



ibaPDA-Interface-Generic-TCP

Datenschnittstelle Generic TCP

Handbuch
Ausgabe 1.9

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

Hersteller

iba AG
Königswarterstraße 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 2023, 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	Autor	Version SW
1.9	01-2023	Diagnosemodule	rm	7.3.0

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 Generic TCP.....	8
3.1	Allgemeine Informationen.....	8
3.2	Konfigurieren des Controllers.....	9
3.3	Konfiguration und Projektierung ibaPDA.....	10
3.3.1	Allgemeine Einstellungen	10
3.3.2	Allgemeine Einstellungen der Schnittstelle.....	11
3.3.3	Modul hinzufügen	12
3.3.4	Allgemeine Moduleinstellungen.....	12
3.3.5	Signalkonfiguration.....	15
3.3.6	Moduldiagnose.....	16
3.4	Konfiguration ibaPDA-Ausgangsmodule.....	17
3.4.1	Allgemeine Moduleinstellungen ibaPDA-Ausgangsmodule.....	17
3.4.2	Signalkonfiguration	18
4	Diagnose	20
4.1	Lizenz	20
4.2	Sichtbarkeit der Schnittstelle.....	20
4.3	Protokolldateien	21
4.4	Verbindungsdiagnose mittels PING	22
4.5	Überprüfung der Verbindung für Empfangstelegramme.....	23
4.6	Überprüfen der Verbindung für Sendetelegramme	25
4.7	Diagnosemodule.....	26
5	Anhang	32
5.1	Fehlerbehebung.....	32
5.1.1	Probleme mit TCP-Performance durch Delayed Acknowledge.....	32
5.1.2	Unbrauchbare TCP-Daten als Folge des Nagle-Algorithmus	34

6	Support und Kontakt.....	36
----------	---------------------------------	-----------

1 Zu diesem Handbuch

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

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-Generic-TCP* 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	<i>Dateiname, Pfad</i> Beispiel: <i>Test.docx</i>

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 *Generic TCP* erforderlich:

- *ibaPDA* v7.3.0 oder höher
- Basislizenz für *ibaPDA+* Lizenz für *ibaPDA-Interface-Generic-TCP*
- Netzwerkanschluss 10/100 Mbit

In der *ibaPDA*-Dokumentation finden Sie weitere Anforderungen an die Computer-Hardware und die unterstützten Betriebssysteme.

Hinweis



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

Systemeinschränkungen

- Je nach Gerätekonstellation und Controller-Typen kann es durch netzwerkspezifische Einstellungen bei der Behandlung des TCP/IP-Acknowledge zu Problemen kommen.
Siehe dazu ➤ *Fehlerbehebung*, Seite 32 (alle *ibaPDA*-Versionen).

Lizenzinformationen

Bestell-Nr.	Produktbezeichnung	Beschreibung
31.001076	ibaPDA-Interface-Generic-TCP	Erweiterungslizenz für ein <i>ibaPDA</i> -System um eine Generic-TCP-Schnittstelle Anzahl der Verbindungen: 64
31.101076	one-step-up-Interface-Generic-TCP	Erweiterungslizenz für eine vorhandene Schnittstelle um 64 weitere Generic-TCP-Verbindungen, maximal 3 Erweiterungslizenzen zulässig

3 Datenschnittstelle Generic TCP

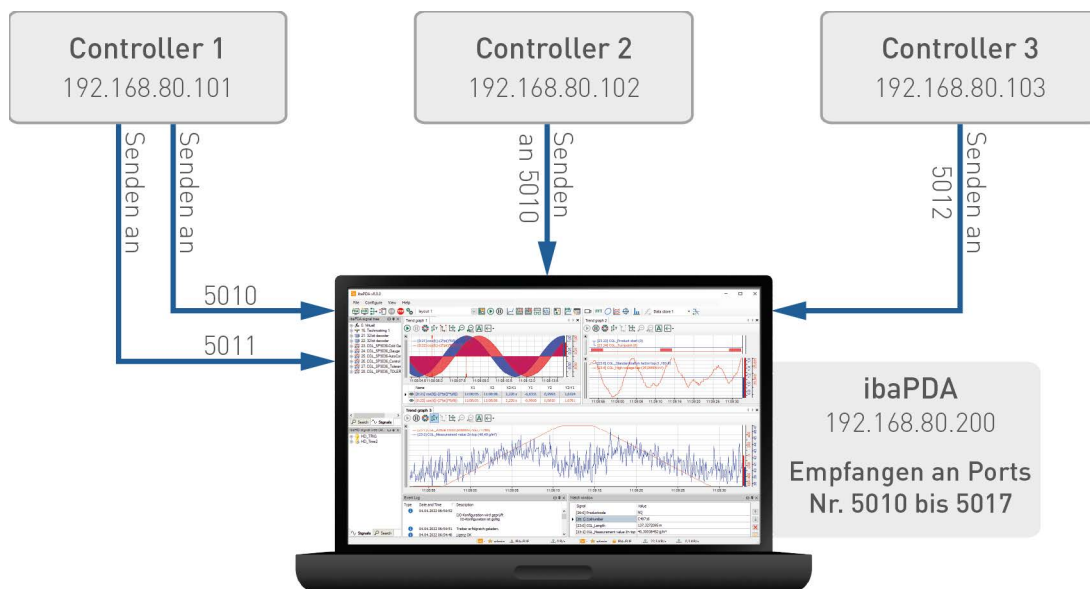
3.1 Allgemeine Informationen

ibaPDA-Interface-Generic-TCP kann von jedem Controller genutzt werden, der TCP/IP-Nachrichten senden kann.

Das Transmission Control Protocol (TCP) ist eines der Kern-Protokolle der Internet Protocol Suite. IP ist auf niedrigerer Ebene für die Vermittlung von Nachrichten über das Internet zuständig. TCP arbeitet auf einem höheren Level und stellt die Verbindung zwischen zwei Endsystemen her. TCP sorgt für eine verlässliche und geordnete Zustellung eines Datenstroms von einem Programm auf einem Computer an ein anderes Programm auf einem anderen Computer.

Die Generic-TCP-Nachrichten sind IP Unicast Nachrichten, die von einem oder mehreren Controllern an das *ibaPDA*-System über eine definierte Portnummer geschickt werden.

Die folgende Darstellung gibt einen Überblick über eine mögliche Konfiguration, in der drei Controller TCP/IP-Nachrichten an ein iba-System senden.



Eigenschaften

- Die Nachrichten, die über die Verbindungen verschickt werden, müssen kein bestimmtes feststehendes Layout aufweisen.
- An der Generic-TCP-Schnittstelle im I/O-Manager von *ibaPDA* können Sie einen Portbereich festlegen. Im Beispiel oben wartet der *ibaPDA*-Treiber von Port 5010 bis 5017 auf eine Verbindung.
- Jede TCP-Verbindung wird von *ibaPDA* eindeutig mit einer Ziel-Portnummer und einer Quell-IP-Adresse identifiziert.

Dadurch kann *ibaPDA* Daten von verschiedenen Controllern empfangen, die denselben Ziel-Port nutzen. Es ist auch möglich, mehrere Nachrichten von einem Controller über verschiedene Ports an *ibaPDA* zu senden.

Dies wird in dem obigen Beispiel anschaulich dargestellt: Controller 1 und 2 nutzen denselben Port 5010, verfügen aber über verschiedene IP-Adressen. Controller 1 verschickt mehrere Nachrichten, verwendet aber dafür verschiedenen Ports (5010 und 5011).

ibaPDA-spezifische Einschränkungen

- Die maximale Länge der TCP-Nachricht ist auf 16384 Bytes begrenzt.
- Die Anzahl der unterstützten Verbindungen in *ibaPDA* ist abhängig von der Generic-TCP-Lizenz (64, 128, 192 oder 256).

Folgende Controller können eingesetzt werden:

- Jedes System, das TCP/IP-Nachrichten versenden kann.

Andere Dokumentation



Verweis auf ergänzende Dokumentation oder weiterführende Literatur.

- *ibaPDA*-Handbuch (<http://www.iba-ag.com/de/support/downloads/>)
 - Ein TCP/IP Tutorial, RFC1180 (<ftp://ftp.ripe.net/rfc/rfc1180.txt>)
 - Transmission Control Protocol, RFC793 (<ftp://ftp.ripe.net/rfc/rfc793.txt>)
-

3.2 Konfigurieren des Controllers

ibaPDA unterstützt zwei Verbindungsarten:

1. *ibaPDA* ist TCP-Server (Moduleinstellungen, Modus = passiv):

Der Controller, der Daten an *ibaPDA* senden soll, muss eine TCP/IP-Verbindung aufbauen. Die Zieladresse ist die IP-Adresse des *ibaPDA*-Servers. Der Ziel-Port muss innerhalb des definierten Port-Bereichs in *ibaPDA* festgelegt werden. Der Standard-Bereich für die Ziel-Ports, der in *ibaPDA* definiert ist, reicht von 5010 bis 5017 (siehe ➔ *Allgemeine Einstellungen der Schnittstelle*, Seite 11).

