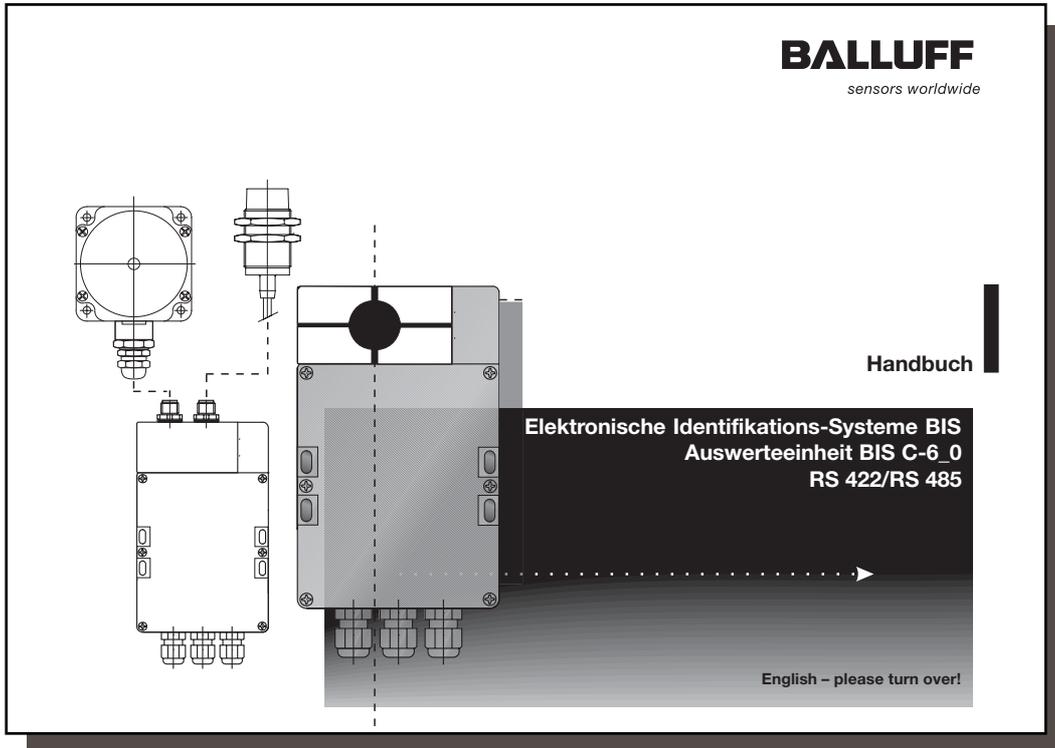


1



2

Nr. 854 487 D/E • Ausgabe 1012
Änderungen vorbehalten.
Ersetzt Ausgabe 0704.

Schreibweise:

Zu sendende Steuerzeichen sind in spitze Klammern gesetzt.

Im ASCII-Code zu übertragende Zeichen sind in Hochkomma gesetzt.

Beispiel: <STX> '1 2 3 4 5 6' BCC

Handbuch gültig ab Hardwareversion HV 2.0

Balluff GmbH
Schurwaldstraße 9
73765 Neuhausen a.d.F.
Deutschland
Tel. +49 7158 173-0
Fax +49 7158 5010
balluff@balluff.de

■ www.balluff.com

Inhaltsverzeichnis

Sicherheitshinweise	4
Einführung BIS C	5-7
Anwendung BIS C	8
Protokollablauf	9-11
Konfiguration	12-27
Kundenkonfiguration	28-31
Schreib-/Lesezeiten	32/33
Programmierinformationen	34-52
Fehlernummern	53/54
BIS C-600: Schnittstelleninformationen	55-58
Montage der Auswerteeinheit / Kopf	59-60
Technische Daten	61/62
Bestellinformationen	63
BIS C-620: Montage der Auswerteeinheit	64
Schnittstelleninformationen	65-67
Technische Daten	68/69
Bestellinformationen	70
Anhang: ASCII-Tabelle	71

Sicherheitshinweise

Auswerteeinheiten BIS C-6_0 bilden zusammen mit den anderen Bausteinen des Systems BIS C das Identifikations-System und dürfen nur für diese Aufgabe eingesetzt werden.

Installation und Betrieb

Installation und Betrieb sind nur durch geschultes Fachpersonal zulässig. Unbefugte Eingriffe und unsachgemäße Verwendung führen zum Verlust von Garantie- und Haftungsansprüchen.

Bei der Installation der Auswerteeinheit sind die Kapitel mit den Anschlussplänen genau zu beachten. Besondere Sorgfalt erfordert der Anschluss der Auswerteeinheit an externe Steuerungen, speziell bezüglich Auswahl und Polung der Verbindungen und der Stromversorgung.

Für die Stromversorgung der Auswerteeinheit dürfen nur zugelassene Stromversorgungen benutzt werden. Einzelheiten enthält das Kapitel Technische Daten.

Einsatz und Prüfung

Für den Einsatz des Identifikations-Systems sind die einschlägigen Sicherheitsvorschriften zu beachten. Insbesondere müssen Maßnahmen getroffen werden, dass bei einem Defekt des Identifikations-Systems keine Gefahren für Personen und Sachen entstehen können.

Hierzu gehören die Einhaltung der zulässigen Umgebungsbedingungen und die regelmäßige Überprüfung der Funktionsfähigkeit des Identifikations-Systems mit allen damit verbundenen Komponenten.

Funktionsstörungen

Wenn Anzeichen erkennbar sind, dass das Identifikations-System nicht ordnungsgemäß arbeitet, ist es außer Betrieb zu nehmen und gegen unbefugte Benutzung zu sichern.

Gültigkeit

Diese Beschreibung gilt für Auswerteeinheiten der Baureihe BIS C-600-007-...-02-KL1 und BIS C-620-007-050-02-ST2.

Einführung Identifikations-System BIS C

Dieses Handbuch soll den Anwender beim Erstellen des Steuerprogramms, der Installation und der Inbetriebnahme der Komponenten des Identifikations-Systems BIS C-6_0 anleiten, so dass sich ein sofortiger, reibungsloser Betrieb anschließt.

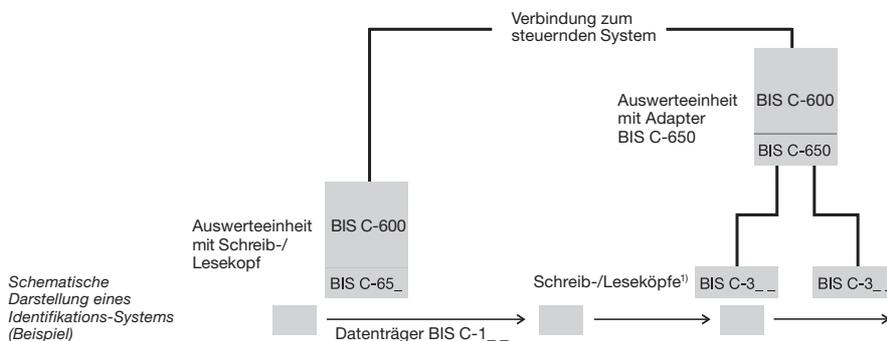
Prinzip	Das Identifikations-System BIS C-6_0 gehört zur Kategorie der berührungslos arbeitenden Systeme, die sowohl lesen als auch schreiben können. Diese Doppelfunktion ermöglicht Einsätze, bei denen nicht nur fest in den Datenträger programmierte Informationen transportiert, sondern auch aktuelle Informationen gesammelt und weitergegeben werden.
Einsatzgebiete	Einige der wesentlichen Einsatzgebiete finden sich <ul style="list-style-type: none"> - in der Produktion zur Steuerung des Materialflusses (z. B. bei variantenspezifischen Prozessen), beim Werkstücktransport mit Förderanlagen, zur Datengewinnung für die Qualitätssicherung, zur Erfassung sicherheitsrelevanter Daten, - in der Werkzeugcodierung und -überwachung; - in der Betriebsmittelorganisation; - im Lagerbereich zur Kontrolle der Lagerbewegungen und -bestände; - im Transportwesen und in der Fördertechnik; - in der Entsorgung zur mengenabhängigen Erfassung.

Einführung Identifikations-System BIS C-600

System-komponenten

Die Hauptbestandteile des Identifikationssystems BIS C-600 sind

- **Auswerteeinheit,**
- **Schreib-/Leseköpfe und**
- **Datenträger.**



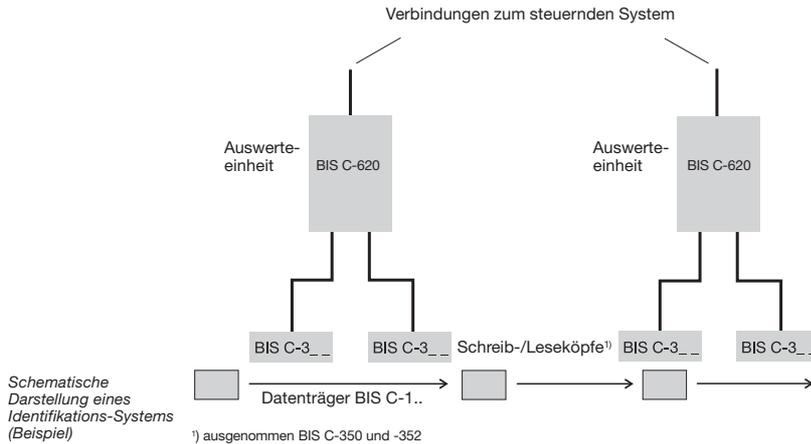
¹⁾ ausgenommen BIS C-350 und -352

Einführung Identifikations-System BIS C-620

System- komponenten

Die Hauptbestandteile des Identifikationssystems BIS C-620 sind

- Auswerteeinheit,
- Schreib-/Leseköpfe und
- Datenträger.



Anwendung Auswerteeinheit BIS C-6_0

Auswahl der Systemkomponenten

Die Auswerteeinheit **BIS C-600** besitzt ein Kunststoffgehäuse. Der Anschluss erfolgt über eine Klemmleiste, wobei die Kabel mittels PG-Verschraubung gesichert werden. An die Auswerteeinheit kann ein einzelner Schreib-/Lesekopf der Baureihe BIS C-65_ direkt montiert werden, wodurch eine kompakte Einheit entsteht. Ist der Adapter BIS C-650 anstatt des Schreib-/Lesekopfes BIS C-65_ montiert, können zwei Schreib-/Leseköpfe abgesetzt über Kabel angeschlossen werden. Ist der Adapter BIS C-670 montiert, kann ein Schreib-/Lesekopf abgesetzt über Kabel angeschlossen werden.

Die Auswerteeinheit **BIS C-620** besitzt ein Metallgehäuse. Der Anschluss erfolgt über Rundsteckverbinder. An die Auswerteeinheit BIS C-620 können zwei Schreib-/Leseköpfe abgesetzt über Kabel angeschlossen werden.

Ob die Kompaktlösung Auswerteeinheit mit integriertem Schreib-/Lesekopf oder die abgesetzte Lösung sinnvoll ist, richtet sich im wesentlichen nach der räumlichen Anordnung der Bausteine. Funktionale Einschränkungen sind nicht gegeben. Alle Schreib-/Leseköpfe sind für statisches und dynamisches Lesen geeignet. Abstand und Relativgeschwindigkeit richten sich nach der Wahl des Datenträgers. Weitere Informationen zu den Schreib-/Leseköpfen der Baureihe BIS C-65_ bzw. der Baureihe BIS C-3_ mit sämtlichen Kombinationen der passenden Datenträger finden Sie in den zugehörigen Handbüchern der Schreib-/Leseköpfe.

Die Systemkomponenten werden von der Auswerteeinheit elektrisch versorgt. Der Datenträger stellt eine eigenständige Einheit dar, benötigt also keine leitungsgebundene Stromzuführung. Er bekommt seine Energie vom Schreib-/Lesekopf. Dieser sendet ständig ein Trägersignal aus, das den Datenträger versorgt, sobald der notwendige Abstand erreicht ist. In dieser Phase findet der Schreib-/Lesevorgang statt. Dieser kann statisch oder dynamisch erfolgen.

Protokollablauf

Steuerfunktion

Die Auswerteeinheit BIS C-600 wird von einer übergeordneten Einheit gesteuert:

- von einem Steuerrechner (z.B. Industrie-PC) oder
- von einer externen speicherprogrammierbaren Steuerung (SPS).

Prinzip des Protokollablaufs mit schematischen Darstellungen

Die Auswerteeinheit steuert und verwaltet die Datenkommunikation zwischen Datenträgern und Schreib-/Leseköpfen. Über die serielle 4-Draht-Schnittstelle RS 422 verbindet sie das Identifikations-System BIS C-600 mit einer externen Steuereinrichtung. Bei Bedarf kann die Auswerteeinheit auf die 2-Draht-Schnittstelle RS 485 umgestellt werden.

Der Datenverkehr zwischen der Auswerteeinheit und dem steuernden System geschieht über festgelegte Telegramme. Der Protokollablauf wird in diesem Kapitel schematisch in Form von Funktionsblöcken dargestellt. Im Kapitel Programmierinformationen wird der Telegramminhalt präzisiert.

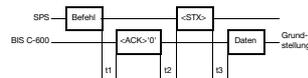
Voraussetzung für die Gültigkeit der folgenden Darstellungen:

- Die Auswerteeinheit muss sich in Grundstellung befinden.
- Vor dem Schreib-/Lesekopf befindet sich ein Datenträger.

Protokollablauf

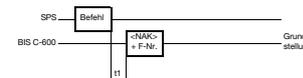
Ohne Kopfschaltung

Lesen: a) Es tritt kein Fehler auf:



t1 je nach Anzahl zu lesender Bytes (siehe ↗ 32/33)
t2 ≥ 0 (wird von der Auswerteeinheit nicht überwacht)
t3 = max. 50 ms

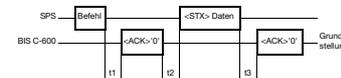
b) Es tritt ein Fehler auf:



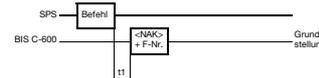
t1 je nach Anzahl zu lesender Bytes (siehe ↗ 32/33) und Fehlerart (empfohlene Überwachungszeit: 15 s)

Schreiben: a) Es tritt kein Fehler auf:

t1 = max. 50 ms
t2 ≥ 0 (wird von der Auswerteeinheit nicht überwacht)
t3 je nach Anzahl zu schreibender Bytes (siehe ↗ 32/33)

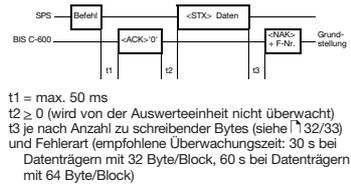


b) Es tritt ein Fehler im Befehl auf:



t1 = max. 50 ms

c) Es tritt ein Fehler beim Schreiben auf:



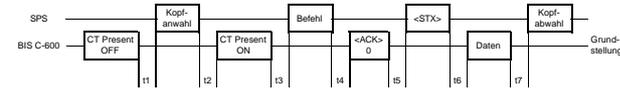
t1 = max. 50 ms
t2 ≥ 0 (wird von der Auswerteeinheit nicht überwacht)
t3 je nach Anzahl zu schreibender Bytes (siehe ↗ 32/33) und Fehlerart (empfohlene Überwachungszeit: 30 s bei Datenträgern mit 32 Byte/Block, 60 s bei Datenträgern mit 64 Byte/Block)

Protokollablauf

Mit Kopfschaltung

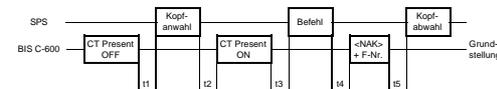
Lesen:

a) Es tritt kein Fehler auf:



t1, t3, t7 ≥ 0
 t2 = max. 500 ms
 t4 je nach Anzahl zu lesender Bytes (siehe ¶ 32/33)
 t5 ≥ 0 (wird von der Auswerteeinheit nicht überwacht)
 t6 = max. 50 ms

b) Es tritt ein Fehler auf:



t1, t3, t5 ≥ 0
 t2 = max. 500 ms
 t4 je nach Anzahl zu lesender Bytes (siehe ¶ 32/33) und Fehlerart (empfohlene Überwachungszeit: 15 s)

Konfiguration

Vor Beginn der Programmierung ist die Konfiguration der Auswerteeinheit durchzuführen, falls nicht mit der Werkseinstellung gearbeitet werden soll.

Die Konfiguration wird mittels Computer und der Balluff-Software BISC600C.EXE vorgenommen und in der Auswerteeinheit gespeichert. Sie kann jederzeit überschrieben werden. Die Konfiguration kann in einer Datei gespeichert werden und ist so jederzeit wieder verfügbar.

Dateien Online Konfiguration Hilfe

BALLUFF

Identifikationssysteme

Inbetriebnahmehilfe für BIS C-600

RS422 oder RS485 Schnittstelle
 als Punkt zu Punkt Verbindung (nicht adressierbar)

Copyright (c) 1997, Balluff GmbH

<F1=Hilfe> <F10=Menü> <Alt+=Befehl>

Wichtig. Bitte dokumentieren Sie die gewählten Einstellungen auf den mitgelieferten Aufklebern (auf die Innenseite des Gerätedeckels kleben) sowie auf den ¶ 28 und 30 in der Kundenkonfiguration. **Bitte beachten.** Damit kann einerseits bei einer Reparatur der Auswerteeinheit Ihre Konfiguration gesichert werden und andererseits für weitere Einheiten benutzt werden.

Konfiguration

Schnittstelle

In der ersten Maske werden die Parameter Übertragungsrate, Anzahl der Daten- und Stopbits sowie die Parity-Art für die serielle Schnittstelle eingestellt. Die Abbildung zeigt die Werks-einstellungen. Die weiteren Einstellungen werden in den Masken vorgenommen, die auf den folgenden Seiten abgebildet sind.

SCHNITTSTELLE BIS C-6_0

<p style="text-align: center; margin: 0;">baudrate</p> <input type="checkbox"/> 600 baud <input type="checkbox"/> 1200 baud <input type="checkbox"/> 2400 baud <input type="checkbox"/> 4800 baud <input checked="" type="checkbox"/> 9600 baud <input type="checkbox"/> 19200 baud	<p style="text-align: center; margin: 0;">databit</p> <input type="checkbox"/> 7 <input checked="" type="checkbox"/> 8	<p style="text-align: center; margin: 0;">parity</p> <input type="checkbox"/> odd <input checked="" type="checkbox"/> even <input type="checkbox"/> none
<p style="text-align: center; margin: 0;">stopbit</p> <input checked="" type="checkbox"/> 1 <input type="checkbox"/> 2		

Kurzform der Einstellungen

Wenn die Initialisierungsdaten in Kurzform vorliegen (z.B. auf dem Gehäusedeckel nach einem Gerätetausch), kann die Eingabe direkt in die Maske "Kurzform der Einstellungen BIS C-6_0" eingetragen werden (siehe auch Kundenkonfiguration auf 128).

KURZFORM DER EINSTELLUNGEN BIS C-6_0

```
[ 09600 ] [ 8 ] [ 1 ] [ E ]
[ 1 ] [ X ] [ X ] [ 1 ] [ 1 ]
[ 0000000 ]
```

Ändern Sie keinesfalls die hier angekreuzten Felder.

Konfiguration

Einstellungen

EINSTELLUNGEN BIS C-6_0

<p style="text-align: center; margin: 0;">Protokolltyp</p> <input checked="" type="checkbox"/> BCC <input type="checkbox"/> CR als Endekennung <input type="checkbox"/> CR <input type="checkbox"/> LFCR als Endekennung	<p style="text-align: center; margin: 0;">Seitengröße</p> <input checked="" type="checkbox"/> 32 Byte <input type="checkbox"/> 64 Byte
<p style="text-align: center; margin: 0;">Parameter</p> <input type="checkbox"/> RS 422 [] / RS 485 [X] <input type="checkbox"/> CT-Present Meldung mit '1'* <input type="checkbox"/> Dynamik-Betrieb <input type="checkbox"/> CT-Present auf Ausgang 1 <input type="checkbox"/> Ausgänge bei CT-Present bearb. <input type="checkbox"/> Schnelle Datenträgererkennung <input type="checkbox"/> CRC_16-Datenprüfung	<p style="text-align: center; margin: 0;">Eingang</p> <input checked="" type="checkbox"/> Reset <input type="checkbox"/> Kopfanwahl <input type="checkbox"/> Datenbit auf Datenträger <input type="checkbox"/> Keine Funktion

Protokolltyp

Werkseitig ist auf Betrieb mit Blockcheck BCC eingestellt. Für Steuergeräte, die ein Endekennungszeichen benötigen, kann die zusätzliche Verwendung von Carriage Return 'CR' oder Linefeed mit Carriage Return 'LF CR' eingerichtet werden. Auf der folgenden Seite finden Sie Beispiele für die verschiedenen Möglichkeiten.

Konfiguration

Protokolltyp (Fortsetzung)

Protokollvarianten	Telegramm mit Befehl, Adresse und Anzahl Bytes	Abschluss	Quittung	Ende- kennung
mit Blockcheck BCC	'R 0000 0001'	BCC	<ACK> '0'	
mit Carriage return	'R 0000 0001'	'CR'	<ACK> '0'	
mit Endeckennung Carriage return	'R 0000 0001'	'CR'	<ACK> '0'	'CR'
mit Endeckennung Carriage return und Line feed	'R 0000 0001'	'LF CR'	<ACK> '0'	'LF CR'

Beispiele für den
Abschluss der
Telegramme:

Parameter

- RS 485, 2-Draht

Werkseitig ist das Gerät auf die 4-Draht-Schnittstelle RS 422 konfiguriert. Mit diesem Parameter wird auf die 2-Draht-Schnittstelle RS 485 umgeschaltet.

Bitte beachten Sie bei Betrieb mit RS 485, dass eine Wartezeit von typisch 10 ms zwischen dem Empfang von Daten und dem Senden von Daten programmiert sein muss.

- CT-Present-Meldung mit '*'

Bei CT-Present-Meldung mit '*' erscheint jedesmal das ASCII-Zeichen '*' (2A Hex) an der Schnittstelle, wenn ein Datenträger erkannt wird. In der Betriebsart mit 'CR' als Endeckennung wird '*' CR' und bei 'LF CR' als Endeckennung wird '*' LF CR' ausgegeben. Im Gegensatz zu CT-Present wird das Entfernen des Datenträgers nicht bemerkt.

- Dynamikbetrieb

Diese Funktion schaltet die Fehlermeldung "Kein Datenträger vorhanden" aus, d.h.:

- > Im Dynamikbetrieb wird ein Lese- oder Schreibtelegramm so lange gespeichert, bis ein Datenträger in den Arbeitsbereich des betreffenden Schreib-/Lesekopfs kommt.
- > Ohne Dynamikbetrieb wird ein Lese- oder Schreibbefehl mit der Fehlermeldung <NAK> '1' abgelehnt, wenn sich kein Datenträger im Arbeitsbereich des Schreib-/Lesekopfs befindet; die Auswerteeinheit geht in den Ruhezustand.

Konfiguration

Parameter (Fortsetzung)

- CT Present auf Ausgang 1

Ist CT Present auf Ausgang 1 gewählt, wird die LED-Meldung Codetag Present auf den Ausgang 1 gegeben. Man kann so das Vorhandensein eines Datenträgers direkt als Hardwaresignal digital weiterverarbeiten.

- Ausgänge bei CT-Present bearbeiten

Die Ausgangsfunktionen werden normalerweise nur nach einem Lesebefehl bearbeitet. Da jedoch die Datenträgererkennung ebenfalls ein Lesen des Datenträgers ist (Lesen der ersten Datenträgerseite, je nach Typ 32 oder 64 Byte), kann auch bereits mit dem Codetag Present die Ausgangsbearbeitung stattfinden. Liegen die zur Ausgangsbearbeitung verwendeten Adressen innerhalb dieser ersten Seite, kann die Auswerteeinheit ohne Befehl über die serielle Schnittstelle kleine Steuerbefehle ausführen.
-> Für sehr schnelle Vorgänge siehe auch unter Schnelle Datenträgererkennung.

- Schnelle Datenträgererkennung

Für sehr schnelle Vorgänge kann die Anzahl der verwendeten Datenträgeradressen für die Datenträgererkennung von 32 bzw. 64 Byte auf 4 Byte reduziert werden. Die Datenträgererkennung verkürzt sich dadurch auf ca. 50 ms (anstatt ca. 150 ms bei Datenträgern < 2 kByte bzw. ca. 250 ms bei Datenträgern ≥ 2 kByte).

-> Beachten Sie dies bei Verwendung von Parameter "Ausgänge bei CT-Present bearbeiten".

