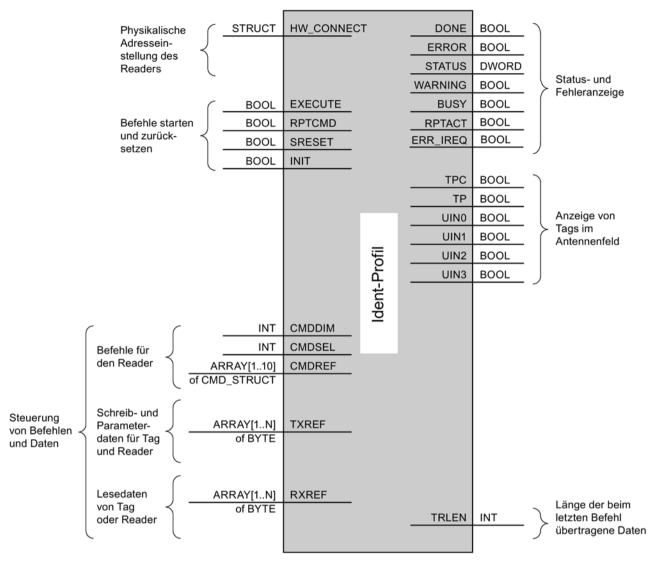


Bild 3-34 Die Input-Parameter des Ident-Profils

Hinweis

Arbeit mit mehreren Kanälen

Wenn Sie mit mehreren Kanälen arbeiten, müssen sie darauf achten, dass für jeden Kanal der Baustein mit einem eigenen Instanz-DB aufgerufen wird.

Schnittstellenbeschreibung

Tabelle 3-49 Eingangsparamater

Eingangsparamater	Datentyp	Default-Wert	Bedeutung
HW_CONNECT	HW_CONNECT		Eigener Datentyp zur physikalischen Adressierung von Kommunikationsmodul und Reader und zum Synchronisieren der Bausteine die pro Reader verwendet werden.
			Die Adressierung erfolgt wie in Kapitel "Datentyp "IID_HW_CONNECT" parametrieren (Seite 15)" beschrieben.
EXECUTE	BOOL	FALSE	TRUE = Anstoß eines neuen Befehls
			Vor dem Starten müssen Sie den Befehl und die entsprechenden Parameter, in den mit "CMDREF" verknüpften Speicher, setzen.
RPTCMD	BOOL	FALSE	TRUE = Wiederholung des aktuell ausgeführten oder als nächstes auszuführenden Befehls durch das Kommunikationsmodul
SRESET	BOOL	FALSE	TRUE = Abbruch des aktuell im Kommunikationsmodul verarbeiteten Befehls
INIT	BOOL	FALSE	TRUE = Kommunikationsmodul führt einen Reset aus und wird neu parametriert
CMDDIM	INT	10	Anzahl der Befehle im Parameter "CMDREF"
CMDSEL	INT	0	Auswahl des auszuführenden Befehls "CMDREF";
			1 ⇒ 1. Befehl,
			Der Wert des Parameters "CMDSEL" kann nie größer als der Wert des Parameters "CMDDIM" sein.
CMDREF	ARRAY[110]		Befehlsfeld
	of CMD_STRUCT		Das Feld kann bis zu 10 Befehle aufnehmen. Die Befehle sind komplexe Variablen des Typs "CMD_STRUCT". Weitere Informationen zu "CMDREF" finden Sie im Kapitel "Befehle des Ident-Profils (Seite 74)".
TXREF	ARRAY[1n] of		Verweis auf globalen Speicherbereich für Sendedaten.
	BYTE		Der Speicherbereich kann mit anderen Baustein-Instanzen geteilt werden. Der Wert "n" der einzelnen Bausteine ist varibale und kannbis zu 32 KB groß sein.
RXREF	ARRAY[1n] of		Verweis auf globalen Speicherbereich für Empfangsdaten.
	BYTE		Der Speicherbereich kann mit anderen Baustein-Instanzen geteilt werden. Der Wert "n" der einzelnen Bausteine ist varibale und kann bis zu 32 KB groß sein.

Tabelle 3- 50 Ausgangsparamater

Ausgangsparamater	Datentyp	Default-Wert	Bedeutung
DONE	BOOL	FALSE	TRUE = Befehl wurde fehlerfrei ausgeführt
ERROR	BOOL	FALSE	TRUE = Fehler wurde erkannt
			Der Fehler wird im Parameter "STATUS" ausgegeben. Das Bit wird mit dem Start eines neuen Befehls automatisch zurückgesetzt.
STATUS	DWORD	FALSE	Warnung- und Fehler
			Bei "ERROR = TRUE" bzw. "WARNING = TRUE" steht im Parameter "STATUS" die Fehler- bzw. Warninformation. Weitere Informationen finden Sie im Kapitel "Fehlermeldungen (Seite 115)".
WARNING	BOOL	FALSE	TRUE = Warnung wurde erkannt
			Die Warnung wird im Parameter "STATUS" ausgegeben. Wenn der Parameter "ERROR" nicht zeitgleich gesetzt ist, wurden die Daten korrekt bearbeitet. Das Bit wird mit dem Start eines neuen Befehls automatisch zurückgesetzt.
BUSY	BOOL	FALSE	TRUE = Der Baustein bearbeitet einen Befehl
			Andere Befehle, außer "INIT" und "SRESET", können nicht gestartet werden.
RPTACT	BOOL	FALSE	TRUE = "RPTCMD" ist aktiv
			Das Quittierungs-Bit zeigt an, dass der "Repeat-Mode" des CMs/Readers aktiv ist.
ERR_IREQ	BOOL	FALSE	TRUE = am Kommunikationsmodul oder Reader ist ein Fehler aufgetreten (z. B. beim Hochlauf oder Verbindungstrennung)
TPC	BOOL	FALSE	Transponder Presence Changed
			TRUE = neuer Transponder im Antennenfeld des Readers.
			Der Parameter wird nach erfolgreicher Ausführung des nächsten "INVENTORY"- oder "INIT"-Befehls auf "FALSE" gesetzt.
TP	BOOL	FALSE	Transponder Presence
			TRUE = ein Transponder befindet sich im Antennenfeld des Readers.
UIN0	BOOL	FALSE	Bei RFID-Readern wird die Anzahl der im Antennenfeld
UIN1	BOOL	FALSE	befindlichen Transponder angezeigt.
UIN2	BOOL	FALSE	Bei Code-Lesegeräte werden die verschiedenen Zustände des Code-Lesegerätes angezeigt.
UIN3	BOOL	FALSE	UIN0: Entspricht IN_OP Bit des Lesegeräts UIN1: Entspricht RDY Bit des Lesegeräts UIN2 + UIN3: Diese zwei Bits werden als unsigned Wert interpretiert (Bit 2 ist das niederwertigere Bit) welcher die Anzahl verfügbarer decodierter Codes darstellt. Wenn der Wert = 3 ist, dann sind drei oder mehr decodierte Codes verfügbar.
TRLEN	INT	0	Anzahl der Daten die nach erfolgreicher Ausführung des Befehls empfangen wurden.

3.5.3 Datenstruktur des Ident-Profils

Bei jedem Aufruf des Ident-Profils müssen Sie die Parameter ("HW_CONNECT", "CMDREF", "TXREF" und "RXREF"), wie im Kapitel "Aufbau des Ident-Profils (Seite 68)" beschrieben, versorgen.

Der Aufruf des Ident-Profils erfolgt immer über den Eingangsparameter "HW_CONNECT" und die "IN/OUT"-Parameter "CMDREF", "TXREF" und "RXREF". Alle drei Parameter müssen in einem Datenbaustein angelegt werden. Der Zusammenhang zwischen den drei "IN/OUT"-Parametern wird im Folgenden genauer beschrieben:

- CMDREF (Befehlspuffer):
 - Array[1...10] of CMD_STRUCT
- TXREF (Sendepuffer):
 - Array[1...n] of Byte
- RXREF (Empfangspuffer):
 - Array[1...n] of Byte

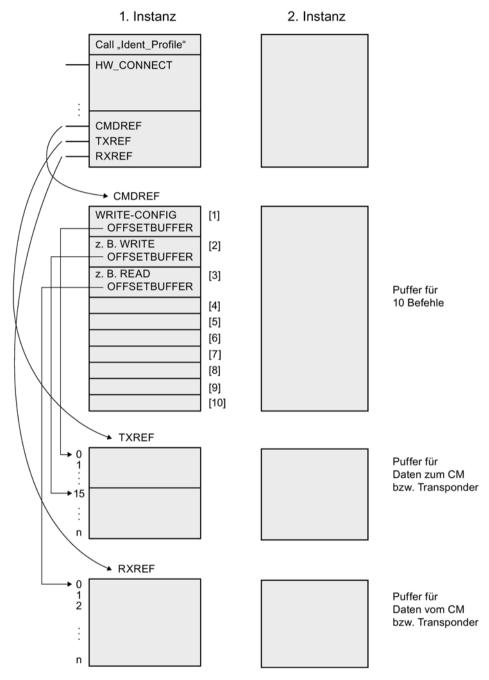


Bild 3-35 Datenstruktur-Beispiel des Ident-Profils

Erläuterung zu dem Beispiel

Eingehende Befehle:

• CMDREF[1]:

Befehl "WRITE-CONFIG", OFFSETBUFFER = 0

An der Stelle CMDREF[1] sollte stets der Befehl "WRITE-CONFIG" stehen, damit der "INIT/Reset" fehlerfrei ausgeführt wird.

• CMDREF[2]:

Befehl "WRITE", OFFSETBUFFER = 15

• CMDREF[3]:

Befehl "READ", OFFSETBUFFER = 0

Wird der Befehl "CMDREF[2]" ausgewählt, wird ein Schreibbefehl gestartet und die zu schreibenden Daten werden aus dem Parameter "TXREF" ab dem Byte 15 geholt. Wird der Befehl "CMDREF[3]" ausgewählt, werden die gelesenen Daten ab Byte 0 im Parameter "RXREF" abgelegt.

3.5.4 Befehle des Ident-Profils

Die nachfolgende Tabelle enthält alle Befehle, die das Ident-Profil und der Baustein "AdvancedCMD" unterstützen.

Tabelle 3-51 Befehle des Ident-Profil

Befehl	Befeh	Iscode	Verwendete Parameter	Beschreibung
	HEX	ASCII		
PHYSICAL-READ	70	'p'	OFFSETBUFFER, EPCID_UID, LEN_ID, LEN_DATA, ADR_TAG, MEM_BANK, PSWD	Liest Daten von einem Transponder/Code- Lesesystem durch Angabe der physikalischen Anfangsadresse, der Länge und des Passworts aus.
PHYSICAL-WRITE	71	'q'	OFFSETBUFFER, EPCID_UID, LEN_ID, LEN_DATA, ADR_TAG, MEM_BANK, PSWD	Schreibt Daten auf einen Transponder/Code- Lesesystem durch Angabe der physikalischen Anfangsadresse, der Länge und des Passworts.
READER-STATUS	74	't'	OFFSETBUFFER, ATTRIBUTES	Liest den Status des Kommunikationsmoduls/Readers aus.
TAG-STATUS	73	's'	OFFSETBUFFER, EPCID_UID, LEN_ID, ATTRIBUTES	Liest den Status eines Transponders aus.
INVENTORY	69	'i'	OFFSETBUFFER, ATTRIBUTES, DURATION, DUR_UNIT	Fordert eine Liste aller gegenwärtig zugänglichen Transponder innerhalb der Antennenreichweite an.
FORMAT	66	'f'	OFFSETBUFFER, EPCID_UID, LEN_ID, LEN_DATA	Initialisiert den Transponder.

Befehl	Befel	nlscode	Verwendete Parameter	Beschreibung
	HEX	ASCII		
PUT	65	'e'	OFFSETBUFFER, EPCID_UID, LEN_ID, LEN_DATA	Überträgt weitere Befehle, die nicht im Normprofil- Standard festgelegt sind. Dazu ist für jeden Befehl eine entsprechende Datenstruktur im Sendedaten- Puffer definiert.
WRITE-ID	67	ʻgʻ	OFFSETBUFFER,	RF680R/RF685R:
			EPCID_UID, LEN_ID, NEW-LEN_ID, PSWD	Schreibt eine neue EPC-ID in den Transponder.
KILL-TAG	6A	ʻjʻ	EPCID_UID, LEN_ID,	RF680R/RF685R:
			PSWD	Der Transponder wird permanent deaktiviert.
LOCK-TAG-BANK	79	'y'	EPCID_UID, LEN_ID,	RF680R/RF685R:
			PSWD, ACTION, MASK	Definiert ein Passwort für den Transponder-Zugriff.
EDIT-BLACKLIST	7A	ʻzʻ	EPCID_UID, LEN_ID,	RF680R/RF685R:
			MODE	Die Black List wird bearbeitet. Es kann der derzeitige Transponder hinzugefügt, alle erkannten Transponder hinzugefügt, einzelne Transponder gelöscht oder alle Transponder gelöscht werden.
GET-BLACKLIST	6C	11'	OFFSETBUFFER,	RF680R/RF685R:
			EPCID_UID, LEN_ID	Die gesamten TagIDs aus der Black List werden ausgelesen.
READ-CONFIG	61	'a'		Liest die Parameter aus dem Kommunikationsmodul/Reader aus.
WRITE-CONFIG	78	'x'	LEN_DATA, CONFIG	Sendet neue Parameter an das Kommunikations- modul bzw. den Reader.

3.5.4.1 Befehlsstruktur

Bevor Sie einen Befehl mit "EXECUTE" bzw. "INIT" starten können, müssen Sie diesen definieren. Für die einfache Definition eines Befehls wurde mit Hilfe des Datentyp "IID_CMD_STRUCT" der Befehlspuffer "CMDREF" erstellt. In dem Befehlspuffer stehen Ihnen 10 Bereiche zur Verfügung, in denen Befehle parametriert werden können. Über den Parameter "CMDSEL" wird festgelegt, welcher Befehl (1...10) mit "EXECUTE" gestartet wird.

Beachten Sie, dass das erste Element im Puffer immer für "INIT" reserviert ist. D. h. wenn "INIT" gesetzt wird, muss "CMDSEL" auf "1" stehen und das Element "1" im CMD-Puffer mit den entsprechenden Einstellungen gefüllt sein. Die folgende Tabelle enthält die Befehlsstruktur der Parameter. Nicht jeder Befehl nutzt alle Parameter.

