

Dresdner Verkehrsbetriebe AG

Dynamisches Fahrgastinformationssystem

Schnittstellenspezifikation (2)

Anzeiger

1	Historie	4
2	Einleitung.....	4
2.1	Zweck des Dokumentes	4
2.2	Inhalt.....	4
3	Telegramminhalte	4
3.1	Telegrammaufbau	4
3.2	Telegrammstruktur	5
3.3	Aufbau der Datenblöcke	5
3.4	Blockdefinition	5
3.4.1	Übersicht	5
3.4.2	Positionieren von Text und Grafik.....	7
3.4.2.1	Definition Textfeld - Koordinatenangabe in Pixel	7
3.4.2.2	Definition Textzeile	7
3.4.2.3	Startposition für Text – Koordinaten in Pixel	7
3.4.2.4	Definition Textfeld - Koordinaten in Charakter	8
3.4.2.5	Definition Textfeld mit Höhe - Koordinaten in Pixel	8
3.4.2.6	Startposition für Text – Koordinaten in Charakter	8
3.4.3	Information darstellen	9
3.4.3.1	Text darstellen	9
3.4.3.2	Grafik anzeigen	9
3.4.3.3	Uhr darstellen	9
3.4.4	Information löschen	9
3.4.4.1	Rechteck löschen - Koordinaten in Pixel.....	9
3.4.4.2	Zeile / Feld löschen.....	9
3.4.4.3	Zeile löschen - Koordinaten in Pixel.....	10
3.4.4.4	Anzeige löschen	10
3.4.4.5	Rechteck löschen - Koordinaten in Charakter.....	10
3.4.4.6	Grafik löschen	10
3.4.5	Betriebsinformationen.....	10
3.4.5.1	Uhrzeit synchronisieren	10
3.4.5.2	Initialisierung.....	10
3.4.5.3	Standby-Betriebsart.....	11
3.4.5.4	Tag / Nacht - Umschaltung	11
3.4.5.5	Sondermaske	11
3.4.5.6	Textattribute setzen	11
3.4.6	Grafische Elemente darstellen.....	11
3.4.6.1	Horizontale Linie	11
3.4.6.2	Vertikale Linie	11
3.4.6.3	Block definieren	12
3.4.7	Parameter übertragen	12
3.4.7.1	Master – Timeout.....	12
3.4.7.2	Protokoll – Timeout.....	12
3.4.7.3	Wechselzeit.....	12
3.4.7.4	Diverse Parameter setzen	13
3.4.8	Datenversorgung laden	13
3.4.8.1	Datenversorgung löschen.....	13
3.4.8.2	Abspeichern von Feldern	13
3.4.8.3	Abspeichern von Zeichensätzen	13
3.4.8.4	Grafik.....	14
3.4.8.5	Grafikfeld definieren	15
3.4.8.6	Grafik definieren	16
3.4.8.7	Grafik als Bilddatei-Inhalt übertragen.....	16
3.4.9	Abfragen.....	17
3.4.9.1	Abfrage der Versionsnummer.....	17

3.4.9.2	Abfrage des Grafikstatus	17
3.4.10	Servicefunktionen	17
3.4.10.1	Selbsttest.....	17
4	Erläuterungen	18
4.1	Textaufbau	18
4.2	Textattribute	18
4.3	Darstellen der Steuerzeichen.....	19
5	Statusabfrage	19
5.1	Statusanfrage	19
5.2	Status- und Datensatzantwort.....	20
6	Statusinformationen.....	21
6.1	Abfrage Umgebungslicht	21
6.2	Abfrage Feuchte	21
6.3	Abfrage Betriebsstunden	22
7	Broadcast Uhrzeit / Datum.....	22
8	Protokoll	22
9	Prüfsumme (CRC)	23
10	Datenversorgung Aushangfahrplan.....	23
11	Codierung	24
12	Schnittstellendefinition.....	24

1 Historie

Version	Datum	Bearbeiter	Änderungsgrund
V 00.01	15.12.04	de	Erstellt
V 00.02	10.01.05	dvb / de	Adresse nur ein Byte Broadcast für Uhrzeit / Datum eingefügt
V 01.00	09.05.05	dvb / de / ep	Kein Abspeichern von Texten %81 %12 eingefügt Status angepasst
V 01.01	11.05.05	Dvb / de	Status korrigiert %31 Hintergrundsteuerung gelöscht
V 01.02	16.09.05	dvb / de / ep	Umstellung auf 7 Datenbit Änderung Angabe Telegrammlänge
V 01.03	29.01.07	de	Prüfsummenberechnung ergänzt
V 01.04	28.01.10	de	Pkt. 3.4.8.4 Grafik ergänzt
V 01.05	27.07.10	de	Löschen einer Grafik %25 Abfrage Grafikstatus %99 Schnittstellendefinition angepasst
V 01.06	30.07.10	de	Telegramm %99, Statusrückmeldung geändert Beschreibung Ablauf Aushangfahrplan ergänzt
V 01.07	14.01.16	de	Abfrage Feuchtigkeit und Betriebsstunden ergänzt
V 01.09	23.01.19	HJ-de / dvb	6.1, 6.2, 6.3 zwei Stellen in der Antwort gelöscht

2 Einleitung

2.1 Zweck des Dokumentes

Dieses Dokument beschreibt Anforderungen, die von den DVB an Anzeigen gestellt werden, die in das Fahrgastinformationssystem integriert werden sollen.

Der Anzeigenlieferant muss diese Spezifikation bei der Erstellung der Offerte berücksichtigen und die DVB über Abweichungen informieren.

2.2 Inhalt

Die Spezifikation beschreibt das Protokoll zwischen Knotenrechner und Anzeiger.

Die Anzeigen müssen als Punktmatrix oder zeilenorientiert aufgebaut sein. Das Protokoll unterstützt beide Formate.

Der Anzeige sollte ein modulares Konzept zugrunde liegen. Es sollen Panels beliebiger Größe angesteuert werden (typisch: 3-zeilig, 5-zeilig und doppelseitig, ...)

3 Telegramminhalte

3.1 Telegrammaufbau

Das Telegramm zur Ansteuerung der Anzeigen enthält gestalterische Möglichkeiten (z.B. beliebiges Positionieren eines Textes, Lauftext etc.). Jedes Telegramm ist durch den Anzeiger mit einer Statusmeldung zu quittieren.

Punktmatrixanzeigen und zeichenorientierte Anzeigen werden mit identischen Datensätzen angesprochen, um alternative Anzeigentechnologien zu ermöglichen.

Das Protokoll ist für Mehrpunkt - Aufruf - Betrieb mit Master/Slave Konfiguration vorgesehen.

Die Initiative geht immer vom Master (Knotenrechner) aus, ein Anzeiger meldet sich nur, wenn im Prozedurablauf eine Antwort erwartet wird.

Jede Anzeige erhält eine eindeutige Adresse. Alle vom Knotenrechner zur Anzeige gesendeten Telegramme sind durch ein Statustelegramm zu quittieren. Die Antwort muss in einer im System definierten Zeit erfolgen. Der default-Wert beträgt 50ms und ist parametrierbar.