Konfiguration

Parameter (Fortsetzung)

– CRC_16 Datenprüfung

Ist die CRC_16-Datenprüfung aktiviert, wird bei Erkennen eines CRC-Fehlers eine spezielle Fehlermeldung an die Schnittstelle ausgegeben.

Wenn die Fehlermeldung keine Folge aus einem mißglückten Schreibauftrag ist, kann davon ausgegangen werden, dass eine oder mehrere Speicherzellen auf dem Datenträger defekt sind. Der betreffende Datenträger ist auszutauschen.

Ist der CRC-Fehler jedoch eine Folge aus einem mißglückten Schreibauftrag, muss der Datenträger neu initialisiert werden, um ihn wieder verwenden zu können.

Die Prüfsumme wird je Seite auf den Datenträger als 2 Byte große Information geschrieben. Es gehen 2 Byte pro Seite verloren, d.h. die Seitengröße beträgt 30 Byte bzw. 62 Byte je nach Datenträgertyp (Seitengröße siehe nächste ).

Dies bedeutet, dass sich die konkret nutzbare Anzahl Byte verringert:

Datenträgertyp	Nutzbare Byte
128 Byte	= 120 Byte
256 Byte	= 240 Byte
511 Byte *)	= 450 Byte
1023 Byte *)	= 930 Byte
2047 Byte *)	= 1922 Byte
2048 Byte	= 1984 Byte
8192 Byte	= 7936 Byte

*) Die letzte Datenträgerseite steht bei diesen EEPROM-Datenträgern nicht zur freien Verfügung.

Konfiguration

Seitengröße

Bei der Speicherorganisation der Datenträger unterscheidet man zwischen den beiden Seitengrößen 32 Byte und 64 Byte (wird auch mit Blockgröße bezeichnet).
Werkseinstellung: 32 Byte.

Datenträger < 2 kB haben 32 Byte Blockgröße,
Datenträger ≥ 2 kB haben 64 Byte Blockgröße.

Eingang

Für den digitalen Steuereingang des BIS C-6_0 wird ausgewählt, mit welcher Funktion er belegt werden soll. Werkseinstellung ist Reset.

– Reset

Ist Reset ausgewählt, bewirkt ein High-Signal an diesem Eingang einen Reset der Auswerteeinheit BIS C-6_0 aus. Alle anstehenden Befehle werden gelöscht.

– Kopfanwahl

Ist Kopfanwahl ausgewählt, erfolgt die Kopfanwahl über diesen Eingang.

Eingang auf Low: Kopf 1 ausgewählt.

Eingang auf High: Kopf 2 ausgewählt.

-> Diese Funktion hat stets Vorrang. So ist z.B. die Funktion "Beide Schreib-/Leseköpfe aktiv" deaktiviert, die über den Befehl 'HT' ausgewählt wird.

– Datenbit auf Datenträger

Mit dem Erkennen eines neuen Datenträgers wird ein frei definiertes Bit einer anzugebenden Adresse direkt oder invertiert auf den Datenträger geschrieben. Nach erfolgreichem Schreiben wird der ebenfalls zu definierende Ausgang so lange gesetzt, bis der Datenträger den aktiven Schreib-/Lesebereich verläßt.

-> Der Parameter Dynamik-Betrieb wird automatisch zurückgesetzt.

– Keine Funktion

Der Eingang wird nicht bearbeitet.

Konfiguration

Eingang/Ausgänge zuweisen

Den Ausgängen können verschiedene Funktionen zugewiesen werden. Die Ausgangsfunktionen werden immer beim Lesen bearbeitet. Bedingung für die Ausführung ist, dass die jeweilige Adresse beim vorangegangenen Leseauftrag gelesen wurde.

EINGANG/AUSGÄNGE ZUWEISEN	
<input checked="" type="checkbox"/>	Keine Verwendung der Ausgänge.
<input type="checkbox"/>	Halbbyte des Dateninhalts einer Adresse ausgeben
<input type="checkbox"/>	Inhalt mehrerer Adressen mit einem Festwert vergleichen.
<input type="checkbox"/>	Inhalt einer Adresse mit unterschiedlichen Festwerten vergleichen.
<input type="checkbox"/>	Inhalt mehrerer Adressen mit dem Inhalt einer Adresse vergleichen.
<input type="checkbox"/>	Datenbits variabler Adressen ausgeben.
<input type="checkbox"/>	Eingang als Datenbit auf Datenträger programmieren.
< OK > < Kurzform > < ESC = Abbruch > < Drucken > < F1 = Hilfe >	

Keine Ausgänge zuweisen

Bei "Keine Verwendung der Ausgänge" ist die Bearbeitung der Ausgänge deaktiviert.

KEINE AUSGÄNGE ZUWEISEN	
<input checked="" type="checkbox"/>	Daten an BIS
< Speichern > < ESC = Abbruch > < F1 = Hilfe >	

"Daten an BIS" überträgt die Daten an die Auswertereinheit. "Speichern" legt die Daten in der Konfigurationsdatei auf Ihrem Computer ab.

Konfiguration

Halbbyte des Dateninhalts einer Adresse ausgeben

EINGANG/AUSGÄNGE ZUWEISEN	
<input type="checkbox"/>	Keine Verwendung der Ausgänge.
<input checked="" type="checkbox"/>	Halbbyte des Dateninhalts einer Adresse ausgeben
<input type="checkbox"/>	Inhalt mehrerer Adressen mit einem Festwert vergleichen.
<input type="checkbox"/>	Inhalt einer Adresse mit unterschiedlichen Festwerten vergleichen.
<input type="checkbox"/>	Inhalt mehrerer Adressen mit dem Inhalt einer Adresse vergleichen.
<input type="checkbox"/>	Datenbits variabler Adressen ausgeben.
<input type="checkbox"/>	Eingang als Datenbit auf Datenträger programmieren.
< OK > < ESC = Abbruch > < Drucken > < F1 = Hilfe >	

HALBBYTE AUSGABE	
Datenträgeradresse [9999]	
<input type="checkbox"/>	High Nibble ausgeben ?
<input checked="" type="checkbox"/>	Low Nibble ausgeben ?
< Daten an BIS > < Speichern > < ESC = Abbruch > < F1 = Hilfe >	

Es werden jeweils 4 bit (Nibble) des 8 bit umfassenden Dateninhalts einer Adresse an den 4 Ausgängen ausgegeben (Bit 0 an Ausgang 1, Bit 1 an Ausgang 2 usw.). Je nach Einstellung sind es entweder die oberen (High Nibble) oder die unteren (Low Nibble). Die Adresse wird dezimal vorgegeben.

Konfiguration

Inhalt mehrerer Adressen mit einem Festwert vergleichen

Die Dateninhalte von bis zu 4 dezimal angegebenen Adressen werden mit einem dezimalen Festwert verglichen. Zu jeder Adresse kann angegeben werden, welcher der Ausgänge 1 bis 4 bei einem positiven Ergebnis des Vergleichs gesetzt oder gelöscht werden soll und ob der Ausgang bei einem negativen Ergebnis des Vergleichs nicht verändert oder im Gegensatz zur Definition bei positivem Vergleich geschaltet werden soll (inverses Verhalten). Es wird jede Adresse bearbeitet, die sich innerhalb des Lesebefehls befindet.

Adressen mit einem Festwert vergleichen			
Festwert: [000]			
Adresse	Ausgang	Vergleich positiv	Vergleich negativ
[9999]	[1]	<input type="checkbox"/> setzen <input type="checkbox"/> löschen	<input type="checkbox"/> nicht verändern <input checked="" type="checkbox"/> invertieren
[9999]	[2]	<input type="checkbox"/> setzen <input type="checkbox"/> löschen	<input type="checkbox"/> nicht verändern <input checked="" type="checkbox"/> invertieren
[9999]	[3]	<input type="checkbox"/> setzen <input type="checkbox"/> löschen	<input type="checkbox"/> nicht verändern <input checked="" type="checkbox"/> invertieren
[9999]	[4]	<input type="checkbox"/> setzen <input type="checkbox"/> löschen	<input type="checkbox"/> nicht verändern <input checked="" type="checkbox"/> invertieren
<input <="" <input="" td="" type="button" value=" < F1 = Hilfe > "/>			

Ist in der Initialisierung der Parameter "Ausgänge bei CT-Present bearbeiten" angegeben, dann wird diese Funktion auch bei einer Neuerkennung eines Datenträgers ausgeführt (eine oder mehrere der angegebenen Adressen sollten sich dann in der ersten Datenträgerseite befinden).

Konfiguration

Inhalt einer Adresse mit unterschiedlichen Festwerten vergleichen

Der Dateninhalt einer dezimal angegebenen Adresse wird mit 4 Dezimal-Festwerten verglichen. Zu jedem Festwert kann angegeben werden, welcher der Ausgänge 1 bis 4 bei einem positiven Ergebnis des Vergleichs gesetzt oder gelöscht und ob der Ausgang bei einem negativen Ergebnis des Vergleichs nicht verändert oder im Gegensatz zur Definition bei positivem Vergleich geschaltet werden soll (inverses Verhalten).

Adressen mit unterschiedlichen Festwerten vergleichen			
Adresse: [9999]			
Festwert	Ausgang	Vergleich positiv	Vergleich negativ
[000]	[1]	<input type="checkbox"/> setzen <input type="checkbox"/> löschen	<input type="checkbox"/> nicht verändern <input checked="" type="checkbox"/> invertieren
[000]	[2]	<input type="checkbox"/> setzen <input type="checkbox"/> löschen	<input type="checkbox"/> nicht verändern <input checked="" type="checkbox"/> invertieren
[000]	[3]	<input type="checkbox"/> setzen <input type="checkbox"/> löschen	<input type="checkbox"/> nicht verändern <input checked="" type="checkbox"/> invertieren
[000]	[4]	<input type="checkbox"/> setzen <input type="checkbox"/> löschen	<input type="checkbox"/> nicht verändern <input checked="" type="checkbox"/> invertieren
<input <="" <input="" td="" type="button" value=" < F1 = Hilfe > "/>			

Ist in der Initialisierung der Parameter "Ausgänge bei CT-Present bearbeiten" angegeben, dann wird diese Funktion auch bei einer Neuerkennung eines Datenträgers ausgeführt (eine oder mehrere der angegebenen Adressen sollten sich dann in der ersten Datenträgerseite befinden).

Konfiguration

Inhalt mehrerer Adressen mit dem Inhalt einer Adresse vergleichen

Die Dateninhalte von bis zu 4 dezimal angegebenen Adressen werden mit dem Dateninhalt einer weiteren Adresse verglichen. Zu jeder Adresse kann angegeben werden, welcher der Ausgänge 1 bis 4 bei einem positiven Ergebnis des Vergleichs gesetzt oder rückgesetzt werden soll und ob der Ausgang bei einem negativen Ergebnis des Vergleichs nicht verändert oder im Gegensatz zur Definition bei positivem Vergleich geschaltet werden soll. Es wird jede Adresse bearbeitet, die sich innerhalb des Lesebefehls befindet, vorausgesetzt, die Vergleichsadresse befindet sich innerhalb des Bereichs.

Inhalt mehrerer Adressen mit Inhalt anderer Adresse vergleichen			
Adresse: [9999]			
Adresse	Ausgang	Vergleich positiv	Vergleich negativ
[9999]	[1]	<input type="checkbox"/> setzen <input type="checkbox"/> löschen	<input type="checkbox"/> nicht verändern <input type="checkbox"/> invertieren
[9999]	[2]	<input type="checkbox"/> setzen <input type="checkbox"/> löschen	<input type="checkbox"/> nicht verändern <input type="checkbox"/> invertieren
[9999]	[3]	<input type="checkbox"/> setzen <input type="checkbox"/> löschen	<input type="checkbox"/> nicht verändern <input type="checkbox"/> invertieren
[9999]	[4]	<input type="checkbox"/> setzen <input type="checkbox"/> löschen	<input type="checkbox"/> nicht verändern <input type="checkbox"/> invertieren

< Daten an BIS > < Speichern > < ESC = Abbruch > < F1 = Hilfe >

Ist in der Initialisierung der Parameter "Ausgänge bei CT-Present bearbeiten" angegeben, dann wird diese Funktion auch bei einer Neuerkennung eines Datenträgers ausgeführt (eine oder mehrere der angegebenen Adressen sollten sich dann in der ersten Datenträgerseite befinden).

Konfiguration

Datenbits variabler Adressen ausgeben

Jeweils 1 Datenbit von bis zu 4 dezimal angegebenen Adressen kann an einen der 4 Ausgänge ausgegeben und dabei invertiert oder nicht invertiert werden.

Datenbits variabler Adressen ausgeben			
Adresse	Bit-Nummer	Ausgang	invertieren
[9999]	[1]	[1]	<input type="checkbox"/> ja
[9999]	[2]	[1]	<input type="checkbox"/> ja
[9999]	[3]	[1]	<input type="checkbox"/> ja
[9999]	[4]	[1]	<input type="checkbox"/> ja

< Daten an BIS > < Speichern > < ESC = Abbruch > < F1 = Hilfe >

Ist in der Initialisierung der Parameter "Ausgänge bei CT-Present bearbeiten" angegeben, dann wird diese Funktion auch bei einer Neuerkennung eines Datenträgers ausgeführt (eine oder mehrere der angegebenen Adressen sollten sich dann in der ersten Datenträgerseite befinden).

Konfiguration

Eingang als Datenbit auf Datenträger programmieren

Beim Erkennen eines neuen Datenträgers wird der Zustand des digitalen Eingangs direkt oder invertiert als Bit auf den Datenträger geschrieben. Zulässiger Adressbereich: 0...31! Bitnummer der Adresse: 1...8. Diegänge für das Quittungs- und das Freigabesignal sind ebenfalls anzugeben. Bei Freigabeausgang = 0 wird das Freigabesignal nicht verwendet. Nachfolgend wird der Ablauf beschrieben.

Eingang als Datenbit auf Datenträger programmieren			
Adresse	[00]	[]	Eingang invertieren ?
Bitnummer	[1]		
Quittungsausgang	[1]		
Freigabeausgang	[1]		
< Daten an BIS >		< Speichern >	< ESC = Abbruch > < F1 = Hilfe >

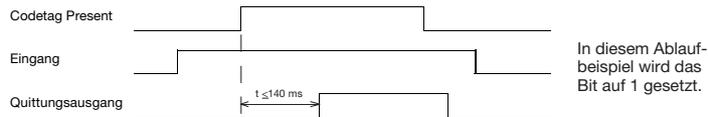
Ablauf ohne Freigabesignal

Mit dem Erkennen eines neuen Datenträgers wird das definierte Bit der angegebenen Adresse direkt oder invertiert programmiert. Nach erfolgreichem Schreiben wird der angegebene Quittungsausgang so lange gesetzt, bis der Datenträger den aktiven Schreib-/Lesebereich verläßt. Der Eingang muss so lange seinen Zustand beibehalten, bis der Quittungsausgang gesetzt ist.

Der Eingangszustand, der als Information auf den Datenträger geschrieben werden soll, muss bereits vor dem Erkennen des Datenträgers anstehen.

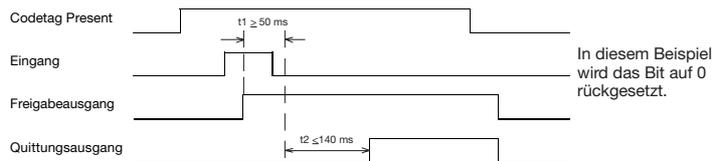
Konfiguration

Ablauf ohne Freigabesignal (Fortsetzung)



Ablauf mit Freigabesignal

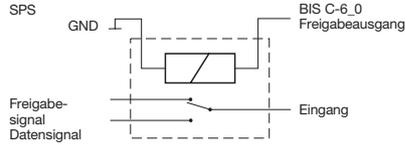
Nach dem Erkennen eines Datenträgers wird der Eingang so lange abgefragt, bis er gesetzt ist (Freigabe erteilt). Die Auswerteeinheit setzt den Freigabeausgang und wartet 50 ms. Danach wird der Zustand des Eingangs abgefragt und als Datenwert übernommen. Dieser wird je nach Vorgabe direkt oder invertiert auf den Datenträger geschrieben. Nach dem Schreiben wird der Quittungsausgang so lange gesetzt, bis der Datenträger den Schreib-/Lesebereich verlassen hat. Jetzt schaltet der Freigabeausgang wieder auf Low.



Konfiguration

Ablauf mit Freigabesignal (Fortsetzung)

Den Freigabeausgang kann man zum Schalten eines Relais verwenden, um damit das Freigabesignal und das Datensignal auf den Eingang zu schalten.



Als Eingangssignal in eine SPS kann dieser Ausgang anzeigen, dass als nächstes das Datensignal auf den Eingang des BIS C-6_0 geschaltet werden muss. Erforderlich ist dieser Ausgang überall, wo der Datenträger in den Bereich des Schreib-/Lesekopfs kommen kann, bevor das Datensignal am Eingang der Auswerteeinheit steht.

Diese Funktion muss zusätzlich in der Initialisierung freigegeben werden.

KURZFORM DER E/A EINSTELLUNGEN BIS C-6_0			
[0000]	[0]	[0]	[0]
[0000]	[0]	[0]	[0]
[0000]	[0]	[0]	[0]
[0000]	[0]	[0]	[0]
[0000]	[0]	[0]	[1]

< Z = <- > < ESC = Abbruch > < F1 = Hilfe >

Wenn die Initialisierungsdaten in Kurzform vorliegen (z.B. auf dem Gehäusedeckel nach einem Gerätetausch), dann kann die Eingabe direkt in die Maske "Kurzform der E/A Einstellungen BIS C-6_0" eingetragen werden (siehe auch Kundenkonfiguration auf 13).

Kundenkonfiguration

Initialisierung

Bitte tragen Sie die Einstellungen in die Felder auf der Innenseite des Gerätedeckels ein, damit die Einstellungen bei einer Reparatur vom Werk wieder eingegeben werden können. Tragen Sie sie auch in die nachstehenden Felder ein, damit Sie identische Einstellungen z.B. bei weiteren Geräten vornehmen können.

	Baudrate	Datenbit	Stopbit	Parität
Schnittstelle	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Protokolltyp	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
		Seitengröße		
		<input type="text"/>	Eingang	
Parameter	<input type="text"/>			

Ändern Sie keinesfalls die hier angekreuzten Felder.

Auf der folgenden Seite finden Sie ein Beispiel wiedergegeben, wie Sie es sich nach der Initialisierung ausdrucken lassen können. Übertragen Sie die Einstellungen in die obigen Felder, damit Sie die Werte jederzeit zur Hand haben und reproduzieren können. Sie können diese Daten dann in Kurzform in die Maske eingeben (siehe auch 13).

KURZFORM DER EINSTELLUNGEN BIS C-6_0			
[19200]	[8]	[1]	[N]
[4]	[<input checked="" type="checkbox"/>]	[<input checked="" type="checkbox"/>]	[2] [1]
[0110110]			

< Z = <- > < ESC = Abbruch > < F1 = Hilfe >

Ändern Sie keinesfalls die hier angekreuzten Felder.

Kundenkonfiguration

Initialisierung
(Fortsetzung)

Beispiel eines Ausdrucks, wie Sie ihn durch das Programm BISC600C.EXE nach der Initialisierung anfertigen können.

Schnittstelleneinstellung

Übertragungsrate : 19200 baud
 Datenbit : 8
 Stopbit : 1
 Parität : None

Protokolltyp

- BCC
- CR als Endeckennung
- CR
- LF CR als Endeckennung

Parameter

- RS 485, 2-Draht.
- Codetag Present mit einem '*' melden
- Dynamik-Betrieb
- Codetag Present auf Ausgang 1
- Ausgänge bei Codetag Present bearbeiten
- Schnelle Datenträgererkennung
- CRC_16-Datenprüfung

Seitengröße

- 32 Byte Seitengröße
- 64 Byte Seitengröße

Eingang

- Eingang = RESET
- Eingang = Kopfanwahl
- Eingang = Datenbitinformation auf Datenträger
- Eingang = Keine Funktion

1	9	2	0	0	8	1	N
4	2	2	2	2	1		
0	1	1	0	1	1	0	

Die Eintragung in die Felder erfolgt entweder als Klartext (wie bei Schnittstelle) oder es wird die Nummer der markierten Zeile eingetragen. Bei Parameter wird die markierte Position durch eine 1 gekennzeichnet.

*) Diese beiden Felder dürfen nicht überschrieben werden!

Kundenkonfiguration

Ein-/Ausgangs-konfiguration

Bitte tragen Sie die Einstellungen in die Felder auf der Innenseite des Gerätedeckels ein, damit die Einstellungen bei einer Reparatur vom Werk wieder eingegeben werden können. Tragen Sie sie auch in die nachstehenden Felder ein, damit Sie identische Einstellungen z.B. bei weiteren Geräten vornehmen können.

Adresse

Adresse

Adresse

Adresse

Festwert Typ

Auf der folgenden Seite finden Sie ein Beispiel wiedergegeben, wie Sie es sich nach der Initialisierung ausdrucken lassen können. Übertragen Sie die Einstellungen in die obigen Felder, damit Sie die Werte jederzeit zur Hand haben und reproduzieren können. Sie können diese Daten dann in Kurzform in die Maske eingeben.

Kurzform der E/A-Einstellungen BIS C-6_0

KURZFORM DER E/A EINSTELLUNGEN BIS C-6_0

```
[0000] [0] [0] [0]
[0000] [0] [0] [0]
[0000] [0] [0] [0]
[0000] [0] [0] [0]
[0000] [0] [0] [1]
```

< Z = <- > < ESC = Abbruch > < F1 = Hilfe >

Kundenkonfiguration

Ein-/Ausgangs-konfiguration (Fortsetzung)

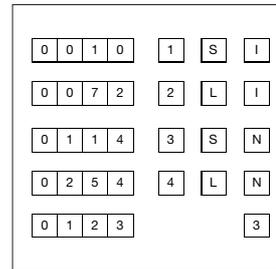
Beispiel eines Ausdrucks, wie Sie ihn durch das Programm BISC600C.EXE nach der Ein-/Ausgangskonfiguration anfertigen können.

Zuweisung

- Keine Verwendung der Ausgänge.
- Halbbyte des Dateninhalts einer Adresse ausgeben.
- Inhalt mehrer Adressen mit einem Festwert vergleichen.
- Inhalt einer Adresse mit unterschiedlichen Festwerten vergleichen.
- Inhalt mehrer Adressen mit dem Inhalt einer Adresse vergleichen.
- Datenbits variabler Adressen ausgeben.
- Eingang als Datenbit auf Datenträger programmieren.

Definition

Festwert: 123
Adresse: 0010
 Ausgang: 1
 Vergleich positiv: setzen nicht verändern
 Vergleich negativ: löschen invertieren
Adresse: 0072
 Ausgang: 2
 Vergleich positiv: setzen nicht verändern
 Vergleich negativ: löschen invertieren
Adresse: 0114
 Ausgang: 3
 Vergleich positiv: setzen nicht verändern
 Vergleich negativ: löschen invertieren
Adresse: 0254
 Ausgang: 4
 Vergleich positiv: setzen nicht verändern
 Vergleich negativ: löschen invertieren



Schreib-/Lesezeiten

Lesezeiten vom Datenträger bis zur Auswerteeinheit im statischen Betrieb (Konfiguration: ohne Dynamikbetrieb, keine CRC_16-Datenprüfung)

Für zweimaliges Lesen und Vergleichen:

Datenträger mit 32 Byte je Block	
Anzahl Byte	Lesezeit [ms]
von 0 bis 31	110
für jeweils weitere angebrochene 32 Byte addieren Sie weitere	
von 0 bis 255	= 120
	= 950

Datenträger mit 64 Byte je Block	
Anzahl Byte	Lesezeit [ms]
von 0 bis 63	220
für jeweils weitere angebrochene 64 Byte addieren Sie weitere	
von 0 bis 2047	= 230
	= 7350

Schreibzeiten von der Auswerteeinheit auf den Datenträger im statischen Betrieb (Konfiguration: ohne Dynamikbetrieb, keine CRC_16-Datenprüfung)

Inclusive Rücklesen und Vergleichen:

Datenträger mit 32 Byte je Block	
Anzahl Byte	Schreibzeit [ms]
von 0 bis 31	110 + n * 10
≥ 32	y * 120 + n * 10

Datenträger mit 64 Byte je Block	
Anzahl Byte	Schreibzeit [ms]
von 0 bis 63	220 + n * 10
≥ 64	y * 230 + n * 10

n = Anzahl der zusammenhängend zu schreibenden Bytes
 y = Anzahl der zu bearbeitenden Blöcke

Beispiel:

Es sollen 17 Byte ab Adresse 187 geschrieben werden. Datenträger = 32 Byte je Block. Bearbeitet werden Block 5 und 6, da Anfangsadresse 187 in Block 5 und Endadresse 203 in Block 6 ist.
 $t = 2 * 120 + 17 * 10 = 410$

Die angegebenen Zeiten gelten, nachdem der Datenträger erkannt wurde. Andernfalls müssen für den Energieaufbau bis zum Erkennen des Datenträgers 45 ms hinzugerechnet werden.

