



ibaPDA-Interface-VIP-TCP/UDP

Datenschnittstelle TCP/UDP für VIP-Protokoll

Handbuch
Ausgabe 2.10

Messsysteme für Industrie und Energie
www.iba-ag.com

Hersteller

iba AG
Königswarterstr. 44
90762 Fürth
Deutschland

Kontakte

Zentrale	+49 911 97282-0
Telefax	+49 911 97282-33
Support	+49 911 97282-14
Technik	+49 911 97282-13
E-Mail	iba@iba-ag.com
Web	www.iba-ag.com

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts sind nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz.

© iba AG 2022, alle Rechte vorbehalten.

Der Inhalt dieser Druckschrift wurde auf Übereinstimmung mit der beschriebenen Hard- und Software überprüft. Dennoch können Abweichungen nicht ausgeschlossen werden, so dass für die vollständige Übereinstimmung keine Garantie übernommen werden kann. Die Angaben in dieser Druckschrift werden jedoch regelmäßig aktualisiert. Notwendige Korrekturen sind in den nachfolgenden Auflagen enthalten oder können über das Internet heruntergeladen werden.

Die aktuelle Version liegt auf unserer Website www.iba-ag.com zum Download bereit.

Version	Datum	Revision - Kapitel / Seite	Autor	Version SW
2.10	03-2022	Fehlerbehebung, Nagle-Algorithmus (Kap. 5.1.2)	RM/IP	7.2.2

Windows® ist eine Marke und eingetragenes Warenzeichen der Microsoft Corporation. Andere in diesem Handbuch erwähnte Produkt- und Firmennamen können Marken oder Handelsnamen der jeweiligen Eigentümer sein.

Inhalt

1	Zu diesem Handbuch.....	5
1.1	Zielgruppe und Vorkenntnisse.....	5
1.2	Schreibweisen.....	5
1.3	Verwendete Symbole.....	6
2	Systemvoraussetzungen	7
3	Datenschnittstelle TCP/UDP für VIP	8
3.1	Allgemeine Informationen.....	8
3.1.1	Was ist VIP?	8
3.1.2	Was ist TCP/IP und UDP?	8
3.1.3	Wie arbeitet VIP?.....	9
3.2	Kommunikation zwischen ibaPDA und ABB (Controller)	11
3.3	Datenstrukturen	14
3.3.1	VIP_32_integer: 32 Integer Werte + 32 Binärwerte	14
3.3.2	VIP_8_real: 8 Real Werte + 32 Binärwerte	14
3.3.3	VIP_16_real: 16 Real Werte + 32 Binärwerte	15
3.3.4	VIP_32_real: 32 Real Werte + 32 Binärwerte	15
3.3.5	VIP_Allgemein: max. 4096 Bytes	15
3.4	Projektierungshinweise	16
3.5	Konfiguration & Projektierung ibaPDA	17
3.5.1	Allgemeine Einstellungen	17
3.5.2	Allgemeine Einstellungen der Schnittstelle.....	18
3.5.3	Allgemeine Moduleinstellungen.....	19
3.5.4	Generelle Signalkonfiguration	20
3.5.5	Modultyp Integer.....	21
3.5.6	Modultyp Real	21
3.5.7	Modultyp Allgemein	22
3.5.8	Moduldiagnose.....	23
4	Diagnose	24
4.1	Überprüfung der Lizenz	24
4.2	Sichtbarkeit der Schnittstelle	24

4.3	Protokolldateien	25
4.4	Verbindungsdiagnose mittels PING	26
4.5	Überprüfung der Verbindung	27
5	Anhang	29
5.1	Fehlerbehebung.....	29
5.1.1	Probleme mit TCP-Performance oder unbrauchbaren Daten durch Verwendung des Delayed ACK-Mechanismus	29
5.1.2	Unbrauchbare TCP-Daten als Folge des Nagle-Algorithmus	31
6	Support und Kontakt	33

1 Zu diesem Handbuch

Diese Dokumentation beschreibt die Funktion und Anwendung der Software-Schnittstelle *ibaPDA-Interface-VIP-TCP/UDP*.

Diese Dokumentation ist eine Ergänzung zum *ibaPDA*-Handbuch. Informationen über alle weiteren Eigenschaften und Funktionen von *ibaPDA* finden Sie im *ibaPDA*-Handbuch bzw. in der Online-Hilfe.

1.1 Zielgruppe und Vorkenntnisse

Diese Dokumentation wendet sich an ausgebildete Fachkräfte, die mit dem Umgang mit elektrischen und elektronischen Baugruppen sowie der Kommunikations- und Messtechnik vertraut sind. Als Fachkraft gilt, wer auf Grund seiner fachlichen Ausbildung, Kenntnisse und Erfahrungen sowie Kenntnis der einschlägigen Bestimmungen die ihm übertragenen Arbeiten beurteilen und mögliche Gefahren erkennen kann.

Im Besonderen wendet sich diese Dokumentation an Personen, die mit Projektierung, Test, Inbetriebnahme oder Instandhaltung von Speicherprogrammierbaren Steuerungen der unterstützten Fabrikate befasst sind. Für den Umgang mit *ibaPDA-Interface-VIP-TCP/UDP* sind folgende Vorkenntnisse erforderlich bzw. hilfreich:

- Betriebssystem Windows
- Grundkenntnisse *ibaPDA*
- Kenntnis von Projektierung und Betrieb des betreffenden Messgeräts/-systems

1.2 Schreibweisen

In dieser Dokumentation werden folgende Schreibweisen verwendet:

Aktion	Schreibweise
Menübefehle	Menü <i>Funktionsplan</i>
Aufruf von Menübefehlen	<i>Schritt 1 – Schritt 2 – Schritt 3 – Schritt x</i> Beispiel: Wählen Sie Menü <i>Funktionsplan – Hinzufügen – Neuer Funktionsblock</i>
Tastaturtasten	<Tastename> Beispiel: <Alt>; <F1>
Tastaturtasten gleichzeitig drücken	<Tastename> + <Tastename> Beispiel: <Alt> + <Strg>
Grafische Tasten (Buttons)	<Tastename> Beispiel: <OK>; <Abbrechen>
Dateinamen, Pfade	"Dateiname", "Pfad" Beispiel: "Test.doc"

1.3 Verwendete Symbole

Wenn in dieser Dokumentation Sicherheitshinweise oder andere Hinweise verwendet werden, dann bedeuten diese:

Gefahr!



Wenn Sie diesen Sicherheitshinweis nicht beachten, dann droht die unmittelbare Gefahr des Todes oder der schweren Körperverletzung!

- Beachten Sie die angegebenen Maßnahmen.

Warnung!



Wenn Sie diesen Sicherheitshinweis nicht beachten, dann droht die mögliche Gefahr des Todes oder schwerer Körperverletzung!

- Beachten Sie die angegebenen Maßnahmen.

Vorsicht!



Wenn Sie diesen Sicherheitshinweis nicht beachten, dann droht die mögliche Gefahr der Körperverletzung oder des Sachschadens!

- Beachten Sie die angegebenen Maßnahmen.

Hinweis



Hinweis, wenn es etwas Besonderes zu beachten gibt, wie z. B. Ausnahmen von der Regel usw.

Tipp



Tipp oder Beispiel als hilfreicher Hinweis oder Griff in die Trickkiste, um sich die Arbeit ein wenig zu erleichtern.

Andere Dokumentation



Verweis auf ergänzende Dokumentation oder weiterführende Literatur.

2 Systemvoraussetzungen

Folgende Systemvoraussetzungen sind für die Verwendung der Datenschnittstelle TCP/UDP für VIP-Protokoll erforderlich:

- *ibaPDA* v7.0.0 oder höher
- Lizenz für *ibaPDA-Interface-VIP-TCP/UDP*
- Netzwerkanschluss 10/100 Mbit
- ABB Controller mit TCP/IP Kommunikationsinterface, z. B. CI861

Sonstige Voraussetzungen an die eingesetzte PC-Hardware und die unterstützten Betriebssysteme entnehmen Sie der *ibaPDA*-Dokumentation.

Hinweis



Es wird empfohlen die TCP/IP- bzw. UDP-Kommunikation auf einem separaten Netzwerksegment durchzuführen, um eine gegenseitige Beeinflussung durch sonstige Netzwerkkomponenten auszuschließen

Systemeinschränkungen

- Unterschiedliche Behandlung des TCP/IP-Acknowledge
siehe ➔ *Probleme mit TCP-Performance oder unbrauchbaren Daten durch Verwendung des Delayed ACK-Mechanismus*, Seite 29 (alle *ibaPDA*-Versionen).

Lizenzen

Bestell-Nr.	Produktbezeichnung	Beschreibung
31.001065	ibaPDA-Interface-VIP-TCP/UDP	Erweiterungslizenz für ein <i>ibaPDA</i> -System um eine TCP/IP und UDP/IP Schnittstelle Anzahl der Verbindungen: 64
31.101065	one-step-up-Interface-VIP-TCP/UDP	Erweiterungslizenz für eine vorhandene <i>ibaPDA-Interface-VIP-TCP/UDP</i> Schnittstelle um 64 weitere VIP-TCP/UDP-Verbindungen, maximal 3 Erweiterungslizenzen zulässig

3 Datenschnittstelle TCP/UDP für VIP

3.1 Allgemeine Informationen

3.1.1 Was ist VIP?

Das Vendor Internet Protocol (VIP) dient vorrangig der Kommunikation zwischen ABB AC450RMC-Controllern sowie anderen Computern und Systemen, die nicht von ABB stammen aber dieses Protokoll verarbeiten können.

Die VIP-Funktionalität setzt auf dem Transmission Control Protocol (TCP), dem User Datagram Protocol (UDP) und dem Internet Protocol (IP) für Ethernet auf.

Im Kontext dieses Handbuchs wird das VIP-Protokoll dazu verwendet, verschiedenste Daten von einem ABB-Controller mit dem Datenerfassungssystem *ibaPDA* zu messen.

Die Auswahl der zu messenden Signale erfolgt durch Rangieren der Werte in Telegrammpuffer, deren Datenstrukturen durch die Modultypen von *ibaPDA* vorgegeben sind. Die Telegramme werden mit Standard-Sendebausteinen an den *ibaPDA*-PC gesendet.