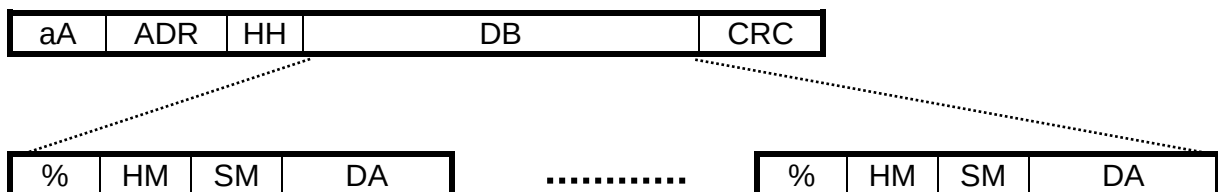
Aufbau eines Telegramms



Kürzel	Bezeichnung	Länge	Beschreibung
AA	Startkennung	2 Byte	Kennzeichnung für Telegrammanfang ,aA'
ADR	Adresse	1 Byte	Adresse des Anzeigers
HH	Teelegrammlänge	2 Byte	Alle Byte der Datenblöcke zwei Hex.-Zeichen (max. 255 = FF)
DB	Datenblöcke	n Byte	Nutzinformationen
CRC	CR + Längsparität	2 Byte	carriage return und CRC (alle Byte des Telegramm ohne CRC)

3.2 Telegrammstruktur

Telegramm



Datenblöcke

3.3 Aufbau der Datenblöcke

DB kann mehrere Datenblöcke nacheinander enthalten.

Aufbau eines Datenblocks:



Kürzel	Bezeichnung	Länge	Beschreibung
%	Startkennung	1 Byte	Kennzeichnung für Blockanfang ,%'
HM	Hauptmode	1 Byte	Blockart
SM	Submode	1 Byte	Ergänzende Spezifikation
DA	Daten	n Byte	Nutzinformationen

Hinweis: Wird das ,%' im Bereich Daten benötigt, muss ein ,%' zusätzlich vorangestellt werden.

3.4 Blockdefinition

3.4.1 Übersicht

Der Koordinatennullpunkt befindet sich oben links auf der Anzeige. Er hat die Koordinaten (1 / 1)

HM	SM	Länge [Byte]	Beschreibung	Gliederung
Positionieren von Text und Grafik				
0	0	13	Definition Textfeld mit Koordinatenangabe in Pixel	3.4.2.1
0	1	7	Definition Textzeile	3.4.2.2
0	2	9	Startposition für Text – Koordinaten in Pixel	3.4.2.3
0	3	13	Definition Textfeld - Koordinaten in Charakter	3.4.2.4
0	4	16	Definition Textfeld mit Höhe - Koordinaten in Pixel	3.4.2.5
0	5	9	Startposition für Text – Koordinaten in Charakter	3.4.2.6
Information darstellen				
1	0	> 3	Text darstellen	3.4.3.1
1	2	2	Grafik anzeigen	3.4.3.2
1	3	4	Uhr darstellen	3.4.3.3
Information löschen				
2	0	15	Rechteck löschen - Koordinaten in Pixel	3.4.4.1
2	1	6	Zeile / Feld löschen	3.4.4.2
2	2	6	Zeile löschen - Koordinaten in Pixel	3.4.4.3
2	3	3	Anzeige löschen	3.4.4.4
2	4	15	Rechteck löschen - Koordinaten in Charakter	3.4.4.5
2	5	2	Grafik löschen	
Betriebsinformationen				
3	0	9	Uhrzeit synchronisieren	3.4.5.1
3	2	3	Initialisierung	3.4.5.2
3	3	4	Standby-Betriebsart	3.4.5.3
3	4	4	Tag / Nacht - Umschaltung	3.4.5.4
3	5	3	Sondermaske	3.4.5.5
3	6	>3	Textattribute setzen	3.4.5.6
Grafische Elemente darstellen				
4	0	13	Horizontale Linie	3.4.6.1
4	1	13	Vertikale Linie	3.4.6.2
4	2	15	Block definieren	3.4.6.3
Parameter übertragen				
7	0	6	Master – Timeout	3.4.7.1
7	1	6	Protokoll – Timeout	3.4.7.2
7	2	6	Wechselzeit	3.4.7.3
7	4	>6	Diverse Parameter setzen	3.4.7.4
Datenversorgung laden				
8	0	4	Datenversorgung löschen	3.4.8.1
8	2	10	Abspeichern von Feldern	3.4.8.2
8	3	>13	Abspeichern von Zeichensätzen	3.4.8.3
8	4	17	Grafikfeld definieren	3.4.8.5
8	5	>8	Grafik definieren	3.4.8.6
Abfragen				
9	0	3	Abfrage der Versionsnummer	3.4.9.1
9	8	2	Abfrage Statusinformationen	6.
9	9	2	Abfrage des Grafikstatus'	3.4.9.2
10	0	3	Selbsttest	3.4.10.1

HM und SM sind in ASCII codiert (,0' = 0x30)

Darstellungsattribute (DA)

- | | |
|---|--------------------------------------|
| 1 (0x31) = rechtsbündig | 2 (0x32) = mittig |
| 3 (0x33) = linksbündig | 4 (0x34) = Lauftext |
| 5 (0x35) = Teilttextsequenz | 6 (0x36) = Rollen |
| 7 (0x37) = Lauftext schnell | 8 (0x38) = Teilttextsequenz schnell, |
| 9 (0x39) = Rollen schnell | : (0x3A) = Abspanntext rechtsbündig, |
| ; (0x3B) = Abspanntext mittig | < (0x3C) = Abspanntext linksbündig |
| = (0x3D) = Abspanntext, rechtsbündig, schnell | |
| > (0x3E) = Abspanntext, mittig, schnell | |
| ? (0x3F) = Abspanntext, linksbündig, schnell | |

3.4.2 Positionieren von Text und Grafik**3.4.2.1 Definition Textfeld - Koordinatenangabe in Pixel**

Es wird ein Darstellungsfeld für den nachfolgenden Text mit dem angegebenen unteren linken Eckpunkt (X1,Y), dem angegebenen unteren rechten Eckpunkt (X2, Y) und dem angegebenen Darstellungs-Attribut (DA) definiert.

Alle Zeichen, welche in diesem Anzeigen-Bereich stehen, werden gelöscht. Die Feldhöhe ist durch den aktiven Zeichensatz definiert. Für den Abspanntext wird besser SM4 verwendet.

Wenn der Lauftext in diesem Anzeigen-Bereich genug Platz hat, wird er statisch linksbündig dargestellt.

%	0	0	X1H	X1M	X1L	X2H	X2M	X2L	YH	YM	YL	DA
---	---	---	-----	-----	-----	-----	-----	-----	----	----	----	----

X1H X1M X1L : X1, dreistellig, dezimal
 X2H X2M X2L : X2, dreistellig, dezimal
 YH YM YL : Y, dreistellig, dezimal
 DA : Formatangabe, hexadezimal

3.4.2.2 Definition Textzeile

Definition der Zeile und des Feldes.