Schreib-/Lesezeiten

Lesezeiten vom Datenträger bis zur Auswerteeinheit im dynamischen Betrieb
(Konfiguration: mit Dynamikbetrieb, keine CRC_16-Datenprüfung)

Lesezeiten innerhalb des 1. Blocks für zweimaliges Lesen und Vergleichen:

Datenträger mit 32 Byte je Block		Datenträger mit 64 Byte je Block	
Anzahl Byte	Lesezeit [ms]	Anzahl Byte	Lesezeit [ms]
von 0 bis 3	= 14	von 0 bis 3	= 14
für jedes weitere Byte	3,5	für jedes weitere Byte	3,5
von 0 bis 31	= 112	von 0 bis 63	= 224

m = größte zu lesende Adresse

Formel: $t = (m + 1) * 3,5 \text{ ms}$

Beispiel: Es sollen 11 Byte ab Adresse 9 gelesen werden. D.h. die größte zu lesende Adresse ist 19.
Dies ergibt 70 ms.

Schreibzeiten von der Auswerteeinheit auf den Datenträger im dynamischen Betrieb
(Konfiguration: mit Dynamikbetrieb, keine CRC_16-Datenprüfung)

Inklusive Rücklesen und Vergleichen:

Datenträger mit 32 Byte je Block		Datenträger mit 64 Byte je Block	
Anzahl Byte	Schreibzeit [ms]	Anzahl Byte	Schreibzeit [ms]
von 0 bis 3	= $14 + n * 10$	von 0 bis 3	= $14 + n * 10$
für jedes weitere Byte	3,5	für jedes weitere Byte	3,5

n = Anzahl der zusammenhängend zu schreibenden Bytes

Die angegebenen Zeiten gelten, nachdem der Datenträger erkannt wurde. Andernfalls müssen für den Energieaufbau bis zum Erkennen des Datenträgers 45 ms hinzugerechnet werden.

Programmierinformationen

Nachdem in den vorangegangenen Kapiteln der prinzipielle Telegrammablauf und die Konfiguration dargestellt wurden, folgen nun die Informationen zum korrekten Aufbau der Telegramme.

Für die einzelnen Aufgaben im Identifikations-System BIS C existieren spezifische Telegramme. Sie beginnen stets mit dem Befehl, der der Telegrammart zugeordnet ist:

Telegrammart mit zugehörigem Befehl (ASCII-Zeichen)

- 'L' Lesen des Datenträgers mit Anwahl des Schreib-/Lesekopfes und der Blockgröße
- 'P' Schreiben auf den Datenträger mit Anwahl des Schreib-/Lesekopfes und der Blockgröße
- 'Q' Schreiben eines konstanten Wertes auf den Datenträger mit Anwahl des Schreib-/Lesekopfes und der Blockgröße
- 'R' Lesen des Datenträgers
- 'W' Schreiben auf den Datenträger
- 'H' Anwahl des Schreib-/Lesekopfes und der Blockgröße mit den Varianten
 - '?' Suchen des nächsten Datenträgers (einmal)
 - !' oder Suchen des nächsten Datenträgers (ständig)
- 'B' Ausgänge bearbeiten
- 'Q' Neustart der Auswerteeinheit (Quitt)
- 'S' Abfrage der Statusmeldung
- 'Z' CRC_16-Datenprüfung initialisieren

Bitte beachten Sie:

- Eine Dauerabfrage auf der Schnittstelle ist nicht zulässig!
- Die Mindestwartezeit zwischen zwei Befehlen beträgt 300 ms!

Programmierinformationen

Erklärung einiger Telegramminhalte

Startadresse und Anzahl Bytes	Die Startadresse (A3, A2, A1, A0) und die Anzahl der zu übertragenden Bytes (L3, L2, L1, L0) werden dezimal als ASCII-Zeichen übertragen. Für die Startadresse kann der Bereich 0000 bis 8191 und für die Anzahl Byte 0001 bis 8192 verwendet werden. A3 ... L0 stehen für je ein ASCII-Zeichen. Bitte beachten Sie: Startadresse + Anzahl Byte dürfen die Datenträgerkapazität nicht überschreiten.
Kopfnummer und Blockgröße	Bei den Befehlen 'L' (Lesen mit Kopfanwahl und Blockgröße) und 'P' (Schreiben mit Kopfanwahl und Blockgröße) wird zuerst die Nummer des Schreib-/Lesekopfes K ('1' oder '2') und danach die Blockgröße B ('0', '1') des Datenträgers übertragen. B = '0' entspricht 64 Byte, B = '1' entspricht 32 Byte.
Quittung	Die Quittung <ACK> '0' wird vom Identifikations-System gesendet, wenn die seriell übertragenen Zeichen als richtig erkannt wurden und sich ein Datenträger im Arbeitsbereich eines Schreib-/Lesekopfes befindet. Beim Befehl 'R' wird <ACK> '0' erst gegeben, wenn die Daten zur Übertragung bereit sind. Mit <NAK> + Fehlernr.' wird quittiert, wenn ein Fehler erkannt wurde oder wenn sich kein Datenträger im Arbeitsbereich des Schreib-/Lesekopfes befindet.
Start	Mit <STX> wird die Datenübertragung gestartet.
Übertragene Bytes	Die Daten werden codetransparent (ohne Datenwandlung) übertragen.

Programmierinformationen

Bildung des Blockchecks BCC

Der Blockcheck BCC wird als EXOR-Verknüpfung aus den seriell übertragenen Binärzeichen des Telegrammblocks gebildet. Beispiel: Lesen ab Adresse 13, 128 Byte sind zu lesen. Die Befehlszeile ohne BCC lautet: 'L 0013 0128 20'. BCC wird gebildet:

```
'L = 0100 1100 EXOR
0 = 0011 0000 EXOR
0 = 0011 0000 EXOR
1 = 0011 0001 EXOR
3 = 0011 0011 EXOR
0 = 0011 0000 EXOR
1 = 0011 0001 EXOR
2 = 0011 0010 EXOR
8 = 0011 1000 EXOR
2 = 0011 0010 EXOR
0' = 0011 0000 EXOR
```

ergibt als Blockcheck: BCC = 0100 0111 = 'G'

Variante bei Abschluss mit BCC, Endekennung

Bei Bedarf kann der Abschluss mittels Blockcheck BCC durch ein spezielles ASCII-Zeichen ersetzt werden. Dies ist:

– Carriage Return 'CR'

Für Steuereinheiten, die immer ein Endekennungszeichen benötigen, muss dieses überall in die Telegramme eingefügt werden. Zur Verfügung stehen:

– Carriage Return 'CR' oder
– Line Feed mit Carriage Return 'LF CR'.

Auf der folgenden  werden die verschiedenen Protokollvarianten dargestellt. Siehe auch: Konfiguration ab  12.

Programmierinformationen

Darstellung der verschiedenen Protokollvarianten

Von der vorangegangenen Seite stammt die Befehlszeile 'L 0013 0128 20 G' mit 'G' als BCC. Diese Befehlszeile wird hier in den möglichen Varianten gegenübergestellt; dabei werden auch die verschiedenen Formen der Quittung mit und ohne Endeckennung dargestellt:

Befehlszeile vom steuernden System zum BIS	Quittung vom BIS bei korrektem Empfang	Quittung vom BIS bei inkorrektem Empfang
mit BCC als Abschluss, ohne Endeckennung 'L 0013 0128 20 G'	ohne Endeckennung <ACK> '0'	ohne Endeckennung <NAK> '1'
mit 'CR' anstatt BCC, ohne Endeckennung 'L 0013 0128 20 CR'	ohne Endeckennung <ACK> '0'	ohne Endeckennung <NAK> '1'
ohne BCC, mit Endeckennung 'CR' 'L 0013 0128 20 CR'	mit Endeckennung 'CR' <ACK> '0 CR'	mit Endeckennung 'CR' <NAK> '1 CR'
ohne BCC, mit Endeckennung 'LF CR' 'L 0013 0128 20 LF CR'	mit Endeckennung 'LF CR' <ACK> '0 LF CR'	mit Endeckennung 'LF CR' <NAK> '1 LF CR'

Bei <NAK> mit Fehlernummer wurde hier '1' (kein Datenträger vorhanden) als Fehlerbeispiel ausgegeben.

Die jeweiligen Positionen für die zusätzliche Endeckennung sind in den tabellarischen Darstellungen kursiv abgesetzt.

Programmierinformationen

Lesen vom Datenträger mit Anwahl des Schreib-/Lesekopfs und der Blockgröße Schreiben auf den Datenträger mit Anwahl des Schreib-/Lesekopfs und der Blockgröße

Task	Datenfluss	Be- fehl	Startadresse des ersten zu übertra- genden Byte	Anzahl der zu über- tragenden Bytes	Kopf- num- mer	Block- größe	Ab- schluss	Quit- tung	Ende- ken- nung	Start zur Über- tragung	Ende- ken- nung	Daten (von Startadresse bis Startadresse + Anzahl Bytes)	Ab- schluss	Quit- tung	Ende- ken- nung
Lesen	vom steuernden System zum BIS	L	A3 A2 A1 A0 '0 0 0 0' bis '8 1 9 1'	L3 L2 L1 L0 '0 0 0 1' '1' oder '8 1 9 2' '2'	K	B	BCC oder siehe 2)			<STX>	'CR' oder 'LF' CR'				
	vom BIS zum steuernden System							<ACK>'0' oder <NAK> + Fehler-Nr.	'CR' oder 'LF' CR'			D1 D2 D3 ... Dn	BCC oder siehe 2)		
			1)								1)				
Schreiben	vom steuernden System zum BIS	P	A3 A2 A1 A0 '0 0 0 0' bis '8 1 9 1'	L3 L2 L1 L0 '0 0 0 1' '1' oder '8 1 9 2' '2'	K	B	BCC oder siehe 2)			<STX>					
	vom BIS zum steuernden System							<ACK>'0' oder <NAK> + Fehler-Nr.	'CR' oder 'LF' CR'						<ACK>'0' oder <NAK> + Fehler-Nr.
			1)								1)				

- 1) Die Befehle 'Status' und/oder 'Quitt' sind an dieser Stelle nicht zugelassen.
- 2) Statt Blockcheck BCC kann je nach Protokollvariante entweder Carriage Return 'CR' oder Line Feed mit Carriage Return 'LF CR' verwendet werden.
- 3) Als Quittung kommt <ACK> '0', wenn kein Fehler aufgetreten ist, oder <NAK> + Fehler-Nr., wenn ein Fehler aufgetreten ist.
- 4) Bei Protokollvarianten, die immer eine Endeckennung benötigen, muss hier eines der Abschlusszeichen 'CR' oder 'LF CR' eingefügt werden.

Angaben in Hochkommata stellen das/die jeweilige/n Zeichen im ASCII-Code dar.

Programmierinformationen

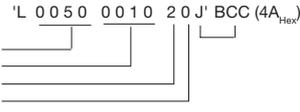
Telegrammbeispiel zu 38:

Lesen vom Code-träger mit Anwahl des Schreib-/Lesekopfs und der Blockgröße mit Blockcheck (BCC)

-> Kopf 1 ist angewählt. Es sollen 10 Byte ab Adresse 50 vom Datenträger am Schreib-/Lesekopf 2 gelesen werden. Der Datenträger vor Kopf 4 hat eine Blockgröße von 64 Byte.

Das Steuersystem sendet

Adresse des ersten zu lesenden Byte
Anzahl der zu lesenden Byte
Schreib-/Lesekopf Nr. 2
Blockgröße 0 = 64 Byte



Die BIS-Auswerteeinheit quittiert mit <ACK> '0'
Das Steuersystem gibt den Startbefehl <STX>
Die BIS-Auswerteeinheit liefert die Daten vom Datenträger 1 2 3 4 5 6 7 8 9 0 '1' BCC (31 Hex)
Nach Ablauf des Telegrammverkehrs bleibt Kopf 2 mit 64 Byte Blockgröße angewählt.

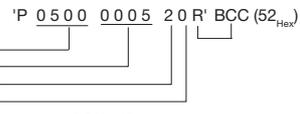
Telegrammbeispiel zu 38:

Schreiben auf den Datenträger mit Anwahl des Schreib-/Lesekopfs und der Blockgröße mit Blockcheck (BCC)

-> Kopf 1 ist angewählt. Es sollen 5 Byte ab Adresse 500 auf den Datenträger am Schreib-/Lesekopf 2 geschrieben werden. Der Datenträger vor Kopf 2 hat eine Blockgröße von 64 Byte.

Das Steuersystem sendet

Adresse des ersten zu schreibenden Byte
Anzahl der zu schreibenden Byte
Schreib-/Lesekopf Nr. 2
Blockgröße 0 = 64 Byte



Die BIS-Auswerteeinheit quittiert mit <ACK> '0'
Das Steuersystem gibt den Startbefehl und die Daten <STX> 1 2 3 4 5 '3' BCC (33 Hex)
Die Auswerteeinheit quittiert mit <ACK> '0'
Nach Ablauf des Telegrammverkehrs bleibt Kopf 2 mit 64 Byte Blockgröße angewählt.

Angaben in Hochkommata stellen das/die jeweilige/n Zeichen im ASCII-Code dar.

Programmierinformationen

Schreiben eines konstanten Wertes auf den Datenträger mit Anwahl des Schreib-/Lesekopfs und der Blockgröße

Dieser Befehl kann zum Löschen eines Datenträgers verwendet werden. Man spart die Zeit zur Übertragung der zu schreibenden Byte.

Task	Datenfluss	Befehl	Startadresse des ersten zu übertragenden Bytes	Anzahl der zu übertragenden Bytes	Kopfnummer	Blockgröße	Ab-schluss	Quit-tung	Ende-ken-nung	Start zur Über-tragung	Ende-ken-nung	Daten (von Startadresse bis Startadresse + Anzahl Bytes)	Ab-schluss	Quit-tung	Ende-ken-nung	
Schreiben	vom steuernden System zum BIS	C' A3 A2 A1 A0 '0 0 0 0' bis '8 1 9 1'	L3 L2 L1 L0 '0 0 0 1' bis '8 1 9 2'	K '1' oder '2'	B '0' oder '1'	BCC oder siehe 2)				<STX>		D	BCC oder siehe 2)			
	vom BIS zum steuernden System								<ACK> '0' oder <NAK> + F-Nr.	'CR' oder 'LF' CR'				<ACK> '0' oder <NAK> + F-Nr.	'CR' oder 'LF' CR'	
			1)											1)		

- 1) Die Befehle Status und/oder Quit sind an dieser Stelle nicht zugelassen.
- 2) Statt Blockcheck BCC kann je nach Protokollvariante entweder Carriage Return 'CR' oder Line Feed mit Carriage Return 'LF CR' verwendet werden.
- 3) Als Quittung kommt <ACK> '0', wenn kein Fehler aufgetreten ist, oder <NAK> + 'Fehlernr.', wenn ein Fehler aufgetreten ist.
- 4) Bei Protokollvarianten, die immer eine Endekennung benötigen, muss hier eines der Abschlusszeichen 'CR' oder 'LF CR' eingefügt werden.

Angaben in spitzen Klammern stellen ein Steuerzeichen dar.
Angaben in Hochkommata stellen das/die jeweilige/n Zeichen im ASCII-Code dar.

Programmierinformationen

Telegrammbeispiel zu 40:
Schreiben auf den Datenträger mit Anwahl des Schreib-/Lesekopfs und der Blockgröße mit Blockcheck (BCC)

-> Kopf 1 ist angewählt. Es sollen 500 Byte ab Adresse 20 auf den Datenträger am Schreib-/Lesekopf 2 mit dem ASCII Datenwert 0 (30_{Hex}) geschrieben werden. Der Datenträger vor Kopf 2 hat eine Blockgröße 64 Byte.



Die BIS-Auswerteeinheit quittiert mit <ACK> '0'
 Das Steuersystem gibt den Startbefehl und die Daten <STX> '0 2' BCC (32_{Hex})
 Die Auswerteeinheit quittiert mit <ACK> '0'

Nach Ablauf des Telegrammverkehrs bleibt Kopf 2 mit 64 Byte Blockgröße angewählt.

Angaben in spitzen Klammern stellen ein Steuerzeichen dar.
 Angaben in Hochkommata stellen das/die jeweilige/n Zeichen im ASCII-Code dar.

Programmierinformationen

Lesen vom Datenträger, Schreiben auf den Datenträger

Task	Datenfluss	Be- fehl	Startadresse des ersten zu übertra- genden Byte	Anzahl zu übertragen- der Bytes	Ab- schluss (2)	Quit- tung (3)	Ende- ken- nung (4)	Start zur Über- tragung	Ende- ken- nung (4)	Daten (von Startadresse bis Startadresse + Anzahl Bytes)	Ab- schluss (2)	Quit- tung (3)	Ende- ken- nung (4)
Lesen	vom steuernden System zum BIS	*R'	A3 A2 A1 A0 '0 0 0 0' bis '8 1 9 1'	L3 L2 L1 L0 '0 0 0 1' oder '8 1 9 2'	BCC oder siehe 2)			<STX>	'CR' oder 'LF' 'CR'				
	vom BIS zum steuernden System						<ACK> '0' 'CR' oder <NAK> + Fehler-Nr. 'CR'			D1 D2 D3 ... Dn	BCC oder siehe 2)		
1)													
Schreiben	vom steuernden System zum BIS	*W'	A3 A2 A1 A0 '0 0 0 0' bis '8 1 9 1'	L3 L2 L1 L0 '0 0 0 1' oder '8 1 9 2'	BCC oder siehe 2)			<STX>		D1 D2 D3 ... Dn	BCC oder siehe 2)		
	vom BIS zum steuernden System						<ACK> '0' 'CR' oder <NAK> + Fehler-Nr. 'CR'					<ACK> '0' 'CR' oder <NAK> + Fehler-Nr. 'LF' 'CR'	
1)													

- 1) Die Befehle 'Status' und/oder 'Quitt' sind an dieser Stelle nicht zugelassen.
- 2) Statt Blockcheck BCC kann je nach Protokollvariante Carriage Return 'CR' oder Line Feed mit Carriage Return 'LF CR' verwendet werden.
- 3) Als Quittung kommt <ACK> '0', wenn kein Fehler aufgetreten ist, oder <NAK> + Fehler-Nr., wenn ein Fehler aufgetreten ist.
- 4) Bei Protokollvarianten, die immer eine Endekennung benötigen, muss hier eines der Abschlusszeichen 'CR' oder 'LF CR' eingefügt werden.

Angaben in Hochkommata stellen das/die jeweilige/n Zeichen im ASCII-Code dar.

Programmierinformationen

Telegrammbeispiel zu ¶ 42:
Lesen vom Datenträger mit Blockcheck (BCC)

Lesen vom Datenträger: -> Es sollen 10 Byte ab Adresse 50 gelesen werden.
Das Steuersystem sendet 'R 0050 0010 V' BCC (56_{Hex})
Adresse des ersten zu lesenden Byte
Anzahl der zu lesenden Byte
Die BIS-Auswerteeinheit quittiert mit <ACK> '0'
Das Steuersystem gibt den Startbefehl <STX>
Die BIS-Auswerteeinheit liefert die Daten vom Datenträger 1 2 3 4 5 6 7 8 9 0 'SOH' BCC (01_{Hex})

Telegrammbeispiel zu ¶ 42:
Schreiben auf den Datenträger mit Blockcheck (BCC)

Schreiben auf den Datenträger: -> Es sollen 5 Byte ab Adresse 500 geschrieben werden.
Das Steuersystem sendet 'W 0500 0005 W' BCC (57_{Hex})
Die BIS-Auswerteeinheit quittiert mit <ACK> '0'
Das Steuersystem sendet die Daten <STX> 1 2 3 4 5 '3' BCC (33_{Hex})
Die BIS-Auswerteeinheit quittiert mit <ACK> '0'

Die Befehle 'R' und 'W' stellen eine Untermenge der Befehle 'L' und 'P' dar.
Angaben in Hochkommata stellen das/die jeweilige/n Zeichen im ASCII-Code dar.

Programmierinformationen

Anwahl des Schreib-/Lesekopfs

Mit dem Befehl 'H1' kann der Schreib-/Lesekopf 1, mit 'H2' der Schreib-/Lesekopf 2 und mit 'HT' (Head Twin) können beide Köpfe angewählt werden.

Sind beide Köpfe angewählt, ist zu beachten:

1. Es darf immer nur an einem Schreib-/Lesekopf ein Datenträger vorhanden sein.
2. Die Schreib- oder Lesezeit verlängert sich um ca. 40 ms - unabhängig von der Datenanzahl, die gelesen oder geschrieben werden soll. (Dies gilt nicht für die Datenträgererkennung).
3. Die positive Quittung bei einem Schreib- oder Leseauftrag heißt nicht mehr <ACK> '0' sondern <ACK> '1' oder <ACK> '2' je nachdem, vor welchem Schreib-/Lesekopf sich gerade ein Datenträger befindet, der gelesen oder beschrieben wurde.

Task	Datenfluss	Befehl	Kopfnummer	Abschluss 2)	Quittung 3)	Endekennung 4)
Anwahl Schreib-/Lesekopf	vom steuernden System zum BIS	'H'	'1', '2' oder 'T'	BCC oder siehe 2)		
	vom BIS zum steuernden System				<ACK>'0' oder >ACK>'2' bzw. NAK> + Fehler-Nr.	'CR' oder 'LF CR'

- 1) Die Befehle Status und/oder Quitt sind an dieser Stelle nicht zugelassen.
- 2) Statt Blockcheck BCC kann je nach Protokollvariante entweder Carriage Return 'CR' oder Line Feed mit Carriage Return 'LF CR' verwendet werden.
- 3) Als Quittung kommt <ACK> '0', wenn kein Fehler aufgetreten ist, oder <NAK> + 'Fehlernr.', wenn ein Fehler aufgetreten ist.
- 4) Bei Protokollvarianten, die immer eine Endekennung benötigen, muss hier eines der Abschlusszeichen 'CR' oder 'LF CR' eingefügt werden.

Telegrammbeispiel:
Anwahl des Schreib-/Lesekopfs mit Blockcheck (BCC)

-> Es soll auf Kopf 1 umgeschaltet werden.
Das Steuersystem sendet 'H 1 y' BCC (79_{Hex})
Die BIS-Auswerteeinheit quittiert mit <ACK> '0'

Angaben in Hochkommata stellen das/die jeweilige/n Zeichen im ASCII-Code dar.

Programmierinformationen

Nächsten Datenträger suchen (einmal)

Mit dem nachfolgend angegebenen Telegramm wird der nächste Datenträger gesucht. Dabei wird zum nächstfolgenden Schreib-/Lesekopf weitergeschaltet und geprüft, ob sich ein Datenträger vor diesem Schreib-/Lesekopf befindet. Wenn ja, enthält die Telegrammrückmeldung die zugehörige Nummer des Schreib-/Lesekopfs und die ersten 4 Byte des Datenträgers. Wenn nein, wird der ursprüngliche Schreib-/Lesekopf wieder angewählt und geprüft. Wird auch hier kein Datenträger gefunden, dann lautet die Telegrammrückmeldung: 'H ? 0000 w'.

'H ?' erkennt jeden Datenträger, unabhängig von der eingestellten Blockgröße, vorausgesetzt, Schreib-/Lesekopf und Datenträger sind kompatibel.

Task	Datenfluss	Be- fehl	Ken- nung	Abschluss	Quittung	Endeken- nung 3)	Rück- meldung	Kopf- nummer	Daten vom Datenträger	Abschluss
Nächsten Datenträger suchen (einmal)	vom steuernden System zum BIS	'H'	'?'	BCC oder siehe 2)						
	vom BIS zum steuernden System				<ACK>'0'	'CR' oder 'LF CR'	'H'	'1', '2' oder '?'	D1 D2 D3 D4	BCC oder siehe 2)

- 1) Die Befehle Status und/oder Quitt sind an dieser Stelle nicht zugelassen.
- 2) Statt Blockcheck BCC kann je nach Protokollvariante entweder Carriage Return 'CR' oder Line Feed mit Carriage Return 'LF CR' verwendet werden.
- 3) Bei Protokollvarianten, die immer eine Endekennung benötigen, muss hier eines der Abschlusszeichen 'CR' oder 'LF CR' eingefügt werden.

Telegrammbeispiel:
**Nächsten
Datenträger suchen
(einmal)**
mit Blockcheck (BCC)

-> Kopf 1 ist angewählt. Es befindet sich nur vor Schreib-/Lesekopf 2 ein Datenträger, dessen erste vier Byte mit 9876 beschrieben sind.