Jede TCP/IP-Verbindung wird von *ibaPDA* eindeutig identifiziert durch die "Ziel-Portnummer" und "Quell-IP-Adresse". Das heißt, dass ein Controller mehrere Nachrichten an dasselbe *ibaPDA*-System senden kann, auch wenn dieser einen Ziel-Port nutzt, der bereits von anderen Controllern genutzt wird.

2. *ibaPDA* ist TCP-Client (Moduleinstellungen, Modus = aktiv):

Der Controller, der Daten an *ibaPDA* senden soll, muss auf einem festgelegten Port auf den Verbindungsaufbau durch *ibaPDA* warten. In *ibaPDA* ist unter den Moduleinstellungen als Quelladresse die IP-Adresse des Controllers und als Portnummer der Port einzugeben auf dem der Controller auf den Verbindungsaufbau wartet. Sobald der Verbindungsaufbau abgeschlossen ist, kann der Controller die Daten senden.

Die maximal unterstützte Länge der TCP-Daten beträgt 4096 Bytes. Längere Nachrichten werden von *ibaPDA* zurückgewiesen.

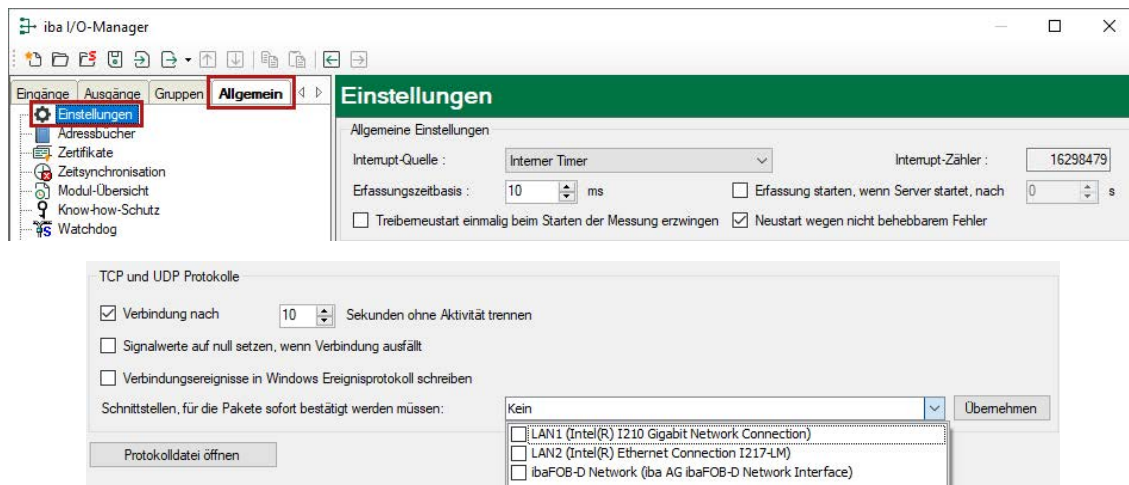
Der Controller sollte in bestimmten Abständen eine Nachricht mit festem Layout an das *ibaPDA*-System schicken. Wird mehr als eine Nachricht mit verschiedenen Inhalten benötigt, muss ein anderer Port innerhalb des definierten Bereichs verwendet werden.

3.3 Konfiguration und Projektierung ibaPDA

Nachfolgend ist die Projektierung in *ibaPDA* beschrieben. Wenn alle Systemvoraussetzungen erfüllt sind, bietet *ibaPDA* im Signalbaum des I/O-Managers die Schnittstelle *Generic TCP* an.

3.3.1 Allgemeine Einstellungen

Die "Totmann-Timeout" Konfiguration erfolgt für alle von *ibaPDA* unterstützten TCP- und UDP-Protokolle gemeinsam.



Verbindung nach ... 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

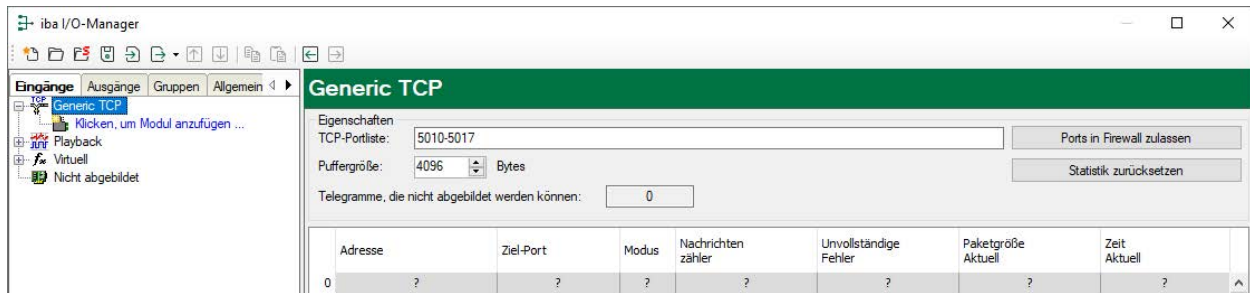
Hinweis



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.

3.3.2 Allgemeine Einstellungen der Schnittstelle

Die Schnittstelle hat folgende Funktionen und Konfigurationsmöglichkeiten.



TCP Portliste

Ports, auf denen *ibaPDA* auf ankommenden Verbindungsaufbau wartet.

Sie können die Portnummern als Bereich oder als Aufzählung oder beides kombiniert eingeben. Geben Sie einen Bereich mit einem Bindestrich ein und trennen Sie nicht aufeinander folgende Portnummern durch Kommas. Standard ist der Bereich 5010 – 5017.

Die Portnummer muss in dem Controller identisch verwendet werden (siehe [↗ Konfigurieren des Controllers](#), Seite 9).

Hinweis



Die Liste gilt nur für den Modus "passiv". Sie enthält die Portnummern, auf denen *ibaPDA* auf einen Verbindungsaufbau durch den "aktiven" Partner wartet. Für aktive Verbindungen kann *ibaPDA* Portnummern außerhalb dieser Liste verwenden. Diese müssen dann beim Verbindungspartner in der Firewall zugelassen sein.

Puffergröße

Die maximale Datengröße einer Verbindung ist konfigurierbar. Sie sollte zwischen 1024 und 16384 Bytes liegen. Voreinstellung: 4096 Bytes

Ports in Firewall zulassen

Bei der Installation von *ibaPDA* werden die Standard-Portnummern der verwendeten Protokolle automatisch in der Firewall eingetragen. Wenn Sie die Portnummer hier verändern oder das Interface nachträglich freischalten, müssen Sie über diesen Button diesen Port in der Firewall zulassen.

<Statistik zurücksetzen>

Klicken Sie diesen Button, wenn Sie die berechneten Zeitwerte und den Fehlerzähler in der Tabelle auf 0 setzen möchten.

Zähler für fehlerhafte Telegramme

siehe [↗ Überprüfung der Verbindung für Empfangstelegramme](#), Seite 23.

Verbindungstabelle

siehe [↗ Überprüfung der Verbindung für Empfangstelegramme](#), Seite 23.

3.3.3 Modul hinzufügen

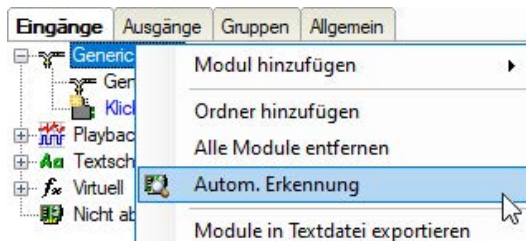
Vorgehen

1. Klicken Sie auf den blauen Befehl *Klicken, um Modul anzufügen*, der sich unter jeder Datenschnittstelle im Register *Eingänge* oder *Ausgänge* befindet.
2. Wählen Sie im Dialogfenster den gewünschten Modultyp aus und vergeben Sie bei Bedarf einen Namen über das Eingabefeld.
3. Bestätigen Sie Ihre Auswahl mit <OK>.

Tipp



Zyklische TCP/IP-Telegramme, deren Ziel-Port im TCP-Port-Bereich der Schnittstelle liegen, werden automatisch in der Verbindungsliste angezeigt, auch wenn keine entsprechenden Module definiert sind und *ibaPDA* nicht gestartet ist. Für diese Verbindungen können Sie auf einfache Weise Module anlegen, indem Sie in der Baumansicht mit der rechten Maustaste auf die Schnittstelle *Generic TCP* klicken und im Kontextmenü *Autom. Erkennung* wählen.



Mit der automatischen Erkennung wird für jede Verbindung ein Generic TCP-Modul mit 32 analogen Signalen (Datentyp FLOAT) und 32 digitalen Signalen angelegt.