Im *ibaPDA-Interface-VIP-TCP/UDP* sind drei Modultypen definiert:

- Integer: 32 Analogwerte (Integer) und 32 Binärsignale
- Real: 8, 16 oder 32 Analogwerte (Real) und 32 Binärsignale
- Allgemein: beliebige Datenstruktur mit maximaler Länge von 40961 Bytes

Jedes Modul ist einer Verbindung zugeordnet. Auf *ibaPDA*-Seite können max. 256 Verbindungen erstellt werden. Auf der ABB-Seite hängt die max. Anzahl der Verbindungen vom CPU-Typ ab.

Die folgenden ABB-Controller können über das VIP-Protokoll mit *ibaPDA* kommunizieren:

- AC450 RMC
- AC800M
- AC80
- AC800 PEC

Der Hauptvorteil dieser Art der Messung liegt darin, dass keine spezielle Hardware benötigt wird, wenn bereits ein Ethernet-Anschluss in der Steuerung existiert.

3.1.2 Was ist TCP/IP und UDP?

TCP/IP ist ein Datentransportprotokoll, welches Daten von beliebigen Applikationen oder Teilnehmern beinhalten kann. Dies bedeutet, dass TCP/IP der Weg ist, mit dem Daten mittels eines Standardprotokolls auf standardisierten Schnittstellen (z. B. Ethernet Netzwerkkarten, die von einer Vielzahl von Anbietern verfügbar sind) übertragen werden können.

Auch wenn ein TCP/IP-Treiber Daten senden und empfangen kann, muss der Nutzdateninhalt vom Anwender interpretiert werden. Nur die Nutzdaten haben für den Anwender wirklich eine Bedeutung.

TCP/IP-Paket	
Kopf (header)	Daten

Der TCP/IP-Nachrichtenkopf (header) beinhaltet im Normalfall neben Steuerinformationen auch die Quell- und Zieladresse der Nachricht. Der Datenteil (data) beinhaltet Daten mit einer speziellen Strukturierung, so dass eine *ibaPDA*-Applikation diese interpretieren kann.

Für die Verbindung zwischen ABB und *ibaPDA* ist sowohl das Verständnis der Datenstruktur wichtig als auch die Übertragungsreihenfolge und der Nachrichtenkopf. Damit ist es möglich, einen speziellen TCP/IP-Treiber zu schreiben, welcher die Datenpakete ausliest und dem *ibaPDA*-Interface zur Aufzeichnung und Analyse zur Verfügung stellt. Mit dem TCP/IP-Treiber arbeitet *ibaPDA* wie ein Verbindungsserver, die Automatisierungsgeräte arbeiten als Clients. Dies bedeutet, dass *ibaPDA* gesendete Informationen und Verbindungsanforderungen abhört.

ibaPDA kann die ABB TCP/IP-Verbindung mit einer maximalen Abtastezeit von 5 ms abarbeiten. Allerdings kann es zusätzliche Restriktionen auf der Sendeseite geben. Diese werden später weiter präzisiert.

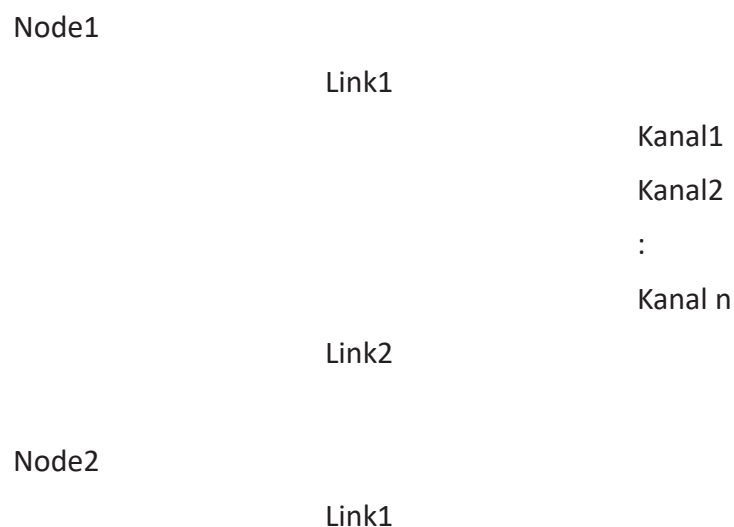
Das Transmission Control Protocol, kurz TCP, ist ein verbindungsorientiertes Protokoll und soll maßgeblich Datenverluste verhindern, Dateien und Datenströme aufteilen und Datenpakete Anwendungen zuordnen können.

Das User Datagram Protocol, kurz UDP, ist ein verbindungsloses Transport-Protokoll und arbeitet auf der Schicht 4, der Transportschicht, des OSI-Schichtenmodells. Es hat damit eine vergleichbare Aufgabe, wie das verbindungsorientierte TCP. Allerdings arbeitet es verbindungslos und damit unsicher. Das bedeutet, der Absender weiß nicht, ob seine verschickten Datenpakete angekommen sind. Während TCP Bestätigungen beim Datenempfang sendet, verzichtet UDP darauf. Das hat den Vorteil, dass der Paket-Header viel kleiner ist und die Übertragungsstrecke keine Bestätigungen übertragen muss. Prinzipiell ist damit eine etwas höhere Datenrate möglich.

3.1.3 Wie arbeitet VIP?

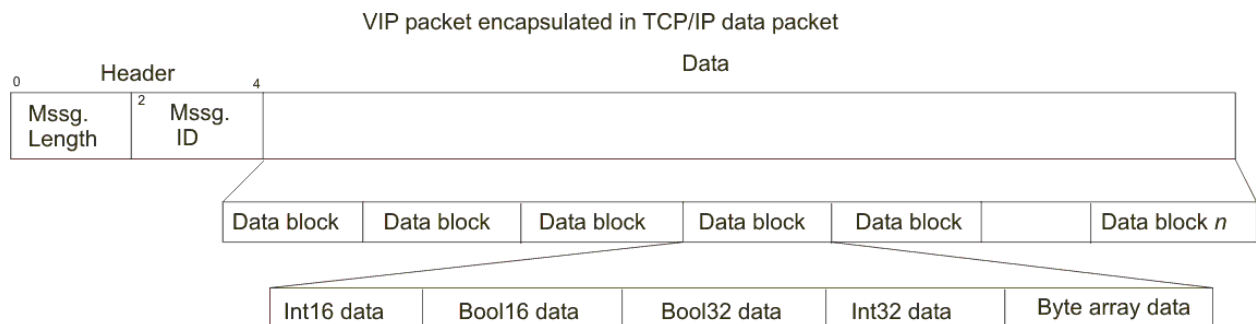
Die Topologie der Kommunikation wird in nachstehendem Bild dargestellt:

Network



Dies bedeutet, dass jede Station am Ethernet mit einer eindeutigen Knotennummer (Node 1..99) und einer eindeutigen IP-Adresse versehen werden muss. Jeder Node kann zu maximal fünf verschiedenen Datenquellen Verbindungen herstellen. Jede Station (Controller) kann als Client oder Server konfiguriert werden (Aufrufe VIP-NETW, VIP-NODE, VIP-LINK, siehe auch VIP-Benutzerhandbuch, Seiten 43, 45 und 39).

Die innerhalb eines TCP/IP- bzw. UDP-Paketes eingeschlossene VIP-Paketstruktur wird in der nachstehenden Abbildung TCP/IP-VIP-Telegrammaufbau erläutert:



Die Maximallänge für TCP/IP-Nachrichtenblöcke beträgt 65535 Byte. Im Normalfall enthält der Bereich Message Length die Anzahl der Bytes des kompletten Datenpaketes, inkl. Header. Message ID ist konfigurierbar und kann jeden beliebigen Wert annehmen, den der Benutzer benötigt. Danach folgt eine beliebige Anzahl von Datenblöcken, die immer den gleichen Aufbau haben. Die ersten Daten sind immer INT16, dann folgen Bool16 usw., bis hin zu den Byte Arrays am Ende eines jeden Blockes. Für fehlende Datentypen werden keine Leerzeichen eingefügt. Die Gesamtstruktur des Datenblocks muss daher für beide Seiten (Sender und Empfänger) verbindlich festgelegt werden.

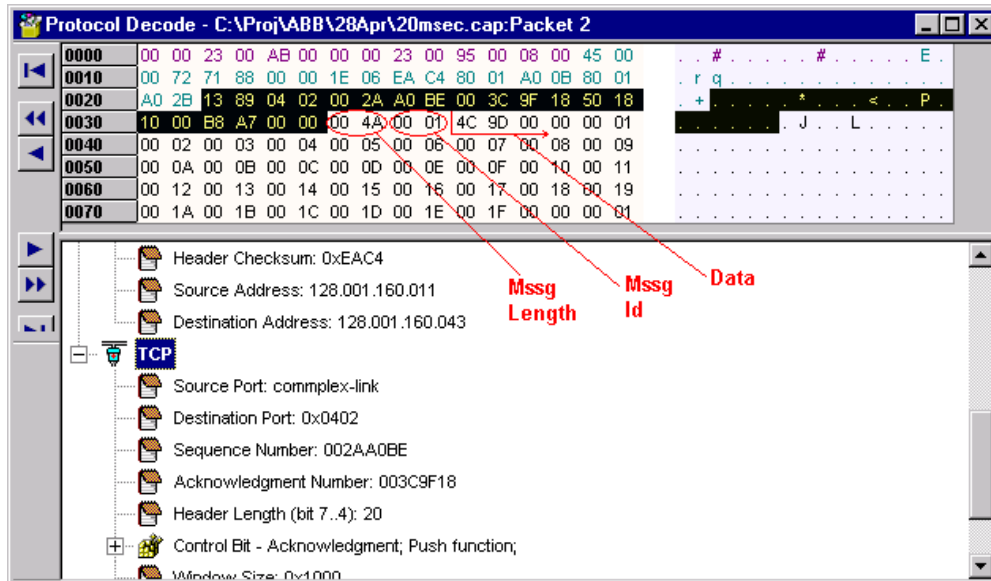
Beispiel

Angenommen der Benutzer sendet nur zwei INT16, ein Bool32 und einen INT32 Wert mit der Message ID 2, dann sieht die Datenstruktur wie folgt aus:

| 16 | 2 | Int16 Val1 | Int16 Val2 | Bool32 Val | Int32 Val |

Dies bedeutet, dass die Länge in Bytes 16 beträgt: 2 für Message Length, 2 für Message ID, 2x2 für die beiden INT16 Werte, und je 4 für Bool32 und INT32.

Überprüfung des Datenpakets: Es ist zu beachten, dass die Notation in Hex-Format vorliegt und die einzelnen Bytes durch zwei Zeichen repräsentiert werden, wobei die Bytes jeweils durch einen schmalen Leerbereich getrennt sind.