Das Steuersystem sendet 'H ? w' BCC (77_{Hex})

Die BIS-Auswerteeinheit quittiert mit <ACK> '0 H 2 9 8 7 6 z' BCC (7A_{Hex})

Die Angaben in Hochkommata stellen das/die jeweilige/n Zeichen im ASCII-Code dar.

Programmierinformationen

Nächsten Datenträger suchen (ständig)

Mit dem nachfolgend angegebenen Telegramm wird der nächste Datenträger gesucht. Dabei wird zum nächstfolgenden Schreib-/Lesekopf weitergeschaltet und geprüft, ob sich ein Datenträger vor diesem Schreib-/Lesekopf befindet. Wenn ja, enthält die Telegrammrückmeldung die zugehörige Nummer des Schreib-/Lesekopfs und die ersten 4 Byte des Datenträgers. Wenn nein, wird zum angewählten Kopf zurückgeschaltet und geprüft, ob sich ein Datenträger vor diesem Schreib-/Lesekopf befindet. Dies wiederholt sich so lange, bis ein Datenträger erkannt wird.

'H !' erkennt jeden Datenträger, unabhängig von der eingestellten Blockgröße, vorausgesetzt, Schreib-/Lesekopf und Datenträger sind kompatibel.

Task	Datenfluss	Be- fehl	Ken- nung	Abschluss	Quittung	Ende- kennung 3)	Rück- meldung	Kopf- nummer	Daten vom Datenträger	Abschluss
Nächsten Datenträger suchen (ständig)	vom steuernden System zum BIS	'H'	'!'	BCC oder siehe 2)						
	vom BIS zum steuernden System				<ACK>'0'	'CR' oder 'LF CR'	'H'	'1' oder '2'	D1 D2 D3 D4	BCC oder siehe 2)

- 1) Die Befehle Status und/oder Quitt sind an dieser Stelle nicht zugelassen.
- 2) Statt Blockcheck BCC kann je nach Protokollvariante entweder Carriage Return 'CR' oder Line Feed mit Carriage Return 'LF CR' verwendet werden.
- 3) Bei Protokollvarianten, die immer eine Endekennung benötigen, muss hier eines der Abschlusszeichen 'CR' oder 'LF CR' eingefügt werden.

Telegrammbeispiel:
**Nächsten
Datenträger suchen
(ständig)**
mit Blockcheck (BCC)

-> Es befindet sich vor Schreib-/Lesekopf 2 ein Datenträger, dessen erste vier Byte mit 9876 beschrieben sind.

Das Steuersystem sendet 'H ! i' BCC (07_{Hex})

Die BIS-Auswerteeinheit quittiert mit <ACK> '0 H 2 9 8 7 6 z' BCC (7A_{Hex})

Die Angaben in Hochkommata stellen das/die jeweilige/n Zeichen im ASCII-Code dar.

Programmierinformationen

Bearbeiten der Ausgänge

Durch das Absenden des Telegramms können die vier Ausgänge gesetzt oder gelöscht werden.

Task	Datenfluss	Befehl	Kennung	Abschluss 2)	Quittung	Ende- kennung 3)
Ausgänge bearbeiten (setzen oder löschen)	vom steuernden System zum BIS	'B'	'00' bis 'A1' (siehe unten)	BCC oder siehe 2)		
	vom BIS zum steuernden System				<ACK>'0'	'CR' oder 'LF CR'
				1)		

Bedeutung der Kennung: Ausgang Nr. 0 1 2 3 alle Ausgänge

Ausgang löschen 00 10 20 30 A0

Ausgang setzen 01 11 21 31 A1

1) Die Befehle Status und/oder Quitt sind an dieser Stelle nicht zugelassen.

2) Statt Blockcheck BCC kann je nach Protokollvariante entweder Carriage Return 'CR' oder Line Feed mit Carriage Return 'LF CR' verwendet werden.

Telegrammbeispiel:
Bearbeiten der Ausgänge
mit Blockcheck (BCC)

Das Steuersystem sendet 'B 21 A' BCC (41_{Hex})
Die BIS-Auswerteeinheit quittiert mit <ACK> '0'

Nach Ablauf des Telegramms ist Ausgang 2 gesetzt.

Angaben in Hochkommata stellen das/die jeweilige/n Zeichen im ASCII-Code dar.

Programmierinformationen

Alle Ausgänge ausgeben

Durch das Absenden des Telegramms können die Zustände aller vier Ausgänge ausgegeben werden.

Task	Datenfluss	Befehl	Kennung	Abschluss 2)	Quittung	Zustand der vier Ausgänge	Ende- kennung 3)	Abschluss 2)
alle Ausgänge ausgeben	vom steuernden System zum BIS	'B'	'A0'	BCC oder siehe 2)				
	vom BIS zum steuernden System				<ACK>'0'	'XXXX' '0' = gelöscht, '1' = gesetzt	'CR' oder 'LF CR'	BCC oder siehe 2)
				1)				

Die Ausgabe des Zustands erfolgt in der Reihenfolge der Ausgänge-Nr. 0 1 2 3

1) Die Befehle Status und/oder Quitt sind an dieser Stelle nicht zugelassen.

2) Statt Blockcheck BCC kann je nach Protokollvariante entweder Carriage Return 'CR' oder Line Feed mit Carriage Return 'LF CR' verwendet werden.

Telegrammbeispiel:
Ausgeben aller Ausgänge
mit Blockcheck (BCC)

-> Die Ausgänge 0 und 1 sind gesetzt, die Ausgänge 2 und 3 gelöscht.

Das Steuersystem sendet 'BAO L' BCC (4C_{Hex})

Die BIS-Auswerteeinheit quittiert mit <ACK> '0 1100 NUL' BCC (00_{Hex})

Angaben in Hochkommata stellen das/die jeweilige/n Zeichen im ASCII-Code dar.

Programmierinformationen

Neustart der Auswerteeinheit (Quitt)

Durch das Absenden des Telegramms Neustart wird ein in Arbeit befindliches Telegramm abgebrochen und die Auswerteeinheit in den Grundzustand gebracht. Nach der Quittierung dieses Telegramms sind ca. 500 ms Pause vorzusehen, bevor ein neues Telegramm gestartet wird.

Wichtig! Der Befehl Quitt ist nicht zugelassen, während die Auswerteeinheit auf ein Abschlusszeichen wartet (BCC, 'CR' oder 'LF CR'). In dieser Situation würde Quitt als Abschluss- oder Nutzzeichen fehlinterpretiert.

Task	Datenfluss	Befehl	Abschluss 2)	Quittung	Abschluss 2)
Neustart (Quit)	vom steuernden System zum BIS	'Q'	BCC oder siehe 2)		
	vom BIS zum steuernden System			'Q'	BCC oder siehe 2)
1)					

- 1) Die Befehle Status und/oder Quitt sind an dieser Stelle nicht zugelassen.
- 2) Statt Blockcheck BCC kann je nach Protokollvariante entweder Carriage Return 'CR' oder Line Feed mit Carriage Return 'LF CR' verwendet werden.

Telegrammbeispiel: Neustart der Auswerteeinheit (Quitt) mit Blockcheck (BCC):

Das System BIS soll in den Grundzustand gebracht werden.
 Das Steuersystem sendet 'Q Q' BCC (51_{Hex})
 Die BIS-Auswerteeinheit quittiert mit 'Q Q' BCC (51_{Hex})

Angaben in Hochkommata stellen das/die jeweilige/n Zeichen im ASCII-Code dar.

Programmierinformationen

Abfrage der Statusmeldung

Mit dem Statustelegamm wird abgefragt, was für ein Telegramm sich in Arbeit befindet.

Wichtig! Der Befehl Status ist nicht zugelassen, während die Auswerteeinheit auf ein Abschlusszeichen wartet (BCC, 'CR' oder 'LF CR'). In dieser Situation würde Status als Abschluss- oder Nutzzeichen fehlinterpretiert.
 Die Angaben zwischen den Hochkommata stellen das/die jeweilige/n Zeichen im ASCII-Code dar.
 ' ' = Leertaste (Space SP) = ASCII-Zeichen (20_{Hex}).

Task	Datenfluss	Befehl	Abschluss 2)	Statusmeldung	Abschluss 2)
Abfrage der Statusmeldung	vom steuernden System zum BIS	'S'	BCC oder siehe 2)		
	vom BIS zum steuernden System			'S' ' ' 'R', 'W', 'L', 'P' oder 'H'	BCC oder siehe 2)
1)					

- 1) Die Befehle Status und/oder Quitt sind an dieser Stelle nicht zugelassen.
- 2) Statt Blockcheck BCC kann je nach Protokollvariante entweder Carriage Return 'CR' oder Line Feed mit Carriage Return 'LF CR' verwendet werden.

Statusmeldungen und ihre Bedeutung:

- 'S L' = Lesen vom Datenträger mit Anwahl des Schreib-/Lesekopfs und der Blockgröße des Datenträgers
- 'S P' = Schreiben auf den Datenträger mit Anwahl des Schreib-/Lesekopfs und der Blockgröße des Datenträgers
- 'S R' = Lesen vom Datenträger
- 'S W' = Schreiben auf den Datenträger
- 'S H' = Anwahl des Schreib-/Lesekopfs und der Blockgröße des Datenträgers
- 'S _' = kein Telegramm in Arbeit

Programmierinformationen

Telegrammbeispiele zu ¶ 50:

Abfrage der Statusmeldung mit Blockcheck (BCC)

-> Es soll der Status im BIS abgefragt werden, nachdem kurz zuvor ein **Lesetelegramm** abgesandt worden war.

Das Steuersystem sendet 'S S' BCC (53_{Hex})
Die BIS-Auswerteeinheit quittiert mit 'S L US' BCC (1F_{Hex})

-> Es soll der Status im BIS abgefragt werden, nachdem kurz zuvor ein **Schreibtelegramm** abgesandt worden war.

Das Steuersystem sendet 'S S' BCC (53_{Hex})
Die BIS-Auswerteeinheit quittiert mit 'S P ETX' BCC (03_{Hex})

-> Es soll der Status im BIS abgefragt werden, nachdem kurz zuvor ein **Telegramm zur Umschaltung des Schreib-/Lesekopfs** abgesandt worden war.

Das Steuersystem sendet 'S S' BCC (53_{Hex})
Die BIS-Auswerteeinheit quittiert mit 'S H ESC' BCC (1B_{Hex})

-> Es soll der Status im BIS abgefragt werden. Zuvor wurde **kein Telegramm** abgesandt.

Das Steuersystem sendet 'S S' BCC (53_{Hex})
Die BIS-Auswerteeinheit quittiert mit 'S _' BCC (20_{Hex})

Wichtig: Eine Statusabfrage während eines Lese- oder Schreibzugriffs auf einen Datenträger (LED Codetag Operating leuchtet) verlängert die Lese- oder Schreibzeit. Dies kann besonders bei dynamischen Betrieb dazu führen, dass die Zeit, in der sich der Datenträger im Arbeitsbereich des Schreib-/Lesekopfs befindet, zum vollständigen Lesen oder Schreiben nicht mehr ausreicht. Die dauernde Statusabfrage stört die Bearbeitung des Datenträgers; evtl. wird der Datenträger nicht erkannt!
Angaben in Hochkommata stellen das/die jeweilige/n Zeichen im ASCII-Code dar.

Programmierinformationen

CRC_16 Datenprüfung initialisieren

Mit diesem Telegramm wird ein Datenträger, der sich vor dem aktiven Schreib-/Lesekopf befindet, für die Verwendung bei CRC_16-Datenprüfung initialisiert. Dieses Telegramm muss auch dann erneut gesendet werden, wenn eine CRC-Fehler als Folge aus einem mißglückten Schreibauftrag aufgetreten ist, d.h. der Datenträger muss neu initialisiert werden, um ihn wieder verwenden zu können.

Bitte beachten Sie die Tabelle auf ¶ 16! Die angegebene Anzahl nutzbarer Byte darf nicht überschritten werden. D.h. die Summe aus Startadresse plus Anzahl Byte darf die nutzbare Datenträger-Kapazität nicht überschreiten!

Task	Datenfluss	Be- fehl	Startadresse des ersten zu übertra- genden Byte	Anzahl der zu über- tragenden Bytes	Kopf- nummer	Block- größe	Ab- schluss 2)	Quit- tung 3)	Ende- kenn- nung 4)	Start zur Über- tragung	Daten (von Startadresse bis Startadresse + Anzahl Byte)	Ab- schluss 2)	Quit- tung 3)	Ende- kenn- nung 4)
CRC_16 Bereich initialisieren	vom steuernden System zum BIS	Z'	A3 A2 A1 A0 '0 0 0 0' bis '8 1 9 1'	L3 L3 L1 L0 '0 0 0 1' bis '8 1 9 2'	K '1', '2', '3' oder '1'	B '0' oder '1'	BCC oder siehe 2)			<STX>	D1 D2 D3 ... Dn	BCC siehe 2)		
	vom BIS zum steuernden System								<ACK>- '0' oder <NAK> + F-Nr.	'CR' oder 'LF' CR'			<ACK>- '0' oder <NAK> + F-Nr.	'CR' oder 'LF' CR'

- 1) Die Befehle Status und/oder Quit sind an dieser Stelle nicht zugelassen.
- 2) Statt Blockcheck BCC kann je nach Protokollvariante entweder Carriage Return 'CR' oder Line Feed mit Carriage Return 'LF CR' verwendet werden.
- 3) Als Quittung kommt <ACK> '0', wenn kein Fehler aufgetreten ist, oder <NAK> + 'Fehlernr.', wenn ein Fehler aufgetreten ist.
- 4) Bei Protokollvarianten, die immer eine Endekennung benötigen, muss hier eines der Abschlusszeichen 'CR' oder 'LF CR' eingefügt werden.

Die Angaben zwischen den Hochkommata stellen das/die jeweilige/n Zeichen im ASCII-Code dar. '_' = Leertaste (Space) = ASCII-Zeichen 20 (Hex).

Fehlernummern

Fehlernummern

BIS C-6_0 gibt immer eine Fehlernummer aus. Deren Bedeutung zeigt nachfolgende Tabelle.

Nr.	Fehlerbeschreibung	Auswirkung
1	Kein Datenträger vorhanden	Telegrammabbruch, Auswerteeinheit geht in den Grundzustand.
2	Fehler beim Lesen	Lesetelegrammabbruch, Auswerteeinheit geht in den Grundzustand.
3	Lesen abgebrochen, da der Datenträger entfernt wurde.	Auswerteeinheit geht in den Grundzustand.
4	Fehler beim Schreiben	Schreibtelegrammabbruch, Auswerteeinheit geht in den Grundzustand.
5	Schreiben abgebrochen, da der Datenträger entfernt wurde.	Auswerteeinheit geht in den Grundzustand.
6	Fehler auf der Schnittstelle	Auswerteeinheit geht in den Grundzustand. (Paritäts- oder Stoppbitfehler)
7	Telegramm-Formatfehler	Auswerteeinheit geht in den Grundzustand. Mögliche Formatfehler: - Befehl ist kein 'L', 'P', 'C', 'R', 'W', 'H', 'B', 'Q', 'S' oder 'Z'. - Startadresse oder Anzahl Byte außerhalb des zugelassenen Bereichs

*) Hinweis: Wird mit CRC-Datencheck gearbeitet, kann beim nächsten Lesebefehl die Fehlermeldung E auftreten, wenn der Fehler 4 oder 5 nicht behoben wurde.

Fehlernummern

Fehlernummern (Fortsetzung)

Nr.	Fehlerbeschreibung	Auswirkung
8	BCC-Fehler, der übertragene BCC ist falsch.	Telegrammabbruch, Auswerteeinheit geht in den Grundzustand.
9	Kabelbruch zum angewählten Schreib-/Lesekopf oder nicht angeschlossen, LED Codetag Present blinkt.	Telegrammabbruch, Auswerteeinheit geht in den Grundzustand. Wurden beide Schreib-/Leseköpfe über den Befehl 'HT' angewählt, könnte ein Kopf nicht angeschlossen sein. Sind beide Schreib-/Leseköpfe angewählt, wird die Kabelbruch-meldung nur angezeigt, wenn sich kein Datenträger vor dem angeschlossenen, nicht defekten Kopf befindet.
A	Neuer Befehl nicht möglich, da bereits ein Lesebefehl in Arbeit ist.	Nach Fehlermeldung wird Lesebefehl intern beendet, aber nicht mehr quittiert. Auswerteeinheit geht in den Grundzustand.
B	Neuer Befehl nicht möglich, da bereits ein Schreibbefehl in Arbeit ist.	Nach Fehlermeldung wird Schreibbefehl intern beendet, aber nicht mehr quittiert. Auswerteeinheit geht in den Grundzustand.
C	Neuer Befehl nicht möglich, da bereits eine Kopfumschaltung in Arbeit ist.	Nach Fehlermeldung wird nicht mehr positiv quittiert, obwohl die Kopfumschaltung ausgeführt wurde. Auswerteeinheit geht in den Grundzustand.
E	CRC-Fehler, der CRC auf dem Datenträger ist falsch. *)	Telegrammabbruch, Auswerteeinheit geht in den Grundzustand.

*) Hinweis: Wird mit CRC-Datencheck gearbeitet, kann die Fehlermeldung E als Folge auftreten, wenn beim vorausgegangenen Befehl der Fehler 4, 5 oder B gemeldet wurde.

BIS C-600 Schnittstelleninformationen

LED-Anzeige: System Ready Codetag Present Codetag Operating

Die Auswerteeinheit BIS C-6_0 meldet die wichtigsten Betriebszustände über drei LED's auf der Frontplatte.

Betriebszustand	LED	Bedeutung
System Ready	an (grün) aus	Betriebsspannung in Ordnung; kein Hardwarefehler Betriebsspannung oder Hardware nicht in Ordnung
Codetag Present	an (gelb) blinkt aus	Datenträger schreib-/lesebereit. (Tritt während des Schreibens/ Lesens ein Schreib-/Lesefehler auf, erlischt System Ready, wenn Protokollvariante "ohne Fehlernummer" eingestellt ist!) Kabelbruch zum angewählten Schreib-/Lesekopf oder nicht ange- schlossen. Wurden beide Schreib-/Leseköpfe über den Befehl 'HT' angewählt, könnte ein Kopf nicht angeschlossen sein. Sind beide Schreib-/Leseköpfe angewählt, wird die Kabelbruch- meldung nur angezeigt, wenn sich vor dem angeschlossenen, nicht defekten Kopf kein Datenträger befindet. Kein Datenträger im Schreib-/Lesebereich
Codetag Operating	an (gelb) aus	Befehl wird bearbeitet Kein Befehl in Arbeit

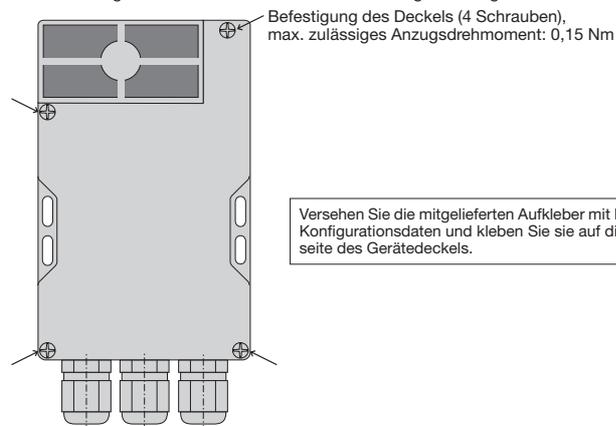
Wenn alle drei LEDs gleichzeitig synchron blinken, muss die Auswerteeinheit zur Reparatur ins Werk.

BIS C-600 Schnittstelleninformationen

Öffnen der Auswerteeinheit BIS C-600

Um die Verbindungen herzustellen, ist die Auswerteeinheit BIS C-600 zu öffnen.

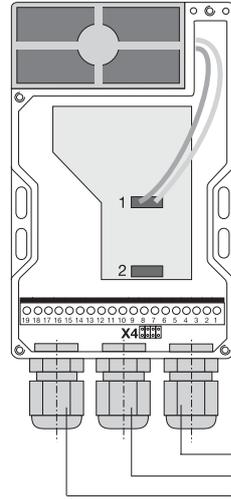
Sorgen Sie dafür, dass das Gerät spannungsfrei geschaltet ist. Öffnen Sie die 4 Schrauben am BIS C-600 und entfernen Sie den Deckel. Führen Sie die Anschlusskabel durch die Klemmverschraubungen. Weitere Einzelheiten zur Verdrahtung siehe folgende .



BIS C-600 Schnittstelleninformationen

**Anschlussplan für
Auswerteeinheiten
BIS C-600**

Auswerteeinheit geöffnet



Belegung der Klemmleiste

19	18	17	16	15	14
+ VS	- VS	$\frac{1}{2}$	TxD	RxD	COM
POWER			Service		

13	12	11	10	9	8	7	6
+ VS	- VS	1	2	3	4	+IN	-IN
OUTPUT					INPUT		

5	4	3	2	1
COM	T (A)	T (B)	R (A)	R (B)
RS 422 (4-Draht)				

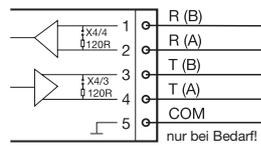
T = Transmit
R = Receive

5	4	3	2	1
COM	R/T (A)	R/T (B)		
RS 485 (2-Draht)				

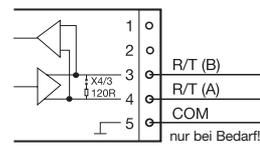
BIS C-600 Schnittstelleninformationen

**Verbindungskabel
und Schnittstelle für
RS 422/RS 485**

Die Auswerteeinheit BIS C-600-007...02 benutzt eine serielle Schnittstelle für die Kommunikation mit dem steuernden System (SPS oder PC). Abhängig von der Konfiguration kann als Schnittstelle entweder RS 422 (4-Draht) oder RS 485 (2-Draht) benutzt werden.



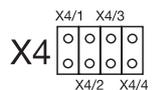
Anschluss an der
Klemmleiste für die
Schnittstelle RS 422
(4-Draht)



Anschluss an der
Klemmleiste für die
Schnittstelle RS 485
(2-Draht)

RS 422: Shunt-Stecker X 4/1
muss gesteckt sein!

RS 485: Shunt-Stecker X 4/2
muss gesteckt sein!



Brückenstecker für
Abschlusswiderstand

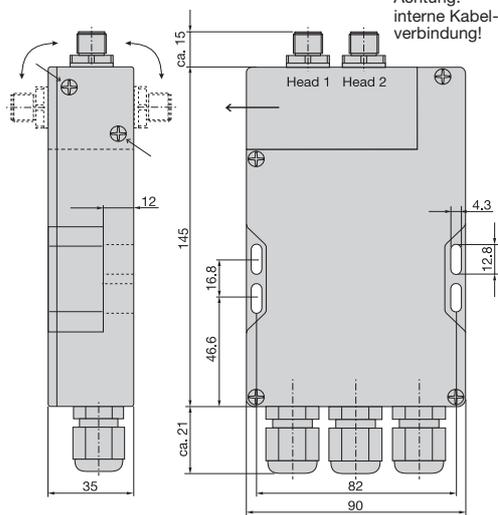
Abschlusswiderstand	Brücken bei X 4/3	X 4/4
passiv	-	-
aktiv	gesteckt	gesteckt

BIS C-600 Montage Auswerteeinheit / Kopf

Montage der Auswerteeinheit BIS C-600 und Anordnung des Schreib-/Lesekopfes bzw. des Adapters BIS C-650

Die Auswerteeinheit wird an den 4 seitlichen Langlöchern befestigt.

Je nach Ausführung ist die Auswerteeinheit mit einem Schreib-/Lesekopf oder dem Adapter für abgesetzte Schreib-/Leseköpfe ausgestattet. Sowohl der Schreib-/Lesekopf als auch der Adapter können vom Anwender durch Umsetzen um + oder -90° in die gewünschte Lage gebracht werden (siehe Bild). Sorgen Sie dafür, dass das Gerät spannungsfrei geschaltet ist. Öffnen Sie die beiden Schrauben (im Bild durch Pfeile gekennzeichnet). Ziehen Sie den Kopf bzw. den Adapter vorsichtig nach der Seite heraus (Pfeilrichtung, rechtes Bild). **Achtung: interne Kabelverbindung!** Montieren Sie ihn in der gewünschten Lage und schrauben Sie ihn wieder an.