Tabelle 3-52 Befehlsstruktur der Parameter

Parameter	Datentyp	Defaultwert	Beschreibung
CMD	BYTE	B#16#0	Befehlscode (vergleiche Tabelle im Kapitel "Befehle des Ident-Profils (Seite 74)")
OFFSETBUFFER	INT	0	Relativer Offset innerhalb des Empfangsdaten-Puffers. Der Parameter gibt die Adresse innerhalb des Speicherbereichs an, an der das erste Byte der empfangenen Daten gespeichert werden muss oder das erste Byte der zu sendenen Daten erwartet wird.
			Alle folgenden Bytes müssen in aufsteigenden Adressen gespeichert werden.
EPCID_UID	ARRAY[16 2] OF BYTE	B#16#0	Puffer für bis zu 62 Byte EPC-ID, 8 Byte UID oder 4 Byte Handle-ID.
			2-62-Byte EPC-ID wird an Anfang des Puffer einge- tragen (Länge wird durch "LEN_ID" beschrieben)
			8-Byte UID wird an Anfang des Puffers eingetragen ("LEN_ID = 8")
			4-Byte Handle-ID muss im Array-Element [5]-[8] eingetragen werden ("LEN_ID = 8")
			Default-Wert: 0x00 ≙ unspezifizierter Singletag-Zugriff
LEN_DATA	WORD	W#16#0	Anzahl der zu lesenden bzw. zu schreibenden Bytes
ADR_TAG	DWORD	DW#16#0	Physikalische Startadresse auf dem Transponder
ATTRIBUTES	BYTE	B#16#0	Subcommand-Bezeichner für einige Befehle (z. B. "DEV-STATUS", "INVENTORY", usw.)
CHAINED	BOOL	FALSE	0x00 = nicht verkettet
			0x01 = verkettet
			Alle verketteten Befehle müssen dieses Bit gesetzt haben, bis auf den letzten Befehl. Die Befehle werden in der Reihenfolge wie sie in der CMD-Struktur stehen abgearbeitet.
CONFIG	BYTE	B#16#0	0x01 = Reset, keine Konfigurationsdaten
			0x02 = kein Reset, zu sendende Konfigurationsdaten
			0x03 = Reset, zu sendende Konfigurationsdaten
			0x80 = kein Reset, nur Einzelparameter

Parameter	Datentyp	Defaultwert	Beschreibung
EXT_UHF	STRUCT		Struktur für Zusatzparameter (nur bei RF680R/RF685R)
LEN_ID	BYTE	B#16#0	Länge der gültigen Daten im Feld "EPCID_UID".
MEM_BANK	BYTE	B#16#3	Memory Bank auf dem Transponder
			0x00 = RESERVED
			• 0x01 = EPC
			• 0x02 = TID
			• 0x03 = USER
PSWD	DWORD	DW#16#0	Passwort für Transponder-Zugriff
			0x00 ≙ kein Passwort
EDIT_BLACKLIST	_MODE BYTE	B#16#0	Modus
			0x00 = TagID hinzufügen
			0x01 = Alle "Observed"-Transponder hinzufügen
			0x02 = TagID löschen
			• 0x03 = Alle löschen
INVENTORY_DUF	RATION WORD	W#16#0	Zeitdauer
			Zeitdauer oder Anzahl Inventories oder Anzahl der "Observed"-Events
			Bsp.:
			0x00 ≜ kein Inventory
			0x01 ≜ ein Inventory
INVENTORY_DUF	R_UNIT WORD	W#16#0	Einheit für "DURATION"
			• 0x00 = Zeit [ms]
			0x01 = Inventories
			0x02 = Anzahl der "Observed"-Events
LOCK-TAG- BANK_ACTION	WORD	W#16#0	Lock-Action (siehe "EPC-Specification")
LOCK-TAG-BANK	_MASK WORD	W#16#0	Lock-Mask (siehe "EPC-Specification")

3.5.4.2 Befehle

Tabelle 3-53 PHYSICAL-READ

CMD	OFFSET BUFFER	LEN_ DATA	ADR_ TAG	CHAINED	EPCID_ UID	LEN_ID	MEM_ BANK	PSWD	RXREF
0x70	Offset im Empfangs- puffer "RXREF"	Länge emp- fange- nen Daten	Adresse auf dem Trans- ponder	True = verkettet False = nicht verkettet	Puffer für bis zu 62 Byte EPC-ID, 8 Byte UID oder 4 Byte Handle-ID. • 2-62-Byte EPC-ID wird an Anfang des Puffer einge- tragen (Länge wird durch "LEN_ID" be- schrieben) • 8-Byte UID wird an Anfang des Puffers eingetra- gen ("LEN_ID = 8") • 4-Byte Handle-ID muss im Array- Element [5]-[8] eingetragen wer- den ("LEN_ID = 8") Default-Wert: 0x00 unspezifizierter Sin- gletag-Zugriff	Länge der EPC- ID (2-62 Byte) 0x00 ≜ unspezi- fizierter Single- tag- Zugriff	Spei-cherbank Ox00 ≜ Reser-ved Ox01 ≜ EPC Ox02 ≜ TID USER	Passwort 0x00 ≜ kein Passwort	ne Daten

Tabelle 3- 54 PHYSICAL-WRITE

CMD	OFFSET BUFFER	LEN_ DATA	ADR_ TAG	CHAINED	EPCID_ UID	LEN_ID	MEM_ BANK	PSWD	TXREF
0x71	Offset im Sende- puffer "TXREF"	Länge der zu schrei ben- den Daten	Adresse auf dem Transponder	True = verkettet False = nicht verkettet	Puffer für bis zu 62 Byte EPC-ID, 8 Byte UID oder 4 Byte Hand- Ie-ID. • 2-62-Byte EPC-ID wird an Anfang des Puffer eingetragen (Länge wird durch "LEN_ID" be- schrieben) • 8-Byte UID wird an Anfang des Puffers eingetragen ("LEN_ID = 8") • 4-Byte Handle-ID muss im Array- Element [5]-[8] eingetragen wer- den ("LEN_ID = 8") Default-Wert: 0x00 ≜ unspezifizierter Single- tag-Zugriff	Länge der EPC-ID (2- 62 Byte) 0x00 ≜ unspezifi- zierter Singletag- Zugriff	Spei- cherbank • 0x00 ≜ Reser- ved • 0x01 ≜ EPC • 0x02 ≜ TID • 0x03 ≜ USER	Passwort 0x00 ≜ kein Passwort	Zu schrei- bende Daten

Tabelle 3- 55 READER-STATUS

CMD	OFFSETBUFFER	ATTRIBUTES	RXREF
0x74	Offset im Empfangspuffer "RXREF"	Kennung der Status-Modi / Mögliche Eingaben: RF200: 0x81 RF300: 0x81, 0x86 RF620R, RF630R: 0x87, 0x88, 0xA0, 0xA1 RF680R, RF685R: 0x89 MOBY U: 0x81, 0x84, 0x85 MOBY D: 0x81	Empfangene Statusdaten Die Datenstruktur der Status-Modi finden Sie im Kapitel "Reader_Status (Seite 55)".

Tabelle 3-56 TAG-STATUS

CMD	OFFSETBUFFER	ATTRIBUTES	EPCID_ UID	LEN_ID	RXREF
0x73	Offset im Empfangspuffer "RXREF"	Kennung der Status- Modi / Mögliche Einga- ben: RF200: 0x83 RF300: 0x04, 0x82, 0x83 (nur ISO- Transponder) RF600, R680R, RF685R: 0x84, 0x85 MOBY D: 0x83 ¹⁾ MOBY U: 80	Puffer für bis zu 62 Byte EPC-ID, 8 Byte UID oder 4 Byte Handle-ID. • 2-62-Byte EPC-ID wird an Anfang des Puffer eingetragen (Länge wird durch "LEN_ID" beschrieben) • 8-Byte UID wird an Anfang des Puffers eingetragen ("LEN_ID = 8") • 4-Byte Handle-ID muss im Array-Element [5]-[8] eingetragen werden ("LEN_ID = 8") Default-Wert: 0x00 ≜	Länge der EPC-ID / UID	Empfangene Statusdaten Die Datenstruktur der Status-Modi finden Sie im Kapi- tel "Tag_Status (Seite 61)".
			unspezifizierter Single- tag-Zugriff		

¹⁾ Nur SLG D10S

Tabelle 3-57 INVENTORY

CMD	OFFSET BUFFER	ATTRIBUTES	INVENTORY_ DURATION	INVENTORY_ DUR_UNIT	RXREF
0x69	Offset im Empfangs- puffer "RXREF"	 Kennung der Status-Modi / Mögliche Eingaben: RF680R/RF685R: 0x80 ≜ Inventory mit kurzen Transponder-Informationen 0x81 ≜ Inventory mit vielen Transponder-Informationen 0x86 ≜ Presence-Mode an 0x87 ≜ Presence-Mode aus RF620R/RF630: 0x82 ≜ Auslesen eines Folgedatensatzes 0x83 ≜ Lese Handle-ID bei MOBY_mode ≜ 7 0x85 ≜ Lese Handle-IDs und EPC-IDs nach mittleren RSSI-Wert absteigend sortiert aus 0x91 ≜ Lese Handle-IDs nach maximalen RSSI-Wert absteigend sortiert aus 0x92 ≜ Lese Handle-IDs nach Lesehäufigkeit absteigend sortiert aus 0x92 ≜ Lese Handle-IDs nach Lesehäufigkeit absteigend sortiert aus 0xA0 ≜ Ersten Einträge aus Black List auslesen 0xA1 ≜ Weitere Einträge aus Black List auslesen RF300/MOBY U: 0x00 ≜ Liste aller Tags mit UID 	Nur bei 0x80 und 0x81: Zeitdauer Zeitdauer oder Anzahl Inventories oder Anzahl der "Observed"-Events Bsp.: • 0x00 ≜ kein Inventory • 0x01 ≜ ein Inventory	Nur bei 0x80 und 0x81: Einheit für "DURATION" • 0x00 ≜ Zeit [ms] • 0x01 ≜ Inventories • 0x02 ≜ Anzahl der "Observed"-Events	Bei RF680R/RF685R nur bei 0x80 und 0x81: Empfangene Daten Bei RF620R/RF630R/R F300/MOBY U: Empfangene Daten Die Datenstruktur der Status-Modi finden Sie im Kapitel "Inventory (Seite 30)".

Tabelle 3-58 FORMAT

CMD	OFFSETBUFFE R	LEN_DATA	EPCID_ UID	LEN_ID	TXREF
0x66	Offset im Sen- depuffer "TXREF"	Länge der zu über- tragenden Parame- terdaten	Puffer für bis zu 62 Byte EPC-ID, 8 Byte UID oder 4 Byte Handle-ID.	Länge der EPC-ID / UID	Zu schreibende Parameterdaten
			2-62-Byte EPC-ID wird an Anfang des Puffer eingetragen (Länge wird durch "LEN_ID" beschrieben)		
			8-Byte UID wird an Anfang des Puffers eingetragen ("LEN_ID = 8")		
			4-Byte Handle-ID muss im Array-Element [5]- [8] eingetragen werden ("LEN_ID = 8")		
			Default-Wert: 0x00 ≙ un- spezifizierter Singletag- Zugriff		

Tabelle 3-59 Aufbau des Datenanhangs für den Befehl "FORMAT" bei Normaladressierung

Byte	18	9	10	11	12	13	14	15
Wert	00h	06h	03h	00h	INIT-Wert	00h	MSB	LSB

Tabelle 3- 60 Erläuterung zum Aufbau des Datenanhangs für den Befehl "FORMAT"

Byte	Beschreibung
Byte 18	reserviert für Security-Code (muss mit "0" belegt sein, da SIMATIC RFID bisher keinen Code hat)
Byte 9	Länge der nachfolgenden Daten, hier 6
Byte 10	fest auf "0x03"
Byte 11	fest auf "0x00"
Byte 12	"INIT"-Wert: Mit diesem Wert wird der Datenbereich des Transponders beschrieben (Hex-Format).
Byte 13	fest auf "00h"
Byte 14	Speichergröße des Transponders (Endadresse + 1; High-Byte, Hex-Format)
Byte 15	Speichergröße des Transponders (Endadresse + 1; Low-Byte, Hex-Format)

Tabelle 3- 61 Speichergrößen der Transponder

	Transponder-Typ			INIT-Dauer
2 KB	MOBY U	RAM *)	08 00	ca. 1 s
32 KB	MOBY U	RAM *)	80 00	ca. 1,5 s
44 Byte	MOBY D	I-Code 1	00 2C	ca. 0,4 s
112 Byte	MOBY D	ISO I-Code SLI	00 70	ca. 0,5 s
256 Byte	MOBY D	ISO Tag-it HF-I	01 00	ca. 1 s
992 Byte	MOBY D	ISO my-d	03 E0	ca. 3 s
2000 Byte	MOBY D	FRAM	07 D0	ca. 3 s
20 Byte	RF300	EEPROM	00 14	ca. 0,2 s
8 KB	RF300	FRAM *)	20 00	0,9 s
32 KB	RF300	FRAM *)	80 00	3,6 s
64 KB	RF300	FRAM *)	FF 00	7,2 s

^{*)} Der OTP-Speicher wird nicht mit diesem Befehl initialisiert.

Tabelle 3-62 PUT

CMD	OFFSETBUFFER	LEN_DATA	TXREF
0x65	Offset im Sendepuffer "TXREF"	Länge der zu übertragenden Parameterdaten	Zu schreibende Parameterdaten

Tabelle 3-63 Datenstruktur des PUT-Befehls

Put_SET_ANT		Schaltet die Antenne des Readers aus und ein.						
		1 2 3						
		'N' 'A' Mode						
	Mode	RF200/RF300, MOBY U/D:						
		0x01 ≜ Antenne ein						
		0x02 ≙ Antenne aus						
		RF600:						
		Bit 0 ANT 1 / interne Antenne (1 = ein)						
		Bit 1 ≙ ANT 2 / externe Antenne (1 = ein)						
		Bit 4 ≙ TagList (0 = initialisieren, 1 = mit vorhandener Liste weiterarbeiten)						
	Length	3						
Put_E	END	Beendet die Kommunikation mit einen Transponder (nur bei MOBY U).						
		1 2 310 11						
		'N' 'K' UID Mode						
	UID	UID des Transponders						
	Mode	0x00 Bearbeitung des Transponders beenden						
		0x01 Bearbeitungspause des Transponders						
	Length	11						

Tabelle 3- 64 WRITE-ID (nur bei RF680R/RF685R)

CMD	OFFSET BUFFER	EPCID_ UID	LEN_ID	LEN_DATA	PSWD	TXREF
0x67	Offset im Sendepuffer "TXREF"	Bisherige EPC-ID 0x00 ≙ unspezifi- zierter Singletag- Zugriff	Länge der bisherigen EPC-ID (2-62 Byte) 0x00 ≙ unspezifizierter Singletag-Zugriff	Länge der neuen EPC-ID	Passwort 0x00 ≙ kein Passwort	Neue EPC-ID

Tabelle 3-65 KILL-TAG (nur bei RF680R/RF685R)

CMD	EPCID_ UID	LEN_ID	PSWD
0x6A	EPC-ID	Länge der EPC-ID (2-62 Byte)	Passwort
	0x00 ≙ unspezifizierter Singletag- Zugriff	0x00 ≙ unspezifizierter Singletag- Zugriff	muss ≠ 0x00 sein

Tabelle 3- 66 LOCK-TAG-BANK (nur bei RF680R/RF685R)

CMD	EPCID_ UID	LEN_ID	PSWD	LOCK_TAG_ BANK_ACTION	LOCK_TAG_ BANK_MASK
0x79	EPC-ID 0x00 ≙ unspezifi- zierter Singletag- Zugriff	Länge der EPC- ID (2-62 Byte) 0x00 ≙ unspezifi- zierter Singletag- Zugriff	Passwort 0x00 ≙ kein Passwort	Siehe EPC-Norm	Siehe EPC-Norm

Tabelle 3- 67 EDIT-BLACKLIST (nur bei RF680R/RF685R)

CMD	EDIT_ BLACKLIST_MODE	EPCID_ UID	LEN_ID
0x7A	 0x00 ≜ EPC-ID hinzufügen 0x01 ≜ Alle "OBSERVED"- Transponder hinzufügen 0x02 ≜ EPC-ID löschen 0x03 ≜ Alle löschen 	EPC-ID 0x00 ≜ unspezifizierter Singletag- Zugriff ¹)	Länge der EPC-ID (2-62 Byte) 0x00 ≙ unspezifizierter Singletag- Zugriff

Wenn bei "EDIT_BLACKLIST_MODE" = 0x00 oder 0x02 ausgewählt wurde, muss die EPC-ID inkl. der ID-Länge angegeben werden.