Das komplette TCP/IP-Paket ist in unterschiedlichen Farben, je nach Zugehörigkeit dargestellt. Am Ende des in Abbildung TCP/IP-VIP Inhalt des Datenpakets schwarz markierten Bereiches beginnt der Datenbereich des VIP-Paketes. Die ersten beiden Bytes 00 4A stellen die Nachrichtenlänge (Message Length) in Bytes dar, sie beträgt demzufolge 74 Byte. Die beiden nächsten Bytes 00 und 01 repräsentieren die Message-ID, also hier 1. Danach folgt der 2-Byte INT16-Wert 4C 9D (19613 dezimal). Anschließend folgen 32 INT16-Werte mit den Inhalten 00 00 bis 00 1F (0 bis 31 dezimal). Am Ende stehen 4 Bytes mit dem INT32-Wert 1. Zusammen demnach 74 Byte Gesamtlänge.

3.2 Kommunikation zwischen ibaPDA und ABB (Controller)

iba AG hat einen TCP/IP-Treiber entwickelt, um Daten aus dem ABB VIP-Paket zu entnehmen und diese mit *ibaPDA* anzuzeigen.

Unterstützte Verbindungen in *ibaPDA*:

- Mit *ibaPDA* ab Version 6.14.0 sind für die Schnittstelle TCP/IP-VIP insgesamt bis zu 64 Module der Typen Integer, Real und/oder „Generisch“ konfigurierbar. Jedes Integer- oder Real-Modul kann bis zu 32 Analog- und Digitalsignale enthalten. Generische Module können bis zu 1000 Analog- und Digitalsignale enthalten.

- Ab *ibaPDA*-Version 6.31.0 werden wie bisher bis zu 64 Module (Verbindungen) pro Interface unterstützt. Jedoch können insgesamt 4 Lizenzen in *ibaPDA* verwendet werden, damit sind bis zu 256 Verbindungen möglich.
Die maximale Länge der Nachricht ist auf 4096 Bytes begrenzt. Generische Module können wie bisher bis zu 1000 Analog- und 1000 Digitalsignale enthalten.
- Ab *ibaPDA*-Version 6.33.2 wird auch das UDP-Protokoll unterstützt. Der Sender kann die Daten über TCP/IP oder UDP senden. Die Kommunikationsparameter und die Datenstrukturen sind identisch.

ibaPDA arbeitet als Server, der auf Clients hört, welche einen gültigen Connect-Request und anschließend Daten schicken (siehe auch Abbildung *Kommunikationsprinzip ibaPDA*). Jede TCP/IP- bzw. UDP-Verbindung kann im Diagnosefenster von *ibaPDA* bzw. im I/O-Manager eingesehen werden. *ibaPDA* ist so voreingestellt, dass der Port 5001 auf ABB VIP-Informationen abgehört wird. Deshalb muss dieser immer als Zielport eingetragen werden. Es ist wichtig festzuhalten, dass die IP-Adresse nur einmalig vorkommen darf, aber mehr als eine Verbindung pro IP-Adresse möglich ist. Jede Verbindung entspricht einem Modul in *ibaPDA*, weshalb eine eindeutige Message ID (entspricht dem *ibaPDA* Modulindex) benötigt wird.

Die Message ID muss im gesamten System einmalig sein, auch wenn mehrere ABB-Controller genutzt werden.

Dies bedeutet, dass der ABB VIP-Controller bis zu 5 solcher Links aufbauen und damit jeder Controller bis zu 5 *ibaPDA*-Module belegen kann. Jedes Modul benötigt hierzu eine einmalige Quellportnummer, welche automatisch vom System vergeben wird, wenn der Benutzer für jedes Modul einen neuen Link konfiguriert.

Modulindex:

Der Modulindex ist die Kennung für die Zuordnung des Datensatzes zu dem Interface-Modul in *ibaPDA*. In diesem Index ist auch der Modultyp verschlüsselt: Der Index wird durch eine laufende Nummer 00....63 und einem dem Modultyp und der Lizenz entsprechenden Offset gebildet.

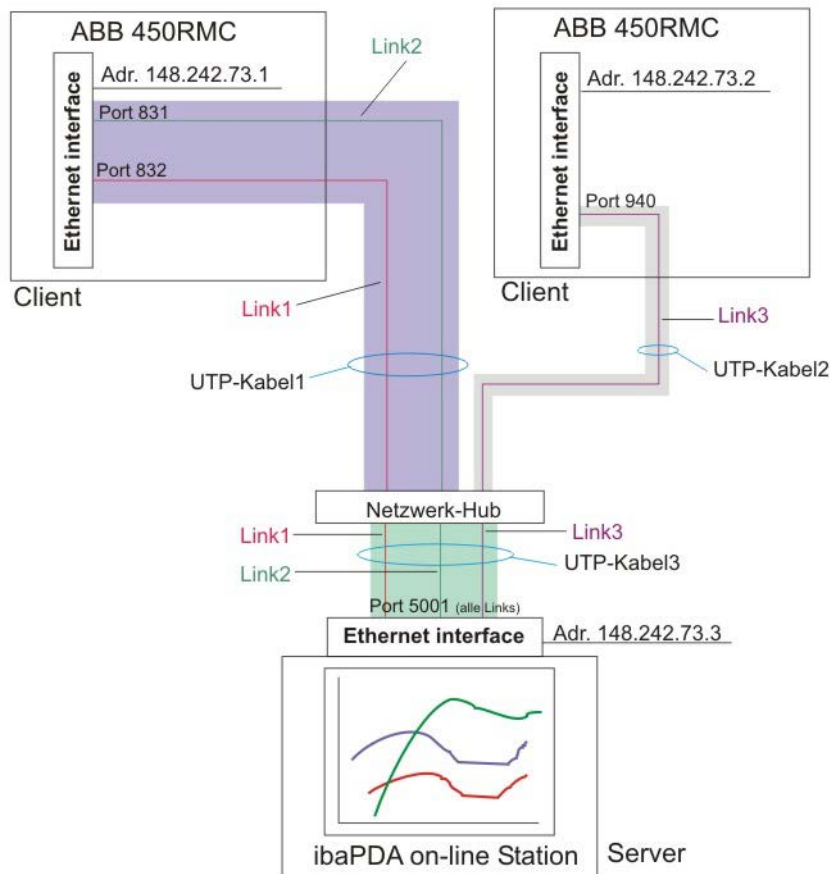
Modultyp	1. Lizenz	2. Lizenz	3. Lizenz	4. Lizenz
Integer	0-63	1000-1063	2000-2063	3000-3063
Real	100-163	1100-1163	2100-2163	3100-3163
Allgemein	200-263	1200-1263	2200-2263	3200-3263

Der Modulindex entspricht dem Index in der *ibaPDA*-Moduleinstellung. Der Wert darf während der Datenübertragung nicht verändert werden.

Um *ibaPDA* darüber zu informieren, welches dieser Module adressiert werden soll, muss in der Message ID der Modulindex hinterlegt werden. Diese ID muss auf der ABB VIP-Seite eingetragen werden.

Sequenzzähler:

ibaPDA ist darüber hinaus in der Lage, Kommunikationsfehler zu erkennen. Für diese Funktion hat iba festgelegt, dass die ersten beiden Datenbytes einen Aufwärtzähler bilden sollen, welcher bei jedem Sendezyklus um 1 inkrementiert werden muss. Dieser Zähler muss auf der ABB Sendeseite der erste INT16 Wert sein. Bei Überlauf muss der Zähler von 32767 auf -32768 springen (0x7FFF à 0x8000) bzw. von 65535 auf 0 (0xFFFF à 0x0000).



Zu Abbildung *Kommunikationsprinzip ibaPDA* ist anzumerken, dass hier zwei ABB AC450-Controller als Clients agieren, wobei jeder über sein eigenes Ethernet-Modul verfügt, welches jeweils über ein Kabel mit dem Netzwerk-Hub verbunden ist. Es existiert ein weiteres Kabel, welches *ibaPDA* mit dem Hub verbindet (die Kabel sind hier mit unterschiedlicher Dicke und Färbung dargestellt). Damit besteht das Netzwerk aus drei Kommunikationspartnern mit eindeutigen IP-Adressen.

Anmerkung: Ob die Verbindungen als UTP, ThinNetCoax oder optisch auszuführen sind, ist für die jeweilige Applikation festzulegen. Da drei Teilnehmer vorhanden sind, bedeutet dies mindestens drei verwendete Module innerhalb von *ibaPDA* (hier angedeutet mit drei Kurven in verschiedenen Farben). Zu beachten ist, dass jeder Teilnehmer über eine zufällige Quellportnummer verfügt (940, 831 und 832), während sie alle dieselbe Zielportnummer (Target Port No.) 5001 haben.

Hinweis

Wichtig für die Verwendung eines Cross-over-Kabels zwischen dem ABB-Controller und *ibaPDA*:

Da die meisten ABB-Controller mit 10MBit/s im Halbduplex-Modus kommunizieren, wird empfohlen, die Ethernet-Schnittstelle des *ibaPDA*-PCs auch auf 10 MBit/s, halbduplex, einzustellen, wenn Sie ein Cross-over-Kabel verwenden.

Wenn Sie die Übertragungsgeschwindigkeit nicht auf halbduplex setzen können, dann verwenden Sie einen Netzwerk-Hub oder -Switch zwischen dem Controller und *ibaPDA*.

3.3 Datenstrukturen

Da *ibaPDA* die exakte Lage und den Typ eines jeden Wertes kennen muss, hat iba folgende Telegrammstrukturen für die verschiedenen Datenformate festgelegt:

Pro Nachricht kann nur eine Telegrammstruktur verwendet werden.