Hinweis

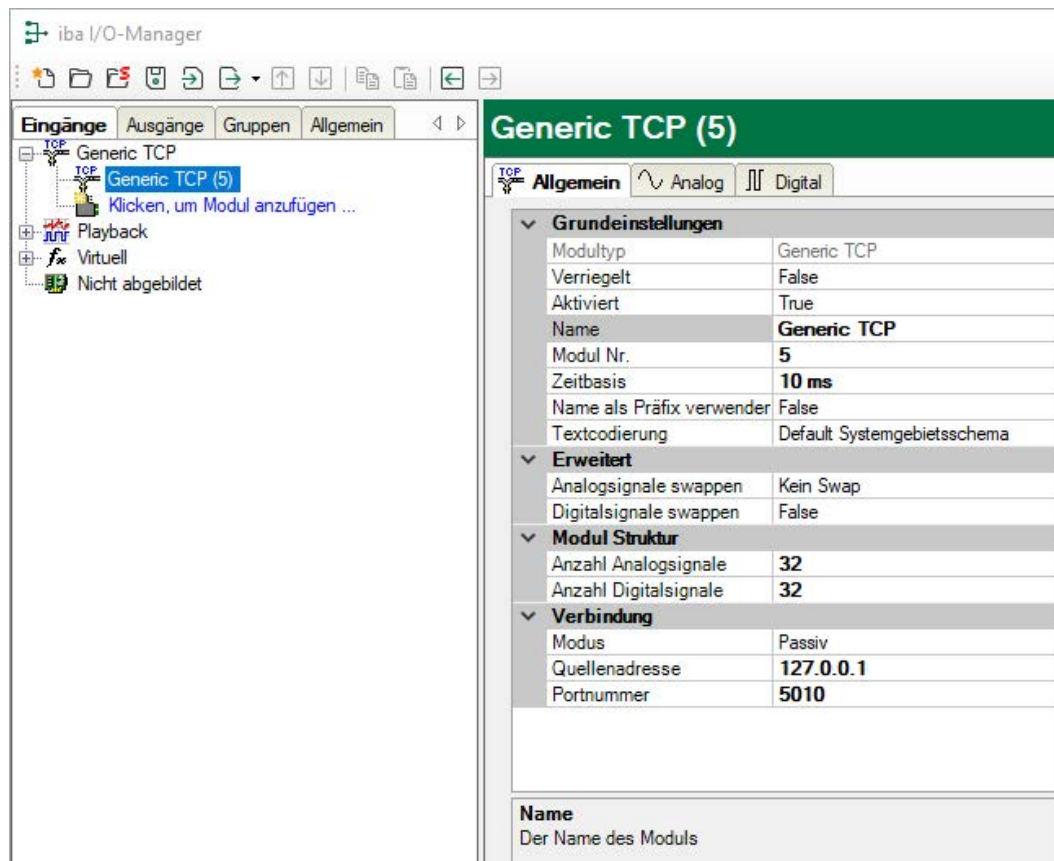


Die automatische Erkennung funktioniert nur für passive TCP-Verbindungen.

3.3.4 Allgemeine Moduleinstellungen

Um ein Modul zu konfigurieren, markieren Sie es in der Baumstruktur.

Alle Module haben die folgenden Einstellmöglichkeiten.



Grundeinstellungen

Modultyp (nur Anzeige)

Zeigt den Typ des aktuellen Moduls an.

Verriegelt

Sie können ein Modul verriegeln, um ein versehentliches oder unautorisiertes Ändern der Einstellungen zu verhindern.

Aktiviert

Aktivieren Sie das Modul, um Signale aufzuzeichnen.

Name

Hier können Sie einen Namen für das Modul eintragen.

Modul Nr.

Diese interne Referenznummer des Moduls 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

Diese Option stellt den Modulnamen den Signalnamen voran.

Textcodierung

Für eine korrekte Interpretation und Anzeige der empfangenen Textdaten können Sie hier die Form der Textcodierung bzw. die Codepage auswählen. Zur Auswahl stehen:

- Default Systemgebietsschema: Codierung gem. der Windows-Systemeinstellungen
- Western European (1252): 8-Bit-Kodierung inkl. ASCII, westeurop. Umlaute, Sonderzeichen
- UTF-8: Unicode, für universelle Zeichen inkl. ASCII, Asiatische Sprachen usw.

Erweitert

Analogsignale swappen, Digitalsignale swappen

Möglichkeit die Auswertereihenfolge der Byte zu ändern

Modul Struktur

Anzahl Analogsignale, Anzahl Digitalsignale

Stellen Sie die Anzahl der konfigurierbaren Analogsignale bzw. Digitalsignale in den Signaltabellen ein. Der Standardwert ist jeweils 32. Der Maximalwert beträgt 4096. Die Signaltabellen werden entsprechend angepasst. Beachten Sie, dass die Gesamtmenge der Daten 4096 Bytes für Analog- und Digitalsignale zusammen nicht überschreiten darf.

Verbindung

Modus

Hier wählen Sie, welcher Partner die Verbindung aufbaut:

- Modus *Passiv* (Standard): *ibaPDA* wartet auf dem eingestellten Port auf Verbindungsaufbau durch den Partner (Controller).
- Modus *Aktiv*: *ibaPDA* baut die Verbindung auf, der Partner muss als passiv eingestellt sein und auf dem eingestellten Port auf Verbindungsaufbau warten.

Andere Bezeichnung für den Verbindungsmodus ist "Client/Server":

Client entspricht aktiv, Server entspricht passiv.

Quellenadresse, Portnummer

- Im Modus *Passiv* dienen IP-Adresse und Portnummer zur eindeutigen Identifizierung der Verbindung zu einem bestimmten Controller. Der Port muss in dem für die Schnittstelle definierten Portbereich enthalten und in der Firewall zugelassen sein (siehe ➤ *Allgemeine Einstellungen der Schnittstelle*, Seite 11).
- Im Modus *Aktiv* dienen IP-Adresse und Portnummer zum Verbindungsaufbau zum sendenden, aber passiven Partner. Der Port kann außerhalb des TCP-Portbereichs sein und er muss in der Firewall des Partners zugelassen sein.

Andere Dokumentation



Weitere Informationen finden Sie im *ibaPDA*-Handbuch.

3.3.5 Signalkonfiguration

Die Auswahl der zu messenden Daten erfolgt auf der Controller-Seite durch Rangieren der Signale in das Telegramm, das zyklisch an *ibaPDA* gesendet wird.

Register Analog und Digital

<div> Allgemein Analog Digital </div>									
	Name	Einheit	Gain	Offset	Adresse	Datentyp	Aktiv	Istwert	±
0	length		1	0	0	INT	<input checked="" type="checkbox"/>	138	^
1	Id		1	0	2	INT	<input checked="" type="checkbox"/>	101	
2	Counter		1	0	4	INT	<input checked="" type="checkbox"/>	24388	
3	Bit vector		1	0	6	DWORD	<input checked="" type="checkbox"/>	327693	
4	Sine		1	0	10	FLOAT	<input checked="" type="checkbox"/>	0,938289	
5	Cosine		1	0	14	FLOAT	<input checked="" type="checkbox"/>	-0,345852	
6	Triangle		1	0	18	FLOAT	<input checked="" type="checkbox"/>	2194,76	

Sie können die analogen und digitalen Signale mit Namen, Einheit, Adresse und Datentyp bzw. Bit-Nummer versehen, sowie aktiv bzw. inaktiv setzen.

Andere Dokumentation



Eine Beschreibung der Spalten finden Sie im *ibaPDA*-Handbuch.

Spezifische Spalten für Generic TCP Module:

Adresse

Die Adresse gibt den Offset des ersten Byte dieses Wertes innerhalb des Nutzdatentelegramms an. Den Offset können Sie als Hexadezimal- oder Dezimalwert eintragen, wenn Sie den entsprechenden Punkt im Kontextmenü auswählen.

	Offset	Adresse	Datentyp	Aktiv	Istwert	±
1					18944	
1					0	
1					-719	
1					0	
1					-13056	

Die Adressierung der Digitalsignale erfolgt über die Spalten *Adresse* und *Bit-Nr.* 0 – 31.

Datentyp (nur Analogsignale)

ibaPDA unterstützt die Datentypen BYTE, WORD, DWORD, SINT, INT, DINT, LINT, FLOAT, DOUBLE, S5 FLOAT und STRING[32] unterstützt.

Der Adressraum ist abhängig vom Datentyp. Daher müssen Sie eventuell die Adresseinträge anpassen, nachdem Sie Datentypen verändert haben.

Hinweis

Der Modultyp *Generic TCP* 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*.

Tipp

Sie können das automatische Ausfüllen der Spalten verwenden (siehe *ibaPDA-Handbuch*).

3.3.6 Moduldiagnose

In den Tabellen *Analog* und *Digital* der Generic TCP Module werden die Inhalte der Telegramme (Istwerte) angezeigt.