Tabelle 3- 68 GET-BLACKLIST (nur bei RF680R/RF685R)

CMD	OFFSETBUFFER	RXREF
0x6C	Offset im Empfangspuffer "RXREF"	Gelesene BlackList-IDs

Tabelle 3- 69 READ-CONFIG

CMD	OFFSETBUFFER	RXREF
0x61	Offset im Empfangspuffer "RXREF"	Gelesene Reset-Parameter

Tabelle 3- 70 WRITE-CONFIG

CMD	LEN_DATA	CONFIG	TXREF
0x78	Länge der Parameterdaten	0x01 ≜ Kommunikations-Reset, keine Konfigurationsdaten	Zu sendende Konfigurationsdaten
		0x02 ≜ kein Kommunikations-Reset, zu sendende Konfigurationsdaten	
		0x03 ≜ Kommunikations-Reset, zu sendende Konfigurationsdaten	
		0x80 ≙ kein Kommunikations-Reset, Einzelparameter	

Tabelle 3-71 Aufbau der Konfigurationsdatenanhangs (gültig für RF200, RF300, RF620R, RF630R, MOBY D/U)

Byte	1	25	6	78	9	10	11	12	1314	15	16
Wert	04h	0	0Ah	0	scanning_ time	param	opti- on_1	distance_ limiting	Anzahl der Trans- ponder	field_on_c ontrol	field_on_ time

Tabelle 3- 72 Bytes des Parameters "PARAM"

Byte	Wert	RFID- System	Bedeu	itung					
Byte 9	scanning_ time	MOBY U	Transp sofort "scanr	"scanning_time" beschreibt die Standby-Zeit für den Transponder. Erhält der Transponder vor Ablauf der "scanning_time" einen weiteren Befehl, so kann dieser sofort bearbeitet werden. Erhält der Transponder einen Befehl nach Ablauf der "scanning_time", so wird die Befehlsbearbeitung um die "sleep_time" des Transponder verzögert.					
			Eine "s	scanning_time" sollte nur eingestellt werde	en, wenn				
			• de	r Transponder mit mehreren Befehlen bea	rbeitet wird un	d			
				r Bearbeitungsvorgang in einer minimalen			muss.		
			0,,00	Lucius Chandley Zait (Dafaydh)					
				keine Standby-Zeit (Default)					
				7 ms Standby-Zeit					
			0x02 =	= 14 ms Standby-Zeit					
			C8 ho	x = 1400 ms Standby-Zeit					
				ten Sie, dass die "scanning_time" die Batt	toriolohonadau	or booinflu	oot lo		
			größei Genau	r die eingestellte "scanning_time",desto kü iere Berechnungen finden Sie im MOBY L ge und Service.	irzer die Batter	rielebensda	auer.		
		RF200, RF300, MOBY D		0x00 (reserviert)					
		RF600	"scanning_time" beschreibt das Funkprofil nach EPC Global. Je nach Land, Sie den Reader betreiben möchten, müssen Sie die korrekte Norm einstelle informieren Sie sich vor der Auswahl eines Funkprofils, welche Norm und w Reader-Typ für Ihr Land zutreffend ist.						
				7.	RF600-Reader-Variante				
			Wert	Beschreibung	ETSI	FCC	CMIIT		
			0	Keine Norm ausgewählt; es wird der Fehler "0x15" ausgegeben					
			1	Reader arbeitet mit dem voreingestellten Funkprofil. Wert des voreingestellten Funkprofils:	ETSI neu	FCC	China		
			2	ETSI neu: EU, EFTA, Türkei; 4-Kanal-Plan	X				
			3	ETSI alt: EU, EFTA, Türkei; Reader die nach dem 31.12.2009 in Betrieb genommen wurden, dürfen nicht mit dieser Einstellung betrieben werden.	Х				
			4	FCC: z. B. USA, Kanada, Mexiko		Х			
			5	reserviert					
			6	China					
			7	Thailand		Х			
			8	Brasilien		Х			
		1	1	Südkorea	1	1	i		

Byte	Wert	RFID- System	Bedeut	Bedeutung						
			C0	Indien		Х				
				inweis: Wenn Sie andere Länderprofile auswählen, als für die jeweilige Read ariante definiert sind, wird die Fehlermeldung "09" quittiert.						
Byte 10	param	RF200,	_							
		RF300,	Bit	instellung der RFID-Betriebsart und Anwesenheitskontrolle it 7 5 4 3 0						
		MOBY D, MOBY U		Anwesenheitskont Transponder-Steu		reserviert	RFII	D-Betriebs	art:	
				0 = keine ANW	/-Kontrolle					
				1 = keine Tran Steuerung;	sponder-					
				ANW-Kontrolle üb	er Firmware					
				(Default)						
			Wert des Bit 3 0	Betriebsart	Hinweis					
			0	reserviert	reserviert für d Parametrierung		g mit	Schalter o	der GSD-	
			1	reserviert	Kurzer "INIT" (nur die Parameter "param" und "option_1" werden zum Reader übertragen).				ınd "opti-	
			5	MOBY U/D, RF200, RF300 - ohne Multitag- Handling	ASM 475, ASM 456, RF170C, RF180C					
			6	MOBY U - mit Mulititag- Handling (FB 55)	einem Tran	rierung mit Multitag > 1 und mehr als ansponder im Antennenfeld: Der Para- D muss mit der Transponder-ID versor			er Para-	
					Parametrierung mit Multitag = 1 und nur ein Transponder im Antennenfeld: Der Parameter UID kann mit der richtigen Transponder-ID oder Null versorgt werden. ANN 455 ANN 455 DE 4550 DE 4550			ameter		
			7	MORY D. DE200	•	1 456, RF170C, RF180C				
			7	MOBY D, RF300 - mit Mulititag- Handling (FB 55)	ASIN 475, ASIN	ASM 475, ASM 456, RF170C, RF180C				
				s: Beachten Sie, dass alten eines CM erfolg		von des Pa	ramet	ers nur nad	ch dem	
		RF600	Einstel	lung der RFID-Betrieb	bsart					
			Wert	Betriebsart						
			4	ISTM-Mode						
			5	Singletag-Betrieb						
			6	ID	Handling (UID =					
				_	andling, 4 Byte Unit beliebig lange		lle-ID	für Zugriffe	e auf	

Byte	Wert	RFID- System	Bedeutung
			 mit Singletag-Handling (UID = 0x00), 8 Byte UID mit Multitag-Handling, 8 Byte UID der Bytes 5-12 der 12 Bytes langen EPC-ID
			Hinweis: Beachten Sie, dass eine Änderung von des Parameters nur nach dem Einschalten eines CM erfolgen darf.
Byte 11	option_1	RF200, RF300, MOBY D, MOBY U	Dieses Byte ist bitweise kodiert. Es hat standardmäßig den Wert "B#16#0". Mit diesem Byte können Sie besondere Steuerungen im CM/Reader vorgenommen werden. Bit 7 6 5 4 3 2 1 0 1 = Das Blinken der ERR-LED wird durch einen init_run zurückgesetzt. Bei RF200/RF300 wird damit das Blinken der ERR-LED auf dem Kommunikationsmodul und auf dem Reader zurückgesetzt.
		RF600	Dieses Byte ist bitweise kodiert. Es hat standardmäßig den Wert "B#16#0". Bit 7 6 5 4 3 2 1 0 1 = Das Blinken der ERR-LED des CM wird durch einen init_run zurückgesetzt Black List: 0 = OFF 1 = ON
Byte 12	distance_ limiting	MOBY U	Reichweitenbegrenzung normale Sendeleis- tung 1) Reichweitenbegrenzung reduzierte Sendeleistung
			0x05 = 0,5 m0x85Stellen Sie die reduzierte Sendeleistung ein, wenn0x0A = 1,0 m0x8A0x8A0x0F = 1,5 m0x8F0x8F0x14 = 2,0 m0x910x910x19 = 2,5 m0x990x990x1E = 3,0 m0x9E0x9E0x23 = 3,5 m0xA30xA3 Stellen Sie die reduzierte Sendeleistung ein, wenn mehrere Reader nahe beieinander positioniert sind oder wenn Transponder, die sich in der Nähe eines Readers aufhalten, später oder nicht mehr erkannt werden sollen.0x19 = 2,5 m0x99Nachteil: Die Feldkeule wird kleiner und damit steht für die Kommunikation weniger Zeit zur Verfügung bzw. es muss genauer positioniert werden.

Byte	Wert	RFID- System	Bedeutui	Bedeutung				
		MOBY D	Sendelei	Sendeleistung von 0,5 W bis 10 W in 0,25 W-Schritten				
			nur wirks (04 hex)	nur wirksam bei SLG D10S; bei SLG D11S / D12S ist eine Leistung von 1 W (04 hex) eingestellt und kann nicht geändert werden.				
			0x02 = 0	,5 W				
			:					
			0x10 = 4	W (default)				
			:					
			0x28 = 1					
		RF200	0x00 (res					
		RF300 (nur RF380R)	dern. Bea kungen a einzuhalt	Mit diesem Parameter können Sie die Sendeleistung des Readers RF380R ändern. Beachten Sie dabei, dass die Änderung der Sendeleistung sowohl Auswirkungen auf die obere als auch auf die untere Grenzreichweite, sowie auf den einzuhaltenden Mindestabstand zwischen benachbarten RF380R hat. Weitere Informationen dazu finden Sie im "Systemhandbuch RF300".				
			Folgende	e Einstellungen sii	nd möglich :			
			Bit Sendeleistung 02 0,5 W 03 0,75 W					
			04 1,0 W					
			05		1,25 W			
			06		1,5 W			
			07		1,75 W			
			08		2,0 W			
			Defaultw	ellungen außerha ert = 1,25 W eing ausgegeben.				
		RF600	Mit "dista	nce_limiting" wire	die Sendelei	stung des Rea	aders eingestell	t.
			Bit: 7	6 5 4 3 2 1	0			
								
			ext. A	NT 2 / ANT 1 Antenne int. Ante OF) (0F	nne			
			Per Defa	ult wird die ANT 1	mit der vorei	naestellten Se	endeleistuna ve	rwendet
			Per Default wird die ANT 1 mit der voreingestellten Sendeleistung verwendet. Hex- RF630R RF620R RF620					RF620R
			Wert	Sendeleistung	Strahlungs	leistung (inter	ne Antenne)	Sendeleis- tung
					ETSI	FCC	CMIIT	
				dBm / (mW)	dBm / (mW) ERP	dBm / (mW) EIRP	dBm / (mW) ERP	dBm / (mW)

Byte	Wert	RFID- System	Bedeutur	ng				
			0	18 / (65)	18 / (65)	20 / (105)	18 / (65)	18 / (65)
			1	19 / (80)	19 / (80)	21 / (130)	19 / (80)	19 / (80)
			9	27 / (500)	27 / (500)	29 / (795)	27 / (500)	27 / (500)
			Α	27 / (500)	28 / (630)	30 / (1000)	28 / (630)	27 / (500)
			B (F)	27 / (500)	29 / (800)	31 / (1260)	29 / (800)	27 / (500)
Byte	Anzahl der	RF600	Anzahl d	er erwarteten Tra	nsponder im A	Antennenfeld.		
1314	Transpon- der		Zugelass	ene Werte:				
	uei		• 0x01	0x28 für RF620)R			
			• 0x01	0x50 für RF630	OR mit 2 Ante	nnen (SET-AN	IT = 0x03)	
			• 0x01 0x02)	0x28 für RF630 	OR mit 1 Ante	nne (SET-ANT	= 0x01 oder S	ET-ANT =
				hier angegeben spondern (EPC-II			ende Anzahl vo	n zu lesen-
			Der Wert beschränkt nicht die Anzahl von zu bearbeitenden Transpondern im Antennenfeld. Um ein performantes Inventory von Transpondern im Antennenfeld zu ermöglichen, sollten die hier angegeben Werte von den maximal im Antennenfeld zu erwartenden Transpondern nicht um mehr als ca. 10 % abweichen.					
Byte 15	field_on_ control	MOBY U		etriebsart; automa enne EIN/AUS" w				des. Der Be-
			0x00	ohne BEROs; ke	ine Reader-S	ynchronisation	า	
			0x01	ein oder zwei BE	ROs			
				Die BEROs sind gung eines BER				t der Betäti-
			0x02	ein oder zwei BE	ROs			
			Der 1. BERO schaltet das Antennenfeld ein und der 2. BERO schaltet das Antennenfeld aus. Wenn zwei BEROs vorhanden sind und ein "field_ON_time" parametriert ist, wird das Antennenfeld automatisch ausgeschaltet, wenn der 2. BERO nicht innerhalbd ieser BERO-Zeit schaltet.					und ein omatisch
			Ist keine "field_ON_time" parametriert, bleibt das Antennenfeld bis zur Betätigung des 2. BERO eingeschaltet.					feld bis zur
			0x03					
				Weitere Informat rung, Montage u		Sie im MOBY	U-Handbuch fü	r Projektie-
		RF200, RF300, MOBY D	0x00 (res	serviert)				

Byte	Wert	RFID- System	Bedeutur	ng	
		RF600	und Tag	ON_control" wird die Kommunikationsgeschwindigkeit (schnell/langsam) Hold (ON/OFF) eingestellt. 5 4 3 2 1 0 res. Geschwindigkeit 0x00 = schnelle Erkennung 0x01 = reserviert 0x02 = sichere Erkennung 0x03 = reserviert	
				Tag Hold: 0 = OFF 1 = ON = ScanningMode OFF = ScanningMode vorbereitet	
			Parametr	schwindigkeit werden applikationsabhängig optimierte Reader- rierungen bereitgestellt: = schnelle Erkennung	
				= langsamere, sichere Erkennung	
				Mode (relevant beim Multitag-Betrieb):	
				= 0: Normaler Multitag-Betrieb (einschließlich "repeat_command")	
			• Bit 6 = 1: Unspezifierte Lesebefehle (UID = 0x00) werden vom CM/Reader auch akzeptiert, wenn sich mehrere Transponder im Antennenfeld befinden.		
			Durch Se	etzen des Bit 6 auf 1 wird der Reader im Multitag-Betrieb für die Verwen- s "ScanningMode" vorbereitet.	
Byte 16	field_on_	MOBY U	Zeit für B	ERO-Betriebsart (field_ON_control = 02)	
	time		0x00	Die Zeitüberwachung ist abgeschaltet. Für die Feldabschaltung wird der 2. BERO benötigt.	
			0x01 0xFF	1 255 s Einschaltzeit für das Reader-Antennenfeld	
		MOBY D	Transpor	nder-Typ	
			0 255	Transponder-Typ	
			0x00	I-Code 1 (z. B. MDS D139)	
			0x01	ISO-Transponder	
			0x02	I-Code 1 und ISO-Transponder	
			0x03	ISO-my-D	
				(nur bei SLG D10S; bei SLG D11S / D12S wird für den ISO-my-D der Wert "0x01" parametriert)	
			0x04	ISO-FRAM (nur bei SLG D11S / D12S; bei SLG D10S wird für den ISOFRAM der Wert "0x01" parametriert)	
		RF200	Transponder-Typ		
			0x01 Beliebiger ISO-Transponder		
		RF300		des Parameter "field_on_time" können Sie festlegen, ob der Reader im lodus oder im ISO15693-Modus betrieben wird (ein Mischbetrieb ist nicht	
			Folgende	e Werte können eingestellt werden:	

Byte	Wert	RFID- System	Bedeutu	ng			
			0x00	RF300	Für alle Tran	nsponder des Typs "RF3xxT"	
			0x01	Beliebiger ISO- Transponder	Aktivierung des allgemeinen ISO-Modus mit rud ren ISO-Befehlen. Mit dieser Einstellung ist in de gel der Betrieb mit jedem ISO-kompatiblen Transponder grundsätzlich gewährleistet (siehe weis).		
			0x03	ISO my-d (In- fineon SRF 55V10P)	z. B. MDS D	324, D339	
			0x04	ISO (Fujitsu MB89R118)	z. B. MDS D D428, D460	421, D422, D423, D424, D425, D426,	
			0x05	ISO I-Code SLI (NXP SL2 ICS20)	z. B. MDS D	100, D124, D126, D139, D150, D165	
			0x06	ISO Tag-it HFI (Texas Instru- ments)	z. B. MDS D200 (MLFB 6GGT2600-1AA00-0AX0). D261		
			0x07	ISO (ST LRI2K)	z. B. MDS D D261	200 (MLFB 6GGT2600-1AA01-0AX0),	
			Hinweis:		•		
			• Folge	Folgende ISO-Spezialfunktionen werden nicht unterstützt:			
			- A	FI (Application Fa	mily Identifier)	
				SFID (Data Stora	_	•	
						en wie EAS, Kill-Kommandos, etc.	
				identifiziert werde		onder aufgrund der obigen Parameter eine Fehlermeldung ("error_MOBY	
				ılässige Paramete wiesen.	r werden mit I	Fehlermeldung ("error_MOBY 15"[hex])	
			sche onsz	Befehle verwende eit zwischen Read	Werte "03 07" werden, soweit vorhanden, chipspezifindet. Die Befehle wirken sich positiv auf die Kommunikatieader und Transponder aus und können dadurch, abhänginuch einen schnelleren Datentransfer ermöglichen.		
		RF600	ETSI-/In	dien-Variante: 0x0	00 0x0F	,	
				ordnung im ETSI- ng_time = 0x02") å	•	Kanalzuordnung im Indien-Funkprofil ("scanning_time = 0xC0") ändern:	
				es. 865 866,3 M 866,9 MHz 867,5 MHz		Bit: 7 6 5 4 3 2 1 0 res. 865,1 MHz 866,3 MHz 866,9 MHz 866,9 MHz	

Byte	Wert	RFID- System	Bedeutung
			0x00: Voreinstellung; die Kanäle des Readers werden im vier Kanal-Betrieb verwendet.
			Anmerkung: Die Einstellung "0x0F" ist identisch mit "0x00".
			Über die Bits 03 des Bytes "field_ON_time" kann ein Kanal-(Frequenz-)Plan erstellt werden, für den Fall, dass mehrere Reader in unmittelbarer Nachbarschaft betrieben werden sollen. Reader, die unterschiedliche Kanäle verwenden, werden sich weniger gegenseitig beeinflussen.
			Für den Fall, dass nur 1 Kanal pro Reader verwendet wird, muss der Reader nach 4 Sekunden jeweils eine Pause von 100 ms (ab ETSI EN 302 208 V1.2.1) einhalten. Damit ist bei zeitkritischen Anwendungen ein geringer Performanzverlust gegenüber 2- bis 4-Kanalbetrieb eines Readers zuverzeichnen.
			Für den Fall, dass 2 bis 4 Kanäle pro Reader verwendet werden, wechselt der Reader bei Zwei-Antennen-Betrieb nach 0,1 Sekunden auf einen anderen Kanal und bei Ein-Antennen-Betrieb nach 4 Sekunden auf einen anderen Kanal. Wenn nur einer der 4 Kanäle ausgewählt ist, wird nach Norm zwangsweise nach 4 Sekunden eine Pause von 100 ms ausgeführt.
			FCC- und CMIIT-Variante: Normal: 0x00

Backup & Restore (bei RF68xR)

Für den Baugruppentausch ist es möglich alle Konfigurationsdaten aus dem Reader zu lesen und in der Steuerung zu speichern. Bei einem Baugruppentausch können diese Daten von der Steuerung in den Reader geladen werden. Dazu werden die Befehle "WRITE-CONFIG" (Config = 3) für den Download zum Reader und "READ-CONFIG" für den Upload aus dem Reader genutzt.