3.3.1 VIP_32_integer: 32 Integer Werte + 32 Binärwerte

rel. #	Bytes	C - Type	Beschreibung	Bemerkung
00	2	short int	Message Length	Festgelegt auf 74 Byte
02	2	short int	Message ID	Modulindex in <i>ibaPDA</i> , siehe ↗ <i>Kommunikation zwischen ibaPDA und ABB (Controller)</i> , Seite 11
04	2	unsigned short	1. INT16-Signal	Sequenzzähler: Bei jedem Sendezyklus inkrementieren! Bei Überlauf 65535 -> 0
06	64	short int	32 Analogwerte	
70	4	long int	32 Digitalwerte	

3.3.2 VIP_8_real: 8 Real Werte + 32 Binärwerte

rel. #	Bytes	C - Type	Beschreibung	Bemerkung
00	2	short int	Message Length	Festgelegt auf 42 Byte
02	2	short int	Message ID	Modulindex in <i>ibaPDA</i> , siehe ↗ <i>Kommunikation zwischen ibaPDA und ABB (Controller)</i> , Seite 11
04	2	unsigned short	1. INT16-Signal	Sequenzzähler: Bei jedem Sendezyklus inkrementieren! Bei Überlauf 65535 -> 0

rel. #	Bytes	C - Type	Beschreibung	Bemerkung
06	4	long int	32 Digitalwerte	
10	32	float	8 Analogwerte	IEEE Format

3.3.3 VIP_16_real: 16 Real Werte + 32 Binärwerte

rel. #	Bytes	C - Type	Beschreibung	Bemerkung
00	2	short int	Message Length	Festgelegt auf 74 Byte
02	2	short int	Message ID	Modulindex in <i>ibaPDA</i> , siehe ↗ <i>Kommunikation zwischen ibaPDA und ABB (Controller)</i> , Seite 11
04	2	unsigned short	1. INT16-Signal	Sequenzzähler: Bei jedem Sendezyklus inkrementieren! Bei Überlauf 65535 -> 0
06	4	long int	32 Digitalwerte	
10	64	float	16 Analogwerte	IEEE Format

3.3.4 VIP_32_real: 32 Real Werte + 32 Binärwerte

rel. #	Bytes	C - Type	Beschreibung	Bemerkung
00	2	short int	Message Length	Festgelegt auf 138 Byte
02	2	short int	Message ID	Modulindex in <i>ibaPDA</i> , siehe ↗ <i>Kommunikation zwischen ibaPDA und ABB (Controller)</i> , Seite 11
04	2	unsigned short	1. INT16-Signal	Sequenzzähler: Bei jedem Sendezyklus inkrementieren! Bei Überlauf 65535 -> 0
06	4	long int	32 Digitalwerte	
10	128	float	32 Analogwerte	IEEE Format

3.3.5 VIP_Allgemein: max. 4096 Bytes

rel. #	Bytes	C - Type	Beschreibung	Bemerkung
00	2	short int	Message Length	Maximum 4102 Bytes
02	2	short int	Message ID	Modulindex in <i>ibaPDA</i> , siehe ↗ <i>Kommunikation zwischen ibaPDA und ABB (Controller)</i> , Seite 11

rel. #	Bytes	C - Type	Beschreibung	Bemerkung
04	2	unsigned short	1. INT16-Signal	Sequenzzähler: Bei jedem Sendezyklus inkrementieren! Bei Überlauf 65535 -> 0
06	max. 4096	byte, int, word, double int, double word, float, string auch gemischt	Analog- / Digital-Werte	Analogwerte unterstützen Textsignale

3.4 Projektierungshinweise

Mit ABB-Controllern, die das TCP/IP- oder das UDP-VIP-Protokoll unterstützen, können bis zu 64 Verbindungen genutzt werden, um Daten an *ibaPDA* zu senden, wobei in *ibaPDA* bis zu 64 Module konfiguriert werden können.

Ab *ibaPDA*-Version 6.31.0 werden bis zu 4 TCPIP-VIP-Interfaces unterstützt, damit sind bis zu 256 Verbindungen möglich.

Es muss sichergestellt sein, dass jede Message ID (Modulindex) einmalig ist und im TCP/IP- bzw. UDP-Telegramm zu *ibaPDA* enthalten ist und damit jeweils genau ein Modul in *ibaPDA* adressiert.

ibaPDA reagiert in Bezug auf ABB VIP auf den Port 5001.

Hinweis



Ab *ibaPDA*-Version 6.33.2 kann in *ibaPDA* auch ein anderer Port eingestellt werden.

ABB Controller dürfen also nur diesen Port benutzen. Bei der Nutzung von Technostrings muss daher auf eine andere verfügbare Portnummer umgeschaltet werden.

Bei jedem Link muss das erste INT16-Signal als Aufwärtszähler projiziert sein, welcher mit jedem Sendezyklus inkrementiert werden muss.

Die Länge der Nachricht (Telegrammlänge / Message Length) hängt vom gewählten Telegramm ab (32 INT, 8-, 16- oder 32 REAL oder Allgemein). Die ABB Controller müssen sendeseitig die Layoutanforderungen erfüllen, da sonst u. U. die Meldung "Unvollständige Datenpakete" im *ibaPDA*-Diagnosefenster (I/O-Manager) erscheint.

Es sollten nur *ibaPDA* und der/die ABB-Controller am Bus angeschlossen sein.

Um die Zeitbasis (Grundtakt) für *ibaPDA* zu generieren, ist in jedem Fall die Installation einer ibaFOB-Karte als Interruptquelle zu empfehlen.

Hinweis



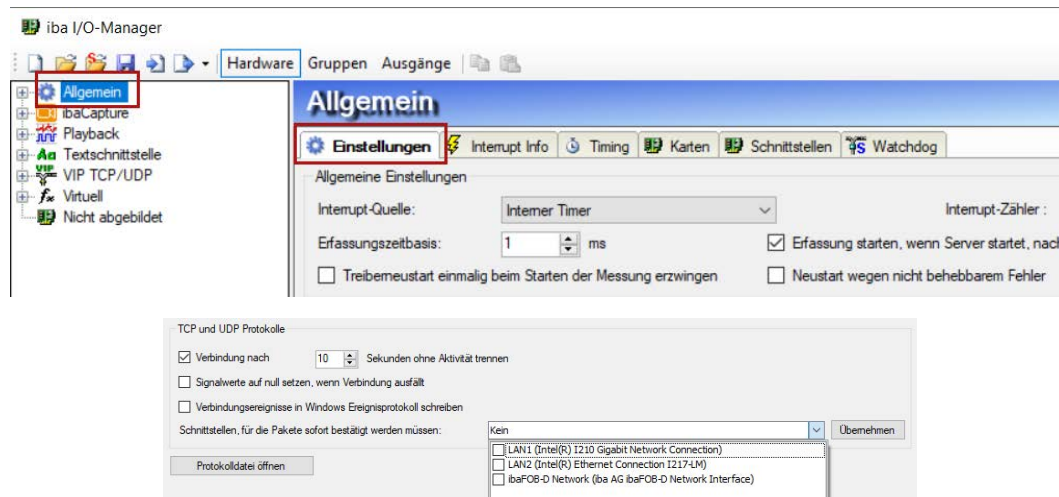
Innerhalb des VIP-Netzwerkes sollten neben *ibaPDA* und den beteiligten Controllern keine weiteren Teilnehmer angeschlossen sein, um eine hohe Dynamik des Systems sicherzustellen.

3.5 Konfiguration & Projektierung ibaPDA

Nachfolgend ist die Projektierung in *ibaPDA* beschrieben. Bei Vorliegen aller Systemvoraussetzungen wird im Signalbaum die Schnittstelle „VIP TCP/UDP“ angezeigt.

3.5.1 Allgemeine Einstellungen

Die „Totmann-Timeout“-Konfiguration erfolgt für alle von *ibaPDA* unterstützten TCP/IP- und UDP-Protokolle gemeinsam.



Verbindung nach x Sekunden ohne Aktivität trennen

Verhalten und Timeout-Zeit ist vorgebar

Signalwerte auf null setzen, wenn Verbindung ausfällt

Wenn deaktiviert, bleibt der zuletzt gelesene Wert erhalten

Verbindungsereignisse in Windows Ereignisprotokoll schreiben

Aktuelle Ereignisse werden in Windows protokolliert

Schnittstellen, für die Pakete sofort bestätigt werden müssen

Auswahl der erforderlichen Schnittstellen

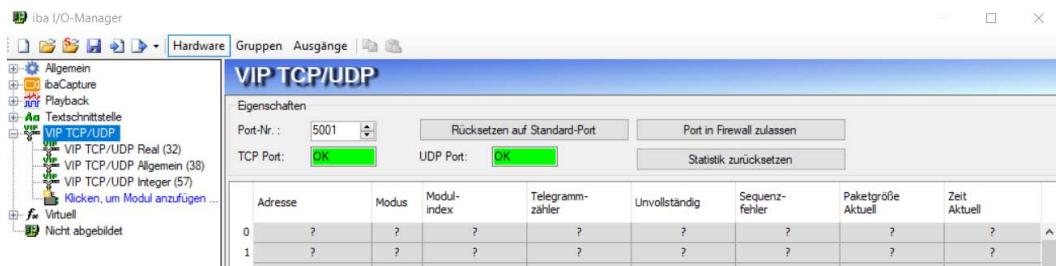
Hinweis



Ist *ibaPDA* der aktive Partner (Client), baut *ibaPDA* nach wenigen Sekunden die Verbindung wieder auf, um dem passiven Partner die Möglichkeit zu geben, wieder Daten zu senden.

3.5.2 Allgemeine Einstellungen der Schnittstelle

Die Schnittstelle selbst hat folgende Funktionen und Konfigurationsmöglichkeiten:



Port-Nr.

Verwendeter Port PC-seitig. Die Portnummer muss in der VIP-Verbindungsprojektion identisch verwendet werden. Ab *ibaPDA*-Version 6.33.2 kann die Portnummer hier frei gewählt werden.

<Rücksetzen auf Standard-Port>

Die Portnummer 5001 wird eingestellt.

Port in Firewall zulassen

Bei der Installation von *ibaPDA* werden die Standard-Portnummern der verwendeten Protokolle automatisch in der Firewall eingetragen. Wird die Portnummer hier verändert oder wurde das Interface nachträglich freigeschaltet, ist es notwendig hier diesen Port in der Firewall zuzulassen.

TCP Port / UDP Port

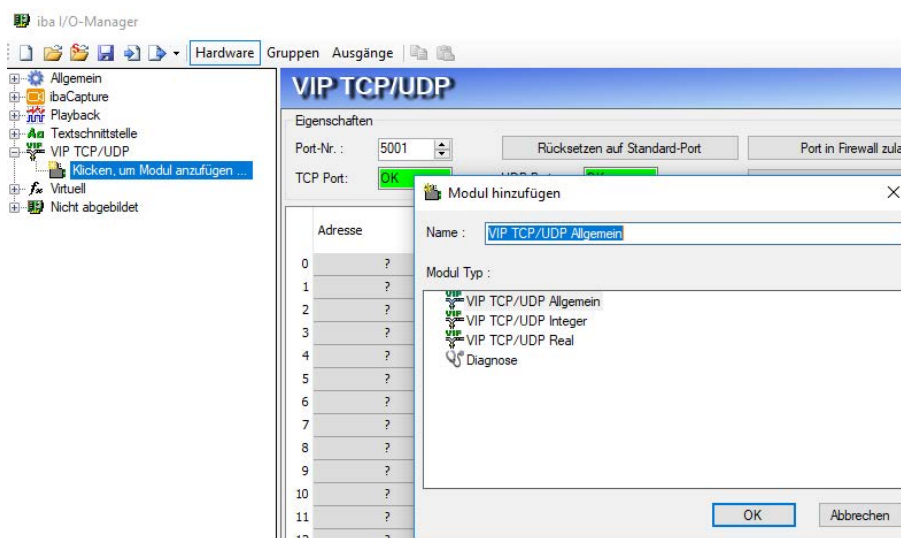
Hier erfolgt die Anzeige OK wenn der Socket auf diesem Port geöffnet werden kann. Die Anzeige FEHLER kommt, wenn Konflikte auftreten, z. B. wenn der Port schon anderweitig belegt ist.