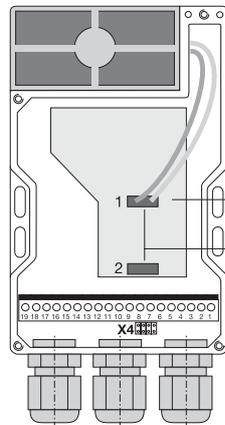
Achtung:
interne Kabelverbindung!

BIS C-600 Montage Auswerteeinheit / Kopf

Montage eines Schreib-/Lesekopfes bzw. Adapters BIS C-650

Wenn Sie einen Schreib-/Lesekopf wechseln wollen: Auswerteeinheit spannungsfrei schalten und öffnen. Befestigungsschrauben des Schreib-/Lesekopfes lösen (siehe ¶ 59) und Deckel der Auswerteeinheit abschrauben (siehe ¶ 56). Lösen Sie die Steckverbindung des Schreib-/Lesekopfes von der Platine und ziehen Sie das Verbindungskabel durch den Kabelschacht heraus. Für die Montage des neuen Kopfes verfahren Sie in umgekehrter Reihenfolge.

Wenn Sie einen Adapter BIS C-650 montieren wollen, verfahren Sie wie oben beschrieben. Es müssen beide Verbindungskabel auf der Platine gesteckt werden.



Anschluss des integrierten Schreib-/Lesekopfes

Anschlüsse für den Adapter BIS C-650
1 = Head 1
2 = Head 2

Befestigung des Deckels (4 Schrauben),
max. zulässiges Anzugsdrehmoment: 0,15 Nm

61

BIS C-600 Technische Daten

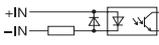
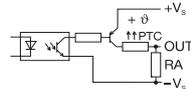
Abmessungen, Gewicht	Gehäuse	Kunststoff PS
	Abmessungen mit Schreib-/Lesekopf BIS C-652	167 x 90 x 35 mm
	Abmessungen mit Adapter BIS C-650	145 x 90 x 35 mm
	Gewicht	500 g
Temperaturbereiche	Umgebungstemperatur	0 °C bis + 60 °C
Anschlüsse	Klemmleiste	19-polig
	Kabeleinführung	3 x Klemmkorb PG 9
	Kabeldurchmesser	4 bis 8 mm
	Leitergrößen	0,14 bis 1 mm ²
	mit Adernhülsen	0,25 bis 0,34 mm ²
Schutzart	Schutzart	IP 65
Elektrische Anschlüsse	Betriebsspannung V_s, Eingang	24 V DC ± 20 %
	Restwelligkeit	≤ 10 %
	Stromaufnahme	≤ 400 mA
	Serielle Schnittstelle	RS 422/RS 485
	Schreib-/Lesekopf alternativ bei montiertem Adapter BIS C-650 *)	integriert, BIS C-65_ und folgende *); 2 x Einbaustecker 4-polig (Stift) für alle Schreib-/Leseköpfe BIS C-3_ _ mit 4-poligem Stecker (Buchse), nicht BIS C-350 und BIS C-352

*) um ± 90° umsetzbar

deutsch BALLUFF 61

62

BIS C-600 Technische Daten

Elektrische Anschlüsse	Digitaler Eingang (+IN, -IN)	über Optokoppler galvanisch getrennt	
	Steuerspannung aktiv	4 V bis 40 V	
	Steuerspannung inaktiv	1,5 V bis -40 V	
Eingangsstrom bei 24 V	11 mA		
	Verzögerungszeit typisch	5 ms	
Funktionsanzeigen	Steuerausgänge (01 bis 04)	über Optokoppler galvanisch getrennt	
	Ausgangsstrom	max. 20 mA	
	Spannungsabfall bei 20 mA	ca. 2,5 V	
	Ausgangswiderstand R _A	10 kΩ gegen -V _s	
	Ausgangsschaltung	PNP (plusschaltend)	
	Betriebsspannung, Ausgang V _s	24 V DC ± 20 %	
Restwelligkeit	≤ 10 %		
	Serielle Schnittstelle	RS 422 (4-Draht) / RS 485 (2-Draht)	über Optokoppler galvanisch getrennt
	System Ready	LED grün	
	Codetag Present	LED gelb	
	Codetag Operating	LED gelb	

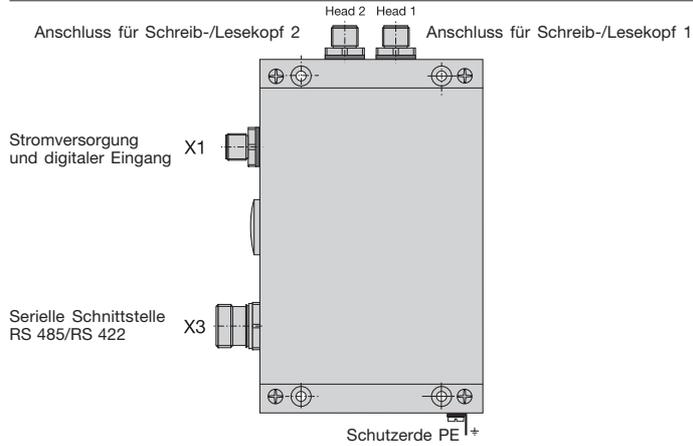


Mit dem CE-Zeichen bestätigen wir, dass unsere Produkte den Anforderungen der EG-Richtlinie 89/336/EWG (EMV-Richtlinie)

und des EMV-Gesetzes entsprechen. In unserem EMV-Labor, das von der DATech für Prüfungen der elektromagnetischen Verträglichkeit akkreditiert ist, wurde der Nachweis erbracht, dass die Balluff-Produkte die EMV-Anforderungen der Fachgrundnorm EN 61000-6-4 (Emission), EN 61000-6-2 (Störfestigkeit) erfüllen.

BIS C-620
Schnittstelleninformationen

Anschlussplan für Auswerteeinheit BIS C-620



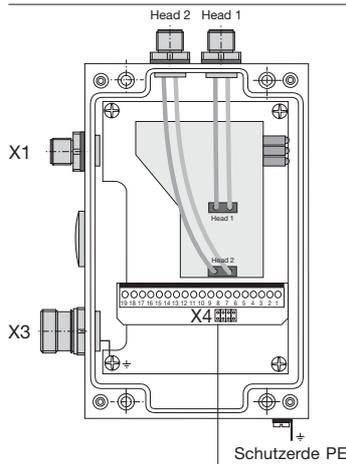
Lage und Bezeichnung der Anschlüsse

Öffnen der Auswerteeinheit BIS C-620

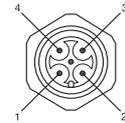
Um den Shunt-Stecker für die Umstellung von RS 422 auf RS 485 zu stecken, ist die Auswerteeinheit BIS C-620 zu öffnen. Sorgen Sie dafür, dass das Gerät spannungsfrei geschaltet ist. Öffnen Sie die 4 Schrauben am BIS C-620 und entfernen Sie den Deckel. Weitere Einzelheiten siehe folgende

BIS C-620
Schnittstelleninformationen

Anschlussplan für Auswerteeinheit BIS C-620

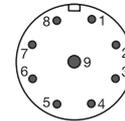


X1, Stromversorgung und digitaler Eingang



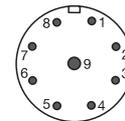
Pin	Funktion
1	+Vs
3	-Vs
2	-IN
4	+N
5	n.c.

X3, RS 485



Pin	Funktion
1	n.c.
2	n.c.
3	R/T (A)
4	n.c.
5	COM
6	n.c.
7	R/T (B)
8	n.c.
9	n.c.

X3, RS 422



Pin	Funktion
1	n.c.
2	R (A)
3	T (A)
4	n.c.
5	COM
6	n.c.
7	T (B)
8	R (B)
9	n.c.

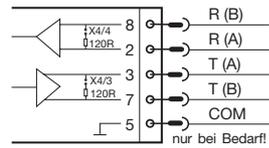
Shunt-Stecker für Umstellung von RS 422 auf RS 485 (siehe folgende

n.c. = nicht anschließen

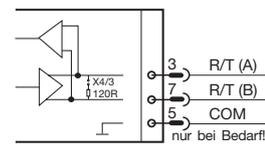
BIS C-620 Schnittstelleninformationen

Verbindungskabel und Schnittstelle für RS 422/RS 485

Die Auswertereinheit BIS C-620-007...02 benutzt eine serielle Schnittstelle für die Kommunikation mit dem steuernden System (SPS oder PC). Abhängig von der Konfiguration kann als Schnittstelle entweder RS 422 (4-Draht) oder RS 485 (2-Draht) benutzt werden.



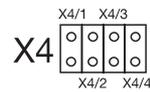
Anschluss an der
Klemmleiste für die
Schnittstelle RS 422
(4-Draht)



Anschluss an der
Klemmleiste für die
Schnittstelle RS 485
(2-Draht)

RS 422: Shunt-Stecker X 4/1
muss gesteckt sein!

RS 485: Shunt-Stecker X 4/2
muss gesteckt sein!



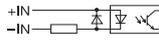
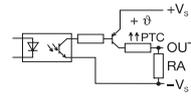
Brückenstecker für
Abschlusswiderstand

Abschluss- widerstand	Brücken bei X 4/3	X 4/4
passiv	-	-
aktiv	gesteckt	gesteckt

BIS C-620 Technische Daten

Abmessungen, Gewicht	Gehäuse	Metall
	Abmessungen Gewicht	175 x 120 x 60 mm 820 g
Temperaturbereich	Umgebungstemperatur	0 °C bis +60 °C
Anschlüsse	Einbaustecker X1	5-polig (Stift)
	Einbaustecker Head 1, Head 2	4-polig (Stift)
	Rundsteckverbinder X3	9-polig (Stift)
Schutzart	Schutzart	IP 65 (in angeschlossenem Zustand)
Elektrische Anschlüsse	Eingang X1, Betriebsspannung V_s	DC 24 V ± 20 %
	Restwelligkeit	≤ 10 %
	Stromaufnahme	≤ 400 mA
	Anschluss X3, serielle Schnittstelle	RS 422 (4-Draht) / RS 485 (2-Draht)
Anschlüsse Head 1, Head 2, Schreib-/Lesekopf	2 x Einbaustecker 4-polig (Stift) für alle Schreib-/Leseköpfe BIS C-3_ _ mit 4-poligem Stecker (Buchse), nicht BIS C-350 und BIS C-352	

BIS C-620 Technische Daten

Elektrische Anschlüsse	Digitaler Eingang (+IN, -IN)	über Optokoppler galvanisch getrennt 4 V bis 40 V Steuerspannung aktiv 1,5 V bis -40 V Steuerspannung inaktiv 11 mA Eingangsstrom bei 24 V 5 ms Verzögerungszeit typisch	
	Steuerausgänge (01 bis 04)	über Optokoppler galvanisch getrennt max. 20 mA Spannungsabfall bei 20 mA ca. 2,5 V Ausgangswiderstand R_A 10 k Ω gegen $-V_S$ Ausgangsschaltung PNP (plusschaltend) Betriebsspannung, Ausgang V_S 24 V DC \pm 20 % Restwelligkeit \leq 10 %	
Funktionsanzeigen	System Ready	LED grün	
	Codetag Present	LED gelb	
	Codetag Operating	LED gelb	



Mit dem CE-Zeichen bestätigen wir, dass unsere Produkte den Anforderungen der EG-Richtlinie

89/336/EWG (EMV-Richtlinie)

und des EMV-Gesetzes entsprechen. In unserem EMV-Labor, das von der DATech für Prüfungen der elektromagnetischen Verträglichkeit akkreditiert ist, wurde der Nachweis erbracht, dass die Balluff-Produkte die EMV-Anforderungen der Fachgrundnorm

EN 61000-6-4 (Emission), EN 61000-6-2 (Störfestigkeit) erfüllen.

BIS C-620 Bestellinformationen

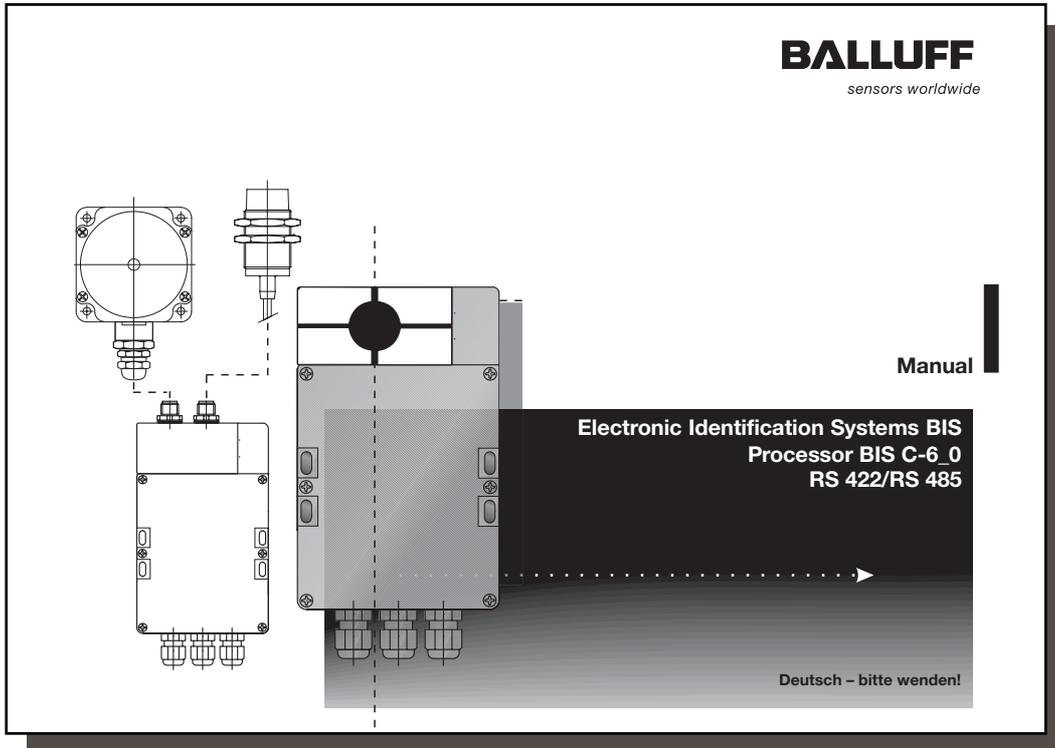
Typenschlüssel	BIS C-620-007-050-02-ST2
Balluff Identifikations-System	BIS
Baureihe C	C
Hardware-Typ	620
620 = Metallgehäuse	
Software-Typ	007
007 = Balluff-Protokoll	
Schreib-/Lesekopf	050
050 = mit zwei Anschlusssteckern für externe Schreib-/Leseköpfe BIS C-3_ _ (ausgenommen BIS C-350 und -352)	
Schnittstelle	02
02 = RS 422 / RS 485	
Kundenanschluss	ST2
ST2 = Steckervariante	
1 Rundsteckverbinder für Stromversorgung,	
1 Rundsteckverbinder für RS 422 / RS 485	

Zubehör	Artikel	Bestellbezeichnung
(optional, nicht im Lieferumfang)	Gegenstecker zu X1	BKS-S 79-00
	zu X3	BKS-S 84-00
	Verschlusskappe für Head 1, Head 2	BES 12-SM-2

Anhang, ASCII-Tabelle

Deci- mal	Hex	Control Code	ASCII	Deci- mal	Hex	Control Code	ASCII	Deci- mal	Hex	ASCII									
0	00	Ctrl @	NUL	22	16	Ctrl V	SYN	44	2C	,	65	41	A	86	56	V	107	6B	k
1	01	Ctrl A	SOH	23	17	Ctrl W	ETB	45	2D	-	66	42	B	87	57	W	108	6C	l
2	02	Ctrl B	STX	24	18	Ctrl X	CAN	46	2E	.	67	43	C	88	58	X	109	6D	m
3	03	Ctrl C	ETX	25	19	Ctrl Y	EM	47	2F	/	68	44	D	89	59	Y	110	6E	n
4	04	Ctrl D	EOT	26	1A	Ctrl Z	SUB	48	30	0	69	45	E	90	5A	Z	111	6F	o
5	05	Ctrl E	ENQ	27	1B	Ctrl [ESC	49	31	1	70	46	F	91	5B	[112	70	p
6	06	Ctrl F	ACK	28	1C	Ctrl \	FS	50	32	2	71	47	G	92	5C	\	113	71	q
7	07	Ctrl G	BEL	29	1D	Ctrl]	GS	51	33	3	72	48	H	93	5D]	114	72	r
8	08	Ctrl H	BS	30	1E	Ctrl ^	RS	52	34	4	73	49	I	94	5E	^	115	73	s
9	09	Ctrl I	HT	31	1F	Ctrl _	US	53	35	5	74	4A	J	95	5F	_	116	74	t
10	0A	Ctrl J	LF	32	20		SP	54	36	6	75	4B	K	96	60	`	117	75	u
11	0B	Ctrl K	VT	33	21		!	55	37	7	76	4C	L	97	61	a	118	76	v
12	0C	Ctrl L	FF	34	22		*	56	38	8	77	4D	M	98	62	b	119	77	w
13	0D	Ctrl M	CR	35	23		#	57	39	9	78	4E	N	99	63	c	120	78	x
14	0E	Ctrl N	SO	36	24		\$	58	3A	:	79	4F	O	100	64	d	121	79	y
15	0F	Ctrl O	SI	37	25		%	59	3B	;	80	50	P	101	65	e	122	7A	z
16	10	Ctrl P	DLE	38	26		&	60	3C	<	81	51	Q	102	66	f	123	7B	{
17	11	Ctrl Q	DC1	39	27		'	61	3D	=	82	52	R	103	67	g	124	7C	
18	12	Ctrl R	DC2	40	28		(62	3E	>	83	53	S	104	68	h	125	7D	}
19	13	Ctrl S	DC3	41	29)	63	3F	?	84	54	T	105	69	i	126	7E	~
20	14	Ctrl T	DC4	42	2A		*	64	40	@	85	55	U	106	6A	j	127	7F	DEL
21	15	Ctrl U	NAK	43	2B		+												

1



2

No. 854 487 D/E • Edition 1012
Subject to modification.
Replaces edition 0704.

Writing convention:

Control characters to be transmitted are in angle brackets.

Characters to be transmitted in ASCII code are enclosed in apostrophes.

Example: <STX> '1 2 3 4 5 6' BCC

This manual is valid from hardware version HV 2.0 or higher

Balluff GmbH
Schurwaldstrasse 9
73765 Neuhausen a.d.F.
Germany
Phone +49 7158 173-0
Fax +49 7158 5010
balluff@balluff.de

■ www.balluff.com

Contents

Safety Considerations	4
Introduction BIS C	5-7
Application BIS C	8
Protocol	9-11
Configuration	12-27
Customer Configuration	28-31
Read/Write Times	32/33
Programming Information	34-52
Error Numbers	53/54
BIS C-600: Interface Information	55-58
Processor / Head Installation.....	59-60
Technical Data	61/62
Ordering Information	63
BIS C-620: Processor Installation	64
Interface Information	65-67
Technical Data	68/69
Ordering Information	70
Appendix: ASCII Table	71

Safety Considerations

BIS C-6_0 processors together with the other BIS C system components comprise the Identification System and may be used only for this purpose.

Installation and Operation

Installation and operation should be carried out by technical trained personnel only. Unauthorized access and improper use will lead to loss of warranty and liability claims.

When installing the processor, consult the section on wiring diagrams carefully. Special caution must be used when wiring the processor to external controllers, particularly with respect to selection and polarity of the signals and power supply.

Only approved power supplies may be used with the processor. See the section on Technical Data for details.

Use and Checking

The relevant safety procedures must be followed when using the Identification System. In particular, steps must be taken to ensure that no danger to persons or equipment can arise should a fault occur in the Identification System.

This includes maintaining the published ambient operating conditions and regular checking of the functionality of the Identification System with all its associated components.

Fault Conditions

As soon as there is evidence that the Identification System is not functioning properly, it should be taken out of service and protected against unauthorized use.

Scope

This description is valid for processors in the series BIS C-600-007...02-KL1 and BIS C-620-007-050-02-ST2.

Introduction BIS C Identification System

This manual is designed to assist the user in setting up the control program and installing and starting up the components of the BIS C-6_0 Identification System, and to assure rapid, trouble-free operation.

Principles

The BIS C-6_0 Identification System belongs in the category of **non-contact systems for reading and writing.**

This dual function permits applications for not only transporting information in fixed-programmed data carriers, but also for gathering and passing along up-to-date information as well.

Applications

Some of the notable areas of application include

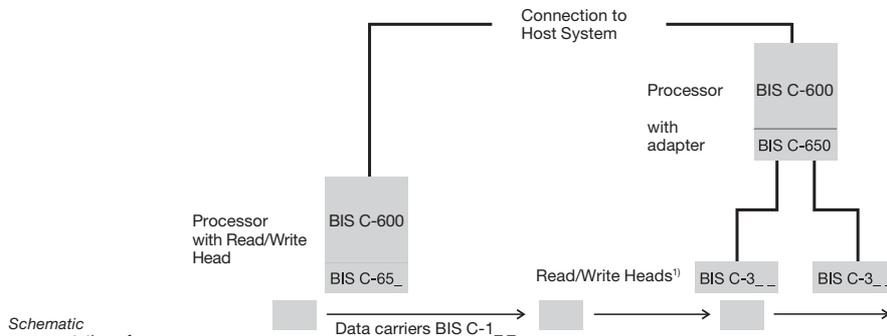
- **for controlling material flow in production processes**
(e.g. in model-specific processes),
for workpiece conveying in transfer lines,
in data gathering for quality assurance,
for gathering safety-related data,
- **in tool coding and monitoring;**
- **in equipment organization;**
- **in storage systems for monitoring inventory movement;**
- **in transporting and conveying systems;**
- **in waste management for quantity-based fee assessment.**

Introduction BIS C-600 Identification System

System Components

The main components of the BIS C-600 Identification System are:

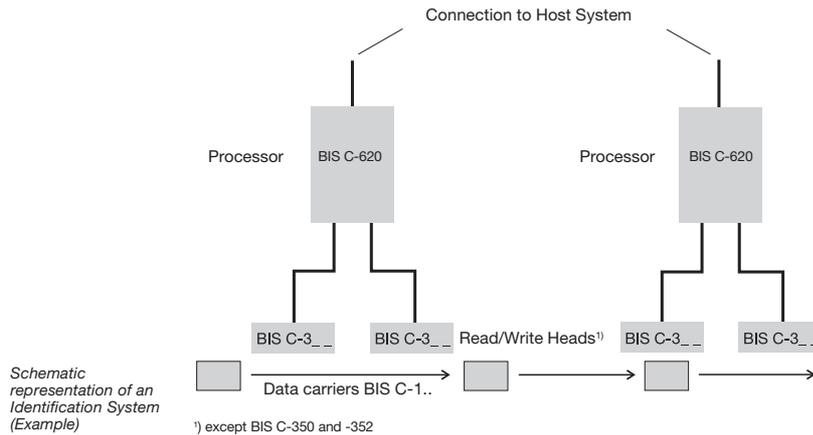
- **Processor,**
- **Read/Write Heads and**
- **Data carriers.**



Introduction BIS C-620 Identification System

System Components The main components of the BIS C-620 Identification System are:

- Processor,
- Read/Write Heads and
- Data carriers.



Schematic representation of an Identification System (Example)

Application BIS C-6_0 Processor

Selecting System Components

The **BIS C-600** processor has a plastic housing. Connection is made through a terminal strip, with the cable secured using a PG fitting. A single read/write head from series BIS C-65_ can be directly mounted to the processor, which creates a compact unit. If the BIS C-650 adapter is attached instead of the BIS C-65_ read/write head, two read/write heads may be cable connected. If the BIS C-670 adapter is used, a read/write head can be connected via cable.

The **BIS C-620** processor has a metal housing. Connection is made through round connectors. Two read/write heads can be connected via cable to the BIS C-620.

Whether the compact processor with integrated read/write head or the cable-connected solution is appropriate depends on the spatial arrangement of the components. There are no functional limitations. All read/write heads are suitable for both static and dynamic reading and writing. Sensing distance and traverse velocity will depend on the individual data carrier(s). For additional information about the series BIS C-65_ or BIS C-3_.. read/write heads, please refer to the respective user guides.

The system components are electrically powered by the processor. The data carrier represents a stand-alone unit, and does not require an external source of power. It receives its energy from the read/write head. The latter sends a constant carrier signal which supplies the data carrier as soon as the required sensing distance is reached. The read/write process takes place during this phase. Reading and writing may be static or dynamic.

Protocol

Host Function

The BIS C-600 processor is controlled by a host device:

- by a controlling computer (e.g. industrial PC) or
- by an external programmable logic controller (PLC)

Basic Protocol Description with Block Diagrams

The processor controls and manages the data communications between data carriers and read/write heads. Through the 4-wire RS 422 interface, it links the BIS C-600 Identification System to an external controlling device. If necessary, the processor can be reconfigured to a 2-wire RS 485 interface.

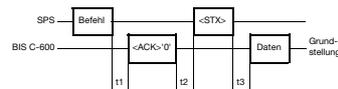
Data exchange between the processor and the host system is done using particular telegrams. The protocol is described in this section in the form of function blocks. Telegram contents are described more precisely in the section "Programming".

The following descriptions are valid only if:
 - The processor is in the base state.
 - A data carrier is present in front of a read/write head.

Protocol

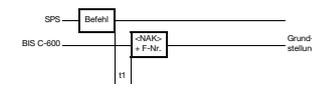
Without Head Select

Read: a) If no error:



t1 depending on no. of bytes to read (see ¶ 32/33)
 t2 ≥ 0 (is not monitored by the processor)
 t3 = max. 50 ms

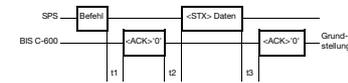
b) With error:



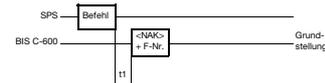
t1 depending on no. of bytes to read (see ¶ 32/33) and error type (recommended monitor time: 15 s)

Write: a) If no error:

t1 = max. 50 ms
 t2 ≥ 0 (is not monitored by the processor)
 t3 depending on no. of bytes to write (see ¶ 32/33)

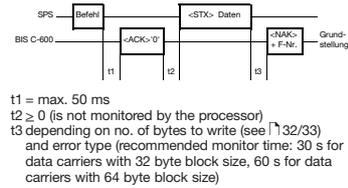


b) With error in command:



t1 = max. 50 ms

c) With error in writing:



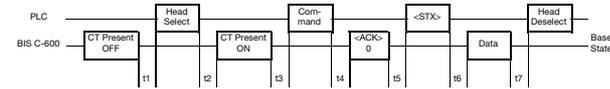
t1 = max. 50 ms
 t2 ≥ 0 (is not monitored by the processor)
 t3 depending on no. of bytes to write (see ¶ 32/33) and error type (recommended monitor time: 30 s for data carriers with 32 byte block size, 60 s for data carriers with 64 byte block size)

Protocol

With Head Select

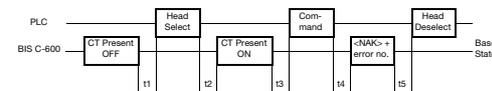
Read:

a) If no error:



- t1, t3, t7 ≥ 0
- t2 = max. 500 ms
- t4 depending on no. of bytes to read (see ¶ 32/33)
- t5 ≥ 0 (is not monitored by the processor)
- t6 = max. 50 ms

b) With error:



- t1, t3, t5 ≥ 0
- t2 = max. 500 ms
- t4 depending on no. of bytes to read (see ¶ 32/33) and error type (recommended monitor time: 15 s)

Configuration

Before programming, the processor configuration must be carried out, in case the factory settings will not be used.

Configuration is done using a computer and the Balluff BISC600C.EXE software, and it is stored in the processor. It may be overwritten at any time. The configuration can be stored in a file, making it accessible when required.

Files Online Configuration Help

BALLUFF

Identification systems

BIS C-600
with
RS422 or RS485 interface
point to point interface

Copyright (c) 1997, Balluff GmbH

<F1=Help> <F10=Menu> <Alt+=Command>

Important.

Please note the selected settings on the stick-on label supplied (to be pasted on the inside of the processor cover) as well as on ¶ 28 and 30 in the customer configuration section, so that in case of repair of the processor the settings can be saved or otherwise can also be used to set other processor units.

Please note.

Please note the selected settings on the stick-on label supplied (to be pasted on the inside of the processor cover) as well as on ¶ 28 and 30 in the customer configuration section, so that in case of repair of the processor the settings can be saved or otherwise can also be used to set other processor units.

Configuration

Interface

The first screen shows the parameters baud rate, number of data and stop bits, and parity type for the serial interface selected. The graphic shows the factory settings. The other settings are carried out in the corresponding masks which are illustrated in the following pages.

INTERFACE BIS C-6_0

<p style="text-align: center; margin: 0;">baud rate</p> <input type="checkbox"/> 600 baud <input type="checkbox"/> 1200 baud <input type="checkbox"/> 2400 baud <input type="checkbox"/> 4800 baud <input checked="" type="checkbox"/> 9600 baud <input type="checkbox"/> 19200 baud	<p style="text-align: center; margin: 0;">data bit</p> <input type="checkbox"/> 7 <input checked="" type="checkbox"/> 8	<p style="text-align: center; margin: 0;">parity</p> <input type="checkbox"/> odd <input checked="" type="checkbox"/> even <input type="checkbox"/> none
<p style="text-align: center; margin: 0;">stopbit</p> <input checked="" type="checkbox"/> 1 <input type="checkbox"/> 2		

Shortform of initialization

If the initializing data are available in short form (e.g. on the processor cover after a replacement of the unit) the data can be entered directly into the "Shortform of initialization BIS C-6_0" mask (see also Customer configuration on p. 28).

SHORTFORM OF INITIALIZATION BIS C-6_0

```

[ 09600 ] [ 8 ] [ 1 ] [ E ]
[ 1 ] [ X ] [ X ] [ 1 ] [ 1 ]
[ 0000000 ]
                
```

Do not change the settings marked here by crosses!

Configuration

Parameters

PARAMETERS BIS C-6_0

<p style="text-align: center; margin: 0;">Protocol type</p> <input checked="" type="checkbox"/> BCC <input type="checkbox"/> CR as terminator <input type="checkbox"/> CR <input type="checkbox"/> LFCR as terminator	<p style="text-align: center; margin: 0;">Paging</p> <input checked="" type="checkbox"/> 32 Byte <input type="checkbox"/> 64 Byte
<p style="text-align: center; margin: 0;">Parameters</p> <input type="checkbox"/> RS 422 [] / RS 485 [X] <input type="checkbox"/> CT-Present with an ' * ' <input type="checkbox"/> Dynamic mode <input type="checkbox"/> CT-Present on output 1 <input type="checkbox"/> Process outputs if CT-Present <input type="checkbox"/> Fast data carrier recognition <input type="checkbox"/> CRC_16 data checking	<p style="text-align: center; margin: 0;">Input</p> <input checked="" type="checkbox"/> Reset <input type="checkbox"/> Head select <input type="checkbox"/> Data bit on Data carrier <input type="checkbox"/> Not used

Protocol Type

Operation with blockcheck BCC is factory set. For host devices which require a terminator, the additional use of Carriage Return 'CR' or Line Feed with Carriage Return 'LF CR' is made available. The following ¶ contains examples of the various possibilities.

Configuration

Protocol Type (continued)

Protocol Variants	Telegram with command, Address and no. of bytes	Terminator	Acknowledge	Terminator
with Blockcheck BCC	'R 0000 0001'	BCC	<ACK> '0'	
with Carriage Return	'R 0000 0001'	'CR'	<ACK> '0'	
with Terminator Carriage Return	'R 0000 0001'	'CR'	<ACK> '0'	'CR'
with Terminator Carriage Return and Line Feed	'R 0000 0001'	'LF CR'	<ACK> '0'	'LF CR'

Examples for terminating telegrams:

Parameters

- RS 485, 2-wire

The unit is factory set for the 4-wire RS 422 interface. This parameter is used to reconfigure the unit for the 2-wire RS 485 interface.

Please note when using RS 485 that a wait time of typ. 10 ms between the receipt of data and sending of new data must be programmed.

- CT-Present-Message with '*'

For CT-Present message with '*', the ASCII character '*' (2A_{hex}) appears on the interface each time a data carrier is recognized. If using 'CR' as terminator, then '* CR' is output, or '* LF CR' if using 'LF CR' as terminator. In contrast to the CT-Present signal the removal of the data carrier will not be noticeable.

- Dynamic Mode

This function switches off the error-message "No data carrier present", i.e.:

- > In dynamic mode, a read or write telegram is stored until a data carrier enters the working range of the corresponding read/write head.
- > Without dynamic mode, a read or write telegram is acknowledged with an error message (<NAK> '1') if there is no data carrier present in front of a read/write head; the processor goes into the ground state.

english

BALLUFF | 15

Configuration

Parameters (continued)

- CT Present on Output 1

If CT Present is selected for Output 1, the LED message Codetag Present is also output on Output 1. In this way the presence of a data carrier can be directly verified as a digital hardware signal.

- Process Outputs if CT Present

The output functions are normally processed only after a read command. But since data carrier recognition is also an automatic data carrier read (reads first page, either 32 or 64 bytes depending on type), the output processing can occur simultaneously with Codetag Present. If the addresses for output processing are located on the first page, then the processor can itself carry out short control commands without a separate command.

-> For very fast transactions, see next section.

- Fast Data Carrier Recognition

For very fast transactions, the number of data carrier addresses used for data carrier recognition can be reduced from 32 or 64 bytes to 4 bytes. The data carrier recognition time is thereby reduced to ca. 50 ms (instead of approx. 150 ms for data carriers with < 2 kBytes or approx. 250 ms for data carriers with ≥ 2 kBytes of memory).

-> Please note this when using the parameter "Process outputs with CT-present".

Configuration

Parameters (continued)

– CRC_16 data checking

If CRC_16 data checking is activated, a special error message is output to the interface whenever a CRC error is detected.

If the error message is not caused by a failed write request, it may be assumed that one or more memory cells on the data carrier is defective. That data carrier must then be replaced.

If the CRC error is however due to a failed write request, you must reinitialize the data carrier in order to continue using it.

The checksum is written to the data carrier as a 2-byte wide datum. Two bytes per page are 'lost', i.e., the page size becomes 30 bytes or 62 bytes depending on data carrier type (see next [7]).

This means that the actual usable number of bytes is reduced!

Data carrier type	Usable bytes
128 bytes	= 120 bytes
256 bytes	= 240 bytes
511 bytes *)	= 450 bytes
1023 bytes *)	= 930 bytes
2047 bytes *)	= 1922 bytes
2048 bytes	= 1984 bytes
8192 bytes	= 7936 bytes

*) The last data carrier page for these EEPROM-based data carriers is not fully available.

Configuration

Paging

In the memory organization of the data carrier, a distinction is made between page size 32 bytes or 64 bytes (sometimes also referred to as block size).

Data carriers < 2 kByte use 32 Byte block size,
Data carriers ≥ 2 kByte use 64 Byte block size.

Input

The function of the digital control input of the BIS C-6_0 can be selected.
The factory setting is "Reset".

– Reset

If Reset is selected, a High signal on this input causes a reset of the BIS C-6_0 processor. Commands already started will be cancelled.

– Head Select

If Head Select is selected, this input is used to select read/write heads.

Input Low: Head 1 selected.

Input High: Head 2 selected.

-> This function has always priority. The function "Both Read/Write Heads Active", which is selected using command 'HT', is then deactivated.

– Data bit on Data carrier

On recognition of a new data carrier a freely definable bit of a given address will be written, direct or inverted, on the data carrier. After a successful write operation, an output which can also be freely defined is set until the data carrier leaves the active read/write range.

-> The parameter "Dynamic operation" will be automatically reset.

– Not used

The input has no function.

Configuration

Configure Input/Output

IN/OUTPUT CONFIGURATION	
<input checked="" type="checkbox"/>	Outputs not used.
<input type="checkbox"/>	Output halfbyte of data contents of an address.
<input type="checkbox"/>	Compare contents of multiple addresses with a fixed value.
<input type="checkbox"/>	Compare contents of an address with various fixed values.
<input type="checkbox"/>	Compare contents of multiple addresses with content of one address.
<input type="checkbox"/>	Output data bits of variable addresses.
<input type="checkbox"/>	Program input as data bit to data carrier.
<p>< OK > < Shortform > < ESC = Exit > < Print > < F1 = Help ></p>	

Outputs not used

"Outputs not used" means processing of the outputs is deactivated.

OUTPUTS NOT USED	
<input checked="" type="checkbox"/>	Data to BIS
<p>< Store > < ESC = Exit > < F1 = Help ></p>	

"Data to BIS" transmits the data to the processor. "Store" stores the data in the configuration file on your computer.

Configuration

Output Halfbyte of the Data Contents of an Address

IN/OUTPUT CONFIGURATION	
<input type="checkbox"/>	Output not used.
<input checked="" type="checkbox"/>	Output halfbyte of data contents of an address.
<input type="checkbox"/>	Compare contents of multiple addresses with a fixed value.
<input type="checkbox"/>	Compare contents of an address with various fixed values.
<input type="checkbox"/>	Compare contents of multiple addresses with a content of one address.
<input type="checkbox"/>	Output data bits of variable addresses.
<input type="checkbox"/>	Program input as data bit to data carrier.
<p>< OK > < Shortform > < ESC = Exit > < Print > < F1 = Help ></p>	

OUTPUT HALFBYTE	
Data carrier address [9999]	
<input type="checkbox"/>	Output High Nibble ?
<input checked="" type="checkbox"/>	Output Low Nibble ?
<p>< Data to BIS > < Store > < ESC = Exit > < F1 = Help ></p>	

Either the upper or lower 4 bits (Nibble) of the 8 bits data contents of an address is output (Bit 0 on Output 1, Bit 1 on Output 2, etc.). Depending on the setting this is either the upper (High Nibble) or lower (Low Nibble). The address is given in decimal.

Configuration

Compare Contents of multiple Addresses with a fixed Value

The data contents of up to 4 addresses are compared with a fixed decimal value. To each address can be assigned which of the Outputs 1 to 4 is set or cancelled by a positive result of the comparison and whether in case of a negative result of the comparison the output shall not be changed or shall be set in contrary to the definition with the positive result (inverted response).
All addresses found within the read command will be processed.

Compare Contents of multiple addresses with fixed value			
Fixed value: [000]			
Address	Output	Positive compare.	Negative compare.
[9999]	[1]	(•) Set () Clear	() No change (•) Invert
[9999]	[2]	(•) Set () Clear	() No change (•) Invert
[9999]	[3]	(•) Set () Clear	() No change (•) Invert
[9999]	[4]	(•) Set () Clear	() No change (•) Invert

< Data to BIS > < Store > < ESC = Exit > < F1 = Help >

If the parameter "Process outputs with CT-Present" is included in the initialization, then this function also will be carried out on recognition of a new data carrier (one or more of the addresses given should be on the first page of the data carrier).

Configuration

Compare an Address with various fixed Values

The data contents of an address is compared with 4 fixed decimal values. For each fixed value is indicated which of the Outputs 1 to 4 is set or cancelled by a positive result of the comparison and whether in case of a negative result of the comparison the output shall not be changed or shall be set in contrary to the definition with the positive result (inverted response).

Compare contents of an address with various fixed Values			
Address: [9999]			
Fixed value	Output	Positive compare	Negative compare
[000]	[1]	(•) Set () Clear	() No change (•) Invert
[000]	[2]	(•) Set () Clear	() No change (•) Invert
[000]	[3]	(•) Set () Clear	() No change (•) Invert
[000]	[4]	(•) Set () Clear	() No change (•) Invert

< Data to BIS > < Store > < ESC = Exit > < F1 = Help >

If the parameter "Process outputs with CT-Present" is included in the initialization, then this function also will be carried out on recognition of a new data carrier (one or more of the addresses given should be on the first page of the data carrier).

Configuration

Compare Contents of multiple Addresses with the Content of another Address

The data contents of up to 4 addresses are compared with the data contents of another address. For each address is indicated which of the Outputs 1 to 4 is set or cancelled by a positive result and whether in case of a negative result of the comparison the output shall not be changed or shall be set in contrary to the definition with the positive result. All addresses found within the read command will be processed provided the address to be compared is within the range.

Compare contents of multiple addresses with content of one address			
Address: [9999]			
Address	Output	Positive compare	Negative compare
[9999]	[1]	(•) Set () Clear	() No change (•) Invert
[9999]	[2]	(•) Set () Clear	() No change (•) Invert
[9999]	[3]	(•) Set () Clear	() No change (•) Invert
[9999]	[4]	(•) Set () Clear	() No change (•) Invert

< Data to BIS > < Store > < ESC = Exit > < F1 = Help >

If the parameter "Process outputs with CT-Present" is included in the initialization, then this function also will be carried out on recognition of a new data carrier (one or more of the addresses given should be on the first page of the data carrier).

Configuration

Output Data Bits of variable Addresses

1 data bit of an address or 1 bit each from up to 4 addresses can be output on one of the 4 outputs and then inverted or not inverted.

Output data bits of variable addresses			
Address	Bit-Number	Output	Invert
[9999]	[1]	[1]	[] yes
[9999]	[2]	[1]	[] yes
[9999]	[3]	[1]	[] yes
[9999]	[4]	[1]	[] yes

< Data to BIS > < Store > < ESC = Exit > < F1 = Help >

If the parameter "Process outputs with CT-Present" is included in the initialization, then this function also will be carried out on recognition of a new data carrier (one or more of the addresses given should be on the first page of the data carrier).

Configuration

Program Input as a Databit to Data carrier

On recognition of a new data carrier the state of the digital input will be written as a direct or inverted bit on the data carrier. The address range is 0...31! Bit number of the address is 1...8. The outputs to be used for the ready and release signals should also be given. If release output is given as "0" then the release function will not be used. The procedure is described below.

Program input as databit to data carrier			
address	[00]	[]	Input inverse ?
bit number	[1]		
ready output	[1]		
release output	[0]		
< Data to BIS >		< Store >	< ESC = Exit > < F1 = Help >

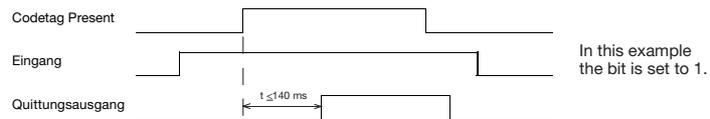
Procedure without Release Signal

On recognition of a new data carrier the defined bit of the given address will be written, direct or inverted. After a successful write operation, the given ready output is set until the data carrier leaves the active read/write range. The input must hold its state until the ready output is set.

The input state that is to be written as information on the data carrier must be present already before the presence of the new data carrier is recognized.

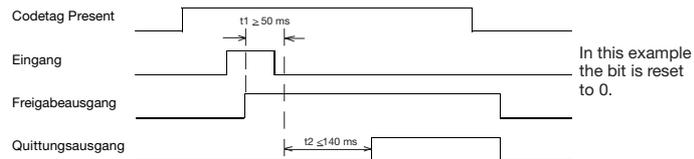
Configuration

Procedure without Release Signal (continued)



Procedure with Release Signal

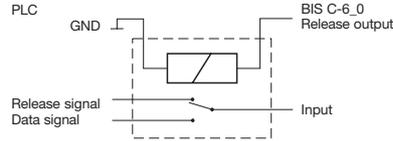
After the recognition of a new data carrier, the input state is continuously sampled until it is set (release given). The processor sets the release output and waits 50 ms. The input is then sampled and its state is taken over as bit-value. This value is written on the data carrier directly or inverted depending on the selected configuration. After the write operation, the given ready output is set until the data carrier leaves the active read/write range. The release output then reverts to low.



Configuration

Procedure with Release Signal
(continued)

The release output can be used to operate a relay in order to switch the input between the release and data signals.



As an input signal to a PLC this output can indicate that it is time to switch the data signal to the input of the BIS C-6_0. This signal is required at all places where there is a possibility that the data carrier can come into the read/write range before the input data signal is present at the processor.

This function must be additionally released for use during the initializing phase.

SHORTFORM I/O CONFIGURATION BIS C-6_0				
[0000]	[0]	[0]	[0]	[0]
[0000]	[0]	[0]	[0]	[0]
[0000]	[0]	[0]	[0]	[0]
[0000]	[0]	[0]	[0]	[0]
[0000]	[0]	[0]	[0]	[1]

< B = <- > < ESC = Exit > < F1 = Help >

If the initializing data are available in short form (e.g. on the processor cover after a replacement of the unit) the data can be entered directly into the mask "Shortform of I/O configuration BIS C-6_0" (see also Customer configuration on ¶ 30).

Customer Configuration

Initialization

Please note the settings in the label fields on the inside of the processor cover so that in case of repair of the processor the settings can be reset in the factory. Note the settings also in the following fields so that you can set, e.g. other units, to an identical configuration.

Interface	Baudrate	Data bit	Stop bit	Parity	
	[][][][][]	[]	[]	[]	
Protocol type	Block size	Input			
	[] [X] [X]	[]			
Parameters					
	[][][][][][]				

Do not change the settings marked here by crosses!

On the following page you will find an example which shows how you can print-out after initializing. Enter the settings in the appropriate fields so that you have them handy and can reproduce the settings at any time. You can then enter the data in short form into the mask. (see also ¶ 13).

SHORTFORM OF INITIALIZATION BIS C-				
[19200]	[8]	[1]	[N]	
[4]	[X]	[X]	[2]	[1]
[0110110]				

< B = <- > < ESC = Exit > < F1 = Help >

Do not change the settings marked here by crosses!

Customer Configuration

Initialization
(continued)

Example of a print-out after initialization which you can print with the software BISC600C.EXE.

```
Interface settings
Baudrate      : 19200 baud
Data bit      : 8
Stop bit      : 1
Parity        : None
Protocol
( ) BCC
( ) CR as terminator
( ) CR
( *) LF CR as terminator
Parameter
[ ] RS 485, 2-wire
[X] Codetag Present signalled with '*'
[X] Dynamic mode
[ ] Codetag Present signal on output 1
[X] Process outputs with Codetag Present
[X] Fast data carrier recognition
[ ] CRC-16 data checking
Block size
( ) 32 Byte Block size
( *) 64 Byte Block size
Input
( *) Input = RESET
( ) Input = Head selection
( ) Input = Data bit information on data carrier
( ) Input = No function
```

1	9	2	0	0	8	1	N
4	2*)	2*)	2	1			
0	1	1	0	1	1	0	

The entries in the field are either in clear text (as with Interface settings) or the number of the line marked. In the case of 'Parameter' the marked line is indicated by a 1.

*) Do not change the settings within the two fields marked.

Customer Configuration

Input/Output Configuration

Please note the settings in the label fields on the inside of the processor cover so that in case of repair of the processor the settings can be reset in the factory. Note the settings also in the following fields so that you can set, e.g. other units, to an identical configuration.

Address

Address

Address

Address

Fixed value Type

On the following page you will find an example which shows how you can print-out after initializing. Enter the settings in the appropriate fields so that you have them handy and can reproduce the settings at any time. You can then enter the data in short form into the mask.

Shortform of I/O Configuration BIS C-6_0

SHORTFORM I/O CONFIGURATION BIS C-6_0						
[0000]	[0]	[0]	[0]	[0]		
[0000]	[0]	[0]	[0]	[0]		
[0000]	[0]	[0]	[0]	[0]		
[0000]	[0]	[0]	[0]	[0]		
[0000]				[1]		
< B = <- > < ESC = Exit > < F1 = Help >						

Customer Configuration

Input/Output Configuration
(continued)

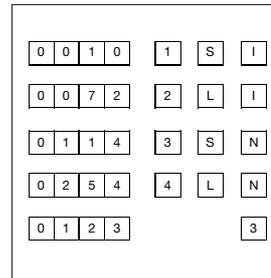
Example of a print-out after initialization which you can print with the software BISC600C.EXE.

Input/output configuration

- Output not used.
- Output halfbyte of data contents of an address.
- Compare contents of multiple addresses with a fixed value.
- Compare contents of an address with various fixed values.
- Compare contents of multiple addresses with content of one address.
- Output data bits of variable addresses.
- Program input as data bit to data carrier.

Definition

Fixed value: 123
 Address: 0010
 Output: 1
 Positive comparison: Set No change
 Negative comparison: Clear Invert
 Address: 0072
 Output: 2
 Positive comparison: Set No change
 Negative comparison: Clear Invert
 Address: 0114
 Output: 3
 Positive comparison: Set No change
 Negative comparison: Clear Invert
 Address: 0254
 Output: 4
 Positive comparison: Set No change
 Negative comparison: Clear Invert



Read/Write Times

Read Times from Data carrier to Processor in Static Mode
(Configuration: without dynamic mode, without CRC_16-data check)

For double read and compare:

Data carrier with 32 byte blocks		Data carrier with 64 byte blocks	
No. of bytes	Read time [ms]	No. of bytes	Read time [ms]
from 0 to 31	110	from 0 to 63	220
for each additional 32 bytes add	120	for each additional 64 bytes add	230
from 0 to 255	= 950	from 0 to 2047	= 7350

Write Times from Processor to Data carrier in Static Mode
(Configuration: without dynamic mode, without CRC_16-data check)

Including readback and compare:

Data carrier with 32 byte blocks		Data carrier with 64 byte blocks	
No. of bytes	Write time [ms]	No. of bytes	Write time [ms]
from 0 to 31	110 + n * 10	from 0 to 63	220 + n * 10
for 32 bytes or more	y * 120 + n * 10	for 64 bytes or more	y * 230 + n * 10

n = number of contiguous bytes to write
 y = number of blocks to be written

Example:

Read 17 bytes starting at address 187. Data carrier with 32 byte blocks.

Blocks 5 and 6 have to be accessed, because start address 187 is in block 5 and end address 203 is in block 6.

$$t = 2 * 120 + 17 * 10 = 410$$

The indicated times apply after the data carrier has been recognized. Otherwise an additional 45 ms must be added to allow for the energy field to be established until the data carrier is recognized.

Read/Write Times

Read Times from Data carrier to Processor in Dynamic Mode
(Configuration: with dynamic mode, without CRC_16-data check)

Read Times for 1 Block with double read and compare:

Data carrier with 32 byte blocks		Data carrier with 64 byte blocks	
No. of bytes	Read time [ms]	No. of bytes	Read time [ms]
from 0 to 3	14	from 0 to 3	14
for each additional byte add	3.5	for each additional byte add	3.5
from 0 to 31	112	from 0 to 63	224

m = highest address to be read

Formula: $t = (m + 1) * 3.5$ ms

Example: Read 11 bytes starting at address 9. Hence the highest address to be read is 19. This results in 70 ms.

Write Times from Processor to Data carrier in Dynamic Mode
(Configuration: with dynamic mode, without CRC_16-data check)

Including readback and compare:

Data carrier with 32 byte blocks		Data carrier with 64 byte blocks	
No. of bytes	Write time [ms]	No. of bytes	Write time [ms]
from 0 to 3	$14 + n * 10$	from 0 to 3	$14 + n * 10$
for each additional byte add	3.5	for each additional byte add	3.5

n = Number of contiguous bytes to be written

The indicated times apply after the data carrier has been recognized. Otherwise an additional 45 ms must be added to allow for the energy field to be established until the data carrier is recognized.

Programming Information

The preceding sections describe basic telegram sequence, and configuration and wiring of the interfaces. What now follows is information about the proper construction of the telegrams themselves.

Specific telegrams exist in the BIS C Identification System for particular tasks. They always begin with the command which is associated with the telegram type.

Telegram types with their associated commands (ASCII characters)

'L'	Read the data carrier with read/write select and block size
'P'	Write to the data carrier with read/write select and block size
'C'	Write a constant value to the data carrier with head and block size selection
'R'	Read the data carrier
'W'	Write to the data carrier
'H'	Select the read/write head and block size with the variations
'?'	Find the next data carrier (one time)
'!''	or find the next data carrier (continuously)
'B'	Process outputs
'Q'	Restart the processor (acknowledge)
'S'	Query the status message
'Z'	Initialize CRC_16-data check

Please note:

- Continuous querying on the interface is not permitted!
- The minimum wait time between two commands is 300 ms!

Programming Information

Telegram Contents

Start address and no. of bytes	The start address (A3, A2, A1, A0) and the number of bytes to send (L3, L2, L1, L0) are sent in decimal as ASCII characters. For the start address, the range 0000 to 8191 can be used, and for the number of bytes 0001 to 8192. A3 ... L0 represent one ASCII character each. Please note: Start address + number of bytes may not exceed the data carrier capacity.
Head number and block size	In the 'L' command (read with head select and block size) and 'P' command (write with head select and block size), first the number of the read/write head K ('1' or '2') and then the data carrier block size B ('0', '1') is sent. B = '0' means 64 Byte, B = '1' means 32 Byte.
Acknowledge	The acknowledgement <ACK> '0' is sent by the Identification System if the serially transmitted characters were recognized as correct and there is a data carrier in the active zone of a read/write head. In the 'R' command, the <ACK> '0' is only sent if the data is ready for transmission. <NAK> + Error No.' is sent if an error was recognized or if there is no data carrier in the active zone of a read/write head.
Start	<STX> starts the data transmission.
Transmitted Bytes	The data are transmitted code transparent (no data conversion).

Programming Information

BCC Block Check

The BCC block check is formed as an EXOR of the serially transmitted binary characters of the telegram block. Example: Read 128 bytes starting at address 13.
The command line without BCC is: 'L 0013 0128 20'. The BCC is formed:

```
'L' = 0100 1100 EXOR
0   = 0011 0000 EXOR
0   = 0011 0000 EXOR
1   = 0011 0001 EXOR
3   = 0011 0011 EXOR
0   = 0011 0000 EXOR
1   = 0011 0001 EXOR
2   = 0011 0010 EXOR
8   = 0011 1000 EXOR
2   = 0011 0010 EXOR
0'  = 0011 0000 EXOR
```

Block check result: BCC = 0100 0111 = 'G'

Variants for finish with BCC, Terminator

If necessary the finish using block check BCC can be replaced with a special ASCII character. This is:

– Carriage Return 'CR'

For hosts which always require a terminator character, this must always be included in the telegrams. Available are:

– Carriage Return 'CR' or
– Line Feed with Carriage Return 'LF CR'.

The various protocol variations are represented on the following \square .
See also: Configuration starting on \square 12.

Programming Information

Description of Various Protocol Variants

Reference is now made to the command string 'L 0013 0128 20 G' with 'G' as BCC. This command string is here shown in its possible variants; also shown are the various forms of acknowledgement with and without terminator:

Command line from host system to BIS	Acknowledge from BIS for correct reception	Acknowledge from BIS for incorrect reception
with BCC, but no terminator 'L 0013 0128 20 G'	No terminator <ACK> '0'	No terminator <NAK> '1'
with 'CR' instead of BCC, no terminator 'L 0013 0128 20 CR'	No terminator <ACK> '0'	No terminator <NAK> '1'
no BCC, with terminator 'CR' 'L 0013 0128 20 CR'	with terminator 'CR' <ACK> '0 CR'	with terminator 'CR' <NAK> '1 CR'
no BCC, with terminator 'LF CR' 'L 0013 0128 20 LF CR'	with terminator 'LF CR' <ACK> '0 LF CR'	with terminator 'LF CR' <NAK> '1 LF CR'

For <NAK> with error number a '1' was used here (no data carrier present) as an error example.

The respective positions for the additional terminator are shown in the tables in italics.

Programming Information

Read from data carrier with head select and block size
Write to data carrier with head select and block size

Task	Data Flow	Com- mand	Start address of first byte to be sent	Number of bytes to be sent	Head No.	Block size	End 2)	Acknowledge 3)	Terminator 4)	Start transmission	Terminator 4)	Data (from start address to start address + no. of bytes)	End 2)	Acknowledge 3)	Terminator 4)
Read	from host system to BIS	'L'	A3 A2 A1 A0 '0 0 0 0' to '8 1 9 1'	L3 L2 L1 L0 '0 0 0 1' to '8 1 9 2'	K	B	BCC '1' or '0' or '2' or '1' see 2)			<STX>	'CR' or 'LF CR'				
	from BIS to host system							<ACK>'0' or <NAK> + Error-No.	'CR' or 'LF CR'			D1 D2 D3 ... Dn	BCC or see 2)		
1)															
Write	from host system to BIS	'P'	A3 A2 A1 A0 '0 0 0 0' to '8 1 9 1'	L3 L2 L1 L0 '0 0 0 1' to '8 1 9 2'	K	B	BCC '1' or '0' or '2' or '1' see 2)			<STX>		D1 D2 D3 ... Dn	BCC or see 2)		
	from BIS to host system							<ACK>'0' or <NAK> + Error-No.	'CR' or 'LF CR'					<ACK>'0' or <NAK> + Error-No.	'CR' or 'LF CR'
1)															

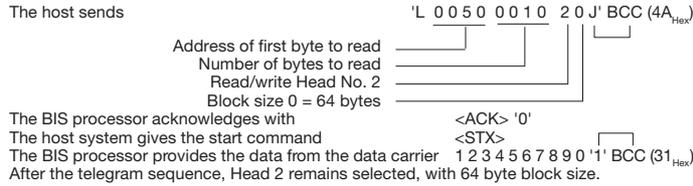
- 1) The commands 'Status' and/or 'Quit' are not permitted at this point.
- 2) Instead of block check BCC, depending on protocol variant either Carriage Return 'CR' or Line Feed with Carriage Return may be used.
- 3) <ACK> '0' is returned as acknowledgement if there is no error, or <NAK> + Error No. if an error occurs.
- 4) For protocol variants which always require a terminator, either 'CR' or 'LF CR' must be inserted here as a terminator.

Values inside apostrophes represent the respective character(s) in ASCII code.

Programming Information

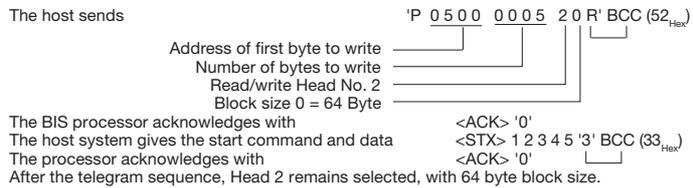
Telegram example for 138:
Read from data carrier with head select and block size
 Telegram with block check (BCC)

-> Head 1 is selected. Read 10 bytes starting at address 50 of the data carrier at read/write Head 2. The data carrier at Head 4 has a block size of 64 bytes.



Telegram example for 138:
Write to data carrier with read/write head select and block size
 Telegram with block check (BCC)

-> Head 1 is selected. Write 5 bytes starting at address 500 of the data carrier at read/write Head 2. The data carrier at Head 2 has a block size of 64 bytes.



Values inside apostrophes represent the respective character(s) in ASCII code.

Programming Information

Writing a constant value in the data carrier with read/write select and block size

This command can be used to erase a data carrier data. One saves the time for the transmission of the write byte.

Task	Data Flow	Command	Start address of first byte to be sent	Number of bytes to be sent	Head No.	Block size	End 2)	Acknowledge 3)	Terminator 4)	Start transmission	Terminator 4)	Data (from start address to start address + no. of bytes)	End 2)	Acknowledge 3)	Terminator 4)
Write	from host system to BIS	C'	A3 A2 A1 A0 '0' '0' '0' '0' to	L3 L2 L1 L0 '0' '0' '0' '1' to	K '1' '0' or or '1'	B '0' or or '1'	BCC see 2)			<STX>		D	BCC or see 2)		
	from BIS to host system		'8' '1' '9' '1'	'8' '1' '9' '2'	'2'			<ACK> '0' or <NAK> + Error- No.	'CR' or 'LF CR'					<ACK> '0' or <NAK> + Error- No.	'CR' or 'LF CR'

- 1) The commands Status and/or Quit are not permitted at this point.
- 2) Instead of block check BCC, depending on protocol variant either Carriage Return 'CR' or Line Feed with Carriage Return may be used.
- 3) <ACK> '0' is returned as acknowledgement if there is no error, or <NAK> + 'Error No.' if an error occurs.
- 4) For protocol variants which always require a terminator, either 'CR' or 'LF CR' must be inserted here as a terminator.

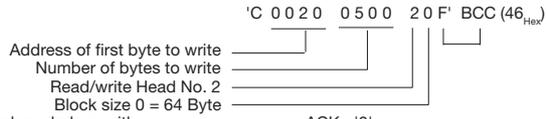
Data within angle brackets are control characters.
 Values inside apostrophes represent the respective character(s) in ASCII code.

Programming Information

Telegram example for 140:
Write to data carrier with read/write head select and block size with block check (BCC)

-> Head 1 is selected. Write 500 bytes of ASCII data value 0 (30_{Hex}) starting at address 20 of the data carrier at read/write Head 2. The data carrier at Head 2 has a block size of 64 bytes.

The host sends



The BIS processor acknowledges with <ACK> '0'
 The host system gives the start command and data <STX> '0 2' BCC (32_{Hex})
 The processor acknowledges with <ACK> '0'

After the telegram sequence, Head 2 remains selected, with 64 byte block size.

Data within angle brackets are control characters.
 Values inside apostrophes represent the respective character(s) in ASCII code.

Programming Information

Read from Data carrier, Write to Data carrier

Task	Data Flow	Com- mand	Start address of first byte to send	Number of bytes to send	End (2)	Acknow- ledge (3)	Termi- nator (4)	Start trans- mission	Termi- nator (4)	Data (from start address to start address + no. of bytes)	End (2)	Acknow- ledge (3)	Termi- nator (4)
Read	from host system to BIS	'R'	A3 A2 A1 A0 '0 0 0 0'	L3 L3 L1 L0 '0 0 0 1'	BCC or see '8 1 9 1'			<STX>	'CR' or 'LF CR'				
	from BIS to host system					<ACK> '0' or <NAK> + Error-No.	'CR' or 'LF CR'			D1 D2 D3 ... Dn	BCC or see (2)		
1)													
Write	from host system to BIS	'W'	A3 A2 A1 A0 '0 0 0 0'	L3 L3 L1 L0 '0 0 0 1'	BCC or see '8 1 9 1'			<STX>		D1 D2 D3 ... Dn	BCC or see (2)		
	from BIS to host system					<ACK> '0' or <NAK> + Error-No.	'CR' or 'LF CR'					<ACK> '0' or <NAK> + Error-No.	'CR' or 'LF CR'
1)													

- 1) The commands 'Status' and/or 'Quit' are not permitted at this point.
- 2) Instead of block check BCC, depending on protocol variant either Carriage Return 'CR' or Line Feed with Carriage Return may be used.
- 3) <ACK> '0' is returned as acknowledgement if there is no error, or <NAK> + Error No. if an error occurs.
- 4) For protocol variants which always require a terminator, either 'CR' or 'LF CR' must be inserted here as a terminator.

Values inside apostrophes represent the respective character(s) in ASCII code.

Programming Information

Telegram example from 42:
Read from Data carrier with block check (BCC)

Read from Data carrier: -> Read 10 bytes starting at address 50.
 The host sends 'R 0 0 5 0 0 0 1 0 V' BCC (56_{Hex})
 Address of first byte to read _____
 Number of bytes to read _____
 The BIS processor acknowledges with <ACK> '0'
 The host gives the start command <STX>
 The BIS processor provides the data from the data carrier 1 2 3 4 5 6 7 8 9 0 'SOH' BCC (01_{Hex})

Telegram example from 42:
Write to Data carrier with block check (BCC)

Write to Data carrier: -> Write 5 bytes starting at address 500.
 The host system sends 'W 0 5 0 0 0 0 0 5 W' BCC (57_{Hex})
 The BIS processor acknowledges with <ACK> '0'
 The host sends the data <STX> 1 2 3 4 5 '3' BCC (33_{Hex})
 The BIS processor acknowledges with <ACK> '0'

The 'R' and 'W' commands represent a subtype of the 'L' and 'P' commands.
 Values inside apostrophes represent the respective character(s) in ASCII code.

Programming Information

Selecting a Read/Write Head

The 'H1' command selects Read/Write Head 1, 'H2' Read/Write Head 2, and 'HT' (Head Twin) both Read/Write Heads.

If both heads are selected, please note:

1. Only one data carrier is allowed to be in the active zone of a read/write head at a time.
2. The read or write time increases by approx. 40 ms - regardless of the data amount to be read or written. (This does not apply to the data carrier recognition).
3. The positive acknowledgement for a read or write action is no longer <ACK> '0' but rather <ACK> '1' or <ACK> '2', depending on at which read/write head there is a data carrier to be read from or written to.

Task	Data Flow	Com-mand	Head number	End 2)	Acknowledge 3)	Terminator 4)
Select Read/Write Head	from host system to BIS	'H'	'1', '2' or 'T'	BCC or see 2)		
	from BIS to host system				<ACK>'0' resp. <NAK> + Error-No.	'CR' or 'LF CR'
			1)			

- 1) The commands 'Status' and/or 'Quit' are not permitted at this point.
- 2) Instead of block check BCC, depending on protocol variant either Carriage Return 'CR' or Line Feed with Carriage Return may be used.
- 3) <ACK> '0' is returned as acknowledgement if there is no error, or <NAK> + Error-No. if an error occurs.
- 4) For protocol variants which always require a terminator, either 'CR' or 'LF CR' must be inserted here as a terminator.

Telegram example:
Selecting a Read/Write Head with block check (BCC)

-> Switch to Head 1.
 The host sends 'H 1 y' BCC (79_{Hex})
 The BIS processor acknowledges with <ACK> '0'

Values inside apostrophes represent the respective character(s) in ASCII code.

Programming Information

Find Next Data carrier (one time)

The following telegram is used to find the next data carrier. The next following read/write head is selected and checked to see if a data carrier is in front of this read/write head. If yes, the first 4 bytes of the data carrier are read. The telegram reply then contains the corresponding number of the read/write head and the four bytes read. If no data carrier is found, the original read/write head is reselected and checked. If no data carrier is found here, then the telegram reply is: 'H ? 0000 w'.

'H ?' recognizes any data carrier, regardless of the preset block size, assuming that read/write head and data carrier are compatible.

Task	Data Flow	Com- mand	Des.	End (2)	Acknow- ledge	Termi- nator 3)	Reply	Head number	Data from data carrier	End (2)
Find next data carrier (one time)	from host system to BIS	'H'	'?'	BCC or see 2)						
	from BIS to host system				<ACK>'0'	'CR' or 'LF CR'	'H'	'1', '2' or '?'	D1 D2 D3 D4	BCC or see 2)
1)										

- 1) The commands 'Status' and/or 'Quit' are not permitted at this point.
- 2) Instead of block check BCC, depending on protocol variant either Carriage Return 'CR' or Line Feed with Carriage Return may be used.
- 3) For protocol variants which always require a terminator, either 'CR' or 'LF CR' must be inserted here as a terminator.

Telegram example:
Find Next Data carrier (one time)
with block check (BCC)

-> Head 1 is selected. Only read/write head 2 has a data carrier in front of it, whose first four bytes are 9876.

The host sends 'H ? w' BCC (77_{Hex})

The BIS processor acknowledges with <ACK> '0 H 2 9 8 7 6 z' BCC (7A_{Hex})

Values inside apostrophes represent the respective character(s) in ASCII code.

Programming Information

Find Next Data carrier (continuous)

The following telegram is used to find the next data carrier. The next following read/write head is selected and checked to see if a data carrier is in front of this read/write head. If yes, the first four bytes of the data carrier are read. The telegram reply then contains the corresponding number of the read/write head and the four bytes read. If no data carrier is found, the original read/write head is reselected and checked. This procedure is repeated until a data carrier is found.

'H !' recognizes any data carrier, regardless of the preset block size, assuming that read/write head and data carrier are compatible.

Task	Data Flow	Com- mand	Des.	End (2)	Acknow- ledge	Termi- nator 3)	Reply	Head number	Data from data carrier	End (2)
Find next data carrier (contin.)	from host system to BIS	'H'	'!'	BCC or see 2)						
	from BIS to host system				<ACK>'0'	'CR' or 'LF CR'	'H'	'1' or '2'	D1 D2 D3 D4	BCC or see 2)
1)										

- 1) The commands 'Status' and/or 'Quit' are not permitted at this point.
- 2) Instead of block check BCC, depending on protocol variant either Carriage Return 'CR' or Line Feed with Carriage Return may be used.
- 3) For protocol variants which always require a terminator, either 'CR' or 'LF CR' must be inserted here as a terminator.

Telegram example:
Find Next Data carrier (continuous)
with block check (BCC)

-> Read/write head 2 has a data carrier in front of it whose first four bytes are 9876.

The host sends 'H ! i' BCC (07_{Hex})

The BIS processor acknowledges with <ACK> '0 H 2 9 8 7 6 z' BCC (7A_{Hex})

Values inside apostrophes represent the respective character(s) in ASCII code.

Programming Information

Processing the Outputs

A telegram can be sent to set or cancel the four outputs.

Task	Data Flow	Com- mand	Designator	Termi- nator 2)	Acknow- ledge	Termi- nator 3)
Process outputs (set or cancel)	from host system to BIS	'B'	'00' bis 'A1' (see below)	BCC or see 2)		
	from BIS to host system				<ACK>'0'	'CR' or 'LF CR'
1)						

Designator meaning:	Output No.	0	1	2	3	all outputs
Cancel output		00	10	20	30	A0
Set output		01	11	21	31	A1

- 1) The commands 'Status' and/or 'Quit' are not permitted at this point.
- 2) Instead of block check BCC, depending on protocol variant either Carriage Return 'CR' or Line Feed with Carriage Return may be used.

Telegram example:
Processing the Outputs
with block check (BCC)

The host sends 'B 21 A' BCC (41_{Hex})
The BIS processor acknowledges with <ACK> '0'

After the telegram is completed, Output 2 is set.

Values inside apostrophes represent the respective character(s) in ASCII code.

Programming Information

Show Output Condition

This telegram is used to check the condition (status) of all four outputs.

Task	Data Flow	Com- mand	Designator	End 2)	Acknow- ledge	Condition of the 4 outputs	Termi- nator 3)	End 2)
Show output condition	from host system to BIS	'B'	'A0'	BCC or see 2)				
	from BIS to host system				<ACK>'0'	'XXXX' '0' =cancelled, '1' = set	'CR' or 'LF CR'	BCC or see 2)
1)								

Output status is shown in sequential order 0 1 2 3

- 1) The commands 'Status' and/or 'Quit' are not permitted at this point.
- 2) Instead of block check BCC, depending on protocol variant either Carriage Return 'CR' or Line Feed with Carriage Return may be used.

Telegram example:
Show Output Condition
with block check (BCC)

-> Outputs 0 and 1 are set, outputs 2 and 3 are cancelled.
The host sends 'BAO L' BCC (4C_{Hex})
The BIS processor acknowledges with <ACK> '0 1100 NUL' BCC (00_{Hex})

Values inside apostrophes represent the respective character(s) in ASCII code.

Programming Information

Restart the Processor (Quit)

Sending the Restart command causes a telegram in process to be aborted and puts the processor in the ground state. After this telegram is acknowledged, an approx. 500 ms pause should be allowed before starting a new telegram.

Important! The Quit command is not permitted while the processor is waiting for a terminator (BCC, 'CR' or 'LF CR'). In this situation, the Quit would be incorrectly interpreted as a terminator or datum.

Task	Data Flow	Command	Terminator 2)	Acknowledge	Terminator 2)
Restart (Quit)	From host system to BIS	'Q'	BCC or see 2)		
	From BIS to host system			'Q'	BCC or see 2)

- 1) The commands 'Status' and/or 'Quit' are not permitted at this point.
 2) Instead of block check BCC, depending on protocol variant either Carriage Return 'CR' or Line Feed with Carriage Return may be used.

Telegram example:
Restart the Processor (Quit)
 with block check (BCC)

Put the BIS system into the ground state.

The host sends 'Q Q' BCC (51_{Hex})
 The BIS processor acknowledges with 'Q Q' BCC (51_{Hex})

Values inside apostrophes represent the respective character(s) in ASCII code.

Programming Information

Checking Status Message

The status telegram is used to check what kind of telegram is in process.

Important! The Quit command is not permitted while the processor is waiting for a terminator (BCC, 'CR' or 'LF CR'). In this situation, the Quit would be incorrectly interpreted as a terminator or datum.
 Values inside apostrophes represent the respective character(s) in ASCII code. ' ' = space = ASCII 20 (Hex).

Task	Data Flow	Command	Terminator 2)	Status Message	Terminator 2)
Check Status Message	From host system to BIS	'S'	BCC or see 2)		
	From BIS to host system			'S ' 'R' 'W' 'L' 'P' or 'H'	BCC or see 2)

- 1) The commands 'Status' and/or 'Quit' are not permitted at this point.
 2) Instead of block check BCC, depending on protocol variant either Carriage Return 'CR' or Line Feed with Carriage Return may be used.

Status Messages and their Meaning:

- 'S L' = Read data carrier with head select and data carrier block size
- 'S P' = Write to data carrier with head select and data carrier block size
- 'S R' = Read data carrier
- 'S W' = Write to data carrier
- 'S H' = Select read/write head and data carrier block size
- 'S _' = No telegram in process

Programming Information

Telegram example from ¶ 50: -> Check the BIS status right after a **read telegram** has been sent.

Checking Status

Message The host sends 'S S' BCC (53_{Hex})
 with block check (BCC) The BIS acknowledges with 'S L US' BCC (1F_{Hex})

-> Check the BIS status right after a **write telegram** has been sent.

The host sends 'S S' BCC (53_{Hex})
 The BIS processor acknowledges with 'S P ETX' BCC (03_{Hex})

-> Check the BIS status right after a **head select telegram** has been sent.

The host sends 'S S' BCC (53_{Hex})
 The BIS processor acknowledges with 'S H ESC' BCC (1B_{Hex})

-> Check the BIS status. **No telegram** was just sent.

The host sends 'S S' BCC (53_{Hex})
 The BIS processor acknowledges with 'S _' BCC (20_{Hex})

Important: a status check during a read or write operation on a data carrier (LED Codetag Operating on) increases the read or write time. This cause a condition, especially while reading on the fly, that the time the data carrier is in the active zone of the read/write head is insufficient for a complete read or write. Continuous status checking affects the processing of the data carrier; in some cases, the data carrier may not be recognized!
 Values inside apostrophes represent the respective character(s) in ASCII code.

Programming Information

Initialize CRC_16-data check

This telegram initializes a data carrier located at the active read/write head for use of CRC_16 data checking. This telegram must also be send again if a CRC error results from a failed write operation, i.e., the data carrier must be reinitialized in order to use it again.

Please note the table on ¶ 16! The indicated number of usable bytes may not be exceeded, i.e., the sum of start address plus number of bytes must not exceed the data carrier memory capacity!

Task	Data Flow	Command	Start address of first byte to be sent	Number of bytes to be sent	Head No.	Block size	End 2)	Acknowledge 3)	Terminator 4)	Start transmission	Data (from start address to start address + no. of bytes)	End 2)	Acknowledge 3)	Terminator 4)
Initialize CRC_16 range	from host system to BIS	Z'	A3 A2 A1 A0 '0 0 0 0' to '8 1 9 1'	L3 L3 L1 L0 '0 0 0 1' to '8 1 9 2'	K '1', '2', '3', or '4'	B '0' or '1'	BCC see 2)		-STX-	D1 D2 D3 ... Dn	BCC or see 2)			
	from BIS to host system							-ACK->'0' or -NAK-> + Error- No.	'CR' or 'LF CR'				-ACK->'0' or -NAK-> + Error- No.	'CR' or 'LF CR'

- 1) The Status and/or Quit commands are not permitted at this point.
- 2) Instead of BCC block check, depending on the protocol variant either Carriage Return 'CR' or Line Feed with Carriage Return 'LF CR' may be used.
- 3) <ACK> '0' is sent as an acknowledgement if there was no error, or <NAK> + 'Error No.' if there was an error.
- 4) For protocol variants which always need a terminator, either 'CR' or 'LF CR' must be inserted here.

The characters between the apostrophes represent the respective ASCII character(s). '_' = Space = ASCII 20_{Hex}.

Error Numbers

Error Numbers

The BIS C-6_0 always outputs an error number. The meaning of these error numbers is indicated in the following table.

No.	Error Description	Effect
1	No data carrier present	Telegram aborted, processor goes into ground state.
2	Read error	Read telegram aborted, processor goes into ground state.
3	Read aborted, since the data carrier was removed.	Processor goes into ground state.
4	Write error	Write telegram aborted, processor goes into ground state.
5	Writing aborted, since the data carrier was removed.	Processor goes into ground state.
6	Interface error	Processor goes into ground state. (parity or stop bit error)
7	Telegram format error	Processor goes into ground state. Possible format errors: - Command is not 'L'/P'/C'/R'/W'/H'/B'/Q'/S'/Z'. - Start address or number of bytes exceed permissible range

*) **Note:** If a CRC data check is used, error message "E" could result if error 4 or 5 was not cleared.

english

BALLUFF 53

Error Numbers

Error Numbers (continued)

No.	Error Description	Effect
8	BCC error, the transmitted BCC is wrong.	Telegram is aborted, processor goes into ground state.
9	Cable break from read/write head or cable not connected, Codetag Present LED flashes.	Telegram is aborted, processor goes into ground state. If both read/write heads were selected using 'HT', one head may not be connected. If both read/write heads are selected, the cable break message only comes if there is no data carrier in front of the connected, functional head.
A	New command not possible, since a read command is already in process.	After error message the read command is stopped, internally, but not acknowledged. Processor goes into ground state.
B	New command not possible, since a write command is already in process.	After error message the write command is stopped, internally, but not acknowledged. Processor goes into ground state.
C	New command not possible, since a head select is already in process.	After the error message, no positive acknowledge is given, even though the head select was successful. Processor goes into ground state.
E	CRC error: the CRC on the data carrier is wrong. *)	Telegram aborted, processor goes into ground state.

*) **Note:** If a CRC data check is used, error message "E" could result if in the preceding command error 4, 5 or B was reported.

BIS C-600 Interface Information

LED Display:
System Ready
Codetag Present
Codetag Operating

The BIS C-6_0 Processor uses three LED's on the front panel to indicate the most important operating conditions.

Condition	LED	Meaning
System Ready	on (green)	Supply voltage OK; no hardware error
	off	Supply voltage or hardware not OK
Codetag Present	on (yellow)	Data carrier ready to read or write. (If a read/write error occurs during a read/write operation, System Ready LED goes out, if the protocol variant "without error number" is used!)
	flashes	Read/write head cable break or not connected. If both read/write heads were selected using 'HT', one head may not be connected. If both read/write heads are selected, the cable break message only comes if there is no data carrier in front of the connected, functional head.
	off	No data carrier in active zone of read/write head
Codetag Operating	on (yellow)	Command being processed
	off	No command in process

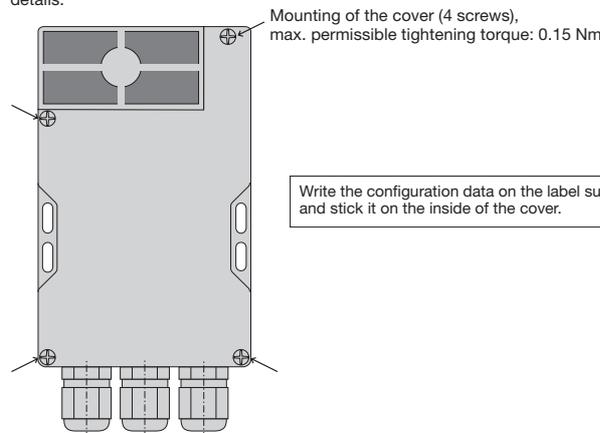
If all three LED's flash on and off synchronously, the processor needs to be returned to the factory for repair.

BIS C-600 Interface Information

**Opening the
BIS C-600 Processor**

In order to make the interface connections, the BIS C-600 processor must be opened.

Take care that the unit has no power to it. Unthread the 4 screws and remove the cover. Feed the connector cable through the cable fittings. See the following  for additional wiring details.

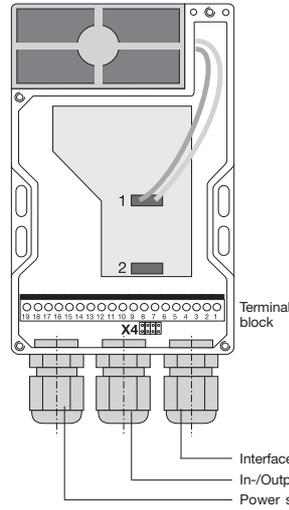


Opening the processor

BIS C-600 Interface Information

Connections to the BIS C-600 Processor

Processor open



Wiring the terminal block

19	18	17	16	15	14
+ VS	- VS	$\frac{1}{2}$	TxD	RxD	COM
POWER			Service		

13	12	11	10	9	8	7	6
+ VS	- VS	1	2	3	4	+IN	-IN
OUTPUT					INPUT		

5	4	3	2	1
COM	T (A)	T (B)	R (A)	R (B)
RS 422 (4-Draht)				

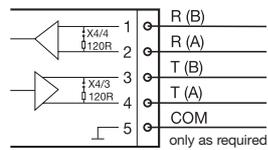
T = Transmit
R = Receive

5	4	3	2	1
COM	R/T (A)	R/T (B)		
RS 485 (2-Draht)				

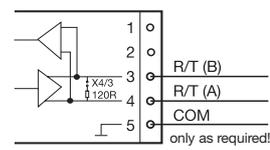
BIS C-600 Interface Information

Connection cable and Interface RS 422/RS 485

The BIS C-600-007...02 processor uses a serial interface for communicating with a host system (e.g. PC or PLC). Depending on configuration, either the interface RS 422 (4-wire) or the RS 485 (2-wire) may be used.



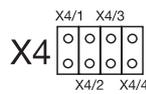
Terminal wiring for
RS 422 interface
(4-wire)



Terminal wiring for
RS 485 interface
(2-wire)

RS 422: Jumper for shunt X 4/1
must be plugged in!

RS 485: Jumper for shunt X 4/2
must be plugged in!



Jumper plug for
termination resistor

Termination resistor	Jumper for X 4/3	Jumper for X 4/4
passive	-	-
active	connected	connected

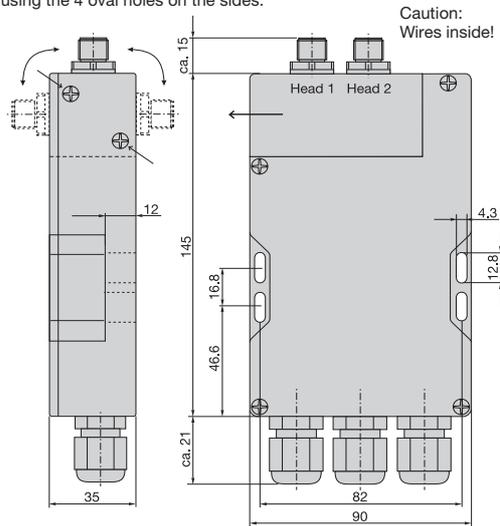
BIS C-600 Processor / Head Installation

Installation of the BIS C-600 Processor and Read/Write Head or Adapter BIS C-650

The processor is mounted using the 4 oval holes on the sides.

Depending on model, the processor includes a read/write head or adapter for separate read/write head. Both the read/write head and the adapter can be rotated as desired by +/-90 deg. (see illustration). Be sure power is off. Remove both screws (indicated by arrows in diagram). Carefully remove the head or adapter in the direction of the arrow shown in right illustration.

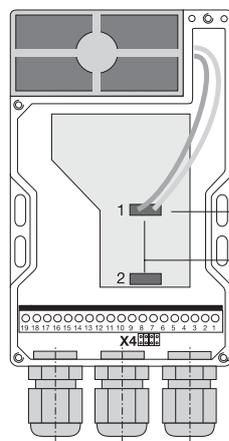
**Caution:
cables inside!**
Assemble in the desired orientation and refasten the screws.



BIS C-600 Processor / Head Installation

Installing a Read/Write Head or Adapter BIS C-650

To change a read/write head: Turn power off and open processor. Remove mounting screws from read/write head (see ¶ 59). Unthread the 4 screws and remove the cover (see ¶ 56). Remove the read/write head connector from the board and pull the cable out through the cable passage. To attach the new head, follows these steps in reverse.



To attach a BIS C-650 Adapter, proceed as described above. Both connection cables must be plugged into the board.

Connection for integrated read/write head

Connections for Adapter BIS C-650

1 = Head 1

2 = Head 2

Mounting of the cover (4 screws),
max. permissible tightening torque: 0.15 Nm

61

BIS C-600 Technical Data

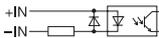
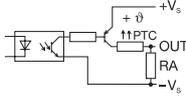
Dimensions, Weight	Housing	Plastic PS
	Dimensions with read/write head BIS C-652	167 x 90 x 35 mm
	Dimensions with Adapter BIS C-650	145 x 90 x 35 mm
	Weight	500 g
Temperature Range	Operating temperature	0 °C to + 60 °C
Connections	Terminal block	19-pin
	Cable fittings	3 x PG 9
	Cable diameter	4 to 8 mm
	Conductor size	0.14 to 1 mm ²
	with crimp terminals	0.25 to 0.34 mm ²
Protection Class	IP Rating	IP 65
Electrical Connections	Supply Voltage V_s, Input	24 V DC \pm 20 %
	Ripple	\leq 10 %
	Current draw	\leq 400 mA
	Serial Interface	RS 422/RS 485
	Read/Write Head	integrated, BIS C-65_ and following *); option with mounted adapter BIS C-650 *) 2 x connectors 4-pin (male) for all read/write heads BIS C-3_ _ with 4-pin connector (female), except BIS C-350 and BIS C-352

*) can be rotated by +/-90 deg.

english BALLUFF 61

62

BIS C-600 Technical Data

Electrical Connections	Digital Input (+IN, -IN)	Optocoupler isolated 4 V to 40 V Control voltage active 1.5 V to -40 V Control voltage inactive 11 mA Input current at 24 V 5 ms Typical delay time	
	Control Outputs (O1 to O4)	Optocoupler isolated max. 20 mA Voltage drop at 20 mA approx. 2.5 V Output resistance R_o 10 k Ω to - V_s Logic PNP (sourcing) Supply voltage, Output V_s 24 V DC \pm 20 % Ripple \leq 10 %	
	Serial Interface (TTY)	RS 422 (4-wire) / RS 485 (2-wire) Optocoupler isolated	
Function Displays	System Ready	LED green	
	Codetag Present	LED yellow	
	Codetag Operating	LED yellow	



With the CE Mark we affirm that our products are in accordance with the requirements of the EU (European Union) Guideline 89/336/EEC (EMC Guideline)

and the EMC Law. It has been verified in our EMC Laboratory, which is accredited by the DATech for Testing of Electromagnetic Compatibility, that Balluff products meet the EMC requirements of the Harmonized Standard

EN 61000-6-4 (Emission), EN 61000-6-2 (Noise Immunity)

62 BALLUFF english

BIS C-600 Ordering Information

Ordering Code

BIS C-600-007- - - -02-KL1

Balluff Identification System

Series C

Hardware type
600 = Compact

Software type
007 = Balluff protocol

Read/Write Head
000 = no read/write head
650 = with two connections for external read/write heads BIS C-3_ _
(except BIS C-350 and -352)
651 = with read/write head BIS C-651 (circular antenna on top)
652 = with read/write head BIS C-652 (circular antenna on front)
653 = with read/write head BIS C-653 (bar antenna)

Interface
02 = RS 422/RS 485

User Connections
KL1 = 3x PG 9 cord seal fittings

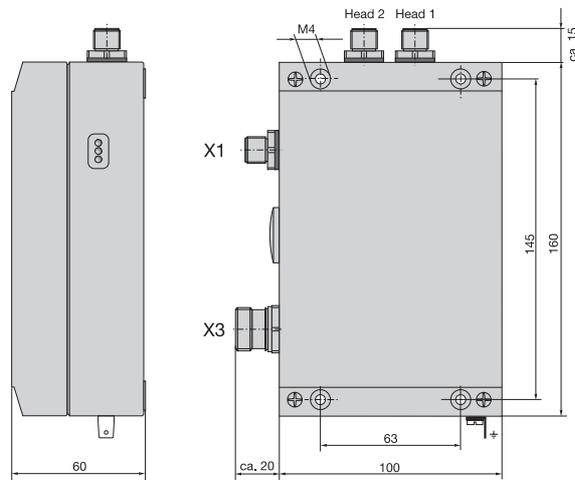
english

BALLUFF | 63

BIS C-620 Processor Installation

Installing the
BIS C-620 processor

The processor is mounted using 4 size M4 screws.

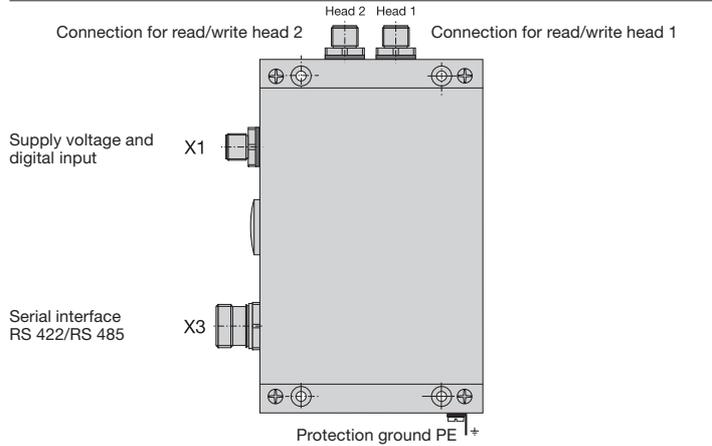


64 | BALLUFF

english

BIS C-620 Interface Information

**Wiring for the
BIS C-620 processor**



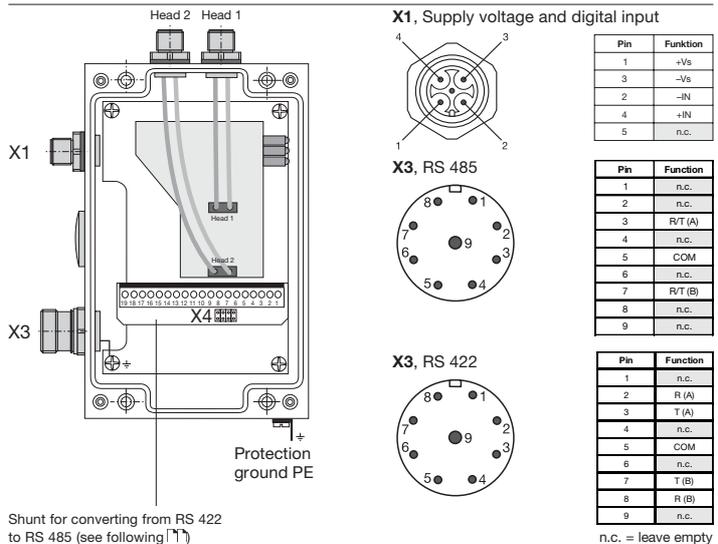
Connection locations
and names

**Opening the
BIS C-620 processor**

To insert the shunt for converting from RS 422 to RS 485, the BIS C-620 processor must be opened.
Ensure that the device is turned off. Remove the 4 screws on the BIS C-620 and lift off the cover. See following for additional details.

BIS C-620 Interface Information

**Wiring for the
BIS C-620 processor**



Shunt for converting from RS 422
to RS 485 (see following)

X1, Supply voltage and digital input

Pin	Function
1	+Vs
3	-Vs
2	-IN
4	+IN
5	n.c.

X3, RS 485

Pin	Function
1	n.c.
2	n.c.
3	R/T (A)
4	n.c.
5	COM
6	n.c.
7	R/T (B)
8	n.c.
9	n.c.

X3, RS 422

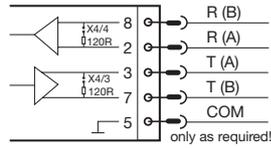
Pin	Function
1	n.c.
2	R (A)
3	T (A)
4	n.c.
5	COM
6	n.c.
7	T (B)
8	R (B)
9	n.c.

n.c. = leave empty

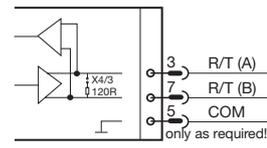
BIS C-620 Interface Information

Connection cable and Interface RS 422/RS 485

The BIS C-620-007...02 processor uses a serial interface for communicating with a host system (e.g. PC or PLC). Depending on configuration, either the interface RS 422 (4-wire) or the RS 485 (2-wire) may be used.



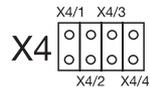
Terminal wiring for RS 422 interface (4-wire)



Terminal wiring for RS 485 interface (2-wire)

RS 422: Jumper for shunt X 4/1 must be plugged in!

RS 485: Jumper for shunt X 4/2 must be plugged in!



Jumper plug for termination resistor

Termination resistor	Jumper for X 4/3	Jumper for X 4/4
passive	-	-
active	connected	connected

BIS C-620 Technical Data

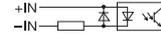
Dimensions, weight	Housing	Metal
	Dimensions	175 x 120 x 60 mm
	Weight	820 g
Temperature range	Ambient temperature	0 °C to +60 °C
Connections	Integrated connector X1	5-pole (male)
	Integrated connector Head 1, Head 2	4-pole (male)
	Round connector X3	9-pole (male)
Protection	Protection	IP 65 (when connected)
Electrical connections	Input X1, supply voltage V_s	DC 24 V ± 20 %
	Ripple	≤ 10 %
	Current draw	≤ 400 mA
	Terminal X3, serial interface	RS 422 (4-wire) / RS 485 (2-wire)
	Connections to read/write head Head 1, Head 2	2 x integral connectors 4-pole (male) for all BIS C-3_ _ read/write heads with 4-pole connector (female), not BIS C-350 and BIS C-352

BIS C-620
Technical Data

Electrical Connections

Digital Input (+IN, -IN)
Control voltage active
Control voltage inactive
Input current at 24 V
Typical delay time

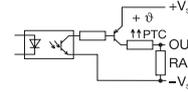
Optocoupler isolated
4 V to 40 V
1,5 V to -40 V
11 mA
5 ms



Control Outputs (01 to 04)

Output current
Voltage drop at 20 mA
Output resistance R_A
Logic
Supply voltage, Output V_S
Ripple

Optocoupler isolated
max. 20 mA
approx. 2.5 V
10 k Ω to $-V_S$
PNP (sourcing)
24 V DC \pm 20 %
 \leq 10 %



Function Displays

System Ready
Codetag Present
Codetag Operating

LED green
LED yellow
LED yellow



With the CE Mark we affirm that our products are in accordance with the requirements of the EU (European Union) Guideline 89/336/EEC (EMC Guideline)

and the EMC Law. It has been verified in our EMC Laboratory, which is accredited by the DATech for Testing of Electromagnetic Compatibility, that Balluff products meet the EMC requirements of the Harmonized Standard

EN 61000-6-4 (Emission), EN 61000-6-2 (Noise Immunity).

BIS C-620
Ordering Information

Ordering code

BIS C-620-007-050-02-ST2

Balluff Identification System

Series C

Hardware Type

620 = Metal housing

Software Type

007 = Balluff protocol

Read/write head

050 = with two connectors for external read/write heads

BIS C-3_ _ (except BIS C-350 and -352)

Interface

02 = RS 422 (4-wire) configurable for RS 485 (2-wire)

User Connections

ST2 = Connector variant

1 round connector for supply voltage,

1 round connector for RS 422 / RS 485

Accessories
(optional, not included)

Article

Ordering code

Mating connector for X1
for X3

BKS-S 79-00
BKS-S 84-00

Protection cap for Head 1, Head 2

BES 12-SM-2

Appendix, ASCII Table

Deci- mal	Hex	Control Code	ASCII	Deci- mal	Hex	Control Code	ASCII	Deci- mal	Hex	ASCII									
0	00	Ctrl @	NUL	22	16	Ctrl V	SYN	44	2C	,	65	41	A	86	56	V	107	6B	k
1	01	Ctrl A	SOH	23	17	Ctrl W	ETB	45	2D	-	66	42	B	87	57	W	108	6C	l
2	02	Ctrl B	STX	24	18	Ctrl X	CAN	46	2E	.	67	43	C	88	58	X	109	6D	m
3	03	Ctrl C	ETX	25	19	Ctrl Y	EM	47	2F	/	68	44	D	89	59	Y	110	6E	n
4	04	Ctrl D	EOT	26	1A	Ctrl Z	SUB	48	30	0	69	45	E	90	5A	Z	111	6F	o
5	05	Ctrl E	ENQ	27	1B	Ctrl [ESC	49	31	1	70	46	F	91	5B	[112	70	p
6	06	Ctrl F	ACK	28	1C	Ctrl \	FS	50	32	2	71	47	G	92	5C	\	113	71	q
7	07	Ctrl G	BEL	29	1D	Ctrl]	GS	51	33	3	72	48	H	93	5D]	114	72	r
8	08	Ctrl H	BS	30	1E	Ctrl ^	RS	52	34	4	73	49	I	94	5E	^	115	73	s
9	09	Ctrl I	HT	31	1F	Ctrl _	US	53	35	5	74	4A	J	95	5F	_	116	74	t
10	0A	Ctrl J	LF	32	20		SP	54	36	6	75	4B	K	96	60	`	117	75	u
11	0B	Ctrl K	VT	33	21		!	55	37	7	76	4C	L	97	61	a	118	76	v
12	0C	Ctrl L	FF	34	22		*	56	38	8	77	4D	M	98	62	b	119	77	w
13	0D	Ctrl M	CR	35	23		#	57	39	9	78	4E	N	99	63	c	120	78	x
14	0E	Ctrl N	SO	36	24		\$	58	3A	:	79	4F	O	100	64	d	121	79	y
15	0F	Ctrl O	SI	37	25		%	59	3B	;	80	50	P	101	65	e	122	7A	z
16	10	Ctrl P	DLE	38	26		&	60	3C	<	81	51	Q	102	66	f	123	7B	{
17	11	Ctrl Q	DC1	39	27		'	61	3D	=	82	52	R	103	67	g	124	7C	
18	12	Ctrl R	DC2	40	28		(62	3E	>	83	53	S	104	68	h	125	7D	}
19	13	Ctrl S	DC3	41	29)	63	3F	?	84	54	T	105	69	i	126	7E	~
20	14	Ctrl T	DC4	42	2A		*	64	40	@	85	55	U	106	6A	j	127	7F	DEL
21	15	Ctrl U	NAK	43	2B		+												