3.5.4.3 Erweiterte Befehle für Code-Lesesysteme (MV400)

Der Befehl "PYSICAL-WRITE"

Die Code-Lesesysteme MV400 besitzen weitere Befehle, die mit dem Befehl "PYSICAL-WRITE" übertragen werden.

Tabelle 3-73 PHYSICAL-WRITE

CMD	OFFSET BUFFER	LEN_DATA	ADR_TAG	TXREF
0x71	Offset im Sende- puffer "TXREF"	Länge der an den Codeleser zu übertragenden Daten:	0x0000	An den Codeleser zu übertragender Sub-Befehl mit Daten. Das erste Byte beinhaltet den Be-
		• 01		fehlsbezeichner:
		• 01		01 = Programmwechsel
		Matchstring-Länge + 3		02 = Programmnummer lesen aktivieren
		• 01		03 = Matchstring schreiben
		• 01		04 = Matchstring lesen aktivieren
		• 01		05 = Disa-Bit setzen
		Gesamtlänge in XMATCH-Nutzdaten + 4		06 = Disa-Bit zurücksetzen
		• 07		07 = Triggersynchrones Matchstring schrei- ben (XMATCH)
				08 = Digital Out setzen

Tabelle 3-74 Befehl-Datenbereich "TXREF" Befehlsbezeichner 03 (Matchstring schreiben)

Adresse	Wert	Bedeutung
0x0000	0x03	Befehlbezeichner "Matchstring schreiben"
0x0001	0x00-0xFF	Länge Matchstring High Byte
0x0002	0x00-0xFF	Länge Matchstring Low Byte
0x0003		1. Zeichen des Matchstrings
n + 2		(n-1). Zeichen des Matchstrings
n + 3		n. Zeichen des Matchstrings

Tabelle 3-75 Befehl-Datenbereich "TXREF" Befehlsbezeichner 07 (XMATCH)

Adresse	Wert	Bedeutung
0x0000	0x07	Befehlsbezeichner "XMATCH"
0x0001	0x00	Reserviert
0x0002	Ausführliche Informationen	XMATCH-Nutzdaten
	finden Sie im Handbuch	
0xN	"SIMATIC MV420 / SIMATIC MV440".	

Tabelle 3-76 Befehl-Datenbereich "TXREF" Befehlsbezeichner 08 (Digital Out setzen)

Adresse	Wert	Bedeutung		
0x0000	0x08	Befehlbezeichner "Digital Out setzen".		
0x0001	0x1-0x4	Nummer des logischen externen Signals. Entspricht "EXT_1", "EXT_2", "EXT_3" und "EXT_4".		
0x0002	0x0-0x2	Pegel des Signals		
		0x0: Pegel statisch auf "low" setzen.		
		0x1: Pegel statisch auf "high" setzen.		
		0x2: Pegel für die konfigurierte Pulszeit auf "high" setzen.		
0x0003	0x1-0x7	Verknüpfungsart		
		0x1: logisches "Oder"		
		0x2: logisches "Und"		
		0x3: logisches "Exklusiv-Oder"		
		0x4: keine Verknüpfung		
		0x5: logisches "Oder Nicht"		
		0x6: logisches "Und Nicht"		
		0x7: logisches "Exklusiv-Oder Nicht"		
0x0004	0x0-0x5	Logisches Signal mit dem verknüpft wird.		
		Wenn die Verknüpfungsart 0x4 ist, hat der Parameter keine Bedeutung.		
		0x0: logisches Signal "IN_OP"		
		0x1: logisches Signal "TRD"		
		0x2: logisches Signal "RDY"		
		0x3: logisches Signal "READ"		
		0x4: logisches Signal "MATCH"		
		0x5: logisches Signal "NOK"		
0x0005	0x0	Reserviert, muss zur Wahrung der Aufwärtskompatibilität 0x0 sein.		
0x0006	0x0	Reserviert, muss zur Wahrung der Aufwärtskompatibilität 0x0 sein.		

Der Befehl "PHYSICAL-READ

Der Befehl "PHYSICAL-READ" wird für folgende Funktionen verwendet:

- Lesen von Codes
- Folgebefehl nach "Programmnummer lesen aktivieren" zum Auslesen der Programmnummer
- Folgebefehl nach "Matchstring Lesen aktivieren" zum Auslesen des Matchstrings

Tabelle 3-77 PHYSICAL-READ

CMD	OFFSET BUFFER	LEN_DATA	ADR_TAG	RXREF
0x70	Offset im Sen- depuffer	Länge der vom Codeleser abzuholenden Daten:	0x0000	Vom Codeleser abgeholte Daten:
	"RXREF"	• ≥ Codelänge		Code-Daten
		• = 01		Programmnummer
		• ≥ Matchstring-Länge		Matchstring

3.5.4.4 Wirkung der Befehle

Die verwendeten Befehle wirken wie folgt:

- Die Eingangsparameter "INIT" und "SRESET" unterbrechen die Befehlsausführung innerhalb des Kommunikationsmoduls/Readers.
- Die Fertigmeldung die dem "INIT" oder "SRESET" folgt ("DONE" bzw. "ERROR") bezieht sich immer auf den Eingangsparameter "INIT" oder "SRESET" und nicht auf den unterbrochenen Befehl.
- Der Eingangsparameter "INIT" setzt die Kommunikation zwischen Ident-Profil und dem Kommunikationsmodul zurück. Nach dem "harten" Zurücksetzen des Kommunikationsmoduls überträgt das Ident-Profil automatisch den Befehl "WRITE-CONFIG" zum Kommunikationsmodul. Deswegen ist es zwingend erforderlich, dass Sie im ersten Element des Befehlspuffers "CMDREF" den Befehl "WRITE-CONFIG" hinterlegen.
- Der Befehl "WRITE-CONFIG" setzt alle Funktionen innerhalb des Kommunikationsmoduls zurück, ausgenommen die Kommunikation.
- Der Parameter "SRESET" unterbricht den laufenden Befehl.

3.5.4.5 Befehlsbearbeitung

Gehen Sie wie folgt vor, um die Befehle zu bearbeiten:

- Beschreiben Sie den Parameter "CMDREF" (Array [1...10]) mit den gewünschten Befehlen.
 - Der Inhalt von "CMDREF" = [1] ist für die Initialisierung reserviert. Er wird ausgeführt, wenn der Eingang "INIT" des Ident-Profil gesetzt wird und "CMDSEL" = [1] ist.
- 2. Übertragen Sie die zu schreibenden Daten in den Sendedaten-Puffer "TXBUF".
- 3. Wählen Sie mit dem Parameter "CMDSEL" den zuvor beschriebenen Befehl aus (Array [1...10]).
- 4. Führen Sie mit Hilfe des Parameters "EXECUTE" den Befehl aus ("EXECUTE" = 1").

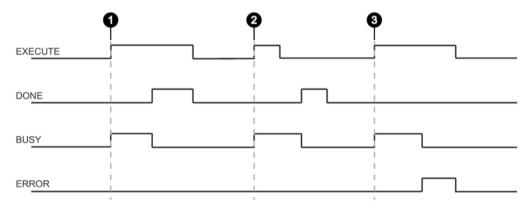
Warten Sie, bis die Bits "BUSY = FALSE" und "DONE = TRUE" gesetzt werden.

Der Befehl ist nun fehlerfrei abgearbeitet.

Wenn "ERROR = TRUE" gesetzt wird, dann fahren Sie mit Punkt 5 fort. Andernfalls fahren Sie mit Punkt 6 fort.

- 5. Werten Sie aufgetretene Fehler aus.
- 6. Setzen Sie das Bit "EXECUTE" zurück.

Das folgende Diagramm zeigt den zeitlichen Ablauf des Ident-Profils. Ein Befehlsstart erfolgt immer mit der positiven Flanke von "EXECUTE", "INIT" oder "SRESET".



- Fall ① Durch Setzen von EXECUTE (EXECUTE = 1) wird die Funktion/Anweisung gestartet. Wurde der Auftrag erfolgreich beendet (DONE = 1) müssen Sie EXECUTE wieder zurücksetzen. DONE wird zeitgleich zurückgesetzt.
- Fall ② EXECUTE wird nur für einen Zyklus gesetzt. Sobald BUSY gesetzt ist (und DONE zurückgesetzt ist) können Sie EXECUTE wieder zurücksetzen. Wurde der Auftrag erfolgreich beendet, wird DONE für einen Zyklus gesetzt.
- Fall ③ Handhabung wie Fall 1, jedoch mit Fehlerausgabe. Sobald ERROR gesetzt ist, steht in der Ausgabe STATUS der genaue Fehlercode. ERROR und STATUS behalten ihren Wert solange bei, wie EXECUTE gesetzt ist.
- Bild 3-36 Allgemeiner Ablauf des Ident-Profils

3.5.4.6 Neustart und Wiederanlauf parametrieren

Ein Neustart des Kommunikationsmoduls und des Readers erfolgt durch das Setzen des Parameters "INIT". Mit dem Parameter werden das CM bzw. der Reader und das Ident-Profil neu parametriert und synchronisiert.

Ein "INIT" ist notwendig nach:

- dem Einschalten oder Wiederanlauf der SIMATIC Steuerung (OB 100 / Startup)
- dem Einschalten der Versorgungsspannung am CM/Reader
- dem Anstecken des Readers am CM
- der Unterbrechung der PROFIBUS/PROFINET-Kommunikation
- einer Fehlermeldung durch den Paramater "STATUS"

3.5.4.7 Verkettung

Mit dem Ident-Profil sowie dem Advanced-Baustein ist es möglich verkettete Befehle zu versenden. Verkettete Befehle werden komplett an den Reader geschickt, ohne dass auf das Ergebnis des ersten Befehls gewartet wird. Diese Funktion ermöglicht es Ihnen verschiedene Transponder-Befehle mit einem Befehlsstart auszuführen.

Bei beiden Bausteinen steht Ihnen ein Befehlspuffer von 10 Befehlen zur Verfügung (Array [1...10] des "IID_CMD_STRUCT"). In jeder Befehlsstruktur befindet sich ein "Chained"-Bit. Dieses Bit muss für jeden verketteten Befehl gesetzt werden. Im letzten verketteten Befehl darf dieses Bit nicht gesetzt werden, damit der Baustein erkennt, dass die Kette beendet ist.

Hinweis

Verkettungsfunktion ist gerätespezifisch

Prüfen Sie bitte, ob das von Ihnen verwendete Ident-Gerät Verkettung unterstützt.

Die Verkettung wird aktuell ausschließlich von den Readern RF680R/RF685R unterstützt (Stand Oktober 2014)

Befehlsübersicht

Tabelle 3-78 Übersicht der Befehle, bei denen eine Verkettung möglich ist

Befehl	Befehlscode		Beschreibung
	HEX	ASCII	
PHYSICAL-READ	70	'p'	Liest Daten von einem Transponder/Code- Lesesystem durch Angabe der physikalischen An- fangsadresse, der Länge und des Passworts aus.
PHYSICAL-WRITE	71	'q'	Schreibt Daten auf einen Transponder/Code- Lesesystem durch Angabe der physikalischen An- fangsadresse, der Länge und des Passworts.
INVENTORY	69	'j'	Fordert eine Liste aller gegenwärtig zugänglichen Transponder innerhalb der Antennenreichweite an.

Befehl	Befehlscode		Beschreibung
	HEX	ASCII	
DEV-STATUS	74	't '	Liest den Status eines Kommunikationsmoduls aus. Dieser Befehl darf nicht der letzte Befehl innerhalb einer Verkettung sein.
WRITE-ID	67	ʻgʻ	RF680R/RF685R:
			Schreibt eine neue EPC-ID in den Transponder.
KILL-TAG	6A	ʻjʻ	RF680R/RF685R:
			Der Transponder wird permanent deaktiviert.
LOCK-TAG-BANK	79	ʻyʻ	RF680R/RF685R:
			Definiert ein Passwort für den Transponder-Zugriff.

Beispiel einer Befehlsstruktur

Tabelle 3-79 Beispiel einer Befehlsstruktur mit 3 Befehlen (ohne EPC-ID)

Befehl	Paramater	Wert	Beschreibung	
Befehl 1	IID_CMD_STRUCT[2].CMD	0x69	Durchführen eines Inventory mit	
	IID_CMD_STRUCT[2].ATTRIBUTES	0x80	der Dauer von 2 Inventories.	
	IID_CMD_STRUCT[2].EXT_UHF.INVENTO RY.DURATION	2		
	IID_CMD_STRUCT[2].EXT_UHF.INVENTO RY.DUR_UNIT	1		
	IID_CMD_STRUCT[2].OPTIONS.CHAINED	true		
Befehl 2	IID_CMD_STRUCT[3].CMD	0x70	Lesen von 10 Byte aus der U-	
	IID_CMD_STRUCT[3].EXT_UHF.MEM_BA NK	3	ser-Bank ab Adresse 0.	
	IID_CMD_STRUCT[3].LEN_DATA	10		
	IID_CMD_STRUCT[3].ADR_TAG	0		
	IID_CMD_STRUCT[3].OPTIONS.CHAINED	true		
Befehl 3	IID_CMD_STRUCT[4].CMD	0x71	Schreiben von 10 Byte in die	
	IID_CMD_STRUCT[4].EXT_UHF.MEM_BA NK	3	User-Bank ab Adresse 20.	
	IID_CMD_STRUCT[4].LEN_DATA	10		
	IID_CMD_STRUCT[4].ADR_TAG	20		
	IID_CMD_STRUCT[4].OPTIONS.CHAINED	false		

Bei der Verkettung kann der komplette "IID_CMD_STRUCT"-Puffer ("IID_CMD_STRUCT[1...10]") genutzt werden. Der Kettenanfang wird über den Parameter "CMDSEL" eingestellt.

Wenn mehrere Befehle in der Kette ausgeführt werden, bei denen Daten zurückgeliefert werden, kann die Position der Daten im Empfangspuffer "RXREF" über den Parameter "IID_CMD_STRUCT[x].OFFSETBUFFER" für jeden einzelnen Befehl eingestellt werden.