Verbindungstabelle

siehe ➔ *Überprüfung der Verbindung*, Seite 27

Hinzufügen eines Moduls

Fügen Sie ein Modul durch Klicken unter der Schnittstelle und Auswahl des gewünschten Modultyps hinzu.

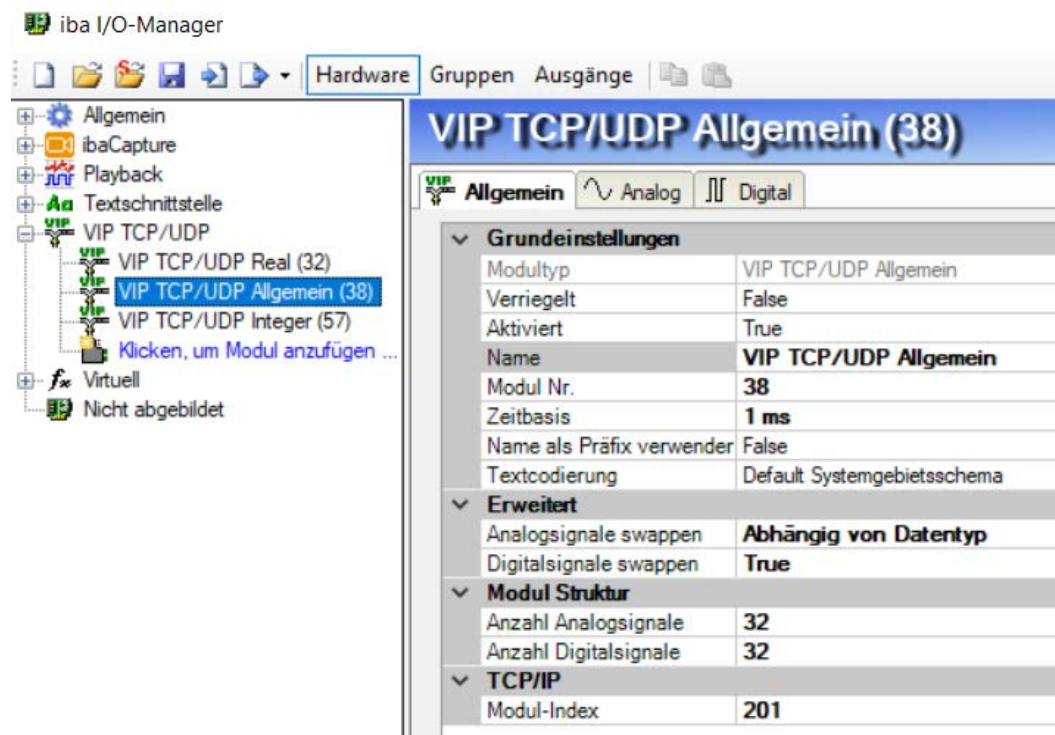


Tipp

Wenn bereits eine TCP/IP- oder UDP-Verbindung besteht, klicken Sie mit der rechten Maustaste auf das Interface und wählen Sie Autom. Erkennung. Dann werden für alle vorhandenen Verbindungen automatisch die richtigen Module angelegt.

3.5.3 Allgemeine Moduleinstellungen

Um ein Modul zu konfigurieren, markieren Sie es in der Baumstruktur und nehmen die Einstellungen im folgenden Dialog vor:



Grundeinstellungen

Modultyp (nur Anzeige)

Zeigt den Typ des aktuellen Moduls an.

Verriegelt

Ein Modul kann verriegelt werden, um ein versehentliches oder unautorisiertes Ändern der Moduleinstellungen zu verhindern.

Aktiviert

Deaktivierte Module werden von der Signalerfassung ausgeschlossen.

Name

Hier ist der Klartextname als Modulbezeichnung einzutragen.

Modul Nr.

Interne Referenznummer des Moduls. Diese Nummer bestimmt die Reihenfolge der Module im Signalbaum von *ibaPDA-Client* und *ibaAnalyzer*.

Zeitbasis

Alle Signale dieses Moduls werden mit dieser Zeitbasis erfasst.

Name als Präfix verwenden

Stellt den Modulnamen den Signalnamen voran.

Textcodierung

Hier können Sie die Zeichensatztabelle auswählen, nach der die empfangenen Textdaten interpretiert werden sollen.

Erweitert**Analogsignale swappen**

Möglichkeit die Auswertereihenfolge der Byte zu ändern

Digitalsignale swappen

Möglichkeit die Auswertereihenfolge der Byte zu ändern

Modul Struktur**Anzahl Analogsignale/Digitalsignale**

Hier wird die Anzahl der zu messenden analogen und digitalen Signale eingetragen. Die maximale Anzahl ist 1000 (Analogsignale/Digitalsignale).

TCP/IP**Modulindex**

Die Modulindizes werden durch eine laufende Nummer 00....63 und einem dem Modultyp und der Lizenz entsprechenden Offset gebildet.

Siehe dazu ➔ *Kommunikation zwischen ibaPDA und ABB (Controller)*, Seite 11.

Für eine detaillierte Beschreibung der Parameter wird auf das *ibaPDA-Handbuch* verwiesen.

3.5.4 Generelle Signalkonfiguration

Die Auswahl der zu messenden Daten erfolgt auf ABB-Seite durch Rangieren der Signale in die Datenbereiche die zyklisch an *ibaPDA* gesendet werden.

Im I/O-Manager können die Signale mit Namen und Einheit (nur Analogsignale) versehen, sowie aktiv und inaktiv gesetzt werden.

VIP Integer TCP (0)							
<div> <div>VIP</div> <div>Allgemein</div> <div>Analog</div> <div>Digital</div> </div>							
	Name	Einheit	Min	Max	Aktiv	Istwert	+
0	TCP Integer 0		-32768	32767	<input checked="" type="checkbox"/>	0	
1	TCP Integer 1		-32768	32767	<input checked="" type="checkbox"/>	205	
2	TCP Integer 2		-32768	32767	<input checked="" type="checkbox"/>	205	
3	TCP Integer 3		-32768	32767	<input checked="" type="checkbox"/>	205	
4	TCP Integer 4		-32768	32767	<input type="checkbox"/>	205	
5	TCP Integer 5		-32768	32767	<input type="checkbox"/>	205	
6	TCP Integer 6		-32768	32767	<input type="checkbox"/>	205	
7	TCP Integer 7		-32768	32767	<input type="checkbox"/>	205	

VIP Integer TCP (0)							
<div> <div>VIP</div> <div>Allgemein</div> <div>Analog</div> <div>Digital</div> </div>							
	Name				Aktiv	Ist...	+
0	Byte 0 bit 0				<input checked="" type="checkbox"/>	0	
1	Byte 0 bit 1				<input checked="" type="checkbox"/>	0	
2	Byte 0 bit 2				<input checked="" type="checkbox"/>	0	
3	Byte 0 bit 3				<input checked="" type="checkbox"/>	0	
4	Byte 0 bit 4				<input checked="" type="checkbox"/>	0	
5	Byte 0 bit 5				<input checked="" type="checkbox"/>	0	
6	Byte 0 bit 6				<input checked="" type="checkbox"/>	0	

Tipp

Sie können das "automatische Ausfüllen" der Spalte verwenden (siehe *ibaPDA*-Handbuch oder Online-Hilfe).

Andere Dokumentation

Für eine detaillierte Beschreibung der weiteren Möglichkeiten wird auf das *ibaPDA*-Handbuch verwiesen.

3.5.5 Modultyp Integer

Mittels des Moduls Integer können bis zu 32 Analogwerte (Integer) und 32 Binärsignale erfasst werden.

Das Modul hat keine modulspezifischen Einstellungen.

3.5.6 Modultyp Real

Mittels des Moduls Real können bis zu 32 Analogwerte (Real) und 32 Binärsignale erfasst werden.

Folgende Moduleinstellungen sind modulspezifisch:

Anzahl Analogsignale

Die zu erfassende Anzahl der Analogsignale ist in den Stufen 8, 16, 32 konfigurierbar (Die Anzahl der Digitalsignale liegt fest bei 32).

3.5.7 Modultyp Allgemein

Mittels des Moduls *Allgemein* kann eine beliebige Datenstruktur mit maximaler Länge von 4096 Bytes erfasst werden.

Folgende Moduleinstellungen sind modulspezifisch:

Anzahl der Analogsignale

Anzahl der maximal konfigurierbaren Analogsignale. Bei den Analogsignalen werden verschiedene Datentypen unterstützt, inkl. Texte: SINT, BYTE, INT, WORD, DINT, DWORD, FLOAT, DOUBLE, STRING[32].

Anzahl der Digitalsignale

Anzahl der maximal konfigurierbaren Digitalsignale.

Bei der Signalkonfiguration tragen Sie für jede Variable die Adresse, d. h. den Offset im Telegrammpuffer, sowie den Datentyp ein. Achten Sie darauf dass hier ab Nutzdatenanfang ohne Header gezählt wird.