Allgemein Analog Digital									
	Name	Einheit	Gain	Offset	Adresse	Datentyp	Aktiv	Istwert	±
0	length		1	0	0	INT	<input checked="" type="checkbox"/>	138	^
1	Id		1	0	2	INT	<input checked="" type="checkbox"/>	101	
2	Counter		1	0	4	INT	<input checked="" type="checkbox"/>	24388	
3	Bit vector		1	0	6	DWORD	<input checked="" type="checkbox"/>	327693	
4	Sine		1	0	10	FLOAT	<input checked="" type="checkbox"/>	0,938289	
5	Cosine		1	0	14	FLOAT	<input checked="" type="checkbox"/>	-0,345852	
6	Triangle		1	0	18	FLOAT	<input checked="" type="checkbox"/>	2194,76	

Folgende Fehler können auftreten:

- Es werden keine Daten angezeigt:
 - Der Telegrammpuffer auf der Controllerseite ist nicht richtig gefüllt.
 - Die Anschlüsse des Sendebausteins sind falsch beschaltet.
- Es werden falsche Werte angezeigt:
 - Der Telegrammpuffer auf der Controllerseite ist nicht richtig gefüllt (Offset-Fehler).
 - Die Byte-Reihenfolge ist falsch eingestellt, siehe [Allgemeine Moduleinstellungen](#), Seite 12.
- Die Digitalsignale sind falsch sortiert:
 - Die Byte-Reihenfolge ist falsch eingestellt, siehe [Allgemeine Moduleinstellungen](#), Seite 12.
- Die Telegramme kommen nicht schneller als ca. 200 ms mit Sequenzfehler:
 - Problem mit "Delayed Acknowledge", siehe [Probleme mit TCP-Performance durch Delayed Acknowledge](#), Seite 32.
 - Probleme durch "Nagle-Algorithmus", siehe [Unbrauchbare TCP-Daten als Folge des Nagle-Algorithmus](#), Seite 34.

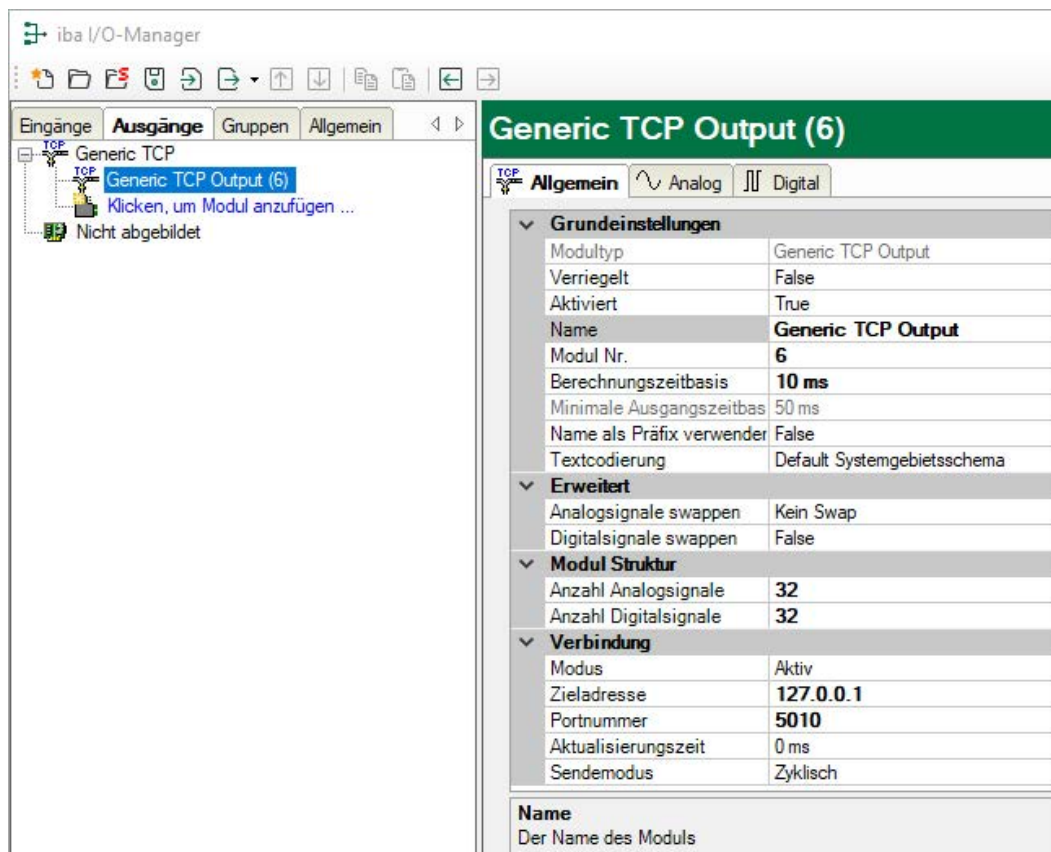
3.4 Konfiguration ibaPDA-Ausgangsmodule

Wenn alle Systemvoraussetzungen erfüllt sind, bietet *ibaPDA* im Signalbaum des I/O-Managers im Register *Ausgänge* die Schnittstelle *Generic TCP* an. Ein manuelles Hinzufügen der Schnittstelle ist nicht erforderlich.

Ausgangsmodule fügen Sie auf die gleiche Weise hinzu wie Eingangsmodule.

3.4.1 Allgemeine Moduleinstellungen ibaPDA-Ausgangsmodule

Um ein Ausgangsmodul zu konfigurieren, markieren Sie es in der Baumstruktur des Registers *Ausgänge*.



Die Parameter sind nahezu identisch zu denen der Eingangsmodule, siehe ↗ *Allgemeine Moduleinstellungen*, Seite 12.

Beachten Sie folgende Unterschiede zu den Einstellungen der Eingangsmodule:

Berechnungszeitbasis

Zeitbasis (in ms), die für die Berechnung der Ausgangswerte verwendet wird.

Die Berechnungszeitbasis ist nicht gleichbedeutend mit der Ausgangszeitbasis, mit der die Werte ausgegeben werden!

Minimale Ausgangszeitbasis

Zeitbasis, mit der die Ausgänge schnellstens aktualisiert werden können.

Der Wert wird vom System anhand der aktuellen I/O-Konfiguration automatisch ermittelt und hier nur angezeigt. Die Ausgangszeitbasis ergibt sich aus dem kleinsten gemeinsamen Vielfachen aller Modulzeitbasen, bzw. beträgt mindestens 50 ms.

Anzahl Analogsignale, Anzahl Digitalsignale

Die Anzahl der Signale und die zugeordneten Datentypen bestimmen die Länge des Telegramms. Für deaktivierte Signale schreibt *ibaPDA* 0 in den Telegrammpuffer.

Modus

- Im Modus *Aktiv* (Standard) baut *ibaPDA* die Verbindung zum Empfänger auf. Die Portnummer muss auf der Empfängerseite in der Firewall zugelassen sein.
- Im Modus *Passiv* baut der Empfänger die Verbindung auf. Die Portnummer darf nicht in dem Portbereich für die Eingangsmodule enthalten sein (siehe ↗ *Allgemeine Einstellungen der Schnittstelle*, Seite 11). Ein Feld für die IP-Adresse des Empfängers ist nicht vorhanden.

Folgende Parameter bestimmen den Telegramm-Sendezyklus:

Aktualisierungszeit

Wenn Sie 0 oder ein Wert kleiner als 50 eingeben, entspricht der Sendezyklus dem *ibaPDA*-Taskzyklus bzw. der Berechnungszeitbasis (mind. 50 ms).

Sendemodus

Gibt an, wann *ibaPDA* neue Telegramme sendet. Die Aktualisierungszeit bestimmt immer die schnellste Senderate.

- *Zyklisch*: *ibaPDA* sendet ein Telegramm im Takt der Aktualisierungszeit.
- *Bei Änderung*: *ibaPDA* sendet ein Telegramm bei jeder Änderung der Signaldaten.
- *Bei Trigger*: *ibaPDA* sendet ein Telegramm bei jeder steigenden Flanke des Triggersignals.

3.4.2 Signalkonfiguration

Wählen Sie in den Registern *Analog* und *Digital* die Signale aus, die Sie als Telegramm senden möchten.

Tipp

Wenn Sie die Ausgabedaten in einem virtuellen Modul definieren und hier nur die Referenzen zu diesen Daten eintragen, besteht die Möglichkeit, diese Daten auch in die Datenaufzeichnung aufzunehmen.

Register Analog und Digital

<div> <div>🔍 Allgemein</div> <div>📡 Analog</div> <div>🔌 Digital</div> </div>							
	Name	Ausdruck	Adresse	Datentyp	Aktiv	Istwert	↕
0	ibaPDA run time (ms)	T()		0 FLOAT	<input checked="" type="checkbox"/>	243,38	⬆
1	Act. Charge no.	[2:14]		4 DINT	<input checked="" type="checkbox"/>	4534	
2	FobD-Status Link 0	FobDLinkStatus(0, 0)		8 WORD	<input checked="" type="checkbox"/>	0	

Andere Dokumentation



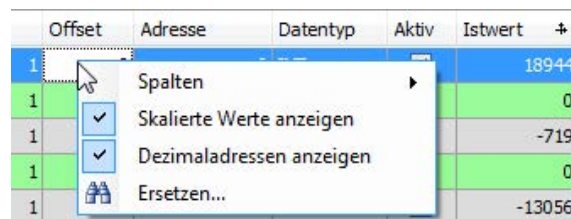
Eine Beschreibung der Spalten finden Sie im *ibaPDA*-Handbuch.