Alle Zeichen die in diesem Feld stehen werden gelöscht

%	0	1	ZH	ZL	FZ	DA
---	---	---	----	----	----	----

ZH ZL : Zeilennummer, zweistellig, dezimal
 FZ : Feldnummer, einstellig, dezimal
 DA : Formatangabe, hexadezimal

Es wird die Zeile definiert: ZZ [01 .. 99]

Es wird das Feld definiert: F [1 .. 9] (Die Anzeige ist Feld-orientiert)

Es wird das Darstellungsattribut definiert (DA siehe 3.4.1)

3.4.2.3 Startposition für Text – Koordinaten in Pixel

Es wird die Startposition für nachfolgenden Text definiert: Y = Zeile / X = Startposition

Bei dieser Textpositionierungsmethode muss der Sender anhand der Textlänge sicherstellen, dass keine anderen Informationen auf derselben Zeile überschrieben werden (z.B. Zieltext überschreibt Abfahrtszeit)

%	0	2	XH	XM	XL	YH	YM	YL
---	---	---	----	----	----	----	----	----

XH XM XL : X dreistellig, dezimal

YH YM YL : Y dreistellig, dezimal

3.4.2.4 Definition Textfeld - Koordinaten in Charakter

Es wird ein Darstellungsfeld für den nachfolgenden Text mit dem angegebenen unteren linken Eckpunkt (X1,Y), dem angegebenen unteren rechten Eckpunkt (X2, Y) und dem angegebenen Darstellungs-Attribut (DA) definiert.

Alle Zeichen, welche in diesem Anzeigen-Bereich stehen, werden gelöscht. Die Feldhöhe ist durch den aktiven Zeichensatz definiert. Für den Abspanntext wird besser SM4 verwendet. Wenn der Lauftext in diesem Anzeigen-Bereich genug Platz hat, wird er statisch linksbündig dargestellt.

%	0	3	X1H	X1M	X1L	X2H	X2M	X2L	YH	YM	YL	DA
---	---	---	-----	-----	-----	-----	-----	-----	----	----	----	----

X1H X1M X1L : X1, dreistellig, dezimal

X2H X2M X2L : X2, dreistellig, dezimal

YH YM YL : Y, dreistellig, dezimal

DA : Formatangabe, hexadezimal

3.4.2.5 Definition Textfeld mit Höhe - Koordinaten in Pixel

Es wird ein Darstellungsfeld für den nachfolgenden Text mit dem angegebenen oberen linken Eckpunkt (X1,Y1), dem unteren rechten Eckpunkt (X2, Y2) und dem Darstellungs-Attribut (DA) definiert.

Alle Zeichen, welche in diesem Anzeigen-Bereich stehen, werden gelöscht.

Wenn der Lauftext in diesem Anzeigen-Bereich genug Platz hat, wird er statisch linksbündig dargestellt.

%	0	4	X1H	X1M	X1L	X2H	X2M	X2L				
			Y1H	Y1M	Y1L	Y2H	Y2M	Y2L	DA			

X1H X1M X1L : X1 dreistellig, dezimal

X2H X2M X2L : X2 dreistellig, dezimal

Y1H Y1M Y1L : Y1 dreistellig, dezimal

Y2H Y2M Y2L : Y2 dreistellig, dezimal

DA : Formatangabe, hexadezimal

3.4.2.6 Startposition für Text – Koordinaten in Charakter

Es wird die Startposition für nachfolgenden Text definiert: Y = Zeile / X = Startposition

Bei dieser Textpositionierungsmethode muss der Sender anhand der Textlänge sicherstellen, dass keine anderen Informationen auf derselben Zeile überschrieben werden (z.B. Zieltext überschreibt Abfahrtszeit)

%	0	5	XH	XM	XL	YH	YM	YL
---	---	---	----	----	----	----	----	----

XH XM XL	:	X	dreistellig, dezimal
YH YM YL	:	Y	dreistellig, dezimal

3.4.3 Information darstellen

3.4.3.1 Text darstellen

Text in Form einer Zeichenkette (siehe Kapitel 4.1 Textaufbau, Seite 18)

[illegible]

C1 .. Cn maximal 245 Zeichen

3.4.3.2 Grafik anzeigen

ZH ZL : Grafiknummer (01..99), zweistellig, dezimal

%	1	2	ZH	ZL
---	---	---	----	----

3.4.3.3 Uhr darstellen

Darstellen der Anzeige-Internen-Uhr (Darstellung erfolgt durch die Anzeige).

%	1	3	M
---	---	---	---

Mode (M): 1 = hh:mm
2 = hh.mm
3 = hh:mm:ss
4 = hh:mm:ss

3.4.4 Information löschen

3.4.4.1 Rechteck löschen - Koordinaten in Pixel

Es wird ein Rechteck-Bereich (x_1, y_1 / x_2, y_2 = Diagonal-Eckpunkte) gelöscht

%	2	0	X1H	X1M	X1L	X2H	X2M	X2L
			Y1H	Y1M	Y1L	Y2H	Y2M	Y2L

X1H X1M X1L	:	X1	dreistellig, dezimal
X2H X2M X2L	:	X2	dreistellig, dezimal
Y1H Y1M Y1L	:	Y1	dreistellig, dezimal
Y2H Y2M Y2L	:	Y2	dreistellig, dezimal

3.4.4.2 Zeile / Feld löschen

Definition der Zeile und des Feldes.

Alle Zeichen die in diesem Feld stehen werden gelöscht

%	2	1	ZH	ZL	FZ
---	---	---	----	----	----

ZH ZL : Zeile (01..99), zweistellig, dezimal

FZ : Feld (0..9), einstellig, dezimal

Beim Senden von FZ=0 wird die gesamte Zeile gelöscht.

3.4.4.3 Zeile löschen - Koordinaten in Pixel

Es wird die Zeile an der Position Y gelöscht (bei Matrix-Anzeigen ergibt sich die Zeilenhöhe aus dem aktiven Zeichengröße)

%	2	2	YH	YM	YL
---	---	---	----	----	----

YH YM YL : Y dreistellig, dezimal

3.4.4.4 Anzeige löschen

Es wird die gesamte Anzeige gelöscht.

%	2	3
---	---	---

3.4.4.5 Rechteck löschen - Koordinaten in Charakter

Es wird ein Rechteck-Bereich (x1,y1 / x2,y2 = Diagonal-Eckpunkte) gelöscht

%	2	4	X1H	X1M	X1L	X2H	X2M	X2L
			Y1H	Y1M	Y1L	Y2H	Y2M	Y2L

X1H X1M X1L : X1 dreistellig, dezimal

X2H X2M X2L : X2 dreistellig, dezimal

Y1H Y1M Y1L : Y1 dreistellig, dezimal

Y2H Y2M Y2L : Y2 dreistellig, dezimal

3.4.4.6 Grafik löschen

Mit Hilfe dieses Telegramms können einzelne Grafiken (Grafiknummer 01 bis 99) oder alle Grafiken (Grafiknummer 00) gelöscht werden. Dabei wird die Anzeige gelöscht und es werden alle Grafikdaten und –definitionen im Speicher gelöscht.