Hinweis

"IID_CMD_STRUCT[1]" für "INIT" reserviert

Bei dem Ident-Profil ist der Paramater "IID_CMD_STRUCT[1]" im Regelfall für "INIT" reserviert. Wenn "IID_CMD_STRUCT[1]" für einen anderen Befehl genutzt werden soll, müssen Sie sicherstellen, dass bei einem "INIT" die Reset-Parameter in diesen Parameter geschrieben werden.

3.5.4.8 Befehlswiederholung

Das Ident-Profil unterstützt die Befehlswiederholung (Repeat-Kommando).

Die Befehlswiederholung wird aktuell nicht von den Readern RF680R/RF685R unterstützt (Stand Oktober 2014). Die Funktion ist jedoch in Vorbereitung und wird in der kommenden Version von den Readern unterstützt.

Hinweis

Befehlswiederholungsfunktion ist gerätespezifisch

Prüfen Sie bitte, ob das von Ihnen verwendete Ident-Gerät Befehlswiederholung unterstützt.

Arbeitsweise

Nach Neustart (bzw. "INIT") des Readers überträgt das Ident-Profil einmalig den Befehl bzw. die Befehlskette zum Reader. Die Befehlsübertragung geschieht automatisch mit dem ersten durchgeführten "EXECUTE". Dieser Befehl (bzw. der letzte Befehl oder die Befehlskette) bleibt immer im Reader zwischengespeichert. Wird die Befehlswiederholung gestartet, so wird der zwischengespeicherte Befehl im Reader erneut ausgeführt und das/die Ergebnis/se zum Ident-Profil übertragen.

Stellen Sie sicher, dass die zu wiederholenden Befehle die "EPC-ID/UID" den Wert 0 aufweisen. Hat die EPC-ID einen anderen Wert, wird eine Fehlermeldung erzeugt.

Auswirkungen der Befehlswiederholung

- Der PROFIBUS-/PROFINET-Datentransfer wird minimiert. Diese Minimierung wirkt sich besonders bei großen Bus-Konfigurationen und langsamen Übertragungsraten positiv aus.
- Der Reader bearbeitet unabhängig von dem Ident-Profil jeden Transponder. Das wirkt sich vor allem auf Gate-Applikationen vorteilhaft aus, da alle Transponder immer mit der vollen Reader-Scan-Geschwindigkeit erfasst werden.
- Bei Steuerungen mit nur wenigen Systemressourcen für azyklische Telegramme wird der Gesamtdatendurchsatz erheblich erhöht.

Befehlsübersicht

Tabelle 3-80 Übersicht der Befehle, bei denen eine Befehlswiederholung möglich ist

Befehl	Befehlscode		Beschreibung
	HEX	ASCII	
PHYSICAL-READ	70	'p'	Liest Daten von einem Transponder/Code- Lesesystem durch Angabe der physikalischen An- fangsadresse, der Länge und des Passworts aus.
PHYSICAL-WRITE	71	'q'	Schreibt Daten auf einen Transponder/Code- Lesesystem durch Angabe der physikalischen An- fangsadresse, der Länge und des Passworts.
INVENTORY	69	'i'	Fordert eine Liste aller gegenwärtig zugänglichen Transponder innerhalb der Antennenreichweite an.
KILL-TAG	6A	ʻjʻ	RF680R/RF685R:
			Der Transponder wird permanent deaktiviert.
LOCK-TAG-BANK	79	ʻyʻ	RF680R/RF685R:
			Definiert ein Passwort für den Transponder-Zugriff.

Befehlswiederholung starten

Sie haben die Möglichkeit die Befehlswiederholung mit oder ohne Befehlsübertragung auszuführen. Nachfolgend werden die verschiedenen Vorgehensweisen beschrieben.

Ablauf des Repeat-Kommandos mit gleichzeitiger Befehlsübertragung:

- 1. Starten Sie den Befehl mit Hilfe des Eingangsparameters "EXECUTE" und gleichzeitig gesetztem "RPTCMD". ①
 - Der Befehl wird bearbeitet und das Ergebnis an das Ident-Profil übertragen.
 - Das Repeat-Kommando wird auf dem Reader aktiviert.
- 2. Der Reader bestätigt die Aktivierung über den Ausgangsparameters "RPTACT" am Ident-Profil. Die Bestätigung erfolgt erst, nachdem der erste Befehl abgearbeitet wurde. ②
 - Der Reader führt den Befehl automatisch aus, sobald ein Transponder im Antennenfeld erkannt wird.
 - Wird das Repeat-Kommando vom Reader nicht unterstützt, bleibt "RPTACT" inaktiv. Wird "EXECUTE" trotzdem gesetzt, wird der Fehler "E7FE0900h" nach einem Timeout von 10 Sekunden ausgegeben.
- 3. Die einzelnen Ergebnisse können Sie über wiederholtes Setzen des Eingangsparameters "EXECUTE" auslesen. ③

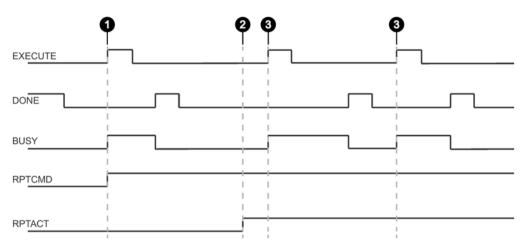


Bild 3-37 Ablauf des Repeat-Kommandos mit gleichzeitiger Befehlsübertragung

Ablauf des Repeat-Kommandos ohne Befehlsübertragung:

Dieser Ablauf ist nur möglich, wenn der betreffende Befehl bereits übertragen wurde.

- Setzen Sie den Eingangsparameter "RPTCMD". ①
 Das Repeat-Kommando wird auf dem Reader aktiviert.
- 2. Der Reader bestätigt die Aktivierung über den Ausgangsparameters "RPTACT" am Ident-Profil. Die Bestätigung erfolgt erst, nachdem der erste Befehl abgearbeitet wurde. ②
 Wird das Repeat-Kommando vom Reader nicht unterstützt, bleibt "RPTACT" inaktiv. Wird "EXECUTE" trotzdem gesetzt, wird der Fehler "E7FE0900h" nach einem Timeout von 10 Sekunden ausgegeben.
- 3. Die einzelnen Ergebnisse können Sie über wiederholtes Setzen des Eingangsparameters "EXECUTE" auslesen. ③

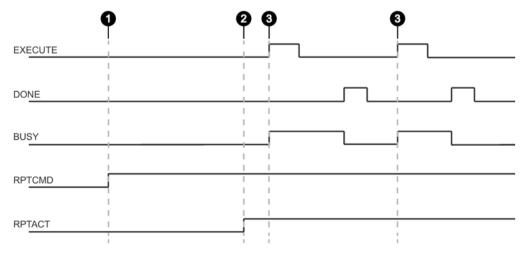


Bild 3-38 Ablauf des Repeat-Kommandos ohne Befehlsübertragung

Befehlswiederholung beenden

Sie haben die Möglichkeit die Befehlswiederholung durch Zurücknahme von "RPTCMD" oder mit Hilfe der Befehle "INIT" oder "SRESET" zu beenden. Nachfolgend werden die verschiedenen Vorgehensweisen beschrieben

Repeat-Kommando beenden und Zurücknahme von "RPTCMD":

- 1. Setzen Sie den Eingangsparameter "RPTCMD" zurück. ①
- Holen Sie evtl. vorhandene Quittungen über den Eingangsparameter "EXECUTE" ab. ②
 Der Ausgangsparameter "RPTACT" bleibt solange vom Reader gesetzt, wie Quittungen vorhanden sind.
- 3. Sind keine Quittungen mehr vorhanden, wird "RPTACT" vom Reader zurückgesetzt. 3

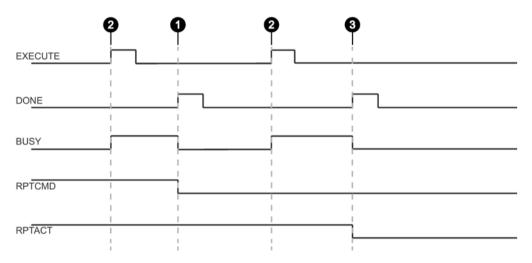


Bild 3-39 Repeat-Kommando beenden durch Zurücknahme von "RPTCMD" (regulär beendet)

Der Ausgangsparameter "RPTACT" wird vom Reader zurückgesetzt. Unter Umständen kann es passieren, dass "RPTACT" verzögert zurückgesetzt wird. D. h. nicht zeitgleich mit dem "DONE" der letzten Quittung. Wenn der Baustein nun erneut mit "EXECUTE" gestartet wird und "RPTACT" noch gesetzt ist, obwohl keine Ergebnisse mehr im Puffer sind, wird der Baustein nicht beendet (BUSY = 1). In diesem Fall können Sie warten bis die nächsten Transponder ausgelesen werden. Alternativ kann der Baustein mit "INIT" oder "SRESET" beendet werden.

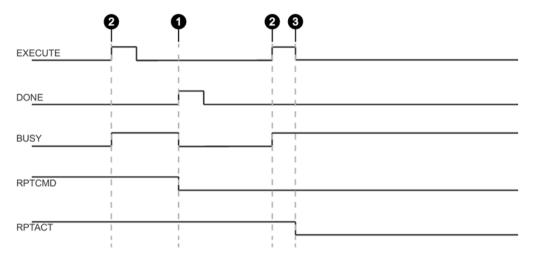


Bild 3-40 Repeat-Kommando beenden durch Zurücknahme von "RPTCMD" (der letzte Befehl bleibt anstehen)

Hinweis

Repeat-Kommando mit "INIT" oder "SRESET" beenden

Beenden Sie das Repeat-Kommando mit Hilfe der Eingangsparameter "INIT" oder "SRESET", wenn nicht bekannt ist, wie viele Transponder nach dem Zurücksetzen des Eingangsparameter "RPTCMD", noch bearbeitet wurden.

In der Regel wird ein "SRESET" deutlich schneller ausgeführt, da keine Reset-Prozedur durchlaufen wird.

Repeat-Kommando durch "INIT" beenden:

- 1. Setzen Sie den Eingangsparameter "RPTCMD" zurück und setzen Sie den Eingangsparameter "INIT". ①
 - Wird "RPTCMD" nicht zurückgesetzt, wird das Repeat-Kommando wieder auf dem Reader aktiviert. Dieses Verhalten löst eine Fehlermeldung aus, da kein Befehl vorhanden ist.
- Der Reader setzt aufgrund des Eingangsparameters "INIT" den Ausgangsparameter "RPTACT" zurück.

Repeat-Kommando durch "SRESET" beenden:

- 1. Setzen Sie den Eingangsparameter "RPTCMD" zurück und setzen Sie den Eingangsparameter "SRESET". ①
- 2. Der Ausgangsparameter "DONE" wird gesetzt und der Reader nimmt den Ausgangsparameter "RPTACT" zurück. ②

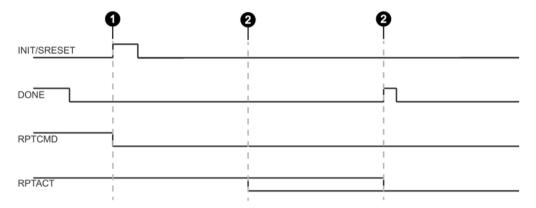


Bild 3-41 Repeat-Kommando durch "INIT"/"SRESET" beenden

Datenpuffer

Eine permanente Befehlswiederholung kann dazu führen, dass die Daten langsamer zum Ident-Profil übertragen werden, als neue Transponder bearbeitet werden. In diesem Fall speichert der Reader die Ergebnisse zwischen. Für diese Zwischenspeicherung steht dem Reader eine Anzahl von Puffern zur Verfügung. Sind die Puffer voll, werden keine neuen Daten von dem Ident-Profil abgeholt, d. h. neu ankommende Transponder werden nicht mehr bearbeitet.

Tabelle 3-81 Reader und Kommunikationsmodule, die die Befehlswiederholung unterstützen

Gerätetyp	Pufferanzahl (Anzahl Befehle)	Max. bearbeitbare Nutzdaten mit Be- fehlswiederholung
RF300 Reader	246	233 Byte × 246 = 57 318 Byte
RF620R/RF630R	150	233 Byte × 150 = 34 950 Byte
RF680R/RF685R	250	1034 Byte × 250 = 258 500 Byte
RF180C	150	233 Byte × 150 = 34 950 Byte
ASM 456	150	233 Byte × 150 = 34 950 Byte

Hinweis

Einschränkung der Befehlswiederholung

Bei RFID-Systemen mit eindeutigen Tag-IDs (UID bzw. EPC-ID) (z. B. RF300, RF600, MOBY U) wird der gespeicherte Befehl nur dann wiederholt, wenn unterschiedliche Transponder in das Antennenfeld gebracht werden. Wenn hingegen immer wieder der gleiche Transponder (gleiche UID / EPC-ID) ins Antennenfeld gebracht wird, so erfolgt keine weitere Bearbeitung des Transponders.

3.6 Transponder-Adressierung

Adressierung

Die Adressierung der Transponder erfolgt linear von Adresse" 0000" (oder der angegebenen Anfangsadresse) bis zur Endadresse. Das CM bzw. der Reader erkennt automatisch die Größe des Speichers auf dem Transponder. Wird die Endadresse auf dem Transponder überschritten, erhalten Sie eine Fehlermeldung.

Die nachfolgende Tabelle zeigt den Adressraum der einzelnen Transponder-Paramater. Die Parameter "ADR_TAG" und "LEN_DATA" müssen gemäß diesem Adressraum parametriert werden.

Adressraum der Transponder-/MDS-Varianten nach ISO 15693 für RF200, RF300 und MOBY D

System	Adressierung	Hexadezimalzahl 16 Bit			
RF200,	MDS D139 (I-Code 1; 44 Byte)				
RF300,	Anfangsadresse	0000			
MOBY D	Endadresse	002B			
	ID-Nr.: (festcodiert; kann nur komplett a	usgelesen werden)			
	Anfangsadresse	FFF0			
	Länge	0008			
	ISO-MDS (I-Code SLI; 112 Byte)				
	Anfangsadresse	0000			
	Endadresse	006F			
	ID-Nr.: (festcodiert; kann nur komplett ausgelesen werden)				
	Anfangsadresse	FFF0			
	Länge	0008			
	ISO-MDS (Tag-it HF-I; 256 Byte)				
	Anfangsadresse	0000			
	Endadresse	00FF			
	ID-Nr.: (festcodiert; kann nur komplett ausgelesen werden)				
	Anfangsadresse	FFF0			
	Länge	8000			

3.6 Transponder-Adressierung

System	Adressierung	Hexadezimalzahl 16 Bit	
	ISO-MDS (my-d SRF55V10P; 992 Byte)		
	Anfangsadresse	0000	
	Endadresse	03DF	
	ID-Nr.: (festcodiert; kann nur komplett ausgelesen werden)		
	Anfangsadresse	FFF0	
	Länge	0008	
	ISO-MDS (MB 89R118B, 2000 Byte)		
	Anfangsadresse	0000	
	Endadresse	07CF	
	ID-Nr.: (festcodiert; kann nur komplett ausgelesen werden)		
	Anfangsadresse	FFF0	
	Länge	0008	

Adressraum der Transponder-Varianten für RF300

System	Adressierung	Hexadezimalzahl 16 Bit		
RF300	20 Byte Datenspeicher (EEPROM)			
	R/W- bzw. OTP-Speicher (EEPROM)			
	(Der EEPROM-Anwenderspeicher bei RF300 kann sowohl als R/W-Speicher, als auch als OTP-Speicher genutzt werden (siehe Systemhandbuch RF300))			
	Anfangsadresse FF00			
	Endadresse	FF13		
	ID-Nr.: (festcodiert, kann nur komplett au	usgegeben werden)		
	Anfangsadresse	FFF0		
	Länge	0008		
	8 KByte Datensp	eicher (FRAM/EEPROM)		
	R/W- bzw. OTP-Speicher (EEPROM)			
	(Der EEPROM-Anwenderspeicher bei Rl als OTP-Speicher genutzt werden (siehe	F300 kann sowohl als R/W-Speicher, als auch Systemhandbuch RF300))		
	Anfangsadresse FF00 Endadresse FF13 R/W-Speicher (FRAM) Anfangsadresse 0000			
	Endadresse	1FFC		
	Id-Nr.: (festcodiert, kann nur komplett ausgelesen werden) Anfangsadresse FFF0			
	Länge	0008		
	32 KByte Datensp	peicher (FRAM/EEPROM)		
	R/W- bzw. OTP-Speicher (EEPROM)			
	(Der EEPROM-Anwenderspeicher bei RF300 kann sowohl als R/W-Speicher, als auch als OTP-Speicher genutzt werden (siehe Systemhandbuch RF300))			
	Anfangsadresse	FF00		
	Endadresse	FF13		

System	Adressierung	Hexadezimalzahl 16 Bit			
	R/W-Speicher (FRAM)				
	Anfangsadresse	0000			
	Endadresse	7FFC			
	ID-Nr.: (festcodiert, kann nur komplett au	usgegeben werden)			
	Anfangsadresse	FFF0			
	Länge	0008			
	64 KByte Datens	peicher (FRAM/EEPROM)			
	R/W- bzw. OTP-Speicher (EEPROM)				
	(Der EEPROM-Anwenderspeicher bei R als OTP-Speicher genutzt werden (siehe	F300 kann sowohl als R/W-Speicher, als auch e Systemhandbuch RF300))			
	Anfangsadresse	FF00			
	Endadresse	FF13			
	R/W-Speicher (FRAM)				
	Anfangsadresse	0000			
	Endadresse	FEFC			
	ID-Nr.: (festcodiert, kann nur komplett ausgegeben werden)				
	Anfangsadresse	FFF0			
	Länge	0008			

RF300: Allgemeine Hinweise zur Bedeutung des OTP-Speichers

RF300-Transponder und ISO-Transponder besitzen einen Speicherbereich, der gegen erneutes Überschreiben geschützt werden kann. Dieser Speicherbereich heißt OTP. Für die Aktivierung der OTP-Funktion sind folgende 5 Blockadressen vorgesehen:

- FF80
- FF84
- FF88
- FF8C
- FF90

Ein Schreibbefehl an diese Blockadresse, mit zulässiger Länge (4, 8, 12, 16, 20 je nach Blockadresse), schützt die geschriebenen Daten vor einem anschließenden Überschreiben.