Ausdruck

Definieren Sie die gewünschten Ausgangssignale auf ähnliche Weise wie bei den virtuellen Signalen. Einfache Ausdrücke oder Referenzen zu bestehenden Signalen können Sie direkt in die Tabellen eingeben. Über den Button öffnen Sie den Ausdruckseditor. Einen fehlerhaften Ausdruck können Sie über den Button analysieren.

Adresse

Die Adresse gibt den Offset des ersten Byte dieses Wertes innerhalb des Sendetelegramms an. Den Offset können Sie als Hexadezimal- oder Dezimalwert eintragen, wenn Sie den entsprechenden Punkt im Kontextmenü auswählen.



Die Adressierung der Digitalsignale erfolgt über die Spalten *Adresse* und *Bit-Nr.* 0 – 15.

Datentyp

ibaPDA unterstützt die Datentypen BYTE, WORD, DWORD, SINT, INT, DINT, FLOAT, DOUBLE, und STRING[32] unterstützt.

Der Adressraum ist abhängig vom Datentyp. Daher müssen Sie eventuell die Adresseinträge anpassen, nachdem Sie Datentypen verändert haben.

Aktiv

Für deaktivierte Signale schreibt *ibaPDA* 0 in den Telegrammpuffer.

4 Diagnose

4.1 Lizenz

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

Die folgende Abbildung zeigt beispielhaft die Lizenz für die Schnittstelle *Codesys-Xplorer*.

Lizenzinformationen

Lizenzcontainer: 3-1000790
 Kundenname: Testsystem
 Nutzungsdauer: Unbegrenzt
 Containertyp: WIBU CmStick v4.40
 Container-Host: 192.168.1.100
 Erforderl. EUP-Datum: 01.02.2023
 EUP-Datum: 31.12.2025

Lizenzen:

- ibaPDA-Interface-Codesys-Xplorer (16)

4.2 Sichtbarkeit der Schnittstelle

Ist die Schnittstelle trotz gültiger Lizenz nicht zu sehen, ist sie möglicherweise verborgen.

Überprüfen Sie die Einstellung im Register *Allgemein* im Knoten *Schnittstellen*.

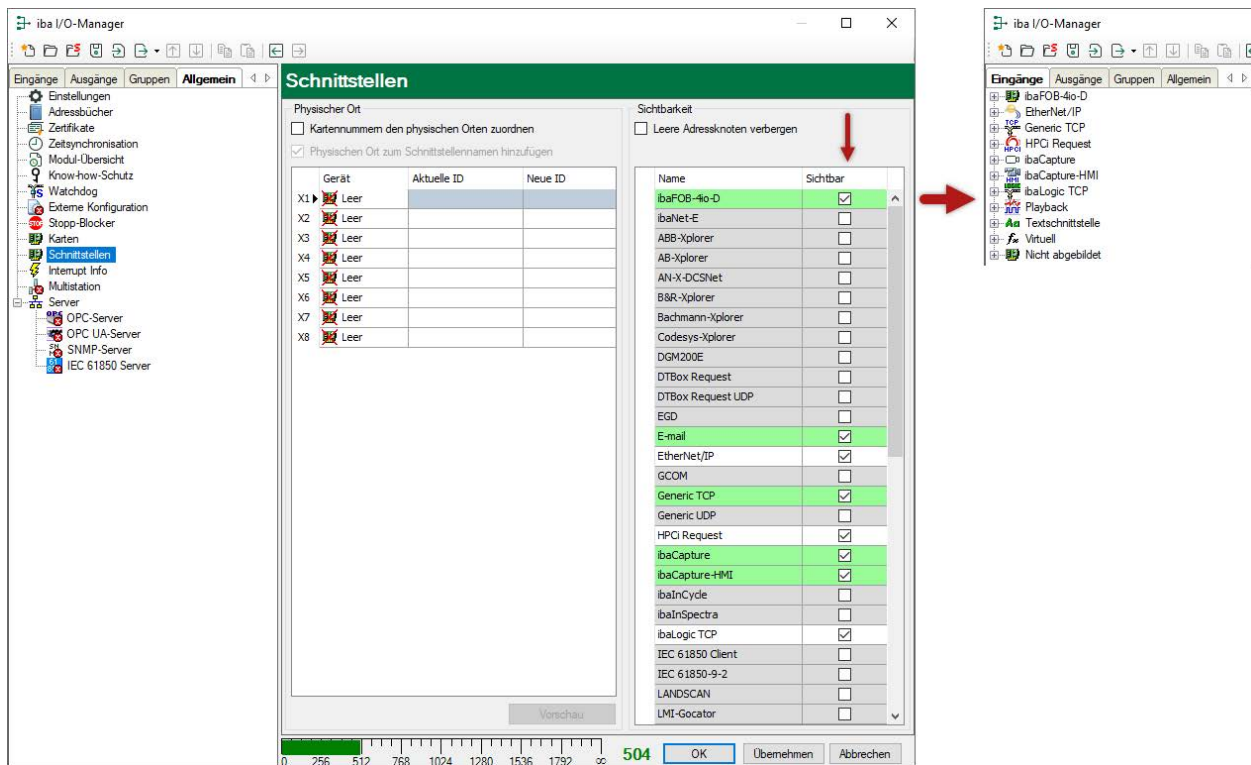
Sichtbarkeit

Die Tabelle *Sichtbarkeit* listet alle Schnittstellen auf, die entweder durch Lizenzen oder installierte Karten verfügbar sind. Diese Schnittstellen sind auch im Schnittstellenbaum zu sehen.

Mithilfe der Häkchen in der Spalte *Sichtbar* können Sie nicht benötigte Schnittstellen im Schnittstellenbaum verbergen oder anzeigen.

Schnittstellen mit konfigurierten Modulen sind grün hinterlegt und können nicht verborgen werden.

Ausgewählte Schnittstellen sind sichtbar, die anderen Schnittstellen sind verborgen:



4.3 Protokolldateien

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

Die Protokolldatei können Sie über den Button <Protokolldatei öffnen> öffnen. 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 von *ibaPDA*-Server (...\[ProgramData\iba\ibaPDA\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.

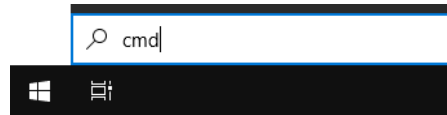
Beispiele:

- [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 Sie überprüfen können, ob ein bestimmter Kommunikationspartner in einem IP-Netzwerk erreichbar ist.

1. Öffnen Sie eine Windows Eingabeaufforderung.



2. 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: Eingabeaufforderung
Microsoft Windows [Version 10.0]
(c) 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: Eingabeaufforderung
Microsoft Windows [Version 10.0]
(c) 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: 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),
    Ca. Zeitangaben in Millisek.:
        Minimum = 0ms, Maximum = 1ms, Mittelwert = 0ms