ZH ZL : Grafiknummer (00..99), zweistellig, dezimal

%	2	5	ZH	ZL
---	---	---	----	----

3.4.5 Betriebsinformationen

3.4.5.1 Uhrzeit synchronisieren

%	3	0	HH	HL	MH	ML	SH	SL
---	---	---	----	----	----	----	----	----

HH HL : Stunden, zweistellig, dezimal

MH ML : Minuten zweistellig, dezimal

SH SL : Sekunden, zweistellig, dezimal

3.4.5.2 Initialisierung

Es wird die gesamte Anzeige initialisiert : Anzeige löschen, Default-Attribute setzen

%	3	2
---	---	---

3.4.5.3 Standby-Betriebsart

Standby-Betriebsart wird eingestellt

%	3	3	N
---	---	---	---

N: 0 = Standard
 1 = Standby

3.4.5.4 Tag / Nacht - Umschaltung

Tag- oder Nachtbetriebsart wird eingestellt.

%	3	4	N
---	---	---	---

N: 0 = Tag
 1 = Nacht

3.4.5.5 Sondermaske

Anzeige löschen, Default-Attribute setzen, Sondermaske darstellen.

%	3	5
---	---	---

3.4.5.6 Textattribute setzen

Die angegebenen Textattribute werden als Default eingestellt (siehe Kapitel 4.2 Textattribute, Seite 18)

[illegible]

T1 .. Tn Textattribute (maximal 245 Zeichen)

3.4.6 Grafische Elemente darstellen

3.4.6.1 Horizontale Linie

Es wird eine horizontale Linie definiert und die Strichstärke vorgegeben.

%	4	0	X1H	X1M	X1L	X2H	X2M	X2L	YH	YM	YL	LD
---	---	---	-----	-----	-----	-----	-----	-----	----	----	----	----

X1H X1M X1L	:	Startposition	X1,	dreistellig, dezimal
X2H X2M X2L	:	Endposition	X2,	dreistellig, dezimal
YH YM YL	:	vertikale Position	Y,	dreistellig, dezimal
LD	:	Strichstärke (1 = 1 Pixel),		dezimal

3.4.6.2 Vertikale Linie

Es wird eine vertikale Linie definiert und die Strichstärke vorgegeben.

%	4	1	XH	XM	XL	Y1H	Y1M	Y1L	Y2H	Y2M	Y2L	LD
---	---	---	----	----	----	-----	-----	-----	-----	-----	-----	----

XH XM XL : Horizontale Position X, dreistellig, dezimal
 Y1H Y1M Y1L : Startposition Y1, dreistellig, dezimal
 Y2H Y2M Y2L : Endposition Y2, dreistellig, dezimal
 LD : Strichstärke (1 = 1 Pixel), dezimal

3.4.6.3 Block definieren

Es wird ein Block definiert (x1,y1 / x2,y2 = Diagonal-Eckpunkte) und der Inhalt gelöscht.

%	4	2	X1H	X1M	X1L	X2H	X2M	X2L
			Y1H	Y1M	Y1L	Y2H	Y2M	Y2L

X1H X1M X1L : X1 dreistellig, dezimal
 X2H X2M X2L : X2 dreistellig, dezimal
 Y1H Y1M Y1L : Y1 dreistellig, dezimal
 Y2H Y2M Y2L : Y2 dreistellig, dezimal

3.4.7 Parameter übertragen

3.4.7.1 Master – Timeout

Es wird das Master-Timeout definiert (001 bis 999 Sekunden). Nach abgelaufenem Timeout bei fehlendem Master wird die Rückfalldarstellung angezeigt. Bei 000 wird die Anzeige auch bei fehlendem Master belassen)

Wenn diese Information nicht zum Anzeiger gesendet wird, soll der Default-Timeout 150 Sekunden betragen.

%	7	0	TH	TM	TL
---	---	---	----	----	----

TH TM TL : Timeout in Sekunden, dreistellig, dezimal

3.4.7.2 Protokoll – Timeout

Es wird das Protokoll-Timeout definiert (001 bis 999 Millisekunden).

Wenn diese Information nicht zum Anzeiger gesendet wird, soll der Default-Timeout 45 ms betragen.

%	7	1	TH	TM	TL
---	---	---	----	----	----

TH TM TL : Timeout in Millisekunden, dreistellig, dezimal

3.4.7.3 Wechselzeit

Wechselzeit der Anzeigenseiten (000 bis 999 Sekunden).

%	7	2	TH	TM	TL
---	---	---	----	----	----

TH TM TL : Wechselzeit in Sekunden, dreistellig, dezimal

3.4.7.4 Diverse Parameter setzen

%	7	4	ANH	ANL	LSH	LSL	LFH	LFL	TSH	TSL	TFH	TFL
			P1H	P1L	P2H	P2L						

ANH ANL : Anzahl folgender Parameter (1..99) zweistellig, dezimal
 LSH LSL : Lauftext-Geschwindigkeit langsam zweistellig, hexadezimal
 LFH LFL : Lauftext-Geschwindigkeit schnell zweistellig, hexadezimal
 TSH TSL : Teiltext-Geschwindigkeit langsam zweistellig, hexadezimal
 TFH TFL : Teiltext-Geschwindigkeit schnell zweistellig, hexadezimal
 P1H P1L : weitere Parameter zweistellig, hexadezimal

Die einmal definierte Reihenfolge für weitere Parameter ist einzuhalten !

3.4.8 Datenversorgung laden

3.4.8.1 Datenversorgung löschen

Datenversorgung löschen (Texte, Felder, Zeichensätze)

%	8	0	N
---	---	---	---

N: 0 = gesamte Datenversorgung löschen (Texte, Felder, Zeichensätze)
 1 = Texte löschen
 2 = Felder löschen
 3 = Zeichensätze löschen

3.4.8.2 Abspeichern von Feldern

Felder werden vordefiniert.

%	8	2	F	X1H	X1M	X1L	X2H	X2M	X2L
---	---	---	---	-----	-----	-----	-----	-----	-----

F : Feldnummer (1..9), einstellig, dezimal
 X1H X1M X1L : X1, Startposition dreistellig, dezimal
 X2H X2M X2L : X2, Endposition dreistellig, dezimal

3.4.8.3 Abspeichern von Zeichensätzen

Abspeichern eines frei definierbaren Bitmusters eines Textzeichens bzw. Textgrafikzeichens unter dem Zugriffsschlüssel 'Schriftsatznummer, Schriftgröße, Zeichennummer'. Vor dem Abspeichern wird das unter diesem Zugriffsschlüssel eventuell früher einmal geladene Bitmuster gelöscht. Wenn unter diesem Schlüssel in der Anzeige bereits ein Bitmuster hinterlegt ist, wird dieses Bitmuster zusätzlich abgelegt. Beim Auslesen des Bitmusters wird immer das zusätzlich geladene Bitmuster, falls vorhanden, genommen. Wenn beim Auslesen unter dem übergebenen Schlüssel kein Bitmuster gefunden wird, werden im Schlüssel mit Ausnahme der Zeichennummer die Defaultwerte genommen, und somit das Bitmuster ausgelesen. Wenn auch unter diesem Schlüssel kein Eintrag gefunden wird, wird ein "Blank" als Bitmuster zurückgegeben.

%	8	3	SN	SG	TA	RA	ZNH	ZNL	ZNH	TSL	ABH	ABL
			B1H	B1L					BNH	BNL		

SN	: Schriftsatznummer (1..9),	einstellig,	dezimal
SG	: Schriftgröße (1..15),	einstellig,	hexadezimal
TA	: Textattribut,	einstellig,	hexadezimal, (Reserve, immer 0)
RA	: Reserveattribut,	einstellig,	hexadezimal, (Reserve, immer 0)
ZNH ZNM ZNL	: Zeichennummer (032..255),	dreistellig,	dezimal
ABH ABL	: Anzahl der Bytes des Bitmusters (01..98),	zweistellig,	dezimal
B1H B1L			
BNH BNL	: Es wird das Bitmuster definiert (0..255)	zweistellig,	hexadezimal

Das Bitmuster für ein Zeichen hat maximal 98 Bytes. Die Übertragung eines Bytes erfolgt in 2 Hexziffern. Das Bitmuster ist in der Regel bei jedem Anzeigentyp unterschiedlich. Deshalb wird es von jedem Anzeigenhersteller separat definiert.

Die so geladenen Bitmuster bleiben auch nach dem Abschalten der Anzeige gespeichert.

3.4.8.4 Grafik

Mit dem Befehl (1/2) können gespeicherte Grafiken auf der Anzeige dargestellt werden. Diese müssen mit den Befehlen (8/4) und (8/5) im Anzeigen-Controller hinterlegt werden.

Ein Grafikfeld hat eine festgelegte Größe von 32x16 Pixel. Somit ist die kleinste Größe einer Grafik auf dieses Maß festgelegt. Grafiken können allerdings mit mehreren Grafikfeldern aufgefüllt werden.

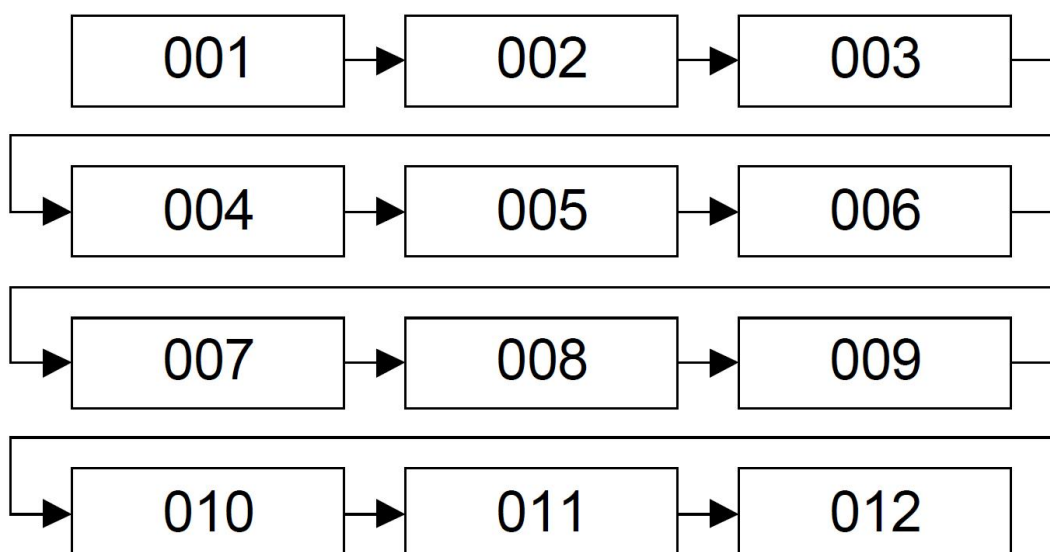
Die Größe einer Grafik kann nur ein Vielfaches der eines Grafikfeldes sein:

$$X = n * 32 \text{ Pixel}$$

$$Y = n * 16 \text{ Pixel}$$

Soll eine Grafik mit abweichenden Maßen dargestellt werden, so muss mit dem Befehl (0/4) ein Anzeigefeld definiert werden, dass kleiner als die betreffende Grafik ist. Das in dem Befehl befindliche Darstellungsattribut [DA] hat keinen Einfluss auf die Darstellung, muss aber einen gültigen Wert besitzen. Die Grafik wird dann mit dem Befehl (1/2) an der Koordinate X1,Y1 eingefügt.

Die folgende Abbildung zeigt die Position der Grafikfelder in einer Grafik.



Aus den 12 Grafikfeldern wird eine Grafik mit den Abmessungen 96 x 64 Pixel gebildet.

Beispiel: Grafik 01 mit den Abmessungen 32 x 16 Pixeln

aA1 11 %84 01 001 001 032 016 FP

in Grafikfeld 01 Grafik 001 abspeichern

aA1 88 %85 01 001 55 50 FF FF 55 50 C0 01 ... AA A0 E0 01 FP

	H1H				H1L				H2H				H2L				H3H				H3L				H4H				H4L			
	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
	5				5				5				0				F				F				F				F			
1	X		X		X		X		X		X		X		X		X		X		X		X		X		X		X		X	
2	X		X		X		X		X		X		X		X		X		X		X		X		X		X		X		X	
3	X		X		X		X		X		X		X		X		X		X		X		X		X		X		X		X	
4	X		X		X		X		X		X		X		X		X		X		X		X		X		X		X		X	
5	X		X		X		X		X		X		X		X		X		X		X		X		X		X		X		X	
6	X		X		X		X		X		X		X		X		X		X		X		X		X		X		X		X	
7	X		X		X		X		X		X		X		X		X		X		X		X		X		X		X		X	
8	X		X		X		X		X		X		X		X		X		X		X		X		X		X		X		X	
9	X		X		X		X		X		X		X		X		X		X		X		X		X		X		X		X	
10	X		X		X		X		X		X		X		X		X		X		X		X		X		X		X		X	
11	X		X		X		X		X		X		X		X		X		X		X		X		X		X		X		X	
12	X		X		X		X		X		X		X		X		X		X		X		X		X		X		X		X	
13	X		X		X		X		X		X		X		X		X		X		X		X		X		X		X		X	
14	X		X		X		X		X		X		X		X		X		X		X		X		X		X		X		X	
15	X		X		X		X		X		X		X		X		X		X		X		X		X		X		X		X	
16	X		X		X		X		X		X		X		X		X		X		X		X		X		X		X		X	
	A				A				A				0				E				0				0				1			
	H61H				H61L				H62H				H62L				H63H				H63L				H64H				H64L			
	X	Bit gesetzt																														
		Bit nicht gesetzt																														

3.4.8.5 Grafikfeld definieren

Es wird ein Grafikfeld definiert. (X1,Y1) / (X2,Y2) = Diagonal-Eckpunkte

%	8	4	GFH	GFL	X1T	X1H	X1Z	X1E	X2T	X2H	X2Z	X2E
					Y1T	Y1H	Y1Z	Y1E	Y2T	Y2H	Y2Z	Y2E

GFH GFL : Grafikfeldnummer(01..09), zweistellig, dezimal
 X1T ... X1E : X1 vierstellig, dezimal
 X2T ... X2E : X2 vierstellig, dezimal
 Y1T ... Y1E : Y1 vierstellig, dezimal

Y2T ... Y2E : Y2 vierstellig, dezimal

Die Feldgröße in X-Richtung muss ein Vielfaches von 32 sein.
Die Feldgröße in y -Richtung muss ein Vielfaches von 16 sein.
Koordinatenangaben in Pixel.

3.4.8.6 Grafik definieren

%	8	5	GFZ	GFE	BNT	BNH	BNZ	BNE	B1H	B1L			BNH	BNL
---	---	---	-----	-----	-----	-----	-----	-----	-----	-----	--	--	-----	-----

GFZ GFE : Grafikfeldnummer(01..09), zweistellig, dezimal

BNT ... BNE : Blocknummer (0001..9999), vierstellig, dezimal

B1L B1H

BNL BNH : Es wird das Bitmuster definiert (0..255) zweistellig, hexadezimal

Es wird das Bitmuster für die Grafik-Teil-Felder definiert. Das Bitmuster für ein Grafik-Teil-Feld hat 64 Bytes. Die Übertragung erfolgt in 2 Hexziffern.

3.4.8.7 Grafik als Bilddatei-Inhalt übertragen

%	8	5	GFZ	GFE	BNT	BNH	BNZ	BNE	mode	content
---	---	---	-----	-----	-----	-----	-----	-----	------	---------

GFZ GFE : Grafikfeldnummer(01..09), zweistellig, dezimal

BNT ... BNE : Blocknummer (0001..9999), vierstellig, dezimal

Mode : legt fest, welcher Art das Telegramm ist

Content : der Telegramminhalt, abhängig von mode (immer 128 byte)

Wenn mode ein Zeichen der Menge Hex = { [0-9], [A-F] } ist, dann handelt es sich um ein normales PACOS-Telegramm, mit dem Pixel-Daten übertragen werden (siehe 3.4.8.6), anderenfalls werden Bilddaten in Dateiform übertragen. Mode und content bilden zusammen immer 128 Byte. Wenn weniger als 128 Byte zusammen kommen, wird bis 128 Byte mit ASCII 0 aufgefüllt.

Mode (ASCII)	Bedeutung	content
S	Startblock	Dateiname ohne Pfad (ASCII, mit Punkt) + „ “ (wenn Länge des Dateinames ungerade) oder + „ “ (wenn Länge der Dateinames gerade) + Länge der folgenden Datenbytes (1 Byte in Hex-Darstellung) + + Teilinhalt der Datei. Ein Byte wird durch 2 Byte ASCII-Hex übermittelt.
P	Teilblock	Länge der folgenden Datenbytes (1 Byte in Hex-Darstellung) + „ “ + Teilinhalt der Datei. Ein Byte wird durch 2 Byte ASCII-Hex übermittelt.
L	Schlussblock	Länge der folgenden Datenbytes (1 Byte in Hex-Darstellung) + „ “ + Teilinhalt der Datei. Ein Byte wird durch 2 Byte ASCII-Hex übermittelt.

Beispiel für „txtT-Datei mit Inhalt „hello world“

„Stxt|0B68656C6C6F20776F726C64000000000000...00“

„L00|00000000..00“

3.4.9 Abfragen

3.4.9.1 Abfrage der Versionsnummer

Es wird die Softwareversionsnummer der Haltestellenanzeige abgefragt. Die Anzeige fügt die Versionsnummer in der Datensatzantwort unmittelbar hinter dem Status ein. Die Datensatzantwort darf maximal um 80 Zeichen erweitert werden.

%	9	0
---	---	---

3.4.9.2 Abfrage des Grafikstatus

Es wird die SHA1¹-Prüfsumme einer Grafik abgefragt. Die Antwort auf diese Anfrage beginnt wie eine Status-Antwort. Bei einer positiven Rückmeldung wird nach dem Statussymbol mit zwei Zeichen die Grafikfeldnummer übertragen. An diese schließt sich die lesbare SHA1-Prüfsumme an. Abschließend folgen noch ein CR und eine Prüfsumme der Nachricht. Eine negative Rückmeldung bringt auch eine positive Statusmeldung hervor, jedoch ohne sich anschließende Prüfsumme.

GFH GFL : Grafiknummer(01..99), zweistellig, dezimal

%	9	9	GFH	GFL
---	---	---	-----	-----

Positive Meldung: Übermittlung der Prüfsumme hinter Status 0.

a	0	GFH	GFL	S1H	S1L			SnH	SnL	CRC
---	---	-----	-----	-----	-----	--	--	-----	-----	-----

SnH SnL : Prüfsumme ($1 < n < 20$), zweistellig, hexadezimal

Negative Meldung (keine Grafik vorhanden): nur Status 0.

a	0	CRC
---	---	-----

Der SHA1-Algorithmus wird auf den gesamten Bildinhalt, beginnend beim Pixel links oben, zeilenweise, angewandt. 8 Pixel bilden ein Byte für den Eingangsdatenstrom. Das linke von acht Pixeln entspricht dabei dem höchstwertigsten Bit und das rechte Pixel dem niederwertigsten Bit eines Bytes. Ein gesetztes Pixel entspricht einem gesetztem Bit. Ist die Anzahl der Pixel in einer Zeile kein Vielfaches von acht, so wird das letzte Byte einer Zeile von rechts mit Null-Bits aufgefüllt. Die Anzahl der Bytes einer Zeile ist ein Vielfaches von 4.

3.4.10 Servicefunktionen

3.4.10.1 Selbsttest

Die Haltestellenanzeige wird aufgefordert, einen Selbsttest durchzuführen. (HM = 10, 0x3a, „.")

%	10	0
---	----	---

¹ SHA1: beschrieben in RFC 3174

4 Erläuterungen

4.1 Textaufbau

Siehe Kommando HM/SM : 1/0 (siehe Kapitel 3.4.3.1 Text darstellen, Seite 9)

Darstellung:

Wie in der Telegrammdefinition bereits beschrieben wurde, kann bei der Darstellung von Texten ein Text-Zielfeld (Blockmode: 0/0) mit zugehörigem Darstellungsattribut definiert werden.

Außerdem kann ein Text Steuerzeichen für den Zeilenvorschub enthalten. Das Steuerzeichen (Zeilenvorschub) hat zwei mögliche Wirkungen:

1.Wirkung

Sofern das Darstellungsattribut des Feldes "Teiltextsequenz" lautet, muss der Anzeigen-Controller die Teiltexte zeitsequenziell auf derselben Zeile darstellen.

Beim Darstellungsattribut "Rollen" verhält es sich ähnlich wie bei der "Teiltextsequenz". Der Teiltext wird jedoch Buchstabenweise dargestellt ("abgerollt") und auch wieder so weggenommen ("aufgerollt").

Beim Darstellungsattribut "Abspanntext" werden die Teiltexte als "vertikaler Lauftext" dargestellt.

2.Wirkung

Bei allen anderen Attributen bewirkt LF einen Zeilenvorschub auf die nächste Feld-Zeile (die Feldausrichtung der Teiltexte ist ebenfalls über das Feldattribut definiert). Ein Text, der aufgrund seiner Länge nicht im Textfeld dargestellt werden kann und keine LF enthält, wird abgeschnitten - sofern das Darstellungsattribut nicht "Lauftext" lautet. Umgekehrt kann ein Text, der kürzer als das Textfeld ist, nicht als "Lauftext" dargestellt werden. Beim Lauftext durchläuft der ganze Text das Textfeld iterativ von rechts nach links.

4.2 Textattribute

Innerhalb von Texten können Attribute der Form:

'\$'HH (\$ = Attributstart-Steuerzeichen / HH = Attribut-Kennung)

vorkommen.

H	H	Attribut
0		Schriftsatz ändern => z.B. für länderspezifische ASCII-Tabellen
	0	Schriftsatz 0 (Default) (fix in der Anzeigen-Software hinterlegt)
	1	Schriftsatz 1 (deutsch) (fix in der Anzeigen-Software hinterlegt)
	2	Schriftsatz 2 (englisch) (fix in der Anzeigen-Software hinterlegt)
	3	Schriftsatz 3 (französisch) (fix in der Anzeigen-Software hinterlegt)
	n	Schriftsatz n (muss in der Regel zuerst einmal in die Anzeige geladen werden)
1		Schriftgröße ändern (Schriftgrößen müssen in der Anzeige hinterlegt sein)
	0	Schriftgröße 0 (Default)
	n	Schriftgröße n
2		Einmaliger Zeichenabstand an der aktuellen Position (nur bei Matrix-Anzeigen)
	1	ein Bildpunkt

	n	Vorschub um n-Bildpunkte
3		Zeichenabstand in einer Zeichenkette (nur bei Matrix-Anzeigen)
	1	ein Bildpunkt (Default)
	n	n-Bildpunkte Abstand
4		Umschaltung auf invers
	0	normal (Default)
	1	Invers
5		Umschaltung auf fett
	0	normal (Default)
	1	Fett
6		Umschaltung auf blinken
	0	normal (Default)
	1	blinken schnell
	2	blinken langsam
7		Farbumschaltung (nur bei mehrfarbiger LED-Matrix)
	0	Farbe 1 (Default)
	n	Farbe n
8		Sonderzeichen-Nummer
	mn	Sonderzeichen mn Sonderzeichen sind die Textgrafikzeichen mit den Zeichennummern 80h. .. FFh. (128 .. 255)

Bemerkung

Die Default-Attribute werden beim Rücksetzen der Anzeige aktiviert.
Diese Textattribute bleiben nicht nur für den nachfolgenden Text,
sondern bis zu ihrem Widerruf (bzw. Korrektur) aktiv !

4.3 Darstellen der Steuerzeichen

Da "%" und "\$" bei dieser Telegrammdefinition Steuerzeichenfunktionen für die Block bzw. Attributstart-Kennung übernehmen, können diese nicht als Textbestandteil vorkommen.
Will man die beiden Zeichen trotzdem als Text darstellen, müssen sie verdoppelt werden.

Beispiel: "Luftfeuchtigkeit: 80%%"

5 Statusabfrage**5.1 Statusanfrage**

Der Anzeigen-Status wird durch den Knotenrechner (Master) zyklisch abgefragt. Auf diese Weise kann ein schnelles Reagieren auf Anzeigerdefekte gewährleistet werden.

aA	ADR	CRC
----	-----	-----

Kürzel	Bezeichnung	Länge	Beschreibung
aA	Startkennung	2 Byte	Kennzeichnung für Telegrammanfang ,aA'
ADR	Adresse	1 Byte	Adresse des Anzeigers
CRC	CR + Längsparität	2 Byte	carriage return und CRC (alle Byte des Telegramm ohne CRC)

Als Reaktion auf die Statusabfrage schickt die Anzeige ein Datensatz-Antworttelegramm.
Das Ausbleiben einer Statusabfrage ist für die Anzeige ein Hinweis auf ein Fehlverhalten

des Masters. Der Anzeiger muss mit löschen der Anzeige reagieren.

5.2 Status- und Datensatzantwort

a	H	CRC
---	---	-----

Kürzel	Bezeichnung	Länge	Beschreibung
a	Startkennung	1 Byte	Kennzeichnung für Telegrammanfang ,a'
H	Status	1 Byte	Siehe Tabelle (0..15), hexadezimal
CRC	CR + Längsparität	2 Byte	carriage return und CRC (alle Byte des Telegramm ohne CRC)

Status	Beschreibung	Hinweise
0	Übertragung i.o	1
1	Unbekannte Text-/Grafiknummer	
2	Unbekanntes Attribut (bzw. Mode/Submode)	
3	Ungültige Positionsangabe => Fehlerhafte x/y-Position oder unbekannte Feld-/Zeilennummer	
4	Text kann nicht vollständig dargestellt werden	
5	Allg. Datensatz-Formatfehler	
6	Empfangspuffer voll	
7	Anzeige defekt (Allg. Störung)	2
8	gestört	
9		
10 (:)	Paritätsfehler	
11 (;)	falsche Konfiguration	
12 (<)	überlastet	
13 (=)	Datenbuffer der Textgrafikzeichen voll	
14 (>)		
15 (?)	Erste Statusantwort nach Reset der Anzeige (nicht nach Anzeigen-Initialisierungs-Befehl)	3

Hinweise

1. Sobald der Anzeiger nicht mehr antwortet, sendet der Knotenrechner diesem Anzeiger einen Löschbefehl und anschließend nur noch Statusabfragen. Wenn der Knotenrechner wieder eine Statusantwort empfängt, wird der Anzeiger neu initialisiert und alle Informationen für die Darstellung werden zu diesem übertragen.
 2. Sobald der Status 7 (=Anzeige defekt) ansteht, sendet der Knotenrechner diesem Anzeiger einen Löschbefehl und anschließend nur noch Statusabfragen. Wenn der Knotenrechner wieder die Statusantwort mit dem Statuswert 0 empfängt, wird der Anzeiger neu initialisiert, und alle Informationen für die Darstellung werden zu diesem übertragen.
 3. Der Anzeiger wird vom Knotenrechner neu initialisiert, und alle Informationen für die Darstellung werden zu diesem neu übertragen.
- 1+2 Der Knotenrechner kann nur durch die Statusantwort mit dem Statuswert 0 erkennen, dass der Fehler „Anzeige defekt (Allg. Störung)“ nicht mehr anstehen.
- 2+3 Wenn keine dieser beiden Fehlermeldungen ansteht, wird der Anzeiger vom Knotenrechner ganz normal beschrieben.

Die Datensatz-Antwort erfüllt zwei unterschiedliche Aufgaben:

1. Die Datensatzantwort bestätigt dem Wagenbus-Master die formale Korrektheit des letzten Datensatzes. Ein Master-Datensatz ist formal korrekt, wenn er über eine gültige Startkennung, eine korrekte Datensatz-Längenangabe, ein CR und eine korrekte Zeichen- und Längsparität verfügt.
2. Die Datensatzantwort informiert den Wagenbus-Master über die inhaltliche Korrektheit eines vergangenen Datensatzes. Ein Master-Datensatz ist inhaltlich korrekt, wenn alle Datensatz-Bestandteile (wie Haupt-/Submode, Text-/Grafiknummern, Feld-/Positionsangaben, Textattribute etc.) erkannt und korrekt auf der Anzeige abgebildet werden können.

Reaktion der Anzeige:

1. Ein formal unkorrekter Datensatz wird von der Anzeige verworfen. Es wird keine Datensatz-Antwort an den Master geschickt. Ein formal korrekter Datensatz führt unverzüglich zu einer Datensatz-Antwort - die Antwort enthält den aktuellsten Eintrag aus dem Fehlerbuffer (der entsprechende Eintrag im Fehlerbuffer wird durch diese Handlung konsumiert).
2. Ein formal korrekter Datensatz wird dem Datensatz-Parser übergeben, dieser ermittelt dessen inhaltliche Korrektheit. Ein inhaltlich unkorrekter Datensatz wird von der Anzeige verworfen - die entsprechende Fehlernummer wird in einen Fehlerbuffer eingetragen. Ein inhaltlich korrekter Datensatz wird auf der Anzeige dargestellt.

6 Statusinformationen

6.1 Abfrage Umgebungslicht

Das Umgebungslicht des Anzeigers kann mit folgendem Kommando abgefragt werden:

Kommando

%	9	8	0	1
---	---	---	---	---

Antwort

a	0	^	01	FV	FH	00	00	00	00	00
---	---	---	----	----	----	----	----	----	----	----

31 : 1 - Kennung Umgebungslicht, zweistellig
 FV : Umgebungslicht vorn (0..255) zweistellig, hexadezimal
 FH : Umgebungslicht hinten (0..255)

6.2 Abfrage Feuchte

Die Feuchte im Anzeiger kann mit folgendem Kommando abgefragt werden:

Kommando

%	9	8	0	2
---	---	---	---	---

Antwort

a	0	^	02	FF	00	00	00	00	00	00
---	---	---	----	----	----	----	----	----	----	----

32 : 2 - Kennung Feuchte zweistellig
 FF : Feuchte in % (z.B. 3; = 59%) zweistellig, hexadezimal
 00 : frei (für Erweiterungen)

6.3 Abfrage Betriebsstunden

Die Betriebsstunden des Anzeigers können mit folgendem Kommando abgefragt werden:

Kommando

%	9	8	0	3
---	---	---	---	---

Antwort

a	0	^	03	F1	F2	F3	F4	00	00	00
---	---	---	----	----	----	----	----	----	----	----

33 : 3 - Kennung Betriebsstunden
 F1 – F4 : Betriebsstunden als uint32 (F1 = LSB), 8-stellig, hexadezimal
 00 : frei (für Erweiterungen)

7 Broadcast Uhrzeit / Datum

Aufbau des Telegramms

dU	tmmjjjj	hhmmss	CRC
----	---------	--------	-----

Kürzel	Bezeichnung	Länge	Beschreibung
dU	Startkennung	2 Byte	Kennzeichnung für Telegrammanfang ‚dU‘
tmmjjjj	Datum	8 Byte	Tag, Monat, Jahr
hhmmss	Zeit	6 Byte	Stunden, Minuten, Sekunden
CRC	CR + Längsparität	2 Byte	carriage return und Alle Byte des Telegramm ohne CRC

Dieses Telegramm kann als Broadcast ca. alle 5 Min. gesendet werden, damit die Uhren aller Anzeigen gleichzeitig synchronisiert werden. Dieses Telegramm darf nicht quittiert werden.

8 Protokoll

Kommunikation

Der Knotenrechner kommuniziert als Master über ein Bussystem mit den Anzeigen. Der Master sendet ein Telegramm an den Anzeige-Slave und wartet auf dessen Datensatz-Antwort. Die Antwort muss innerhalb einer Frist erfolgen (default 40ms, siehe auch 3.4.7.2 Protokoll – Timeout, Seite 12) erfolgen, sonst bricht der Master den Empfang ab und wiederholt das Telegramm.

Müssen während einem bestimmten Zeitraum (150 Sekunden) keine Telegramme zum Anzeiger gesendet werden, vergewissert sich der Master mittels einem Statusabfrage-Telegramm über die

Funktionstüchtigkeit der Anzeige. Bleibt diese zyklische Statusabfrage durch den Master aus, ist entweder der Master nicht mehr funktionstüchtig, oder das Bussystem ist unterbrochen. Der Anzeiger muss auf diesen Betriebszustand reagieren und die Anzeige löschen.

9 Prüfsumme (CRC)

Die Prüfsumme wird über alle Byte des Telegramms gebildet, mit Ausnahme der Prüfsumme selbst. Die Berechnung erfolgt durch Verknüpfung der einzelnen Byte über XOR mit definierter Anfangsbelegung.

Zur Prüfung empfangener Telegramme kann über alle Byte einschließlich der Prüfsumme die XOR- Verknüpfung berechnet werden. Das Ergebnis muss immer 0x7f (127) sein.

CRC-Berechnung

```
bCrc = 0x7F; // Anfangsbelegung
```

```
iLen = Telegrammlänge (ohne CRC);
```

```
for (iCnt = 0; iCnt < iLen; iCnt++)  
{  
    bCrc = bCrc ^ bData[iCnt]; // ^ = XOR  
}
```

CRC-Prüfung

```
iLen = Telegrammlänge (mit CRC)
```

```
bCrc = bData[0];  
for (iCnt = 1; iCnt < iLen; iCnt++)  
{  
    bCrc = bCrc ^ bData[iCnt]; // ^ = XOR  
}  
Ergebnis: bCrc = 0x7F;
```

Beispiele

Telegramm (Hex.): 61 41 31 0d 63 mit 63 als CRC (Prüfsumme)

61 30 0D 23 mit 23 als CRC

61 41 32 0D 60 mit 60 als CRC

10 Datenversorgung Aushangfahrplan

(1) die Datenversorgung des Anzeigers wird über den angeschlossenen Knotenrechner gewährleistet. Der Knotenrechner lädt die Datei(n) auf den Anzeiger und gibt diesem vor, welche aktuell anzuzeigen ist.

(2) nach einem Stromausfall und der Spannungswiderkehr am Anzeiger verändert der Anzeiger seine Anzeige nicht (Anzeige wird nicht gelöscht!). Der Anzeiger sendet an den Knoten einmalig bei der ersten Statusabfrage die Statusinformation "Spannungswiderkehr" (Nr. 15, ASCII '?')

(siehe "erste Statusantwort nach Reset der Anzeige", Spezifikation 2

V01.05, Abschnitt 5.2, S. 19)

(3) Der Knotenrechner prüft nach einem Neustart (z.B. nach Spannungsausfall) und nach der Statusinformation "Spannungswiderkehr" vom Anzeiger die aktuellen Informationen auf dem Anzeiger und korrigiert diese bei Bedarf.

Die Anzeige kann nur noch durch einen aktiven Knotenrechner verändert werden.

11 Codierung

Hexadezimal

0123456789ABCDEF	hexadezimal
0123456789:;<=>?	Im Protokoll

z.B. hexadezimal zweistellig: 2; (Protokoll) = 2B (hex.) = 43 (dezimal)

12 Schnittstellendefinition

Serielle asynchrone RS-422 Schnittstelle (4-Draht).

Baudrate:	9600
Bit pro Character:	7Bit
Stop Bits:	2
Parity:	Even

Zeichensatz mit 256 Zeichen. Sonderzeichensatz Dresden (siehe Anlage).