Hinweis

OTP-Bereich nur im statischen Betrieb nutzen

Nutzen Sie den OTP-Berichs nur im statischen Betrieb.

Hinweis

Nutzung des OTP-Bereichs ist nicht reversibel

Wenn Sie den OTP-Bereich nutzen, können Sie diese Belegung nicht mehr rückgängig machen, da der OTP-Bereich nur einmalig beschrieben werden kann.

RF300: Adressmapping des OTP-Speichers auf dem RF300-Transponder

R/W EEPROM-Speicher und OTP-Speicher ist im Transponder nur einmal vorhanden.

Die folgende Tabelle zeigt das Mapping der Adressen auf dem Transponder.

Das Lesen der Daten kann sowohl über die R/W-Adresse als auch über die OTP-Adresse erfolgen.

R/W EE	PROM	RF300, OTP ein	malig beschreiben
Adresse	Länge	Adresse	Länge
FF00	1 20	FF80	4,8,12,16,20
FF01	1 19		
FF02	1 18		
FF03	1 17		
FF04	1 16	FF84	4,8,12,16
FF05	1 15		
FF06	1 14		
FF07	1 13		
FF08	1 12	FF88	4,8,12
FF09	1 11		
FF0A	1 10		
FF0B	1 9		
FF0C	1 8	FF8C	4,8
FF0D	17		
FF0E	1 6		
FF0F	1 5		
FF10	1 4	FF90	4
FF11	1 3		
FF12	12		
FF13	1		

Hinweis

Schreibschutz aktivieren

Ein Schreiben ab der Adresse FF80 bis FF93 setzt den EEPROM-Anwenderspeicher auf Schreibschutz (OTP-Funktion). Der Vorgang ist nicht reversibel. Die Einschaltung des Schreibschutzes muss immer in lückenlos aufsteigender Reihenfolge erfolgen, beginnend mit Adresse FF80.

Adressraum der Transponder-Varianten für RF600

Tabelle 3-82 Adressräume der Transponder-Varianten für RF620R/RF630R

Tags	Chip-Typ	User 1) [hex]	EPC		TID (read only)	RESERVED specia (Passwörter)		cial
		Bereich / Länge	Bereich / Länge (max. und Default)	Zugriff	Bereich / Länge	Bereich / Län- ge	KILL-PW	Lock- Funktion
RF630L (-2AB00, -2AB01)	Impinj Monza 2	-	FF00-FF0B / 96 Bit FF00-FF0B / 96 Bit	read/ write	FFC0-FFC3 4 Byte	FF80-FF87 8 Byte	Ja	Ja
RF630L (-2AB02)	Impinj Monza 4QT	00 - 3F 64 Byte	FF00-FF0F / 128 Bit FF00-FF0B / 96 Bit	read/ write	FFC0-FFCB 12 Byte	FF80-FF87 8 Byte	Ja	Ja
RF630L (-2AB03)	NXP G2XM	00 - 3F 64 Byte	FF00-FF1D / 240 Bit FF00-FF0B / 96 Bit	read/ write	FFC0-FFC7 8 Byte	FF80-FF87 8 Byte	Ja	Ja
RF640L	Alien Higgs 3	00 - 0F/3F ³⁾ 16/64 Byte	FF00-FF3C / 480 Bit FF00-FF0B / 96 Bit	read/ write	FFC0-FFD8 24 Byte	FF80-FF87 8 Byte	Ja	Ja
RF680L	NXP G2XM	00 - 3F 64 Byte	FF00-FF1D / 240 Bit FF00-FF0B / 96 Bit	read/ write	FFC0-FFC7 8 Byte	FF80-FF87 8 Byte	Ja	Ja
RF690L	Alien Higgs 3	00 - 0F/3F ³⁾ 16/64 Byte	FF00-FF3C / 480 Bit FF00-FF0B / 96 Bit	read/ write	FFC0-FFD8 24 Byte	FF80-FF87 8 Byte	Ja	Ja
RF610T	NXP G2XM	00 - 3F 64 Byte	FF00-FF1D / 240 Bit FF00-FF0B / 96 Bit	read/ write	FFC0-FFC7 8 Byte	FF80-FF87 8 Byte	LOCKED	Ja
RF620T	Impinj Monza 4QT	00 - 3F 64 Byte	FF00-FF0F / 128 Bit FF00-FF0B / 96 Bit	read/ write	FFC0-FFCB 12 Byte	FF80-FF87 8 Byte	LOCKED	Ja
RF625T	Impinj Monza 4QT	00 - 3F 64 Byte	FF00-FF0F / 128 Bit FF00-FF0B / 96 Bit	read/ write	FFC0-FFCB 12 Byte	FF80-FF87 8 Byte	LOCKED	Ja
RF630T	NXP G2XM	00 - 3F 64 Byte	FF00-FF1D / 240 Bit FF00-FF0B / 96 Bit	read/ write	FFC0-FFC7 8 Byte	FF80-FF87 8 Byte	LOCKED	Ja
RF640T	NXP G2XM	00 - 3F 64 Byte	FF00-FF1D / 240 Bit FF00-FF0B / 96 Bit	read/ write	FFC0-FFC7 8 Byte	FF80-FF87 8 Byte	LOCKED	Ja
RF680T	NXP G2XM	00 - 3F 64 Byte	FF00-FF1D / 240 Bit FF00-FF0B / 96 Bit	read/ write	FFC0-FFC7 8 Byte	FF80-FF87 8 Byte	LOCKED	Ja
RF630T	NXP G2XM	00 - 3F 64 Byte	FF00-FF1D / 240 Bit FF00-FF0B / 96 Bit	read/ write	FFC0-FFC7 8 Byte	FF80-FF87 8 Byte	LOCKED	Ja

3.6 Transponder-Adressierung

Tags	Chip-Typ	User 1) [hex]	EPC		TID (read only)	RESERVED (Passwörter)	spe	cial
		Bereich / Länge	Bereich / Länge (max. und Default)	Zugriff	Bereich / Länge	Bereich / Län- ge	KILL-PW	Lock- Funktion
RF640T	NXP G2XM	00 - 3F 64 Byte	FF00-FF1D / 240 Bit FF00-FF0B / 96 Bit	read/ write	FFC0-FFC7 8 Byte	FF80-FF87 8 Byte	LOCKED	Ja
RF680T	NXP G2XM	00 - 3F 64 Byte	FF00-FF1D / 240 Bit FF00-FF0B / 96 Bit	read/ write	FFC0-FFC7 8 Byte	FF80-FF87 8 Byte	LOCKED	Ja

- Der User-Bereich gilt auch für die neuen Reader RF650R/RF680R/RF685R in der MemoryBank 3.
- 2) Verwendet User Memory Indicator (UMI).
- ³⁾ Der EPC-Speicherbereich der Alien Higgs Chips kann zu Lasten des Anwenderspeichers vergrößert werden. Weitere Informationen dazu finden Sie in den entsprechenden Transponder-Kapiteln.

Adressräume der Transponder-Varianten für RF650R/RF680R/RF685R

Bei den neuen Readern RF650R/RF680R/RF685R werden die Anwenderdaten, TID, EPC und Passwörter über die betreffenden Memory Banks ausgelesen. Um die gewünschten Daten auszulesen, muss die betreffende Memory Bank ausgewählt werden.

Bereich und Länge der Anwenderdaten können Sie aus der obrigen Tabelle (Spalte "USER") entnehmen. Die EPC-ID können Sie über einen Inventory-Befehl auslesen. Alternativ können Sie EPC-ID auch über einen Read-Befehl auf die Memory Bank 1, Startadresse 0x04 auslesen.

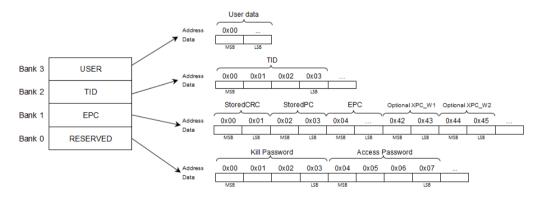


Bild 3-42 Speicheraufbau

Adressraum der Transponder-/MDS-Varianten für MOBY U

System	Adressierung	Hexadezimalzahl 16 Bit			
MOBY U	J 2 KByte Datenspeicher				
	Anfangsadresse	0000			
	Endadresse	07FF			
	OTP-Speicher lesen				
	(Schreiben ist nur einmalig möglich. Der OTP-Speicher von MOBY U kann nur k bearbeitet werden. D. h. die Anfangsadresse muss immer mit dem Wert FFF0 he die Länge mit dem Wert 10 hex angegeben werden.)				
	Anfangsadresse FFF0				
	Länge	10			
	ID-Nr.: (festcodiert 4 Byte; das Auslesen	Byte; das Auslesen ist nur mit dem Befehl MDS-Status möglich)			
	32 KByte Datenspeicher				
	Anfangsadresse	0000			
	Endadresse	7FFF			
	OTP-Speicher lesen (Schreiben ist nur einmalig möglich)*				
	Anfangsadresse	FFF0			
	Länge 10				
	ID-Nr.: (festcodiert 4 Byte; das Auslesen	ist nur mit dem Befehl MDS-Status möglich)			

3.6 Transponder-Adressierung

Fehlermeldungen

4.1 Aufbau des Ausgangsparameters "STATUS"

Ein Fehlerzustand ist in der Funktion Ident-Profil immer dann gegeben, wenn im Ausgangsparameter "ERROR = TRUE" erscheint. Die Analyse (Dekodierung) des Fehlers kann dann über den Ausgangsparameter "STATUS" erfolgen.

Der Ausgangsparameter "STATUS" umfasst folgende 4 Byte:

Tabelle 4-1 Bytes des Ausgangsparameters "STATUS"

Byte	Bedeutung		
	 		
Byte 0	Funktionsnummern		
	Cx - Fehler in der Buskommunikation (Rückwandbus, PROFINET, PROFIBUS)		
	E1 - Transponderbezogene Fehler		
	E2 - Fehler der Luftschnittstelle		
	E4 - Fehler der Reader Hardware		
	E5 - Fehler in der Kommunikation zwischen Reader und FB		
	E6 - Fehler im Befehl des Anwenders		
	E7 - vom FB erzeugte Fehlermeldung		
Byte 1 Fehlernummern			
	Dieses Byte definiert die Bedeutung der Fehlercodes und der Warnungen. Die Fehlernummern haben folgende Bedeutung:		
	0x00 - kein Fehler, keine Warnung		
	0x80 - Fehlermeldung vom Rückwandbus bzw. von PROFIBUS DP-V1 bzw. PROFINET (gemäß IEC 61158-6)		
	0x810x8F - Die Steuerung meldet einen Fehler gemäß dem Parameter "x" (0x8x).		
	0xFE - Fehler vom Ident-Profil bzw. Kommunikationsmodul / Reader		
Byte 2	Fehlercode		
Byte 3	Warnungen		
	In diesem Byte hat jedes Bit eine eigene Bedeutung.		

4.2 STEP 7 - Fehlermeldungen

Sollten Sie die Bausteine und Datentypen in Ihr Projekt eingefügt haben und Probleme bei der Übersetzung haben, prüfen Sie bitte folgende Punkte:

- Der Bausteinname, die Bausteinnummer und der Datentypname dürfen nicht verändert werden.
- Die Datentypen "IID_CMD_STRUCT" und "IID_HW_CONNECT" dürfen in Ihrem Inhalt nicht verändert werden. Erst wenn diese in einen Datenbaustein als Variable eingebunden wurden, dürfen die Inhalte angepasst werden.
- Wenn Sie die Ident-Bausteine nutzen, müssen immer die Datentypen "IID_HW_CONNECT" und "IID_CMD_STRUCT" sowie der Baustein "Ident_Profile_1KB" in Ihrem Projekt vorhanden sein.

4.3 Fehler vom Kommunikationsmodul/Reader

Ursachen für diese Fehler können z. B. sein:

- Die Kommunikation zwischen CM und Reader oder zwischen Reader und Transponder ist fehlerhaft.
- Das Kommunikationsmodul kann den Befehl nicht bearbeiten.

Für die Fehlermeldungen ist Byte 3 des "STATUS" nicht relevant.

Tabelle 4- 2 Fehlermeldungen von Kommunikationsmodul/Reader oder vom Ident-Profil über den Ausgangsparameter STATUS

Fehlermel- dung (hex)	Beschreibung
0xE1FE01	Speicher des Transponders kann nicht beschrieben werden
	Speicher des Transponders ist defekt
	EEPROM-Transponder wurde zu oft beschrieben und hat sein Lebensende erreicht
	RF620R/RF630R: Transponder ist schreibgeschützt (Memory Lock)
0xE1FE02	Anwesenheitsfehler: Der Transponder ist aus dem Übertragungsfenster des Readers gefahren. Der Befehl wurde nur teilweise abgearbeitet.
	Lesebefehl: Es sind keine gültigen Daten bei "IDENT_DATA" vorhanden.
	Schreibbefehl: Der Transponder, der gerade das Antennenfeld verlassen hat, beinhaltet einen unvollständigen Datensatz.
	Arbeitsabstand von Reader zu Transponder wird nicht eingehalten
	Projektierungsfehler: Zu bearbeitender Datensatz ist zu groß (im dynamischen Betrieb)
	mit Timeout: kein Transponder im Antennenfeld

Fehlermel-	Beschreibung
(hex)	
0xE1FE03	Adressfehler
	Der Adressbereich des Transponders wird überschritten.
	Anfangsadresse beim Befehlsstart ist falsch aufgesetzt
	Transponder ist nicht vom richtigen Typ
	Versuch des schreibenden Zugriffs auf schreibgeschützte Bereiche
0xE1FE04	Nur bei Initialisierung: Transponder kann den Initialisierungsbefehl nicht durchführen
	Transponder ist defekt
0xE1FE06	Fehler im Speicher des Transponders
	Der Transponder wurde noch nie beschrieben oder hat durch einen Ausfall der Batterie seinen Speicherinhalt verloren.
	Transponder wechseln (wenn das Batterie-Bit gesetzt ist)
	Transponder neu initialisieren
0xE1FE07	Passwort-Fehler
	RF620R/RF630R: falsches Passwort
0xE1FE08	Der Transponder im Antennenfeld hat nicht die erwartete UID bzw. hat keine UID.
0xE1FE0A	Der Transponder ist lese-/schreibgeschützt.
0xE1FE81	Der Transponder antwortet nicht.
0xE1FE82	Das Transponder-Passwort ist falsch. Zugriff wird verweigert.
0xE1FE83	Die Verifikation der geschriebenen Transponder-Daten ist fehlgeschlagen.
0xE1FE84	Allgemeiner Transponder-Fehler
0xE1FE85	Der Transponder hat zu wenig Leistung, um das Kommando auszuführen.