VIP TCP/UDP Allgemein (105)								
<div> <div>VIP TCP/UDP Allgemein</div> <div>Analog</div> <div>Digital</div> </div>								
Name	Einheit	Gain	Offset	Adresse	Datentyp	Aktiv	Istwert	
0		1	0	0x0	INT	<input checked="" type="checkbox"/>	0	
1		1	0	0x2	INT	<input checked="" type="checkbox"/>	0	
2		1	0	0x4	INT	<input checked="" type="checkbox"/>	0	
3		1	0	0x6	INT	<input checked="" type="checkbox"/>	0	
4		1	0	0x8	INT	<input checked="" type="checkbox"/>	0	
5		1	0	0xA	INT	<input checked="" type="checkbox"/>	0	
6		1	0	0xC	INT	<input checked="" type="checkbox"/>	0	
7		1	0	0xE	INT	<input checked="" type="checkbox"/>	0	
8		1	0	0x10	INT	<input checked="" type="checkbox"/>	0	
9		1	0	0x12	INT	<input checked="" type="checkbox"/>	0	
10		1	0	0x14	INT	<input checked="" type="checkbox"/>	0	
11		1	0	0x16	INT	<input checked="" type="checkbox"/>	0	
12		1	0	0x18	INT	<input checked="" type="checkbox"/>	0	

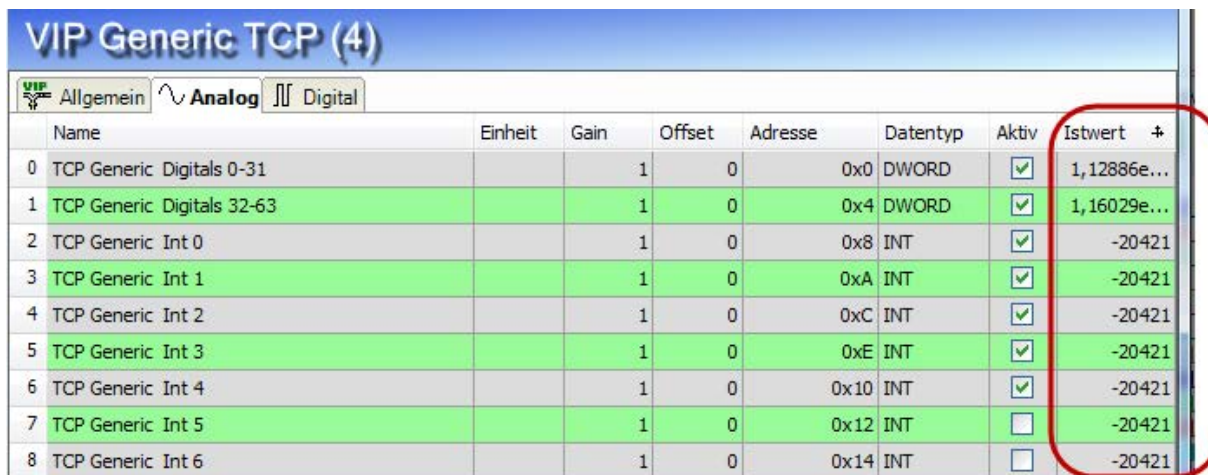
Hinweis



Der Modultyp *VIP TCP/UDP Allgemein* unterstützt auch die Erfassung und Verarbeitung von Texten. Hierzu kann im Register *Analog* der Datentyp STRING[32] ausgewählt werden. Zur Wandlung eines Textsignals bzw. Unterteilung in mehrere Textsignale verwenden Sie ein *Textrenner*-Modul unter der Schnittstelle *Virtuell*.

3.5.8 Moduldiagnose

In den Tabellen *Analog* und *Digital* der VIP-TCP/UP-Module werden die Inhalte der Telegramme angezeigt.



	Name	Einheit	Gain	Offset	Adresse	Datentyp	Aktiv	Istwert	±
0	TCP Generic Digitals 0-31		1	0	0x0	DWORD	<input checked="" type="checkbox"/>	1,12886e...	
1	TCP Generic Digitals 32-63		1	0	0x4	DWORD	<input checked="" type="checkbox"/>	1,16029e...	
2	TCP Generic Int 0		1	0	0x8	INT	<input checked="" type="checkbox"/>	-20421	
3	TCP Generic Int 1		1	0	0xA	INT	<input checked="" type="checkbox"/>	-20421	
4	TCP Generic Int 2		1	0	0xC	INT	<input checked="" type="checkbox"/>	-20421	
5	TCP Generic Int 3		1	0	0xE	INT	<input checked="" type="checkbox"/>	-20421	
6	TCP Generic Int 4		1	0	0x10	INT	<input checked="" type="checkbox"/>	-20421	
7	TCP Generic Int 5		1	0	0x12	INT	<input type="checkbox"/>	-20421	
8	TCP Generic Int 6		1	0	0x14	INT	<input type="checkbox"/>	-20421	

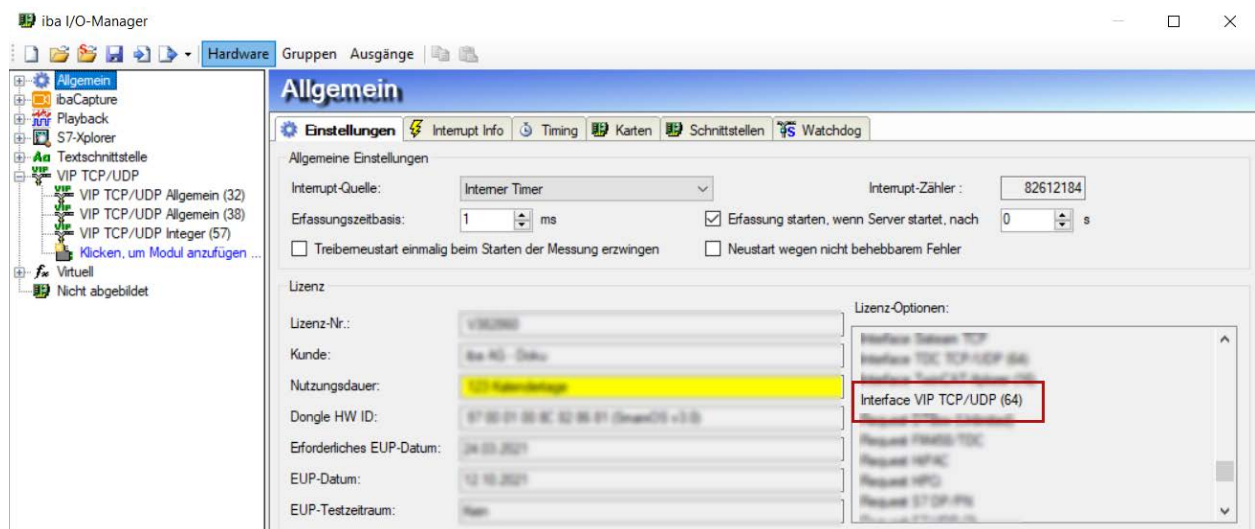
Folgende Fehler können auftreten:

- Es werden keine Daten angezeigt:
 - Der Telegrammpuffer auf der Senderseite ist nicht richtig gefüllt
 - Die Anschlüsse des Sendbausteins sind falsch beschaltet
- Es werden falsche Werte angezeigt:
 - Der Telegrammpuffer auf der Senderseite ist nicht richtig gefüllt (Offset-Fehler)
 - Bytereihenfolge ist falsch eingestellt (siehe ↗ *Allgemeine Moduleinstellungen*, Seite 19)
 - Es gibt mehrere Module mit dem gleichen Modulindex
- Die Digitalsignale sind falsch sortiert
 - Bytereihenfolge ist falsch eingestellt (siehe ↗ *Allgemeine Moduleinstellungen*, Seite 19).
- Die Telegramme kommen nicht schneller als ca. 200 ms mit Sequenzfehler
 - Problem mit "Delayed Acknowledge", siehe ↗ *Probleme mit TCP-Performance oder unbrauchbaren Daten durch Verwendung des Delayed ACK-Mechanismus*, Seite 29
 - Probleme durch "Nagle-Algorithmus", siehe ↗ *Unbrauchbare TCP-Daten als Folge des Nagle-Algorithmus*, Seite 31

4 Diagnose

4.1 Überprüfung der Lizenz

Falls die Schnittstelle „VIP TCP/UDP“ nicht im Signalbaum angezeigt wird, können Sie im I/O-Manager unter *Allgemein – Einstellungen – Lizenz* oder in der *ibaPDA*-Dienststatus-Applikation überprüfen, ob Ihre Lizenz ordnungsgemäß erkannt wird. Die Anzahl der lizenzierten Verbindungen ist in Klammern angegeben.



4.2 Sichtbarkeit der Schnittstelle

Ist die Schnittstelle trotz gültiger Lizenz nicht zu sehen, ist sie möglicherweise verborgen. Klicken Sie auf das Register *Schnittstellen* und aktivieren Sie das „Interface VIP TCP/UDP“.



4.3 Protokolldateien

Wenn Verbindungen zu Zielsystemen bzw. Clients hergestellt wurden, dann werden alle verbindungspezifischen Aktionen in einer Textdatei protokolliert. Diese (aktuelle) Datei können Sie z. B. nach Hinweisen auf mögliche Verbindungsprobleme durchsuchen.

Die Protokolldatei kann über den Button <Protokolldatei öffnen> geöffnet werden. Der Button befindet sich im I/O-Manager:

- bei vielen Schnittstellen in der jeweiligen Schnittstellenübersicht
- bei integrierten Servern (z.B. OPC UA-Server) im Register Diagnose.

Im Dateisystem auf der Festplatte finden Sie die Protokolldateien im Programmpfad von *ibaPDA*-Server (...\\Programme\\iba\\ibaPDA\\Server\\Log\\). Die Dateinamen der Protokolldateien werden aus der Bezeichnung bzw. Abkürzung der Schnittstellenart gebildet.

Dateien mit Namen `Schnittstelle.txt` sind stets die aktuellen Protokolldateien. Dateien mit Namen `Schnittstelle_yyyy_mm_dd_hh_mm_ss.txt` sind archivierte Protokolldateien.

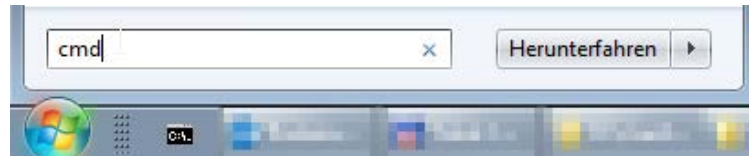
Beispiele:

- `ethernetipLog.txt` (Protokoll von EtherNet/IP-Verbindungen)
- `AbEthLog.txt` (Protokoll von Allen-Bradley-Ethernet-Verbindungen)
- `OpcUAServerLog.txt` (Protokoll von OPC UA-Server-Verbindungen)

4.4 Verbindungsdiagnose mittels PING

Ping ist ein System-Befehl, mit dem überprüft werden kann, ob ein bestimmter Kommunikationspartner in einem IP-Netzwerk erreichbar ist.

Öffnen Sie eine Windows Eingabeaufforderung.



Geben Sie den Befehl „ping“ gefolgt von der IP-Adresse des Kommunikationspartners ein und drücken Sie <ENTER>.