C:\Windows\system32>
```


4.5 Überprüfung der Verbindung für Empfangstelegramme

Nach Übernahme der Konfiguration werden folgende Daten dargestellt:

Generic TCP

Eigenschaften

TCP-Portliste: Ports in Firewall zulassen

Puffergröße: ▲ ▼ Bytes Statistik zurücksetzen

Telegramme, die nicht abgebildet werden können:

	Adresse	Ziel-Port	Modus	Nachrichten zähler	Unvollständige Fehler	Paketgröße Aktuell	Zeit Aktuell
0	192.168.110.108	5010	Passiv	212	0	19	99,9 ms
1	192.168.110.108	5011	Passiv	212	0	19	101,7 ms
2	192.168.110.108	5012	Passiv	212	0	19	100,0 ms
3	192.168.110.108	5022	Passiv	212	0	19	99,8 ms
4	192.168.110.108	5025	Passiv	212	0	19	100,0 ms
5	192.168.110.108	5023	Passiv	?	?	?	?
6	?	?	?	?	?	?	?
7	?	?	?	?	?	?	?
8	?	?	?	?	?	?	?

Telegrammzähler

Telegramme, die nicht abgebildet werden können

Zählt bei Empfang von Daten einer neuen Verbindung hoch, wenn alle verfügbaren Verbindungseinträge bereits genutzt werden.

Ursache: *ibaPDA* empfängt mehr Verbindungen, als die Lizenz unterstützt.

Verbindungstabelle

Darstellungsvarianten

Grün/Fett	Die Verbindung ist OK und der Verbindung ist ein Modul zugewiesen.
Grün/Normal	Die Verbindung ist OK, aber der Verbindung ist kein Modul zugewiesen.
Grau/Fett	Ein Modul ist definiert, jedoch ist keine Verbindung mit diesen Parametern aufgebaut.

Verbindungsdaten

- IP-Adresse Quelle
- Ziel-Port
- Verbindungsmodus
- Zähler für empfangene Telegramme
- Datenlänge der empfangenen Telegramme

Mögliche Fehler

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

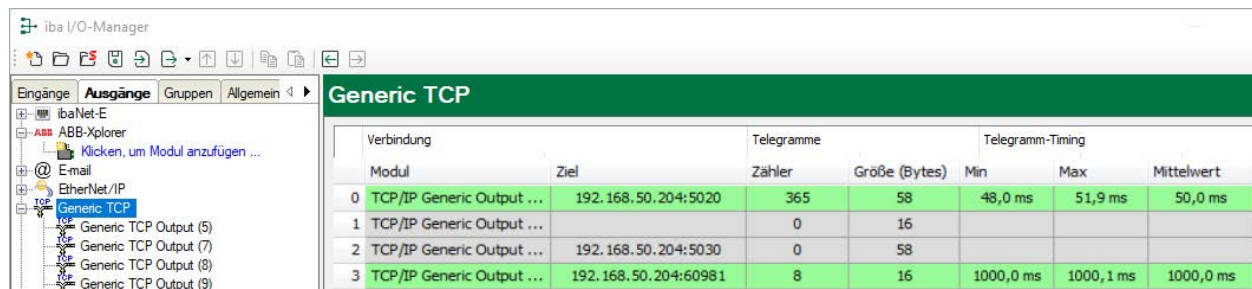
- Controller sendet nicht.
- Es besteht keine Ethernet-Verbindung zwischen *ibaPDA*-PC und Controller.
- Fehler in der Controller-Projektierung:
 - falsche Remote-IP-Adresse
 - Portnummer stimmt nicht mit den Einstellungen von *ibaPDA* überein.
 - Portnummer ist durch die Firewall geblockt.

Weitere Fehler:

- Zählen die Telegrammzähler nicht kontinuierlich hoch, werden auf Controllerseite die Telegramme nicht zyklisch gesendet.
- Ändern sich die Werte für Datenlänge, deutet das auf einen der folgenden Fehler hin:
 - Es werden verschiedene Telegramme mit verschiedenen Layouts über ein und dieselbe Verbindung gesendet.
 - Das "Delayed Acknowledge"-Problem tritt auf (siehe ➤ *Probleme mit TCP-Performance durch Delayed Acknowledge*, Seite 32)

4.6 Überprüfen der Verbindung für Sendetelegramme

Unter der Ausgangsschnittstelle *Generic TCP* finden Sie die Verbindungstabelle.



Generic TCP							
Verbindung		Telegramme		Telegramm-Timing			
Modul	Ziel	Zähler	Größe (Bytes)	Min	Max	Mittelwert	
0 TCP/IP Generic Output ...	192.168.50.204:5020	365	58	48,0 ms	51,9 ms	50,0 ms	
1 TCP/IP Generic Output ...		0	16				
2 TCP/IP Generic Output ...	192.168.50.204:5030	0	58				
3 TCP/IP Generic Output ...	192.168.50.204:60981	8	16	1000,0 ms	1000,1 ms	1000,0 ms	

Darstellungsvarianten

Grün mit Ziel-Adresse	Die Verbindung ist OK und <i>ibaPDA</i> sendet Daten.
Grau mit Ziel-Adresse	Ein aktives Modul ist definiert, jedoch ist kein Empfänger mit dieser IP-Adresse oder Portnummer für den Verbindungsaufbau vorhanden.
Grau ohne Ziel-Adresse	Ein passives Modul ist definiert, jedoch hat der Empfänger keine Verbindung mit dem eingestellten Port aufgebaut. Deswegen ist das Feld für den Partner leer.

Verbindungsdaten, Telegrammdaten und Statistik

- Modulname
- Ziel: IP-Adresse: Portnummer der Verbindung
- Zähler für gesendete Telegramme
- Größe der gesendeten Telegramme (Nutzdaten)
- Telegrammzyklus, minimaler, maximaler und durchschnittlicher Wert

4.7 Diagnosemodule

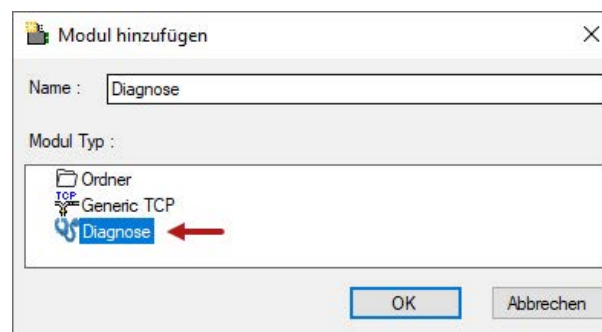
Diagnosemodule sind für die meisten Ethernet-basierten Schnittstellen und Xplorer-Schnittstellen verfügbar. Mit einem Diagnosemodul können Informationen aus den Diagnoseanzeigen (z. B. Diagnoseregister und Verbindungstabellen einer Schnittstelle) als Signale erfasst werden.

Ein Diagnosemodul ist stets einem Datenerfassungsmodul derselben Schnittstelle zugeordnet und stellt dessen Verbindungsinformationen zur Verfügung. Durch die Nutzung eines Diagnosemoduls können die Diagnoseinformationen auch im *ibaPDA*-System durchgängig aufgezeichnet und ausgewertet werden. Diagnosemodule verbrauchen keine Verbindung der Lizenz, da sie keine Verbindung aufbauen, sondern auf ein anderes Modul verweisen.

Nutzungsbeispiele für Diagnosemodule:

- Wenn der Fehlerzähler einer Kommunikationsverbindung einen bestimmten Wert überschreitet oder eine Verbindung abbricht, kann eine Benachrichtigung generiert werden.
- Bei einem Störfall können die aktuellen Antwortzeiten im Telegrammverkehr in einem Störungsreport dokumentiert werden.
- Der Status der Verbindungen kann in *ibaQPanel* visualisiert werden.
- Diagnoseinformationen können über den in *ibaPDA* integrierten SNMP-Server oder OPC DA/UA-Server an übergeordnete Überwachungssysteme wie Netzwerkmanagement-Tools weitergegeben werden.

Wenn für eine Schnittstelle ein Diagnosemodul verfügbar ist, wird im Dialog "Modul hinzufügen" der Modultyp "Diagnose" angezeigt (Beispiel: Generic TCP).



Moduleinstellungen Diagnosemodul



Bei einem Diagnosemodul können Sie folgende Einstellungen vornehmen (Beispiel: Generic TCP):