4.3 Fehler vom Kommunikationsmodul/Reader

Fehlermel- dung	Beschreibung			
(hex)				
0xE2FE01	Feldstörung am Reader			
	 Reader empfängt Störimpulse aus der Umgebung. Externes Störfeld; das Störfeld kann mit dem "induktiven Feldindikator" des STG nachgewiesen werden Der Abstand zwischen zwei Readern ist zu klein und entspricht nicht den Projektierungsrichtlinien Das Verbindungskabel zum Reader wird gestört, ist zu lang oder entspricht nicht der Spezifikation MOBY U: Transponder hat während der Kommunikation das Antennenfeld verlassen. MOBY U: Die Kommunikation zwischen Reader und Transponder wurde durch Störeinflüsse abgebrochen (z. B. Person / Fremdkörper bewegt sich zwischen Reader und Transponder). zu viele Sendefehler 			
	Der Transponder konnte den Befehl oder die Schreibdaten vom Kommunikationsmodul nach mehreren Versuchen nicht richtig empfangen.			
	Transponder steht genau im Grenzbereich des Übertragungsfensters			
	Datenübertragung zum Transponder wird durch externe Störungen beeinflusst			
	CRC-Sendefehler			
	 Der Transponder meldet sehr oft CRC-Fehler (Transponder steht im Grenzbereich des Readers; Transponder und/oder Reader haben einen Hardwaredefekt) 			
	nur bei Initialisierung: CRC-Fehler beim Quittungsempfang vom Transponder (Ursache wie bei Feldstörung am Reader)			
	Bei der Formatierung muss der Transponder im Übertragungsfenster des Readers stehen, ansonsten erfolgt ein Timeout-Fehler, d. h.:			
	Der Transponder steht genau im Grenzbereich des Übertragungsfensters			
	Der Transponder verbraucht zuviel Strom (defekt)			
	EEPROM-Transponder bei FORMAT falsch parametriert			
	• RF600:			
	- Kein ETSI-Kanal frei			
	Falscher Kommunikationsstandard im "INIT"-Befehl ausgewählt - Falscher Kommunikationsst			
	– Expertenparameter fehlerhaft– Leistungsprüfung des ETSI-Funkprofils fehlerhaft			
0xE2FE02				
UXEZFEUZ	Es sind mehr Transponder im Übertragungsfenster, als der Reader gleichzeitig bearbeiten kann. DE620D/DE620D/Leistungsverserzung des Transponders ist im Cransborsieh.			
	RF620R/RF630R: Leistungsversorgung des Transponders ist im Grenzbereich.			
	Erhöhen Sie die Antennenleistung oder verringern Sie den Abstand zu dem Transponder.			
0xE2FE81	Kein Transponder mit der gewünschten EPC-ID befindet sich im Übertragungsfenster, bzw. es befindet sich gar kein Transponder im Antennenfeld.			
0xE2FE82	Die angeforderten Daten sind nicht verfügbar.			
0xE2FE83	Der Transponder meldet einen CRC-Fehler.			
0xE2FE84	Die ausgewählte Antenne ist nicht aktiviert.			
0xE2FE85	Die ausgewählte Frequenz ist nicht aktiviert.			
0xE2FE86	Das Trägersignal ist nicht aktiviert.			
0xE2FE87	Mehr als ein Transponder befinden sich im Übertragungsfenster.			
0xE2FE88	Allgemeiner Funkprotokoll-Fehler			

Fehlermel- dung (hex)	Beschreibung		
0xE4FE01	 Kurzschluss oder Überlastung der 24 V-Ausgänge Der Reader verbraucht zu viel Strom. Das Readerkabel verursacht einen Kurzschluss. Mögliche Folgen: Der betroffene Ausgang wird abgeschaltet Bei Gesamtüberlastung werden alle Ausgänge abgeschaltet Ein Rücksetzen ist nur durch das Aus- und Wiedereinschalten der 24 V-Versorgungsspannung möglich 		
0xE4FE03	 anschließend "Reset_Reader" starten Fehler in der Verbindung zum Reader; Reader antwortet nicht. 		
	 Kabel zwischen Kommunikationsmodul und Reader ist falsch verdrahtet oder Kabelbruch 24 V-Versorgungsspannung ist nicht angeschlossen oder abgeschaltet bzw. kurzzeitig ausgefallen Automatische Sicherung auf dem Kommunikationsmodul hat angesprochen Hardware defekt 		
	- Anderer Reader ist in der Nähe und ist aktiv geschaltet - nach der Fehlerbehebung "init_run" durchführen Die Anteren von Besche ist ab geschaltet in diesen Zustand ununde ein Tee Befahl zum Konsenwilletig		
	 Die Antenne am Reader ist abgeschaltet. In diesem Zustand wurde ein Tag-Befehl zum Kommunikationsmodul gestartet. Mit dem Befehl "Antenne Ein/Aus" die Antenne einschalten Die Antenne ist eingeschaltet (ausgeschaltet) und hat einen weiteren Einschaltbefehl (Ausschaltbefehl) erhalten 		
	Der Modus im Befehl" SET_ANT" ist unbekannt		
	Die Antenne am Reader ist abgeschaltet oder das Antennenkabel ist defekt		
0xE4FE04	Der Puffer im Kommunikationsmodul oder Reader zur Zwischenspeicherung des Befehls reicht nicht aus.		
0xE4FE05	Der Puffer im Kommunikationsmodul oder Reader zur Zwischenspeicherung der Daten reicht nicht aus.		
0xE4FE06	Das Kommando ist in diesem Status nicht erlaubt, bzw. wird nicht unterstützt.		
0xE4FE07	Hochlaufmeldung vom Reader/Kommunikationsmodul. Der Reader bzw. das Kommunikationsmodul war abgeschaltet und hat noch keinen Befehl "Reset_Reader" ("WRITE_CONFIG") erhalten. • "INIT" durchführen		
	Die gleiche physikalische Adresse im Parameter "IID_HW_CONNECT" wird mehrmals verwendet. Überprüfen Sie ihre "IID_HW_CONNECT"-Parametrierungen. Adverse im Parametrierungen.		
	Verbindung zum Reader überprüfen Paudeste wurde umgeschelten, aber nach kein Spannung AUS/FIN durchgeführt.		
0.545504	Baudrate wurde umgeschalten, aber noch kein Spannung AUS/EIN durchgeführt		
0xE4FE81	Reserviert		
0xE4FE8A	Allgemeiner Fehler		
0xE4FE8B	Es wurden keine oder fehlerhafte Konfigurationsdaten übertragen.		

4.3 Fehler vom Kommunikationsmodul/Reader

Fehlermel-	Beschreibung
dung	
(hex)	
0xE4FE8C	 Kommunikationsfehler zwischen Ident-Profil und Kommunikationsmodul. Handshake-Fehler. UDT dieses Kommunikationsmoduls wird durch andere Programmteile überschrieben Parametrierung des Kommunikationsmoduls im UDT überprüfen Befehl des Ident-Profils überprüfen, der zu diesem Fehler führt Nach Fehlerbehebung "INIT" starten Rückwandbus- / PROFIBUS DP- / PROFINET-Fehler aufgetreten Dieser Fehler wird nur angezeigt, wenn die "Ansprechüberwachung" bei der PROFIBUS-Konfigurierung eingeschaltet wurde. Rückwandbus/PROFIBUS DP / PROFINET-Busverbindung war unterbrochen (Drahtbruch am Bus; Busstecker am Kommunikationsmodul war kurzzeitig gezogen) Rückwandbus/PROFIBUS DP / PROFINET-Master spricht Kommunikationsmodul nicht mehr an "INIT" durchführen Das Kommunikationsmodul hat am Bus eine Telegrammunterbrechung festgestellt. Eventuell wurde der Rückwandbus, PROFIBUS bzw. PROFINET neu konfiguriert (z. B. mit HW-Konfig oder TIA Portal)
0xE4FE8D	 Interner Kommunikationsfehler des Kommunikationsmoduls/Reader Stecker-Kontaktproblem auf dem Kommunikationsmodul/Reader Hardware des Kommunikationsmoduls/Reader hat einen Defekt; → Kommunikationsmodu/Readerl zur Reparatur einschicken Nach Fehlerbehebung "INIT" starten Interner Überwachungsfehler des Kommunikationsmoduls/Reader Programmablauffehler auf dem Kommunikationsmodul/Reader Versorgungsspannung des Kommunikationsmoduls/Reader aus- und wiedereinschalten Nach Fehlerbehebung "INIT" starten MOBY U: Watchdog-Fehler auf dem Reader
0xE4FE8E	Laufender Befehl durch "WRITE-CONFIG" ("INIT" oder "SRESET") abgebrochen bzw. der Busstecker wurde abgezogen Die Kommunikation mit dem Transponder wurde mit "INIT" abgebrochen Dieser Fehler kann nur bei einem "INIT" oder "SRESET" zurückgemeldet werden
0xE5FE01	Falsche Sequenz-Nummernfolge (SN) im Reader/Kommunikationsmodul
0xE5FE02	Falsche Sequenz-Nummernfolge (SN) im Ident-Profil Mögliche Ursache: In der Gerätekonfigurationist nicht der User-Mode "RFID-Normprofil" eingestellt.
0xE5FE04	Ungültige Datenblock-Nummer (DBN) im Reader/Kommunikationsmodul
0xE5FE05	Ungültige Datenblock-Nummer (DBN) im Ident-Profil
0xE5FE06	Ungültige Datenblock-Länge (DBL) im Reader/Kommunikationsmodul
0xE5FE07	Ungültige Datenblock-Länge (DBL) im Ident-Profil

Fehlermel-	Beschreibung
dung (box)	
(hex) 0xF5FF08	Vorheriger Befehl ist aktiv bzw. Pufferüberlauf
OXEST EGG	An den Reader bzw. das Kommunikationsmodul wurde ein neuer Befehl geschickt, obwohl der letzte Befehl noch aktiv war.
	Aktiver Befehl kann nur durch "INIT" abgebrochen werden
	Vor dem Start eines neuen Befehls muss das "DONE-Bit = 1" sein; Ausnahme "INIT"
	Zwei Ident-Profil-Aufrufe wurden mit den gleichen Parametern "HW_ID", "CM_CHANNEL" und "LADDR" parametriert
	Zwei Ident-Profil-Aufrufe arbeiten mit dem gleichen Zeiger
	Nach Fehlerbehebung "INIT" starten
	Beim Arbeiten mit Befehlswiederholung (z. B. Festcode-Transponder) werden keine Daten vom Transponder abgeholt. Der Datenpuffer im Reader/Kommunikationsmodul ist übergelaufen. Es sind Transponder-Daten verloren gegangen.
0xE5FE09	Der Reader bzw. das Kommunikationsmodul führt einen Hardware-Reset aus ("INIT_ACTIVE" auf "1" gesetzt), vom Ident-Profil wird "INIT" erwartet (Bit 15 im zyklischen Steuerwort).
0xE5FE0A	Der Befehlscode "CMD" und die entsprechende Bestätigung stimmen nicht überein. Hierbei kann es sich um einen Software- oder Synchronisationsfehler handeln, der im Normalbetrieb nicht auftreten kann.
0xE5FE0B	Falsche Reihenfolge der Quittungstelegramme (TDB / DBN)
0xE5FE0C	Synchronisationsfehler (falsches Inkrement von AC_H / AC_L und CC_H / CC_L im zyklischen Steuerwort), "INIT" musste ausgeführt werden
0xE6FE01	Unbekannter Befehl
	Ident-Profil gibt einen nicht interpretierbaren Befehl an das Kommunikationsmodul.
	Der Baustein "AdvancedCmd" wurde mit einem falschen "CMD" versorgt.
	Der Eingang "CMD" des Bausteins "AdvancedCmd" wurde vom Anwender überschrieben.
	Der Transponder hat einen Adressfehler gemeldet.
0xE6FE02	Ungültiger Kommandoindex CI

4.3 Fehler vom Kommunikationsmodul/Reader

Fehlermel-	Beschreibung			
dung	beschiebung			
(hex)				
0xE6FE03	 Fehlerhafte Parametrierung des Kommunikationsmoduls bzw. Readers "INPUT"-Parameter im Ident-Profil überprüfen. Parametrierung in HW-Konfig / TIA Portal überprüfen. Befehl "WRITE-CONFIG" ist falsch parametriert. Nach einem Hochlauf hat der Reader bzw. das Kommunikationsmodul noch keinen "INIT" erhalten. Der Reader bzw. das Kommunikationsmodul am PROFIBUS/PROFINET wurde falsch parametriert und der Befehl kann nicht abgearbeitet werden. Länge der Ein-/Ausgangsbereiche zu klein für das zyklische Wort E/A. Richtige GSD-Datei verwendet? Befehl (z. B. "READ") mit zu großer Länge der Nutzdaten aufgesetzt. Fehler beim Bearbeiten des Befehls Die Daten im "AdvancedCmd" bzw. "IID_CMD_STRUCT" sind fehlerhaft (z. B. "WRITE"-Befehl mit Länge = 0); "AdvancedCmd" bzw. "IID_CMD_STRUCT" überprüfen und "INIT" durchführen. Hardware des Reader/Kommunikationsmoduls defekt: Bei "INIT" erhält der Reader bzw. das Kommunikationsmodul falsche Daten. AB-Byte stimmt nicht mit der Nutzdatenlänge überein. Falscher Reset-Baustein wurde ausgewählt 			
	 Verwenden Sie, unabhängig vom gewählten Reader-System, den Funktionsbaustein "Reset_Reader". 			
0xE6FE04	Anwesenheitsfehler:			
UXEOFEU4	Ein Transponder ist an einem Reader vorbeigefahren und wurde mit keinem Befehl bearbeitet.			
	 Diese Fehlermeldung wird nicht sofort gemeldet. Vielmehr wartet der Reader bzw. das Kommunikationsmodul auf den n\u00e4chsten Befehl (Lesen, Schreiben). Dieser Befehl wird sofort mit diesem Fehler beantwortet. Dies bedeutet, dass ein Lese- oder Schreibbefehl nicht bearbeitet wird. Erst der n\u00e4chste Befehl wird wieder normal vom Reader/Kommunikationsmodul ausgef\u00fchrt. 			
	Ein "INIT" von Ident-Profil setzt diesen Fehlerzustand ebenfalls zurück.			
	Im Parameter OPT1 ist das Bit 2 gesetzt und es befindet sich kein Transponder im Übertragungsfenster.			
0xE6FE05	Es ist ein Fehler aufgetreten, der ein Reset_Reader ("WRITE-CONFIG" mit "Config = 3") erforderlich macht. • Der Befehl "WRITE-CONFIG" ist fehlerhaft. • Nach Fehlerbehebung "INIT" starten • Überprüfen Sie den Parameter "IID_HW_CONNECT".			
0xE6FE06	Der Reset-Timer ist abgelaufen.			
0xE6FE81	Reserviert			
0xE6FE82	Reserviert			
0xE6FE83	Reserviert			
0xE6FE84	Reserviert			
0xE6FE85	Reserviert			
0xE6FE86	Der Inventory-Befehl ist fehlgeschlagen.			
0xE6FE87	Der Lesezugriff auf den Transponder ist fehlgeschlagen.			

Fehlermel- dung (hex)	Beschreibung		
0xE6FE88	Der Schreibzugriff auf den Transponder ist fehlgeschlagen.		
0xE6FE89	Das Schreiben der EPC-ID auf dem Transponder ist fehlgeschlagen.		
0xE6FE8A	Das Aktivieren des Schreibschutzes auf dem Transponder ist fehlgeschlagen.		
0xE6FE8B	Das "Kill"-Kommando ist fehlgeschlagen.		
0xE7FE01	In diesem Zustand ist nur der Befehl "Reset_Reader" ("WRITE-CONFIG") zulässig.		
0xE7FE02	Der Befehlscode "CMD" ist nicht zulässig.		
0xE7FE03	Der Parameter "LEN_DATA" des Befehls ist zu lang.		
	Er passt nicht zu den globalen Daten, die innerhalb des Sendedaten-Puffers (TXBUF) reserviert wurden.		
0xE7FE04	Der Empfangsdaten-Puffer (RXBUF) oder der Sendedaten-Puffer (TXBUF) ist zu klein, der angelegte Puffer an TXBUF/RXBUF hat nicht den richtigen Datentypen oder der Parameter "LEN_DATA" hat einen negativen Wert.		
	Mögliche Ursache / weiteres Vorgehen:		
	Überprüfen Sie ob die Puffer TXBUF/RXBUF mindestens so groß sind wie bei LEN_DATA angegeben.		
	Bei S7-1200/1500:		
	Am Ident_Profil darf nur ein "Array of Byte" an TXBUF und RXBUF angelegt werden.		
	 An den Bausteinen "Tag_Status" und "Reader_Status" dürfen nur ein "Array of Byte" oder die da- zugehörigen Datentypen angelegt werden ("IID_TAG_STATUS_XX_XXX" oder "IID_READER_STATUS_XX_XXX") 		
0xE7FE05	Dieser Fehler informiert Sie, dass als nächster Befehl nur ein "INIT"-Befehl zulässig ist. Alle anderen Befehle werden zurückgewiesen.		
0xE7FE06	Falscher Index (außerhalb des Bereichs von "101 108" und "-2040120418")		
0xE7FE07	Der Reader bzw. das Kommunikationsmodul antwortet nicht auf "INIT" (in zyklischer Statusmeldung wird "INIT_ACTIVE" erwartet).		
	Weiteres Vorgehen:		
	Überprüfen Sie den Adress-Parameter "LADDR".		
0xE7FE08	Zeitüberschreitung während "INIT"		
	(60 Sekunden gemäß "TC3WG9")		
0xE7FE09	Befehlswiederholung wird nicht unterstützt.		
0xE7FE0A	Fehler während der Übertragung der PDU (Protocol Data Unit).		
0xFxFExx	Ein Fehler "FxFExxh" ist identisch mit dem entsprechenden Fehler "ExFExxh" (siehe Fehler "ExFExxh"). Zusätzlich stehen im Byte 3 Warnhinweise.		

4.4 Fehler vom Rückwandbus

Die Transportschicht vom verwendeten Bussystem (Rückwandbus, PROFIBUS, PROFINET) meldet einen Fehler. Zur genauen Fehlersuche und -analyse ist ein PROFIBUS-Tracer eine wertvolle Hilfe. Für PROFINET kann die open Source-Software "Wireshark" verwendet werden. Die Systemdiagnose des PROFIBUS bzw. PROFINET kann weiteren Aufschluss über die Fehlerursache liefern.

Tabelle 4-3 Fehlermeldungen vom Rückwandbus über den Ausgangsparameter "STATUS"

Fehlermel- dung (hex)	Beschreibung		
Cx800A	Kommunikationsmodul ist nicht bereit (temporäre Meldung)		
	Diese Meldung erhält ein Anwender, der nicht mit dem Ident-Profil arbeitet und sehr schnell nacheinander das Kommunikationsmodul azyklisch abfragt.		
Cx8x7F	Interner Fehler am Parameter "x". Kann vom Anwender nicht behoben werden.		
Cx8x22	Bereichslängenfehler beim Lesen eines Parameters.		
	Dieser Fehlercode zeigt an, dass sich der Parameter "x" vollständig oder teilweise außerhalb des Operandenbereichs befindet oder die Länge eines Bitfeldes bei einem "ANY"-Parameter nicht durch 8 teilbar ist.		
Cx8x23	Bereichslängenfehler beim Schreiben eines Parameters.		
	Dieser Fehlercode zeigt an, dass sich der Parameter "x" vollständig oder teilweise außerhalb des Operandenbereichs befindet oder die Länge eines Bitfeldes bei einem "ANY"-Parameter nicht durch 8 teilbar ist.		
Cx8x24	Bereichsfehler beim Lesen eines Parameters.		
	Dieser Fehlercode zeigt an, dass sich der Parameter "x" in einem Bereich befindet, der für die System- funktion unzulässig ist.		
Cx8x25	Bereichsfehler beim Schreiben eines Parameters.		
	Dieser Fehlercode zeigt an, dass sich der Parameter "x" in einem Bereich befindet, der für die System- funktion unzulässig ist.		
Cx8x26	Der Parameter enthält eine zu große Nummer einer Zeitzelle.		
Cx8x27	Der Parameter enthält eine zu große Nummer einer Zählerzelle.		
Cx8x28	Ausrichtungsfehler beim Lesen eines Parameters.		
	Der Verweis auf den Parameter "x" ist ein Operand, dessen Bitadresse ≠ 0 ist.		
Cx8x29	Ausrichtungsfehler beim Schreiben eines Parameters.		
	Der Verweis auf den Parameter "x" ist ein Operand, dessen Bitadresse ≠ 0 ist.		
Cx8x30	Der Parameter befindet sich in dem schreibgeschützten Global-DB.		
Cx8x31	Der Parameter befindet sich in dem schreibgeschützten Instanz-DB.		
Cx8x32	Der Parameter enthält eine zu große DB-Nummer.		
Cx8x34	Der Parameter enthält eine zu große FC-Nummer.		
Cx8x35	Der Parameter enthält eine zu große FB-Nummer.		
Cx8x3A	Der Parameter enthält die Nummer eines DB, der nicht geladen ist.		
Cx8x3C	Der Parameter enthält die Nummer einer FC, die nicht geladen ist.		
Cx8x3E	Der Parameter enthält die Nummer eines FB, der nicht geladen ist.		
Cx8x42	Es ist ein Zugriffsfehler aufgetreten, während das System einen Parameter aus dem Peripheriebereich der Eingänge auslesen wollte.		
Cx8x43	Es ist ein Zugriffsfehler aufgetreten, während das System einen Parameter in den Peripheriebereich der Ausgänge schreiben wollte.		

Fehlermel- dung	Beschreibung		
(hex)			
Cx8x44	Fehler beim n-ten (n > 1) Lesezugriff nach Auftreten eines Fehlers.		
Cx8x45	Fehler beim n-ten (n > 1) Schreibzugriff nach Auftreten eines Fehlers.		
Cx8090	Angegebene logische Basisadresse ungültig. Bitte überprüfen Sie den Parameter "HW_ID" in "IID_HW_CONNECT".		
Cx8092	In "ANY"-Referenz ist eine Typangabe ≠ BYTE angegeben.		
Cx8093	Die mittels "ID" bzw. "F_ID" adressierte DP-Komponente ist nicht konfiguriert. Bitte überprüfen Sie die "HW-ID".		
Cx80A0	Negative Quittung beim Lesen von Baugruppe; Ident-Profil holt Quittung, obwohl keine zum Abholen bereit ist.		
	Ein Anwender, der nicht mit dem Ident-Profil arbeitet, möchte den DS 101 (oder DS 102 bis DS 104) abholen, es steht aber noch keine Quittung zur Verfügung.		
	zur Neusynchronisation zwischen Kommunikationsmodul und Anwendung einen "init_run" durchführen		
Cx80A1	Negative Quittung beim Schreiben zur Baugruppe; Ident-Profil schickt Befehl, obwohl ein Kommunikati- onsmodul keinen Befehl entgegen nehmen kann		
Cx80A2	DP-Protokollfehler bei Layer 2, evtl. Hardware defekt		
Cx80A3	DP-Protokollfehler bei Direct-Data-Link-Mapper oder User-Interface / User, evtl. Hardware defekt		
Cx80B0	Bitte überprüfen Sie den Adress-Parameter "HW_ID" in der "IID_HW_CONNECT"-Variablen.		
	Baugruppe kennt den Datensatz nicht.		
	Datensatznummer ≥ 241 ist unzulässig.		
Cx80B1	Die Längenangabe in Parameter "RECORD" ist falsch.		
Cx80B2	Der projektierte Steckplatz ist nicht belegt. Bitte überprüfen Sie die GSDML-Version.		
Cx80B3	Ist-Baugruppentyp ungleich Soll-Baugruppentyp im SDB1		
Cx80C0	RDREC:		
	Die Baugruppe führt den Datensatz, aber es sind noch keine Lesedaten da. • WRREC:		
	Das Kommunikationsmodul ist nicht bereit, neue Daten entgegenzunehmen.		
	Warten Sie auf das Hochzählen des zyklischen Zählers.		
Cx80C1	Die Daten des auf der Baugruppe vorangegangenen Schreibauftrags für denselben Datensatz sind von der Baugruppe noch nicht verarbeitet.		
Cx80C2	Die Baugruppe bearbeitet momentan das mögliche Maximum an Aufträgen für eine CPU.		
Cx80C3	Benötigte Betriebsmittel (Speicher etc.) sind momentan belegt.		
	Dieser Fehler wird vom Ident-Profil nicht gemeldet. Im Falle dieses Fehlers wartet Ident-Profil, bis wieder Betriebsmittel vom System zur Verfügung gestellt werden.		
Cx80C4	Kommunikationsfehler		
	Parity-Fehler		
	SW-Ready nicht gesetzt		
	Fehler in der Blocklängenmitführung		
	Prüfsummenfehler auf CPU-Seite		
	Prüfsummenfehler auf Baugruppenseite		

4.5 Warnungen

Fehlermel- dung (hex)	Beschreibung
Cx80C5	Dezentrale Peripherie nicht verfügbar.
Dx8xxx	Ein Fehler "Dx8xxxh" ist identisch mit dem entsprechenden Fehler "Cx8xxxh" (siehe Fehler "Cx8xxxh"). Zusätzlich stehen im Byte 3 Warnhinweise.

4.5 Warnungen

Das Byte 3 des Ausgangsparameters "STATUS" zeigt Warnungen an, wenn das Byte 0 des "STATUS" (Funktionsnummern) den Wert "Fxh" bzw. "Dxh" aufweist.

Tabelle 4-4 Mögliche Warnungen beim Arbeiten mit dem Ident-Profil

Byte 02	Byte 3	Bedeutung	
FxFExxh	Bit 0	Das Bit ist fest auf "0" gesetzt.	
	Bit 1	Herstellerabhängig	
	Bit 2	Batterie niedrig	
	Bit 3	Herstellerabhängig	
	Bit 4	Herstellerabhängig	
	Bit 5	Herstellerabhängig	
	Bit 6	Herstellerabhängig	
	Bit 7	Herstellerabhängig	

Anhang

A.1 Versteckte Statusparameter

Status-Variablen

Jeder Ident-Baustein verfügt über Statusausgänge, um im Fehlerfall im Anwenderprogramm entsprechend zu reagieren und um eine Fehlerdiagnose am Gerät zu vereinfachen. Zusätzlich dazu beinhaltet jeder Ident-Baustein einen Zeitstempel und einen Fehlerspeicher, um vergangene Probleme besser nachzuvollziehen.

Diese Variablen sind gespeichert im jeweiligen Instanz-DB des Bausteins.

Tabelle A- 1 Status-Variablen im Instanz-Datenbaustein

Name	Datentyp	Pfad	Beschreibung
Last_error_ status	DWORD	Instanz-Datenbaustein/ Ident_Instance/Static/ Last_error_status	Diese Variable beinhaltet den letzten Anweisungsstatus im Fehlerfall. Dieser Wert wird immer überschrieben, wenn ein neuer Fehler am Baustein auftritt.
Last_error_ timestamp	DTL (S7-1200/- 1500) DATE_AND_TIM E S7-300/-400)	Instanz-Datenbaustein/ Ident_Instance/Static/ Last_error_timestamp	Diese Variable speichert den Zeitstempel des letzten aufge- tretenen Fehlers (Last_error_status) an der An- weisung.

A.1 Versteckte Statusparameter

Weitere Status-Variablen sind in der Variablen "IID_HW_CONNECT" vorhanden.

Tabelle A- 2 Status-Variablen in "IID_HW_CONNECT"

Name	Datentyp	Pfad	Beschreibung
STATUS_IN_ WORK	BOOL	IID_HW_CONNECT-Variable/ Static/STATUS_IN_WORK	Befehl wird aktuell ausgeführt True = Ein Baustein bzw. das Ident-Profil greift auf diesen Kanal/Reader zu. False = Der Kanal/Reader wird aktuell nicht verwendet.
STATUS_ INITIALISATION	BOOL	IID_HW_CONNECT-Variable/ Static/ STATUS_INITIALISATION	Reset-Anzeige True = An diesem Reader/Kanal ist ein Reset aktiv. False = An diesem Reader/Kanal ist kein Reset aktiv.
LAST_CMD_ INIT	BOOL	IID_HW_CONNECT-Variable/ Static/ STATUS_LAST_CMD_INIT	Dieses Bit zeigt an, dass der zuletzt ausgeführte Befehl ein Reset war. True = Letzter Befehl war ein Reset False = Letzter Befehl war kein Reset Dieses Bit wird beim nächsten Befehlsstart zurückgesetzt

A.2 Service & Support

Technische Unterstützung

Sie erreichen den Technical Support für alle IA-/DT-Produkte über folgende Kommunikationswege:

- Telefon: +49 (0) 911 895 7222
- Fax: +49 (0) 911 895 7223
- E-Mail (mailto:support.automation@siemens.com)
- Internet: Web-Formular f
 ür Support Request (http://www.siemens.de/automation/support-request)

Ansprechpartner

Falls Sie noch Fragen zur Nutzung unserer Produkte haben, wenden Sie sich bitte an Ihren Siemens-Ansprechpartner in den für Sie zuständigen Vertretungen und Geschäftsstellen.

Die Adressen finden Sie an folgenden Stellen:

- Im Internet (http://www.siemens.de/automation/partner)
- Im Katalog CA 01
- Im Katalog ID 10 speziell f
 ür Industrial Identification Systems

Service & Support bei Industry Automation and Drive Technologies

Im Internet finden Sie auf der Support-Homepage (http://www.siemens.com/automation/service&support) von IA/DT verschiedene Service-Leistungen.

Dort finden Sie z. B. folgende Informationen:

- Den Newsletter, der Sie ständig mit den aktuellsten Informationen zu Ihren Produkten versorgt.
- Die für Sie richtigen Dokumente über unsere Suche in "Produkt Support".
- Ein Forum, in welchem Anwender und Spezialisten weltweit Erfahrungen austauschen.
- Ihren Ansprechpartner f
 ür IA/DT vor Ort.
- Informationen über Vor-Ort-Service, Reparaturen, Ersatzteile. Vieles mehr steht für Sie unter "Unser Service-Angebot" bereit.

RFID-Homepage

Allgemeine Neuigkeiten zu unseren Identifikationssystemen finden Sie im Internet auf unserer RFID-Homepage (http://www.siemens.de/ident/rfid).

A.2 Service & Support

Online-Katalog und -Bestellsystem

Den Online-Katalog und das Online-Bestellsystem finden Sie ebenfalls auf der Industry Mall-Homepage (http://www.siemens.com/industrymall/de).

Trainingscenter

Um Ihnen den Einstieg zu erleichtern, bieten wir Ihnen entsprechende Kurse an. Wenden Sie sich bitte an Ihr regionales Trainingscenter oder an das zentrale Trainingscenter in

D-90327 Nürnberg.

Telefon: +49 (0) 180 523 56 11

(0,14 €/Min. aus dem deutschen Festnetz, abweichende Mobilfunkpreise möglich)

Informationen zum Kursangebot finden Sie auch auf der SITRAIN-Homepage (http://www.sitrain.com).