

# Network Interface Appendix Anybus® CompactCom EtherNet/IP Functionality 2-Port

Doc.Id. JCM-1201-034

Rev. 1.21



HALMSTAD • CHICAGO • KARLSRUHE • TOKYO • BEIJING • MILANO • MULHOUSE • COVENTRY • PUNE • COPENHAGEN

---

HMS Industrial Networks  
Mailing address: Box 4126, 300 04 Halmstad, Sweden  
Visiting address: Stationsgatan 37, Halmstad, Sweden

E-mail: [info@hms-networks.com](mailto:info@hms-networks.com)  
Web: [www.anybus.com](http://www.anybus.com)

# 重要なユーザ情報

このマニュアルは、EtherNet/IP によって提供される機能について十分にご理解いただくことを意図しています。このマニュアルは、Anybus CompactCom EtherNet/IP w.IT Functionality 2-Port 特有の機能についてのみ説明します。Anybus CompactCom に関する概要については、Anybus CompactCom の設計ガイドを参照してください。

このマニュアルの読者は、ハイレベルのソフトウェア設計および一般的な通信システムに精通していることを前提としています。高度な EtherNet/IP 特有の機能には、EtherNet/IP ネットワーキング内部に関する詳細な知識または公式な EtherNet/IP 仕様からの情報（あるいはその両方）が必要になることがあります。このような場合、本製品の実装を担当するユーザは、十分な知識を獲得するために EtherNet/IP 仕様を取得するか、これが必要とされないような方法に実装を制限する必要があります。

## 責任

このマニュアルはあらゆる点を考慮して作成しています。不正確な記述や記載漏れがあった場合には、HMS Industrial Networks AB までご連絡ください。このマニュアルに含まれるデータや説明には拘束力がありません。HMS Industrial Networks AB は、継続的な製品開発を旨とする当社のポリシーに則って、弊社の製品を改良する権利を留保します。このマニュアルに含まれる情報は予告なく変更される場合があります。ただし、HMS Industrial Networks AB は変更に関して義務を負うものではありません。HMS Industrial Networks AB はこのマニュアルに現れるあらゆるエラーに対して責任を負いません。

この製品には多くのアプリケーションがあります。この装置の使用責任者は、アプリケーションが該当する法律、規則、規定、および規格を含む全ての性能および安全要求事項を満たしており、これを確認するために全ての必要な手順がとられたことを保証する必要があります。

HMS Industrial Networks AB は、いかなる状況においても、文書化されていない機能の使用、タイミング、またはこの製品の文書化された範囲外で見つかった機能面での副次的な影響によって発生する可能性がある問題に対する義務または責任を負いません。製品のこのような側面の直接的または間接的な使用によって発生する結果は不明確であり、互換性の問題や安定性の問題などを含む可能性があります。

このマニュアルに含まれる例および図表は、説明のためにのみ記載されています。特定の実装には多くの要素や要件が関連しているため、HMS Industrial Networks AB は、これらの例や図表に基づいた実際の使用に対する責任を負いません。

## 知的財産権

HMS Industrial Networks AB は、このマニュアルに記載された製品に組み入れられた技術に関する知的財産権を所有します。これらの知的財産権には、米国およびその他の国での特許および出願中の特許が含まれる可能性があります。

## 商標について

Anybus® は、HMS Industrial Networks AB の登録商標です。その他の全ての商標は、各所有者の資産です。

**警告：** これはクラス A 製品です。国内の環境では、この製品は無線妨害を発生させる可能性があります。この場合、ユーザは適切な対策をとる必要があります。

**ESD に関する注意：** この製品は ESD (Electrostatic Discharge : 静電気放電) に敏感な部分が含まれているため、ESD 対策が十分でない場合には破損する可能性があります。製品を直接手で扱うときは静電気対策が必要です。これらを行わないと製品を破損させる可能性があります。

## 目次

	<b>重要なユーザ情報</b>	
	責任.....	1
	知的財産権.....	1
	商標について.....	1
はじめに	このマニュアルについて	
	関連マニュアル.....	7
	マニュアル更新履歴.....	7
	慣例と用語集.....	8
	販売およびサポート.....	9
<b>第1章</b>	<b>Anybus-CompactCom EtherNet/IP 2-port について</b>	
	概要.....	10
	特長.....	10
	正面図.....	11
	ネットワークコネクタ、ブリックバージョン.....	13
<b>第2章</b>	<b>チュートリアル</b>	
	はじめに.....	14
	フィールドバス適合性に関する注意.....	14
	コンFORMANCEテストガイド.....	14
	お客様製品の <b>Identity</b> 情報確定.....	15
	工場出荷時設定リセット.....	15
	<b>IP</b> アドレス.....	16
<b>第3章</b>	<b>基本操作</b>	
	概要.....	17
	ソフトウェアの要件.....	17
	デバイスカスタマイズ.....	18
	ネットワーク <b>Identity</b> .....	18
	電子的データシート ( <b>EDS</b> ).....	18
	<b>EtherNet/IP</b> および <b>CIP</b> の実装.....	19
	<b>Web</b> インターフェース.....	19
	ソケットインターフェース (上級ユーザのみ).....	19
	通信設定.....	20
	診断.....	20
	ネットワークのデータ交換.....	21
	アプリケーションデータ ( <b>ADI</b> ).....	21
	プロセスデータ.....	21
	データタイプの変換.....	21
	ファイルシステム.....	22
	概要.....	22
	システムファイル.....	22

---

第 4 章	FTP サーバ	
	概要.....	23
	ユーザアカウント.....	23
	セッションの例.....	24
第 5 章	Web サーバ	
	概要.....	25
	デフォルトの Web ページ.....	25
	ネットワークのコンフィグレーション.....	26
	イーサネット統計ページ.....	28
	サーバコンフィグレーション.....	29
	概要.....	29
	インデックスページ.....	29
	デフォルトのコンテンツタイプ.....	30
	認証.....	30
第 6 章	電子メールクライアント	
	概要.....	32
	電子メールメッセージの送信方法.....	32
第 7 章	SSI (Server Side Include)	
	概要.....	33
	Include File.....	33
	コマンド関数.....	34
	概要.....	34
	<i>GetConfigItem()</i> .....	35
	<i>SetConfigItem()</i> .....	36
	<i>SsiOutput()</i> .....	38
	<i>DisplayRemoteUser</i> .....	38
	<i>ChangeLanguage()</i> .....	39
	<i>IncludeFile()</i> .....	40
	<i>SaveDataToFile()</i> .....	41
	<i>printf()</i> .....	42
	<i>scanf()</i> .....	44
	引数の関数.....	47
	概要.....	47
	<i>ABCCMessage()</i> .....	47
	SSI 出力コンフィグレーション.....	52

---

<b>第 8 章</b>	<b>CIP オブジェクト</b>	
	概要.....	53
	Identity オブジェクト (01h) .....	54
	メッセージルータ (02h) .....	57
	アセンブリオブジェクト (04h) .....	58
	コネクションマネージャ (06h) .....	61
	パラメータオブジェクト (0Fh).....	64
	DLR オブジェクト (47h) .....	68
	QoS オブジェクト (48h).....	69
	ADI オブジェクト (A2h).....	70
	ポートオブジェクト (F4h).....	72
	TCP/IP インターフェースオブジェクト (F5h) .....	74
	Ethernet リンクオブジェクト (F6h) .....	78
<b>第 9 章</b>	<b>Anybus モジュールオブジェクト</b>	
	概要.....	82
	Anybus オブジェクト (01h) .....	83
	診断オブジェクト (02h).....	85
	ネットワークオブジェクト (03h).....	86
	ネットワークコンフィグレーションオブジェクト (04h) .....	87
	ソケットインターフェースオブジェクト (07h) .....	95
	SMTP クライアントオブジェクト (09h) .....	115
	ファイルシステム・インターフェースオブジェクト (0Ah) .....	120
	ネットワークイーサネットオブジェクト (0Ch) .....	133
	CIP ポートコンフィグレーションオブジェクト (0Dh) .....	134
<b>第 10 章</b>	<b>ホストアプリケーションオブジェクト</b>	
	概要.....	136
	EtherNet/IP ホストオブジェクト (F8h) .....	137
	Ethernet ホストオブジェクト (F9h) .....	147
<b>アペンディックス A</b>	<b>機能のカテゴリ化</b>	
	基本.....	150
	拡張.....	150
	高度.....	150
<b>アペンディックス B</b>	<b>実装の詳細</b>	
	拡張 LED 機能.....	151
	SUP ビットの定義.....	151

---

Anybus ステートマシン .....	152
アプリケーション・ウォッチドッグタイムアウトの処理.....	152
<b>アペンディックス C メッセージのセグメント化</b>	
概要.....	153
コマンドのセグメント化.....	154
レスポンスのセグメント化.....	155
<b>アペンディックス D HICP (ホスト IP コンフィグレーションプロトコル)</b>	
概要.....	156
操作.....	156
<b>アペンディックス E 技術仕様</b>	
保護接地 (PE) 要件.....	157
電源供給.....	157
環境仕様.....	157
EMC 適合性 .....	157
<b>アペンディックス F 著作権表示</b>	

## P. このマニュアルについて

詳細な情報や資料などについては、HMS の Web サイト 'www.anybus.com' を参照してください。

### P.1 関連マニュアル

マニュアル名	作成者
Anybus-CompactCom Software Design Guide	HMS
Anybus-CompactCom Hardware Design Guide	HMS
Anybus-CompactCom Software Driver User Guide	HMS

### P.2 マニュアル更新履歴

#### 最新の更新 (1.20 .....1.21)

変更内容	ページ
セグメント化が必要な場合のデータサイズの修正	143、146
Get_Instance_Number_By_Order へのコマンド名の修正	17、18
ネットワークコンフィグレーションオブジェクト内のインスタンス #13 のアトリビュート #5 の説明の修正	92
HICP の説明のアップデート	156

#### 改定版リスト

改定番号	改定日	作成者	章	説明
1.00	2010-03-25	KeL	全て	最初のリリース
1.01	2011-02-10	KeL	G、P、 9、1、8、 5、C	マイナー修正およびアップデート
1.02	2011-08-08	KaD	5	マイナー追加およびアップデート
1.03	2012-01-30	KeL	2	マイナーアップデート
1.04	2012-03-07	KaD	2、8	マイナー修正およびアップデート
1.10	2012-09-12	KeL、 KaD	1、8、9、 10、B	新しいコネクタおよびマイナー修正
1.20	2012-12-04	KeL	1、10	ブリックバージョンによるアップデート、オブジェクトの説明のマイナーアップデート
1.21	2013-02-20	KeL	3、9、10	マイナー修正

## P.3 慣例と用語集

このマニュアルでは下記の慣例を使用しています。

- 番号が付いたリストは、連続した手順を示します。
- 黒丸が付いたリストは、手順ではなく情報を示します。
- 'Anybus' または 'モジュール' という用語は、Anybus-CompactCom モジュールを意味します。
- 'ホスト' または 'ホストアプリケーション' という用語は、Anybus モジュールをホストする装置を意味します。
- 16 進値は NNNNh というフォーマットで書かれます。NNNN は 16 進法の値です。



## P.4 販売およびサポート

Sales		Support	
<b>HMS Sweden (Head Office)</b>			
E-mail:	sales@hms-networks.com	E-mail:	support@hms-networks.com
Phone:	+46 (0) 35 - 17 29 56	Phone:	+46 (0) 35 - 17 29 20
Fax:	+46 (0) 35 - 17 29 09	Fax:	+46 (0) 35 - 17 29 09
Online:	www.anybus.com	Online:	www.anybus.com
<b>HMS North America</b>			
E-mail:	us-sales@hms-networks.com	E-mail:	us-support@hms-networks.com
Phone:	+1-312 - 829 - 0601	Phone:	+1-312-829-0601
Toll Free:	+1-888-8-Anybus	Toll Free:	+1-888-8-Anybus
Fax:	+1-312-629-2869	Fax:	+1-312-629-2869
Online:	www.anybus.com	Online:	www.anybus.com
<b>HMS Germany</b>			
E-mail:	ge-sales@hms-networks.com	E-mail:	ge-support@hms-networks.com
Phone:	+49 (0) 721-989777-000	Phone:	+49 (0) 721-989777-300
Fax:	+49 (0) 721-989777-010	Fax:	+49 (0) 721-989777-010
Online:	www.anybus.de	Online:	www.anybus.de
<b>HMS Japan</b>			
E-mail:	jp-sales@hms-networks.com	E-mail:	jp-support@hms-networks.com
Phone:	+81 (0) 45-478-5340	Phone:	+81 (0) 45-478-5340
Fax:	+81 (0) 45-476-0315	Fax:	+81 (0) 45-476-0315
Online:	www.anybus.jp	Online:	www.anybus.jp
<b>HMS China</b>			
E-mail:	cn-sales@hms-networks.com	E-mail:	cn-support@hms-networks.com
Phone:	+86 (0) 10-8532-3183	Phone:	+86 (0) 10-8532-3023
Fax:	+86 (0) 10-8532-3209	Fax:	+86 (0) 10-8532-3209
Online:	www.anybus.cn	Online:	www.anybus.cn
<b>HMS Italy</b>			
E-mail:	it-sales@hms-networks.com	E-mail:	it-support@hms-networks.com
Phone:	+39 039 59662 27	Phone:	+39 039 59662 27
Fax:	+39 039 59662 31	Fax:	+39 039 59662 31
Online:	www.anybus.it	Online:	www.anybus.it
<b>HMS France</b>			
E-mail:	fr-sales@hms-networks.com	E-mail:	fr-support@hms-networks.com
Phone:	+33 (0) 3 68 368 034	Phone:	+33 (0) 3 68 368 033
Fax:	+33 (0) 3 68 368 031	Fax:	+33 (0) 3 68 368 031
Online:	www.anybus.fr	Online:	www.anybus.fr
<b>HMS UK &amp; Eire</b>			
E-mail:	uk-sales@hms-networks.com	E-mail:	support@hms-networks.com
Phone:	+44 (0) 1926 405599	Phone:	+46 (0) 35 - 17 29 20
Fax:	+44 (0) 1926 405522	Fax:	+46 (0) 35 - 17 29 09
Online:	www.anybus.co.uk	Online:	www.anybus.com
<b>HMS Denmark</b>			
E-mail:	dk-sales@hms-networks.com	E-mail:	support@hms-networks.com
Phone:	+45 (0) 35 38 29 00	Phone:	+46 (0) 35 - 17 29 20
Fax:	+46 (0) 35 17 29 09	Fax:	+46 (0) 35 - 17 29 09
Online:	www.anybus.com	Online:	www.anybus.com
<b>HMS India</b>			
E-mail:	in-sales@hms-networks.com	E-mail:	in-support@hms-networks.com
Phone:	+91 (0) 20 40111201	Phone:	+91 (0) 20 40111201
Fax:	+91 (0) 20 40111105	Fax:	+91 (0) 20 40111105
Online:	www.anybus.com	Online:	www.anybus.com

# 1. Anybus-CompactCom EtherNet/IP 2-port について

## 1.1 概要

Anybus-CompactCom EtherNet/IP 2-port 通信モジュールは、特許取得の Anybus-CompactCom ホストインターフェースを通して、イーサネットおよび EtherNet/IP 接続を提供します。この標準をサポートする任意のデバイスはモジュールによって提供される機能を活用することができ、ネットワークタイプに関係なくシームレスなネットワーク統合を可能にします。モジュールは、リニアおよびリングの両方のネットワークトポロジ（DLR、Device Level Ring：デバイスレベルリング）をサポートします。

Anybus-CompactCom プラットフォームのモジュールアプローチを使用すると、CIP オブジェクトの実装を拡張して、特定のアプリケーション要件を満たすことができます。さらに、Identity オブジェクトをカスタマイズできるため、最終製品を汎用の Anybus モジュールではなくベンダ固有の実装とすることができます。

この製品は、"Anybus-CompactCom Hardware Design Guide" および "Anybus-CompactCom Software Design Guide" 内で定義されたアクティブなモジュールのホストインターフェースの全側面に適合するため、この仕様に従った他の任意のデバイスと完全な互換性があります。一般的に、追加のネットワーク関連のソフトウェアサポートは必要ありませんが、高度なネットワーク特有の機能を完全に活用できるようにするためには、ある程度の専用のソフトウェアサポートが必要になる場合があります。

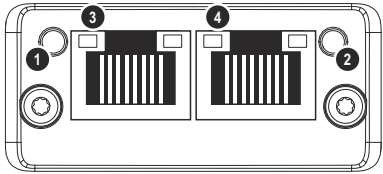
## 1.2 特長

- 2つの EtherNet/IP ポート
- イーサネットまたは M12 コネクタ
- ブリックバージョン
- デバイスレベルリング（DLR）およびリニアネットワークトポロジに対応
- 直流的に絶縁されたバスエレクトロニクス
- 10/100Mbit、全二重 / 半二重での動作
- カスタマイズ可能なコンテンツが含まれた Web サーバ
- FTP サーバ
- 電子メールクライアント
- SSI（Server Side Include）機能
- カスタマイズ可能な Identity 情報
- 最大 65535 の ADI
- CIP パラメータオブジェクトのサポート
- 拡張可能な CIP オブジェクト実装
- 非コネクション型 CIP ルーティングのサポート
- 透過的なソケットインターフェース

## 1.3 正面図

### イーサネットコネクタ

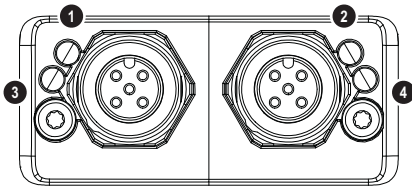
#	アイテム
1	ネットワークステータス LED <sup>a</sup>
2	モジュールステータス LED <sup>a</sup>
3	リンク/アクティビティ LED (ポート 1)
4	リンク/アクティビティ LED (ポート 2)



a. テストシーケンスは、起動時にネットワークおよびモジュールステータス LED で実行されます。

### M12 コネクタ

#	アイテム
1	ネットワークステータス LED <sup>a</sup>
2	モジュールステータス LED <sup>a</sup>
3	リンク/アクティビティ LED (ポート 1)
4	リンク/アクティビティ LED (ポート 2)



a. テストシーケンスは、起動時にネットワークおよびモジュールステータス LED で実行されます。

### ネットワークステータス LED

注意：テストシーケンスは、起動時にこの LED で実行されます。

LED の状態	説明
消灯	電源オフ、または IP アドレスなし
緑点灯	オンライン、1 つ以上の接続が確立済み (CIP Class 1 または 3)
緑点滅	オンライン、接続未確立
赤点灯	重複 IP アドレス、致命的なエラー
赤点滅	1 つ以上の接続がタイムアウト (CIP Class 1 または 3)