Die Grundeinstellungen eines Diagnosemoduls entsprechen denen der anderen Module. Es gibt nur eine für das Diagnosemodul spezifische Einstellung, die vorgenommen werden muss: das Zielmodul.

Mit der Auswahl des Zielmoduls weisen Sie das Diagnosemodul dem Modul zu, dessen Verbindungsinformationen erfasst werden sollen. In der Drop-down-Liste der Einstellung stehen die unterstützten Module derselben Schnittstelle zur Auswahl. Pro Diagnosemodul kann genau ein Datenerfassungsmodul zugeordnet werden. Wenn Sie ein Modul ausgewählt haben, werden in den Registern *Analog* und *Digital* umgehend die verfügbaren Diagnosesignale hinzugefügt. Welche Signale das sind, hängt vom Schnittstellentyp ab. Im nachfolgenden Beispiel sind die Analogwerte eines Diagnosemoduls für ein Generic TCP-Modul aufgelistet.

Allgemein Analog Digital						
	Name	Einheit	Gain	Offset	Aktiv	Istwert
0	IP-Adresse (Teil 1)		1	0	<input checked="" type="checkbox"/>	
1	IP-Adresse (Teil 2)		1	0	<input checked="" type="checkbox"/>	
2	IP-Adresse (Teil 3)		1	0	<input checked="" type="checkbox"/>	
3	IP-Adresse (Teil 4)		1	0	<input checked="" type="checkbox"/>	
4	Port		1	0	<input checked="" type="checkbox"/>	
5	Telegrammzähler		1	0	<input checked="" type="checkbox"/>	
6	Unvollständig		1	0	<input checked="" type="checkbox"/>	
7	Paketgröße (aktuell)	Bytes	1	0	<input checked="" type="checkbox"/>	
8	Paketgröße (max)	Bytes	1	0	<input checked="" type="checkbox"/>	
9	Zeit zwischen Daten (aktuell)	ms	1	0	<input checked="" type="checkbox"/>	
10	Zeit zwischen Daten (min)	ms	1	0	<input checked="" type="checkbox"/>	
11	Zeit zwischen Daten (max)	ms	1	0	<input checked="" type="checkbox"/>	

Die IP(v4-)Adresse eines Generic TCP-Moduls, z. B. (siehe Abbildung), wird entsprechend der 4 Bytes bzw. Oktetts in 4 Teile zerlegt, um sie leichter lesen und vergleichen zu können. Andere Größen, wie Portnummer, Zählerstände für Telegramme und Fehler, Datengrößen und Laufzeiten für Telegramme werden ebenfalls ermittelt. Im nachfolgenden Beispiel sind die Digitalwerte eines Diagnosemoduls für ein Generic TCP-Modul aufgelistet.

<div>  Allgemein  Analog  Digital </div>				
Name	Aktiv	Istwert		
0 Aktiver Verbindungsmodus	<input checked="" type="checkbox"/>			
1 Ungültiges Paket	<input checked="" type="checkbox"/>			
2 Verbinde	<input checked="" type="checkbox"/>			
3 Verbunden	<input checked="" type="checkbox"/>			

Diagnosesignale

Abhängig vom Schnittstellentyp stehen folgende Signale zur Verfügung:

Signalname	Bedeutung
Aktualisierungszeit (Istwert/konfiguriert/max/min/Mittelwert)	<p>Gibt die Aktualisierungszeit an, in der die Daten aus der SPS, der CPU oder vom Server abgerufen werden sollen (konfiguriert). Standard ist gleich dem Parameter "Zeitbasis". Während der Messung kann die reale aktuelle Aktualisierungszeit (Istwert) höher sein als der eingestellte Wert, wenn die SPS mehr Zeit zur Übertragung der Daten benötigt. Wie schnell die Daten wirklich aktualisiert werden, können Sie in der Verbindungstabelle überprüfen. Die minimal erreichbare Aktualisierungszeit wird von der Anzahl der Signale beeinflusst. Je mehr Signale erfasst werden, desto größer wird die Aktualisierungszeit.</p> <p>Max/min/Mittelwert: statische Werte der Aktualisierungszeit seit dem letzten Start der Erfassung bzw. Rücksetzen der Zähler</p>
Anforderungen Sendewiederholung	Anzahl der nochmals angeforderten Datentelegramme (in) bei Verlust oder Verspätung
Antwortzeit (aktuell/konfiguriert/max/min/Mittelwert)	<p>Antwortzeit ist die Zeit zwischen Messwertanforderung von <i>ibaPDA</i> und Antwort von der SPS bzw. Empfang der Daten.</p> <p>Aktuell: Istwert</p> <p>Max/min/Mittelwert: statische Werte der Antwortzeit seit dem letzten Start der Erfassung bzw. Rücksetzen der Zähler</p>
Anzahl Anforderungsbefehle	Zähler für Anforderungstelegramme von <i>ibaPDA</i> an die SPS/CPU
Aufgebaute Verbindungen (in)	Anzahl der aktuell gültigen Datenverbindungen für den Empfang
Aufgebaute Verbindungen (out)	Anzahl der aktuell gültigen Datenverbindungen für das Senden
Ausgangsdatenlänge	Länge der Datentelegramme mit Ausgangssignalen in Bytes (<i>ibaPDA</i> sendet)
Datenlänge	Länge der Datentelegramme in Bytes

Signalname	Bedeutung
Datenlänge des Inputs	Länge der Datentelegramme mit Eingangssignalen in Bytes (<i>ibaPDA</i> empfängt)
Datenlänge O->T	Größe des Output-Telegramms in Byte
Datenlänge T->O	Größe des Input-Telegramms in Byte
Definierte Topics	Anzahl der definierten Topics
Empfangene Telegramme seit Konfiguration	Anzahl der empfangenen Datentelegramme (in) seit Beginn der Erfassung
Empfangene Telegramme seit Verbindungsstart	Anzahl der empfangenen Datentelegramme (in) seit Beginn des letzten Verbindungsaufbaus
Empfangszähler	Anzahl der empfangenen Telegramme
Exchange ID	ID des Datenaustauschs
Falscher Telegrammtyp	Anzahl der Empfangstelegramme mit falschem Telegrammtyp
Fehlerzähler	Zähler der Kommunikationsfehler
Gepufferte Anweisungen	Anzahl der noch nicht ausgeführten Anweisungen im Zwischenspeicher
Gepufferte Anweisungen sind verloren	Anzahl der gepufferten aber nicht ausgeführten und verlorenen Anweisungen
Gesendete Telegramme seit Konfiguration	Anzahl der gesendeten Datentelegramme (out) seit Beginn der Erfassung
Gesendete Telegramme seit Verbindungsstart	Anzahl der gesendeten Datentelegramme (out) seit Beginn des letzten Verbindungsaufbaus
ID der Verbindung O->T	ID der Verbindung für Output-Daten (vom Zielsystem an <i>ibaPDA</i>) Entspricht der Assembly-Instanznummer
ID der Verbindung T->O	ID der Verbindung für Input-Daten (von <i>ibaPDA</i> an Zielsystem) Entspricht der Assembly-Instanznummer
IP-Adresse (Teil 1-4)	4 Oktets der IP-Adresse des Zielsystems
IP-Quelladresse (Teil 1-4) O->T	4 Oktets der IP-Adresse des Zielsystems Output-Daten (vom Zielsystem an <i>ibaPDA</i>)
IP-Quelladresse (Teil 1-4) T->O	4 Oktets der IP-Adresse des Zielsystems Input-Daten (von <i>ibaPDA</i> an Zielsystem)
IP-Zieladresse (Teil 1-4) O->T	4 Oktets der IP-Adresse des Zielsystems Output-Daten (vom Zielsystem an <i>ibaPDA</i>)
IP-Zieladresse (Teil 1-4) T->O	4 Oktets der IP-Adresse des Zielsystems Input-Daten (von <i>ibaPDA</i> an Zielsystem)
KeepAlive-Zähler	Anzahl der vom OPC UA-Server empfangenen KeepAlive-Telegramme
Lesezähler	Anzahl der Lesezugriffe/Datenanforderungen
Multicast Anmeldefehler	Anzahl der Fehler bei Multicast-Anmeldung
Paketgröße (aktuell)	Größe der aktuell empfangenen Telegramme
Paketgröße (max)	Größe des größten empfangenen Telegramms

Signalname	Bedeutung
Ping-Zeit (Istwert)	Antwortzeit für ein Ping-Telegramm
Port	Portnummer für die Kommunikation
Producer ID (Teil 1-4)	Producer-ID als 4 Byte unsigned Integer
Profilzähler	Anzahl der vollständig erfassten Profile
Pufferdateigröße (aktuell/mittl./max)	Größe der Pufferdatei zum Zwischenspeichern der Anweisungen
Pufferspeichergröße (aktuell/mittl./max)	Größe des belegten Arbeitsspeichers zum Zwischenspeichern der Anweisungen
Schreibverlustzähler	Anzahl missglückter Schreibzugriffe
Schreibzähler	Anzahl erfolgreicher Schreibzugriffe
Sendezähler	Anzahl der Sendetelegramme
Sequenzfehler	Anzahl Sequenzfehler
Synchronisation	Gerät wird für die isochrone Erfassung synchronisiert
Telegramme pro Zyklus	Anzahl der Telegramme im Zyklus der Aktualisierungszeit
Telegrammzähler	Anzahl der empfangenen Telegramme
Topics aktualisiert	Anzahl der aktualisierten Topics
Trennungen (in)	Anzahl der aktuell unterbrochenen Datenverbindungen für den Empfang
Trennungen (out)	Anzahl der aktuell unterbrochenen Datenverbindungen für das Senden
Unbekannter Sensor	Anzahl unbekannter Sensoren
Ungültiges Paket	Ungültiges Datenpaket erkannt
Unvollständig	Anzahl unvollständiger Telegramme
Verarbeitete Anweisungen	Anzahl der ausgeführten SQL-Anweisungen seit dem letzten Start der Erfassung
Verbinde	Verbindung wird aufgebaut
Verbindungsphase (in)	Zustand der ibaNet-E Datenverbindung für den Empfang
Verbindungsphase (out)	Zustand der ibaNet-E Datenverbindung für das Senden
Verbindungsversuche (in)	Anzahl der Versuche, die Empfangsverbindung (in) aufzubauen
Verbindungsversuche (out)	Anzahl der Versuche, die Sendeverbindung (out) aufzubauen
Verbunden	Verbindung ist aufgebaut
Verbunden (in)	Eine gültige Datenverbindung für den Empfang (in) ist vorhanden
Verbunden (out)	Eine gültige Datenverbindung für das Senden (out) ist vorhanden
Verlorene Images	Anzahl der verlorenen Images (in), die selbst nach einer Sendewiederholung nicht empfangen wurden
Verlorene Profile	Anzahl unvollständiger/fehlerhafter Profile

Signalname	Bedeutung
Zeilen (letzte)	Anzahl der Ergebniszeilen der letzten SQL-Abfrage (innerhalb der projektierten Anzahl von Ergebniszeilen)
Zeilen (Maximum)	Höchste Anzahl der Ergebniszeilen einer SQL-Abfrage seit dem letzten Start der Erfassung (maximal gleich der projektierten Anzahl von Ergebniszeilen)
Zeit zwischen Daten (aktuell/max/min)	Zeit zwischen zwei korrekt empfangenen Telegrammen Aktuell: zwischen den letzten zwei Telegrammen Max/min: statistische Werte seit Start der Erfassung oder Rücksetzen der Zähler
Zeit-Offset (Istwert)	Gemessene Zeitdifferenz der Synchronität zwischen dem ibaNet-E-Gerät und <i>ibaPDA</i>

5 Anhang

5.1 Fehlerbehebung

5.1.1 Probleme mit TCP-Performance durch Delayed Acknowledge

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.

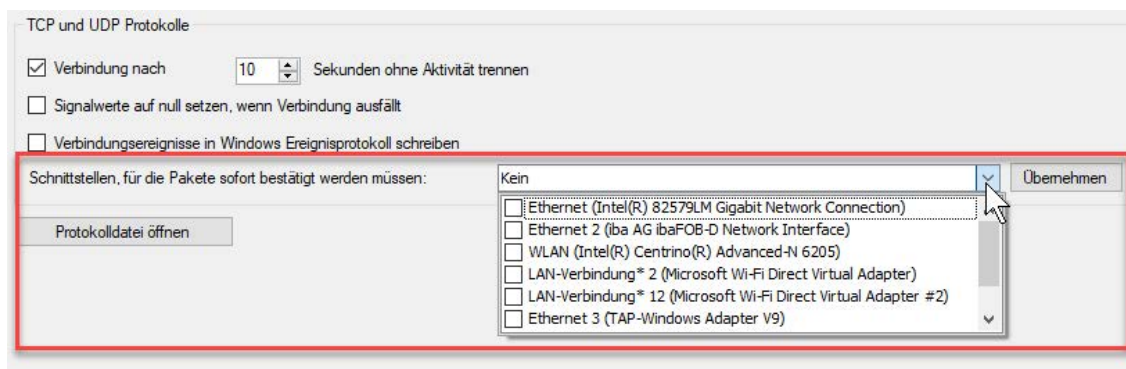
Der Datenfluss wird durch ein "Sliding Window" (Parameter Win=nnnn) gesteuert. Der Empfänger gibt an, wie viele Bytes er empfangen kann ohne eine Quittung zu senden.

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:

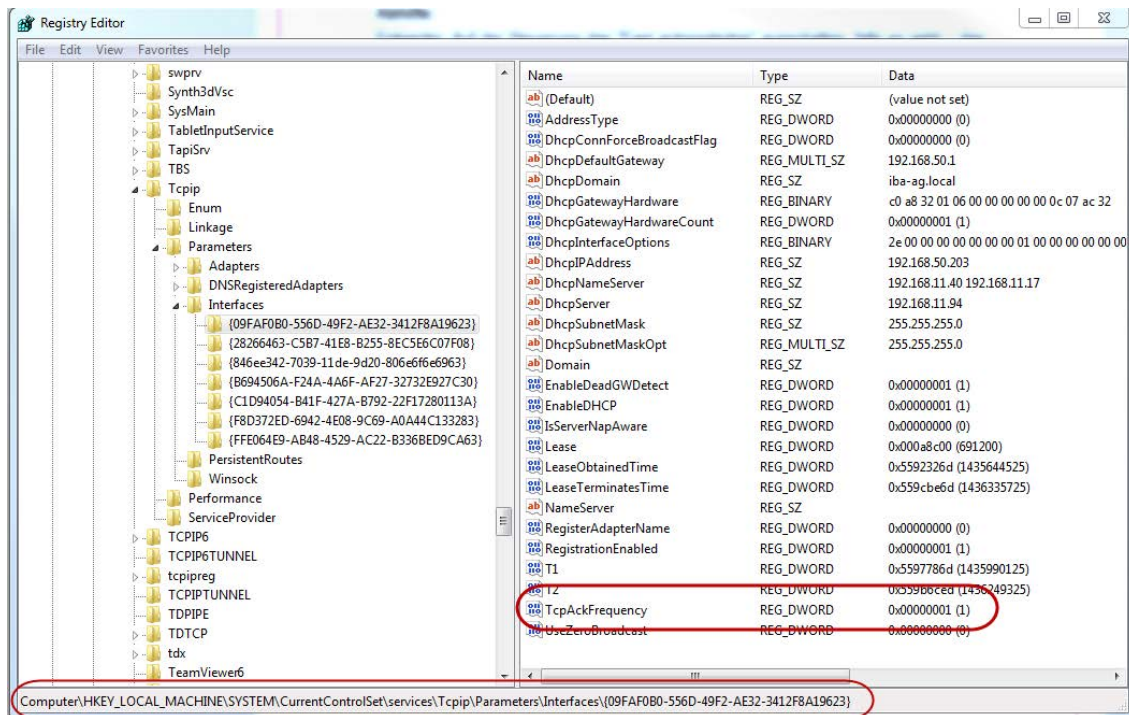
Das "Delayed Acknowledge" lässt sich einzeln pro Netzwerkadapter über einen Eintrag in der Windows Registry abschalten. Zur einfachen Änderung bietet *ibaPDA* im IO-Manager unter *Allgemein* im Register *Einstellungen* einen entsprechenden Dialog.

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.

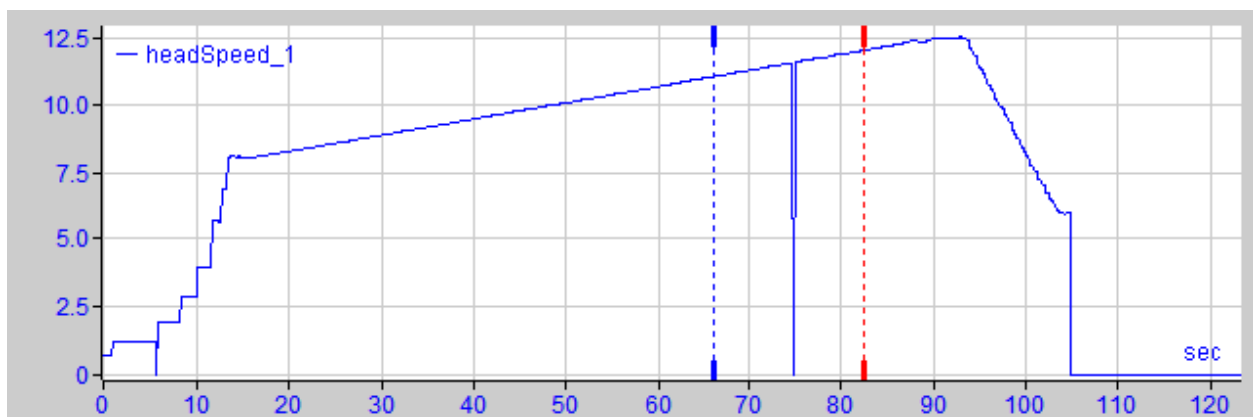
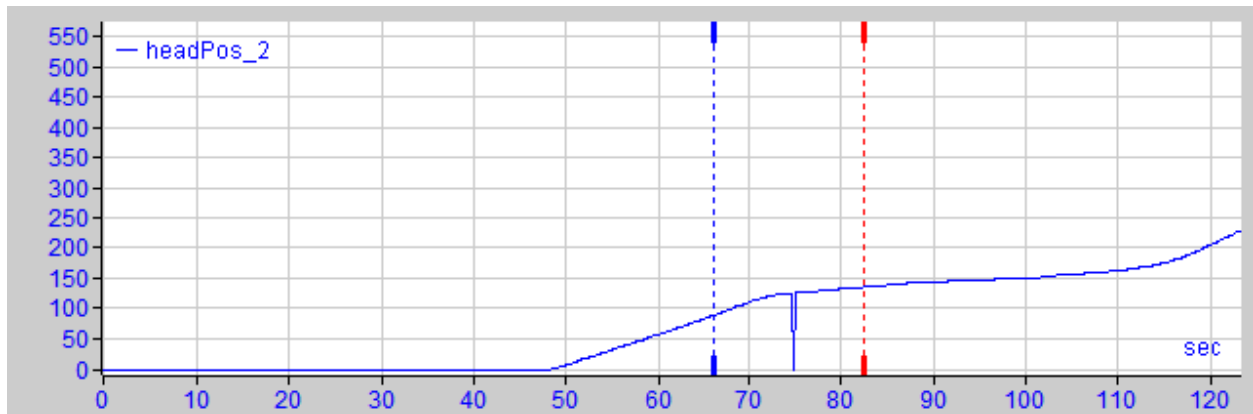
5.1.2 Unbrauchbare TCP-Daten als Folge des Nagle-Algorithmus

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 32) 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 32 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 Nummer des Lizenzcontainers 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.