Bei bestehender Verbindung erhalten Sie mehrere Antworten.

```
Administrator: C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. Alle Rechte vorbehalten.

C:\Windows\system32>ping 192.168.1.10

Ping wird ausgeführt für 192.168.1.10 mit 32 Bytes Daten:
Antwort von 192.168.1.10: Bytes=32 Zeit=1ms TTL=30
Antwort von 192.168.1.10: Bytes=32 Zeit<1ms TTL=30
Antwort von 192.168.1.10: Bytes=32 Zeit<1ms TTL=30
Antwort von 192.168.1.10: Bytes=32 Zeit<1ms TTL=30

Ping-Statistik für 192.168.1.10:
    Pakete: Gesendet = 4, Empfangen = 4, Verloren = 0
    (0% Verlust),
    Ca. Zeitangaben in Millisek.:
        Minimum = 0ms, Maximum = 1ms, Mittelwert = 0ms

C:\Windows\system32>
```

Bei nicht bestehender Verbindung erhalten Sie Fehlermeldungen.

```
Administrator: C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. Alle Rechte vorbehalten.

C:\Windows\system32>ping 192.168.1.10

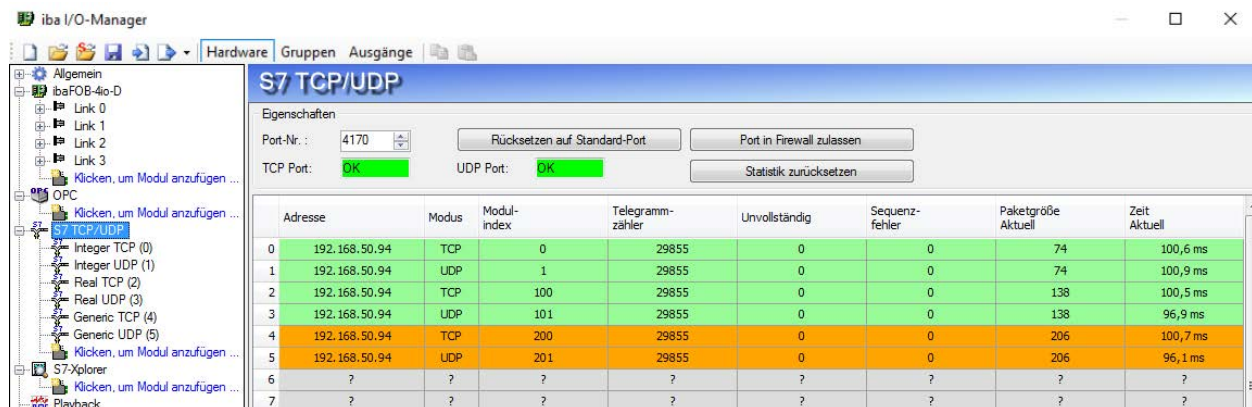
Ping wird ausgeführt für 192.168.1.10 mit 32 Bytes Daten:
Antwort von 192.168.1.100: Zielhost nicht erreichbar.
Zeitüberschreitung der Anforderung.
Zeitüberschreitung der Anforderung.
Zeitüberschreitung der Anforderung.

Ping-Statistik für 192.168.1.10:
    Pakete: Gesendet = 4, Empfangen = 1, Verloren = 3
    (75% Verlust),

C:\Windows\system32>
```

4.5 Überprüfung der Verbindung

Nach Übernahme der der Konfiguration werden in der Verbindungsübersicht alle Verbindungen, sortiert nach Modulindex, angezeigt.



Die Hintergrundfarbe der Zeilen hat folgende Bedeutung:

Farbe	Bedeutung
Grün	Die Verbindung ist OK. Der <i>ibaPDA</i> -Modulzeitbasis ist gleich oder langsamer als der Telegrammzyklus. Der aktuellen Telegrammzyklus ist aus der Spalte "Zeit Aktuell" ersichtlich.
Orange	Die Verbindung ist OK, aber der Telegrammzyklus ist wesentlich langsamer als die <i>ibaPDA</i> -Modulzeitbasis. Es ist sinnvoll, die Modulzeitbasis an den Telegrammzyklus anzupassen.

Werden die Verbindungen nicht oder nur teilweise angezeigt kann das folgende Ursachen haben:

- Sender ist in Stopp
- keine Ethernet-Verbindung zwischen *ibaPDA*-PC und der ABB-Steuerung
- Fehler in der Verbindungs-Projektierung:
 - falsche Remote-IP-Adresse
 - *ibaPDA*-Portnummer stimmt nicht mit der Verbindungsprojektierung überein.
 - Portnummer ist durch die Firewall geblockt.
- Falscher Modulindex im Telegramm-Header angegeben

Weitere Fehler:

- Zählen die Telegrammzähler nicht kontinuierlich hoch, werden auf Senderseite die Telegramme nicht zyklisch gesendet.
- Zählen Werte in der Spalten "Unvollständig" und/oder "Sequenzfehler" hoch, deutet das auf einen der folgenden Fehler hin:
 - Die "message_length" im Telegrammheader hat nicht den erwarteten Wert.

- Der "sequence_counter" im Telegrammheader wird nicht richtig inkrementiert.
- Das "Delayed Acknowledge"-Problem tritt auf, siehe ➤ *Probleme mit TCP-Performance oder unbrauchbaren Daten durch Verwendung des Delayed ACK-Mechanismus*, Seite 29

5 Anhang

5.1 Fehlerbehebung

5.1.1 Probleme mit TCP-Performance oder unbrauchbaren Daten durch Verwendung des Delayed ACK-Mechanismus

Symptome:

ibaPDA-Messungen von Automatisierungsgeräten mit TCP/IP funktionieren manchmal nicht mit Zykluszeiten < 200 ms.

Fehlerbild in *ibaPDA*:

Unvollständige Telegramme und/oder Ausreißer in den Datenwerten (je nach Controller-Typ des Senders)

Ursache:

Es gibt im TCP/IP-Protokoll verschiedene Varianten, wie das "Acknowledge" behandelt wird:

Der Standard WinSocket arbeitet nach RFC1122 mit dem "Delayed Acknowledge"-Mechanismus (Delayed ACK). Dieser sagt aus, dass das "Acknowledge" verzögert wird bis weitere Telegramme eintreffen, um diese dann gemeinsam zu quittieren. Falls keine weiteren Telegramme kommen, wird spätestens nach 200 ms (abhängig vom Socket) das ACK-Telegramm gesendet.

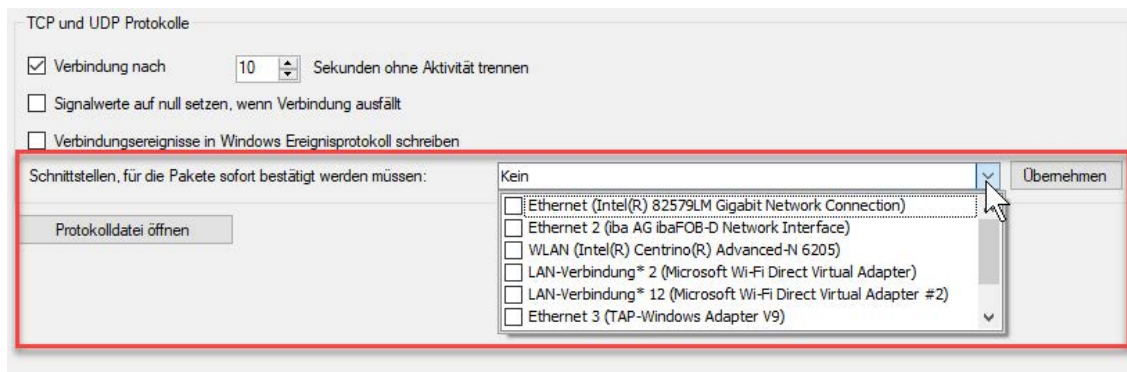
Der Datenfluss wird durch ein "Sliding Window" (Parameter Win=nnnn) gesteuert. Der Empfänger gibt an, wie viele Bytes er empfangen kann ohne eine Quittung zu senden.

Manche Controller akzeptieren dieses Verhalten nicht, sondern erwarten nach jedem Datentelegramm eine Quittung. Falls dieses nicht innerhalb einer bestimmten Zeit (200 ms) kommt, wiederholt er das Telegramm und packt evtl. neu zu sendende Daten dazu, was beim Empfänger zu einem Fehler führt, da das alte ja korrekt empfangen wurde.

Abhilfe:

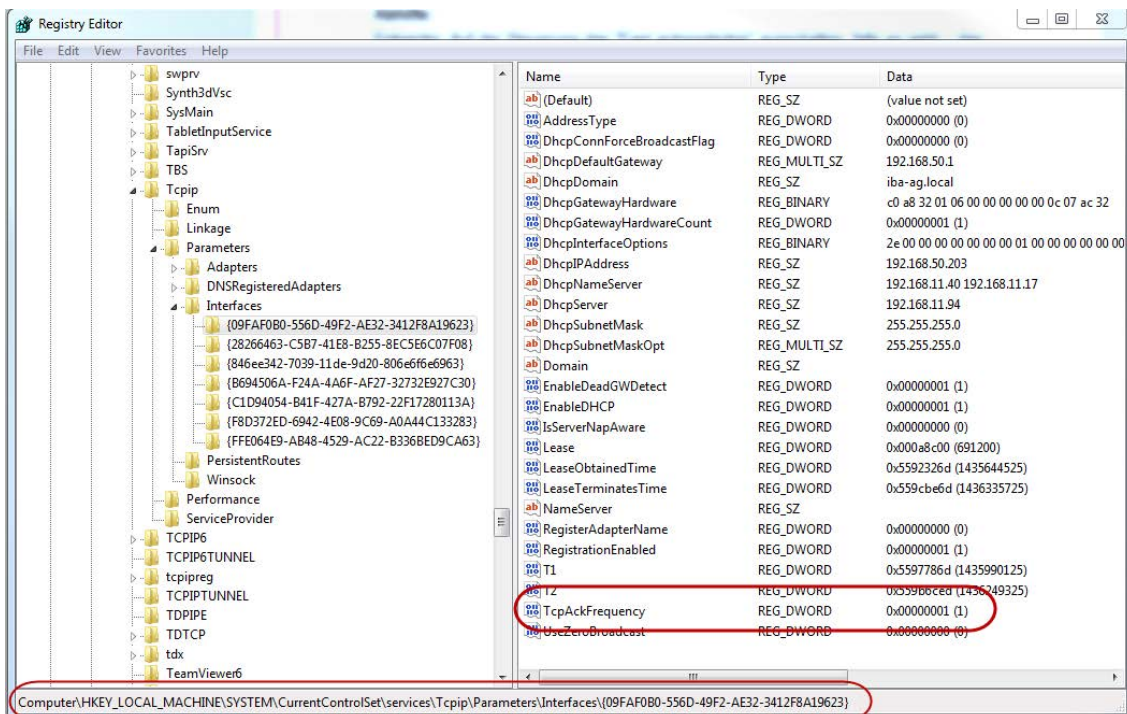
Das "Delayed Acknowledge" lässt sich einzeln pro Netzwerkadapter über einen Eintrag in der Windows Registry abschalten. Zur einfachen Änderung bietet *ibaPDA* im IO-Manager unter *Allgemein* im Register *Einstellungen* einen entsprechenden Dialog.