### モジュールステータス LED

注意：テストシーケンスは、起動時にこの LED で実行されます。

LED の状態	説明
消灯	電源オフ
緑点灯	Run 状態のスキヤナによって制御されている
緑点滅	設定されていない、またはスキヤナはアイドル状態
赤点灯	メジャーフォルト (EXCEPTION 状態、致命的なエラーなど)
赤点滅	回復可能な障害

### リンク / アクティビティ LED 3/4

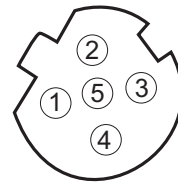
LED の状態	説明
消灯	リンクなし、アクティビティなし
緑点灯	リンク (100Mbit/s) 確立済み
緑点滅	アクティビティ (100Mbit/s)
黄点灯	リンク (10Mbit/s) 確立済み
黄点滅	アクティビティ (10Mbit/s)

### イーサネットインターフェース

イーサネットインターフェースは、10/100Mbit、全二重または半二重での動作をサポートします。

### M12 コネクタ、コード D

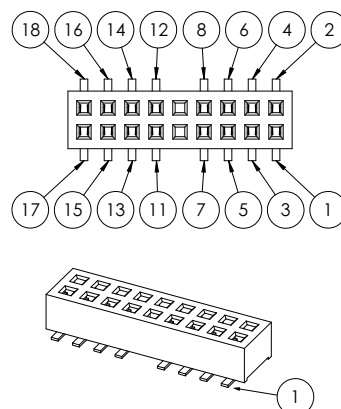
ピン	名前	説明
1	TXD+	送信 +
2	RXD+	受信 +
3	TXD-	送信 -
4	RXD-	受信 -
5 (スレッド)	シールド	シールド



## 1.4 ネットワークコネクタ、ブリックバージョン

Anybus CompactCom EtherNet/IP w.IT Functionality 2-Port はブリックバージョンでも入手できますが、キャリアボード（ホストデバイス）にはフィールドバスコネクタではなく、ピンコネクタが付属します。コンセプトおよびアセンブリは、"Anybus CompactCom Mounting Kit Appendix" (Doc. Id. HMSI-168-30) に記載されています。

ピン No.	信号	ポート No.
1	シールド	2
2	TXD+	
3	TXD-	
4	シールド	
5	シールド	
6	RXD-	
7	RXD+	
8	シールド	
11	シールド	1
12	TXD+	
13	TXD-	
14	シールド	
15	シールド	
16	RXD-	
17	RXD+	
18	シールド	



## 2. チュートリアル

### 2.1 はじめに

この章は、"Anybus CompactCom Implementation Tutorial" を補足するものです。ABCC チュートリアルでは、Anybus CompactCom の実装における簡単な例を取り上げて説明します。この章には、ホストアプリケーションが稼働し、EtherNet/IP ネットワーク上での使用を認定できるようにするのに必要なネットワーク特有の設定が含まれています。

### 2.2 フィールドバス適合性に関する注意

- ODVA は、全ての製造業者が独自のベンダ ID を使用することを求めています。ベンダ ID は、ODVA から申請することができます。
- Anybus-CompactCom EtherNet/IP は、ODVA の認定された独立テストラボによって事前に適合性がテストされており、ODVA Conformance Test Software に準拠しています。ただし、ODVA のコンフォーマンステストポリシーに従い、フィールドバス適合性を確保するために最終製品の適合性をテストする必要があります。これを実行できるようにするには、EtherNet/IP ホストオブジェクト内のベンダ情報をカスタマイズする必要があります。
- さらに、汎用の Anybus モジュールではなく、ベンダ固有の実装として示すことができるように、Identity オブジェクト (CIP) 内の情報もカスタマイズすることを強くお勧めします。

### 2.3 コンフォーマンステストガイド

全てのパラメータのデフォルト設定を使用している場合、Anybus CompactCom EtherNet/IP 2-Port モジュールは、ネットワーク適合性に対して事前認定されています。この事前認定はお客様の製品を認定できるようにするために実施されますが、このことはお客様の製品に対して認定が必要なくなるという意味ではありません。

HMS によって提供される EDS ファイル内のパラメータでの任意の変更には、認定が必要になります。ベンダ ID は ODVA から取得でき、認定には必須です。このセクションでは、ODVA によって設定されたネットワーク認定の要求に準拠するために、Anybus CompactCom EtherNet/IP w.IT Functionality 2-Port モジュールが含まれたお客様の製品に対するコンフォーマンステストを成功させるためのガイドを提供します。

選択された操作モードとは関係なく、このセクションで説明されるアクションは、認定プロセスにおいて考慮される必要があります。製品の Identity は、お客様の会社およびデバイスに合致させるために変更する必要があります。

---

**重要：**このセクションでは、認定に何が 필요한のかについて、そのガイドラインと例を提供します。お客様のアプリケーションの機能によっては、追加のステップを実行する必要がある場合があります。詳細については、HMS Industrial Networks ([www.anybus.com](http://www.anybus.com)) までご連絡ください。

### 2.3.1 お客様製品の Identity 情報確定

Anybus オブジェクト (01h) で "Setup Complete" アトリビュートを正常に設定した後、Anybus モジュールは EtherNet/IP ホストオブジェクト (F8h) およびイーサネットホストオブジェクト (F9h) から Identity データを求めます。従って、以下のリストに掲載したアトリビュートを実装して、適切な値が返される必要があります。

オブジェクト/ インスタンス	アトリビュート	説明	デフォルト	お客様のサンプル	コメント
EtherNet/IP ホスト オブジェクト (F8h)、インスタ ンス 1	#1、Vendor ID	このアトリビュートを使用してデバイスのベンダ ID を設定します。	Vendor ID:005Ah	Vendor ID:1111h	この情報は、EDS ファイル内の "Device" セクションのキーワード値に一致する必要があります。
EtherNet/IP ホスト オブジェクト (F8h)、インスタ ンス 1	#2、Device Type <sup>a</sup>	このアトリビュートを使用してデバイスのデバイスタイプを設定します。	Device Type:0000h (HMS)	Device Type:002Bh <sup>a</sup>	
EtherNet/IP ホスト オブジェクト (F8h)、インスタ ンス 1	#3、Product Code	このアトリビュートを使用してデバイスの製品コードを設定します。	002Eh	2222h	
EtherNet/IP ホスト オブジェクト (F8h)、インスタ ンス 1	#4、Revision	このアトリビュートを使用してデバイスのリビジョンを設定します。		1.1	
EtherNet/IP ホスト オブジェクト (F8h)、インスタ ンス 1	#5、Serial Number	このアトリビュートを使用してデバイスのシリアルナンバーを設定します。		12345678h	同じ 'ベンダ ID' を使用して製造された全ての CIP デバイスに一意の番号です。
EtherNet/IP ホスト オブジェクト (F8h)、インスタ ンス 1	#6、Product Name	このアトリビュートを使用してデバイスの製品名を設定します。	Anybus-CC EtherNet/IP (2- Port)	"Widget"	この情報は、EDS ファイル内の "Device" セクションのキーワード値に一致する必要があります。
イーサネットホスト オブジェクト (F9h)、インスタ ンス 1	#1、MAC address	このアトリビュートを使用してデバイスの MAC アドレスを設定します。		00-11-22-33-44- 55	IEEE から取得された範囲での 6 バイトの物理アドレス値です。

a. コンフォーマンステストに合格するためには、デバイスタイプのデフォルト値 0000h をモジュールに対して変更する必要があります。他の特定プロファイルが実装されない場合、値 002Bh (Generic Device (keyable)) を使用してください。

### 2.3.2 工場出荷時設定リセット

#### アプリケーションオブジェクト (FFh) への Reset コマンドがサポートされる必要がある

Anybus CompactCom EtherNet/IP w.IT Functionality 2-Port モジュールが出荷される際は、"工場出荷時設定" の状態である必要があります。Anybus CompactCom EtherNet/IP w.IT Functionality 2-Port デバイスの場合、これは IP スイッチが割り当てられていないことを意味します (IP 0.0.0.0)。ネットワークから '工場出荷時設定リセット' コマンドが受信されると、Anybus モジュールは全ての IP 情報を消去し、ホストアプリケーションに Anybus モジュールのリセットが必要であることを伝えます。これは、Reset コマンドをホストのアプリケーションオブジェクト (FFh) に送信することで実行されます (電源オン + 工場出荷時設定)。詳細については、"Anybus CompactCom Software Design Guide" を参照してください。

### 2.3.3 IP アドレス

通常、Anybus CompactCom EtherNet/IP w.IT Functionality 2-Port デバイスの IP 番号は DHCP サーバを通して割り当てられます。HMS は、エンドユーザが特定の値への IP アドレスの設定を要求していない限り（例えばキーパッドの使用による場合など）、EtherNet/IP モジュールの初期化フェーズ中にネットワークコンフィグレーションオブジェクト（04h、インスタンス #3 ~ #11）を使用しないことをお奨めします。その理由は、EtherNet/IP ネットワークから（DHCP を通して）工場出荷時設定コマンドが受信される際に、DHCP が有効化されてリセットされた後にノードを使用できる必要があるためです。



## 3. 基本操作

### 3.1 概要

#### 3.1.1 ソフトウェアの要件

一般的には、Anybus-CompactCom EtherNet/IP をサポートするために追加のネットワークサポートコードを書く必要はありません。ただし、EtherNet/IP ネットワーキングシステムの性質により、特定の制限を考慮する必要があります。

- モジュール内の特定の機能では、コマンド 'Get\_Instance\_Number\_By\_Order' (アプリケーションデータオブジェクト、FEh) をホストアプリケーションに実装する必要があります。
- 通常の状態では、ホストアプリケーションによって最大5つの診断インスタンス (85 ページの "診断オブジェクト (02h)" を参照) を作成できます。メジャーフォールトが発生すると、追加の6番目のインスタンスが作成される場合があります。
- EtherNet/IP 自体は、アサイクリックリクエスト (つまりアプリケーションデータオブジェクト内のインスタンスに対するリクエスト) に関して特定のタイミング要求を課しません。このようなリクエストに対しては適切な期間内での処理とレスポンスが一般的に推奨されます (実際にこのことが厳密に何を意味するのかは、実装と実際のインストールによって異なる)。
- 高度な CIP 特有の機能には、CIP ネットワーキング内部に関する詳細な知識または公式な CIP 仕様と EtherNet/IP 仕様からの情報 (あるいはその両方) が必要になることがあります。このような場合、本製品の実装を担当しているユーザは、十分な知識を獲得するためにこれらの仕様を取得するか、これが必要とされないような方法に実装を制限する必要があります。

Anybus-CompactCom ソフトウェアインターフェースに関する詳細な情報については、一般の "Anybus-CompactCom Software Design Guide" を参照してください。

下記も参照してください。

- 85 ページの "診断オブジェクト (02h)" (Anybus モジュールオブジェクト)
- "Anybus-CompactCom Software Design Guide"、"Application Data Object (FEh)"

## 3.2 デバイスカスタマイズ

### 3.2.1 ネットワーク Identity

デフォルトでは、モジュールは以下の Identity 設定を使用します。

- Vendor ID: 005Ah (HMS Industrial Networks)
- Device Type: 0000h (汎用デバイス)
- Product Code: 002Eh (Anybus-CompactCom EtherNet/IP)
- Product Name: 'Anybus-CC EtherNet/IP (2-Port)'

オプションとして、対応しているインスタンスアトリビュートを EtherNet/IP ホストオブジェクトに実装して、モジュールの識別をカスタマイズすることができます。

下記も参照してください。

- 54 ページの "Identity オブジェクト (01h) " (CIP オブジェクト)
- 137 ページの "EtherNet/IP ホストオブジェクト (F8h) " (ホストアプリケーションオブジェクト)

---

**重要:** CIP 仕様によると、ベンダ ID とシリアルナンバーの組み合わせは一意である必要があります。HMS のベンダ ID (005Ah) との組み合わせにおいてカスタムシリアルナンバーを使用することは許可されず、任意にベンダ ID を選択することも許可されません。この要件に従わない場合、相互運用性の問題またはその他の好ましくない副次的な不具合 (あるいはその両方) が発生する原因となります。HMS は、実装が標準の EDS ファイルからの逸脱を必要としないという条件の下で、デフォルトのシリアルナンバーとの組み合わせにおける HMS のベンダ ID (005Ah) の使用を承認します。

ベンダ ID を取得するには、ODVA までご連絡ください。

### 3.2.2 電子的データシート (EDS)

EtherNet/IP では、デバイスの特性は拡張子が EDS になる ASCII データファイル内に保存されます。このファイルは、ネットワークコンフィグレーションをセットアップする際に、コンフィグレーションツールなどによって使用されます。HMS は、モジュール内のデフォルト設定に対応する標準 (汎用) の EDS ファイルを提供します。ただし、Anybus-CompactCom コンセプトの柔軟な性質により、汎用の EDS ファイルを無効にする方法に製品の挙動を変更することができます。このような場合、同様にデフォルトの Identity 情報を無効にして製品の再証明を必要とするカスタム EDS ファイルを作成する必要があります。

**注意:** モジュールはパラメータオブジェクトを実装するため、RSNetWorx などのコンフィグレーションツールは適切な EDS ファイルを自動的に生成することができます。この機能では、コマンド 'Get\_Instance\_Number\_By\_Order' (アプリケーションデータオブジェクト、FEh) がホストアプリケーションに実装されている必要があることに注意してください。

下記も参照してください。

- 64 ページの "パラメータオブジェクト (0Fh) " (CIP オブジェクト)
- "Anybus-CompactCom Software Design Guide", "Application Data Object (FEh)"

---

**重要:** HMS は、標準の EDS ファイルが実際の実装に合致し、Identity 情報が変更されないままであるという条件の下においてのみ、標準の EDS ファイルの使用を承認します。

### 3.2.3 EtherNet/IP および CIP の実装

デフォルトでは、モジュールは汎用の CIP プロファイルをサポートします。オプションとして、実装されていない CIP オブジェクトへの要求をホストアプリケーションへ再ルーティングすることで、他のプロファイルのサポートを有効にすることができます。

特定のプロファイルをサポートするには、以下の手順を実行してください。

- プロファイル要件に従って、EtherNet/IP ホストオブジェクト内で Identity 設定をセットアップします。
- プロファイル要件に従って、アセンブリインスタンス番号をセットアップします。
- EtherNet/IP ホストオブジェクト内でホストアプリケーションへの CIP メッセージのルーティングを有効にします。
- ホストアプリケーション内で必要な CIP オブジェクトを実装します。

非接続型 CIP ルーティングがサポートされますが、このことは最初に接続をセットアップしなくてもメッセージをデバイスに送信できることを意味します。

下記も参照してください。

- 137 ページの "EtherNet/IP ホストオブジェクト (F8h) " (ホストアプリケーションオブジェクト)
- 142 ページの " コマンドの詳細 : Process\_CIP\_Object\_Request"

### 3.2.4 Web インターフェース

Web インターフェースは、特定のアプリケーションに適合するように完全にカスタマイズすることができます。データおよび Web のページは、任意の標準 FTP クライアントを使用してアクセスできる FLASH ベースのファイルシステム内に保存されます。

下記も参照してください。

- 22 ページの " ファイルシステム "
- 23 ページの "FTP サーバ "
- 25 ページの "Web サーバ "

### 3.2.5 ソケットインターフェース (上級ユーザのみ)

組み込みのソケットインターフェースを使用すると、TCP/IP 上に追加のプロトコルを実装できるようになります。

下記も参照してください。

- 95 ページの " ソケットインターフェースオブジェクト (07h) " (Anybus モジュールオブジェクト)
- 153 ページの " メッセージのセグメント化 "

## 3.3 通信設定

他の Anybus-CompactCom 製品の場合と同様に、ネットワーク関連の通信設定はネットワークコンフィグレーションオブジェクト (04h) 内でグループ化されます。

この場合、以下のものが含まれます。

- **TCP/IP 設定**

モジュールをネットワーク上で参加できるようにするには、これらの設定を適切に指定する必要があります。

モジュールは DHCP をサポートしますが、これは TCP/IP 設定を DHCP サーバから自動的に取得するために使用される場合があります。DHCP はデフォルトで有効になっていますが、必要に応じて無効にすることができます。

- **物理リンク設定**

デフォルトでは、モジュールはオートネゴシエーションを使用して物理リンク設定を構築しますが、必要に応じて特定の設定を強制することができます。

ネットワークコンフィグレーションオブジェクト (04h) 内のパラメータは、組み込まれた Web サーバを通して、さらには TCP/IP インターフェースオブジェクト (CIP) を通してネットワークから使用できます。

下記も参照してください。

- 25 ページの "Web サーバ "
- 74 ページの "TCP/IP インターフェースオブジェクト (F5h) " (CIP)
- 78 ページの "Ethernet リンクオブジェクト (F6h) " (CIP)
- 87 ページの " ネットワークコンフィグレーションオブジェクト (04h) " (Anybus モジュールオブジェクト)
- 156 ページの "HICP (ホスト IP コンフィグレーションプロトコル) "

## 3.4 診断

保留されている全てのイベントの重大度値は(論理 OR を使用して)組み合わせられ、Identity オブジェクト (CIP) の 'Status' アトリビュート内の対応するビットにコピーされます。

下記も参照してください。

- 54 ページの "Identity オブジェクト (01h) " (CIP)
- 85 ページの " 診断オブジェクト (02h) " (Anybus モジュールオブジェクト)

## 3.5 ネットワークのデータ交換

### 3.5.1 アプリケーションデータ (ADI)

ADI は、ADI オブジェクト (CIP) を通して表されます。このオブジェクト内部の各インスタンスは、ホストアプリケーション側のアプリケーションデータオブジェクト内のインスタンスに直接対応します。

ADI のアクセス可能な範囲は 1 ～ 65535 です。

下記も参照してください。

- 64 ページの "パラメータオブジェクト (0Fh) " (CIP)
- 70 ページの "ADI オブジェクト (A2h) " (CIP)

### 3.5.2 プロセスデータ

プロセスデータは、アセンブリオブジェクト (CIP) を通して表されます。各 ADI エレメントは、バイト境界上にマッピングされることに注意してください。各 BOOL が 1 バイトを占有します。

下記も参照してください。

- 58 ページの "アセンブリオブジェクト (04h) " (CIP)
- 61 ページの "コネクションマネージャ (06h) " (CIP)

### 3.5.3 データタイプの変換

Anybus データタイプは、以下のように CIP 標準に変換されます (その逆も同様)。

Anybus データタイプ	CIP データタイプ	コメント
BOOL	BOOL	このタイプの各 ADI エレメントは、1 バイトを占有します。
ENUM	USINT	
SINT8	SINT	
UINT8	USINT	
SINT16	INT	このタイプの各 ADI エレメントは、2 バイトを占有します。
UINT16	UINT	
SINT32	DINT	このタイプの各 ADI エレメントは、4 バイトを占有します。
UINT32	UDINT	
FLOAT	REAL	
CHAR	SHORT_STRING	SHORT_STRING は、実際の文字データ (この場合は実際の ADI エレメント) が次に続く、1 バイトの長さのフィールド (この場合は ADI エレメントの数を表す) で構成されます。これは、10 文字の文字列が 11 バイトを占有することを意味します。
SINT64	LINT	このタイプの各 ADI エレメントは、8 バイトを占有します。
UINT64	ULINT	

## 3.6 ファイルシステム

### 3.6.1 概要

カテゴリ：拡張

組み込みのファイルシステムは、不揮発性ストレージの 1.43MB をホストしますが、これには HTTP サーバ、FTP サーバ、電子メールクライアント、およびホストアプリケーションによってアクセスできます。

ファイルシステムは、以下の規則を使用します。

- '\' (バックスラッシュ) は、パス区切り文字として使用されます。
- 'パス' はシステムルートを開始点とするため、 '\' で開始する必要があります。
- 'パス' は '\' で終了しないようにしてください。
- 名前にはスペース (' ') を含めることができますが、スペースで開始または終了しないようにしてください。
- 名前には以下の記号は使用しないでください。 '\ / : \* ? " < > |'
- 名前は 48 文字以下でなければなりません。
- パスは 255 文字 (ファイル名を含む) 以下でなければなりません。

下記も参照してください。

- 23 ページの "FTP サーバ"
- 25 ページの "Web サーバ"
- 32 ページの "電子メールクライアント"
- 33 ページの "SSI (Server Side Include)"
- 120 ページの "ファイルシステム・インターフェースオブジェクト (0Ah)"

---

**重要：** ファイルシステムはフラッシュメモリ内に配置されます。技術的な理由により、各フラッシュセグメントは不具合が発生する前に約 100000 回消去されることができ、ランダムアクセスストレージには適していません。

以下の操作を行うと、1 つ以上のフラッシュセグメントが消去されます。

- ファイルまたはディレクトリの削除、移動、または名前変更
- 既存ファイルへのデータのライトまたは追加
- ファイルシステムのフォーマット

### 3.6.2 システムファイル

ファイルシステムには、システムコンフィグレーションで使用される一連のファイルが含まれています。これらのファイル ("システムファイル" と呼ばれる) は、標準テキストエディタ (Microsoft Windows™ のメモ帳など) を使用して変更可能な通常の ASCII ファイルです。一部の例外もありますが、これらのファイルの形式は 'キー' のコンセプトに基づき、各 'キー' には値を割り当てることができます。以下を参照してください。

例：

```
[Key1]
value of Key1

[Key2]
value of Key2
```

## 4. FTP サーバ

### 4.1 概要

カテゴリ：拡張

組み込みの FTP サーバは、標準の FTP クライアントを使用することでファイルシステムの管理を容易にします。

デフォルトでは、以下のポート番号が FTP 通信に使用されます。

- TCP、ポート 20 (FTP データポート)
- TCP、ポート 21 (FTP 通信ポート)

FTP サーバは、最大 8 つの同時接続をサポートします。

### 4.2 ユーザアカウント

ユーザアカウントは、コンフィグレーションファイル '\ftp.cfg' に保存されます。このファイルは、全てのユーザのユーザ名、パスワード、ホームディレクトリを保持します。ユーザは、ホームディレクトリ外のファイルにはアクセスできません。

ファイルフォーマット：

```
User1:Password1:Homedir1
User2:Password2:Homedir2
User3:Password3:Homedir3
```

オプションとして、以下のように UserN:PasswordN セクションをユーザのリストが含まれているファイルへのパスで置き換えることができます。

ファイルフォーマット ('\ftp.cfg')：

```
User1:Password1:Homedir1
User2:Password2:Homedir2
\path\userlistA:HomedirA
\path\userlistB:HomedirB
```

ユーザリストが含まれたファイルは以下のフォーマットになります。

ファイルフォーマット：

```
User1:Password1
User2:Password2
User3:Password3
```

#### 注意

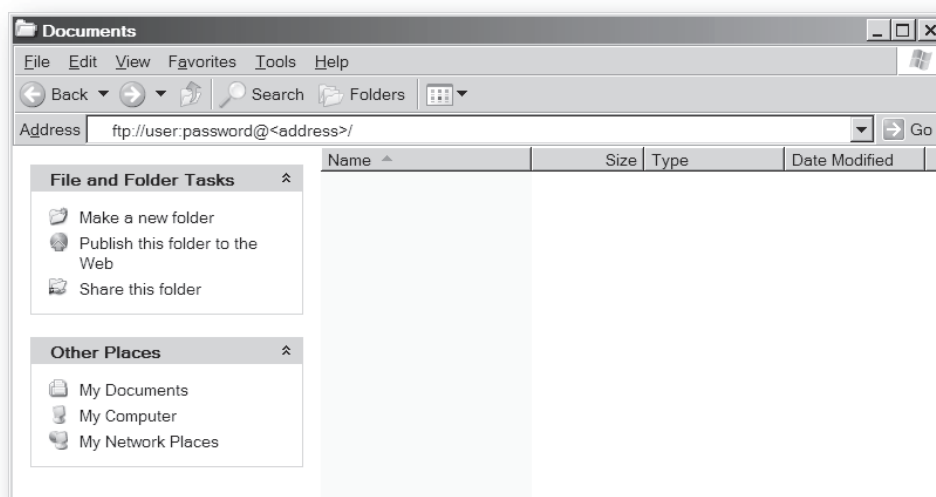
- ユーザ名は 15 文字以下の長さにする必要があります。
- パスワードは 15 文字以下の長さにする必要があります。
- ユーザ名とパスワードには、英文字または数字（あるいはその両方）のみを含める必要があります。
- '\ftp.cfg' が欠落していたり解釈できなかつたりする場合、全てのユーザ名 / パスワードの組み合わせが許容され、ホームディレクトリは FTP ルート（つまり '\ftp\'）になります。

- ユーザのホームディレクトリをログインできるようにする場合には、それもファイルシステム内に存在する必要があります。ユーザ情報を 'ftp.cfg' に追加するだけでは十分ではありません。
- Ethernet オブジェクト内で 'Admin Mode' が有効になっている場合、全てのユーザ名 / パスワードの組み合わせが許容され、ユーザにはファイルシステムへの無制限のアクセスが与えられます (つまりホームディレクトリはシステムルートになる)。
- 少なくとも一人のユーザにルートアクセス ('\') の許可を与えることが強く推奨されます。そうでない場合、システムファイルの変更が必要になるたびに 'Admin Mode' を有効にする必要があります ('\ftp.cfg' を含む)。

## 4.3 セッションの例

Windows エクスプローラは組み込みの FTP クライアントを備えており、これは以下のよう  
にファイルシステムにアクセスするために簡単に使用できます。

1. 'スタート' ボタンを右クリックして 'エクスプローラ' を選択し、Windows エクスプローラを開きます。
2. アドレスフィールドで、FTP://<user>:<password>@<address> と入力します。
  - <address> を Anybus モジュールの IP アドレスで置き換えます。
  - <user> をユーザ名で置き換えます。
  - <password> をパスワードで置き換えます。
3. Enter を押します。エクスプローラは、指定された設定で Anybus モジュールへの接続を試みます。成功すると、ファイルシステムがエクスプローラのウィンドウ内に表示されます。





## 5. Web サーバ

### 5.1 概要

カテゴリ：拡張

組み込みの Web サーバは、エンドユーザのやり取りとコンフィグレーションのために、柔軟な環境を提供します。SSI とクライアント側のスクリプト作成の強力な組み合わせにより、オブジェクトとファイルシステムデータへのアクセスが可能になり、高度なグラフィカルユーザインターフェースの作成が実現されます。

Web インターフェースはファイルシステム内に保存され、FTP サーバを通してアクセスすることができます。必要に応じて、イーサネットホストオブジェクト内で Web サーバを完全に無効にすることができます。

Web サーバは、最大 20 の同時接続をサポートし、ポート 80 を通して通信します。

下記も参照してください。

- 23 ページの "FTP サーバ "
- 33 ページの "SSI (Server Side Include) "
- 147 ページの "Ethernet ホストオブジェクト (F9h) "

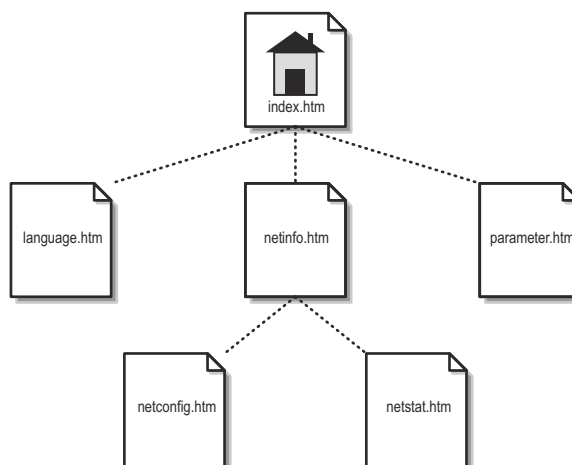
### 5.2 デフォルトの Web ページ

デフォルトの Web インターフェースは、一連の仮想ファイルから構成されます。これらの仮想ファイルは、同じ場所（つまり Web ルート）内に同じ名前が付けられたファイルを配置することで置き換えることが可能ですが、永久に消去することはできません。

ファイルは現状のまま使用できますが、カスタマイズされた Web 環境から呼び出すこともできます。

ファイルは以下の通りです。

```
<WebRoot>\style.css
<WebRoot>\arrow_red.gif
<WebRoot>\index.htm
<WebRoot>\netinfo.htm
<WebRoot>\netconfig.htm
<WebRoot>\netstat.htm
<WebRoot>\parameter.htm
<WebRoot>\language.htm
```



**注意：**これらのファイルのいずれも使用しない場合、ファイルシステム・インターフェースオブジェクト内の仮想ファイルシステムを全て完全に無効にすることが推奨されます。

下記も参照してください。

- 22 ページの " ファイルシステム "
- 120 ページの " ファイルシステム・インターフェースオブジェクト (0Ah) "

## 5.2.1 ネットワークのコンフィグレーション

ネットワークコンフィグレーションページは、ネットワークコンフィグレーションオブジェクト内で TCP/IP と SMTP の設定を変更するためのインターフェースを提供します。

### Anybus-CC EtherNet/IP (2-Port)

#### Network configuration

IP Configuration	
IP address:	<input type="text" value="10.11.20.152"/>
Subnet mask:	<input type="text" value="255.255.0.0"/>
Gateway:	<input type="text" value="10.11.0.1"/>
Host name:	<input type="text"/>
Domain name:	<input type="text" value="hms.se"/>
DNS1:	<input type="text" value="0.0.0.0"/>
DNS2:	<input type="text" value="0.0.0.0"/>
DHCP:	<input checked="" type="checkbox"/>
<input type="button" value="Store settings"/>	

SMTP Settings	
SMTP Server:	<input type="text"/>
SMTP User:	<input type="text"/>
SMTP Pswd:	<input type="text"/>
<input type="button" value="Store settings"/>	

Ethernet Configuration	
Comm 1:	<input type="text" value="Auto"/>
Comm 2:	<input type="text" value="Auto"/>
<input type="button" value="Store settings"/>	

▶ Main ▶ Network interface

TCP/IP と SMTP の設定を有効にするには、モジュールをリセットする必要があります。イーサネットコンフィグレーションの設定はすぐに有効になります。

使用できる編集可能な設定については、次のページで説明します。

## IP Configuration

任意の変更を有効にするには、モジュールをリセットする必要があります。

名前	説明
IP address	モジュールの TCP/IP 設定
Subnet mask	デフォルト値 : 0.0.0.0
Gateway	値の範囲 : 0.0.0.0 ~ 255.255.255.255
Host name	IP アドレスまたは名前 最大 64 文字
Domain name	IP アドレスまたは名前 最大 48 文字
DNS 1	プライマリおよびセカンダリ DNS サーバ、ホスト名を解決するために使用
DNS 2	デフォルト値 : 0.0.0.0 値の範囲 : 0.0.0.0 ~ 255.255.255.255
DHCP	DHCP の有効化または無効化のためのチェックボックス デフォルト値 : 有効

## SMTP Settings

任意の変更を有効にするには、モジュールをリセットする必要があります。

名前	説明
SMTP Server	IP アドレスまたは名前 最大 64 文字
SMTP User	最大 64 文字
SMTP Pswd	最大 64 文字

## Ethernet Configuration

変更はすぐに有効になります。

名前	説明
Comm 1	イーサネット speed/duplex の設定
Comm 2	デフォルト値 : auto

## 5.2.2 イーサネット統計ページ

イーサネット統計 Web ページには以下の情報が含まれています。

イーサネットリンク		説明
Port 1	Speed:	現在のリンク速度
	Duplex:	現在の duplex のコンフィグレーション
Port 2	Speed:	現在のリンク速度
	Duplex:	現在の duplex のコンフィグレーション

Ethernet/IP 統計	説明
Established Class1 Connections	確立された Class 1 コネクションの現在の数
Established Class3 Connections	確立された Class 3 コネクションの現在の数
Connection Open Requests	受信されたコネクション開始要求の数
Connection Open Format Rejects	要求フォーマットエラーにより却下されたコネクション開始要求
Connection Open Resource Rejects	リソースの欠如により却下されたコネクション開始要求
Connection Open Other Rejects	他の理由により却下されたコネクション開始要求
Connection Close Requests	受信されたコネクション開始要求の数
Connection Close Format Rejects	要求フォーマットエラーにより却下されたコネクション終了要求
Connection Close Other Rejects	他の理由により却下されたコネクション終了要求
Connection Timeouts	コネクションタイムアウトの数

インターフェースカウンタ	説明
In Octets:	受信されたバイト数
In Ucast Packets:	受信されたユニキャストパケット数
In NUCast packets:	受信された非ユニキャストパケット数（ブロードキャストおよびマルチキャスト）
In Discards:	使用できるメモリバッファが存在しないために破棄された、受信されたパケット数
In Errors:	受信エラーにより破棄された、受信されたパケット数
In Unknown Protos:	受信されたサポート外のプロトコルタイプのパケット数
Out Octets:	送信されたバイト数
Out Ucast packets:	送信されたユニキャストパケット数
Out NUCast packets:	送信された非ユニキャストパケット数（ブロードキャストおよびマルチキャスト）
Out Discards:	使用できるメモリバッファが存在しないために破棄された、送信されたパケット数
Out Errors:	送信エラー

## 5.3 サーバコンフィグレーション

### 5.3.1 概要

カテゴリ：高度

基本的な Web サーバコンフィグレーション設定は、システムファイル '\http.cfg' に保存されます。このファイルには、Web インターフェース、コンテンツタイプ、および SSI にスキャンされることになるファイルタイプのリストのルートディレクトリが保持されます。

<pre> ファイルフォーマット： [WebRoot] \web  [FileTypes] FileType1:ContentType1 FileType2:ContentType2 ... FileTypeN:ContentTypeN  [SSIFileTypes] FileType1 FileType2 ... FileTypeN </pre>	<ul style="list-style-type: none"> <li> <b>Web ルートディレクトリ</b>  Web サーバは、このディレクトリ外のファイルにはアクセスできません。 </li> <li> <b>コンテンツタイプ</b>  ファイル拡張子およびレポートされたコンテンツタイプのリストです。  下記も参照してください。  - 30 ページの " デフォルトのコンテンツタイプ " </li> <li> <b>SSI ファイルタイプ</b>  デフォルトでは、拡張子 'shtm' を持つファイルのみが SSI にスキャンされます。必要に応じて、追加の SSI ファイルタイプをここに追加できます。 </li> </ul>
---	---

Web ルートディレクトリは、Web インターフェースに関連する全てのファイルの位置を決定します。このディレクトリおよびそのサブディレクトリの外部のファイルは、Web サーバによってアクセスできません。

### 5.3.2 インデックスページ

モジュールは、潜在的なインデックスページを以下の順番で検索します。

1. <WebRoot>\index.htm
2. <WebRoot>\index.html
3. <WebRoot>\index.shtm
4. <WebRoot>\index.wml

**注意 1：** <WebRoot> を '\http.cfg' で指定された Web ルートディレクトリと置き換えます。

**注意 2：** インデックスページが見つからない場合、モジュールはデフォルトで仮想インデックスファイルになります（有効な場合）。

下記も参照してください。

- 25 ページの " デフォルトの Web ページ "

### 5.3.3 デフォルトのコンテンツタイプ

デフォルトでは、以下のコンテンツタイプがファイル拡張子によって認識されます。

ファイル拡張子	レポートされるコンテンツタイプ
htm、html、shtm	text/html
gif	image/gif
jpeg、jpg、jpe	image/jpeg
png	image/x-png
js	application/x-javascript
bat、txt、c、h、cpp、hpp	text/plain
zip	application/x-zip-compressed
exe、com	application/octet-stream
wml	text/vnd.wap.wml
wmlc	application/vnd.wap.wmlc
wbmp	image/vnd.wap.wbmp
wmls	text/vnd.wap.wmlscript
wmlsc	application/vnd.wap.wmlscriptc
xml	text/xml
pdf	application/pdf
css	text/css

コンテンツタイプは、サーバコンフィグレーションファイルに追加することで、追加または再定義できます。29 ページの " 概要 " を参照してください。

### 5.3.4 認証

ディレクトリは、保護するディレクトリに 'web\_accs.cfg' というファイルを配置すると、Web アクセスから保護することができます。このファイルには、ディレクトリとそのサブディレクトリへのアクセスを許可されるユーザのリストが含まれるようになります。

ファイルフォーマット：

```
Username1:Password1
Username2:Password2
...
```

```
UsernameN:PasswordN
```

```
[AuthName]
(message goes here)
```

● 認証されたユーザのリストです。

● オプションとして、キー [AuthName] を含めることでログインメッセージを指定することができます。このメッセージは、保護されたディレクトリにアクセスすると Web ブラウザによって表示されます。

認証されたユーザのリストは、オプションで他の1つまたは複数のファイルにリダイレクトすることができます。

例:

この例では、認証されたユーザのリストが 'here.cfg' と 'too.cfg' からロードされます。

```
[File path]
|i\put\some\over\here.cfg
|i\actually\put\some\of\it\here\too.cfg

[AuthType]
Basic

[AuthName]
Howdy. Password, please.
```

フィールド 'AuthType' は、認証スキームを識別するために使用されます。

値	説明
Basic	プレーンテキストを使用した Web 認証方法です。
Digest	暗号化されたパスワードを使用した、より安全な Web 認証方法です。[AuthType] フィールドが指定されない場合、デフォルトとして使用されます。

## 6. 電子メールクライアント

### 6.1 概要

カテゴリ：拡張

組み込みの電子メールクライアントを使用すると、アプリケーションはSMTPサーバを通して電子メールメッセージを送信することができます。メッセージは、SMTPクライアントオブジェクト内で直接的に指定されるか、ファイルシステムから取得することができます。後者の場合にはSSIが含まれることがありますが、技術的な理由により、特定のコマンドを使用できないことに注意してください（各SSIコマンドに個別に指定される）。

クライアントは、'LOGIN'方法を使用して認証をサポートします。アカウント設定などは、ネットワークコンフィグレーションオブジェクト内に保存されます。

下記も参照してください。

- 87 ページの " ネットワークコンフィグレーションオブジェクト (04h) "
- 115 ページの "SMTP クライアントオブジェクト (09h) "

### 6.2 電子メールメッセージの送信方法

電子メールメッセージの送信を可能にするには、SMTP 設定を指定する必要があります。

これには以下のものが含まれます。

- 有効な SMTP サーバアドレス
- 有効なユーザ名
- 有効なパスワード

電子メールメッセージを送信するには、次の手順を行います。

1. 'Create' コマンド (03h) を使用して、新しい電子メールインスタンスを作成します。
2. 電子メールインスタンス内で送信者、受信者、トピック、およびメッセージ本文を指定します。
3. 'Send Instance Email' コマンド (10h) を電子メールインスタンスに発行します。
4. オプションとして、'Delete' コマンド (04h) を使用して電子メールインスタンスを削除します。

ファイルシステム内のファイルに基づいたメッセージの送信は、'Send Email from File' コマンドを使用して達成されます。ファイルフォーマットの説明については、118 ページの " コマンドの詳細：Send Email From File" を参照してください。



## 7. SSI (Server Side Include)

### 7.1 概要

カテゴリ：高度

SSI (Server Side Include) 機能を使用すると、Web ページ上および電子メールメッセージ内にファイルとオブジェクトからのデータを表すことができるようになります。

SSI は、ソースドキュメント内に埋め込まれた特別なコマンドです。Anybus モジュールにこのようなコマンドが発生すると、Anybus モジュールはこれを実行し、結果として指定された操作と置き換えます (該当する場合)。

デフォルトでは、拡張子 '.shtml' を持つファイルのみが SSI にスキャンされます。

### 7.2 Include File

この関数には、ファイルのコンテンツが含まれます。コンテンツは SSI にスキャンされません。

**注意：**この関数は、電子メールメッセージでは使用できません。

構文：

```
<?--#include file="filename"-->
```

filename- ソースファイル

デフォルト出力：

シナリオ	デフォルト出力
Success	(ファイルのコンテンツ)

## 7.3 コマンド関数

### 7.3.1 概要

コマンド関数はコマンドを実行し、その結果を含めます。

一般的な構文：

```
<?--#exec cmd_argument='command'-->
```

command- コマンド関数。以下を参照してください。

コマンド関数：

コマンド	電子メールメッセージに対する有効性	ページ
GetConfigItem()	はい	35
SetConfigItem()	いいえ	36
SsiOutput()	はい	38
DisplayRemoteUser	いいえ	38
ChangeLanguage()	いいえ	39
IncludeFile()	はい	40
SaveDataToFile()	いいえ	41
printf()	はい	42
scanf()	いいえ	44

### 7.3.2 GetConfigItem()

このコマンドは、ファイルシステム内のファイルから特定の情報を返します。

ファイルフォーマット：

ソースファイルのフォーマット以下ようになります。

```
[key1]
value1

[key2]
value2
...
[keyN]
valueN
```

構文：

```
<?--exec cmd_argument='GetConfigItem("filename", "key",
                                "separator")'-->
```

filename- リード元のソースファイル。  
 key - ファイル内のソース [キー]。  
 separator- オプション。行区切り文字を指定します (例 "<br>")。  
 (デフォルトは CRLF)

デフォルト出力：

シナリオ	デフォルト出力
Success	(指定されたキーの値)
Authentication Error	"Authentication error "
File open error	"Failed to open file "filename" "
Key not found	"Tag (key) not found "

例：

SSI

```
<?--exec cmd_argument='GetConfigItem("\fruit.cnf", "Lemon")'-->
```

上記 SSI と組み合わせるファイル (\fruit.cnf)

```
[Apple]
Green

[Lemon]
Yellow

[Banana]
Blue
```

この場合、文字列 'Yellow' を返します。

### 7.3.3 SetConfigItem()

この関数は、ファイルシステム内にファイルとして HTML フォームを保存します。

**注意：**この関数は、電子メールメッセージでは使用できません。

構文：

```
<?--#exec cmd_argument='SetConfigItem("filename" [, Overwrite])'-->
```

filename- 宛先ファイル。指定されたファイルが存在しない場合は作成されます (パスが有効な場合)。

Overwrite- オプション。コマンドが発行されるたびにモジュールが新しいファイルを作成するように強制します。  
デフォルトの挙動は、既存のファイルの変更です。

ファイルフォーマット：

各フォームオブジェクトは [tag] として保存され、実際の値がそれに続きます。

```
[form object name 1]
form object value 1
```

```
[form object name 2]
form object value 2
```

```
[form object name 3]
form object value 3
```

...

```
[form object name N]
form object value N
```

**注意：**アンダースコア (‘\_’) で開始される名前を持つフォームオブジェクトは保存されません。

デフォルト出力：

シナリオ	デフォルト出力
Success	"Configuration stored to "filename" "
Authentication Error	"Authentication error "
File open error	"Failed to open file "filename" "
File write error	"Could not store configuration to "filename" "

例:

以下の例は、この関数をどのように使用するかを示しています。結果のページはフォームをそれ自体に送信し、続いてこれが SetConfigItem コマンドによって評価されます。

```
<HTML>
<HEAD><TITLE>SetConfigItem Test</TITLE></HEAD>
<BODY>

<?--#exec cmd_argument='SetConfigItem("\food.txt")'-->

<FORM action="test.shtm">
  <P>
    <LABEL for="Name">Name: </LABEL><BR>
    <INPUT type="text" name="Name"><BR><BR>

    <LABEL for="_Age">Age: </LABEL><BR>
    <INPUT type="text" name="_Age"><BR><BR>

    <LABEL for="Food">Food: </LABEL><BR>
    <INPUT type="radio" name="Food" value="Cheese"> Cheese<BR>
    <INPUT type="radio" name="Food" value="Sausage"> Sausage<BR><BR>

    <LABEL for="Drink">Drink: </LABEL><BR>
    <INPUT type="radio" name="Drink" value="Wine"> Wine<BR>
    <INPUT type="radio" name="Drink" value="Beer"> Beer<BR><BR>

    <INPUT type="submit" name="_submit">
    <INPUT type="reset" name="_reset">
  </P>
</FORM>

</BODY>
</HTML>
```

結果のファイル ('\food.txt') は、以下のように表示される場合があります。

```
[Name]
Cliff Barnes

[Food]
Cheese

[Drink]
Beer
```

**注意:** この例が機能するためには、HTML ファイルに 'test.shtm' と名前を付ける必要があります。

### 7.3.4 SsiOutput()

このコマンドは、以下のコマンド関数の SSI 出力を一時的に変更します。

構文：

```
<?--#exec cmd_argument='SsiOutput("success", "failure")'-->
```

success- 成功の場合に使用する文字列

failure - 失敗の場合に使用する文字列

デフォルト出力：

(このコマンドは独自には出力を生成しない)

例：

以下の例は、このコマンドをどのように使用するかを示しています。

```
<?--#exec cmd_argument='SsiOutput ("Parameter stored", "Error")'-->
<?--#exec cmd_argument='SetConfigItem("File.cfg", Overwrite)'-->
```

下記も参照してください。

- 52 ページの "SSI 出力コンフィグレーション "

### 7.3.5 DisplayRemoteUser

このコマンドは、認証セッションにおいてユーザ名を返します。

**注意：**このコマンドは、電子メールメッセージでは使用できません。

構文：

```
<?--#exec cmd_argument='DisplayRemoteUser'-->
```

デフォルト出力：

シナリオ	デフォルト出力
Success	(現在のユーザ)

### 7.3.6 ChangeLanguage()

このコマンドは、HTML フォームオブジェクトに基づいて言語設定を変更します。

**注意：**このコマンドは、電子メールメッセージでは使用できません。

構文：

```
<?--#exec cmd_argument='ChangeLanguage( "source" )'-->
```

source - 新しい言語設定を含むフォームオブジェクトの名前。  
渡された値は、以下のように 1 桁である必要があります。

フォーム値	言語
"0"	英語
"1"	ドイツ語
"2"	スペイン語
"3"	イタリア語
"4"	フランス語

デフォルト出力：

シナリオ	デフォルトデフォルト出力
Success	"Language changed"
Error	"Failed to change language"

例：

以下の例は、この関数をどのように使用するかを示しています。結果のページはフォームをそれ自体に送信し、続いてこれが ChangeLanguage() コマンドによって評価されます。

```
<HTML>
<HEAD><TITLE>ChangeLanguage Test</TITLE></HEAD>
<BODY>

<?--#exec cmd_argument='ChangeLanguage("lang")'-->

<FORM action="test.shtm">
  <P>
    <LABEL for="lang">Language (0-4): </LABEL><BR>
    <INPUT type="text" name="lang"><BR><BR>

    <INPUT type="submit" name="_submit">
  </P>
</FORM>

</BODY>
</HTML>
```

**注意：**この例が機能するためには、HTML ファイルに 'test.shtm' と名前を付ける必要があります。

### 7.3.7 IncludeFile()

このコマンドには、ファイルのコンテンツが含まれます。コンテンツは SSI にスキャンされないことに注意してください。

構文:

```
<?--#exec cmd_argument='IncludeFile("filename" [, separator])'-->
```

filename- ソースファイル

separator- オプション。行区切り文字を指定します (例 "<br>")。

デフォルト出力:

シナリオ	デフォルト出力
Success	(ファイルコンテンツ)
Authentication Error	"Authentication error "
File open error	"Failed to open file "filename" "

例:

以下の例は、この関数をどのように使用するかを示しています。

```
<HTML>
<HEAD><TITLE>IncludeFile Test</TITLE></HEAD>
<BODY>
  <H1> Contents of 'info.txt':</H1>
  <P>
    <?--#exec cmd_argument='IncludeFile("info.txt")'-->.
  </P>
</BODY>
</HTML>
```

'info.txt' のコンテンツ:

```
Neque porro quisquam est qui dolorem ipsum quia dolor sit amet,
consectetur, adipisci velit...
```

ブラウザで表示すると、結果のページは以下のように表示されます。



下記も参照してください。

- 33 ページの "Include File"



### 7.3.8 SaveDataToFile()

このコマンドは、ファイルシステム内にファイルとして HTML フォームからデータを保存します。異なるフォームオブジェクトからのコンテンツは、空白行 (2\*CRLF) によって区切られます。

**注意：**このコマンドは、電子メールメッセージでは使用できません。

構文：

```
<?--#exec cmd_argument='SaveDataToFile("filename" [, "source"],
                                Overwrite|Append) '-->
```

- filename- 宛先ファイル。指定されたファイルが存在しない場合は作成されます (パスが有効な場合)。
- source - オプション。フォームオブジェクトを指定することで、その特定のフォームオブジェクトからのデータのみが保存されます。  
デフォルトの挙動は、アンダースコア ('\_') で始まる名前を持つフォームオブジェクトを除き、全てのフォームオブジェクトからのデータの保存です。
- Overwrite|Append- 既存のファイルへのデータの上書きまたは追加を指定します。

デフォルト出力：

シナリオ	デフォルト出力
Success	"Configuration stored to "filename" "
Authentication Error	"Authentication error "
File write error	"Could not store configuration to "filename" "

例：

以下の例は、この関数をどのように使用するかを示しています。結果のページはフォームをそれ自体に送信し、続いてこれが SaveDataToFile コマンドによって評価されます。

```
<HTML>
<HEAD><TITLE>SaveDataToFile Test</TITLE></HEAD>
<BODY>

<?--#exec cmd_argument='SaveDataToFile("\stuff.txt", "Meat", Overwrite) '-->

<FORM action="test.shtm">
  <P>
    <LABEL for="Fruit">Fruit: </LABEL><BR>
    <INPUT type="text" name="Fruit"><BR><BR>

    <LABEL for="Meat">Meat: </LABEL><BR>
    <INPUT type="text" name="Meat"><BR><BR>

    <LABEL for="Bread">Bread: </LABEL><BR>
    <INPUT type="text" name="Bread"><BR><BR>

    <INPUT type="submit" name="_submit">
  </P>
</FORM>

</BODY>
</HTML>
```

結果のファイル ('stuff.txt') には、'Meat' と呼ばれるフォームオブジェクトに指定された値が含まれます。

**注意：**この例が機能するためには、HTML ファイルに 'test.shtm' と名前を付ける必要があります。

### 7.3.9 printf()

この関数は、Anybus モジュールまたはアプリケーション (あるいはその両方) からのデータが含まれる場合がある、フォーマットされた文字列を返します。使用されるフォーマット構文は、標準の C 関数 printf() の構文と類似しています。

この関数は、ゼロ以上のフォーマットタグが含まれたテンプレート文字列を受け入れ、次に多くの引数が続きます。各フォーマットタグは単一の引数に対応し、ユーザがリードできるフォームにその引数をどのように変換するのかを判断します。

構文：

```
<?--#exec cmd_argument='printf("template" [, argument1, ..., argumentN])'-->
```

**template-** 引数がどのように表されるのかを決定するテンプレートです。後続の引数によって置き換えられ、要求に従ってフォーマットされた、任意の数のフォーマットタグが含まれる場合があります。フォーマットタグの数は引数の数に一致する必要があります。一致しない場合、結果は定義されません。

フォーマットタグは以下のようにライトされます。

```
%[Flags][Width][.Precision][Modifier]type
```

下記も参照してください。

- 43 ページの "フォーマットタグ"

**argument-** ソース引数。オプションのパラメータであり、テンプレート文字列に挿入されるデータの実際のソースを指定します。引数の数はフォーマットタグの数に一致する必要があります。一致しない場合、結果は定義されません。

ライトの際には、許容される唯一の引数は ABCCMessage() です。

下記も参照してください。

- 47 ページの "ABCCMessage()"

デフォルト出力：

シナリオ	デフォルト出力
Success	(printf() result)
ABCCMessage error	ABCCMessage エラー文字列 ( 51 ページの "エラー" を参照)

例：

下記も参照してください。

- 47 ページの "ABCCMessage()"
- 50 ページの "例 (Get\_Attribute)：" "

## フォーマットタグ

- **タイプ (必須)**

タイプ文字は必須であり、以下のように基本的な表示を決定します。

タイプ文字	表示	例
c	単一の文字	b
d, i	符号付き 10 進整数	565
e, E	指数表記での浮動小数点数	5.6538e2
f	通常の固定小数点表示法での浮動小数点数	565.38
g, G	指数が -4 より小さいか精度以上の場合は、%e または %E が使用されます。それ以外の場合は、%f が使用されます。後続のゼロと小数点はプリントされません。	565.38
o	符号なし 8 進数表記	1065
s	文字列	Text
u	符号なし 10 進整数	4242
x, X	16 進整数	4e7f
%	(文字としての) %。代入は行われません。	%

- **フラグ (オプション)**

フラグ文字	意味
-	任意の幅内で結果を左詰めにします (デフォルトは右詰め)。
+	常に '+' または '-' を含めて、数値が正なのか負なのかを示します。
(空白文字)	数値が '+' または '-' で開始されない場合、代わりに空白文字を前に付けます。
0 (ゼロ)	空白文字の代わりにゼロでフィールドを埋めます。
#	%e、%E、および %f では、次に数字が続かない場合であっても数値に小数点を含めることを強制します。%x および %X の場合、0x または 0X をそれぞれ前に付けます。

- **幅 (オプション)**

幅	意味
数値	プリントされる最小文字数を指定します。プリントされる値がこの数値よりも短い場合、フィールド幅を構成するために結果が埋められます。結果が大きかったとしても、結果は切り捨てられません。
*	幅はフォーマット文字列で指定されず、フォーマットされる必要がある引数が前に付けられた整数値で指定されます。

- **精度 (オプション)**

このフィールドの厳密な意味はタイプ文字によって異なります。

タイプ文字	意味
d, i, o, u, x, X	プリントされる最小の小数桁数を指定します。プリントされる値がこの数値よりも短い場合、結果は空白文字で埋められます。結果が大きかったとしても、結果は切り捨てられないことに注意してください。
e, E, f	プリントされる小数点の後の桁数を指定します (デフォルトは 6)。
g, G	プリントされる最大の有効桁数を指定します。
s	プリントされる最大の文字数を指定します。
c	(影響なし)

- 変更子

変更子文字	意味
h	引数は SINT8、SINT16、UINT8、または UINT16 として解釈されます。
l	引数は SINT32 または UINT32 として解釈されます。

### 7.3.10 scanf()

出力ではなく入力に使用されることを除き、この関数は前半で説明した printf() 関数と非常に類似しています。この関数は HTML フォームオブジェクトから渡された文字列をリードし、テンプレート文字列によって指定されたように文字列を解析し、結果のデータを指定された引数に送信します。使用されるフォーマット構文は、標準の C 関数 scanf() の構文と類似しています。

この関数は、ソースやゼロ以上のフォーマットタグが含まれたテンプレートを受け入れ、次に多くの引数が続きます。各引数は、HTML フォームからリードされたデータを宛先の引数に送信する前に、それがどのように解釈されるのかを判断するフォーマットタグに対応します。

**注意：**このコマンドは、電子メールメッセージでは使用できません。

構文：

```
<?--#exec cmd_argument='scanf("source", "template" [,
                                argument1, ..., argumentN)']-->
```

source - 文字列が抽出される HTML フォームオブジェクトの名前です。

template- データをどのように解析して解釈するのかを指定するテンプレートです。後続の引数にデータを送信する前に変換を判断する、任意の数のフォーマットタグが含まれる場合があります。フォーマットタグの数は引数の数に一致する必要があります。一致しない場合、結果は定義されません。

フォーマットタグは以下のようにライトされます。

```
 %[*][Width][Modifier]type
```

下記も参照してください。

- 45 ページの "フォーマットタグ"

argument- 解釈されたデータの送信先を指定する宛先の引数。引数の数はフォーマットタグの数に一致する必要があります。一致しない場合、結果は定義されません。

ライトの際には、許容される唯一の引数は ABCCMessage() です。

下記も参照してください。

- 47 ページの "ABCCMessage()"

デフォルト出力:

シナリオ	デフォルト出力
Success	"Success"
Parsing error	"Incorrect data format "
Too much data for argument	"Too much data "
ABCC Message error	ABCCMessage エラー文字列 ( 51 ページの " エラー ")

例:

下記も参照してください。

- 47 ページの "ABCCMessage()"
- 50 ページの " 例 (Set\_Attribute) : "

## フォーマットタグ

- **タイプ (必須)**

タイプ文字は必須であり、以下のように基本的な表示を判断します。

タイプ	入力	引数データタイプ
c	単一の文字	CHAR
d	符号付き 10 進整数を受け入れます。	SINT8 SINT16 SINT32
i	符号付きまたは符号なし 10 進整数を受け入れます。入力データの最初の文字によって判断され、10 進数、16 進数、または 8 進数として提供される場合があります。 <u>最初の文字: フォーマット:</u> 0x 16 進数 0 8 進数 1...9 10 進数	SINT8/UINT8 SINT16/UINT16 SINT32/UINT32
u	オプションとして符号付き 10 進整数を受け入れます。	UINT8 UINT16 UINT32
o	オプションとして符号付き 8 進整数を受け入れます。	SINT8/UINT8 SINT16/UINT16 SINT32/UINT32
x, X	オプションとして符号付き 16 進整数を受け入れます。	SINT8/UINT8 SINT16/UINT16 SINT32/UINT32
e, E、 f、 g, G	オプションとして符号付き浮動小数点数を受け入れます。浮動小数点数の入力フォーマットは、一部のオプションの文字が含まれた数字の文字列です。 - 符号付きの値である可能性があります。 - 整数が続く 'E' または 'e' で構成された、次に指数フィールドが続く 10 進の有理数が含まれた指数の値である可能性があります。	FLOAT
n	入力を消費しません。対応する引数は、scanf がその内部にオブジェクト入力からリードされた文字数をライトする整数です。	SINT8/UINT8 SINT16/UINT16 SINT32/UINT32
s	非空白文字のシーケンスを受け入れます。	STRING

タイプ	入力	引数データタイプ
[scanset]	スキャンリストによって指定された、予期されるバイトのセットから非空白文字のシーケンスを受け入れます ('[0123456789ABCDEF]' など)。文字としての ']' は、セットの最初の文字として指定できません。最初の文字 ']' 直後のキャレット文字 (^) は、スキャンリストを反転させます。つまり、リストに記載された文字を除き、全ての文字を許可します。	STRING
%	この時点で単一の '%' 入力を受け入れます。代入または変換は実行されません。完全な変換指定は '%%' とする必要があります。	-

- \* (オプション)  
データはリードされますが、無視されます。対応する引数に代入されません。
- 幅 (オプション)  
リードされる最大文字数を指定します。
- 変更子 (オプション)  
異なるデータサイズを指定します。

変更子	意味
h	SINT8、SINT16、UINT8、または UINT16
l	SINT32 または UINT32

## 7.4 引数の関数

### 7.4.1 概要

引数の関数は、特定のコマンド関数へのパラメータとして提供されます。

一般的な構文：

(コンテキストにより構文が異なる)

引数の関数：

関数	説明	ページ
ABCCMessage()	-	47

### 7.4.2 ABCCMessage()

この関数は、オブジェクト要求をモジュール内またはホストアプリケーション内のオブジェクトに発行します。

構文：

```
ABCCMessage(object, instance, command, ce0, ce1,
             msgdata, c_type, r_type)
```

object - 宛先オブジェクトを指定します。

object - 宛先インスタンスを指定します。

command- コマンド番号を指定します。

ce0 - コマンドメッセージの CmdExt[0] を指定します。

ce1 - コマンドメッセージの CmdExt[1] を指定します。

msgdata- コマンド内の MsgData[] サブフィールドの実際のコンテンツを指定します。

- データは直接的なフォームで提供できます(フォーマットは c\_type に依存)。
- キーワード "ARG" は、データが親コマンド (scanf() など) によって提供されると使用されます。

c\_type - コマンド (msgdata) 内でデータタイプを指定します。

下記も参照してください。

- 48 ページの " コマンドデータタイプ (c\_type) "

r\_type - レスポンス (msgdata) 内でデータタイプを指定します。

下記も参照してください。

- 48 ページの " レスポンスデータタイプ (r\_type) "

数値入力は、以下のフォーマットで提供できます。

10 進数 (50 など)- (プレフィックスなし)  
 8 進数 (043 など)- プレフィックス 0 (ゼロ)  
 16 進数 (0x1f など)- プレフィックス 0x

下記も参照してください。

- 50 ページの " 例 (Get\_Attribute) : "
- 50 ページの " 例 (Set\_Attribute) : "

- **コマンドデータタイプ (c\_type)**

配列をサポートするタイプの場合、エレメントの数はサフィックス '[n]' を使用して指定できますが、この場合、'n' はエレメントの数を指定します。各データエレメントは、空白文字によって区切る必要があります。

タイプ	サポートされる配列	(msgdata での提供における) データフォーマット
BOOL	はい	1
SINT8	はい	-25
SINT16	はい	2345
SINT32	はい	-2569
UINT8	はい	245
UINT16	はい	40000
UINT32	はい	32
CHAR	はい	A
STRING	いいえ	"abcde" 注意：バックスラッシュ（\）が前に付けられている場合、文字列内に引用符を含めることができます。 例："We usually refer to it as \"the Egg\""
FLOAT	はい	5.6538e2
NONE	いいえ	コマンドはデータを保持しないためデータタイプはありません。

- **レスポンスデータタイプ (r\_type)**

配列をサポートするタイプの場合、エレメントの数はサフィックス '[n]' を使用して指定できますが、この場合、'n' はエレメントの数を指定します。

タイプ	サポートされる配列	コメント
BOOL	はい	オプションとして、値 (true または false) に基づいて BOOL データをメッセージと交換することができます。このような場合、関数から返される実際のデータタイプは STRING になります。 構文：BOOL<true><false> 配列の場合、フォーマットは BOOL[n]<true><false> になります。
SINT8	はい	-
SINT16	はい	-
SINT32	はい	-
UINT8	はい	このタイプもオブジェクトから ENUM データタイプをリードする際に使用できます。このような場合、実際の ENUM 値が返されます。
UINT16	はい	-
UINT32	はい	-
CHAR	はい	-
STRING	いいえ	-
ENUM	いいえ	このデータタイプを使用すると、ABCCMessage() 関数は最初に ENUM 値をリードします。次に、'Get Enum String' コマンドを発行して、実際の列挙文字列を取得します。レスポンス内の実際のデータタイプは STRING になります。



タイプ	サポートされる配列	コメント
FLOAT	はい	-
NONE	いいえ	レスポンスはデータを保持しないためデータタイプはありません。

**重要:** メッセージがアドレス指定されたオブジェクトに透過的に渡されるということに注意する必要があります。SSI エンジンではオブジェクトアドレス指定方式の違反に対するチェックを実行しません。例えば、(不当に) メッセージデータを含む不正な形式の `Get_Attribute` 要求は、これが明らかに誤りである場合であっても、変更されずにオブジェクトに渡されます。これに従わない場合、データの損失や他の不必要な副次的影響が発生することがあります。

*例 (Get\_Attribute) :*

この例は、printf() および ABCCMessage() を使用して、IP アドレスを取得する方法を示しています。

```
<?--#exec cmd_argument='printf( "%u.%u.%u.%u",
                                ABCCMessage(4,3,1,5,0,0,NONE,UINT8[4] ) )'-->
```

変数	値	コメント
object	4	ネットワークコンフィグレーションオブジェクト (04h)
instance	3	インスタンス #3 (IP アドレス)
command	1	Get_attribute
ce0	5	アトリビュート #5
ce1	0	-
msgdata	0	-
c_type	NONE	コマンドメッセージがデータを保持していません。
r_type	UINT8[4]	4つの符号なしの8ビット整数の配列

下記も参照してください。

- 42 ページの "printf()"

*例 (Set\_Attribute) :*

この例は、scanf() および ABCCMessage() を使用して、IP アドレスを設定する方法を示しています。(scanf() によって解析された) 渡されたフォームデータを使用するようにモジュールに指示する、特別なパラメータ値 'ARG' に注意してください。

```
<?--#exec cmd_argument='scanf("IP", "%u.%u.%u.%u",
                                ABCCMessage(4,3,2,5,0,ARG,UINT8[4],NONE ) )'-->
```

変数	値	コメント
object	4	ネットワークコンフィグレーションオブジェクト (04h)
instance	3	インスタンス #3 (IP アドレス)
command	2	Set_attribute
ce0	5	アトリビュート #5
ce1	0	-
msgdata	ARG	scanf() の呼び出しによって解析されたデータを使用してください。
c_type	UINT8[4]	4つの符号なしの8ビット整数の配列
r_type	NONE	レスポンスメッセージがデータを保持していません。

下記も参照してください。

- 44 ページの "scanf()"

## エラー

オブジェクト要求が結果としてエラーになる場合、レスポンス内のエラーコードが評価され、以下のようにユーザが読むことのできるフォームに変換されます。

エラーコード	出力
0	"Unknown error"
1	"Unknown error"
2	"Invalid message format"
3	"Unsupported object"
4	"Unsupported instance"
5	"Unsupported command"
6	"Invalid CmdExt[0]"
7	"Invalid CmdExt[1]"
8	"Attribute access is not set-able"
9	"Attribute access is not get-able"
10	"Too much data in msg data field"
11	"Not enough data in msg data field"
12	"Out of range"
13	"Invalid state"
14	"Out of resources"
15	"Segmentation failure"
16	"Segmentation buffer overflow"
17...255	"Unknown error"

下記も参照してください。

- 52 ページの "SSI 出力コンフィグレーション "

## 7.5 SSI 出力コンフィグレーション

オプションとして、ファイル '\output.cfg' を追加して、SSI 出力を永久に変更することができます。

ファイルフォーマット：

```
[ABCCMessage_X]
0:"Success string"
1:"Error string 1"
2:"Error string 2"
...
16:"Error string 16"
```

各エラーコードは、1～16のラベルが付けられた専用の出力文字列に対応します。

下記も参照してください。

- 51 ページの "エラー "

```
[GetConfigItem_X]
0:"Success string"
1:"Authentication error string"
2:"File open error string"
3:"Tag not found string"
```

ファイルの名前を含めるには '%s' を使用します。

```
[SetConfigItem_X]
0:"Success string"
1:"Authentication error string"
2:"File open error string"
3:"File write error string"
```

ファイルの名前を含めるには '%s' を使用します。

```
[IncludeFile_X]
0:"Success string"
1:"Authentication error string"
2:"File readS error string"
```

ファイルの名前を含めるには '%s' を使用します。

```
[scanf_X]
0:"Success string"
1:"Parsing error string"
```

```
[ChangeLanguage_X]
0:"Success string"
1:"Change error string"
```

上記の全てのコンテンツは、異なる言語の各タグ内の値 'X' を変更することで、複数回ファイルに含めることができます。次に、モジュールは言語設定に基づいて正確な出力文字列を選択します。選択された言語の情報が見つからない場合、デフォルトの SSI 出力を使用します。

Xの値	言語
0	英語
1	ドイツ語
2	スペイン語
3	イタリア語
4	フランス語

下記も参照してください。

- 38 ページの "SsiOutput()"

## 8. CIP オブジェクト

### 8.1 概要

この章では、モジュール内の CIP オブジェクトを記述します。これらのオブジェクトはネットワークからアクセスできますが、ホストアプリケーションが直接アクセスすることはできません。

必須オブジェクト：

- 54 ページの "Identity オブジェクト (01h) "
- 57 ページの " メッセージルータ (02h) "
- 58 ページの " アセンブリオブジェクト (04h) "
- 61 ページの " コネクションマネージャ (06h) "
- 64 ページの " パラメータオブジェクト (0Fh) "
- 68 ページの "DLR オブジェクト (47h) "
- 69 ページの "QoS オブジェクト (48h) "
- 72 ページの " ポートオブジェクト (F4h) " (オプション)
- 74 ページの "TCP/IP インターフェースオブジェクト (F5h) "
- 78 ページの "Ethernet リンクオブジェクト (F6h) "

ベンダ特有オブジェクト：

- 70 ページの "ADI オブジェクト (A2h) "

CIP フォワーディング機能を使用して、ホストアプリケーション内に追加の CIP オブジェクトを実装することができます。137 ページの "EtherNet/IP ホストオブジェクト (F8h) " および 142 ページの " コマンドの詳細：Process\_CIP\_Object\_Request" を参照してください。

非コネクション型 CIP ルーティングがサポートされますが、このことは最初にコネクションをセットアップしなくてもメッセージをデバイスに送信できることを意味します。

## 8.2 Identity オブジェクト (01h)

### カテゴリ

拡張

### オブジェクトの説明

-

### サポートされるサービス

クラス :           Get\_Attribute\_Single  
                  Get\_Attributes\_All

インスタンス :   Get\_Attribute\_Single  
                  Set\_Attribute\_Single  
                  Get\_Attributes\_All  
                  Reset

## クラスのアトリビュート

#	名前	アクセス	タイプ	値
1	Revision	Get	UINT	0001h (オブジェクトリビジョン)

## インスタンス #1 アトリビュート

### 拡張

#	名前	アクセス	タイプ	値
1	Vendor ID	Get	UINT	005Ah (HMS Industrial Networks AB) <sup>a</sup>
2	Device Type	Get	UINT	0000h (汎用デバイス) <sup>a</sup>
3	Product Code	Get	UINT	002Eh (Anybus CompactCom EtherNet/IP 2-Port) <sup>a</sup>
4	Revision	Get	構成 : {USINT, USINT}	メジャーおよびマイナーのファームウェアリビジョン <sup>a</sup>
5	Status	Get	WORD	56 ページの " デバイスステータス " を参照
6	Serial Number	Get	UDINT	一意のシリアルナンバー (HMS によって割り当て済み) <sup>a</sup>
7	Product Name	Get	SHORT_STR ING	"Anybus-CC EtherNet/IP (2-Port)" (製品名) <sup>a</sup>
11	Active language	Set	構成 : USINT USINT USINT	このインスタンスに送信された要求はアプリケーションオブジェクトに転送されます。これに続いてホストアプリケーションは適宜、言語設定のアップデートを実行します。
12	Supported Language List	Get	配列 : 構成 : USINT USINT USINT	ホストアプリケーションによってサポートされる言語のリスト。このリストは、NW_INIT 状態の間にアプリケーションオブジェクトからリードされ、CIP 標準に変換されます。

a. EtherNet/IP ホストオブジェクトを実装することで、カスタマイズすることができます。137 ページの "EtherNet/IP ホストオブジェクト (F8h) " を参照してください。

## デバイスステータス

ビット	名前
0	Module Owned
1	(予約)
2	Configured <sup>a</sup>
3	(予約)
4...7	拡張デバイスステータス : 値:           意味: 0000b       不明 0010b       失敗した I/O コネクション 0011b       I/O コネクション未確立 0100b       不揮発性コンフィグレーション不良 0110b       コネクションが Run モード 0111b       コネクションが Idle モード (その他)   (予約)
8	回復可能なマイナーフォールトの場合にセット <sup>b</sup>
9	回復不能なマイナーフォールトの場合にセット <sup>b</sup>
10	回復可能なメジャーフォールトの場合にセット <sup>b</sup>
11	回復不能なメジャーフォールトの場合にセット <sup>b</sup>
12...15	(予約)

a. このビットは、製品が "工場出荷時設定" 以外の設定が指定されている場合に表示されます。アプリケーションオブジェクト内に設定済みのアトリビュートが指定されている場合、またはモジュールの NV ストレージがデフォルトから変更されている場合 (あるいはその両方)、値は true に設定されます。

b. 85 ページの "診断オブジェクト (02h)" を参照してください。

## サービスの詳細 : リセットサービス

モジュールは、リセット要求をネットワークからホストアプリケーションに転送します。ネットワークのリセット処理の詳細については、一般の "Anybus-CompactCom Design Guide" を参照してください。

EtherNet/IP に対するネットワークリセット要求には、以下の 2 つのタイプがあります。

- タイプ 0 : 'Power Cycling Reset'**  
 このサービスはモジュールの出力サイクルをエミュレートし、Anybus のリセットタイプ 0 (出力サイクル) に対応します。詳細については、一般の "Anybus-CompactCom Software Design Guide" を参照してください。
- タイプ 1 : 'Out of box reset'**  
 このサービスは "工場出荷時設定" コンフィグレーションを設定してリセットを実行し、Anybus のリセットタイプ 2 (出力サイクル + 工場デフォルト) に対応します。詳細については、一般の "Anybus-CompactCom Software Design Guide" を参照してください。



## 8.3 メッセージルータ (02h)

### カテゴリ

拡張

### オブジェクトの説明

-

### サポートされるサービス

クラス : -

インスタンス : -

### クラスのアトリビュート

-

### インスタンスのアトリビュート

-

## 8.4 アセンブリオブジェクト (04h)

### カテゴリ

拡張

### オブジェクトの説明

アセンブリオブジェクトは静的アセンブリを使用し、ホストアプリケーションによって送信 / 受信されたプロセスデータを保持します。使用されるデフォルトのアセンブリインスタンス ID は、ベンダ特有の範囲内にあります。

下記も参照してください。

- 21 ページの " プロセスデータ "
- 137 ページの "EtherNet/IP ホストオブジェクト (F8h) "

### サポートされるサービス

クラス :           Get\_Attribute\_Single  
 インスタンス :    Get\_Attribute\_Single  
                   Set\_Attribute\_Single

### クラスのアトリビュート

#	名前	アクセス	タイプ	値
1	Revision	Get	UINT	0002h (オブジェクトリビジョン)
2	Max Instance	Get	UINT	(最大のインスタンス番号)

### インスタンス 03h アトリビュート (ハートビート、Input-Only)

#### 拡張

このインスタンスは、Input-Only コネクションのハートビートとして使用されます。Forward\_Open 要求内のハートビートインスタンスのデータサイズは 0 バイトである必要がありますが、他の値も許容されます。

#	名前	アクセス	タイプ	値
3	Data	Set	該当なし	- (このアトリビュートのデータサイズは 0)

このインスタンスのインスタンス番号は、EtherNet/IP ホストオブジェクト内に対応するアトリビュートを実装することで変更できます。

## インスタンス 04h アトリビュート (ハートビート、Listen-Only)

### 拡張

このインスタンスは、Listen-Only コネクションのハートビートとして使用されます。Forward\_Open 要求内のハートビートインスタンスのデータサイズは 0 バイトである必要がありますが、他の値も許容されます。

#	名前	アクセス	タイプ	値
3	Data	Set	該当なし	- (このアトリビュートのデータサイズは 0)

このインスタンスのインスタンス番号は、EtherNet/IP ホストオブジェクト内に対応するアトリビュートを実装することで変更できます。

## インスタンス 05h アトリビュート (コンフィグレーションデータ)

### 拡張

'Forward\_Open' サービスを通して送信されたコンフィグレーションデータは、このインスタンスへライトされます。

#	名前	アクセス	タイプ	値
3	Data	Get/Set	該当なし	- (Forward_Open コマンドにコンフィグレーションデータが含まれている際にアプリケーションにライトされたコンフィグレーションデータ)

このインスタンスのインスタンス番号は、EtherNet/IP ホストオブジェクト内に対応するアトリビュートを実装することで変更できます。

下記も参照してください。

- 143 ページの " コマンドの詳細 : Set\_Configuration\_Data"

## インスタンス 06h アトリビュート (ハートビート、拡張 Input-Only)

### 拡張

このインスタンスは拡張 Input-Only コネクションのハートビートとして使用され、どのアトリビュートも含まれません。このインスタンスに実行されたコネクションの状態は、Anybus-CompactCom モジュールの状態に影響しません。つまり、コネクションにタイムアウトが発生しても、モジュールは Error 状態に切り替わりません。Forward\_Open 要求内のハートビートインスタンスのデータサイズは 0 バイトである必要がありますが、他の値も許容されます。

このインスタンスのインスタンス番号は、EtherNet/IP ホストオブジェクト内に対応するアトリビュートを実装することで変更できます。

## インスタンス 07h アトリビュート (ハートビート、拡張 Listen-Only)

### 拡張

このインスタンスは、拡張 Listen-Only コネクションのハートビートとして使用され、どのアトリビュートも含まれません。このインスタンスに実行されたコネクションの状態は、Anybus-CompactCom モジュールの状態に影響しません。つまり、コネクションにタイムアウトが発生しても、モジュールは Error 状態に切り替わりません。Forward\_Open 要求内のハートビートインスタンスのデータサイズは 0 バイトである必要がありますが、他の値も許容されます。

このインスタンスのインスタンス番号は、EtherNet/IP ホストオブジェクト内に対応するアトリビュートを実装することで変更できます。

## インスタンス 64h アトリビュート (Producing インスタンス)

### 拡張

このインスタンスのインスタンス番号は、EtherNet/IP ホストオブジェクト内に対応するアトリビュートを実装することで変更できます。

#	名前	アクセス	タイプ	値
3	Produced Data	Get	BYTE の配列	このデータはライトプロセスデータに対応します。

下記も参照してください。

- 21 ページの "ネットワークのデータ交換"
- 137 ページの "EtherNet/IP ホストオブジェクト (F8h)" (インスタンスアトリビュート #7)

## インスタンス 96h アトリビュート (Consuming インスタンス)

### 拡張

このインスタンスのインスタンス番号は、EtherNet/IP ホストオブジェクト内に対応するアトリビュートを実装することで変更できます。

#	名前	アクセス	タイプ	値
3	Consumed Data	Set	BYTE の配列	このデータはリードプロセスデータに対応します。

下記も参照してください。

- 21 ページの "ネットワークのデータ交換"
- 137 ページの "EtherNet/IP ホストオブジェクト (F8h)" (インスタンスアトリビュート #8)

## 8.5 コネクションマネージャ (06h)

### カテゴリ

拡張

### オブジェクトの説明

-

### サポートされるサービス

クラス : -  
インスタンス : Forward\_Open  
Forward\_Close

### インスタンスの説明

(サポートされるインスタンスアトリビュートなし)

### Class 1 コネクションの詳細

#### 概要

Class 1 コネクションは I/O データを転送するために使用され、アセンブリオブジェクト内のインスタンスに対して確立できます。各 Class 1 コネクションは、2 つのデータ伝送 (consuming および producing) を確立します。ハートビートインスタンスは、入力のみアクセスすることになるコネクションに使用できます。Class 1 コネクションは UDP 伝送を使用します。

- サポートされた Class 1 コネクションの合計数 :4
- 最大入力コネクションサイズ :256 バイト
- 最大出力コネクションサイズ :256 バイト
- サポートされる API : 2...3200ms
- T->O コネクションタイプ :Point-to-point、マルチキャスト
- O->T コネクションタイプ :Point-to-point
- サポートされるトリガタイプ :サイクリック、COS

## コネクシオンタイプ

- **Exclusive-Owner コネクション**

このタイプのコネクションは Anybus モジュールの出力を制御し、他のコネクションには依存しません。

- Exclusive-Owner コネクションの最大数 :  
1
- コネクションポイント O ⇒ T :  
アセンブリオブジェクト、インスタンス 64h  
(デフォルト)
- コネクションポイント T ⇒ O :  
アセンブリオブジェクト、インスタンス 96h  
(デフォルト)

- **Input-Only コネクション**

このタイプのコネクションは、出力を制御しないで Anybus モジュールからのデータをリードするために使用されます。他のコネクションには依存しません。

- Input-Only コネクションの最大数 :  
最大 4<sup>1</sup>
- コネクションポイント O ⇒ T :  
アセンブリオブジェクト、インスタンス 03h  
(デフォルト)
- コネクションポイント T ⇒ O :  
アセンブリオブジェクト、インスタンス 96h  
(デフォルト)

**注意 :** Exclusive-Owner コネクションがモジュールに対して開かれていてタイムアウトが発生する場合、Input-Only コネクションにもタイムアウトが発生します。Exclusive-Owner コネクションが適切に閉じられた場合は、Input-Only コネクションにも影響はありません。

- **拡張 Input-Only コネクション**

このコネクション機能は、標準の Input-Only コネクションと同一です。ただし、このコネクションにタイムアウトが発生しても、アプリケーションの状態に影響しません。

- コネクションポイント O ⇒ T :  
アセンブリオブジェクト、インスタンス 05h  
(デフォルト)
- コネクションポイント T ⇒ O :  
アセンブリオブジェクト、インスタンス 64h  
(デフォルト)

- **Listen-Only コネクション**

このタイプのコネクションは、存在するために別のコネクションを必要とします。そのコネクション (Exclusive-Owner または Input-Only) が閉じられると、Listen-Only コネクションも閉じられます。

- Input-Only コネクションの最大数 :最大 4<sup>2</sup>

- 
1. Exclusive-Owner コネクションと Listen-Only コネクションで共有
  2. Exclusive-Owner コネクションと Input-Only コネクションで共有

- コネクションポイント O ⇒ T :  
アセンブリオブジェクト、インスタンス 04h  
(デフォルト)
- コネクションポイント T ⇒ O :  
アセンブリオブジェクト、インスタンス 96h  
(デフォルト)
- **拡張 Listen-Only コネクション**  
このコネクション機能は、標準の Listen-Only コネクションと同一です。ただし、このコネクションにタイムアウトが発生しても、アプリケーションの状態に影響しません。
  - コネクションポイント O ⇒ T :  
アセンブリオブジェクト、インスタンス 07h  
(デフォルト)
  - コネクションポイント T ⇒ O :  
アセンブリオブジェクト、インスタンス 64h  
(デフォルト)
- **Redundant-Owner コネクション**  
このコネクションタイプはモジュールによってサポートされません。

### Class 3 コネクションの詳細

- **Explicit メッセージコネクション**  
Class 3 コネクションは、メッセージルータに対してコネクションを確立するために使用されます。従って、コネクションは Explicit メッセージに使用されます。Class 3 コネクションは、TCP 伝送を使用します。
  - 同時の Class 3 コネクションの数 : 16
  - サポートされる API : 2 ~ 10000ms
  - T->O コネクションタイプ : Point-to-point
  - O->T コネクションタイプ : Point-to-point
  - サポートされるトリガタイプ : アプリケーション

## 8.6 パラメータオブジェクト (0Fh)

### カテゴリ

拡張

### オブジェクトの説明

このオブジェクトを使用すると、RSNetworx などのコンフィグレーションツールがアプリケーションデータインスタンス (ADI) に関する情報を抽出し、その実際の名前と範囲を使用してユーザに表示できるようになります。

特にシリアルホストインターフェースを使用している場合、このプロセスにはある程度の時間がかかるため、EtherNet/IP ホストオブジェクト内でこの機能のサポートを無効にすることができます。

CIP 標準によって課される制限により、複数のエレメントが含まれている ADI (つまり配列など) はこのオブジェクトを通して表すことができません。このような場合、デフォルト値が返されます。66 ページの "デフォルト値" を参照してください。

下記も参照してください。

- 66 ページの "デフォルト値"
- 70 ページの "ADI オブジェクト (A2h)" (CIP オブジェクト)
- 137 ページの "EtherNet/IP ホストオブジェクト (F8h)" (ホストアプリケーションオブジェクト)

### サポートされるサービス

クラス :           Get\_Attribute\_Single

インスタンス :    Get\_Attribute\_Single  
                  Set\_Attribute\_Single  
                  Get\_Attributes\_All  
                  Get\_Enum\_String



## クラスのアトリビュート

#	名前	アクセス	タイプ	値
1	Revision	Get	UINT	0001h (オブジェクトのリビジョン)
2	Max instance	Get	UINT	作成された最大のインスタンス番号 = アプリケーションデータオブジェクト内のクラスアトリビュート 3 <sup>a</sup>
8	Parameter class descriptor	Get	WORD	デフォルト : 0000 0000 0000 01011b ビット : 内容 : 0    パラメータインスタンスをサポート 1    完全なアトリビュートをサポート 2    不揮発性ストレージ保存コマンドの実行が必要 3    パラメータは不揮発性ストレージ内に保存
9	Configuration Assembly instance	Get	UINT	0000h (コンフィグレーションアセンブリはサポートされない)

a. 詳細については、一般の "Anybus-CompactCom Software Design Guide" を参照してください。

## インスタンスの属性

### 拡張

#	名前	アクセス	タイプ	値
1	Parameter Value	Get/Set	アトリビュート 4、5、6 で指定される	パラメータの実際の値 このアトリビュートはアトリビュート #4 のビット 4 が true の場合にリードオンリー
2	Link Path Size	Get	USINT	0007h
3	Link Path	Get	Packed EPATH	20 A2 25 nn nn 30 05h (このパラメータの値の取得元のオブジェクトに対するパス、この場合は ADI オブジェクト)
4	Descriptor	Get	WORD	<u>ビット : 内容 :</u> 0 設定可能なパスをサポート (該当なし) 1 列挙された文字列をサポート 2 スケーリングをサポート (該当なし) 3 スケーリングリンクをサポート (該当なし) 4 リードオンリーパラメータ 5 監視パラメータ (該当なし) 6 拡張精度スケーリングをサポート (該当なし)
5	Data type	Get	EPATH	データタイプコード
6	Data size	Get	USINT	パラメータ値内のバイト数
7	Parameter Name String	Get	SHORT_STRING	パラメータの名前、16 文字に切り捨て
8	Units String	Get	SHORT_STRING	(サポートされない)
9	Help String	Get	SHORT_STRING	
10	Minimum value	Get	(データタイプ)	パラメータの最小値
11	Maximum value	Get	(データタイプ)	パラメータの最大値
12	Default value	Get	(データタイプ)	パラメータのデフォルト値
13	Scaling Multiplier	Get	UINT	0001h (サポートされない)
14	Scaling Divisor	Get	UINT	
15	Scaling Base	Get	UINT	
16	Scaling Offset	Get	INT	0000h (サポートされない)
17	Multiplier link	Get	UINT	
18	Divisor Link	Get	UINT	
19	Base Link	Get	UINT	
20	Offset Link	Get	UINT	
21	Decimal precision	Get	USINT	

### デフォルト値

#	名前	値	コメント
1	Parameter Value	0	-
2	Link Path Size	0	リンクパスのサイズ (バイト単位)
3	Link Path	-	ヌルパス
4	Descriptor	0010h	リードオンリーパラメータ
5	Data type	C6h	USINT

#	名前	値	コメント
6	Data size	1	-
7	Parameter Name String	(予約)	-
8	Units String	""	-
9	Help String	""	-
10	Minimum value	該当なし	0
11	Maximum value	該当なし	0
12	Default value	該当なし	0

## 8.7 DLR オブジェクト (47h)

### カテゴリ

拡張

### オブジェクトの説明

-

### サポートされるサービス

クラス :           Get\_Attribute\_Single  
                  Get\_Attributes\_All

インスタンス :    Get\_Attribute\_Single  
                  Set\_Attribute\_Single

### クラスのアトリビュート

#	名前	アクセス	タイプ	値
1	Revision	Get	UINT	0002h (オブジェクトリビジョン)

### インスタンス #1 アトリビュート

#### 拡張

#	名前	アクセス	タイプ	値
1	Network Topology	Get	USINT	ビット:内容: 0 "リニア" 1 "リング"
2	Network Status	Get	USINT	ビット:内容: 0 "通常" (該当なし) 1 "リングの障害" 2 "検出された予期しないループ" 3 "部分的なネットワーク障害" 4 "高速な障害 / 回復サイクル"
10	Active Supervisor Address	Get	構成 : {UDINT、 6つの USINT の配列}	このアトリビュートは、IP アドレス (IPv4) または アクティブなリングスーパーバイザの Ethernet Mac アドレス (あるいはその両方) を保持します。
12	Capability Flags	Get	DWORD	01h (アナウンスベースのリングノード)

## 8.8 QoS オブジェクト (48h)

### カテゴリ

拡張

### オブジェクトの説明

-

### サポートされるサービス

クラス : Get\_Attribute\_Single

インスタンス : Get\_Attribute\_Single  
Set\_Attribute\_Single

### クラスのアトリビュート

#	名前	アクセス	タイプ	値
1	Revision	Get	UINT	0001h (オブジェクトリビジョン)

### インスタンス #1 アトリビュート

#### 拡張

#	名前	アクセス	タイプ	値
1	802.1Q Tag Enable	Set	USINT	802.1Q フレームの送信の有効化または無効化 ビット:内容: 0 無効 (デフォルト) 1 有効
4	DSCP Urgent	Set	USINT	優先順位が Urgent の CIP 伝送 Class 1 メッセージ デフォルト: 55
5	DSCP Scheduled	Set	USINT	優先順位が Scheduled の CIP 伝送 Class 1 メッセージ デフォルト: 47
6	DSCP High	Set	USINT	優先順位が High の CIP 伝送 Class 1 メッセージ デフォルト: 43
7	DSCP Low	Set	USINT	優先順位が Low の CIP 伝送 Class 1 メッセージ デフォルト: 31
8	DSCP Explicit	Set	USINT	CIP UCMM および CIP Class 3 デフォルト: 27

## 8.9 ADI オブジェクト (A2h)

### カテゴリ

拡張

### オブジェクトの説明

このオブジェクトは、アプリケーションデータオブジェクト内で EtherNet/IP にインスタンスをマッピングします。このオブジェクトへの全ての要求は、ホストアプリケーション内のアプリケーションデータオブジェクトに対する明示的なオブジェクト要求に変換されます。次に、レスポンスは CIP フォーマットに再度変換され、要求の発信元に送信されます。

下記も参照してください。

- アプリケーションデータオブジェクト ("Anybus-CompactCom Software Design Guide" を参照)
- 64 ページの "パラメータオブジェクト (0Fh) " (CIP オブジェクト)

### サポートされるサービス

クラス : Get\_Attribute\_Single

インスタンス : Get\_Attribute\_Single  
Set\_Attribute\_Single

### クラスのアトリビュート

#	名前	アクセス	タイプ	値
1	Revision	Get	UINT	オブジェクトバージョン (現在値 = 0001h)
2	Max Instance	Get	UINT	アプリケーションデータオブジェクト内のアトリビュート #4 に等しい <sup>a</sup>
3	Number of instances	Get	UINT	アプリケーションデータオブジェクト内のアトリビュート #3 に等しい <sup>a</sup>

a. 詳細については、一般の "Anybus-CompactCom Software Design Guide" を参照してください。

## インスタンスの属性

各インスタンスは、アプリケーションデータオブジェクト内部のインスタンスに対応します（詳細については、一般の "Anybus-CompactCom Software Design Guide" を参照）。

### 拡張

#	名前	アクセス	タイプ	説明
1	Name	Get	SHORT_STRING	パラメータ名（長さを含む）
2	ABCC Data type	Get	USINT	インスタンス値のデータタイプ
3	No. of elements	Get	USINT	指定されたデータタイプのエレメントの数
4	Descriptor	Get	USINT	このインスタンスのアクセス権を説明するビットフィールド ビット：意味： 0 Set = Get アクセス 1 Set = Set アクセス
5	Value <sup>a</sup>	Get/Set	アトリビュート #2 によって決定 される	インスタンス値
6	Max value <sup>a</sup>	Get		最大の許容パラメータ値
7	Min value <sup>a</sup>	Get		最小の許容パラメータ値
8	Default value <sup>a</sup>	Get		デフォルトのパラメータ値

a. CIP 標準を対象にモジュールによって変換されます。

## 8.10 ポートオブジェクト (F4h)

### カテゴリ

拡張

### オブジェクトの説明

このオブジェクトは、EtherNet/IP ホストオブジェクト(インスタンスアトリビュート #17) 内で有効になっている場合にのみ存在します。

下記も参照してください。

- 137 ページの "EtherNet/IP ホストオブジェクト (F8h) " (Anybus モジュールオブジェクト)
- 134 ページの "CIP ポートコンフィグレーションオブジェクト (0Dh) " (ホストアプリケーションオブジェクト)

### サポートされるサービス

クラス :           Get\_Attributes\_All  
                  Get\_Attribute\_Single

インスタンス :   Get\_Attributes\_All  
                  Get\_Attribute\_Single

### クラスのアトリビュート

#	名前	アクセス	タイプ	値
1	Revision	Get	UINT	オブジェクトバージョン (現在値 = 0001h)
2	Max Instance	Get	UINT	最大のインスタンス番号
3	Number of instances	Get	UINT	ポートの数
8	Entry Port	Get	UINT	この要求のデバイスへの入力に使用されたポートを説明するポートオブジェクトのインスタンスを返します。
9	Port Instance Info	Get	配列 :	各インスタンスからのインスタンスアトリビュート 1 と 2 を含む構造の配列。配列は、インスタンス番号によってインスタンスの最大数までインデックスが作成されます。インデックス 1 (オフセット 0) および任意のインスタンス化されていないインスタンスの値は、0 になります。
			構成 : UINT (タイプ) UINT (ナンバー)	ポートのタイプを列挙します (インスタンスアトリビュート #1 を参照)。 このポートに関連付けられた CIP ポート番号 (インスタンス #2 を参照)。



## インスタンスの属性 (インスタンス #1)

### 拡張

このインスタンスは、Ethernet インターフェースに関連付けられた特性を反映します。

#	名前	アクセス	タイプ	値
1	Port Type	Get	UINT	0h (デフォルト) 4h (アプリケーションがポートを登録する場合)
2	Port Number	Get	UINT	2h
3	Link Object	Get	構成 : UINT Padded EPATH	- 2h 20 F5 24 01h
4	Port Name	Get	SHORT_STRING	"Ethernet/IP"
7	Node Address	Get	Padded EPATH	-
8	Port Node Range	Get	構成 : UINT (最小) UINT (最大)	-

下記も参照してください。

- 134 ページの "CIP ポートコンフィグレーションオブジェクト (0Dh) "

## インスタンスの属性 (インスタンス #2...#8)

### 拡張

#	名前	アクセス	タイプ	値
1	Port Type	Get	UINT	ポートのタイプを列挙します。
2	Port Number	Get	UINT	このポートに関連付けられた CIP ポート番号
3	Link Object	Get	構成 : UINT Padded EPATH	- パスの長さ (16 ビット文字の数)。 このポートのオブジェクトを識別する論理パスセグメントパスは、1つの論理クラスセグメントと1つの論理インスタンスセグメントから構成される必要があります。最大サイズは 12 バイトです。
4	Port Name	Get	SHORT_STRING	ポートの名前、"Port A" など。最大 64 文字。
7	Node Address	Get	Padded EPATH	ポート上のこのデバイスのノード番号。このデータタイプ内部の範囲はポートセグメントに制限されます。
8	Port Node Range	Get	構成 : UINT (最小) UINT (最大)	- ポート上の最小のノード番号 ポート上の最大のノード番号

下記も参照してください。

- 134 ページの "CIP ポートコンフィグレーションオブジェクト (0Dh) " (135 ページの "インスタンスの属性")

## 8.11 TCP/IP インターフェースオブジェクト (F5h)

### カテゴリ

拡張

### オブジェクトの説明

オブジェクトはTCP/IP 関連の設定をグループ化します。下記も参照してください。

- 20 ページの " 通信設定 "
- 87 ページの " ネットワークコンフィギュレーションオブジェクト (04h) " (Anybus モジュールオブジェクト)

### サポートされるサービス

クラスサービス :                   Get\_Attribute\_All  
  Get\_Attribute\_Single

インスタンスサービス :            Get\_Attribute\_All  
  Get\_Attribute\_Single  
  Set\_Attribute\_Single<sup>1</sup>

### クラスのアトリビュート

#	アクセス	名前	タイプ	値	説明
1	Get	Revision	UINT	0002h	リビジョン 2

1. このサービスのサポートは、EtherNet/IP ホストオブジェクト内にアトリビュート #9 を実装することで無効できます。

## インスタンスのアトリビュート

## 拡張

#	アクセス	名前	タイプ	値	コメント
1	Get	Status	DWORD	-	<p><u>ビット#:</u> 意味:</p> <p>0 ~ 3: 1hに設定されると、アトリビュート #5にはDHCPまたは不揮発性ストレージからの有効なコンフィグレーションが含まれます。2hに設定されると、アトリビュート #5にはハードウェア設定からの有効なコンフィグレーションが含まれます。残りの値は将来的な使用に予約されています。</p> <p>4: 1に設定される場合、マルチキャスト保留。</p> <p>5: (予約、0に設定)</p> <p>6: AcdStatus。アドレス競合が検出されると1に設定されます。アドレス競合の検出はアトリビュート #10 で有効/無効になります。</p> <p>7 ~ 31: (予約、0に設定)</p>
2	Get	Configuration Capability	DWORD	0000 0086h - または - 0000 0096h - または - 0000 00A6h - または - 0000 00B6h	<p><u>ビット#:</u> 意味:</p> <p>0: (予約、0に設定)</p> <p>1: 常に1。モジュールは、DNSサーバをクエリすることでホスト名を解決できます。</p> <p>2: 常に1。モジュールは、DHCPを通してネットワークコンフィグレーションを取得できます。</p> <p>3: (予約、0に設定)</p> <p>4: 'Configuration Settable' ビットは、137 ページの "EtherNet/IP ホストオブジェクト (F8h) " 内のインスタンスアトリビュート #9 の値を反映します。</p> <p>5: モジュールは、このビットが1に設定されると hardware configurable です。</p> <p>6: (予約、0に設定)</p> <p>7: 常に1。デバイスはアドレス競合を検出できます。</p> <p>8 ~ 31: (予約、0に設定)</p>
3	Get/Set	Configuration Control	DWORD	-	<p><u>値:</u> 意味:</p> <p>0: 不揮発性メモリからのコンフィグレーション</p> <p>2: DHCPからのコンフィグレーション</p>

#	アクセス	名前	タイプ	値	コメント
4	Get	Port Object	構成 : UINT (パス サイズ) Padded EPATH	- 0002h 20 F6 24 01h	- 2 ワード Ethernet リンクオブジェクトへのパス、イ ンスタンス #1
5	Get/Set	Interface Configuration	構成 : UDINT (IP) UDINT (マスク) UDINT (GW) UDINT (DNS1) UDINT (DNS2) STRING (ドメイン)	-	- IP アドレス  サブネットマスク  デフォルトゲートウェイ  プライマリ DNS  セカンダリ DNS  デフォルトドメイン
6	Get/Set	Host Name	STRING	-	Anybus モジュールのホスト名
7	(使用されない)				
8	Get/Set	TTL Value	USINT	1	EtherNet/IP マルチキャストパケットの TTL 値
9	Get/Set	Mcast Config	構成 :		
		Alloc Control	USINT	0	値 :           意味 : 0 :           マルチキャストアドレスを生 成するために、デフォルトの 割り当てアルゴリズムを使用 します。 1 :           'Num Mcast' フィールドおよび 'Mcast Start Addr' フィールド 内の値に従って、マルチキャ ストアドレスを割り当てます。
		(予約)	USINT	0	ゼロに設定。変更しないでください。
		Num Mcast	UINT	1	EtherNet/IP に割り当てるマルチキャスト アドレスの数
		Mcast Start Addr	UDINT	-	割り当てを開始する最初のマルチキャスト アドレス
10	Set	SelectAcd	BOOL	1	値 :           意味 : 0 :           ACD の無効化。 1 :           ACD の有効化 (デフォルト)。 ACD (アドレス競合の検出) が有効な場合、アトリビュ ート #1 内のビット 6 は ACD 競 合が検出される場合に設定さ れます。ネットワークステ ータス LED も検出された競合を 示します。11 ページの "ネッ トワークステータス LED" を 参照してください。

#	アクセス	名前	タイプ	値	コメント
11	Set	LastConflictDetected	構成 :		ACD 診断パラメータ。最後に検出された競合に関連しています。 最後に競合が検出された際の ACD アクティビティの状態。 競合が検出された ARP PDU からのリモートノードの MAC アドレス。 競合が検出された未処理の ARP PDU のコピー。
		Acctivity	USINT	-	
		RemoteMAC	6 つの USINT の配列	-	
		ArpPdu	28 の USINT の配列	-	
12	Set	EIP QuickConnect <sup>a</sup>	BOOL	0	値 : 0 : 1 : 意味 : EIP QuickConnect の無効化 (デフォルト)。 EIP QuickConnect の有効化。 EIP QuickConnect が有効になると、すばやく起動して EtherNet/IP ネットワークに加わるように EIP QuickConnect 機能が EtherNet/IP の対象デバイスに指示を与えます。

a. モジュールが EIP QuickConnect 機能を使用するように設定されている場合、EDS ファイルを変更する必要があります。EDS ファイルが変更されるため、モジュールの識別を変更する必要があります。14 ページの "コンフォーマンステストガイド" を参照してください。

## 8.12 Ethernet リンクオブジェクト (F6h)

### カテゴリ

拡張

### オブジェクトの説明

このオブジェクトは、Ethernet インターフェースの診断情報をグループ化します。

下記も参照してください。

- 20 ページの " 通信設定 "
- 87 ページの " ネットワークコンフィギュレーションオブジェクト (04h) " (Anybus モジュールオブジェクト)

### サポートされるサービス

クラスサービス :                   Get\_Attribute\_All  
  Get\_Attribute\_Single

インスタンスサービス :           Get\_Attribute\_All  
  Get\_Attribute\_Single  
  Set\_Attribute\_Single  
  Get\_And\_Clear

### クラスのアトリビュート

#	アクセス	名前	タイプ	値	コメント
1	Get	Revision	UINT	3	リビジョン 3
2	Get	Max Instance	UINT	3	インスタンス 3 が最大インスタンス
3	Get	Number of instances	UINT	3	3 つのインスタンス

### インスタンスのアトリビュート

#### 拡張

#	アクセス	名前	タイプ	値	コメント
1	Get	Interface Speed	UDINT	10 または 100	実際のイーサネットインターフェース速度
2	Get	Interface Flags	DWORD	-	81 ページの " インターフェースフラグ " を参照
3	Get	Physical Address	6 つの USINT の配列	(MAC ID)	物理ネットワークアドレス

#	アクセス	名前	タイプ	値	コメント
4	Get	Interface Counters	構成 :		
		In Octets	UDINT	該当なし	インターフェース上で受信されたオクテット
		In Ucast Packets	UDINT	該当なし	インターフェース上で受信されたユニキャストパケット
		In NUcast Packets	UDINT	該当なし	インターフェース上で受信された非ユニキャストパケット
		In Discards	UDINT	該当なし	不明なプロトコルを使用したインバウンドパケット
		In Errors	UDINT	該当なし	エラーが含まれたインバウンドパケット (In Discards は含まれない)
		In Unknown Protos	UDINT	該当なし	不明なプロトコルを使用したインバウンドパケット
		Out Octets	UDINT	該当なし	インターフェース上で送信されたオクテット
		Out Ucast Packets	UDINT	該当なし	インターフェース上で送信されたユニキャストパケット
		Out NUcast Packets	UDINT	該当なし	インターフェース上で送信された非ユニキャストパケット
		Out Discards	UDINT	該当なし	不明なプロトコルを使用したアウトバウンドパケット
		Out Errors	UDINT	該当なし	エラーが含まれたアウトバウンドパケット (Out Discards は含まれない)
5	Get	Media Counters	構成 :		メディア特有のカウンタ
		Alignment Errors	UDINT	該当なし	長さが整数のオクテットではない受信フレーム
		FCS Errors	UDINT	該当なし	FCS チェックを通過しない受信フレーム
		Single Collisions	UDINT	該当なし	1つの衝突のみ発生した正常送信済みフレーム
		Multiple Collisions	UDINT	該当なし	複数の衝突が発生した正常送信済みフレーム
		SQE Test Errors	UDINT	0	SQE テストエラーメッセージが生成される回数 (現在の PHY インターフェースにはカウンタは提供されない)
		Deferred Transmissions	UDINT	該当なし	媒体がビジー状態であるために最初の送信試行が遅延されるフレーム
		Late Collisions	UDINT	該当なし	パケットの送信に対し 512 ビットの回数以後にパケットの衝突が検出された回数
		Excessive Collisions	UDINT	該当なし	過剰な衝突が原因で送信が失敗するフレーム
		MAC Transmit Errors	UDINT	該当なし	内部 MAC サブレイヤ受信エラーが原因で送信が失敗するフレーム
		Carrier Sense Errors	UDINT	該当なし	フレームを送信しようとする際に状況が失われた、あるいは有効な状態にならなかったことをキャリアが検知した回数
		Frame Too Long	UDINT	該当なし	最大許容フレームサイズを超えている受信フレーム
		MAC Receive Errors	UDINT	該当なし	内部 MAC サブレイヤ受信エラーが原因でインターフェース上での受信に失敗するフレーム

#	アクセス	名前	タイプ	値	コメント
6	Get/Set <sup>a</sup>	Interface Control	構成 :		
		Control Bits	WORD	-	インターフェース制御ビット
		Forced Interface Speed	UINT	-	インターフェースに対して強制される動作速度。オートネゴシエーションが有効な場合、'Object state Conflict' を返します。
7	Get	Interface Type	USINT	-	81 ページの " インターフェースタイプ " を参照
10	Get	Interface Label	SHORT_STRING	-	81 ページの " インターフェースラベル " を参照

a. このアトリビュートのサポートは、EtherNet/IP ホストオブジェクト内にアトリビュート #9 を実装することで無効にできます。



## インターフェースフラグ

ビット	名前	説明
0	Link status	Ethernet 802.3 通信インターフェースがアクティブなネットワークに接続されているかどうかを示します。 値: 意味: 0 非アクティブなリンク 1 アクティブなリンク
1	Half/full duplex	現在使用されている二重モードを示します。 値: 意味: 0 半二重 1 全二重
2 ~ 4	Negotiation Status	リンクのオートネゴシエーションのステータスを示します。 値: 意味: 0 オートネゴシエーションが進行中 1 オートネゴシエーションおよび速度検出に失敗 (デフォルト値を使用) 2 オートネゴシエーションに失敗、ただし速度を検出 (デフォルトの二重値を使用) 3 速度および二重のネゴシエーションに成功 4 オートネゴシエーションの試みなし。強制された速度および二重。
5	Manual Setting requires Reset	値: 意味: 0 インターフェースはランタイム中にリンクパラメータへの変更をアクティブにできる 1 変更を有効にするにはリセットが必要
6	Local Hardware Fault	値: 意味: 0 ローカルなハードウェア障害が検出されない 1 ローカルなハードウェア障害が検出された
7 ~ 31	(予約)	(無視する)

## インターフェースタイプ

インスタンス	値	説明
1	2	ツイストペア
2	2	ツイストペア
3	1	内部インターフェース

## インターフェースラベル

インスタンス	値
1	Port 1
2	Port 2
3	Internal

## 9. Anybus モジュールオブジェクト

### 9.1 概要

この章では、Anybus モジュールオブジェクトの実装とこれらが Anybus-CompactCom EtherNet/IP 内の機能にどのように対応するのかを記述します。

標準オブジェクト：

- 83 ページの "Anybus オブジェクト (01h) "
- 85 ページの " 診断オブジェクト (02h) "
- 86 ページの " ネットワークオブジェクト (03h) "
- 87 ページの " ネットワークコンフィグレーションオブジェクト (04h) "

ネットワーク特有のオブジェクト：

- 95 ページの " ソケットインターフェースオブジェクト (07h) "
- 115 ページの "SMTP クライアントオブジェクト (09h) "
- 120 ページの " ファイルシステム・インターフェースオブジェクト (0Ah) "
- 19 ページの "Network Ethernet Object (0Ch)"
- 134 ページの "CIP ポートコンフィグレーションオブジェクト (0Dh) "

## 9.2 Anybus オブジェクト (01h)

### カテゴリ

基本

### オブジェクトの説明

このオブジェクトは、全ての一般的な Anybus データをアSEMBルします。これについては、一般の "Anybus-CompactCom Software Design Guide" で詳細が説明されています。

### サポートされるコマンド

オブジェクト :    Get\_Attribute  
インスタンス :    Get\_Attribute  
                  Set\_Attribute  
                  Get\_Enum\_String

### オブジェクトのアトリビュート (インスタンス #0)

(詳細については、一般の "Anybus-CompactCom Software Design Guide" を参照。)

## インスタンスの属性 (インスタンス #1)

### 基本

#	名前	アクセス	タイプ	値
1	Module type	Get	UINT16	0401h (標準の Anybus-CompactCom)
2...11	-	-	-	詳細については、一般の "Anybus-CompactCom Software Design Guide" を参照してください。
12	LED colours	Get	構成 : UINT8 (LED1A) UINT8 (LED1B) UINT8 (LED2A) UINT8 (LED2B)	値 : 色 : 01h 緑 02h 赤 01h 緑 02h 赤
13...15	-	-	-	詳細については、一般の "Anybus-CompactCom Software Design Guide" を参照してください。

### 拡張

#	名前	アクセス	タイプ	値
16	GPIO configuration	Get/Set <sup>a</sup>	UINT16	ホストインターフェースの GPIO ピンのコンフィグレーション。以下の表を参照してください。

a. アトリビュート GPIO コンフィグレーションの Set アクセスは、状態 SETUP でのみ有効です。

### GPIO コンフィグレーションの設定

値	機能	説明
0x0000	標準	汎用の入力 / 出力ピンとして GIP[0..1] と GOP[0..1] が使用されます。
0x0001	拡張 LED 機能	GIP1 はリンク / アクティビティ LED (ポート 1) に使用されます。10/100MHz の表示は 1 つの信号に組み合わされて緑の LED に接続されますが、これはリンクが存在する場合は点灯し、アクティビティの場合は点滅します。 GOP1 はリンク / アクティビティ LED (ポート 2) に使用されます。10/100MHz の表示は 1 つの信号に組み合わされて緑の LED に接続されますが、これはリンクが存在する場合は点灯し、アクティビティの場合は点滅します。

下記も参照してください。

- 151 ページの "アペンディックス B" の "拡張 LED 機能"

## 9.3 診断オブジェクト (02h)

### 概要

基本

### オブジェクトの説明

このオブジェクトは、ホストアプリケーションのイベントと診断を処理する標準化された方法を提供します。詳細については、一般の "Anybus-CompactCom Software Design Guide" を参照してください。

### サポートされるコマンド

オブジェクト :   Get\_Attribute  
                  Create  
                  Delete

インスタンス :   Get\_Attribute

### オブジェクトの属性 (インスタンス #0)

#	名前	アクセス	データタイプ	値
1...4	-	-	-	詳細については、一般の "Anybus-CompactCom Software Design Guide" を参照してください。
11	Max no. of instances	Get	UINT16	5+1

### インスタンスの属性

#### 基本

#	名前	アクセス	タイプ	値
1	Severity	Get	UINT8	詳細については、一般の "Anybus-CompactCom Software Design Guide" を参照してください。
2	Event Code	Get	UINT8	

この実装では、全てのインスタンスの重大度レベルが組み合わされ (論理 'OR' を使用)、CIP Identity オブジェクトを通してネットワーク上に表されます。Event Code はネットワーク上で表すことができないため、モジュールによって無視されます。

下記も参照してください。

- 20 ページの " 診断 "
- 54 ページの "Identity オブジェクト (01h) " (CIP オブジェクト)

## 9.4 ネットワークオブジェクト (03h)

### カテゴリ

基本

### オブジェクトの説明

このオブジェクトの詳細情報については、一般の "Anybus-CompactCom Software Design Guide" を参照してください。

### サポートされるコマンド

オブジェクト : Get\_Attribute  
 インスタンス : Get\_Attribute  
                   Set\_Attribute  
                   Get\_Enum\_String  
                   Map\_ADI\_Write\_Area  
                   Map\_ADI\_Read\_Area

### オブジェクトのアトリビュート (インスタンス #0)

(詳細については、一般の "Anybus-CompactCom Software Design Guide" を参照。)

### インスタンスのアトリビュート (インスタンス #1)

#### 基本

#	名前	アクセス	タイプ	値
1	Network type	Get	UINT16	009Ch
2	Network type string	Get	CHAR の配列	'EtherNet/IP (2-Port)'
3	Data format	Get	ENUM	00h (LSB ファースト)
4	Parameter data support	Get	BOOL	True
5	Write process data size	Get	UINT16	現在のライトプロセスデータのサイズ (バイト) Map_ADI_Write_Area の成功ごとにアップデートされる <sup>a</sup>
6	Read process data size	Get	UINT16	現在のリードプロセスデータのサイズ (バイト) Map_ADI_Read_Area の成功ごとにアップデートされる <sup>a</sup>
7	Exception Information	Get	UINT8	値 意味 0 : 使用できる情報なし 1 : 無効なアセンブリインスタンスマッピング
8...10	-	-	-	詳細については、一般の "Anybus-CompactCom Software Design Guide" を参照してください。

a. 詳細については、一般の "Anybus-CompactCom Software Design Guide" を参照してください。

## 9.5 ネットワークコンフィグレーションオブジェクト (04h)

### カテゴリ

拡張、高度

### オブジェクトの説明

このオブジェクトは、エンドユーザによって設定される場合があるネットワーク特有のコンフィグレーションパラメータを保持します。このオブジェクトに対して発行されるリセットコマンド（工場出荷時設定）によって、全てのインスタンスがデフォルト値に設定されるようになります。

下記も参照してください。

- 20 ページの " 通信設定 "
- 74 ページの "TCP/IP インターフェースオブジェクト (F5h) " (CIP オブジェクト)
- 78 ページの "Ethernet リンクオブジェクト (F6h) " (CIP オブジェクト)
- 32 ページの " 電子メールクライアント "

**注意：**フィールドバス適合性を実現するために、このオブジェクトに関して "Anybus-CompactCom Software Design Guide" に記載された推奨事項に従う必要があります。

### サポートされるコマンド

オブジェクト：   Get\_Attribute  
                  リセット

インスタンス：    Get\_Attribute  
                  Set\_Attribute  
                  Get\_Enum\_String

### オブジェクトのアトリビュート（インスタンス #0）

（詳細については、一般の "Anybus-CompactCom Software Design Guide" を参照。）

## インスタンスの属性（インスタンス #3、IP アドレス）

変更はリセット後に有効になります。

### 高度

#	名前	アクセス	タイプ	説明
1	Name <sup>a</sup>	Get	CHAR の配列	'IP address'
2	Data type	Get	UINT8	04h (= UINT8)
3	Number of elements	Get	UINT8	04h (4 エlement)
4	Descriptor	Get	UINT8	07h (リード/ライト/共有アクセス)
5	Value	Get/Set	UINT8 の配列	有効な範囲 : 0.0.0.0 ~ 255.255.255.255 (デフォルト = 0.0.0.0)

a. 多言語、94 ページの " 多言語文字列 " を参照してください。

## インスタンスの属性（インスタンス #4、サブネットマスク）

変更はリセット後に有効になります。

### 高度

#	名前	アクセス	タイプ	説明
1	Name <sup>a</sup>	Get	CHAR の配列	'Subnet mask'
2	Data type	Get	UINT8	04h (= UINT8)
3	Number of elements	Get	UINT8	04h (4 エlement)
4	Descriptor	Get	UINT8	07h (リード/ライト/共有アクセス)
5	Value	Get/Set	UINT8 の配列	有効な範囲 : 0.0.0.0 ~ 255.255.255.255 (デフォルト = 0.0.0.0)

a. 多言語、94 ページの " 多言語文字列 " を参照してください。

## インスタンスの属性（インスタンス #5、ゲートウェイ）

変更はリセット後に有効になります。

### 高度

#	名前	アクセス	タイプ	説明
1	Name <sup>a</sup>	Get	CHAR の配列	'Gateway'
2	Data type	Get	UINT8	04h (= UINT8)
3	Number of elements	Get	UINT8	04h (4 エlement)
4	Descriptor	Get	UINT8	07h (リード/ライト/共有アクセス)
5	Value	Get/Set	UINT8 の配列	有効な範囲 : 0.0.0.0 ~ 255.255.255.255 (デフォルト = 0.0.0.0)

a. 多言語、94 ページの " 多言語文字列 " を参照してください。



## インスタンスの属性（インスタンス #6、DHCP）

変更はリセット後に有効になります。

### 拡張

#	名前	アクセス	タイプ	説明
1	Name <sup>a</sup>	Get	CHAR の配列	'DHCP'
2	Data type	Get	UINT8	08h (= ENUM)
3	Number of elements	Get	UINT8	01h (1 エlement)
4	Descriptor	Get	UINT8	07h (リード/ライト/共有アクセス)
5	Value <sup>a</sup>	Get/Set	ENUM	値: Enum. 文字列: 意味: 00h 'Disable' DHCP は無効 01h 'Enable' DHCP は有効 (デフォルト)

a. 多言語、94 ページの "多言語文字列" を参照してください。

## インスタンスの属性（インスタンス #7、Ethernet 通信設定 1）

変更はすぐに有効になります。

### 高度

#	名前	アクセス	タイプ	説明
1	Name <sup>a</sup>	Get	CHAR の配列	'Comm 1'
2	Data type	Get	UINT8	08h (= ENUM)
3	Number of elements	Get	UINT8	01h (1 エlement)
4	Descriptor	Get	UINT8	07h (リード/ライト/共有アクセス)
5	Value <sup>a</sup>	Get/Set	ENUM	値: Enum. 文字列: 意味: 00h 'Auto' オートネゴシエーション (デフォルト) 01h '10 HDX' 10Mbit、半二重 02h '10 FDX' 10Mbit、全二重 03h '100 HDX' 100Mbit、半二重 04h '100 HDX' 100Mbit、全二重

a. 多言語、94 ページの "多言語文字列" を参照してください。

## インスタンスの属性 (インスタンス #8、Ethernet 通信設定 2)

変更はすぐに有効になります。

### 高度

#	名前	アクセス	タイプ	説明
1	Name <sup>a</sup>	Get	CHAR の配列	'Comm 2'
2	Data type	Get	UINT8	08h (= ENUM)
3	Number of elements	Get	UINT8	01h (1 エlement)
4	Descriptor	Get	UINT8	07h (リード/ライト/共有アクセス)
5	Value <sup>a</sup>	Get/Set	ENUM	値: Enum. 文字列: 意味: 00h 'Auto' 自動ネゴシエーション (デフォルト) 01h '10 HDX' 10Mbit、半二重 02h '10 FDX' 10Mbit、全二重 03h '100 HDX' 100Mbit、半二重 04h '100 HDX' 100Mbit、全二重

a. 多言語、94 ページの "多言語文字列" を参照してください。

## インスタンスの属性 (インスタンス #9、DNS1)

このインスタンスは、プライマリ DNS サーバへのアドレスを保持します。変更はリセット後に有効になります。

### 高度

#	名前	アクセス	タイプ	説明
1	Name <sup>a</sup>	Get	CHAR の配列	'DNS1'
2	Data type	Get	UINT8	04h (= UINT8)
3	Number of elements	Get	UINT8	04h
4	Descriptor	Get	UINT8	07h (リード/ライト/共有アクセス)
5	Value	Get/Set	UINT8 の配列	有効な範囲: 0.0.0.0 ~ 255.255.255.255 (デフォルト = 0.0.0.0)

a. 多言語、94 ページの "多言語文字列" を参照してください。

## インスタンスの属性 (インスタンス #10、DNS2)

このインスタンスは、セカンダリ DNS サーバへのアドレスを保持します。変更はリセット後に有効になります。

### 高度

#	名前	アクセス	タイプ	説明
1	Name <sup>a</sup>	Get	CHAR の配列	'DNS2'
2	Data type	Get	UINT8	04h (= UINT8)
3	Number of elements	Get	UINT8	04h
4	Descriptor	Get	UINT8	07h (リード/ライト/共有アクセス)
5	Value	Get/Set	UINT8 の配列	有効な範囲 : 0.0.0.0 ~ 255.255.255.255 (デフォルト =0.0.0.0)

a. 多言語、94 ページの " 多言語文字列 " を参照してください。

## インスタンスの属性 (インスタンス #11、ホスト名)

このインスタンスは、モジュールのホスト名を保持します。変更はリセット後に有効になります。

### 高度

#	名前	アクセス	タイプ	説明
1	Name <sup>a</sup>	Get	CHAR の配列	'Host name'
2	Data type	Get	UINT8	07h (= CHAR)
3	Number of elements	Get	UINT8	40h
4	Descriptor	Get	UINT8	07h (リード/ライト/共有アクセス)
5	Value	Get/Set	CHAR の配列	ホスト名、64 文字 (完全な長さにするために空白文字で埋められる)

a. 多言語、94 ページの " 多言語文字列 " を参照してください。

## インスタンスの属性 (インスタンス #12、ドメイン名)

このインスタンスは、ドメイン名を保持します。変更はリセット後に有効になります。

### 高度

#	名前	アクセス	タイプ	説明
1	Name <sup>a</sup>	Get	CHAR の配列	'Domain name'
2	Data type	Get	UINT8	07h (= CHAR)
3	Number of elements	Get	UINT8	40h
4	Descriptor	Get	UINT8	07h (リード/ライト/共有アクセス)
5	Value	Get/Set	CHAR の配列	ドメイン名、64 文字 (完全な長さにするために空白文字で埋められる)

a. 多言語、94 ページの " 多言語文字列 " を参照してください。

## インスタンスの属性 (インスタンス #13、SMTP サーバ)

このインスタンスは、SMTP サーバアドレスを保持します。変更はリセット後に有効になります。

### 高度

#	名前	アクセス	タイプ	説明
1	Name <sup>a</sup>	Get	CHAR の配列	'SMTP Server'
2	Data type	Get	UINT8	04h (= UINT8)
3	Number of elements	Get	UINT8	04h
4	Descriptor	Get	UINT8	07h (リード/ライト/共有アクセス)
5	Value	Get/Set	UINT8 の配列	SMTP サーバアドレス、64 文字 (完全な長さにするために空白文字で埋められる)

a. 多言語、94 ページの " 多言語文字列 " を参照してください。

## インスタンスの属性 (インスタンス #14、SMTP ユーザ)

このインスタンスは、SMTP アカウントのユーザ名を保持します。変更はリセット後に有効になります。

### 高度

#	名前	アクセス	タイプ	説明
1	Name <sup>a</sup>	Get	CHAR の配列	'SMTP User'
2	Data type	Get	UINT8	07h (= CHAR)
3	Number of elements	Get	UINT8	40h
4	Descriptor	Get	UINT8	07h (リード/ライト/共有アクセス)
5	Value	Get/Set	UINT8 の配列	SMTP アカウントユーザ名、64 文字 (完全な長さにするために空白文字で埋められる)

a. 多言語、94 ページの "多言語文字列" を参照してください。

## インスタンスの属性 (インスタンス #15、SMTP パスワード)

このインスタンスは、SMTP アカウントのパスワードを保持します。変更はリセット後に有効になります。

### 高度

#	名前	アクセス	タイプ	説明
1	Name <sup>a</sup>	Get	CHAR の配列	'SMTP Pswd'
2	Data type	Get	UINT8	07h (= CHAR)
3	Number of elements	Get	UINT8	40h
4	Descriptor	Get	UINT8	07h (リード/ライト/共有アクセス)
5	Value	Get/Set	UINT8 の配列	SMTP アカウントユーザ名、64 文字 (完全な長さにするために空白文字で埋められる)

a. 多言語、94 ページの "多言語文字列" を参照してください。

## 多言語文字列

このオブジェクト内のインスタンス名および列挙文字列は多言語であり、以下のように現在の言語設定に基づいて翻訳されます。

インスタンス	英語	ドイツ語	スペイン語	イタリア語	フランス語
3	IP address	IP-Adresse	Dirección IP	Indirizzo IP	Adresse IP
4	Subnet mask	Subnetzmaske	Masac. subred	Sottorete	Sous-réseau
5	Gateway	Gateway	Pasarela	Gateway	Passerelle
6	DHCP	DHCP	DHCP	DHCP	DHCP
	Enable	Einschalten	Activado	Abilitato	Activé
	Disable	Ausschalten	Desactivado	Disabilitato	Désactivé
7	Comm 1	Komm 1	Comu 1	Connessione 1	Comm 1
	Auto	Auto	Auto	Auto	Auto
	10 HDX	10 HDX	10 HDX	10 HDX	10 HDX
	10 FDX	10 FDX	10 FDX	10 FDX	10 FDX
	100 HDX	100 HDX	100 HDX	100 HDX	100 HDX
8	Comm 2	Komm 2	Comu 2	Connessione 2	Comm 2
	Auto	Auto	Auto	Auto	Auto
	10 HDX	10 HDX	10 HDX	10 HDX	10 HDX
	10 FDX	10 FDX	10 FDX	10 FDX	10 FDX
	100 HDX	100 HDX	100 HDX	100 HDX	100 HDX
9	DNS1	DNS 1	DNS Primaria	DNS1	DNS1
10	DNS2	DNS 2	DNS Secundia.	DNS2	DNS2
11	Host name	Host name	Nombre Host	Nome Host	Nom hôte
12	Domain name	Domain name	Nobre Domain	Nome Dominio	Dom Domaine
13	SMTP Server	SMTP Server	Servidor SMTP	Server SMTP	SMTP serveur
14	SMTP User	SMTP User	Usuario SMTP	Utente SMTP	SMTP utiliza.
15	SMTP Pswd	SMTP PSWD	Clave SMTP	Password SMTP	SMTP mt passe

## 9.6 ソケットインターフェースオブジェクト (07h)

### カテゴリ

高度

### オブジェクトの説明

このオブジェクトは、TCP/IP スタックソケットインターフェースへの直接的なアクセスを提供し、TCP/UDP 上にカスタムプロトコルを実装できるようにします。

このオブジェクトにアクセスする際に使用されるコマンドの一部には、セグメント化が必要であることを注意してください。詳細については、153 ページの "メッセージのセグメント化" を参照してください。

**重要:** このオブジェクトによって提供される機能の使用は、ソケットインターフェースプログラミングにすでに精通していて、TCP/IP プログラミングに参与するコンセプトを完全に理解しているユーザによってのみ試行される必要があります。

### サポートされるコマンド

オブジェクト : Get\_Attribute  
 Create ( 98 ページの " コマンドの詳細 : Create" を参照)  
 Delete ( 99 ページの " コマンドの詳細 : Delete" を参照)

インスタンス : Get\_Attribute  
 Set\_Attribute  
 Bind ( 100 ページの " コマンドの詳細 : Bind" を参照)  
 Shutdown ( 101 ページの " コマンドの詳細 : Shutdown" を参照)  
 Listen ( 102 ページの " コマンドの詳細 : Listen" を参照)  
 Accept ( 103 ページの " コマンドの詳細 : Accept" を参照)  
 Connect ( 104 ページの " コマンドの詳細 : