

FND
in öffentlichen Gebäuden
Spezifikation
(FND 2009)

FND
in öffentlichen Gebäuden
(FND 2009)

Lfd. Nr. 104
Aufgestellt und herausgegeben vom
Arbeitskreis Maschinen- und Elektrotechnik
staatlicher und kommunaler Verwaltungen
(AMEV)
Berlin 2009

Geschäftsstelle des AMEV im
Bundesministerium für Verkehr, Bau und Stadtentwicklung (BMVBS)
Ref. B12
Krausenstraße 17-20, 10117 Berlin,
Telefon (030) 2008-7321,
E-Mail: AMEV@BMVBS.Bund.de

Dieses Dokument enthält Links zu externen Webseiten mit begleitenden und ergänzenden Informationen. Der AMEV hat keinen Einfluss auf die Inhalte dieser Webseiten und übernimmt dafür keine Verantwortung.

Vorwort	5
1 Geltungsbereich und Anwendungshinweise	7
2 Einleitung	8
2.1 Grundidee des FND	8
2.2 Datenpunkte	10
3 FND-Detailfestlegungen.....	11
3.1 Definitionen und Bezeichnungsweisen	11
3.2 Kommunikationsmodell	12
3.3 Beschreibung der Datenpunkte.....	15
3.3.1 Meldepunkt.....	16
3.3.2 Schaltpunkt (mit Rückmeldung)	22
3.3.3 Messpunkt.....	28
3.3.4 Stellpunkt	36
3.3.5 Zählpunkt	40
3.3.6 Karteipunkt.....	47
3.4 FND – APDU-Format	70
3.4.1 Operationsbeschreibung	75
3.4.2 Befehle	75
3.4.2.1 Datenteil	76
3.4.2.2 Modifikations-Kontrolle.....	76
3.4.2.3 Meldungen	78
3.4.2.4 Datenteil	78
3.4.2.5 Ereignis-Anzeige	78
3.5 Beschreibung des DP-Status.....	80
3.6 Sperren von Ereignismeldungen.....	82
3.7 Zugriffskontrolle.....	84
3.8 Segmentierung	88
4 Anhang	90
A Telegramm-Bearbeitung	90
B Netzwerk-Interface.....	95
C Attribute und Dimensionen	96
D Fehlerbehandlung	104
D.1 REJECT-APDU-Format (Ablauffehler)	104
D.2 ERROR-APDU-Format (syntaktische/semantischer Fehler)	105
E Übersicht der Datenpunkt-Tabellen	106
E.1 Meldepunkt	106

E.2 Schaltpunkt (mit Rückmeldung)	108
E.3 Messpunkt	110
E.4 Stellpunkt	112
E.5 Zählpunkt	114
F Übersicht der Codierungen im FND-APDU	116
G Format der Referenzdatei	118
H Zahlendarstellung im Gleitkomma-Format nach IEEE 754	119
5 Literaturverzeichnis	120
6 Mitarbeiter	121

-Diese Seite ist leer.-

-Diese Seite ist leer.-

Vorwort

Grundsätze und unterschiedliche Verfahren für die Planung, Errichtung und den Betrieb einer systemübergreifenden Gebäudeautomation (GA) in öffentlichen Gebäuden hat der Arbeitskreis Maschinen- und Elektrotechnik staatlicher und kommunaler Verwaltungen (AMEV) in der Empfehlung „Gebäudeautomation 2005“ zusammen gefasst.

Schon 1988 wurde vom AMEV das von ihm entwickelte firmenneutrale Datenübertragungsprotokoll (FND) in der Spezifikation Version 1.0 eingeführt. Dies war eines der ersten Protokolle mit dem eine herstellerübergreifende Verbindung von Gebäudeautomationssystemen ermöglicht werden konnte.

Diese Protokollspezifikation wurde im Laufe der Zeit aufgrund der praktischen Erfordernisse und der technischen Entwicklung modifiziert und ergänzt und fand ihren Niederschlag in weiteren AMEV-Empfehlungen. Zwischenzeitlich hat die auf deren Grundlage von der Stadt München entwickelte Protokollerweiterung über mehrere Jahre ihre Praxistauglichkeit bewiesen und bildet jetzt die Basis für alle FND-basierten Kopplungen von Gebäudeautomations-einrichtungen.

Die vorliegende Empfehlung „FND 2009“ dokumentiert den aktuellen Stand dieser stabilen und praxiserprobten FND-Protokollspezifikation und ergänzt damit die Empfehlung „Gebäudeautomation 2005“. Sie ersetzt die Empfehlungen „Planung und Ausführung von firmenneutralen Datenübertragungssystemen in öffentlichen Gebäuden und Liegenschaften (FND)“ Teile 1 bis 5 einschließlich Ergänzungen, die hiermit zurück gezogen werden.

Dr. Georg Printz
Vorsitzender des AMEV

Dr. Bernhard Hall
Obmann AK Gebäudeautomation

-Diese Seite ist leer.-

1 Geltungsbereich und Anwendungshinweise

Die nachfolgenden Hinweise gelten für Planung, Ausführung und den Betrieb der Gebäudeautomation in öffentlichen Gebäuden unter Verwendung des FND-Protokolls.

Das FND-Protokoll bietet die Möglichkeit zur objekt-orientierten Zusammenfassung der einzelnen Informationen einer Mess- bzw. Stellgröße in einem Datenpunkt und deren Übertragung in einem gemeinsamen Datentelegramm. Die Spezifikation legt aber nicht fest, ob und wie diese Informationen auf der Sensor-/Aktor-Ebene oder innerhalb der Automationseinrichtungen realisiert werden müssen. Hier können durchaus datenpunkt- und herstellersizifische, kostenrelevante Unterschiede auftreten (z.B. Geber-Störung für eine Meldung erfordert 4/20mA-Schleife o.ä.).

„FND 2009“ beschränkt sich auf die Beschreibung der FND-Datenstrukturen und der zugehörigen Operationen. Anforderungen bezüglich des einzusetzenden Datentransport-Systems sind nicht enthalten. Die Schnittstelle zum Transport-System ist projektspezifisch festzulegen und in die Ausschreibung aufzunehmen. Bewährte Schnittstellen existieren z. B. für IP-Netze, ISDN-Wählverbindungen und serielle V.24/RS232-Schnittstellen.

Grundsatzthemen, die unabhängig vom FND-Einsatz zu betrachten sind (z.B. Wirtschaftlichkeit, Energieeinsparung, Systemintegration, GA-Konzepte, Nutzerforderungen, Wettbewerb, Kostenplanung, Abnahme, Betriebsunterlagen, Betriebspersonal und Instandhaltung), sind nicht Gegenstand dieser Empfehlung. Hierzu sind die entsprechenden Hinweise in der AMEV-Empfehlung „Gebäudeautomation 2005“ [1] zu beachten. Diese AMEV-Empfehlung wird hinsichtlich der grundlegenden Begriffe und Beschreibungen zur GA nachfolgend als bekannt vorausgesetzt.

Dem Leser, der sich einen Überblick über das FND-Protokoll verschaffen möchte, werden folgende Kapitel empfohlen

- Abschnitt 2 Einleitung
- Abschnitt 3.2 Kommunikationsmodell

Eine zusammenfassende Übersicht zum FND-Protokoll für GA-Fachleute findet sich unter

- Anhang E Übersicht der Datenpunkt-Tabellen
- Anhang F Übersicht der Codierungen im FND-APDU

2 Einleitung

Das *firmenneutrale* **D**atenübertragungssystem (**FND**) stellt eine umfassende Spezifikation der Datenkommunikation für Systeme insbesondere der Gebäudeleittechnik (**GLT**) dar und ermöglicht eine flächendeckende objekt-, gewerke- und systemübergreifende Gebäudeautomation (**GA**).

Bereits 1988 hat der AMEV die FND-Spezifikation in der Version 1.0 veröffentlicht, die 1990 mit Anmerkungen und Ergänzungen in die DIN V 32735 und die EN V 1805-2 eingegangen ist. Um die hinderliche Beschränkung auf X.25 als einziges von der Spezifikation zugelassenes Transportprotokoll aufzuheben und eine divergente Weiterentwicklung der Spezifikation zu vermeiden, wurde 1996 vom AMEV die FND Version 2.0 veröffentlicht.

Motiviert durch technologische Entwicklungen und veränderte Anforderungen wurde das FND-Konzept von AMEV-Mitgliedern konsequent weiterentwickelt. Unter Anwendung offener Standards und Schnittstellen wurde eine GA-Plattform, bestehend aus einer Managementeinrichtung und einem integrierenden Rechnersystem (**GA-Knoten**; vgl. [2]), entwickelt. Damit können Automationseinrichtungen unterschiedlichster Hersteller und Technologien in das Managementsystem eingebunden werden und die dauerhafte Herstellerunabhängigkeit sicher gestellt werden.

Die Erfahrungen beim Einsatz von FND werden im Arbeitskreis Gebäudeautomation des AMEV auch im Hinblick auf organisatorische und administrative Belange (z.B. VOB) diskutiert und bei der kontinuierlichen Weiterentwicklung der Managementeinrichtungen berücksichtigt. Unter www.fnd-forum.de stehen – neben den quelloffenen Programmen der Managementeinrichtung – Organisationshilfen wie Ausschreibungstexte und Richtlinien zur Gebäudeautomation zur Verfügung, so dass ein vom Planer bis zum Betreiber und Nutzer handhabbares Gesamtsystem entsteht.

Dieses Verfahren wird von mehreren Verwaltungen genutzt. Es führt bei der Beschaffung von GA-Anlagen zu Einsparungen von 30-50% gegenüber produktgebundenen Vergaben und sichert langfristig Unabhängigkeit, Wettbewerb und auch die getätigten Investitionen (vgl. [2]).

Die Mitglieder des AMEV Arbeitskreises Gebäudeautomation unterstützen mit ihren Erfahrungen interessierte Verwaltungen bei der Einführung einer GA auf Basis dieser Plattform.

2.1 Grundidee des FND

Ausgangspunkt für die Entwicklung des FND waren Probleme und Abhängigkeiten bei der Kopplung von GA-Systemen unterschiedlicher Hersteller und deren Zusammenführung auf einer gemeinsamen Leitzentrale (**LZ**).

Werden die LZ oder das Netz für die Datenübertragung von und zur LZ hersteller- bzw. produktspezifisch realisiert, so entsteht eine Abhängigkeit an zentraler Stelle. Dadurch wird sowohl der Wettbewerb für zukünftige Anpassungen oder Erweiterungen als auch die getätigten Investitionen gefährdet.

Zur Lösung dieses Problems wird zwischen der übergeordneten Managementeinrichtung und der Automationseinrichtung eine Schnittstelle als Liefergrenze vorgegeben, die sowohl technische als auch organisatorische und administrative Festlegungen enthält. Bei der Beschaffung von Automationseinrichtungen werden diese Festlegungen zum Vertragsbestandteil.

Der Ersteller der Automationseinrichtung hat die benötigten GA-Informationen als Datenpunkte (**DP**; vgl. nachfolgenden Abschnitte) an der Liefergrenze bereitzustellen. Hierzu wird das FND-Protokoll gefordert.

FND basiert auf einer firmen-/produkt-neutralen, formalen, technischen „Sprache“ (Protokoll) zur Übertragung von Datenpunkt-Informationen und den zugehörigen Operationen.

Der Ersteller kann z.B. einen Standardschnittstellenadapter (**SSA**) verwenden, der das hersteller- oder produktspezifische Protokoll in das FND-Protokoll umsetzt. Der SSA kann als Hardware- und/oder Software-Komponente ausgeführt werden.

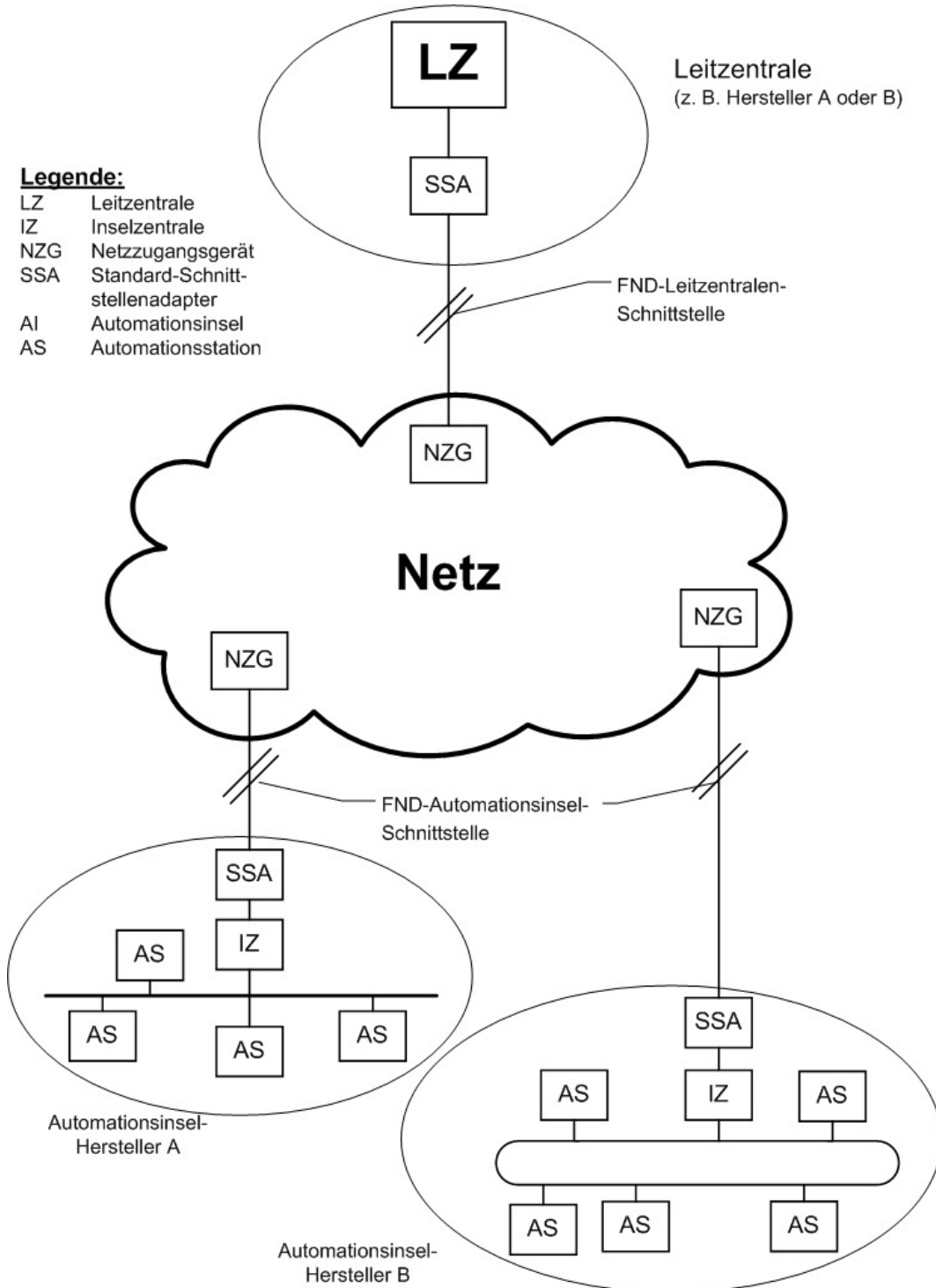


Abb. 2.1-1: Prinzipdarstellung eines FND-GA-Systems

Automationseinrichtungen inkl. aller für die Anbindung erforderlichen Komponenten eines jeden Erstellers bilden somit eine abgeschlossene Automationsinsel (**AI**), deren Grenzen hinsichtlich Ausführung, Inbetriebnahme, Abnahme, Gewährleistung und Haftung durch den Planer und die zuständige Vergabestelle definiert werden.

Der Informationsaustausch erfolgt über logische Punkt-zu-Punkt-Verbindungen zwischen LZ und AI. Die direkte Kommunikation zwischen den Automationseinrichtungen unterschiedlicher Ersteller wird bewusst ausgeschlossen, um gegenseitige Beeinflussungen und die damit verbundenen Probleme beim Nachweis im Gewährleistungs- oder Schadensfall zu vermeiden (vgl. VOB).

Für die Identifikation wird jedem DP eine eindeutige **Datenpunkt-Adresse** zugeordnet. Ergänzt um Angaben zum Datenpunkt-Typ, Attribut bzw. Dimension und Beschreibung der physikalischen Größe sowie den zugeordneten Zugriffsrechten werden die Datenpunkte-Adressen in einer Textdatei mit vorgegebenem Format aufgelistet. Diese **Referenzdatei** beschreibt die von der Automationsinsel bereitgestellten Datenpunkte und ist Bestandteil des Lieferumfangs (vgl. Anhang G).

2.2 Datenpunkte

Basierend auf den physikalischen Ein/Ausgabefunktionen der DIN EN ISO 16484-3 und VDI 3814 werden von FND folgende Datenpunkt-Typen bereitgestellt:

- Meldepunkt
- Schaltpunkt
- Messpunkt
- Stellpunkt
- Zählpunkt

Je DP werden die Informationen in bis zu 4 Gruppen unterteilt. Neben den eigentlichen **Nutzdaten** (z.B. Zustand/Wert des Sensor bzw. Aktor) werden dem DP **Status-Informationen** (z.B. Geber-Störung, Bedienung vor Ort, temporäre Verriegelung,...) und **Parameter** (z.B. Grenzwerte) zugeordnet. Dadurch kann der Benutzer an der LZ erkennen, unter welchen Randbedingungen der Zustand/Wert interpretiert werden muss. Meldet ein DP z.B. gleichzeitig einen erhöhten Wert mit Grenzwertverletzung und die Bedienung vor Ort, so kann davon ausgegangen werden, dass zu Testzwecken bei einer Wartung der Wert manuell eingestellt wurde.

Beispiel: Auf diese Weise besteht die Möglichkeit, z.B. einen DP „Pumpe 1“ nach seinem aktuellen Zustand bzw. den DP „Kesseltemperatur“ nach seinem aktuellen Wert zu fragen. Genauso kann die LZ einem DP „Flurlicht“ befahlen: „Schalte EIN“.

3 FND-Detailfestlegungen

3.1 Definitionen und Bezeichnungen

ISO-/CCITT-Definitionen

- APCI „Application-Protocol-Control-Information“, die den Anwender-Daten (ASDU) beigefügte applikations-spezifische Kontroll-Information (nachfolgend auch „**Header**“ genannt).
- APDU „Application-Protocol-Data-Unit“, die Dateneinheit, die dem Transport-System (Netz) übergeben wird (nachfolgend auch „**Telegramm**“ genannt).
- ASDU „Application-Service-Data-Unit“ (nachfolgend auch „**Datenteil**“ genannt)

Zahlen werden dargestellt in

- binärer Schreibweise als XX ...XXb, wobei „X“ jeweils für eines der Binärsymbole {0,1} steht
- dezimaler Schreibweise als XX...XX, wobei „X“ jeweils für eines der Dezimalsymbole {0,1,2,3,5,6,7,8,9 } steht
- hexadezimaler Schreibweise als XX ...XXh, wobei „X“ jeweils für eines der Hexadezimalsymbole {0,1,2,3,5,6,7,8,9,A,B,C,D,E,F} steht

3.2 Kommunikationsmodell

Die LZ sendet bei Bedarf Befehle (**CMD**) zum Lesen oder Modifizieren von DP-Informationen und erhält normalerweise eine Antwort (**RSP**). Tritt am DP ein relevantes Ereignis auf (z.B. Zustands-/Wertänderung am Geber), so kann der DP auch spontan eine Meldung (**USM**) an die LZ senden und erhält eine Bestätigung (**ACK**).

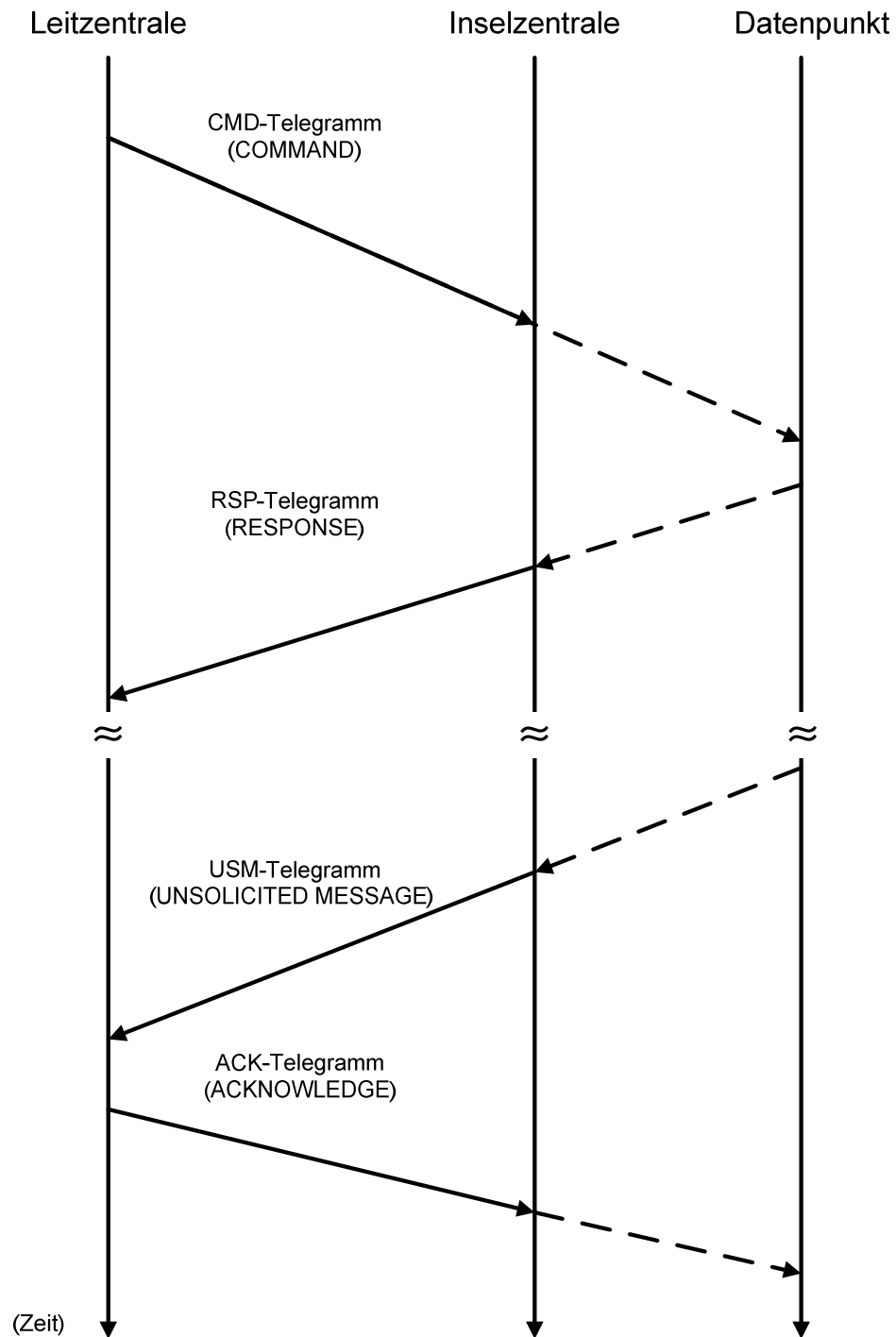


Abb. 3.2-1: FND-Kommunikationsmodell

Ereignismeldungen (USM-Telegramme) können bei Bedarf für jeden DP gesondert in 4 Prioritätsstufen gesperrt werden (vgl. 3.6 Sperren von Ereignismeldungen).

Im Fehlerfall wird das RSP- bzw. ACK-Telegramm als REJECT- oder ERROR-Telegramm gekennzeichnet (vgl. Anhang D Fehlerbehandlung).

Die Prozedur-Elemente für Sendeaufträge und Empfangsbestätigungen sind beschrieben im Anhang A Telegramm-Bearbeitung.

Ein FND-Telegramm (APDU) setzt sich aus zwei Teilstrukturen zusammen:

- dem für alle Telegramme gleich aufgebauten Header (APCI) fester Länge, der Adress- und Kontroll-Informationen enthält
- dem Datenteil (ASDU), der auch leer sein kann

Der Header selbst gliedert sich wiederum in zwei Teilstrukturen (APCI.1 und APCI.2). APCI.1 enthält die Adress- und Kontrollinformationen zur Steuerung des Transports zum bzw. von dem betreffenden Datenpunkt und APCI .2 die eigentlichen DP-spezifischen Informationen.

Folgende Bestandteile des APCI haben grundlegende Bedeutung (vgl. 3.4 FND – APDU-Format):

- d_r Telegramm-Typ (Richtung/Antwort =>CMD, RSP, USM, ACK)
- tag Telegramm-Attribut (NORMAL, REJECT, ERROR)
- dp_id Datenpunkt-Adresse
- fct_id Funktion/Operation auf dem Datenpunkt (Lesen, Modifizieren)
- tab_id Datenpunkt-Tabelle, auf die die Funktion/Operation wirkt
- dp_type Datenpunkt-Typ (Melde-, Schalt-, Mess-, Stell-, Zählpunkt,...)

Länge und Inhalt des Datenteils sind abhängig vom Telegrammtyp, Datenpunkt-Typ und der gewünschten Funktion sowie den Prozessbedingungen im Umfeld des Datenpunkts (vgl. Symbol-Erklärung zur Operations-Übersicht des jeweiligen Datenpunkt-Typs).

-Diese Seite ist leer.-

3.3 Beschreibung der Datenpunkte

In den FND-Telegrammen ist jedem Datenpunkt-Typ ein Wert im Feld „dp_type“ zugeordnet:

dp_type	Datenpunkt-Typ
0	Fehlerbehandlung (vgl. Anhang D)
1	Meldepunkt
2	Schaltpunkt (mit Rückmeldung)
3	Messpunkt
4	Stellpunkt
5	Zählpunkt
6	* reserviert *
7	Karteipunkt

Bei der Beschreibung der Datenpunkte werden unter einer Datenpunkt-Adresse (**dp_id**) mehrere Einzelinformationen in Datenpunkt-Tabellen zusammengefasst, die für jeden Datenpunkt-Typ spezifisch aufgebaut sind.

DP-Tabelle	Informations-Inhalt	Funktion
DP_0	Status	Anzeige von Ereignissen und Zusatz-Informationen, Sperren / Freigeben von Ereignisquellen
DP_1	Nutzdaten	Lesen, Modifizieren, Melden
DP_2	(Konfigurations)-Parameter	Parametrieren

Als Bestandteil des „Operations-Codes“ ist die Nummer der DP-Tabelle (**tab_id**) im FND-Telegramm wie folgt vergeben:

tab_id	DP-Tabelle
0	DP_0
1	DP_1
2	DP_2
3	DP_1&2

Hinweis: „DP_1&2“ bedeutet DP_1 gefolgt von DP_2

Auf diesen DP-Tabellen können lesende und ggf. modifizierende Zugriffe durchgeführt werden. Sie unterscheiden sich durch die Funktionskennung (fct_id):

fct_id	Funktion
1	Lesen
2	Modifizieren (Schreiben)

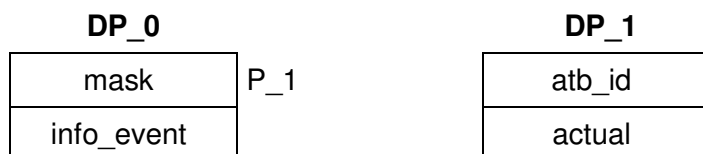
Hinweis: Modifizierbare Parameter in einer Datenpunkt-Tabelle sind in den nachfolgenden Detailbeschreibungen mit „P_1“, „P_2“, „P_3“ oder „P_4“ gekennzeichnet

3.3.1 Meldepunkt

Charakterisierung:

Meldepunkte stellen binäre (boolesche) Größen dar, welche bei physikalischen Gebern i.a. den Zustand elektrischer Kontakte („offen“/ „geschlossen“), bei logischen (virtuellen) Gebern den Aktivitätszustand eines Prozesses widerspiegeln. Mehrstufigen Betriebsmeldungen sind bei physikalischen Gebern i.a. die Einzelkontakte eines Stufenschalters, bei logischen (virtuellen) Datenpunkten etwa Ablaufvarianten (Modi) von Zeit- oder Ereignisprogrammen zugeordnet.

Datenpunkt-Tabellen:

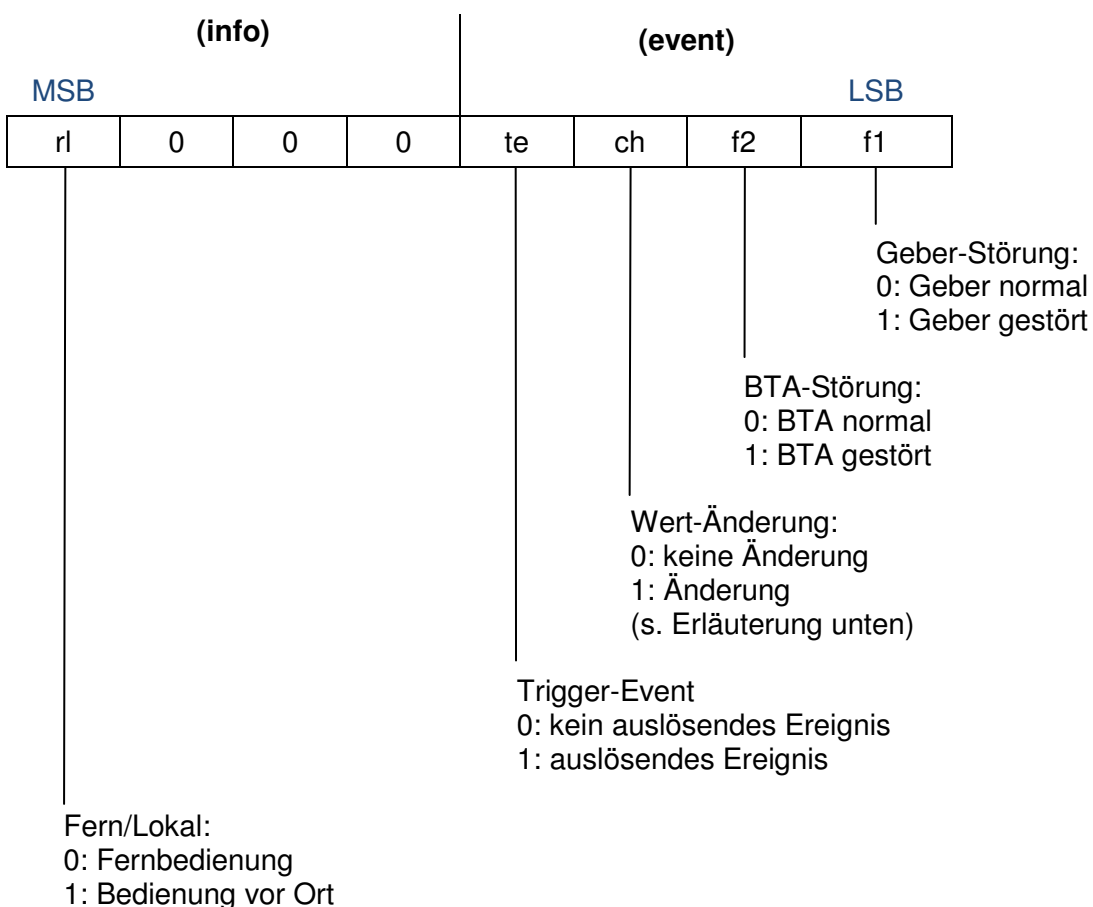


Variablenbeschreibung DP_0:

mask: (P_1)

(Zur Beschreibung dieses Bytes:siehe 3.6 Sperren von Ereignismeldungen)

info_event:



Erläuterungen zu info_event:

- f1 (primary fault):
Ein gesetztes „f1“-Bit bedeutet eine Störung des Geberteils, die bei virtuellen Datenpunkten als ein erkannter interner Hard- oder Software-Fehler zu interpretieren ist.
Art und Ursache der Störung werden durch den Info-Code („exc“) näher spezifiziert.
- f2 (secondary fault):
Eine Störung der BTA (Warn-, Alarm-Zustand etc.) wird durch ein gesetztes „f2“-Bit angezeigt.
- ch (change):
Die Wertänderung eines Meldepunkts wird durch Setzen des „ch“-Bits signalisiert; Rücksetzen erfolgt aufgrund der LZ-seitigen Bestätigung der Meldung, bzw. bei Ausbleiben derselben nach einem intern gesetzten Timeout; der Rücksetzvorgang wird nicht als Ereignis gewertet.
Solange „ch“ auf 1 steht, bewirken weitere Wertänderungen keine Meldung.
Sind Ereignisse dieser Art (AI- oder LZ-seitig) ausgeblendet, so hat „ch“ während dieser Zeit den Wert 0.
- te (triggering-event):
Ein gesetztes „te“-Bit signalisiert das Eintreten eines projektspezifisch vereinbarten Ereignisses (z.B. den Ablauf eines AI-internen implementierten (zyklischen) Timers), welches die Übertragung der Datenpunkt-Tabelle DP_1 impliziert.
Das Rücksetzen des Bits im info_event_Byte der DP-Tabelle DP_0 erfolgt aufgrund der LZ-seitigen Bestätigung der Meldung bzw. bei Ausbleiben derselben nach einem intern gesetzten Timeout. Der Rücksetzvorgang selbst wird nicht als Ereignis gewertet.
Solange „te“ auf 1 steht, bewirken weitere „te“-bezogene Ereignisse keine Meldung.
Sind Ereignisse dieser Art (AI-oder LZ-seitig) ausgeblendet, (d.h. $msk \geq 1$ oder $mis \geq 1$), so hat „te“ während dieser Zeit den Wert 0.
- rl (remote/local):
Die Zustände Fern-/Vor-Ort-Bedienung werden durch das „rl“-Bit unterschieden. Diese Kennung ist insbesondere für solche Datenpunkte von Bedeutung, die Betriebsmeldungen zu Schaltpunkten darstellen.

Parameter-Übersicht:

P_	Name	Erläuterung
1	mask	Unterdrückung von Spontanmeldungen aufgrund spezieller Ereignisse

Variablenbeschreibung DP 1:**atb_id:**

Zur Kennzeichnung der (im physikalischen Sinn) dimensionslosen Meldungen wird eine (erforderlichenfalls erweiterbare) Liste codierter Attributkombinationen verwendet, welche auf die GLT-spezifischen Anforderungen zugeschnitten ist; Beispiele hierfür sind etwa die Kombinationen („EIN“/„AUS“), („AUF“/„ZU“), oder („Stufe 0“/„Stufe I“/„Stufe II“). Die Wertigkeit der Meldung (ein- oder mehrstufig) ist dabei implizit durch die Anzahl der in der Kombination enthaltenen Attribute gegeben. Der Wert von atb-id ist identisch mit der Code-Nummer der Attribut-Kombination.

Die derzeit definierten Attribut-Kombinationen einschließlich ihrer Codierungen sind in Anhang C aufgelistet.

actual:

<FFh> kennzeichnet einen ungültigen Wert).

Andere Werte bezeichnen die aktuelle Meldung(s-Stufe), wobei

- höchstens ein Bit gesetzt und
- die Nummer dieses Bits (1-8) höchstens gleich der um 1 verminderten Anzahl der in der durch „atb_id“ adressierten Attribut-Kombination vorhandenen Attribut

sein darf.

„actual“ ist dabei als Index im Wertebereich von 0 - 8 in der durch atb_id adressierten Attribut-Kombination zu interpretieren:

actual (binär)	Index
0 0 0 0 0 0 0 0	0
0 0 0 0 0 0 0 1	1
0 0 0 0 0 0 1 0	2
0 0 0 0 0 1 0 0	3
0 0 0 0 1 0 0 0	4
0 0 0 1 0 0 0 0	5
0 0 1 0 0 0 0 0	6
0 1 0 0 0 0 0 0	7
1 0 0 0 0 0 0 0	8

Zwei Beispiele sollen die vorstehenden Ausführungen verdeutlichen:

Beispiel 1:

Für einen Datenpunkt mit den Meldungen („EIN“/„AUS“) ergeben sich nach der durch Anhang C gegebenen Codierung für die DP-Tabelle 1 die Werte

atb_id	0 0 0 0 1 0 1 0	(Code 10 für „EIN“/„AUS“)
actual	0 0 0 0 0 0 0 0	(Index: 0) für „EIN“ und

atb_id	0 0 0 0 1 0 1 0	(Code 10 für „EIN“/„AUS“)
actual	0 0 0 0 0 0 0 1	(Index: 1) für „AUS“.

Beispiel 2:

Ein dreistufiger Meldepunkt für die Zustände „Normal“ / „Warnung“ / „Alarm“ (Code 44) ist folgendermaßen zu behandeln:

atb_id	0 0 1 0 1 1 0 0	(Code: 44)
actual	0 0 0 0 0 0 0 0	(Index: 0) für „Normal“

atb_id	0 0 1 0 1 1 0 0	(Code: 44)
actual	0 0 0 0 0 0 0 1	(Index: 1) für „Warnung“

und

atb_id	0 0 1 0 1 1 0 0	(Code: 44)
actual	0 0 0 0 0 0 1 0	(Index: 2) für „Alarm“.

Andere Werte für „actual“ sind ungültig.

Operations-Übersicht

Meldepunkt		fct_id	
d_r	tab_id	1 (Lesen)	2 (Mod.)
0 (CMD)	0 (DP_0)	C1	C2
	1 (DP_1)	C3	* entfällt *
3 (RSP)	0 (DP_0)	R1	R2
	1 (DP_1)	R3	* entfällt *
2 (USM)	1 (DP_1)	U1	* entfällt *
1 (ACK)	1 (DP_1)	A1	* entfällt *

Symbol-Erklärung

Operation	Bedeutung	Daten-Teil
C1	DP-Status (DP_0) anfordern	(leer)
C2	Meldungs-Kontroll-Feld (msk in DP_0) mod.	(leer)
C3	DP-Nutzdaten (DP_1) anfordern	(leer)
R1	DP-Status (DP_0) übermitteln	(leer)
R2	Modifikationsbestätigung des Meldungs-Kontrollfeldes	(leer)
R3	DP-Nutzdaten (DP_1) übermitteln	DP_1
U1	Spontanmeldung	DP_1
A1	Bestätigung Spontanmeldung	leer

Beispiel:

Die LZ möchte die aktuelle Meldungs-Stufe lesen. Diese Information („actual“) ist Bestandteil der Nutzdaten in DP-Tabelle 1. Die LZ sendet daher ein CMD-Telegramm gemäß Operation C3 und erhält im RSP-Telegramm gemäß R3 als Datenteil die 2 Bytes „atd_id“ und „actual“ der DP_1.

-Diese Seite ist leer.-

3.3.2 Schaltpunkt (mit Rückmeldung)

Charakterisierung:

Der Schaltpunkt verfügt über ein „integriertes Feedback“, d.h. die zugehörige (Betriebs-)Meldung ist über die gleiche Datenpunkt-Adresse verfügbar.

Ein Schaltpunkt mit Rückmeldung entspricht einem Meldepunkt (inkl. Spontanmeldung bei Wertänderung), dem zusätzlich ein Sollwert für das Schalten vorgegeben werden kann. Die AI entscheidet aufgrund der zugeordneten Funktion, ob und wann der Schaltvorgang ausgeführt wird. Die Rückmeldung kann über einen gesonderten Geber oder intern in der AI erzeugt werden.

Beispiel: Eine helligkeitsgesteuerte Außenbeleuchtung wird nur in bestimmten Zeitbereichen genutzt. Innerhalb dieser Zeitbereiche wird der Zustand EIN für den Schaltpunkt vorgegeben. Die Außenbeleuchtung schaltet aber nur dann ein, wenn auch der integrierte Dämmerungsschalter einschaltet. Die AI sendet dann für die Wertänderung spontan eine Schaltmeldung. Beim Vorgabewert AUS bleibt die Außenbeleuchtung unabhängig vom Dämmerungsschalter ausgeschaltet.

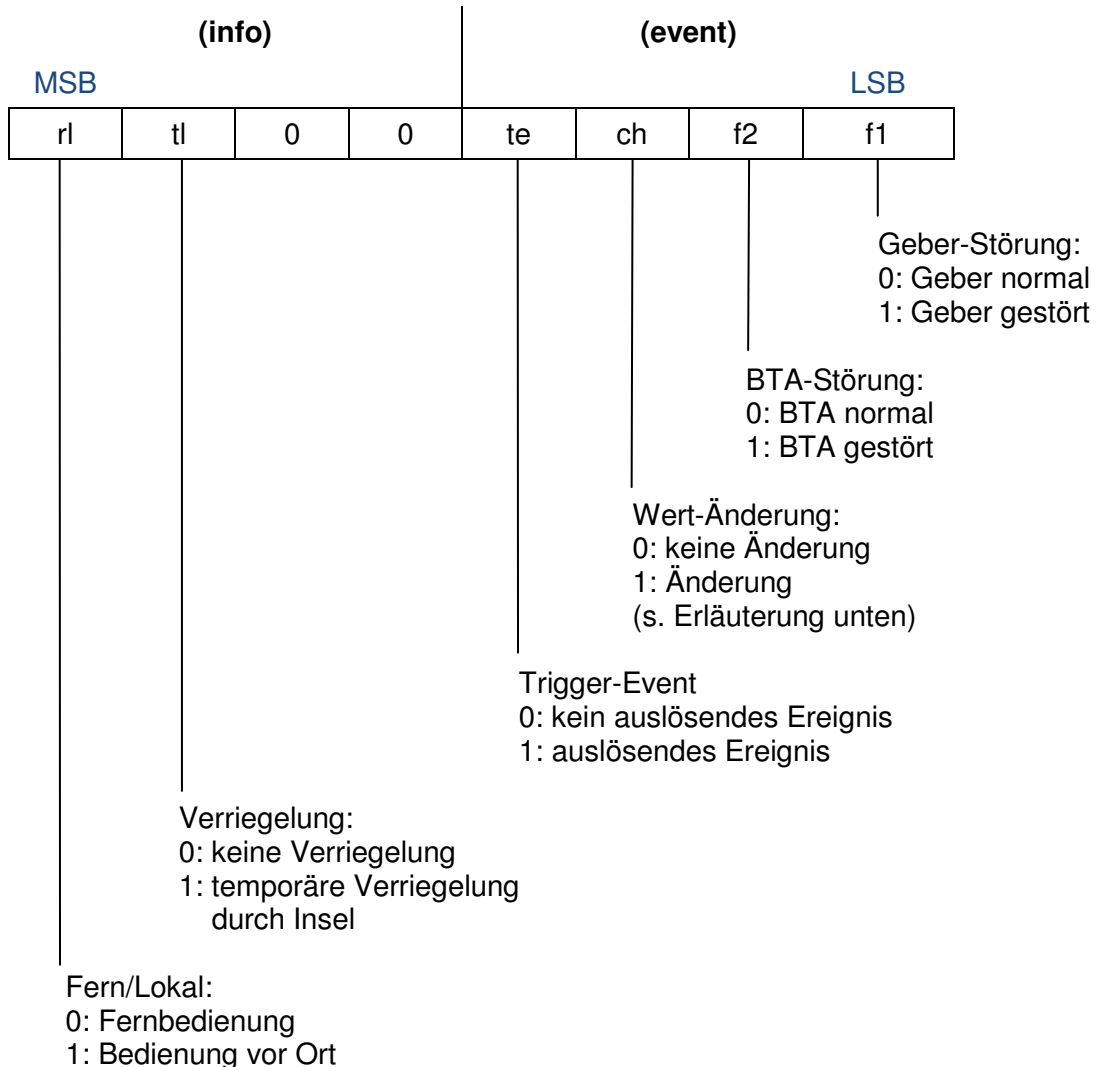
Hinweis: Es wird empfohlen Schaltpunkte immer mit Rückmeldung einzusetzen. Dabei sollte abhängig von der Funktion entschieden werden, ob die Rückmeldung über einen gesonderten Geber oder intern in der AI erzeugt werden soll.

Datenpunkt-Tabellen:

DP_0			DP_1	
mask	P_1		atb_id	P_1
info_event			actual	
			nominal	

Variablenbeschreibung DP 0:**mask: (P_1)**

(Zur Beschreibung dieses Bytes:siehe 3.6 Sperren von Ereignismeldungen)

info_event:**Erläuterungen zu info_event:**

- f1 (primary fault):
 Ein gesetztes „f1“-Bit bedeutet eine Störung des Geberteils, die bei virtuellen Datenpunkten als ein erkannter interner Hard- oder Software-Fehler zu interpretieren ist.
 Art und Ursache der Störung werden durch den Info-Code („exc“) näher spezifiziert.
- f2 (secondary fault):
 Eine Störung der BTA (Warn-, Alarm-Zustand etc.) wird durch ein gesetztes „f2“-Bit angezeigt.
- ch (change):
 Wertänderung eines Meldepunkts wird durch Setzen des „ch“-Bits signalisiert; Rücksetzen erfolgt aufgrund der LZ-seitigen Bestätigung der Meldung, bzw. bei Ausbleiben derselben nach einem intern gesetzten Timeout- Der Rücksetzvorgang wird nicht als Ereignis gewertet.

3.3.2 Schaltpunkt

Solange „ch“ auf 1 steht, bewirken weitere Wertänderungen keine Meldung.

Sind Ereignisse dieser Art (AI- oder LZ-seitig) ausgeblendet, so hat „ch“ während dieser Zeit den Wert 0.

In der Antwort auf einen Schaltbefehl ist das „ch“-Bit zu setzen, falls der Befehl sofort ausgeführt werden konnte.

Für Schaltpunkte ohne Rückmeldung ist das „ch“-Bit stets 0 zu setzen.

te (triggering-event):

Ein gesetztes „te“-Bit signalisiert das Eintreten eines projektspezifisch vereinbarten Ereignisses (z.B. den Ablauf eines AI-internen implementierten (zyklischen) Timers), welches die Übertragung der Datenpunkt-Tabelle DP_1 impliziert.

Das Rücksetzen des Bits im info_event_Byte der DP-Tabelle DP_0 erfolgt aufgrund der LZ-seitigen Bestätigung der Meldung bzw. bei Ausbleiben derselben nach einem intern gesetzten Timeout. Der Rücksetzvorgang selbst wird nicht als Ereignis gewertet.

Solange „te“ auf 1 steht, bewirken weitere „te“-bezogene Ereignisse keine Meldung.

Sind Ereignisse dieser Art (AI-oder LZ-seitig) ausgeblendet, (d.h. msk > 1 oder mis > 1), so hat „te“ während dieser Zeit den Wert 0.

tl (temporary lock):

Eine durch die Insel gesteuerte, zeitlich begrenzte Verriegelung des Schaltpunktes wird durch das „tl“-Bit angezeigt. Schaltbefehle seitens FND werden in diesem Zeitraum abgewiesen

rl (remote/local):

Die Zustände Fern-/Vor-Ort-Bedienung werden durch das „rl“-Bit unterschieden.

Parameter-Übersicht

P_	Name	Erläuterung
1	mask	Unterdrückung von Spontanmeldungen aufgrund spezieller Ereignisse

Variablenbeschreibung DP 1:**atb_id:**

(wie bei 3.3.1 Meldepunkt)

actual:

wie bei 3.3.1 Meldepunkt)

Für Schaltpunkte ohne Rückmeldung ist stets FFh (Kennzeichnung für einen ungültigen Wert) einzutragen.

nominal: (P_1)

Dieses Byte enthält die LZ-seitige Vorgabe des Schaltwertes, wobei

- höchstens ein Bit gesetzt sein darf (welches dem Schaltwert entspricht)
- die Nummer dieses Bits (1 – 8) höchstens gleich der um 1 verminderten Anzahl der in der durch „atb_id“ adressierten Attribut-Kombination vorhandenen Attribute sein darf.

(vgl. Erläuterung zu „actual“ eines Meldepunktes)

Parameter-Übersicht

P_	Name	Erläuterung
1	nominal	Der zum LZ- seitigen Schaltbefehl gehörende Wert

*) Anhand der auf einen LZ-seitigen Schaltbefehl eintreffenden Antwort kann abgelesen werden, ob der Vorgang bereits ausgeführt wurde (der zurückgemeldete „actual“-Wert stimmt mit dem gesendeten „nominal“-Wert überein) oder erst mit einer zeitlichen Verzögerung als Ereignis (Ausführungsbestätigung in Form einer Spontanmeldung) gemeldet wird.

Operations-Übersicht

Schaltpunkt		fct_id	
d_r	tab_id	1 (Lesen)	2 (Mod.)
0 (CMD)	0 (DP_0)	C1	C2
	1 (DP_1)	C3	C4
3 (RSP)	0 (DP_0)	R1	R2
	1 (DP_1)	R3	R4
2 (USM)	1 (DP_1)	U1	* entfällt *
1 (ACK)	2 (DP_1)	A1	* entfällt *

Symbol-Erklärung

Operation	Bedeutung	Daten-Teil
C1	DP-Status (DP_0) anfordern	(leer)
C2	Meldungs-Kontroll-Feld (msk in DP_0) mod.	(leer)
C3	DP-Nutzdaten (DP_1) anfordern	(leer)
C4	Schalten	DP_1
R1	DP-Status (DP_0) übermitteln	(leer)
R2	Modifikationsbestätigung des Meldungs-Kontrollfeldes	(leer)
R3	DP-Nutzdaten (DP_1) übermitteln	DP_1
R4	Schalt-Quittung	DP_1
U1	Schaltmeldung	DP_1
A1	Bestätigung Spontanmeldung	leer

Beispiel:

Die LZ möchte den Zustand EIN vorgeben. Diese Information („nominal“) ist Bestandteil der Nutzdaten in DP-Tabelle 1. Die LZ sendet somit ein CMD-Telegramm gemäß Operation C4 und erhält ein RSP-Telegramm gemäß R4. Wurde der Schaltbefehl sofort ausgeführt, so ist dabei das ch-Bit gesetzt und der „actual“ entspricht dem „nominal“. Sonst bleibt „actual“ unverändert und mit der späteren Ausführung wird eine spontane Schaltmeldung U1 erzeugt.

Für Schaltpunkte ohne Rückmeldung entfällt die Schaltmeldung (Operationen U1 und A1)

(Für weitergehende Ausführungen zu dieser tabellarischen Übersicht wird auf das Kapitel „Operationsbeschreibung“ verwiesen).

-Diese Seite ist leer.-

3.3.3 Messpunkt

Charakterisierung:

Messpunkte sind kontinuierlich veränderbaren Größen zugeordnet, die auf Einhaltung vorgegebener Grenzwerte überwacht werden können. Messpunkten können sowohl reale (z. B. Temperatur, Position eines Stellgliedes) wie auch virtuelle (z. B. ein aus mehreren Größen rechnerisch ermittelten Funktionswert) Größen zugeordnet werden.

Datenpunkt-Tabellen:

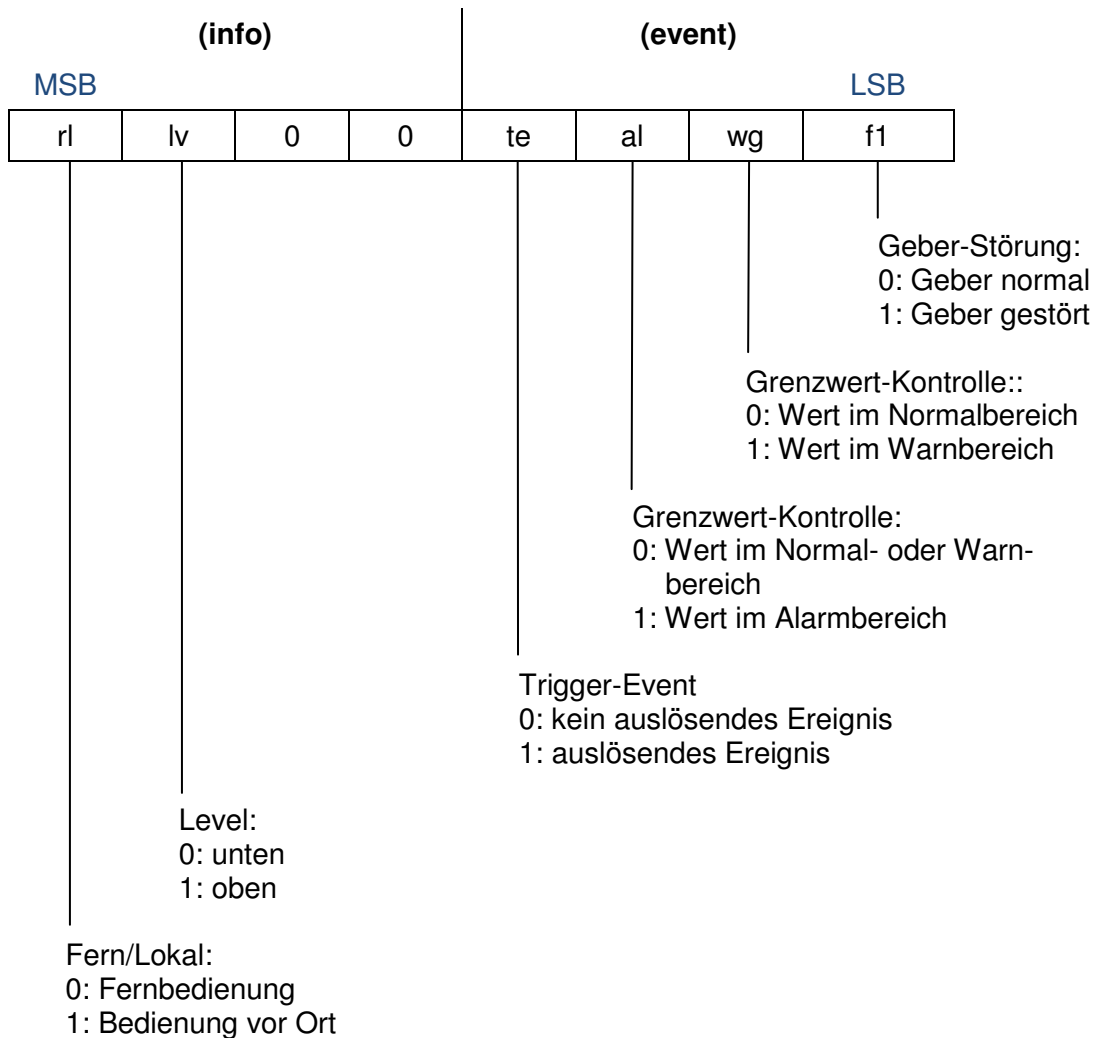
DP_0	
mask	P_1
info_event	

DP_1	
dimension	
actual	

DP_2	
update_control	
fixed	
alarm_low	P_1
warning_low	P_2
warning_high	P_3
alarm_high	P_4

Variablenbeschreibung DP 0:**mask: (P_1)**

(Zur Beschreibung dieses Bytes: siehe 3.6 Sperren von Ereignismeldungen)

info_event:**Erläuterungen zu info_event:**

- f1** (primary fault):
ein gesetztes „f1“-Bit bedeutet eine Störung des Geberteils, die bei virtuellen Datenpunkten als ein erkannter interner Hard- oder Software-Fehler zu interpretieren ist.

Art und Ursache der Störung werden durch den Info-Code („exc“) näher spezifiziert.
- wg** (warning):
Ein gesetztes „wg“-Bit zeigt an, dass der Messwert innerhalb des oberen (gekennzeichnet durch lv = 1) bzw. unteren (gekennzeichnet durch (lv = 0) Warnbereichs liegt. Der Übergang aus dem bzw. in den Warnbereich wird als Ergebnis gewertet.
- al** (alarm):
Analog zu „wg“, wobei „Warnbereich“ durch „Alarmbereich“ zu ersetzen ist. Der Warnbereich muss Teilmenge des Alarmbereiches sein.
- te** (triggering-event):

3.3.3 Messpunkt

Ein gesetztes „te“-Bit signalisiert das Eintreten eines projektspezifisch vereinbarten Ereignisses (z.B. den Ablauf eines AI-internen implementierten (zyklischen) Timers), welches die Übertragung der Datenpunkt-Tabelle DP_1 impliziert.

Das Rücksetzen des Bits im info_event_Byte der DP-Tabelle DP_0 erfolgt aufgrund der LZ-seitigen Bestätigung der Meldung bzw. bei Ausbleiben derselben nach einem intern gesetzten Timeout. Der Rücksetzvorgang selbst wird nicht als Ereignis gewertet.

Solange „te“ auf 1 steht, bewirken weitere „te“-bezogene Ereignisse keine Meldung.

Sind Ereignisse dieser Art (AI-oder LZ-seitig) ausgeblendet, (d.h. $msk \geq 1$ oder $mis \geq 1$), so hat „te“ während dieser Zeit den Wert 0.

lv (level):

Dieses Bit liefert bei Grenzwertmeldungen die Zusatzinformation „Überschreitung“ (= „oben“) bzw. „Unterschreitung“ (= „unten“).

rl (remote/local):

Die Zustände Fern-/Vor-Ort-Bedienung werden durch das „rl“-Bit unterschieden

Bei den zulässigen Kombinationen der vier Variablen „wg“, „al“, „lv“ und „emq“ ist zwischen einem statischen (Abfragen) und transienten Kontext (Spontanmeldungen) zu unterscheiden.

Zulässige Kombination bei Abfragen				
emq	lv	al	wg	Bedeutung
0 0 0 0	0	1	1	Messwert im unteren Alarmbereich
0 0 0 0	0	0	1	Messwert im unteren Warnbereich
0 0 0 0	0	0	0	Messwert im Normalbereich
0 0 0 0	1	0	1	Messwert im oberen Warnbereich
0 0 0 0	1	1	1	Messwert im oberen Alarmbereich
andere Kombinationen: reserviert				

Zulässige Kombination bei Spontanmeldungen (X = 0 oder 1)				
emq	lv	al	wg	Bedeutung
0 0 1 X	1	0	1	Überschreitung der oberen Warngrenze kommend Neuer Zustand: oberer Warnbereich
0 1 0 X	1	1	1	Überschreitung der oberen Alarmgrenze kommend Neuer Zustand: oberer Alarmbereich
0 1 0 X	1	0	1	Überschreitung der oberen Alarmgrenze gehend Neuer Zustand: oberer Warnbereich
0 0 1 X	1	0	0	Überschreitung der oberen Warngrenze gehend Neuer Zustand: Normalbereich
0 0 1 X	0	0	1	Unterschreitung der unteren Warngrenze kommend Neuer Zustand: unterer Warnbereich
0 1 0 X	0	1	1	Unterschreitung der unteren Alarmgrenze kommend Neuer Zustand unterer Alarmbereich
0 1 0 X	0	0	1	Unterschreitung der unteren Alarmgrenze gehend Neuer Zustand: unterer Warnbereich
0 0 1 X	0	0	0	Unterschreitung der unteren Warngrenze gehend Neuer Zustand: Normalbereich
andere Kombinationen: reserviert				

Parameter-Übersicht

P_	Name	Erläuterung
1	mask	Unterdrückung von Spontanmeldungen aufgrund spezieller Ereignisse

Variablenbeschreibung DP 1:

dimension:

„dimension“ repräsentiert in codierter Form die physikalische Dimension des Messwerts. Die Liste der von FND unterstützten Größen und ihrer Codes ist in Anhang C enthalten.

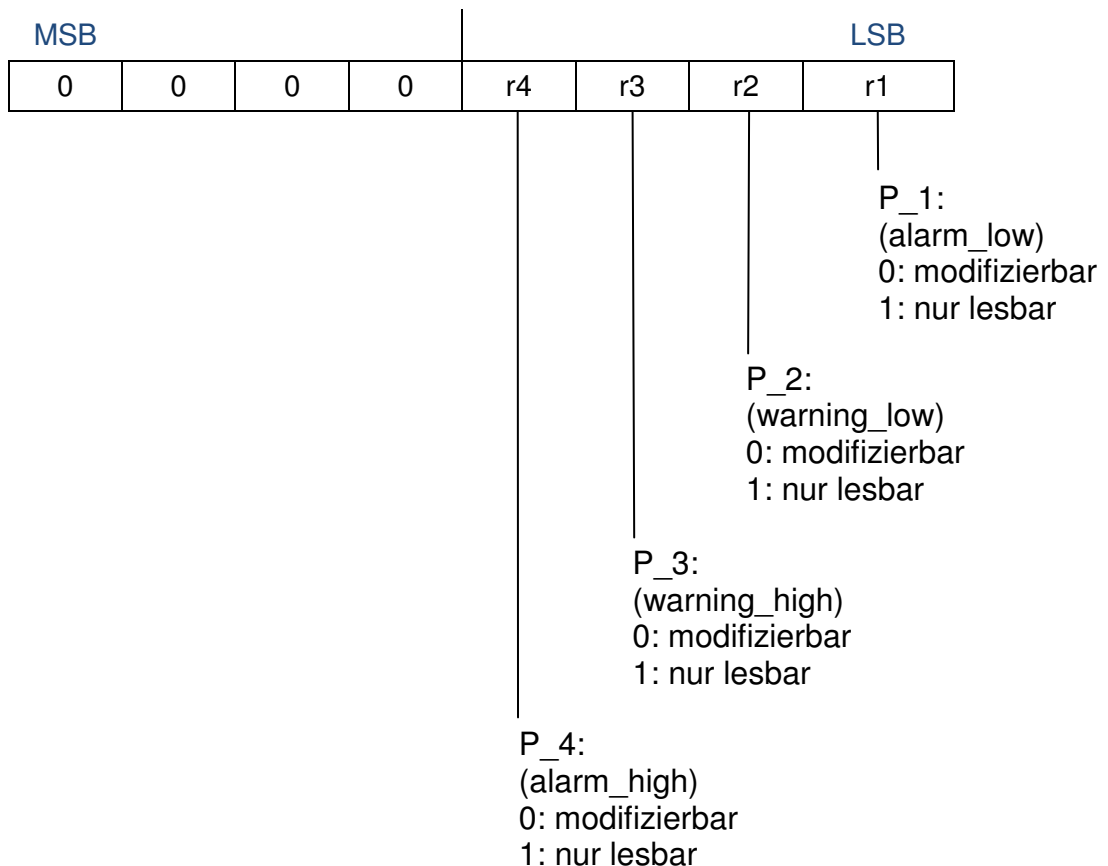
actual:

„actual“ enthält den aktuellen Messwert in IEEE 754-Darstellung. Ein Wert von FFFFFFFFh (alle Bits gesetzt) markiert einen ungültigen Messwert.

Variablenbeschreibung DP 2:**update_control:**

Dieses Byte stellt eine zusätzliche Modifizierungskontrolle der im Folgenden beschriebenen Parameter P_1 bis P_4 dar. Ein gesetztes Bit bedeutet, dass die LZ nur lesenden Zugriff auf den zugeordneten Parameter hat.

Dies kann relevant sein, wenn ein Parameter des Datenpunkts generell nicht unterstützt wird (angezeigt durch den Wert FFFFFFFFh) oder wenn z. B. nur der obere Grenzwert von der LZ aus verändert werden darf.

update_control:**fixed:**

Kennzeichnung, ob die nachstehenden Grenzen als absolut oder gleitend zu interpretieren sind. Dabei bedeutet:

- TRUE (=1): Grenzwerte absolut
- FALSE (=0): Grenzwerte gleitend

(Sofern keine Grenzwertvorgaben existieren, ist der Wert von „fixed“ ohne Belang).

Hinweis: Der Wert von „fixed“ ist stets auf TRUE zu setzen, da noch keine Spezifikation gleitender Grenzwerte vorliegt.

alarm_low: (P_1)

Untere Alarmgrenze des Messwerts. Ein Eintrag von FFFFFFFFh (alle Bits gesetzt) bedeutet die Inaktivität (oder Nichtexistenz) des Grenzwerts.

3.3.3 Messpunkt

warning_low: (P_2)

Untere Warngrenze des Messwerts. Ein Eintrag von FFFFFFFFh (alle Bits gesetzt) bedeutet die Inaktivität (oder Nichtexistenz) des Grenzwerts.

warning_high: (P_3)

Obere Warngrenze des Messwerts. Ein Eintrag von FFFFFFFFh (alle Bits gesetzt) bedeutet die Inaktivität (oder Nichtexistenz) des Grenzwerts.

alarm_high (P_4)

Obere Alarmgrenze des Messwerts. Ein Eintrag von FFFFFFFFh (alle Bits gesetzt) bedeutet die Inaktivität (oder Nichtexistenz) des Grenzwerts.

Parameter-Übersicht:

P_	Name	Erläuterung
1	alarm_low	Untere Alarmgrenze
2	warning_low	Untere Warngrenze
3	warning_high	Obere Warngrenze
4	alarm_high	Obere Alarmgrenze

Operations-Übersicht

Schaltpunkt		fct_id	
d_r	tab_id	1 (Lesen)	2 (Mod.)
0 (CMD)	0 (DP_0)	C1	C2
	1 (DP_1)	C3	* entfällt *
	2 (DP_2)	C4	C5
	3 (DP_1&2)	C6	* entfällt *
3 (RSP)	0 (DP_0)	R1	R2
	1 (DP_1)	R3	* entfällt *
	2 (DP_2)	R4	R5
	3 (DP_1&2)	R6	* entfällt *
2 (USM)	1 (DP_1)	U1	* entfällt *
1 (ACK)	1 (DP_1)	A1	* entfällt *

Symbol-Erklärung

Operation	Bedeutung	Daten-Teil
C1	DP-Status (DP_0) anfordern	(leer)
C2	Meldungs-Kontroll-Feld (msk in DP_0) mod.	(leer)
C3	DP-Nutzdaten (DP_1) anfordern	(leer)
C4	DP-Parameter (DP_2) anfordern	(leer)
C5	DP-Parameter (DP_2) modifizieren	DP_2
C6	DP-Nutzdaten+Par. (DP_1&2) anfordern	(leer)
R1	DP-Status (DP_0) übermitteln	(leer)
R2	Modifikationsbestätigung des Meldungs-Kontrollfeldes	(leer)
R3	DP-Nutzdaten (DP_1) übermitteln	DP_1
R4	DP-Parameter (DP_2) übermitteln	DP_2
R5	Modifikationsbestätigung Param. (DP_2)	DP_2
R6	DP-Nutzdaten+Par. (DP_1&2) übermitteln	DP_1&2
U1	Messwertmeldung	DP_1
A1	Bestätigung Messwertmeldung	(leer)

Beispiel:

Die LZ möchte neue Grenzwerte vorgeben. Diese Information („alarm_low“ usw.) ist Bestandteil der Parameter in DP-Tabelle 2. Die LZ sendet somit ein CMD-Telegramm C5 und erhält ein RSP-Telegramm R5.

(Für weitergehende Ausführungen zu dieser tabellarischen Übersicht wird auf das Kapitel 3.4.1 „Operationsbeschreibung“ verwiesen).

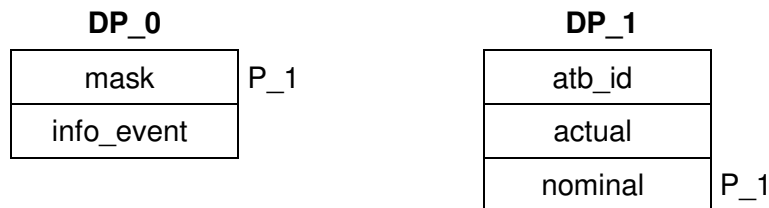
-Diese Seite ist leer.-

3.3.4 Stellpunkt

Charakterisierung:

Bei Stellpunkten führt die Vorgabe eines Stellwerts zu einer entsprechenden Stellung des Gebers. Es sind sowohl reale (Position eines physikalischen Stellglieds) als auch virtuelle Stellpunkte (z. B. Programm-Parameter) möglich.

Datenpunkt-Tabellen:

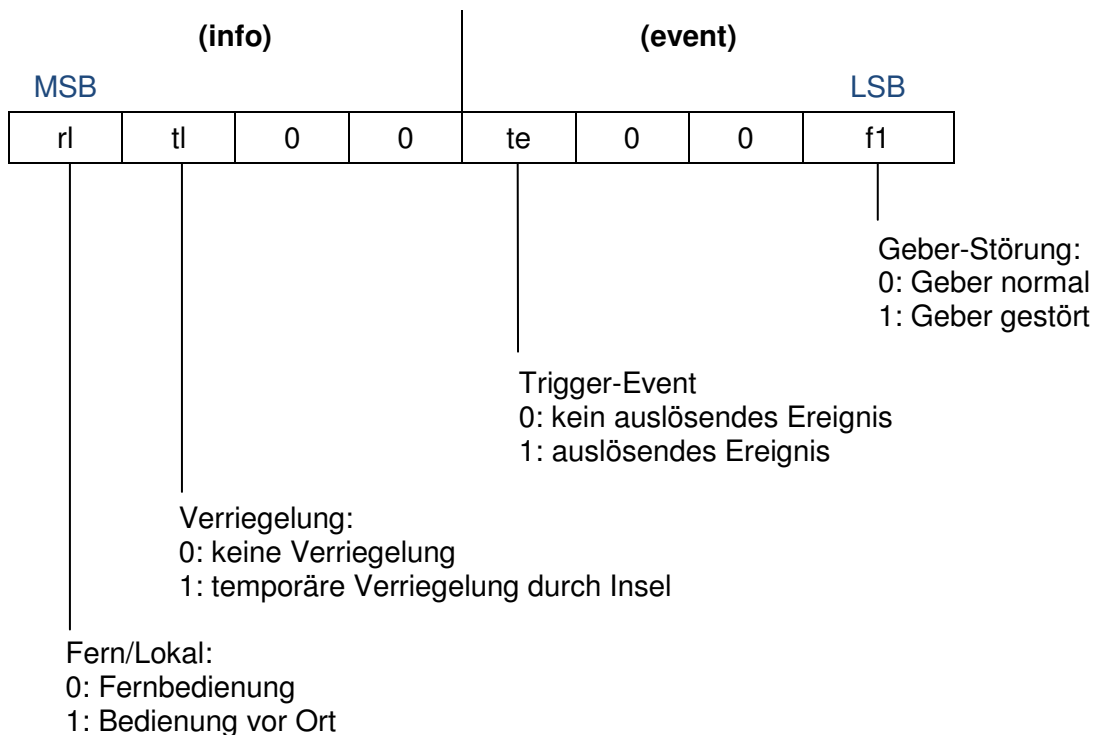


Variablenbeschreibung DP_0:

mask: (P_1)

(Zur Beschreibung dieses Bytes: siehe 3.6 Sperren von Ereignismeldungen)

info_event:



Erläuterungen zu info_event:

f1 (primary fault):

ein gesetztes „f1“-Bit bedeutet eine Störung des Geberteils, die bei virtuellen Datenpunkten als ein erkannter interner Hard- oder Software-Fehler zu interpretieren ist.

Art und Ursache der Störung werden durch den Info-Code („exc“) näher spezifiziert.

te (triggering-event):

3.3.4 Stellpunkt

Ein gesetztes „te“-Bit signalisiert das Eintreten eines projektspezifisch vereinbarten Ereignisses (z.B. den Ablauf eines AI-internen implementierten (zyklischen) Timers), welches die Übertragung der Datenpunkt-Tabelle DP_1 impliziert.

Das Rücksetzen des Bits im info_event_Byte der DP-Tabelle DP_0 erfolgt aufgrund der LZ-seitigen Bestätigung der Meldung bzw. bei Ausbleiben derselben nach einem intern gesetzten Timeout. Der Rücksetzvorgang selbst wird nicht als Ereignis gewertet.

Solange „te“ auf 1 steht, bewirken weitere „te“-bezogene Ereignisse keine Meldung.

Sind Ereignisse dieser Art (AI-oder LZ-seitig) ausgeblendet, (d.h. $msk \geq 1$ oder $mis \geq 1$), so hat „te“ während dieser Zeit den Wert 0.

- tl (temporary lock):
Eine durch die AI gesteuerte, zeitlich begrenzte Verriegelung des Schaltpunktes wird durch das „tl“-Bit angezeigt. Stellbefehle seitens FND werden in diesem Zeitraum abgewiesen.
- rl (remote/local):
Die Zustände Fern-/Vor-Ort-Bedienung werden durch das „rl“-Bit unterschieden

Parameter-Übersicht:

P_	Name	Erläuterung
1	mask	Unterdrückung von Spontanmeldungen aufgrund spezieller Ereignis-

Variablenbeschreibung DP 1:

dimension:

(wie bei Messpunkt, siehe 3.3.1)

actual:

„actual“ enthält die zu „nominal“ gehörende Stellungsmessung in Short-Real-Darstellung gemäß IEEE 754.

Ein Wert von FFFFFFFh (alle Bits gesetzt) markiert einen ungültigen Messwert.

nominal:

„nominal“ enthält den letzten (aktuellen) über FND vorgegebenen Stellwert in Short-Real-Darstellung gemäß IEEE 754.

Parameter-Übersicht:

P_	Name	Erläuterung
1	nominal	Vorgegebener Stellwert

Operations-Übersicht

Stellpunkt		fct_id	
d_r	tab_id	1 (Lesen)	2 (Mod.)
0 (CMD)	0 (DP_0)	C1	C2
	1 (DP_1)	C3	C4
3 (RSP)	0 (DP_0)	R1	R2
	1 (DP_1)	R3	R4
2 (USM)	1 (DP_1)	U1	* entfällt *
1 (ACK)	1 (DP_1)	A1	* entfällt *

Symbol-Erklärung

Operation	Bedeutung	Daten-Teil
C1	DP-Status (DP_0) anfordern	(leer)
C2	Meldungs-Kontroll-Feld (msk in DP_0) mod.	(leer)
C3	DP-Nutzdaten (DP_1) anfordern	(leer)
C4	Stellen	DP_1
R1	DP-Status (DP_0) übermitteln	(leer)
R2	Modifikationsbestätigung des Meldungs-Kontrollfeldes	(leer)
R3	DP-Nutzdaten (DP_1) übermitteln	DP_1
R4	Stell-Quittung	DP_1
U1	Stellmeldung	DP_1
A1	Bestätigung Stellmeldung	leer

Beispiel:

Die LZ möchte den Wert „50%“ vorgeben. Diese Information („nominal“) ist Bestandteil der Nutzdaten in DP-Tabelle 1. Die LZ sendet somit ein CMD-Telegramm C4 und erhält ein RSP-Telegramm R4.

(Für weitergehende Ausführungen zu dieser tabellarischen Übersicht wird auf das Kapitel 3.4.1 „Operationsbeschreibung“ verwiesen).

- Diese Seite ist leer.-

3.3.5 Zählpunkt

Charakterisierung:

Der Zählpunkt ist einer Größe zugeordnet, die im zeitlichen Verlauf monoton wachsenden Charakter hat. Unstetigkeiten entstehen lediglich durch (Rück-) Setzvorgänge oder bei Überlauf des Zählers.

Datenpunkt-Tabellen:

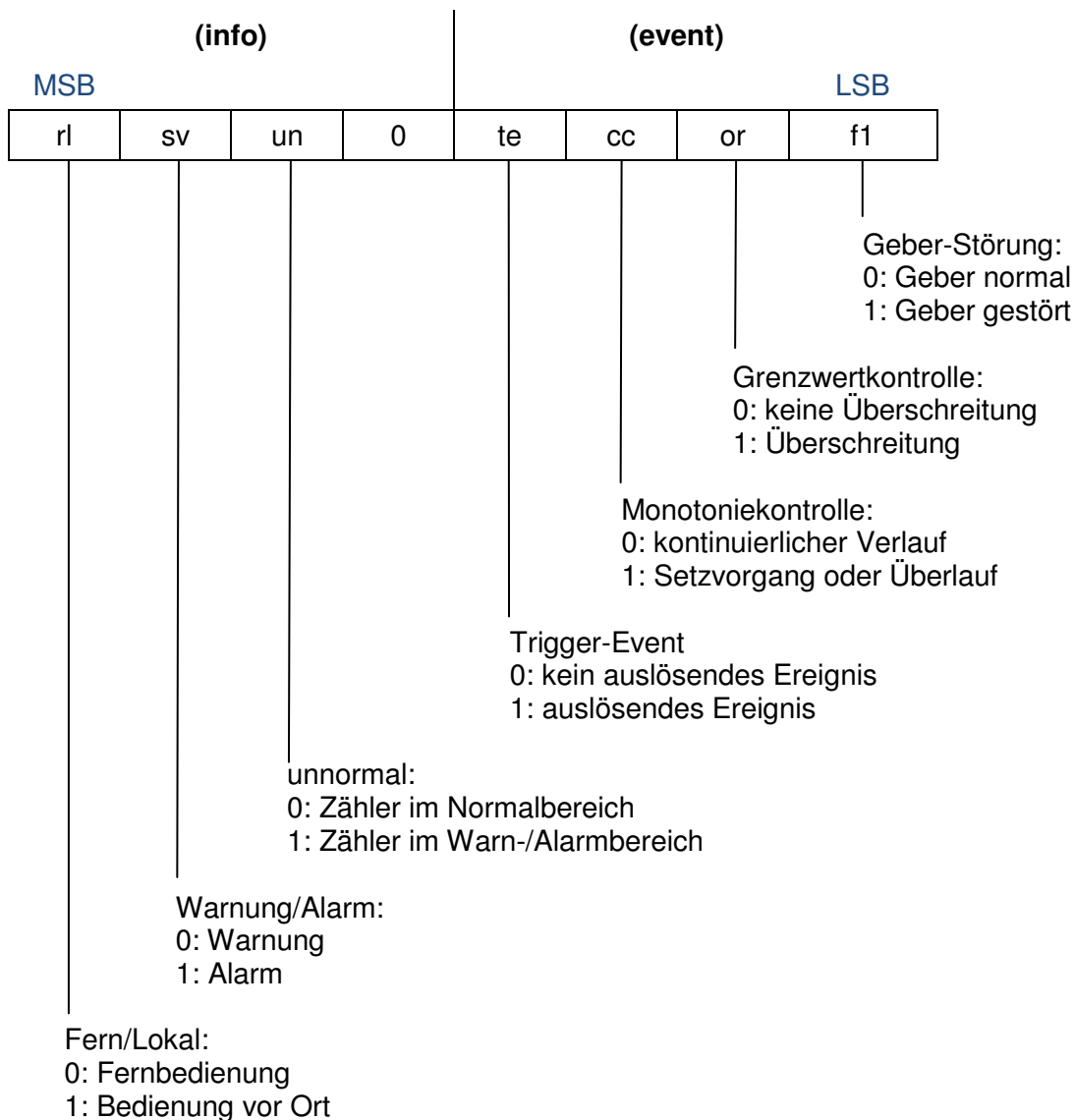
DP_0	
mask	P_1
info_event	

DP_1	
atb_id	
actual /(preset)	P_1
last_save	

DP_2	
update_control	
warning_high	P_1
alarm_high	P_2

Variablenbeschreibung DP 0:**mask: (P_1)**

(Zur Beschreibung dieses Bytes: siehe 3.6 Sperren von Ereignismeldungen)

info_event:**Erläuterungen zu info_event:**

- f1 (primary fault):
 ein gesetztes „f1“-Bit bedeutet eine Störung des Geberteils, die bei virtuellen Datenpunkten als ein erkannter interner Hard- oder Software-Fehler zu interpretieren ist.
 Art und Ursache der Störung werden durch den Info-Code („exc“) näher spezifiziert.
- or (overrange):
 Ein gesetztes „or“-Bit zeigt die Überschreitung einer oberen Warn- oder Alarmgrenze an. Der Wechsel vom Normalbereich in den Überschreibungsbereich wird als Ereignis gewertet. Die Unterscheidung zwischen Warnung und Alarm wird durch das „sv“-Bit ermöglicht.

3.3.5 Zählpunkt

Das „or“-Bit ist nach der LZ-seitigen Bestätigung der Meldung bzw. bei Ausbleiben deselben nach einem intern gesetzten Timeout zurückzusetzen. Der Rücksetzvorgang wird nicht als Ereignis gewertet. Sind Ereignisse dieser Art ausgeblendet, so hat „or“ während dieser Zeit den Wert 0.

cc (continuity control):

Ein gesetztes „cc“-Bit signalisiert einen (Rück-)Setzvorgang oder einen Überlauf des Zählers. Dabei steht der unmittelbar vor dem Ereignis gültige Zählwert in „last_save“, der neue Wert in „actual“.

Rücksetzen des „cc“-Bits erfolgt aufgrund der LZ-seitigen Bestätigung der Meldung, bzw. bei Ausbleiben derselben nach einem intern gesetzten Timeout. Der Rücksetzvorgang wird nicht als Ereignis gewertet.

Solange „cc“ auf 1 steht, bewirken weitere (Rück-)Setz- oder Überlaufvorgänge keine Meldung.

Sind Ereignisse dieser Art (Insel- oder LZ-seitig) ausgeblendet, so hat „cc“ während dieser Zeit den Wert 0.

te (triggering-event):

Ein gesetztes „te“-Bit signalisiert das Eintreten eines projektspezifisch vereinbarten Ereignisses (z.B. den Ablauf eines AI-internen implementierten (zyklischen) Timers), welches die Übertragung der Datenpunkt-Tabelle DP_1 impliziert.

Das Rücksetzen des Bits im info_event_Byte der DP-Tabelle DP_0 erfolgt aufgrund der LZ-seitigen Bestätigung der Meldung bzw. bei Ausbleiben derselben nach einem intern gesetzten Timeout. Der Rücksetzvorgang selbst wird nicht als Ereignis gewertet.

Solange „te“ auf 1 steht, bewirken weitere „te“-bezogene Ereignisse keine Meldung.

Sind Ereignisse dieser Art (AI-oder LZ-seitig) ausgeblendet, (d.h. msk > 1 oder mis > 1), so hat „te“ während dieser Zeit den Wert 0.

un (unnormal):

Das „un“-Bit informiert über eine Grenzwert-Überschreitung, abgestuft in die Kategorien „Warnung“ und „Alarm“ gemäß „sv“-Bit. Es ist zu setzen, falls der Zählwert nicht im Normalbereich, also im Warn- oder Alarmbereich liegt.

sv (severity):

Das „sv“-Bit informiert über den Grad der Grenzwert-Überschreitung, abgestuft in die Kategorien „Warnung“ und „Alarm“.

Die zulässigen Kombinationen der zwei Variablen „or“ und „sv“ sowie deren Bedeutung ist nachstehender Tabelle zu entnehmen:

sv	un	Bedeutung
0	0	Zählwert im Normalbereich
0	1	Zählwert im Warnbereich
1	1	Zählwert im Alarmbereich
1	0	(illegal)

rl (remote/local):

Die Zustände Fern-/Vor-Ort-Bedienung werden durch das „rl“-Bit unterschieden.

Parameter-Übersicht:

P_	Name	Erläuterung
1	mask	Unterdrückung von Spontanmeldungen aufgrund spezieller Ereignisse

Variablenbeschreibung DP 1:**dimension:**

(wie bei Messpunkt, siehe 3.3.1)

actual:

„actual“ enthält den aktuellen Zählwert in IEEE 754-Darstellung; ein Eintrag von FFFFFFFFh markiert die Ungültigkeit des Zählwerts.

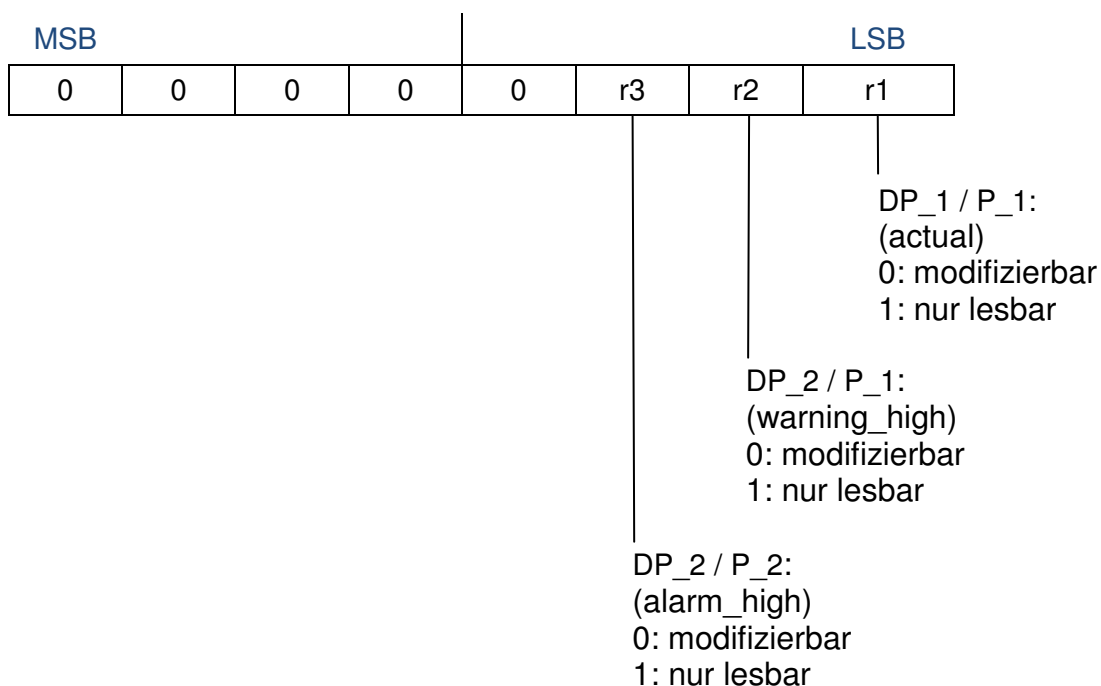
last_save:

Der Wert dieser Variablen, die zu 0 initialisiert wird, enthält den gesicherten Wert unmittelbar vor dem letzten (Rück-)Setzvorgang oder Überlauf des Zählers.

Variablenbeschreibung DP 2:**update_control:**

Dieses Byte stellt eine zusätzliche Modifizierungskontrolle der im Folgenden beschriebenen Parameter P_1 bis P_4 dar. Ein gesetztes Bit bedeutet, dass die LZ nur lesenden Zugriff auf den zugeordneten Parameter hat.

Dies kann relevant sein, wenn ein Parameter des Datenpunkts generell nicht unterstützt wird (angezeigt durch den Wert FFFFFFFFh) oder wenn z. B. nur der obere Grenzwert von der LZ aus verändert werden darf.

update_control:**warning_high: (P_1)**

Obere Warngrenze des Zählwerts in Real-Darstellung. Ein Eintrag von FFFFFFFFh (alle Bits gesetzt) bedeutet die Inaktivität (oder Nichtexistenz) des Grenzwerts.

alarm_high (P_2)

Obere Alarmgrenze des Zählwerts in Real-Darstellung. Ein Eintrag von FFFFFFFFh (alle Bits gesetzt) bedeutet die Inaktivität (oder Nichtexistenz) des Grenzwerts.

Parameter-Übersicht:

Parameter	Name	Erläuterung
DP-Tab. 1 / P_1	actual / (preset)	Vorbesetzung des Zählwerts über FND
DP-Tab. 2 / P_1	warning_high	Obere Warngrenze
DP-Tab. 2 / P_2	alarm_high	Obere Alarmgrenze

Operations-Übersicht

Zählpunkt		fct_id	
d_r	tab_id	1 (Lesen)	2 (Mod.)
0 (CMD)	0 (DP_0)	C1	C2
	1 (DP_1)	C3	C4
	2 (DP_2)	C5	C6
	3 (DP_1&2)	C7	* entfällt *
3 (RSP)	0 (DP_0)	R1	R2
	1 (DP_1)	R3	R4
	2 (DP_2)	R4	R6
	3 (DP_1&2)	R7	* entfällt *
2 (USM)	1 (DP_1)	U1	* entfällt *
1 (ACK)	1 (DP_1)	A1	* entfällt *

Symbol-Erklärung

Operation	Bedeutung	Daten-Teil
C1	DP-Status (DP_0) anfordern	(leer)
C2	Meldungs-Kontroll-Feld (msk in DP_0) mod.	(leer)
C3	DP-Nutzdaten (DP_1) anfordern	(leer)
C4	DP-Nutzdaten (DP_1) modifizieren	DP_1
C5	DP-Parameter (DP_2) anfordern	(leer)
C6	DP-Parameter (DP_2) modifizieren	DP_2
C7	DP-Nutzdaten+Par. (DP_1&2) anfordern	(leer)
R1	DP-Status (DP_0) übermitteln	(leer)
R2	Modifikationsbestätigung des Meldungs-Kontrollfeldes	(leer)
R3	DP-Nutzdaten (DP_1) übermitteln	DP_1
R4	Modifikationsbestätigung Nutzdaten (DP_1)	DP_1 *)
R5	DP-Parameter (DP_2) übermitteln	DP_2
R6	Modifikationsbestätigung Param. (DP_2)	DP_2
R7	DP-Nutzdaten+Par. (DP_1&2) übermitteln	DP_1&2
U1	Zählwertmeldung	DP_1
A1	Bestätigung Zählwertmeldung	(leer)

*) Anhand der auf einen LZ-seitigen (Rück-)Setzbefehl eintreffenden Antwort kann abgelesen werden, ob der Vorgang bereits ausgeführt wurde (der zurückgemeldete „actual“-Wert stimmt mit dem gesendeten „actual / (preset)“-Wert überein) oder erst mit einer zeitlichen Verzögerung als Ereignis (Ausführungsbestätigung in Form einer Spontanmeldung) gemeldet wird. „last_save“ enthält den unmittelbar vor dem Setzvorgang gültigen Zählwert.

Beispiel:

Die LZ möchte nach einem Austausch des mechanischen Zählwerkes den Zählerstand „1.2345“ vorgeben. Diese Information („actual / (preset)“) ist Bestandteil der Nutzdaten in DP-Tabelle 1. Die LZ sendet somit ein CMD-Telegramm C4 und erhält ein RSP-Telegramm R4. Da der Setzbefehl sofort ausgeführt wird, enthält das RSP-Telegramm den bisherigen Zählwert nun im „last_save“, „actual“ enthält 1.2345 und das gesetzte „cc“-Bit zeigt den Setzvorgang an.

Werden über die gesamte Laufzeit der LZ alle Datensätze mit gesetztem „cc“-Bit gespeichert, so kann jederzeit der Gesamtzählwert (z.B. Verbrauch von Wasser, Gas, Wärme, Strom,...) durch Kumulieren der Differenz zwischen „actual“ und dem „last_save“ des nächsten Datensatzes berechnet werden.

(Für weitergehende Ausführungen zu dieser tabellarischen Übersicht wird auf das Kapitel 3.4.1 „Operationsbeschreibung“ verwiesen).

-Diese Seite ist leer.-

3.3.6 Karteipunkt

Charakterisierung:

Mit einem Karteipunkt lassen sich in einer 3-dimensionalen Matrix strukturierte Daten übertragen. Diese Struktur entspricht den Arbeitsmapen in handelsüblichen Tabellen-Kalkulations-Programmen. Karteipunkte dienen z.B. der AI-internen Speicherung von Daten oder zur Vorgabe eines Zeitschaltprogramms.

Wie **Abb. 3.3.6-1** zeigt, basiert die Grundkonzeption des Karteipunkts auf der tabellarischen Datenorganisation. Eine **Tabelle** besteht aus **Zellen**, die der Speicherung von Informationen eines bestimmten, projektspezifisch vereinbarten Datentyps dienen.

Koordinatensystem

Eine **Kartei** ist die Aneinanderreihung von Einzeltabellen, die sich jeweils aus **Zeilen** und **Spalten** zusammensetzen.

Jede Zelle (oder synonym: jeder **Karteipunkt**) wird durch Angabe der Koordinaten **[i,j,k]** für

- die Tabelle (Index „i“)
- die Zeile (Index „j“)
- die Spalte (Index „k“)

eindeutig festgelegt (**adressiert**).

Die Übertragung und Manipulation von Zellen erfolgt in **Blöcken** – quaderartigen Karteiausschnitten, wie **Abb. 3.3.6-2** veranschaulicht. Die Auswahl eines Blocks erfolgt durch Angabe der Koordinaten von zwei sich diagonal gegenüberliegenden Eckpunkten P_0 und P_1 (**Blockdefinitionsunkte**) des ausgewählten Quaders.

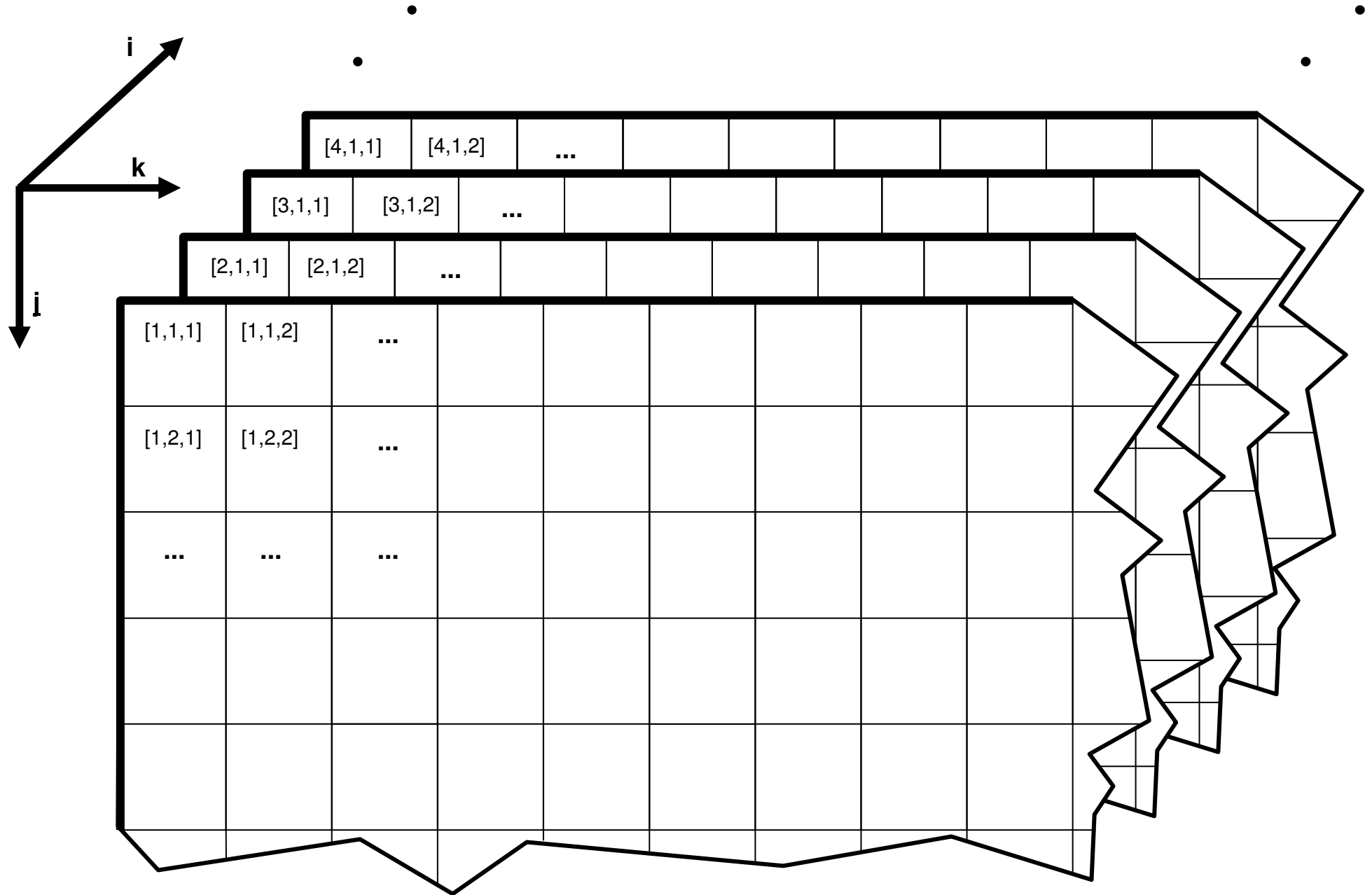


Abb. 3.3.6-1: Kartei-Koordinatensystem [i,j,k]

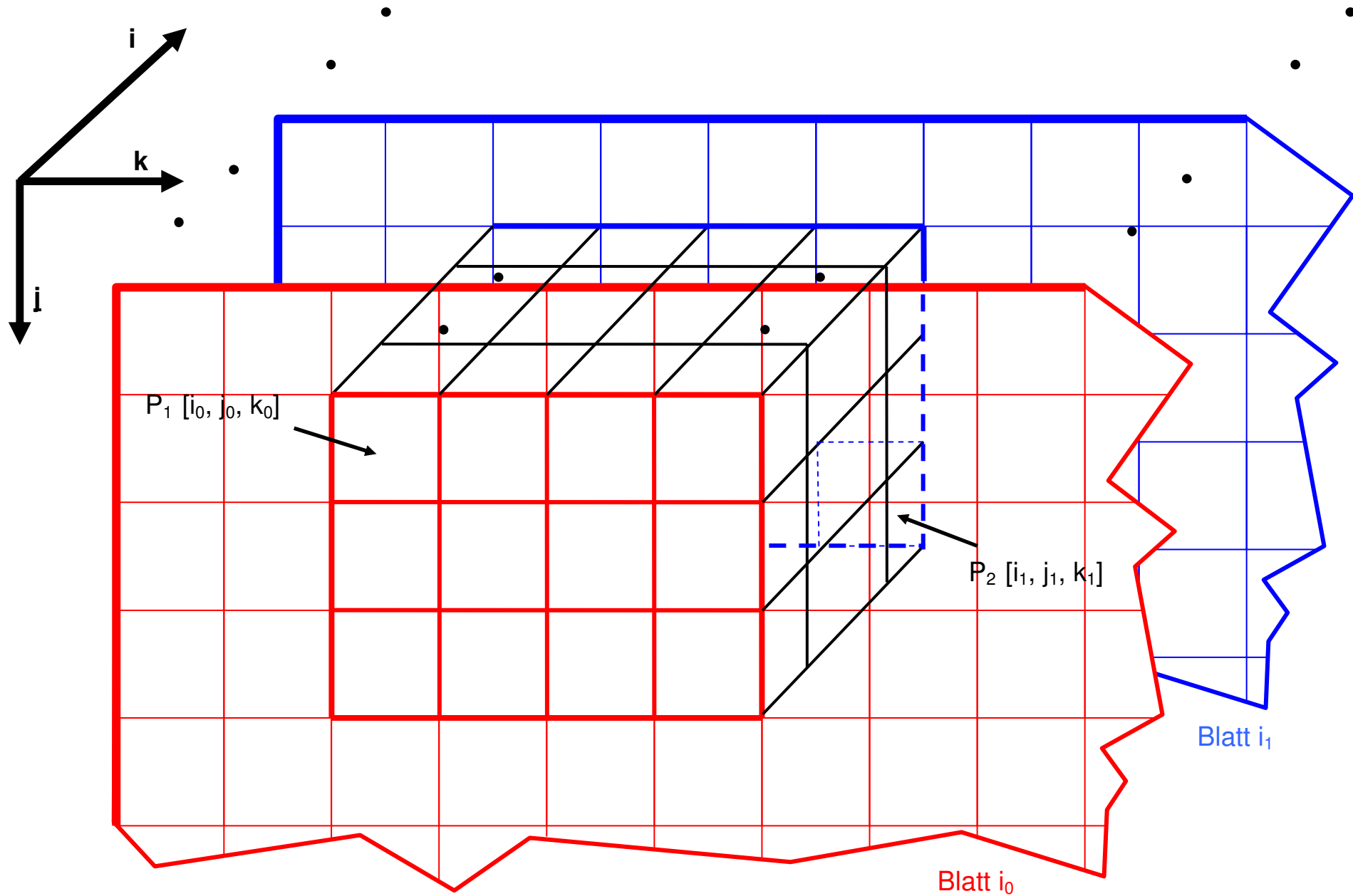


Abb. 3.3.6-2: Auswahl eines Blocks innerhalb einer Kartei

DP_3

mode_i
atb_dim_i
org_i
scal_i
min_i
max_i

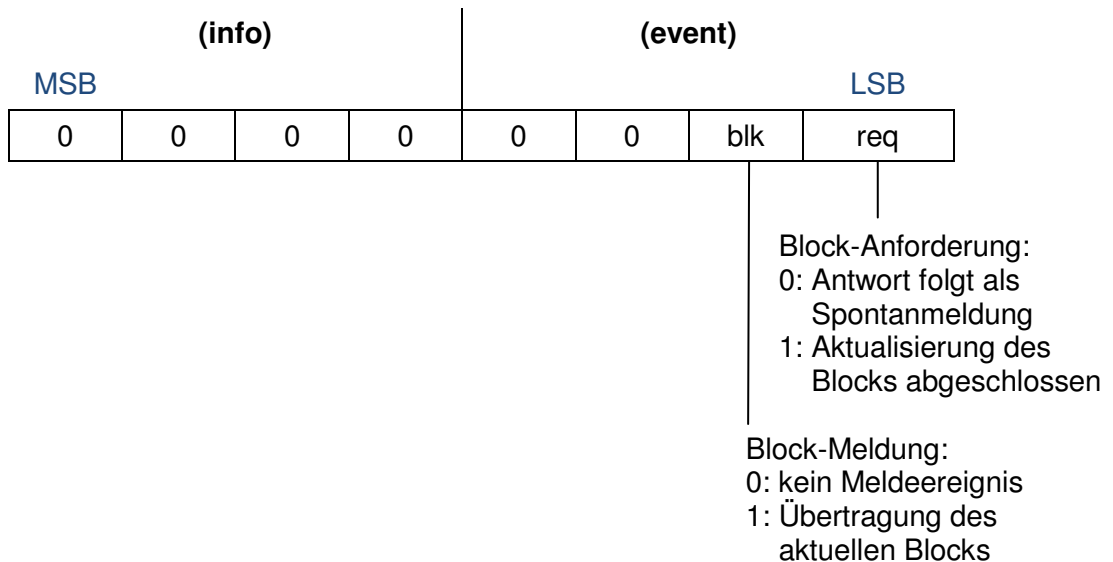
mode_j
atb_dim_j
org_j
scal_j
min_j
max_j

mode_k
atb_dim_k
org_k
scal_k
min_k
max_k

Variablenbeschreibung DP 0:**mask: (P_1)**

(Zur Beschreibung dieses Bytes: siehe 0)

Anzumerken ist, dass zur Unterdrückung aller Karteimeldungen bereits die Bedingung $mis > 0$ oder $msk > 0$ ausreicht (Unterdrückungsstufe = „Betriebsmeldung“).

info_event:**Erläuterungen zu info_event:****req (request):**

Bei der Beschreibung dieses Bits wird im Folgenden zwischen der Antwort auf einen Lesebefehl und einer Meldung unterschieden.

1. Antwort auf einen Lesebefehl (RSP-Telegramm):

Bei der Antwort auf einen Lesebefehl ist der Wert des „req“-Bits davon abhängig, ob der Befehl sofort oder erst nach einer Bearbeitungszeit ausgeführt werden kann.

- a) Ein Wert von 1 bedeutet, dass die aktuellen Werte des selektierten Karteiblocks zur Verfügung stehen und in DP_2 des RSP-Telegramms enthalten sind. Mit dem Absenden des Telegramms wird das „req“-Bit zurückgesetzt.
- b) Ein Wert von 0 bedeutet, dass der Lesebefehl empfangen wurde, der Karteiblock sich aber noch in der Aktualisierungsphase befindet. Der Abschluss der Aktualisierung wird durch Spontanmeldung (s. u.) signalisiert.

2. Spontanmeldung:

Der Abschluss der durch den Lesebefehl initiierten Aktualisierung des Karteiblocks wird durch das gesetzte „req“-Bit signalisiert. Das Rücksetzen des „req“-Bits erfolgt aufgrund der LZ-seitigen Bestätigung der Meldung (ACK) bzw. nach Ablauf des Timeout-Timers.

blk (block-transmission):

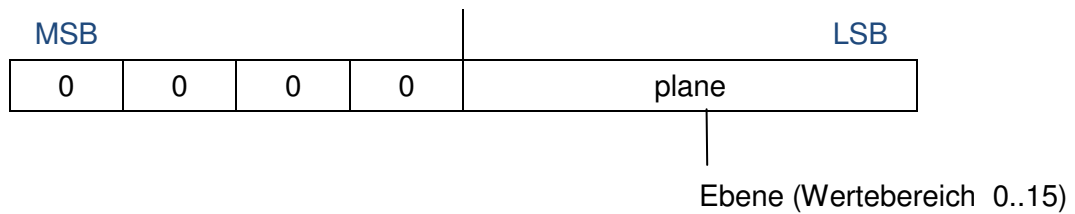
Eine weitere Ereignisquelle für die Übermittlung eines Karteiblocks kann ein AI-intern implementierter Mechanismus (z.B ein Timer) sein. Dabei wird das „blk“-Bit auf 1 gesetzt und der entsprechenden Block zur LZ übertragen. Das Rücksetzen des „blk“-Bits erfolgt aufgrund der LZ-seitigen Bestätigung der Meldung (ACK) bzw. nach Ablauf des Timeout-Timers.

Solange „blk“ auf 1 steht, bewirken weitere „blk“-bezogene Ereignisse keine Meldung. Sind Ergebnisse dieser Art (AI- oder LZ-seitig) ausgeblendet ($msk > 0$ oder $mis > 0$), so hat „blk“ während dieser Zeit den Wert 0.

3.3.6 Karteipunkt

Variablenbeschreibung DP 1:

sel_ctl (selection-control):



Dieses „Auswahl-Kontroll-Byte“ enthält als Parameter die durch 4 Bit dargestellte Variable „plane“ (Wertebereich 0 - 15), deren Bedeutung weiter unten im Abschnitt 3.2.4, „**Kartei-Informationsebenen**“, erläutert wird.

first_i, first_j, first_k und last_i, last_j, last_k :

Tabellen-, Zeilen- und Spaltenindex des ersten und letzten Blockdefinitionspunktes in IEEE-754-Darstellung.

Ein Wert von FFFFFFFFh für first_i,.., last_k bedeutet „nicht angegeben“ (implizit).

Variablenbeschreibung DP 2:

block:

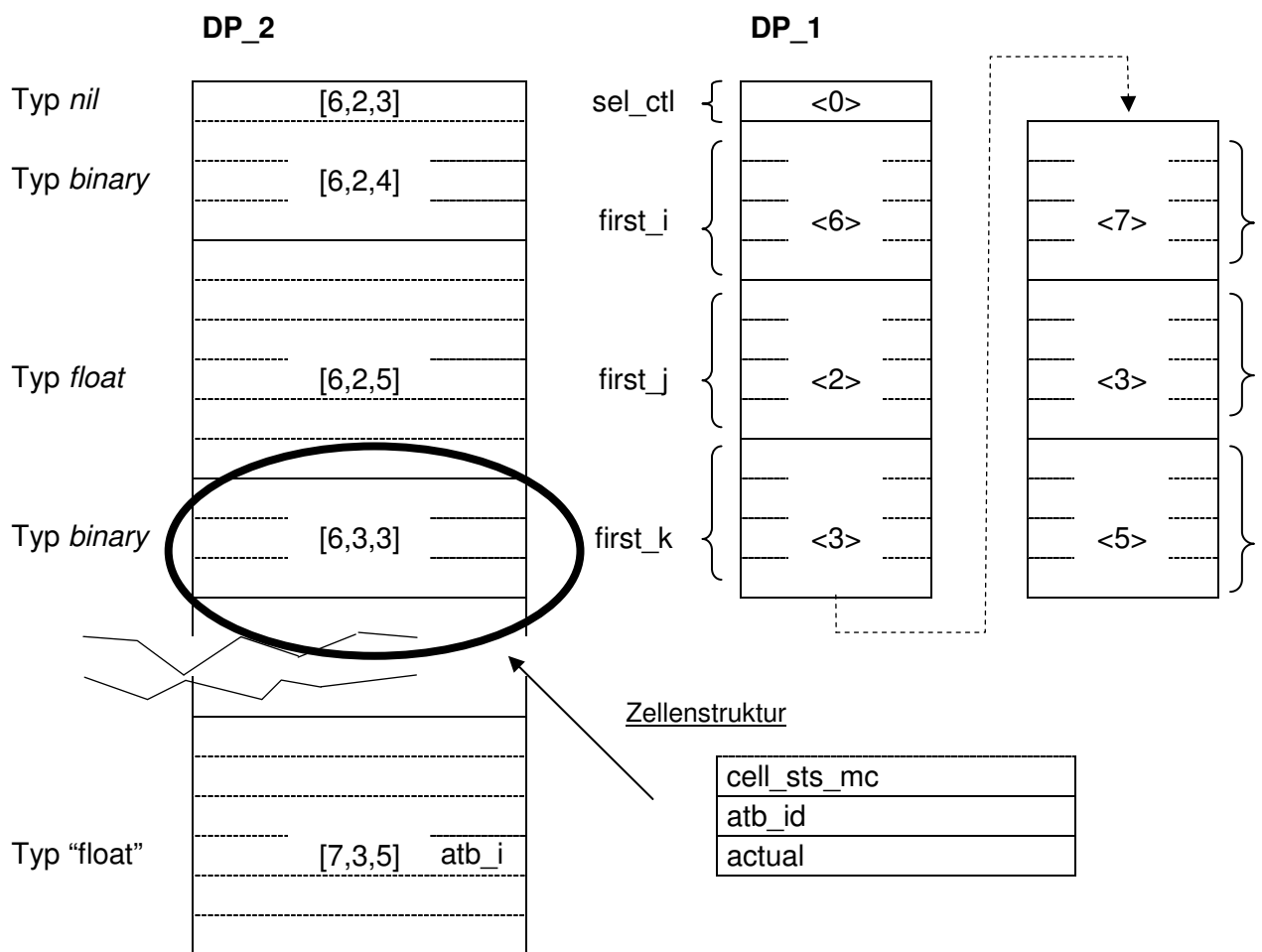
DP_2 ist als „virtuelle dynamische“ Tabelle aufzufassen, d.h. sie wird nicht statisch im Speicher gehalten, sondern gemäß dem benötigten Karteiausschnitt (Block) dynamisch zusammengestellt.

Die serielle Übertragung eines Blocks erfolgt durch Verkettung der einzelnen selektierten Zellen gemäß (der für Matrizen üblichen) Anordnungsregel $k \Rightarrow j \Rightarrow i$, also „Spaltenindex vor Zeilenindex vor Tabellenindex“

Das in Abb. 3.3.6-4 gegebene Beispiel verdeutlicht sowohl die Konzeption der Blockauswahl als auch die Übertragungsreihenfolge. Der ausgewählte Karteiausschnitt ist :

- [6,2,3], [6,2,4], [6,2,5] (Tabelle 6)
- [6,3,3], [6,3,4], [6,3,5] (Tabelle 6)
- [7,2,3], [7,2,4], [7,2,5] (Tabelle 7)
- [7,3,3], [7,3,4], [7,3,5] (Tabelle 7)

Diese Auswahl spiegelt sich in DP_1 durch die Angabe der „Blockeckpunkte“ [6,2,3] und [7,3,5]. DP_2 enthält die in den einzelnen Zellen enthaltenen Informationen in verketteter Form gemäß der „Spaltenindex-vor-Zeilenindex-vor-Tabellenindex“-Regel.



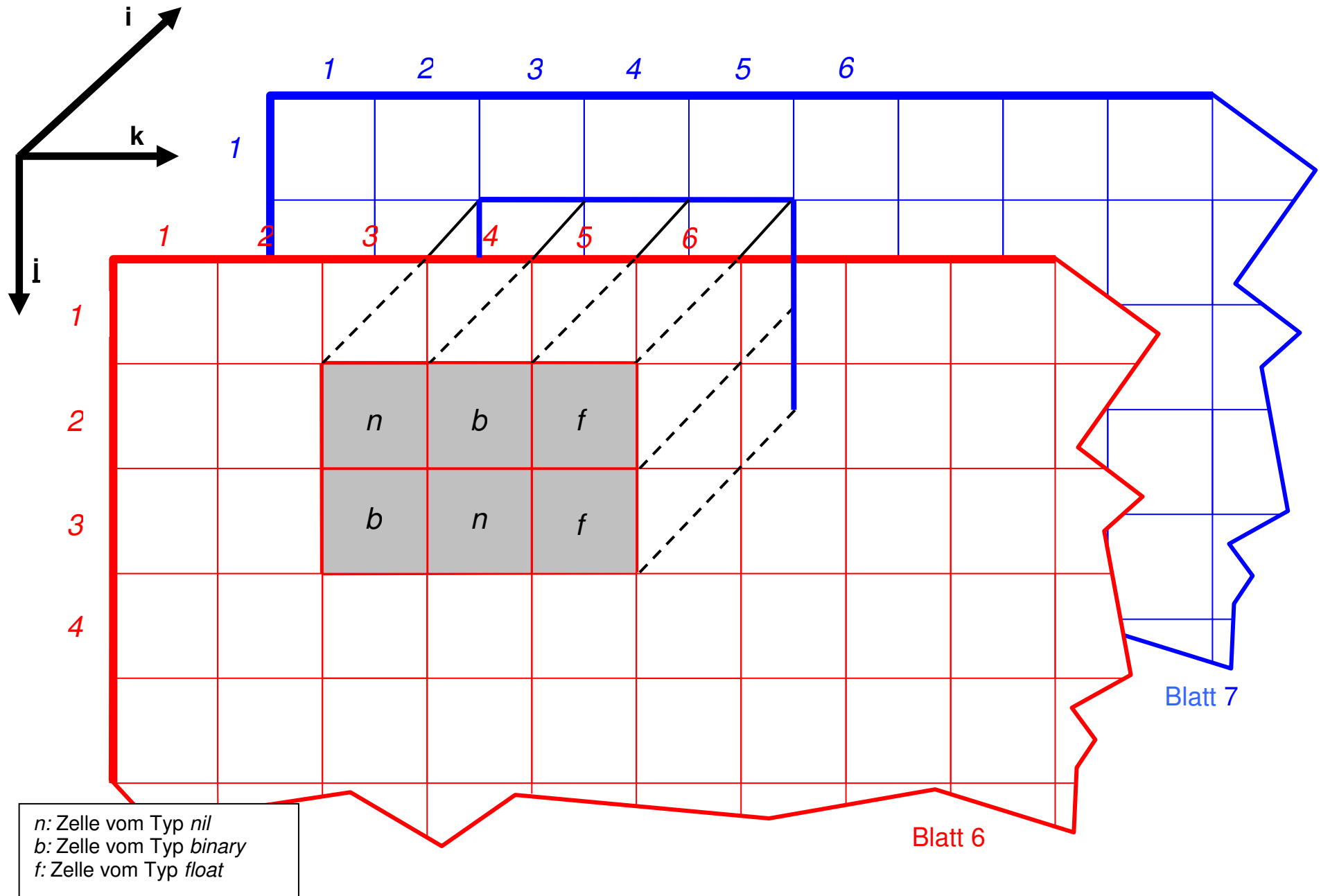


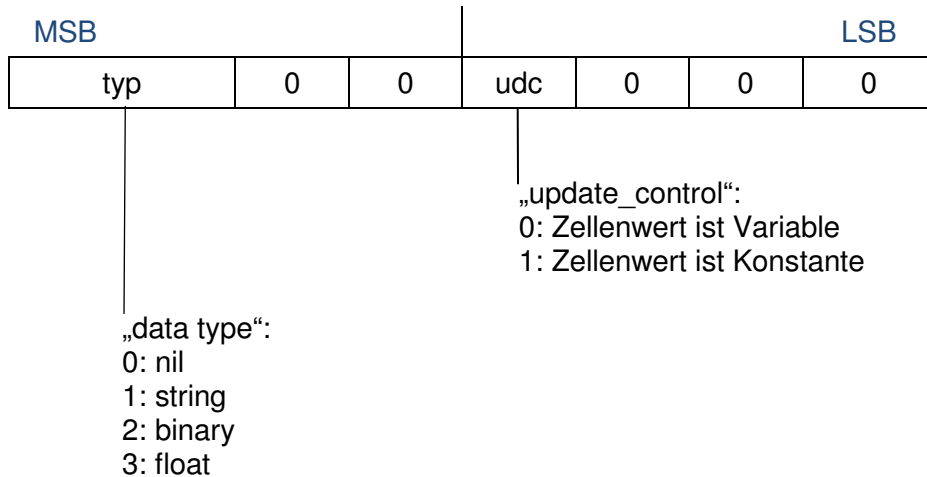
Abb. 3.3.6-3 Beispiel eines Karteiausschnitts (Block)

Zellenstruktur

Jeder Zelle eines Karteipunkts sind beim **RSP/USM/ACK-Telegramm** zwei Größen zugeordnet (vgl. Zelle [6,3,3] in DP_2 in Abb. 3.3.6-4):

- der Zellen-Status („cell_sts_mc“) und
- der Zellen-Wert (bestehend aus „atb_id“ und „actual“)

cell_sts_mc (RSP/USM/ACK-Telegramm)



udc (update-control):

Dieses Bit stellt eine Modifizierungskontrolle für den Zelleninhalt dar. Ist das Bit gesetzt, so weist es den Zelleninhalt als Konstante (d.h. als projektspezifisch festgelegte, nur durch Systemrekonfiguration veränderbare Größe) aus. Andernfalls ist der Zellenwert seitens LZ oder AI (bei vorhandener Zugriffsberechtigung) modifizierbar.

typ (data-type):

diese Bitkombination weist den mit der Zelle verbundenen (projektspezifisch festgelegten) Datentyp gemäß nachstehender Tabelle 3.3.6-1 aus.

Mit einer Zelle des Typs **nil** ist kein Wert verbunden; leere Zellen entstehen i.d.R. als „Lücken“ in der Anordnung der „Nutzinformationszellen“.

Eine Zelle des Typs **string** enthält eine Sequenz von maximal 255 Bytes, der ein Byte mit der entsprechenden Längenangabe vorangestellt ist. Die leere Zeichenkette („Null-String“) besteht folglich nur aus einem Byte des Inhalts 0.

Die Codierung der Datentypen **binary** und **float** durch die kombinierte Angabe von

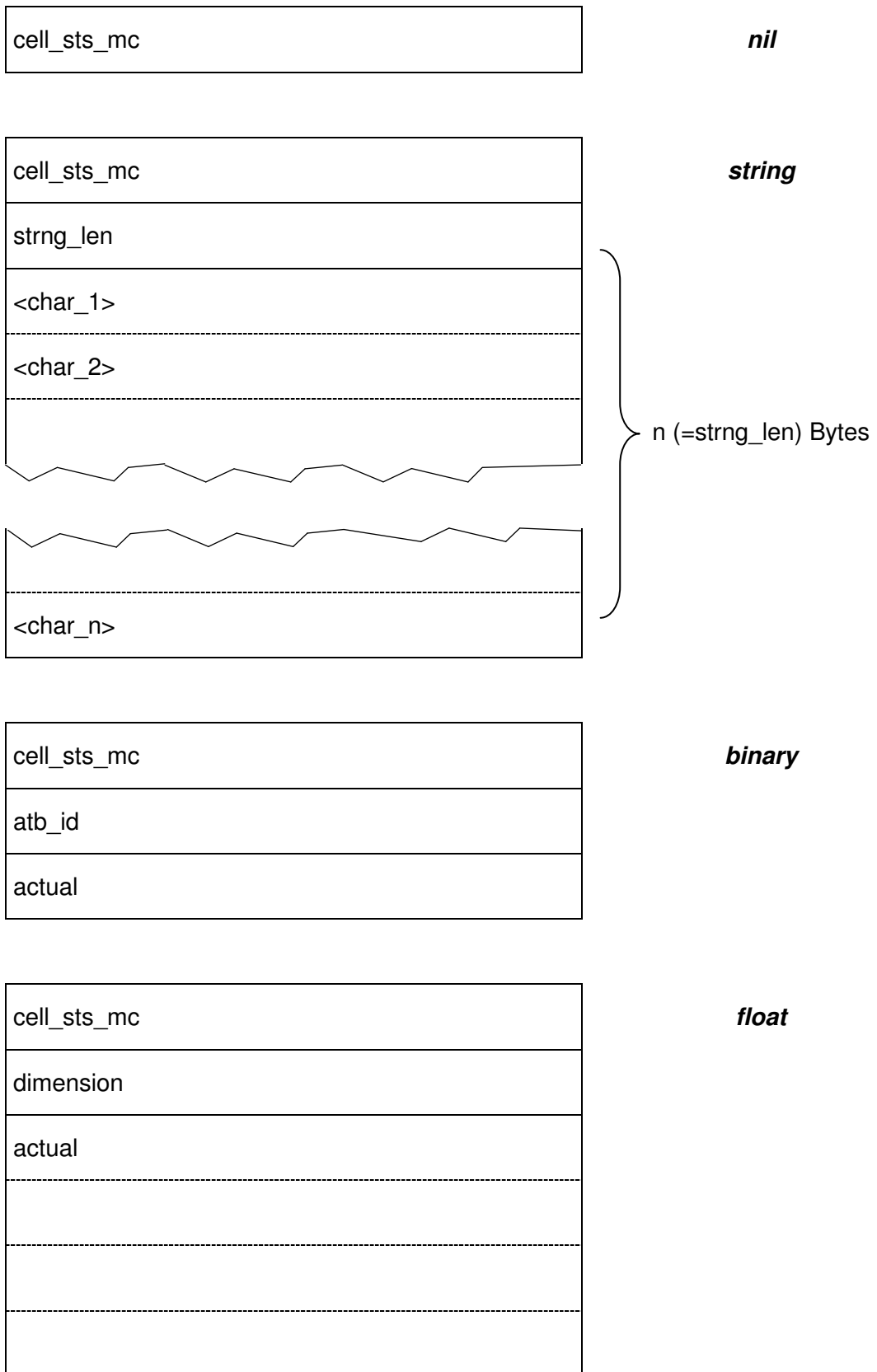
„atb_id“ + „actual“ (binäres Attribut wie bei DP_1, Meldepunkt)

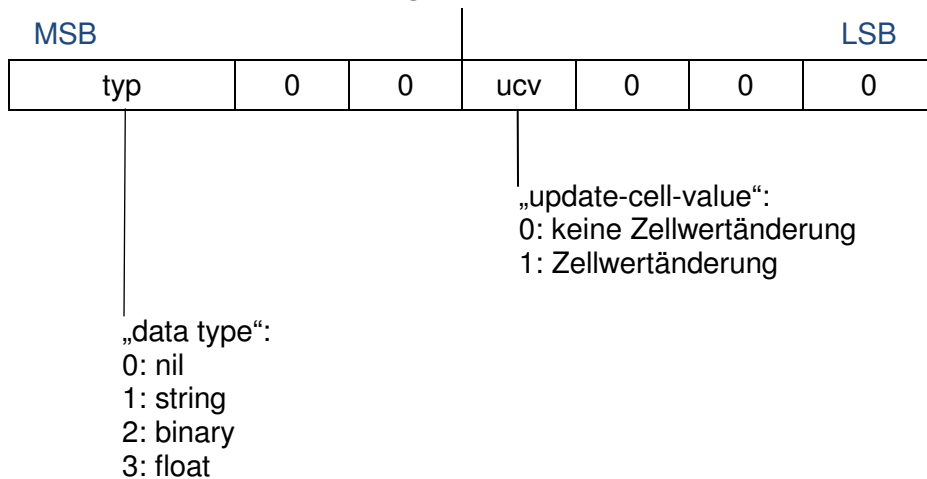
„dimension“ + „actual“ (physikalische Dimension wie bei DP_3, Messpunkt)

Zellen des Typs nil, string mit Inhalt „Nullstring“, binary mit Inhalt FFh, float mit Inhalt FFFFFFFFh heißen unbenutzt („unused“).

Das für die einzelnen Datentypen im DP_2 verwendete Format zur Darstellung der Zelleninhalte bei USM-/ACK- und RSP-Telegrammen ist nachfolgend dargestellt.

3.3.6 Karteipunkt



Zellen-Modifikations-Kontrolle**cell_sts_mc (beim CMD-Telegramm)****ucv (update cell value):**

Ist „ucv“ gesetzt, wird der Zellenwert entsprechend den im CMD-Telegramm nachfolgenden Bytes überschrieben. Ist „ucv“=0, folgen keine weiteren Bytes auf cell_sts_mc. Der Datentyp kann nicht verändert werden.

typ (data-type):

(s. Beschreibung beim Zellen-Status)

Kartei-Informationsebenen

Der Inhalt einer Zelle im Karteipunkt kann aus einem der vier Basis-Datenpunkttypen

nil, *string*, *binary*, *float* bestehen. Darüber hinaus besteht die Möglichkeit, dass einer Zelle mehrere Informationen unterschiedlichen Charakters zugeordnet werden. Beispielsweise könnte eine Zelle folgende zusammengehörige Daten enthalten:

- der Wert einer physikalische Meßgröße
- der Klartextname zur „technischen Zellen-Adresse“ [i,j,k)
- die „Berechnungsformel“ des Zellenwertes
- eine „Triggerformel“ zur Codierung eines Zellenereignisses.

Zur Unterstützung solcher „vektorartiger“ Informationen dient das Konzept der Kartei-Informationsebenen, dessen Prinzip in Abb. 3.3.6-8 dargestellt ist. So könnte – anhand des eben gegebenen Beispiels jeder Zelle

- auf Ebene 0 der Zellenwert,
- auf Ebene 1 das Zellenattribut,
- auf Ebene 2 die Berechnungsformel
- usw.

zugeordnet sein.

Die Auswahl der gewünschten Ebene erfolgt durch die bereits bei der Beschreibung der Datenpunkttable DP_1 erwähnten Variablen „plane“ im „sel_ctl“-Byte.

Die eindeutige Dekodierung eines übertragenen Zellblock-Strings auf der Empfängerseite wird durch Angabe der Blockdefinitionsunkte [first_i, first_j, first_k] und [last_i, last_j, last_k] und die im Zellen-Status (bzw. Zellen-Modifikations-Kontrollbyte) enthaltene Information bezüglich des Datentyps (und damit der Länge des nachfolgenden Zelleninhalts) sichergestellt.

Anmerkung zur Segmentierung:

3.3.6 Karteipunkt

Bei einem segmentierten, DP_2 modifizierenden CMD-Telegramm enthält das zum CMD-Endtelegramm gehörende RSP-Endtelegramm bei

- info_event.req = 1 (Aktualisierung des Blocks abgeschlossen)
die (evtl. segmentierte) Antwort auf den Modifikationsbefehl
- info_event.req = 0 (Antwort folgt als Spontanmeldung)
nur die letzte Kurzquittung. Die Modifikationsbestätigung erfolgt dann
in Form einer Meldung mit info_event.req = 1.

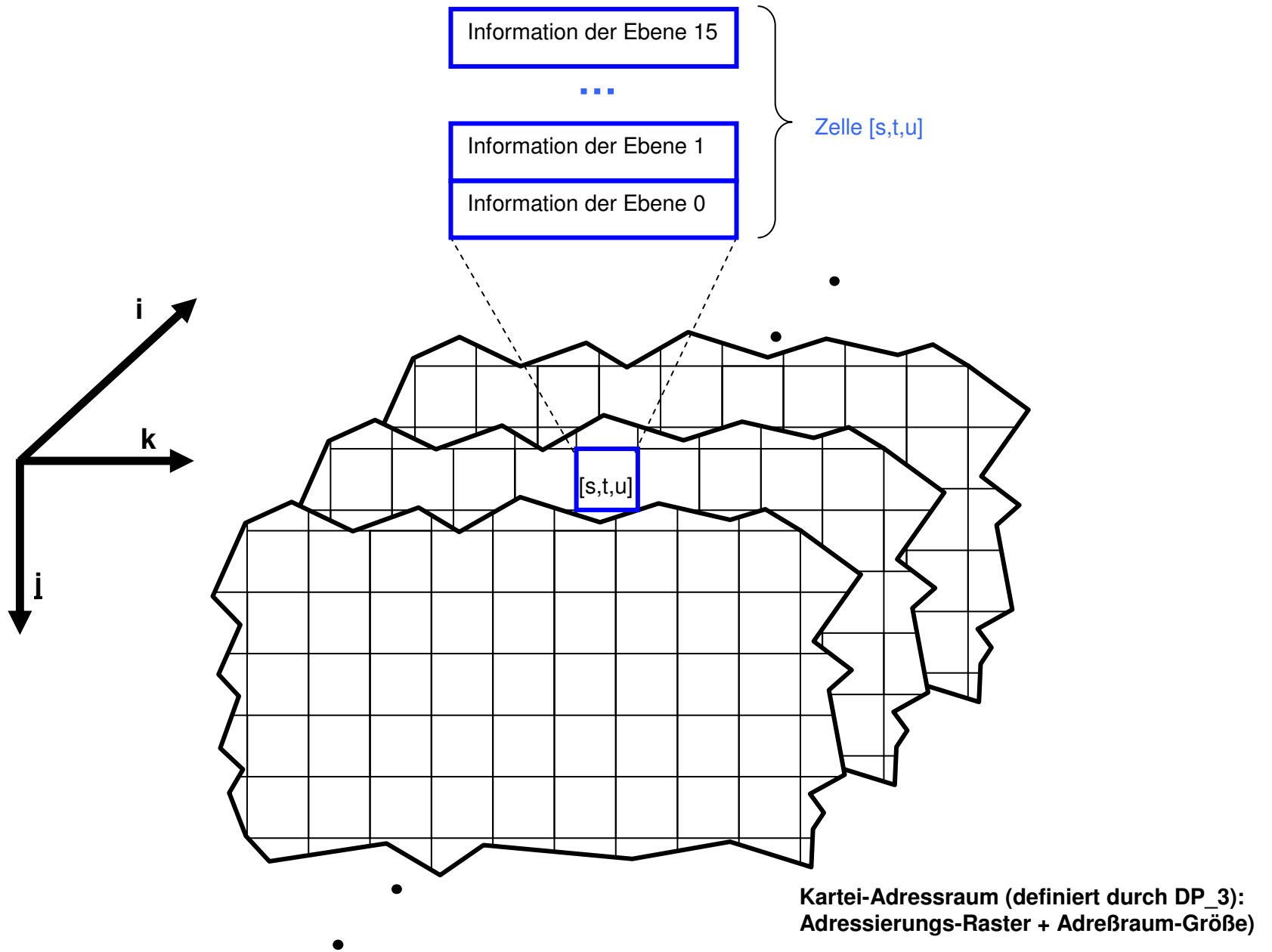


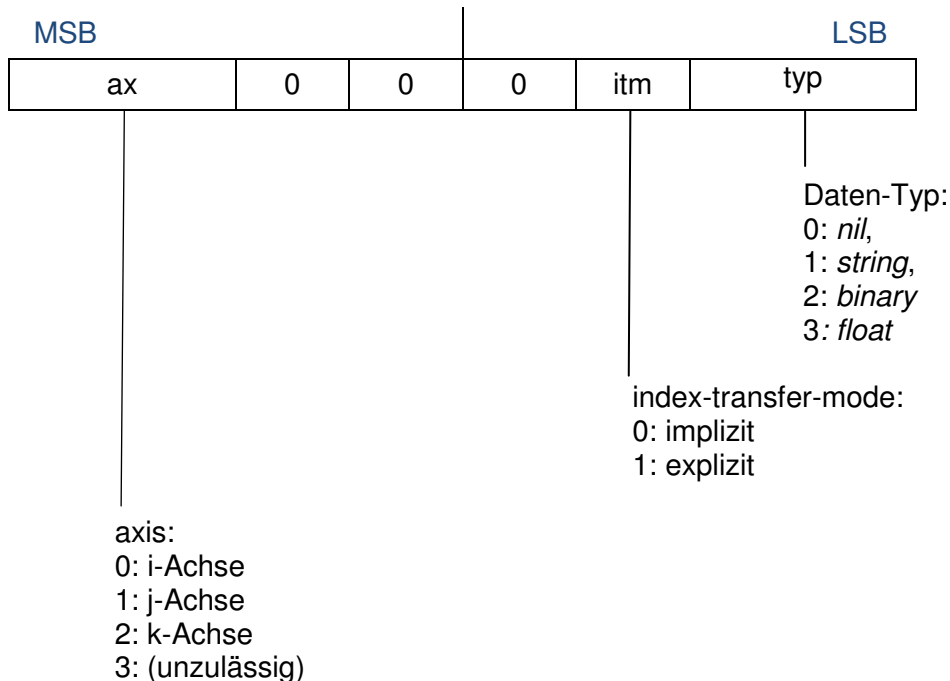
Abb. 3.3.6-4: Konzept der Kartei-Informationsebenen

-Diese Seite ist leer.-

Variablenbeschreibung DP 3:

Die Konfigurationstabelle DP_3 enthält die (projektspezifisch festgelegten) Parameter des [i,j,k] - Koordinatensystems.

mode_i, mode_j, mode_k:



„**ax**“ legt fest, welcher Koordinate (i, j, oder k) die nachfolgenden Bytes („atb_dim_“, „org_“, „scal_“, „min_“ und „max_“) zuzuordnen sind.

Ein „**itm**“-Wert von 1 gibt an, dass die Indexwerte explizit übertragen werden. 0 bedeutet, dass der Index aus der Stellung des Werts in der Datenabfolge mit Hilfe der Parameter „org_“ und „scal_“ errechnet wird und nicht explizit übertragen wird.

„**typ**“ gibt zusammen mit „atb_dim_“ den Datentyp und die Dimension des Achsenindex an. Diese können z.B. Schaltzustände (0,1,2,3) oder die Wochentage (Montag - Sonntag) sein.

Als Index zur Unterscheidung der einzelnen Tabellen, Zeilen und Spalten können auch reelle Zahlen (typ = 3) herangezogen werden. So ist es z.B. möglich, Datum und Uhrzeit auf eine Koordinatenachse zu legen (z. B. mit atb_dim_i=159).

atb_dim_i, atb_dim_j, atb_dim_k :

binäres Attribut („atb_id“) bzw. physikalische Dimension („dimension“), vgl. C Attribute und Dimensionen.

atb_dim_i/j/k = 0 kennzeichnet den dimensionslosen Fall.

org_i, org_j, org_k:

Durch das Zahlentripel [org_i, org_j, org_k] wird bei die „Ursprungszelle“ (= Nullpunkt) des [i,j,k]-Koordinatensystems fixiert.

scal_i, scal_j, scal_k:

Die Skalierung („Auflösung“) der Werte einer Achse i/j/k wird bei durch den nichtnegativen Wert „scal_i/j/k“ als inkrementelle Einheit des Index festgelegt.

So bedeutet z. B. ein Wert von „scal_i“ = 1 eine ganzzahlige Tabellenindizierung und „scal_k“= 10 eine Spaltenindexschrittweite von 10 Einheiten.

Sind als Tabellen-/Zeilen-/Spaltenindizes reelle Zahlen zugelassen, d.h. es gilt („mode_i/j/k.typ“=3), so erfolgt die Indizierung explizit, d. h. für die betroffene Achse gilt itm=1 und für den Skalierungswert „scal_i/j/k“ = 0.

**min_i, min_j, min_k,
max_i, max_j, max_k:**

Die projektspezifisch festgelegten Größen „min_i/j/k“ und „max_i/j/k“ markieren den prinzipiell zur Verfügung stehenden [i,j,k]-Adressraum als Block mit den Definitionspunkten [„min_i“, „min_j“, „min_k“] und [„max_i“, „max_j“, „max_k“].

Die Werte $-\infty$ (für „min_i/j/k“) und $+\infty$ (für „max_i/j/k“) sind zugelassen

Operations-Übersicht

Karteipunkt		fct id	
d_r	tab_id	1 (Lesen)	2 (Mod.)
0 (CMD)	0 (DP_0)	C1	C2
	1 (DP_1)	C3	* entfällt *
	3 (DP_1&2)	C4	C5
	4 (DP_3)	C6	* entfällt *
3 (RSP)	0 (DP_0)	R1	R2
	1 (DP_1)	R3	* entfällt *
	3 (DP_1&2)	R4	R5
	4 (DP_3)	R6	* entfällt *
2 (USM)	3 (DP_1&2)	U1	* entfällt *
	4 (DP_3)	U2	* entfällt *
1 (ACK)	3 (DP_1&2)	A1	* entfällt *
	4 (DP_3)	A2	* entfällt *

Symbol-Erklärung

Operation	Bedeutung	Daten-Teil
C1	DP-Status DP_0 anfordern	(leer)
C2	Meldungs-Kontrollfeld modifizieren	(leer)
C3	„Used-Block“-Information (DP_1) anfordern	(leer)
C4	Block DP_2 (gemäß Auswahl DP_1) anfordern	DP_1
C5	Block DP_2 (gemäß Auswahl DP_1) modifizieren	DP_1&2
C6	Konfigurationstabelle DP_3 anfordern	(leer)
R1	DP-Status DP_0 übermitteln	(leer)
R2	Modifikationsbestätigung des Meldungs-Kontrollfelds	(leer)
R3	„Used-Block“-Information (DP_1) übermitteln	DP_1
R4	Block DP_2 zusammen mit Auswahl DP_1 übermitteln	[DP_1&2] ^{*)}
R5	Modifikationsbestätigung der DP-Tabelle DP_2 zus. mit Auswahl DP_1	[DP_1&2] ^{*)}
R6	DP-Tabelle DP_3 übermitteln	DP_3
U1	Meldung des Blocks DP_2 zusammen mit Auswahl DP_1	DP_1&2
U2	Meldung einer Kartei-Konfigurationsänderung (DP_3)	DP_3
A1	Bestätigung der Blockmeldung	(leer)
^{*)} Falls Datenteil leer (info_event.req = 0) erfolgt die Übermittlung von DP_1&2 (mittels info_event.req = 1) später optional in Form einer Spontanmeldung		

Erläuterung zu C3/R3:

Zur Ermittlung des „kleinsten Blocks, außerhalb dessen alle Zellen unbenutzt sind“ in der interessierenden Kartei-Ebene (plane) sendet die LZ einen DP_1-Lesebefehl (C3) auf eine virtuelle Tabelle namens „Used-Block“. Die IZ bestimmt daraufhin diesen Block und übermittelt die Koordinaten mit einem DP_1-Telegramm (R3) an die LZ.

Explizite Indexangabe

Bei reellwertig Achsenindizes („mod_i/j/k.typ“ = 3 und „scal_i/j/k“ = 0) ist eine implizite Indizierung nicht sinnvoll, da in aller Regel nur sehr wenige Zellen zwischen „first_used_i/j/k“ und „last_used_i/j/k“ belegt sind. Um eine effiziente Kommunikation zu gewährleisten wird hier die explizite Indexübertragung genutzt (gesetzte(s) „itm“-Bit(s) der Variablen „mode_i/j/k“ in DP_3). Dabei werden nur die explizit gespeicherten Zellen bzw. Zellbereiche übertragen.

In diesem Transfermodus müssen im Gegensatz zur impliziten Blockübertragung die Index-Informationen mitgeliefert werden, da diese nicht aus „first_i/j/k“ und „scal_i/j/k“ ermittelt werden können.

Dies geschieht dadurch, dass bei der sequentiellen Blockübertragung

Bei „mode_i.itm“ = 1: dem Tabelleninhalt der Tabellenindex

bei „mode_j.itm“ = 1: dem Zeileninhalt der Zeilenindex

bei „mode_k.itm“ = 1: der Spaltenindex dem Zelleninhalt

vorangestellt wird.

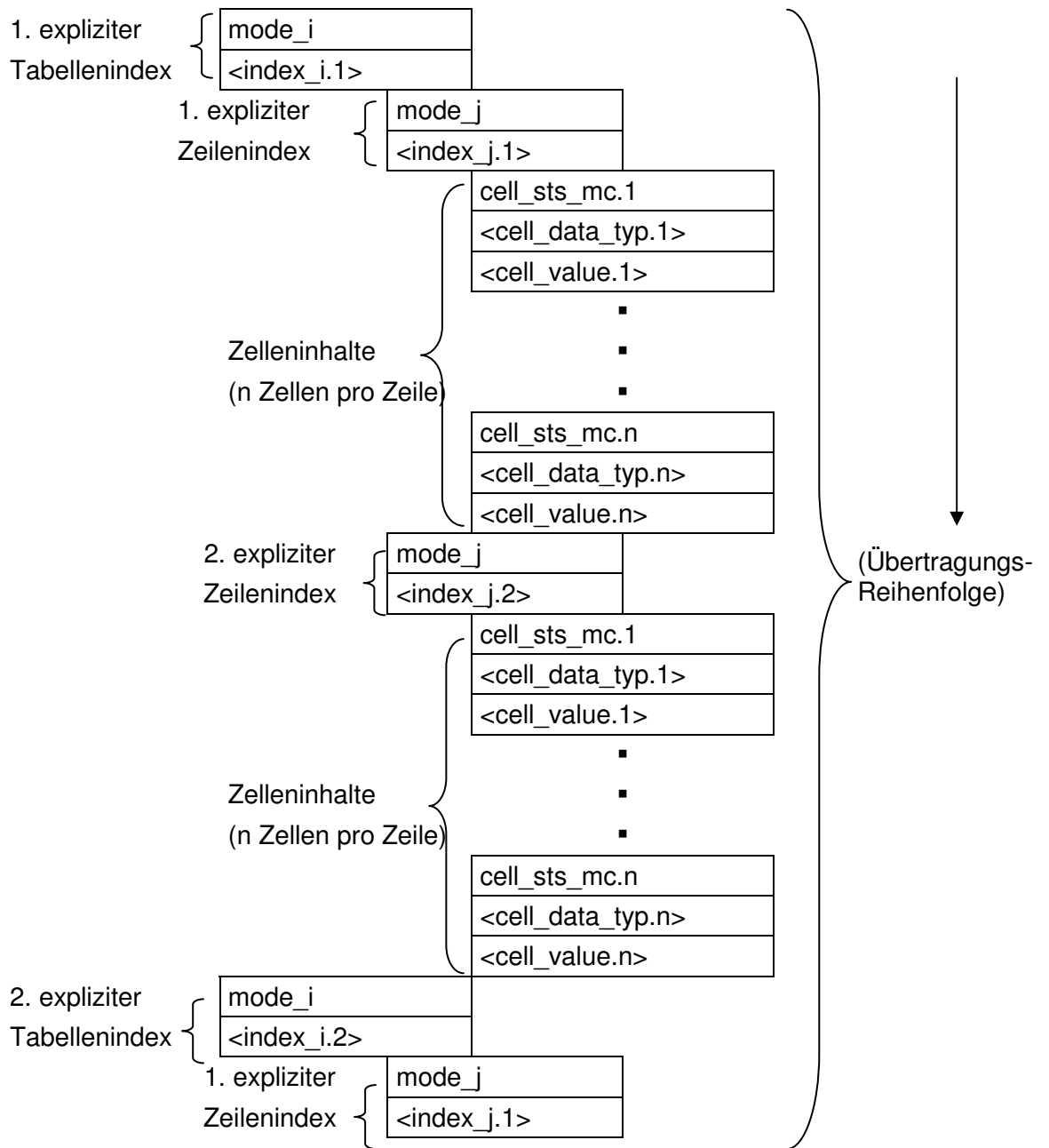
Dabei ist zu beachten, dass zur Sicherstellung der eindeutigen Adressenzuordnung die Übertragungsmodi „mode_i/j/k“ nur entsprechend der nachfolgenden Tabelle gewählt werden dürfen:

	mode_i	mode_j	mode_k
itm	0	0	0
	1	0	0
	1	1	0
	1	1	1
andere Kombinationen: illegal			

Nachfolgend wird die Blockübertragung mit expliziter Indexangabe beispielhaft veranschaulicht. Dabei soll gelten:

Spaltenzahl (= n)	für alle Zeilen und Tabellen konstant
„mode_i.itm“ = 1	(explizite Indexangabe, da Tabellen selektiert)
„mode_j.itm“ = 1	(explizite Indexangabe, da Zeilen selektiert)
„mode_k.itm“ = 0	(implizite Indexangabe, da Spaltenzahl (= n) konstant)

3.3.6 Karteipunkt



Konkretes Beispiel:

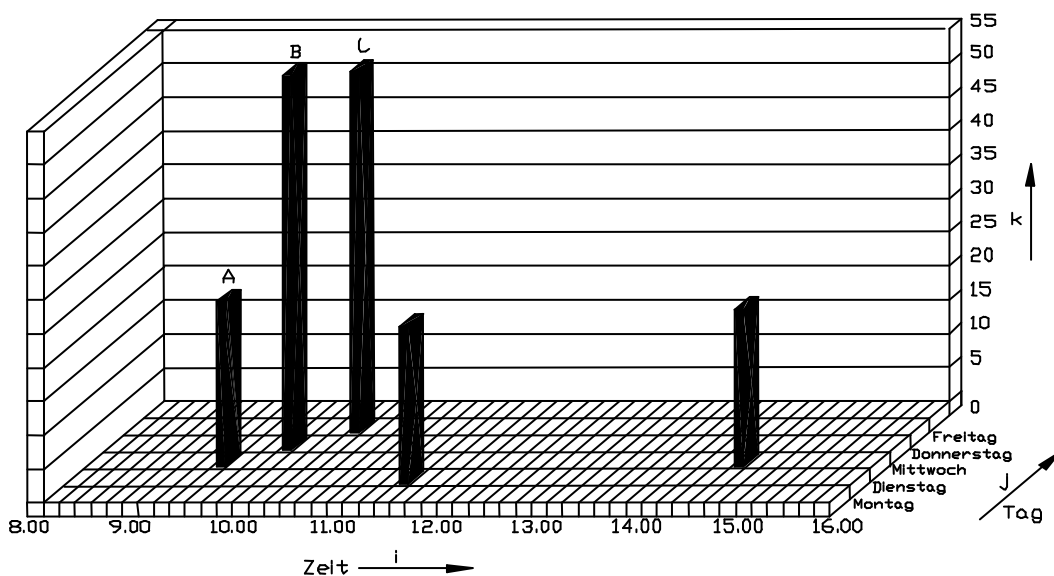
Mit einem Karteipunkt werden die im Verlauf einer Woche von 55 Sensoren registrierten Störmeldungen erfasst. Die Achsenzuordnung ist wie folgt festgelegt:

- i - Achse: kontinuierlicher Zeitbezug innerhalb des aktuellen Tages (8^{00} bis 16^{00})¹
 „mode_i.typ“=3; „atm_dim_i“=167 (Stunde relativ zum aktuellen Tag);
 „org_i“=0; „scal_i“=0; „min_i“=8; „max_i“=16
- j - Achse: Tag innerhalb der aktuellen Woche (Montag - Freitag)
 „mode_j.typ“=2; „atm_dim_j“=165 (Wochentag);
 „org_j“=1; „scal_j“=1; „min_j“=1; „max_j“=7
- k - Achse: laufende Nummer der Sensoren
 „mode_k.typ“=3; „scal_k“=1; „atm_dim_k“=9 (dimensionslos)
 „org_k“=0; „scal_k“=1; „min_k“=1; „max_k“=7

Die Zellen können einen binären Wert (cell_sts_mc.typ=2) mit einem der Attribute „NORMAL“ bzw. „ALARM“ („atb_id“=27) enthalten.

Die Gesamtheit der in der Woche aufgelaufenen Störmeldungen kann von der LZ auf unterschiedliche Weise bei der IZ abgerufen werden. Wird der gesamte der gesamte in Abb.3.3.6-5 gezeigte Block abgerufen, ergibt sich Übertragungsvolumen von $8.338.608 \times 5 \times 55 \approx 2,3$ Mrd. Zellwerten, von denen nur fünf (A-E) relevant sind.

Wesentlich effizienter ist demgegenüber die ausschließliche Berücksichtigung der Zeitpunkte i_A, i_B, \dots, i_E , zu denen die einzelnen Störmeldungen jeweils registriert wurden; das Übertragungsvolumen reduziert sich hierdurch bereits auf $5 \times 5 \times 55 \approx 1.400$ Zellenwerte.



Werden zudem nur die relevanten Sensoren k_A, k_B, \dots, k_E übertragen, so sind nur $5 \times 5 = 25$ Zellenwert zu übertragen.

Die „ökonomischste“ Lösung besteht in der exklusiven Übertragung der Zellen $[i,j,k]A, [i,j,k]B, \dots, [i,j,k]E$.

¹ Zwischen 8 und 16 Uhr sind damit aufgrund der 23 bit langen Mantisse bei single-precision-Darstellung nach IEEE 574 insgesamt $2^{23} = 8.338.608$ Einträge möglich. Daraus ergibt sich eine Zeitaufösung von unter 4 ms.

- Diese Seite ist leer.-

3.4 FND – APDU-Format

Die folgende Abbildung zeigt die Struktur der APDU in schematischer Form.

Bit-Position im Byte		Byte-Nr.								APCI.1	
		MSB				LSB					
		8	7	6	5	4	3	2	1		
Start	1	1	1	1	1	0	0	0	0	}	
Länge (=APDU-Länge-2)	2	length									
Transport-Kontrolle	3	d_r		tag		mpr		seg			
Segment-Folge-Nummer	4	seg_nr									
Dialog-Kennung	5	invoke_id									
Leitzentralen-Kennung	6	lz_id									
	7	lz_subid									
Automationsinsel-Kennung	8	iz_id									
	9	iz_subid									
Datenpunkt-Adresse (16 Bytes)	10	dp_id (erstes Zeichen)									
 (Zeichen / NUL)									
	25	dp_id (letztes Zeichen / NUL)									
Operations-Code	26	fct_id				tab_id				}	
	27	dp_type				dp_subtype					
Applikations-Kontrolle	28	prd		rpr		emq					
DP-Status (= DP_0)	mask	29	val		apr		mis		msk		
	Info_event	30	info				event				
Info-Code	31	exc_1									}
	32	exc_2									
Datum/Uhrzeit	33	year									
	34	month									
	35	day									
	36	hour									
	37	minute									
	38	second									
Datenteil (falls vorhanden, max 90 Bytes)	39	<data>								}	
	...										
	max. 128										

Ein FND-Telegramm besteht aus mindestens 38 Bytes (bei leerem Datenteil) und höchstens 128 Bytes bei maximalem Nutzdateninhalte (ASDU) von 90 Bytes.

Erläuterung der einzelnen Variablen

Start:

Das erste Byte eines FND-Telegramms ist immer F0h.

Länge:

Die Längenangabe gibt die Anzahl der nachfolgenden, zur APDU gehörenden Bytes an. Entsprechend der maximalen Paketlänge von 128 Bytes ist dieser Wert auf 126 beschränkt.

Transport-Kontrolle:

Die Transport-Steuerung setzt sich aus vier 2-Bit-Worten nachstehender Bedeutung zusammen:

d_r: (Richtungs- und Antwort-Bit)

„d_r“ (Bits 8 und 7) kennzeichnen vier Telegrammtypen, die nach Übertragungsrichtung und Mitteilungscharakter wie folgt unterschieden werden:

a) „d“- („direction“) Bit: Übertragungsrichtung

0: Richtung LZ → AI

1: Richtung AI → LZ

b) „r“- („response“) Bit: Antwortkennung

0: Befehls- oder Meldungs-Telegramm

1: Antwort- oder Bestätigungs-Telegramm

Die möglichen „d_r“-Werte, ihre Bedeutung sowie ihre Zuordnung zu den Telegrammtypen sind in nachstehender Tabelle zusammengefasst:

d_r	Tel. Typ	Erläuterung
0 (00b)	CMD	„Command“ Befehl der LZ an eine AI
3 (11b)	RSP	„Response“ Antwort einer AI auf einen Befehl der LZ
2 (10b)	USM	„Unsolicited Message“ Spontanmeldung einer AI an die LZ
1 (01b)	ACK	„Acknowledge“ Bestätigung eines USM-Telegramms

tag: (Telegramm-Attribut)

Zur Kennzeichnung von Telegrammbesonderheiten wird die durch Bit 5 und 6 codierte Variable „tag“ wie folgt verwendet:

tag		Erläuterung
0	0	NORMAL-Telegramm (kein Fehler)
0	1	REJECT-Telegramm (Ablauf-Fehler)
1	0	ERROR-Telegramm (syntaktisch/semantischer Fehler)
1	1	(reserviert)

Eine ausführliche Beschreibung des Exception-Handling ist **Anhang D** zu entnehmen.

mpr: (Nachrichten-Priorität, message priority)

Die vier durch Bit 3 und 4 codierten, in nachstehender Reihenfolge wachsenden Vorrangstufen

- 0: (normal)
- 1:
- 2:
- 3: (dringlichst)

dienen der Prioritätskennzeichnung der Nachricht auf Anwendungsebene. Die Reaktion hierauf ist projektspezifisch zu vereinbaren.

Bei RSP- bzw. ACK-Telegrammen ist mpr aus dem jeweils zugehörigen CMD- bzw. USM-Telegramm unverändert zu übernehmen.

seg: (Segmentierungs-Kontrolle)

Bits 1 und 2 kennzeichnen den Segmentierungsstatus von Nachrichten, die aufgrund ihrer Länge nicht in ein Datenpaket (max. 128 Bytes) passen:

seg	Erläuterung
00b	Keine Segmentierung
11b	Segmentierung: erstes Element der Folge
10b	Segmentierung: weiteres, aber nicht letztes Element der Folge
01b	Segmentierung: letztes Element der Folge

(Einzelheiten siehe 3.8 Segmentierung).

Segment-Folge-Nummer:

„seq_nr“ hat bei unsegmentierten Telegrammen den Wert 0.

Bei Segmentierung dient „seq_nr“ (modulo 256 der mit 0 beginnenden laufenden Segmentnummer) der Kennzeichnung der Reihenfolge, in der die einzelnen Teillegramme auf der Empfängerseite zusammengesetzt sind.

((Einzelheiten siehe 3.8 Segmentierung).

Dialog-Kennung:

Die „invoke_id“ dient der Flußkontrolle und der Dialogsicherung (Beschreibung siehe Anhang A Telegramm-Bearbeitung)

Leitzentralen-Kennung:

Die Leitzentralen-Kennung setzt sich aus den zwei Teilfeldern „lz_id“ und „lz_subid“ zusammen:

lz_id:

„lz_id“ ist eine zur Kennzeichnung der LZ projektspezifisch zu vergebende Nummer im Bereich 0 - 255.

lz_subid:

„lz_subid“ ist eine von der LZ für CMD-Telegramme optional vergebene Kennung (Defaultwert = 0), welche die Zuordnung des Telegramms zum auftraggebenden internen Prozess herstellt.

„lz_subid“ ist von der AI unverändert in das korrespondierende RSP-Telegramm zu übernehmen.

Bei USM-Telegrammen kann „lz_subid“ projektspezifisch bestimmten Nachrichtenkategorien zugeordnet sein und ist von der LZ unverändert in das ACK-Telegramm zu übernehmen.

Insel-Kennung:

Die Insel-Kennung setzt sich aus den zwei Teilfeldern „iz_id“ und „iz_subid“ zusammen:

iz_id:

Die Kennzeichnung der einzelnen Automationsinsel geschieht durch die Zuordnung einer eindeutigen Nummer im Bereich 0 - 255.

iz_subid:

„iz_subid“ ist eine von der AI für USM-Telegramme optional vergebene (Defaultwert = 0), welche die Zuordnung des Telegramms zum auftraggebenden internen Prozess herstellt

„iz_subid“ ist von der LZ unverändert in das korrespondierenden ACK-Telegramm zu übernehmen.

Bei CMD-Telegrammen kann „iz_subid“ projektspezifisch bestimmten Nachrichtenkategorien zugeordnet sein und ist von der Insel unverändert in das RSP-Telegramm zu übernehmen.

Datenpunkt-Adresse:

Die Datenpunkt-Adresse wird durch einen bis zu 16 Zeichen langen ASCII-String repräsentiert, der eindeutige Kennzeichnung eines physikalischen oder virtuellen Datenpunkts darstellt

Der zugelassene Zeichenvorrat entspricht ANSI X3.4 ohne die Zeichen 01h bis 1Fh und ohne 7Fh (DEL)

Datenpunkt-Adressen mit weniger als 16 Zeichen werden durch binäre Nullen (NUL=00h) auf 16 Bytes ergänzt .

Operations-Code:

Der Operations-Code ist detailliert im Abschnitt 3.4.1 beschrieben.

Applikations-Kontrolle:

Das Applikations-Kontroll-Byte setzt sich zusammen aus:

„prd“: (Durchgriffs-Priorität):

Bits 7 und 8 codieren die Durchgriffspriorität, die im Zusammenwirken mit der Sperrpriorität „rpr“ einen effektiven Arbitrierungsmechanismus zwischen AI-internen und LZ-spezifischem Zugriff auf Datenpunkte darstellt. Die durch die „prd“-Bitkombination darstellbaren Werte von 0 - 3 sind in aufsteigender Reihenfolge wachsenden Zugriffsprioritäten zugeordnet, wobei 0 dem „Normalanwender“, und der Wert 3 dem besonders privilegierten Systemmanager zugeordnet ist.

„rpr“-Bitgruppe: (Sperr-Priorität)

Bits 5 und 6 codieren die Sperrpriorität, die im Zusammenwirken mit der Durchgriffspriorität als Arbitrierungsmechanismus zwischen LZ- und AI-seitigem DP-Zugriff dient. Die durch die Bitkombination darstellbaren Werte 0 - 3 sind in aufsteigender Reihenfolge wachsenden Sperrprioritäten zugeordnet, wobei 0 allgemeinen und der Wert 3 (sofern zugelassen) LZ-Befehlen ausschließlichen Zutritt sichert.

(Details siehe Abschnitt 3.7 zu finden).

„emq“: (Ereignis-Anzeige/Modifikationskontrolle)

Der in Bit 1 - Bit 4 codierte „event-modification-qualifier“ ist in den 3.4.2 Befehle und 3.4.2.3 Meldungen detailliert beschrieben.

DP-Status:

Der Datenpunkt-Status, bestehend aus den beiden Bytes „mask“ und „info-event“, wird detailliert in den Abschnitten 3.4.2.3 Meldungen und 3.7 Zugriffskontrolle, erläutert.

Info-Code:

Bedeutung und Verwendung des aus zwei Bytes „exc_1“, „exc_2“ zusammengesetzten Info-Codes „exc“ wird in Anhang D Fehlerbehandlung beschrieben. Bei CMD- und USM-Telegrammen sind „exc_1“, „exc_2“ auf den Wert 0 zu setzen.

Datum/Uhrzeit:

Datum und Uhrzeit werden in der 6-Byte-Form

year (Jahr)	(Wertebereich: 0 – 99)
month (Monat)	(Wertebereich: 1 – 12)
day (Tag)	(Wertebereich: 1 – 31)
hour (Stunden)	(Wertebereich: 0 – 23)
minute (Minuten)	(Wertebereich: 0 – 59)
second (Sekunden)	(Wertebereich: 0 – 59)

angegeben.

Bei RSP- bzw. ACK-Telegrammen wird Datum/Uhrzeit aus dem jeweils zugehörigen CMD- bzw. USM-Telegramm unverändert übernommen.

Datenteil:

Die Beschreibung des Datenteils ist Gegenstand der Abschnitte 3.4.2 Befehle und 3.4.2.3 Meldungen.

3.4.1 Operationsbeschreibung

Zwischen Leit- und Inselzentrale sind die zwei Dialogformen
 "CMD / RSP" (Command / Response) und
 "USM/ ACK" (Unsolicited Message / Acknowledge)
 definiert.

Bei Kommando-Dialog "CMD / RSP" geht die Initiative von der Leitzentrale aus, die auf einen Datenpunkt in lesender oder verändernder Absicht zugreift. Er ist beschreiben im Abschnitt 3.4.2 Befehle.

Beim rein informativen Spontanmeldungs-Dialog "USM/ ACK", der in Abschnitt 3.4.2.3 Meldungen beschrieben ist, geht die Initiative von einem Datenpunkt innerhalb einer AI.aus.

Der Operations-Code setzt sich aus einer Elementarfunktion ("fct_id"), dem Datenpunkt-Typ ("dp_type" und "dp_subtype"), sowie der (bzw. den) adressierten DP-Tabelle(n) ("tab_id") zusammen:

Position im Byte	MSB				LSB			
	8	7	6	5	4	3	2	1
Operations-Code	fct_id				tab_id			
	dp_type				dp_subtype			

Der Operations-Code wird bei RSP- bzw. ACK-Telegrammen unverändert aus dem jeweils zugehörigen CMD- bzw. USM-Telegramm übernommen.

Die Bedeutung der Variablen des Operations-Codes ist für Befehle (CMD/RSP) und Meldungen (USM/ACK) leicht unterschiedlich.

3.4.2 Befehle

fct_id:

Für die Codierung gilt:

fct_id	Funktion	Bedeutung
0 (0000b)	Fehlerbehandlung	Antwort auf fehlerhaftes Telegramm
1 (0001b)	Lesen	Lesender Zugriff auf DP-Tabellen
2 (0010b)	Modifizieren	Modifizierender Zugriff auf DP-Tabellen

tab_id:

Die Gesamtheit der Informationen eines Datenpunkts wird durch bis zu vier DP-Tabellen repräsentiert:

- DP-Tabelle 0: Status-Informationen
- DP-Tabelle 1: Nutz-Informationen
- DP-Tabelle 2: Parameter (sofern vorhanden)
- DP-Tabelle 3: Konfiguration (nur Karteipunkt)

Diese werden wie folgt codiert:

tab_id	DP-Tabelle	Befehle
0 (0000b)	DP_0	Sperrern von Ereignismeldungen Lesen des DP-Status
1 (0001b)	DP_1	Lesen / Modifizieren DP-spezifischer Nutzdaten
2 (0010b)	DP_2	Lesen / Modifizieren von DP-Parametern
3 (0011b)	DP_1&2	Lesen DP-Nutzdaten und -Parametern
4 (0100b)	DP_3	Lesen DP-Konfiguration (nur Karteipunkt)

dp_type:

Die verwendeten Datenpunkt-Typen werden wie folgt codiert

dp_type	DP-Kategorie
0 (0000b)	Fehlerbehandlung vgl. D.2 ERROR-APDU-Format (syntaktische/semantischer Fehler)
1 (0001b)	Meldepunkt
2 (0010b)	Schaltpunkt (mit Rückmeldung)
3 (0011b)	Messpunkt
4 (0100b)	Stellpunkt
5 (0101b)	Zählpunkt
7 (0111b)	Karteipunkt

dp_subtype:

Zur zusätzlichen Unterscheidung wird ein sog. "dp_subtype" vergeben:

dp_type	dp_subtype	Bemerkung
1-5, 7	0 (0000b)	(keine Subkategorie definiert)
0	1 (0001b)	kennzeichnet Fehlerbehandlung (vgl. D.2 ERROR-APDU-Format (syntaktische/semantischer Fehler))

3.4.2.1 Datenteil

Sofern der Datenteil nicht leer ist, stimmt er bei allen Datenpunktentypen mit Ausnahme des Fehler-Telegramms mit einer der Strukturen

DP_1 , DP_2, DP_1&2 oder DP_3
des adressierten Datenpunkts überein.

3.4.2.2 Modifikations-Kontrolle

Modifizierende Befehle auf DP-Tabellen können sich auf eines oder mehrere der hierin ausgewiesenen Parameterfelder (P_1, P_2 etc.) beziehen. Die Auswahl der zu ändernden Parameter erfolgt mit der Variable "emq" im Applikations-Kontrollbyte:

emq =	P_4	P_3	P_2	P_1
	p4	p3	p2	p1

Die Bedeutung der vier Bitvariablen p1 - p4 ist dabei:

p<i> = 0:	das dem Parameter P_<i> entsprechende Feld im Datenteil des Telegramms ist zu ignorieren
p<i> = 1:	das dem Parameter P_<i> entsprechende Feld im Datenteil des Telegramms ersetzt den Wert des gleichnamigen Parameters in der durch "tab-id" adressierten DP-Tabelle. (Der Modifikationsversuch eines nicht vorhandenen Parameters wird als semantischer Fehler gewertet, der Überschreitungsversuch eines durch ein gesetztes "update-control"-Bit geschützten Feldes zurückgewiesen).

Beim RSP-Telegramm ist „emq“ aus dem CMD-Telegramm unverändert zu übernehmen.

Beispiel:

Obere und untere Grenzwerte eines Messpunkts sollen geändert werden, oberer und unterer Alarmwert sollen unverändert bleiben.. Der Datenteil des Modifikations-telegramms hat dann die Gestalt,

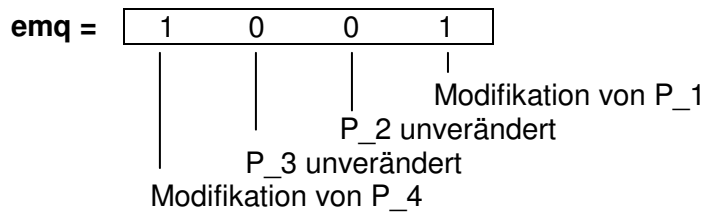
DP_2

X X X X X X X X

update_control irrelevant, da kein Parameter

XXXXXXXXXX	fixed	Wert irrelevant, da kein Parameter
<alarm_low>	P_1 alarm_low	neuer Wert
XXXXXXXXXX XXXXXXXXXX XXXXXXXXXX XXXXXXXXXX	P_2 warning_low	Wert irrelevant, da keine Modifikation
XXXXXXXXXX XXXXXXXXXX XXXXXXXXXX XXXXXXXXXX	P_3 warning_high	Wert irrelevant, da keine Modifikation
<alarm_high>	P_4 alarm_high	neuer Wert

wobei "emq" als Selektions-Steuerung den Wert



besitzt.

(Ende Beispiel)

Anmerkungen:

- (1) Die Modifikation des Parameters P_1 der DP-Tabelle 0 ("msk") wird in Abschnitt 0 gesondert behandelt.
- (2) Für die Elementar-Funktionen „Fehlerbehandlung" und "Lesen" hat "emq" keine Bedeutung und ist auf 0 zu setzen.
- (3) Der Wert von "emq" wird beim RSP-Telegramm unverändert aus dem CMD-Telegramm übernommen.

3.4.2.3 Meldungen

fct_id:

Für die Codierung gilt:

fct_id	Funktion	Bedeutung
0 (0000b)	Fehlerbehandlung	Antwort auf fehlerhaftes Telegramm
1 (0001b)	Lesen	Spontane Meldung von Ereignissen Datenteil = DP-Tabelle 1

tab_id:

Alle Meldungen enthalten als Datenteil die DP-Tabelle 1, codiert durch tab_id = 0001b.

dp_type:

(Bedeutung und Codierung wie unter 3.5.1 beschrieben)

dp_subtype:

(Bedeutung und Codierung wie unter 3.5.1 beschrieben)

3.4.2.4 Datenteil

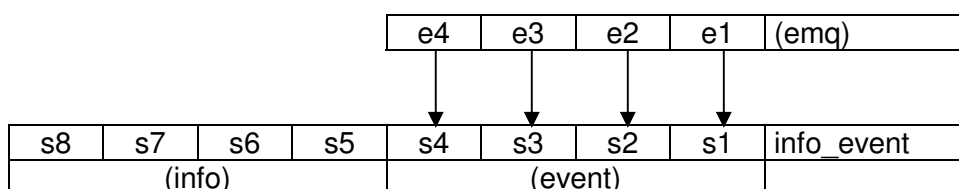
Sofern nicht leer, stimmt der Datenteil eines Meldung-Telegramms bei Datenpunkten der Kategorie 1 - 6 strukturell wie semantisch mit der DP-Tabelle 1 des adressierten Datenpunkts überein.

Beim ERROR-Telegramm ist

3.4.2.5 Ereignis-Anzeige

Spontanmeldungen werden durch Ereignisse, das sind Wechsel des Werts mindestens eines der "event-Bits" von 0 auf 1 bzw. von 1 auf 0, ausgelöst.

Die Variable „emq“ (Bits 1 - 4 des Applikations-Kontroll-Bytes) markiert dabei die geänderten event-Bits:



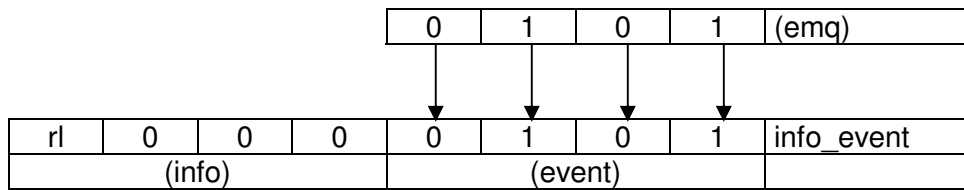
Die Bits "e1" - "e4" sind mit folgender Bedeutung belegt:

e<i> = 0:	(kein Ereignis)
e<i> = 1:	Spontanmeldung aufgrund des s<i> zugeordneten Ereignisses
<i> ::= 1 / 2 / 3 / 4	

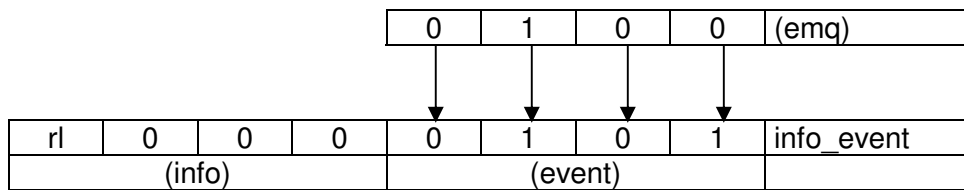
Bei ACK-Telegrammen wird emq aus dem jeweils zugehörigen USM-Telegramm unverändert übernommen.

Beispiel:

Entdeckt ein Meldepunkt (siehe 3.3.1 Meldepunkt) gleichzeitig eine Geber-Störung (Bit 1) und eine (möglicherweise hierdurch bedingte) Wertänderung (Bit 3), so sind im entsprechenden Melde-Telegramm die Variablen "emq" und "info_event" wie folgt belegt:



Eine erneute Wertänderung bei unveränderter Geberstörung führt zu einer Spontanmeldung mit



3.5 Beschreibung des DP-Status

Jedem Datenpunkt ist unter der Bezeichnung "DP-Tabelle 0" ein Datenpunkt-Status² zugeordnet, der sich aus den beiden mit "mask" und "info_event" bezeichneten Bytes zusammensetzt:

val_apr		mis_msk		
MSB			LSB	
val	apr	mis	msk	mask
info		event		info_event
info		event		

Die Informationsgruppen info_event, val_apr und mis_msk sind dabei unterschiedlichen Aufgaben zugeordnet:

info_event

Den Bits im "event"-Teil sind Zustände zugeordnet, deren Wechsel als Ereignis interpretiert wird; die erforderlichen Zusatzinformationen sind dabei im Halb-Byte "info" enthalten.

val_apr

Das aus den beiden Variablen "val" und "apr" bestehende Halb-Byte val_apr dient im wesentlichen folgenden Zwecken:

- der Rechtezuteilung (Arbitrierung) bei gleichzeitigem AI- und LZ-seitigem Versuch, DP-Tabellen zu ändern,
- der LZ bzw. der AI für eine bestimmte Zeitdauer ausschließliche Verfügungsgewalt über einen Datenpunkt zu garantieren.

(Das Verfahren ist detailliert in Abschnitt 3.7 Zugriffskontrolle beschrieben).

mis_msk

Das abgestufte Ausblenden von Ereignismeldungen wird durch die beiden Variablen "mis" und "msk" gesteuert, wobei erstere der AI-internen, und letztere der LZ-seitigen Kontrolle unterstellt ist. Dabei wird beim AI-seitigen Ausblenden auch die entsprechende Meldung zur LZ unterdrückt.

(Details hierüber sind in Abschnitt 3.6 Sperren von Ereignismeldungen zu finden).

Bei ACK-Telegrammen wird der DP-Status aus dem jeweils zugehörigen USM-Telegramm unverändert übernommen.

² Das APDU-Feld namens "DP-Status" ist identisch mit der aktuellen DP-Tabelle 0, dem DP-Status des Datenpunkts. Dies gilt für RSP- und USM-Telegramme, bei denen relevante Statusinformation übertragen wird. Bei ACK-Telegrammen wird der DP-Status aus dem jeweils zugehörigen USM-Telegramm unverändert übernommen. Im ERROR-Telegramm ist der "DP-Status" stets gleich 0.

Obwohl in allen RSP- und USM-Telegrammen der Inhalt der DP-Tabelle 0 als Kopie in der Telegramm-Variablen "DP-Status" enthalten ist, besteht auch die Möglichkeit der gezielten Status-Anforderung durch einen an DP-Tabelle 0 adressierten Lesezugriff:

CMD-Telegramm:

fct_id		tab_id		...	MSB				LSB			
dp_type		dp_subtype		...	0	0	0	1	0	0	0	0
prd	rpr	emq		...	dp_type		0 0 0 0					
val	apr	mis	msk	...	0	0	0	0	0	0	0	0
				...	0	0	0	0	0	0	0	0

Der aktuelle Status ist dann dem Feld "DP-Status" in dem zugehörigen RSP-Telegramm, das einen leeren Datenteil enthält, zu entnehmen.

3.6 Sperren von Ereignismeldungen

Das in der Datenpunkt-Tabelle DP_0 enthaltene „info_event“-Byte setzt sich aus einem „info“- und einem „event“-Halbbyte zusammen. Den vier Bit in „event“ sind die DP-spezifischen Ereignis-Quellen (z.B. Wertänderung, Grenzwertverletzung, Geber-/BTA-Störung) zugeordnet. Die Änderung eines dieser Bits kennzeichnet ein Ereignis und führt zu einer spontanen Meldung (USM).

In gewissen Situationen kann es sinnvoll sein, die Meldungen einer bestimmten Kategorie zu unterdrücken.

Beispiel 1:

Die Betriebsmeldungen eines Brenners sind nur während der Einfahrphase relevant. Während des größten Teils der Betriebszeit belasten diese Meldungen nur die Ressourcen des GA-Systems und sollen deshalb nicht übertragen werden. Anderer Ereignisse wie Geberstörungen sollen aber weiterhin gemeldet werden.

Beispiel 2:

Führt ein Kontaktproblem zu einer „flatternden“ Geberstörung kommt es zu ständigen überflüssigen Neumeldungen. In diesem Fall sollen die Ereignismeldungen bzgl. der Wertänderung als auch der Geberstörung gesperrt werden, bis die Störung behoben ist.

Die Ausblendung von Ereignismeldungen erfolgt durch Modifikation des Meldungskontroll-Feldes „msk“ im „mask“-Byte der Datenpunkt-Tabelle DP_0.

Das "mask"-Byte setzt sich aus vier 2-Bit-Worten zusammen:

val_apr		mis_msk		
MSB			LSB	
val	apr	mis	msk	mask

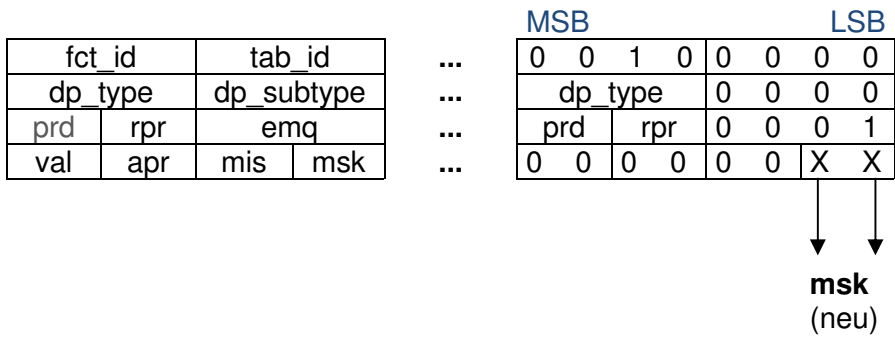
Das Ausblenden von Ereignismeldungen wird durch das niederwertige Halbbyte "mis_msk" kontrolliert, während "val_apr" Bestandteil der in Abschnitt 3.7 behandelten Zugriffskontrolle ist.

Das Ausblenden von Ereignismeldungen erfolgt in abgestufter Weise, wobei die Variablen "mis" und "msk" identische Funktionen erfüllen, mit dem Unterschied, dass "mis" ausschließlich der AI-internen und "msk" ausschließlich der LZ-seitigen Kontrolle unterstellt ist. Dabei wird beim AI-seitigen Ausblenden auch die entsprechende Meldung zur LZ unterdrückt, nicht jedoch umgekehrt, d. h. der AI-interne Meldungsverkehr wird durch den Wert von "msk" nicht beeinflusst.

Codierung und Bedeutung der beiden Variablen "mis" und "msk" sind der nachstehenden Tabelle zu entnehmen:

mis/msk	Effekt
0 (00b)	Freigabe aller Ereignismeldungen
1 (01b)	Ausblenden der Betriebs-Meldungen; gemeldet werden Geber- und BTA-Störungen
2 (10b)	Ausblenden der Betriebs- und BTA-Störungs-Meldungen; gemeldet werden Geber- Störungen
3 (11b)	Ausblenden aller Ereignismeldungen

Die Modifikation des Meldungs-Kontroll-Felds erfolgt durch ein CMD-Telegramm der nachstehend gezeigten Art:



Bits 1 und 2 des "mask"-Bytes enthalten also den neuen Wert der Ausblendungs-Stufe, während die höherwertigen Bit-Stellen auf 0 zu setzen sind.

3.7 Zugriffskontrolle

Zugriffe auf Datenpunkte können – auch gleichzeitig – von zwei Seiten erfolgen:

- von Prozessen innerhalb der AI
- von der LZ

Während bei lesenden Zugriffen keine Konflikte entstehen, ist bei modifizierenden Zugriffen (z.B. Schaltprogramm) ein Entscheidungsmechanismus erforderlich, der die Behandlung der konkurrierenden Zugriffe steuert.

Beispiel:

In einer Schule soll bei ausreichender Helligkeit zu bestimmten Zeiten verhindert werden, dass Schüler das Flurlicht aus Spaß einschalten (Tastersperre).

Der Zuteilungsmechanismus arbeitet wie folgt:

Jedem Datenpunkt ist eine "access-priority" ("apr") zugeordnet. Nur Prozessen, deren Durchgriffspriorität "prd" (= "privilege degree") mindestens so groß wie "apr" ist, wird der modifizierende Zugriff auf den Datenpunkt gestattet. Nachdem der zugriffsberechtigte Prozess die Daten modifiziert hat, überschreibt er die DP-Zugriffspriorität "apr" mit der von ihm bestimmten Sperrpriorität "rpr" (= "residual priority"). Damit wird eine neue Zugriffssituation für die anderen um den DP konkurrierenden Prozesse geschaffen.

Bei LZ-Kommandos führt die Zugriffsverweigerung zu einem REJECT-Telegramm mit Fehler-Code "Ungenügende Durchgriffspriorität".

"prd" und "rpr" sind im Applikations-Kontrollbyte in 2-Bit-Codierung enthalten:

prd_rpr		emq		Applikations-Kontrolle
MSB			LSB	
prd	rpr	emq		

"prd" und "rpr" sind nur in CMD-Telegrammen relevant, die modifizierend auf DP_1 oder DP_2 eines Datenpunkts zugreifen; andernfalls sind beide Variablen auf 0 zu setzen.

Bei USM- und ERROR-Telegrammen sind "prd" und "rpr" ohne Bedeutung und ebenfalls auf 00b zu setzen.

"prd_rpr" ist bei RSP- bzw. ACK-Telegrammen unverändert aus dem jeweils zugehörigen CMD- bzw. USM-Telegramm zu übernehmen.

Durch die Einführung von vier „Valenzkategorien“

- LZ-Exklusivität
- LZ-Vorrang
- Insel-Vorrang
- Insel-Exklusivität

wird zusätzliche Zugriffskontrolle ermöglicht, um der LZ bzw. der AI einen projektierten exklusiven bzw. bevorzugten Modifikations-Zugriff auf den Datenpunkt zu gewähren.

Valenz- und Zugriffspriorität eines Datenpunkts sind im "val_apr"-Teil des "mask"-Bytes wie folgt verschlüsselt:

val_apr		mis_msk		mask
MSB			LSB	
val	apr	mis	msk	

val:

Die Valenz eines Datenpunkts ist durch zwei Bits gemäß nachstehender Tabelle codiert:

val	Bedeutung
0 (00b)	<u>LZ-Exklusivität:</u> Unbeschränkte Modifizierungsberechtigung der DP-Tabellen seitens der LZ. AI-seitig modifizierender Zugriff auf die DP-Tabellen nicht gestattet.
1 (01b)	<u>LZ-Vorrang:</u> Setzen der Zugriffspriorität 3 nur seitens der LZ möglich.
2 (10b)	<u>Insel-Vorrang:</u> Setzen der Zugriffspriorität 3 nur seitens der AI möglich.
3 (11b)	<u>Insel-Exklusivität:</u> Unbeschränkte Modifizierungsberechtigung der DP-Tabellen seitens der AI LZ-seitig modifizierender Zugriff nur auf die DP-Tabelle 0 gestattet.

apr:

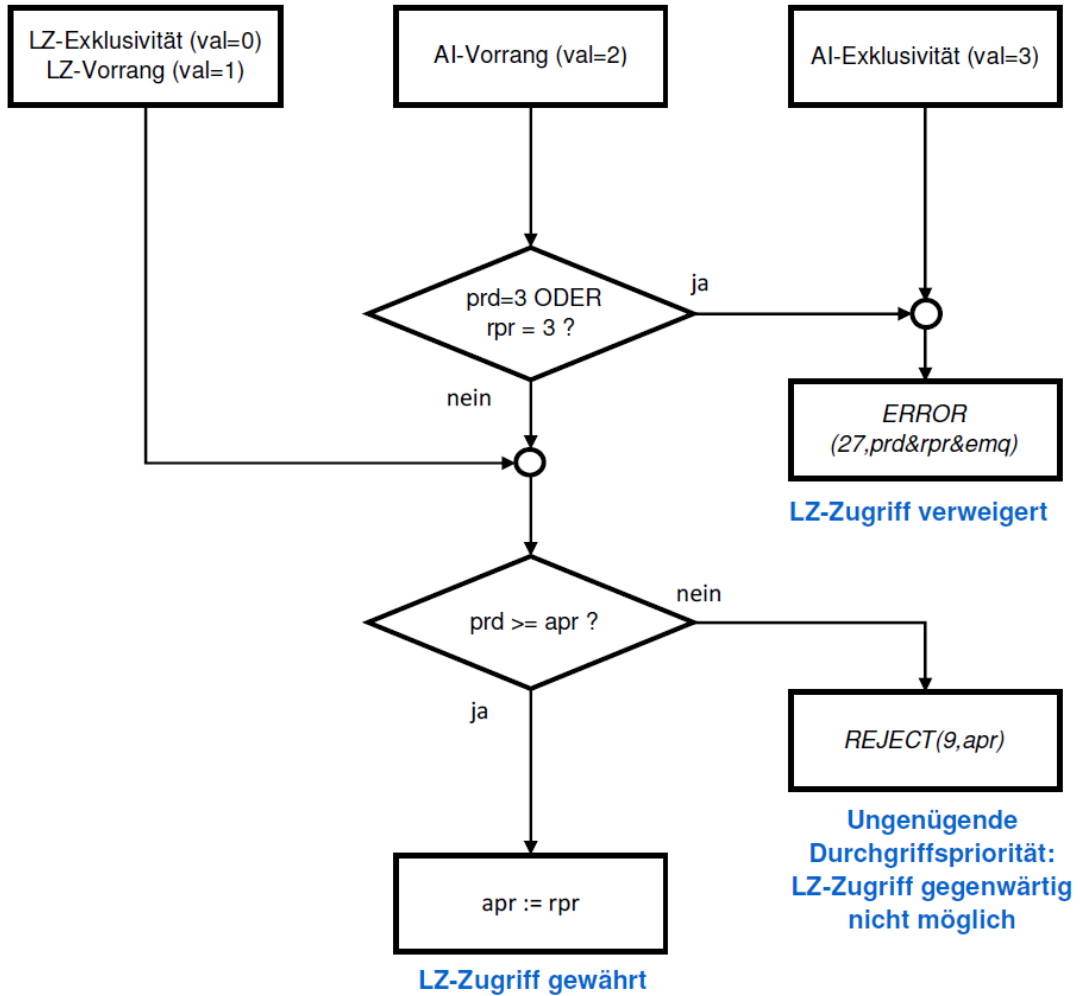
Das Feld "apr" enthält die im Bereich 0 - 3 zeitlich variierende Sperrpriorität des Datenpunkts, welche die LZ- bzw. AI-seitige Modifikationsberechtigung bzgl. des Datenpunkts kontrolliert.

Bei CMD-Telegrammen, sowie generell bei Transferpunkten sind "val" und "apr" ohne Bedeutung und auf 00b zu setzen.

Eine präzise Fassung des vorstehend erläuterten Sachverhalts vermitteln die beiden folgenden Ablaufdiagramme:

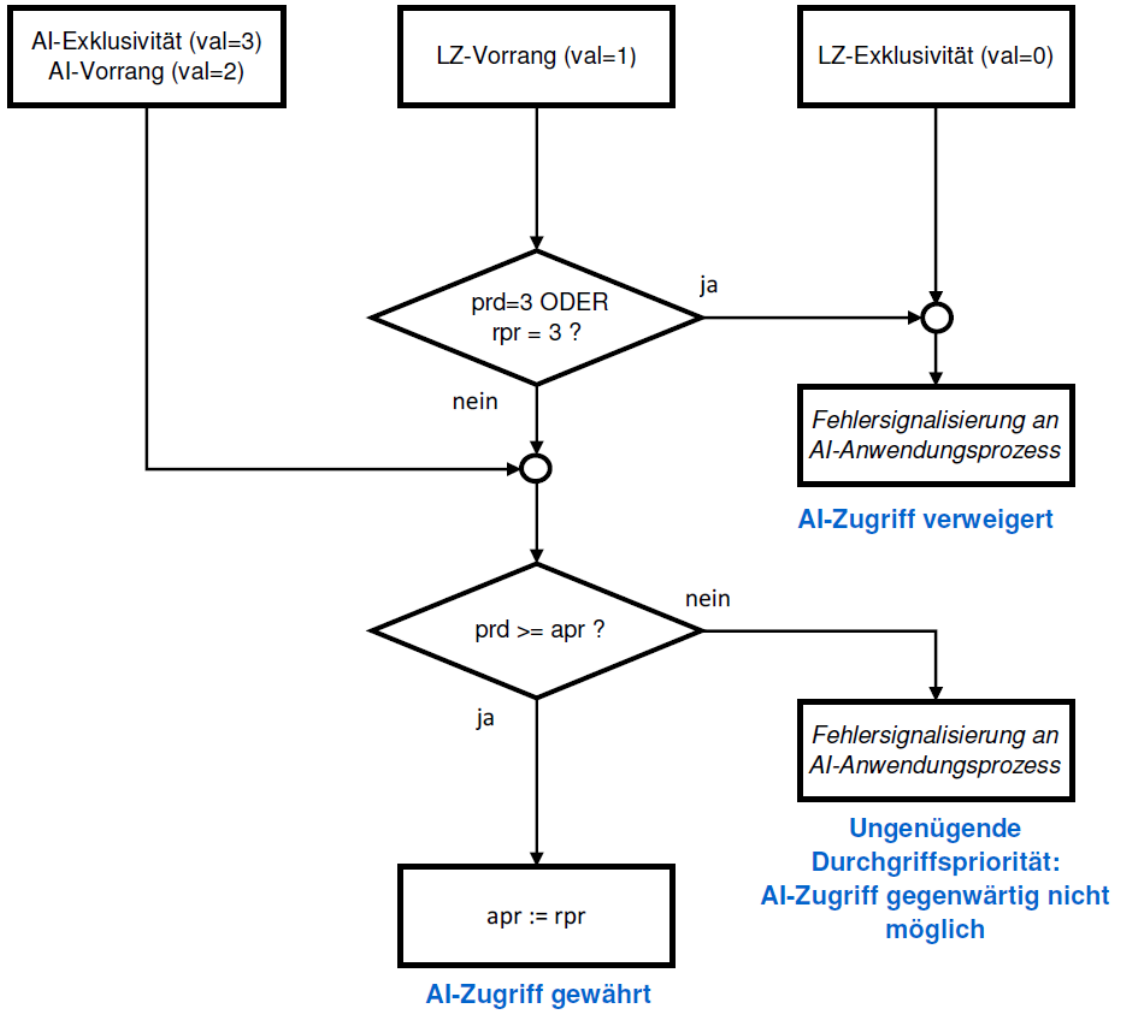
Klärung der Berechtigung der LZ zur DP-Modifikation

Die LZ sendet im CMD-Telegramm die Werte für „prd“ und „rpr“ an die AI, wo diese Informationen mit den aktuellen DP-Werten für val und apr wie folgt verarbeitet werden:



Klärung der der Berechtigung der AI zur DP-Modifikation

Der modifikationswillige Prozess in der AI sendet die Werte für „prd“ und „rpr“ an den AI-internen DP-Prozess, der diese Informationen mit den aktuellen DP-Werten für val und apr wie folgt verarbeitet:



3.8 Segmentierung

Unter Segmentierung wird die Zerlegung einer längeren Nachricht in eine Folge kleinerer Informationspakete verstanden, die – am Bestimmungsort in der korrekten Reihenfolge zusammengesetzt – wieder die ursprüngliche Nachricht ergeben. Da die Länge eines FND-Telegrammes auf 128 Bytes beschränkt ist, darf der Datenteil aus max. 90 Bytes bestehen. Diese Begrenzung betrifft nur Karteipunkte.

Eine ASDU mit mehr als 90 Bytes ist so in (n+1) nichtleere, in ihrer Länge auf jeweils 90 Bytes beschränkte Teilssegmente ASDU(0),...,ASDU(n) zu zerlegen, dass

- die Segmente ASDU(0),...,ASDU(n-1) eine Länge von genau 90 Bytes aufweisen,
- das Segment ASDU(n) eine Länge von mindestens einem und maximal. 90 Bytes aufweist und
- die ASDU sich als Verkettung der so durchnummerierten Segmente darstellt, also ASDU = ASDU(0)&ASDU(1)&...&ASDU(n).

Daraus ergeben sich die segmentierten APDU(0),...,APDU(n) wie folgt :

APDU(0) = APCI(0)&ASDU(0)

...

APDU(n) = APCI(n)&ASDU(n)

APDU(0),...,APDU(n) sind in dieser Reihenfolge sukzessive zu übertragen.

Die Header APCI(0),...,APCI(n) müssen mit Ausnahme der Bestandteile

- length
- seg
- seq_nr
- invoke_id
- year, month, day, hour, minute, second

übereinstimmen, ansonsten wird die Übertragung abgebrochen

Die „seq_nr“ durchläuft in lückenlos aufsteigender Reihenfolge die Werte (0,...,n) modulo 256. Werden mehr als 256 Segmente übertragen, wiederholen sich die „seq_nr“. Daher muss für „seq_nr“ sicher gestellt sein, dass ein vorausgegangenes APDU mit gleicher „seq_nr“ vom Empfänger mittels RSP-/ACK-Telegramm quittiert wurde.

Für die Besetzung von „seg“ gilt :

APDU(0)	seg = 11b	(erstes Element der Folge)
APDU(1), ..., APDU(n-1)	seg = 10b	(weiteres Folgeelement)
APDU(n)	seg = 01b	(letztes Element der Folge)

Die Header der zu APDU(0),...,APDU(n) gehörenden RSP-/ACK-Telegramme stellen (mit Ausnahme von „d_r“) Kopien der entsprechenden CMD-/USM-Header dar.

Die Datenteile der zu APDU(0), ..., APDU(n-1) gehörenden RSP-/ACK-Telegramme bleiben leer, da es sich hier lediglich um Teil-Empfangsquittungen (Kurzquittungen) der gesamten APDU handelt, die erst nach Eingang des letzten Folgeelements APDU(n) vollständig zusammen gesetzt und damit von der Empfängerseite interpretiert werden kann.

Timeouts oder Übertragungsfehler eines der APDU(i) führen zum Gesamt-Abbruch der Übertragung, d.h. alle noch nicht übertragenen Folgeelemente werden aus der Sende-Warteliste gestrichen und der Sende-Auftrag mit entsprechendem Hinweis an den Anwendungsprozess storniert.

Auf der Empfängerseite löscht spätestens eine Zeit-Überwachungsfunktion unvollständig empfangene Segment-Folgen, wenn eine projektspezifisch vereinbarte ma-

ximale Zeitspanne zwischen dem Empfang zweier aufeinander folgender APDU(i), APDU(i+1) überschritten wird.

4 Anhang

A Telegramm-Bearbeitung

Die folgenden Abbildungen zeigen die Abläufe bei der Bearbeitung von Sendeaufträgen bzw. Empfangsbestätigungen durch AI bzw. LZ. Dabei werden aufgrund der weitgehenden Übereinstimmungen die Fälle CMD und USM sowie RSP und ACK zusammen behandelt.

Der Bearbeitung erfolgt in LZ und AI jeweils durch zwei parallel ablaufende Software-Prozesse. Der LZ-Anwendungsprozess kommuniziert mit dem Bediener, während der AI-Anwendungsprozess mit den Datenpunkten in Verbindung steht. Beide Anwendungsprozesse tauschen ihre FND-Telegramme untereinander über ihre Kommunikationsprozesse aus. Deren Aufgabe besteht zum einen darin, Sendewünsche des Anwendungsprozesses entgegenzunehmen und auszuführen. Andererseits sind ankommende Telegramme in Empfang zu nehmen und an den Anwendungsprozess weiterzuleiten. Darüber hinaus überwachen die Kommunikationsprozesse die Transportverbindungen und den Ablauf der FND-Kommunikation.

Die Abläufe bei der Telegrammbearbeitung sind dargestellt als Abfolge von Entscheidungen und Aktionen, ausgehend von einem Ruhezustand „Idle“. Die Abläufe werden beim Auftreten bestimmter Ereignisse gestartet und enden immer mit dem Ruhe- Zustand.

Einige Hinweise zum besseren Verständnis:

Jedem Datenpunkt ist lokal, d.h. nach außen nicht sichtbar, eine Variable `dp_busy` zugeordnet, die die Werte „belegt (TRUE) oder „frei“ (FALSE) annehmen kann. „Belegt“ bedeutet hierbei, dass der Datenpunkt zum gegenwärtigen Zeitpunkt in der Dialogphase steht. „frei“ beschreibt den Zustand nach Dialog-Abschluss, also nach Senden eines RSP-Telegrammes im Falle der AI bzw. nach Senden eines ACK-Telegrammes im Falle der LZ.

Durch das Abweisen von Aufträgen im „Belegt“-Zustand durch Senden von REJECT-Telegrammen wird vermieden, dass ein Datenpunkt mit mehreren – ggf. konkurrierenden – Aufträgen gleichzeitig konfrontiert wird und damit ein unvorhersehbarer Reaktion provoziert wird.

Jeder Kommunikationsprozess verfügt über einen Empfangs- und einen Sendepool. Beim Start des Prozesses ist der Empfangspool leer und der Sendepool mit einer vorgegeben Anzahl unterschiedlicher Marken (`invoke_id`) gefüllt. Die beiden Pools dienen der Sicherstellung des korrekten Telegrammaustauschs.

Beim Versand eines CMD- oder USM-Telegramms wird eine Marke entnommen und dem adressierten Datenpunkt zugeordnet. Die Marke kommt erst dann zurück in den Sendepool, wenn das zugehörige RSP- bzw. ACK-Telegramm eingegangen ist. Ein Auftragstimer überwacht, ob die Antwort in der vorgegebenen Zeit eintrifft. Ist dies nicht der Fall, so wird der Kommunikationsvorgang abgebrochen, die Marke kommt zurück in den Pool und der beauftragende Prozess wird informiert.

In ähnlicher Weise wird der Eingang segmentierter Telegramme durch den Empfangspool und den Empfangstimer kontrolliert.

Kurzquittungen sind RSP- oder ACK-Telegramme ohne Datenteil, die den Eingang von segmentierten Teitelegrammen signalisieren. Nach Eingang des Endetelegramms einer Sequenz erfolgt eine normale Quittierung.

REJECT-Telegramme werden als Sonderfälle von RSP/ACK-Telegrammen behandelt, während ERROR-Telegramme den RSP-Telegrammen zugerechnet werden.

Die „Stornierung“ eines Auftrags bzw. Vorgangs ist verbunden mit der Freigabe aller hierdurch gebundenen Ressourcen bzw. Referenzen

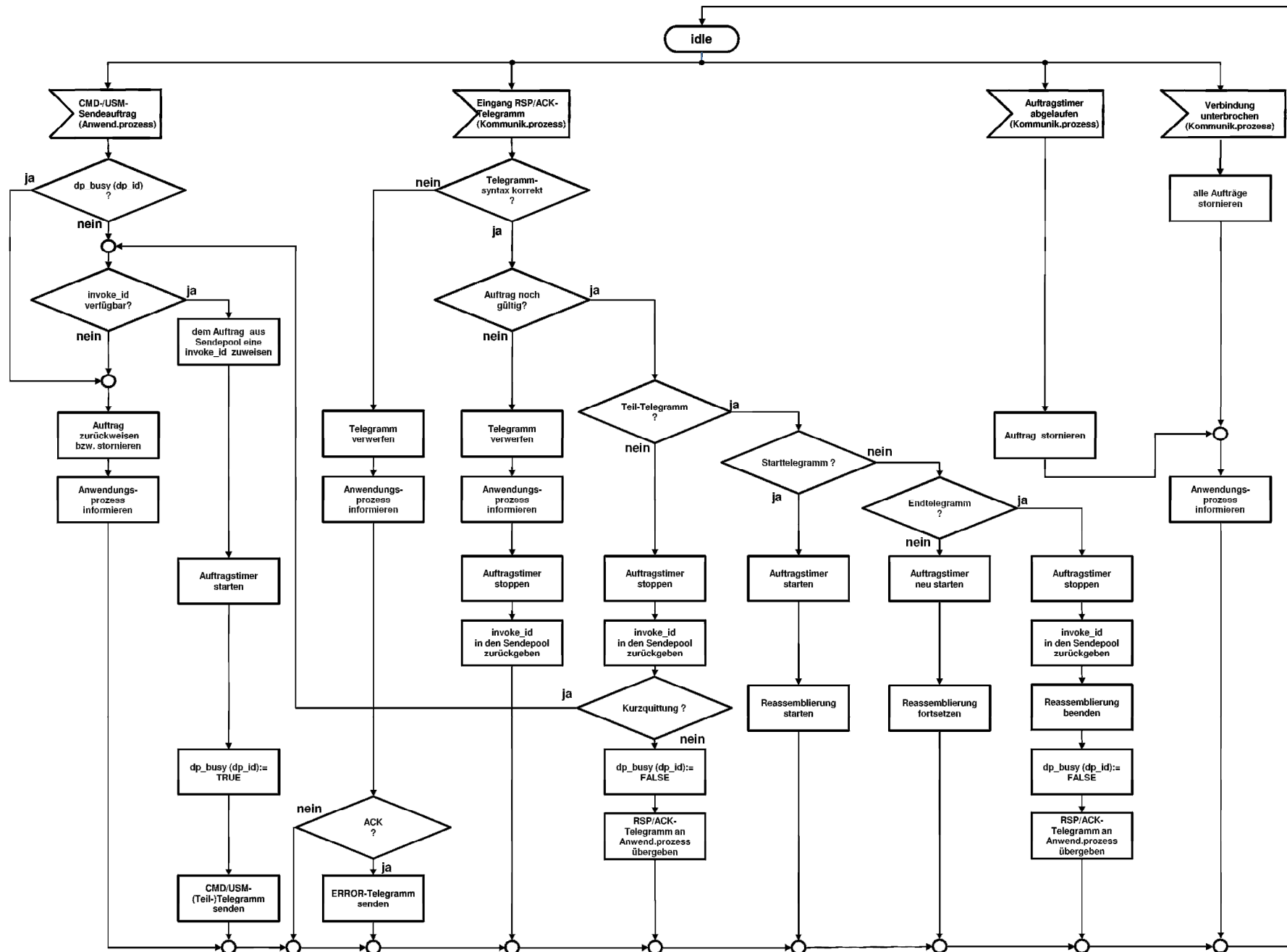


Abb. A-1: Prozedur für CMD/USM-Sendeauftrag

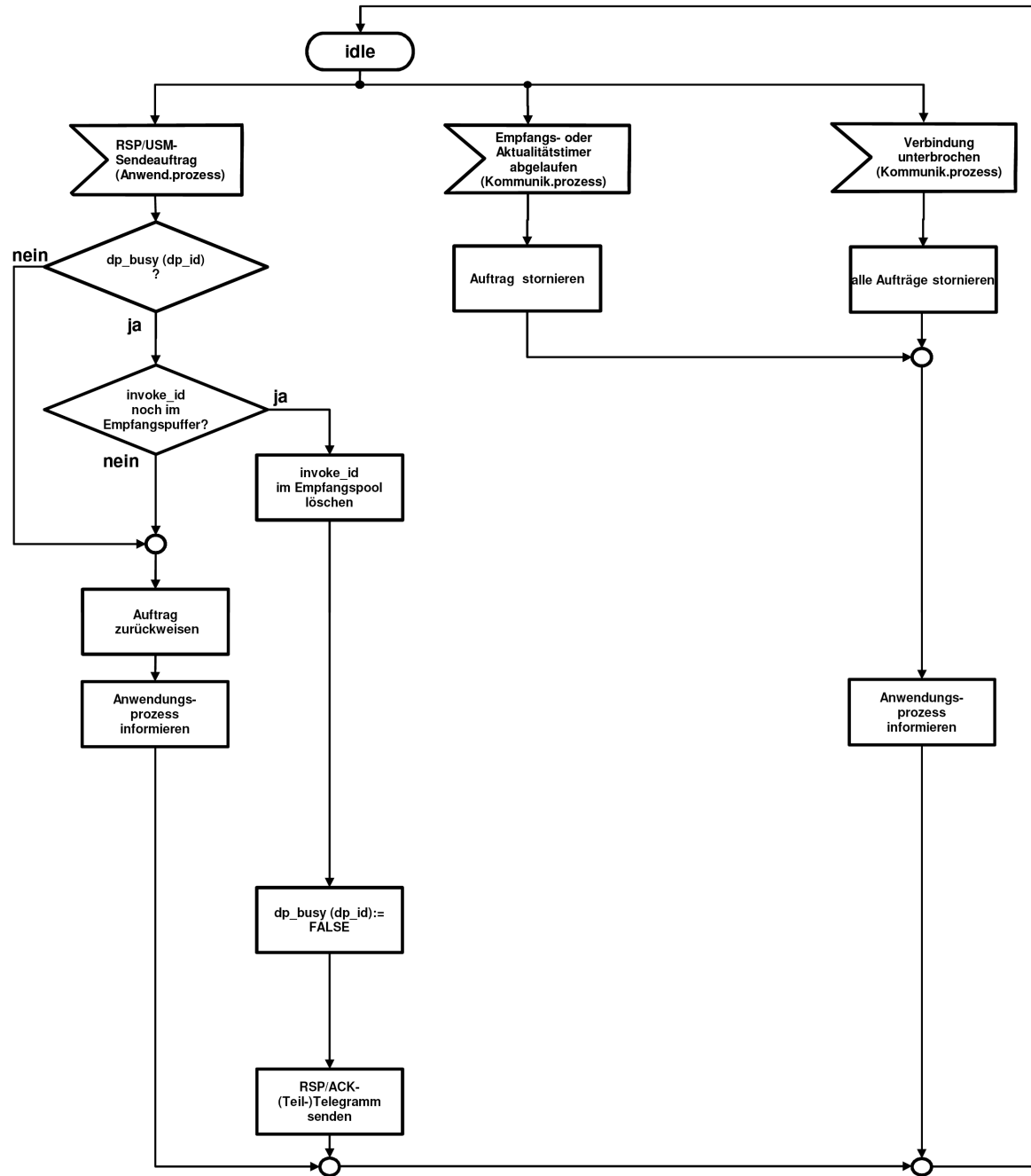


Abb. A-2a: Prozedur für CMD/USM-Empfangsbestätigung (Teil 1)

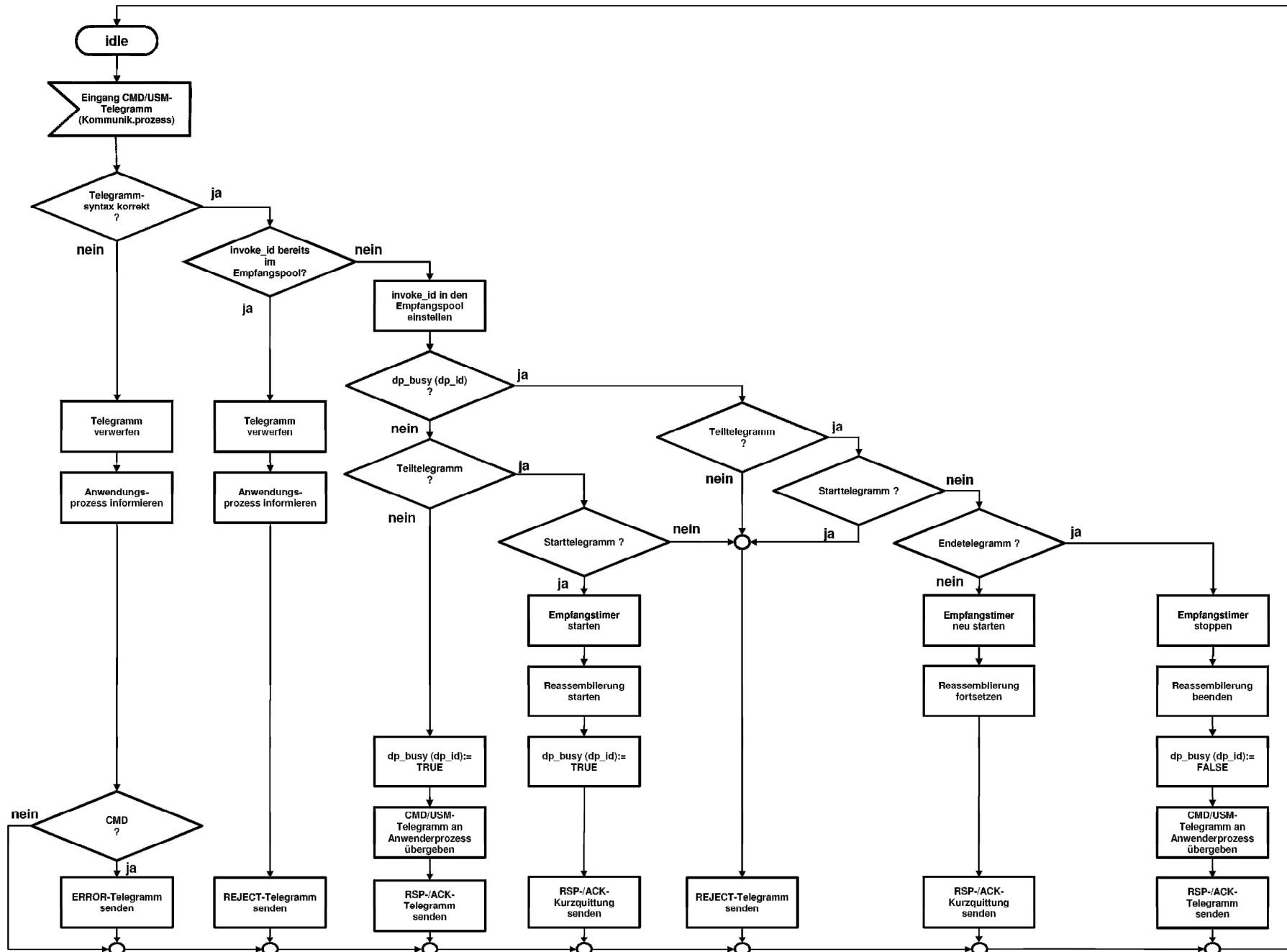


Abb. A-2b: Prozedur für CMD/USM-Empfangsbestätigung (Teil 2)

B Netzwerk-Interface

FND 2009 spezifiziert nur die Anwendungsschicht (Layer 7 – application) des OSI-Schichtenmodells nach ISO/IEC 7498-1. Bezüglich der Ausführung des Datentransportsystems, d. h. der Schichten 1 (physical) bis 4 (transport) werden keine Vorgaben gemacht. Die Schichten 5 (session) und 6 (presentation) sind leer.

Zum Zeitpunkt der Veröffentlichung existieren FND-Verbindungen über TCP/IP-Ethernet-Netze, ISDN-Wählverbindungen und serielle V.24/RS232- Schnittstellen.

C Attribute und Dimensionen

Die nachstehenden Tabellen geben die Belegung der in den Datenpunkt-Beschreibungen mit „atb_id“ (für Melde- und Schaltpunkte) bzw. „dimension“ (für Mess-, Stell- und Zählpunkte) bezeichneten Variablen wieder. Sie sind dabei als Grundversion zu verstehen, die sich durch Erweiterung dem Anforderungsprofil in flexibler Weise anpassen lassen.

Hinweis: Es wird empfohlen, ausschließlich die nachfolgend angegebenen Codierungen einzusetzen, um projektspezifische Vereinbarungen mit ihren aufwendigen Test- und Dokumentations-Problemen zu vermeiden. Die korrekte Abbildung von anlageninternen Zuständen und Werten auf die FND-Datenpunkte – z.B. Invertierung, Umrechnung auf die phys. Dimension – ist Aufgabe der Automationsinsel.

CODIERUNG BINÄRER ATTRIBUTE		
Attributtext		atb_id
STUFE (MODUS)	0	1
STUFE (MODUS)	I	
STUFE (MODUS)	0	2
STUFE (MODUS)	I	
STUFE (MODUS)	II	
STUFE (MODUS)	0	3
STUFE (MODUS)	I	
STUFE (MODUS)	II	
STUFE (MODUS)	III	
STUFE (MODUS)	0	4
STUFE (MODUS)	I	
STUFE (MODUS)	II	
STUFE (MODUS)	III	
STUFE (MODUS)	IV	
STUFE (MODUS)	0	5
STUFE (MODUS)	I	
STUFE (MODUS)	II	
STUFE (MODUS)	III	
STUFE (MODUS)	IV	
STUFE (MODUS)	V	
STUFE (MODUS)	0	6
STUFE (MODUS)	I	
STUFE (MODUS)	II	
STUFE (MODUS)	III	
STUFE (MODUS)	IV	
STUFE (MODUS)	V	
STUFE (MODUS)	VI	

CODIERUNG BINÄRER ATTRIBUTE		
Attributtext		atb_id
STUFE (MODUS)	0	7
STUFE (MODUS)	I	
STUFE (MODUS)	II	
STUFE (MODUS)	III	
STUFE (MODUS)	IV	
STUFE (MODUS)	V	
STUFE (MODUS)	VI	
STUFE (MODUS)	VII	
STUFE (MODUS)	0	8
STUFE (MODUS)	I	
STUFE (MODUS)	II	
STUFE (MODUS)	III	
STUFE (MODUS)	IV	
STUFE (MODUS)	V	
STUFE (MODUS)	VI	
STUFE (MODUS)	VII	
STUFE (MODUS)	VIII	
EIN AUS		10
OFFEN GESCHLOSSEN		11
AUF AB		12
AUF ZU		13
START STOP		14
SETZEN RÜCKSETZEN		15
VOR ZURÜCK		16
AUSGANGSSTELLUNG ENDSTELLUNG		17
TAGBETRIEB NACHTBETRIEB		18

CODIERUNG BINÄRER ATTRIBUTE	
Attributtext	atb_id
SCHNELL LANGSAM	19
HEIZEN KÜHLEN	20
SOMMER WINTER	21
RECHTS LINKS	22
AUTOMATIK HAND	23
AKTIV PASSIV	24
NORMAL ANORMAL	25
NORMAL WARTUNG	26
NORMAL STÖRUNG	27
NORMAL ALARM	28
NORMAL GEFAHR	29
(NORMAL-) BETRIEB INITIALISIERUNG	30
(NORMAL-) BETRIEB OPTIMIERUNG	31
UNTEN MITTE (LSTELLUNG) OBEN	32
AUF MITTE (LSTELLUNG) ZURÜCK	33

CODIERUNG BINÄRER ATTRIBUTE	
Attributtext	atb_id
VOR MITTE (LSTELLUNG) ZURÜCK	34
AUSGANGSSTELLUNG MITTELSTELLUNG ENDSTELLUNG	35
SCHNELL MITTEL LANGSAM	36
SCHNELL LANGSAM AUS	37
HEIZEN NEUTRAL KÜHLEN	38
RECHTS MITTE LINKS	39
RECHTS AUSGANGSSTELLUNG LINKS	40
RECHTS RUHESTELLUNG LINKS	41
RECHTS AUS LINKS	42
AUTOMATIK HAND AUS	43
NORMAL WARNUNG ALARM	44
AUS EIN	45
Montag Dienstag Mittwoch Donnerstag Freitag Samstag Sonntag	165

CODIERUNG PHYSIKALISCHER DIMENSIONEN		
Art	Einheit	dimension
	(dimensionslos)	0
Länge	Millimeter	1
	Meter	2
	Kilometer	3
Fläche	Quadratmeter	6
Volumen	Liter	9
	Kubikmeter	10
Zeit	Sekunde	12
	Minute	13
	Stunde	14
	Tag	15
	Monat	16
	Jahr	17
	Zeit als „hhmm[.ss]“ Datum als „ttmm[jjj]“	18 19
Kraft	Newton	20
	Kilonewton	21
	Meganewton	22
Druck	Hektopascal	24
	Pascal	25
	Millibar	26
	Bar	27
Masse	Milligramm	28
	Gramm	29
	Kilogramm	30
	Tonne	31
Energie, Arbeit	Joule	36
	Kilojoule	37
	Megajoule	38
	Gigajoule	39
	Wattstunde	40
	Kilowattstunde	41
	Megawattstunde	42
Wirkleistung	Watt	48
	Kilowatt	49
	Megawatt	50
Scheinleistung	Voltampere	51
	Kilovoltampere	52
	Megavoltampere	53
Drehzahl	1 / Sekunde	56
	1 / Minute	57
	1 / Stunde	58

CODIERUNG PHYSIKALISCHER DIMENSIONEN		
Art	Einheit	dimension
Winkel	Sekunde	60
	Minute	61
	(Alt) Grad	62
	Neugrad	63
	Radian	64
Geschwindigkeit	Millimeter / Sekunde	68
	Meter / Sekunde	69
	Millimeter / Minute	70
	Meter / Minute	71
	Kilometer / Minute	72
	Millimeter / Stunde	73
	Meter / Stunde	74
	Kilometer / Stunde	75
	Millimeter / Tag	76
	Meter / Tag	77
Volumenstrom	Liter / Sekunde	80
	Liter / Minute	81
	Liter / Stunde	82
	Kubikmeter / Sekunde	83
	Kubikmeter / Minute	84
	Kubikmeter / Stunde	85
Massenstrom	Gramm/Sekunde	90
	Kilogramm / Sekunde	91
	Tonne / Sekunde	92
	Gramm / Minute	93
	Kilogramm / Minute	94
	Tonne / Minute	95
	Gramm / Stunde	96
	Kilogramm / Stunde	97
	Tonne / Stunde	98
Drehmoment	Newtonmeter	100
	Kilonewtonmeter	101
	Meganewtonmeter	102

CODIERUNG PHYSIKALISCHER DIMENSIONEN		
A r t	E i n h e i t	dimension
Temperatur	Grad Celsius	108
Temperatur-Differenz	Kelvin	109
Entropie	Joule / Kelvin	112
	Kilojoule / Kelvin	113
	Megajoule / Kelvin	114
Enthalpie	Joule / Kilogramm	115
	Kilojoule / Kilogramm	116
	Megajoule / Kilogramm	117
Elektrische Spannung	Millivolt	119
	Volt	120
	Kilovolt	121
	Megavolt	122
Elektrischer Strom	Milliampere	124
	Ampere	125
	Kiloampere	126
Elektrischer Widerstand	Milliohm	130
	Ohm	131
	Kiloohm	132
	Megaohm	133
Stellung, Verhältnis, Konzentration ...	parts per million (ppm)	148
	Promille	149
	Prozent	150
Relative Feuchte	Prozent	151
Absolute Feuchte	Gramm / Kilogramm	152

CODIERUNG PHYSIKALISCHER DIMENSIONEN		
A r t	E i n h e i t	dimension
Zeitpunkt	Jahr, absolut seit 01.01.1900	159
	Tag, absolut seit 01.01.1900	160
	Tag, relativ zum aktuellen Jahr	161
	Kalenderwoche, relativ zum akt. Jahr	162
	Tag, relativ zum aktuellen Monat	164
	Tag, relativ zur aktuellen Woche	165
	Stunde, relativ zur aktuellen Woche	166
	Stunde, relativ zum aktuellen Tag	167
	Minute, relativ zum aktuellen Tag	168
	Minute, relativ zur aktuellen Stunde	169
Keine Dimensionsangabe		255

Anmerkungen:

Für Code 18 (Zeit) können optional die Sekunden als Nachkommastellen angegeben werden, z.B. 2359,59 als max. Wert vor dem Tageswechsel. Fehlende Nachkommastellen gelten implizit als 00.

Für Code 19 (Datum) kann optional das Jahr bezogen auf das Jahrhundert angegeben werden, z.B. 250209 für 25. Februar 2009. Ansonsten gilt die Angabe – z.B. 2502 - implizit für das nächste Auftreten des angegebenen Tages.

Für Code 160 (Tag, absolut seit 01.01.1900) ist aufgrund der eingeschränkten Genauigkeit der IEEE 754-Floating-Point-Darstellung nur eine „Rasterschärfe“ ≥ 10 Minuten realisierbar ist. Erweist sich dies für spezielle Anwendungen als ungenügend, so besteht eine Lösungsmöglichkeit z.B. ist der Verwendung zweier assoziierter Zellen, deren erste mittels Code 160 den absoluten Tag, und deren zweite mittels Code 167 (oder 168) den exakten Zeitpunkt innerhalb dieses Tages fixiert.

Für Code 160 ff. sind Nachkommastellen bezogen auf die Einheiten Stunde, Tag bzw. Woche zu interpretieren, d.h. 0,5 bedeutet bei :

- Stundenbezug: 30 Minuten der angefangenen Stunde
- Tagesbezug: 12 Uhr Mittag des Tages
- Wochenbezug: Donnerstag, 12 Uhr der Woche

Code 255 ist für den Fall reserviert, dass die Dimension entweder unbekannt oder implizit vereinbart ist.

D Fehlerbehandlung

Die Variable „tag“ im RSP- oder ACK-Telegramm weist der APDU eine der drei Kategorien zu:

- NORMAL (tag=0: kein Fehler, Normalfall)
- REJECT (tag=1: Zurückweisung beim Ablauf)
- ERROR (tag=2: syntaktischer/semantischer Fehler)

Bei REJECT kann ein korrektes FND-Kommando aufgrund des augenblicklichen Zustands des Datenpunktes (z.B. Verriegelung) nicht ausgeführt werden. Wird dasselbe FND-Telegramm zu einem späteren Zeitpunkt erneut gesendet, so ist eine Ausführung evtl. dann möglich.

Für Telegramme, die wegen eines syntaktischen oder semantischen Fehlers (z.B. unzulässige „fct_id“ oder „dp_id“) mit ERROR abgewiesen werden, macht wiederholtes Senden keinen Sinn.

D.1 REJECT-APDU-Format (Ablauffehler)

Ein REJECT-Telegramm ist ein RSP- oder ACK-Telegramm mit „tag“=1. Der Grund für die Zurückweisung wird im Info-Code wie folgt angegeben:

exc_1	exc_2	Beschreibung
1	0	Unerwarteter Empfang eines Segment-Teil-Telegrammes (seg = 10b oder 01b)
2	0	Unerwarteter Empfang eines Segment-Start- oder Einzel-Telegrammes in der Reassemblierungsphase (seg = 11b oder 00b)
3	seq_exp	Unerwartete Segment-Folge-Nummer „seq_nr“; „seq_exp“ gibt die erwartete Segment-Folge-Nummer an
4	0	„invoke_id“ noch belegt; Bearbeitung eines (anderen) Vorgangs noch nicht abgeschlossen
5	0	Datenpunkt gegenwärtig für FND nicht verfügbar
	1	Datenpunkt lokal bedient
	2	Datenpunkt verriegelt
6	0	Datenpunkt noch belegt; Bearbeitung eines (anderen) Vorgangs noch nicht abgeschlossen
7	0	LZ gegenwärtig nicht verfügbar
8		AI gegenwärtig nicht verfügbar
9	apr	Ungenügende Durchgriffspriorität (prd < apr)

D.2 ERROR-APDU-Format (syntaktische/semantischer Fehler)

Von einer AI empfangene und als syntaktisch fehlerhaft eingestufte Telegramme – z.B. in der AI nicht existierende Datenpunkt-Adresse – stellen gravierende Protokoll-Fehler dar, von denen die LZ über sog. ERROR-Telegramme in Kenntnis gesetzt wird. Der Header der Fehler-APDU wird dabei in den Datenteil eines speziellen, an die LZ gerichteten Telegramms übernommen, welches als Absender die Adresse des AI-„Fault-Managers“ enthält.

Ein ERROR-Telegramm ist ein RSP-Telegramm mit „tag“=2. Der Datenteil enthält den APCI des als syntaktisch falsch eingestuften CMD-Datagrammes. Der Info-Code „exc_1“ bezeichnet das Byte im Datenteil (beginnend mit 0), bei dem der Fehler erkannt wurde. „exc_2“ kann Zusatzinformationen enthalten. Zusätzlich gilt :

Information	Wert	Bedeutung
fct_id	0	Fehlerbehandlung
dp_type	0	* reserviert *
dp_subtype	1	* reserviert *
prd	0	* reserviert *
rpr	0	* reserviert *
emq	0000b	* reserviert *
val	0	* reserviert *
apr	0	* reserviert *

E Übersicht der Datenpunkt-Tabellen

Nachfolgend werden die Datenpunkt-Tabellen kurz zusammengefasst.

E.1 Meldepunkt

Charakterisierung:

Einfach-Meldepunkte stellen binäre (boolesche) Größen dar, welche bei physikalischen Gebern i.a. den Zustand elektrischer Kontakte („offen“/ „geschlossen“) bei logischen (virtuellen) Gebern etwa den Aktivitätszustand eines Prozesses widerspiegeln.

Mehrstufigen Betriebsmeldungen sind bei physikalischen Gebern i.a. die Einzelkontakte eines Stufenschalters, bei logischen (virtuellen) Datenpunkten etwa Ablaufvarianten (Modi) von Zeit- oder Ereignisprogrammen zugeordnet.

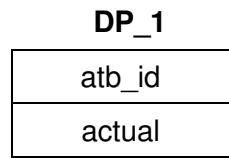
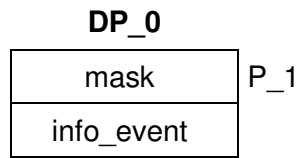
Ursachen für Ereignismeldungen

- Geber-Störung
- BTA-Störung
- Wert-Änderung
- projektspezifisch vereinbartes Ereignis (triggering-event)

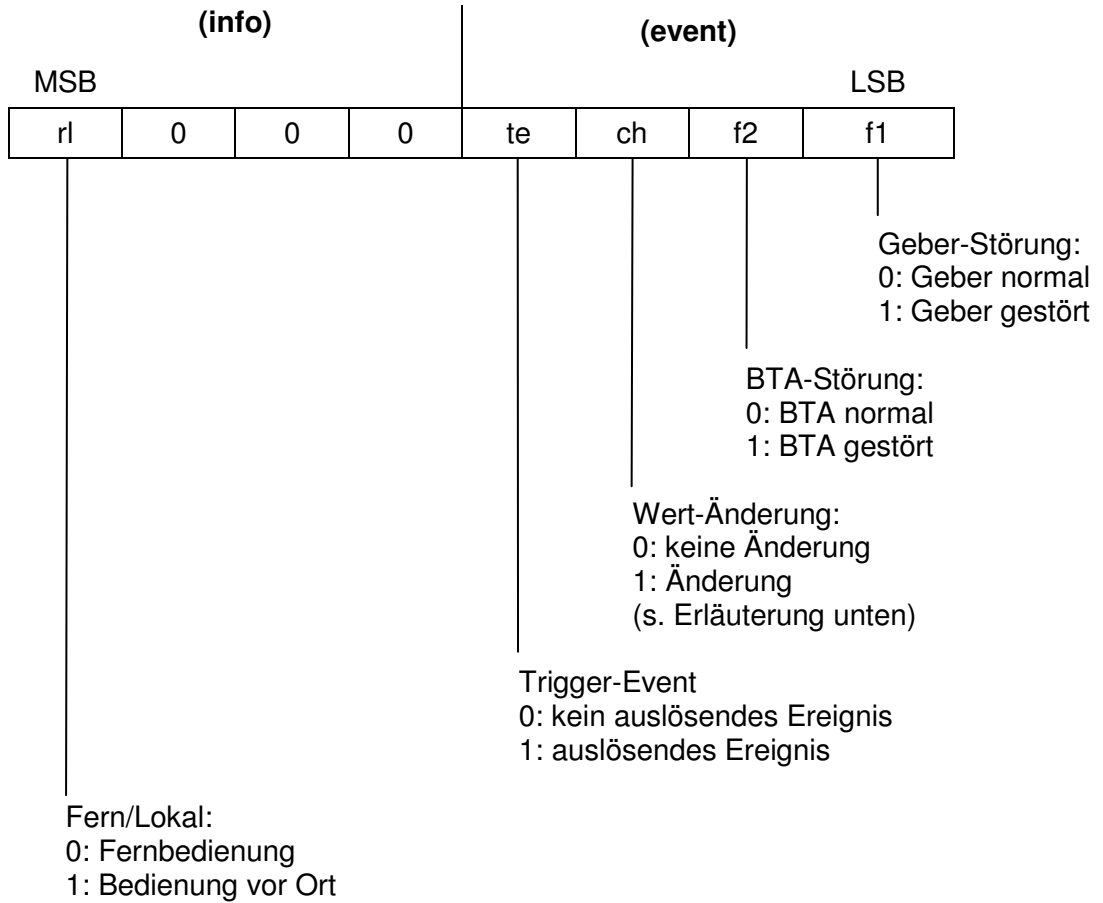
Operationen

- Status anfordern
- Ereignismeldungen sperren/freigeben
- Meldewert lesen

Datenpunkt-Tabellen:



info_event:



E.2 Schaltpunkt (mit Rückmeldung)

Charakterisierung:

Ein Schaltpunkt mit Rückmeldung stellt einen Schaltpunkt mit „integriertem feedback“ dar, d.h. die zugehörige (Betriebs-)Meldung ist über die gleiche Datenpunkt-Adresse verfügbar.

Ein Schaltpunkt mit Rückmeldung entspricht einem Meldepunkt (inkl. Spontanmeldung bei Wertänderung), dem zusätzlich ein Sollwert für das Schalten vorgegeben werden kann. Die AI entscheidet aufgrund der zugeordneten Funktion, ob und wann der Schaltvorgang ausgeführt wird.

Ursachen für Ereignismeldungen

- Geber-Störung
- BTA-Störung
- Schaltwert-Änderung (Rückmeldung)
- projektspezifisch vereinbartes Ereignis (triggering-event)

Operationen

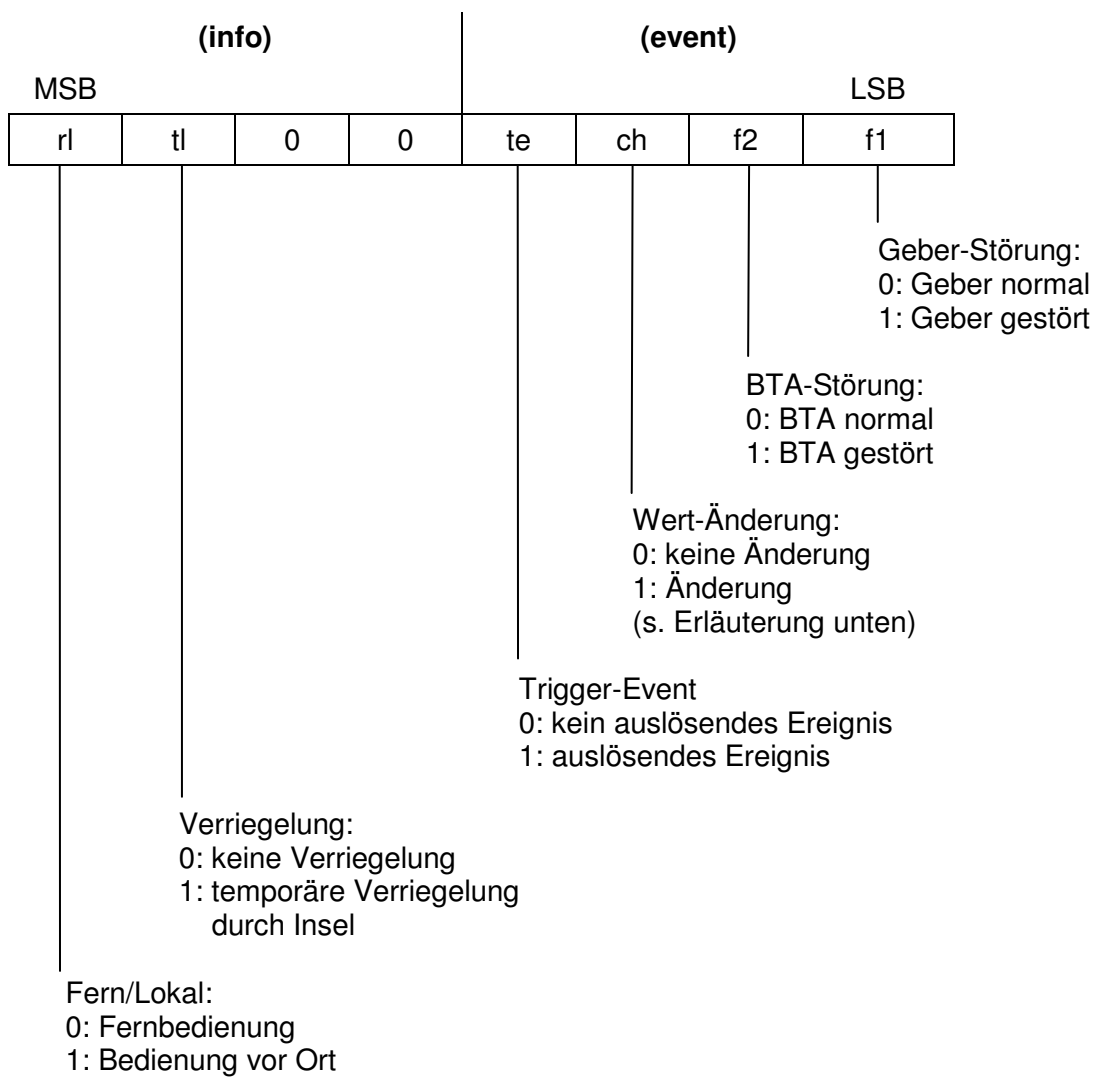
- Status anfordern
- Ereignismeldungen sperren/freigeben
- Schaltvorgabe und Schaltwert lesen
- Schaltvorgabe modifizieren (Schalten)

Datenpunkt-Tabellen:

DP_0	
mask	P_1
info_event	

DP_1	
atb_id	P_1
actual	
nominal	

info_event:



E.3 Messpunkt

Charakterisierung:

Messpunkte sind kontinuierlich veränderbaren Größen zugeordnet, die auf Einhaltung vorgegebener Grenzwerte überwacht werden können. Gemäß der realen oder logischen (virtuellen) Natur des Messwertes kann es sich dabei um eine Temperatur, die Position eines Stellgliedes, aber auch um einen aus mehreren Größen rechnerisch ermittelten Funktionswert handeln.

Ursachen für Ereignismeldungen

- Geber-Störung
- Grenzwert-Über-/Unterschreitung
- projektspezifisch vereinbartes Ereignis (triggering-event)

Operationen

- Status anfordern
- Ereignismeldungen sperren/freigeben
- Messwert lesen
- Grenzwerte lesen
- Grenzwerte modifizieren

Datenpunkt-Tabellen:

DP_0

mask	P_1
info_event	

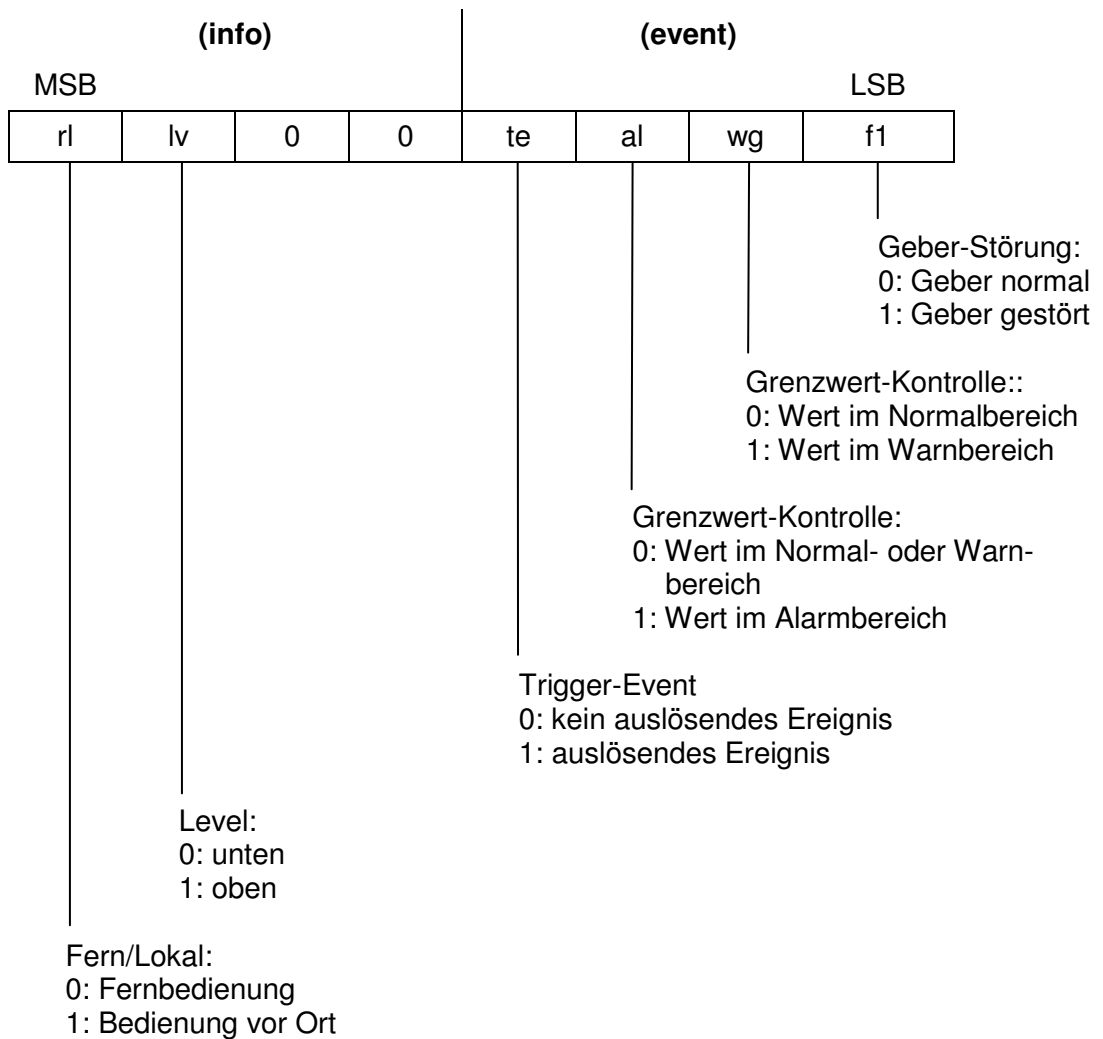
DP_1

dimension
actual

DP_2

update_control	
fixed	
alarm_low	P_1
warning_low	P_2
warning_high	P_3
alarm_high	P_4

info_event:



E.4 Stellpunkt

Charakterisierung:

Bei Stellpunkten führt die Vorgabe eines Stellwerts zu einer entsprechenden Stellung des Gebers. Gemäß dessen realer oder logischer (virtueller) Natur kann es sich dabei sowohl um die Position eines physikalischen Stellglieds, als auch etwa um einen Programm-Parameter handeln.

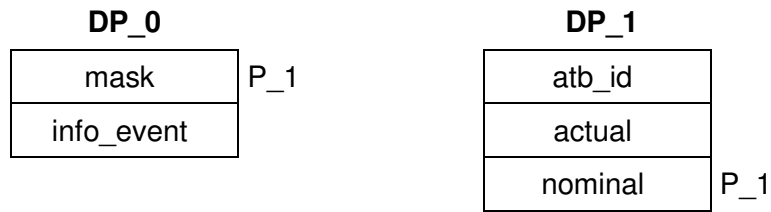
Ursachen für Ereignismeldungen

- Geber-Störung
- projektspezifisch vereinbartes Ereignis (triggering-event)

Operationen

- Status anfordern
- Ereignismeldungen sperren/freigeben
- Stellvorgabe und Stellwert lesen
- Stellvorgabe modifizieren (Stellen)

Datenpunkt-Tabellen:

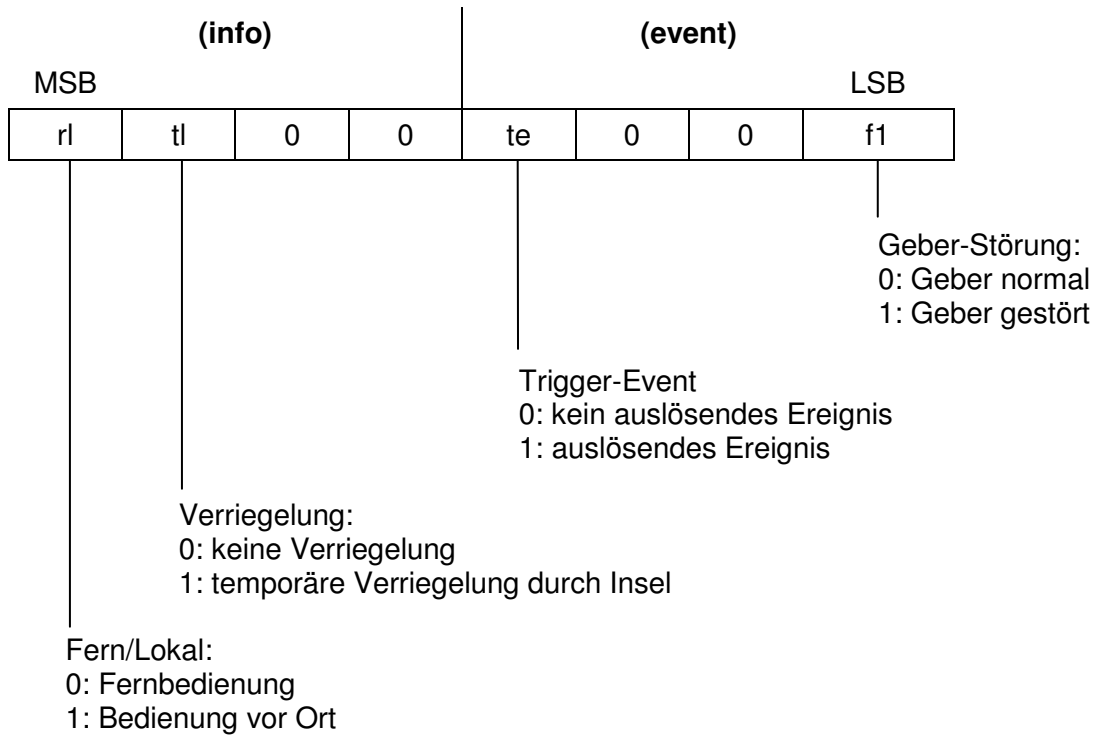


Variablenbeschreibung DP 0:

mask: (P_1)

(Zur Beschreibung dieses Bytes: siehe Abschnitt 3.7, „Sperrungen von Meldungen“)

info_event:



E.5 Zählpunkt

Charakterisierung:

Der Zählpunkt ist einer Größe zugeordnet, die im zeitlichen Verlauf monoton wachsenden Charakter hat. Unstetigkeiten entstehen lediglich durch (Rück-)Setzvorgänge oder Überlauf des Zählers.

Ursachen für Ereignismeldungen

- Geber-Störung
- Grenzwert-Überschreitung
- Wert-Änderung (Rücksetzen oder Überlauf)
- projektspezifisch vereinbartes Ereignis (triggering-event)

Operationen

- Status anfordern
- Ereignismeldungen sperren/freigeben
- Zählwert und gesicherten Wert lesen
- Zählwert setzen
- Grenzwerte lesen
- Grenzwerte modifizieren

Datenpunkt-Tabellen:

DP_0

mask	P_1
info_event	

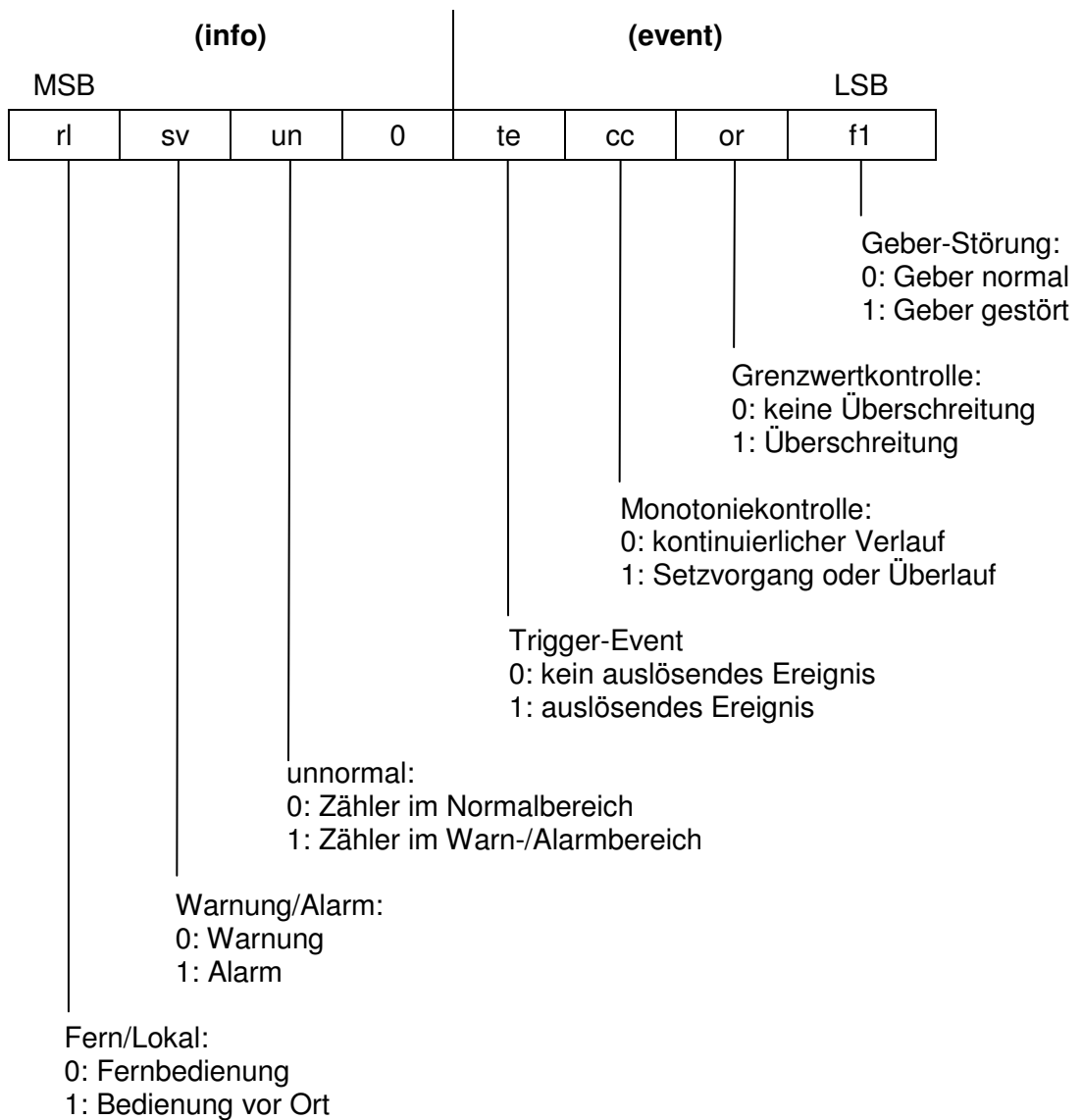
DP_1

atb_id	P_1
actual	
last_save	

DP_2

update_control	P_1
warning_high	
alarm_high	

info_event:



F Übersicht der Codierungen im FND-APDU

Nachfolgend werden nochmals alle wesentlichen Parameter und ihre Codierungen zum FND-Datagramm zusammengefasst. Die zentrale Rolle übernimmt die Datenpunkt-Adresse.

dp_type	Datenpunkt-Typ
0 (0000b)	Fehlerbehandlung
1 (0001b)	Meldepunkt
2 (0010b)	Schaltpunkt (mit Rückmeldung)
3 (0011b)	Messpunkt
4 (0100b)	Stellpunkt
5 (0101b)	Zählpunkt
6 (0110b)	* reserviert *
7 (0111b)	Karteipunkt

tab_id	Datenpunkt – Tabelle
0 (0000b)	DP_0
1 (0001b)	DP_1
2 (0010b)	DP_2
3 (0011b)	DP_1&DP_2
4 (0100b)	DP_3 (nur Karteipunkt)

fct_id	Funktion
1 (0001b)	Lesen
2 (0010b)	Modifizieren (Schreiben)

d_r	Typ	Erläuterung
0 (00b)	CMD	„Command“: Befehl der LZ an eine AI
3 (11b)	RSP	„Response“: Antwort einer AI auf einen Befehl der LZ
2 (10b)	USM	„Unsolicited Message“: Spontanmeldung einer AI an die LZ
1 (01b)	ACK	„Acknowledge“: Bestätigung eines USM-Telegramms

mis/msk	Effekt
0 (00b)	Freigabe aller Ereignismeldungen
1 (01b)	Ausblenden der Betriebs-Meldungen; gemeldet werden Geber- und BTA- Störungen
2 (10b)	Ausblenden der Betriebs- und BTA- Störungs-Meldungen; gemeldet werden Geber- Störungen
3 (11b)	Ausblenden aller Ereignismeldungen

G Format der Referenzdatei

Die Referenzdatei ist eine editierbare Textdatei zur Beschreibung der vom GA-System mittels FND-Protokoll bereitgestellten Datenpunkte und der zugehörigen Zugriffsrechte.

Es gelten folgende Vereinbarungen:

Alle Zeilen werden mittels CarriageReturn (0Dh) und/oder LineFeed (0Ah) hinter dem letzten druckbaren Zeichen abgeschlossen

Alle Zahlen stehen rechtsbündig in den entsprechenden Spalten und repräsentieren in Dezimaldarstellung die entsprechenden Bitkombinationen der FND-Variablen; Leerstellen sind mit Leerzeichen (Blank, 20h) oder führenden Nullen (30h) aufzufüllen

Die Datenpunkt-Adressen (dp_id) und die Datenpunkt-Bezeichnungen stehen linksbündig in der jeweiligen Spalte. Sie beginnen und enden mit einem alphanumerischen Zeichen. Nicht benutzte Stellen werden mit Leerzeichen ausgefüllt.

Die erste Zeile hat folgenden Aufbau:

Spalte	Anfang	Länge	Inhalt	Bemerkungen
1	1	5	Revisionsnummer	
2	7	10	Datum	Format: tt/mm/jjjj
3	18	5	Uhrzeit	Format: hh:mm
4	24	56	Projektname	

Die zweite und alle folgenden Zeilen beschreiben jeweils einen Datenpunkt und haben folgenden Aufbau :

Spalte	Anfang	Länge	Inhalt	Bemerkungen
1	1	5	Lfd. Nummer	beginnend mit 1
2	7	16	dp_id	Datenpunkt-Adresse gemäß Vorgaben des Auftraggebers
3	24	1	dp_type	vgl. Abschnitt 3.4.2
4	26	1	dp_subtype	vgl. Abschnitt 3.4.2
5	28	2	emq DP_0	
6	31	2	emq DP_1	
7	34	2	emq DP_2	
8	37	1	val	vgl. Abschnitt 3.6
9	39	1	msk	vgl. Abschnitt 3.7
10	41	3	update_control	vgl. Abschnitt 3.3.3, 3.3.5 und 3.4
11	45	3	atb_id / dimension	vgl. Anhang C
12	49	23	Datenpunkt-Bezeichnung	Beschreibender Klartext gemäß Vorgaben des Auftraggebers (Struktur, Abkürzungen usw.)
13	73	2	Messnutzbereich	Projekt-spez. kodiert
14	76	5	* reserviert *	
15	82		projekt-spez. Ergänzungen	z.B. GA-Knoten: technische Adresse usw. zur Parametrierung des eingebundenen AI-Treibers (FND-SSA/LIB)

H Zahlendarstellung im Gleitkomma-Format nach IEEE 754

Analogwerte werden im Einfach-Format (Single-Precision, 32-Bit) nach IEEE 754 dargestellt.

MSB		LSB
sign	exponent	Byte 0
exp	fraction	Byte 1
	fraction	Byte 2
	fraction	Byte 3

Die Werte $+\infty$ und $-\infty$ werden die Bitmuster 7F800000h und FF800000h repräsentiert.

Für die Reihenfolge der Bytes im Telegramm gilt, dass das signifikanteste Byte – bestehend aus Vorzeichenbit (sign) und den 7 signifikantesten Bits des Exponenten (exponent) – zuerst zu übertragen ist, gefolgt von den anderen Bytes in der Reihenfolge abnehmender Signifikanz.

Dies ist insbesondere bei Software für x86-kompatible Prozessoren zu beachten, da hier intern die umgekehrte Reihenfolge verwendet wird.

5 Literaturverzeichnis

- [1] AMEV 2005.: Hinweise für Planung, Ausführung und Betrieb der Gebäudeautomation in öffentlichen Gebäuden (Gebäudeautomation 2005)
- [2] Bundesbaublatt 11/1998, Ltd. BD Helmut Schmidt, Dipl.-Ing. Heinz Funk und Dipl.-Inf. Wolfgang Fries: Das Münchner-Energie-Management-System, Dezentraler Aufbau des Netzwerks spart Kosten

6 Mitarbeiter

Bawej	Landeshauptstadt München
Bentrup	Stadt Düsseldorf
Prof. Büchel	Fachhochschule Gelsenkirchen
Crecelius	Energie- & Umweltbüro e.V., Berlin
Prof. Fetzer	Hochschule Esslingen
Fries	Daten-Technik FRIES, Dachau
Gerhard	Handwerkskammer des Saarlandes, Saarbrücken
Dr. Hall	Vermögen und Bau Baden-Württemberg, Stuttgart (Obmann)
Kaps	Landeshauptstadt München
Maurer	Energie- & Umweltbüro e.V., Berlin
Person	HIS Hochschul-Informationen-System GmbH, Hannover
Röhnert	Stadt Düsseldorf
Roßbach	Energie- & Umweltbüro e.V., Berlin
Speelmanns	Bundesinstitut für Bau-, Stadt- und Raumforschung, Bonn
Stelzer	Hochschule Esslingen
von der Pütten	Hansestadt Lüneburg