Wählen Sie in der Liste der Netzwerkadapter diejenigen aus, für die das "Delayed Acknowledge" deaktiviert werden soll und klicken Sie danach auf <Übernehmen>.



Der Parameter "TcpAckFrequency" (REG_DWORD = 1) wird dadurch im Registry-Pfad der ausgewählten Netzwerkadapter angelegt:

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters\Interfaces\{InterfaceGUID}



Hinweis



Grundsätzlich können Sie derartige, TCP-spezifische Probleme umgehen, indem Sie *UDP* anstelle von *TCP* nutzen.

Das User Datagram Protocol (UDP) ist ein minimales, nicht verbindungsorientiertes und gegen Telegrammverluste ungesichertes Netzwerkprotokoll. Dabei wird u. a. auf Empfangsquittierung der gesendeten Daten verzichtet. In stabilen und performanten Netzwerken fällt dies jedoch nicht nennenswert ins Gewicht und kann aufgrund der bei *ibaPDA* üblichen zyklischen Datenübertragung vernachlässigt werden.

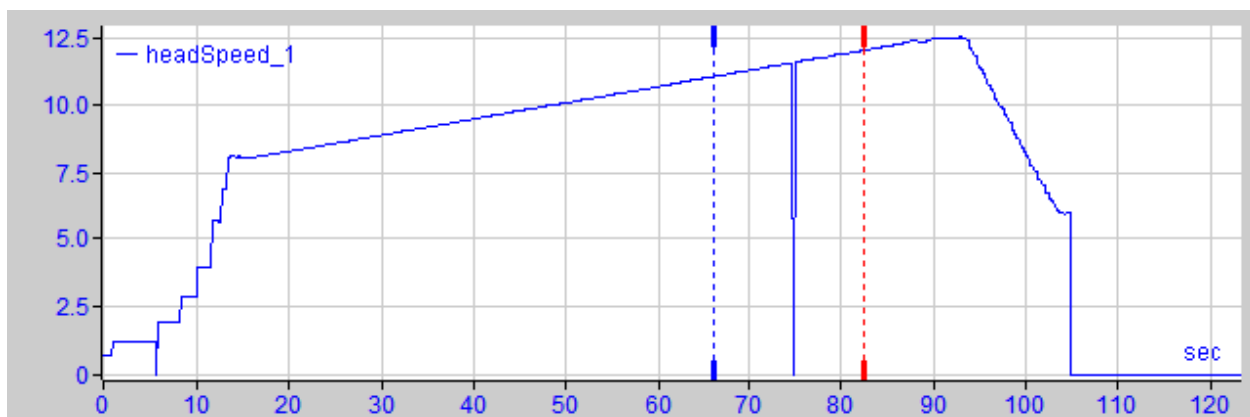
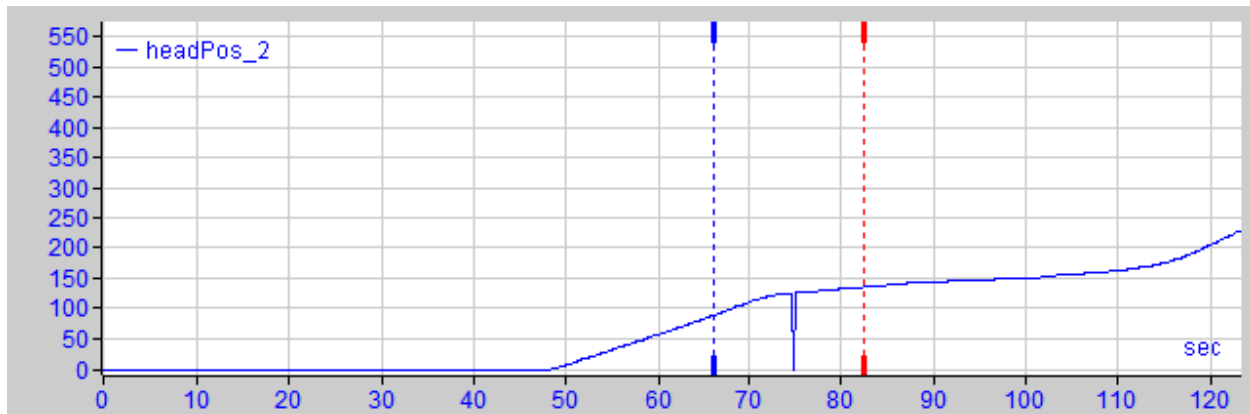
5.1.2 Unbrauchbare TCP-Daten als Folge des Nagle-Algorithmus

Symptome:

ibaPDA-Messungen von Automatisierungsgeräten mit TCP/IP zeigen Ausreißer in den Messwerten.

Fehlerbild in *ibaPDA*:

Unvollständige Telegramme und/oder Ausreißer in den Datenwerten (siehe Beispiele in den folgenden Abbildungen)



Ursache:

Der Nagle-Algorithmus (Nagle's algorithm), benannt nach seinem Schöpfer John Nagle, ist ein Mechanismus zur Verbesserung der TCP-Effizienz, indem er die Anzahl der über das Netz gesendeten kleinen Pakete reduziert und mehrere Datenblöcke sammelt, bevor die Daten über das Netz gesendet werden.

Da die Schnittstelle "Generic-TCP" kein Protokoll auf Anwendungsebene verwendet, kann der Empfänger *ibaPDA* diese zusammengefassten Nachrichten nicht korrekt verarbeiten. Die Schnittstelle "Generic TCP" erwartet nur ein Datagramm pro TCP-Telegramm mit stets gleichem Layout und gleicher Länge.

Aber der Nagle-Algorithmus und die Option *Delayed ACK* (Delayed Acknowledge, siehe 5.1.1, Seite 29) spielen in einem TCP/IP-Netzwerk nicht gut zusammen:

Der Delayed ACK-Mechanismus versucht, mehr Daten pro Segment zu senden, wenn er kann.

Ein Teil des Nagle-Algorithmus hängt aber von einem ACK ab, um Daten zu senden.

Nagle-Algorithmus und Delayed ACK ergeben zusammen ein Problem, weil Delayed ACKs darauf warten, das ACK zu senden, während "Nagle" darauf wartet, das ACK zu empfangen!

Dies führt zu zufälligen Verzögerungen von 200 ms bis 500 ms bei Segmenten, die sonst sofort gesendet und an den empfangsseitigen Stack von *ibaPDA* als Anwendung übergeben werden könnten.

Abhilfe:

Wir empfehlen, zunächst den *Delayed ACK*-Mechanismus zu deaktivieren, wie in Kapitel 5.1.1, Seite 29 erläutert. In einer typischen Echtzeitanwendung schickt der Sender dann die neuen Daten mit einer bestimmten Zykluszeit an *ibaPDA*, da die vorherigen Daten sofort quittiert wurden. Je nach Implementierung des TCP/IP-Stacks auf der Senderseite kann der Nagle-Algorithmus dennoch aktiv werden und automatisch eine Reihe kleiner Puffernachrichten aggregieren, wodurch der Algorithmus die Übertragung absichtlich verlangsamt.

Dies kann auch sporadisch durch eine kurzzeitige Überlastung auf der Senderseite geschehen, die den Stack dazu veranlasst, einige Nachrichten zusammenzulegen.

Um den puffernden Nagle-Algorithmus zu deaktivieren, verwenden Sie die Socket-Option *TCP_NODELAY*. Die Socket-Option *TCP_NODELAY* ermöglicht es dem Netzwerk, die durch den Nagle-Mechanismus verursachten Delays zu umgehen, indem der Nagle-Algorithmus deaktiviert wird und die Daten gesendet werden, sobald sie verfügbar sind.

Die Aktivierung von *TCP_NODELAY* zwingt einen Socket, die Daten in seinem Puffer zu senden, unabhängig von der Paketgröße. Das *TCP_NODELAY*-Flag ist eine Option, die für jeden einzelnen Socket aktiviert werden kann und beim Erstellen eines TCP-Sockets angewendet wird.

(Siehe Eigenschaft *Socket.NoDelay* in .NET-Anwendungen im Namespace *System.Net.Sockets*)

Hinweis



Grundsätzlich können Sie derartige, TCP-spezifische Probleme umgehen, indem Sie *UDP* anstelle von *TCP* nutzen.

Das User Datagram Protocol (UDP) ist ein minimales, nicht verbindungsorientiertes und gegen Telegrammverluste ungesichertes Netzwerkprotokoll. Dabei wird u. a. auf Empfangsquittierung der gesendeten Daten verzichtet. In stabilen und performanten Netzwerken fällt dies jedoch nicht nennenswert ins Gewicht und kann aufgrund der bei *ibaPDA* üblichen zyklischen Datenübertragung vernachlässigt werden.

6 Support und Kontakt

Support

Tel.: +49 911 97282-14
Fax: +49 911 97282-33
E-Mail: support@iba-ag.com

Hinweis



Wenn Sie Support benötigen, dann geben Sie bitte bei Softwareprodukten die Lizenznummer bzw. die CodeMeter-Containernummer (WIBU-Dongle) an. Bei Hardwareprodukten halten Sie bitte ggf. die Seriennummer des Geräts bereit.

Kontakt

Hausanschrift

iba AG
Königswarterstraße 44
90762 Fürth
Deutschland

Tel.: +49 911 97282-0
Fax: +49 911 97282-33
E-Mail: iba@iba-ag.com

Postanschrift

iba AG
Postfach 1828
90708 Fürth

Warenanlieferung, Retouren

iba AG
Gebhardtstraße 10
90762 Fürth

Regional und weltweit

Weitere Kontaktadressen unserer regionalen Niederlassungen oder Vertretungen finden Sie auf unserer Webseite

www.iba-ag.com.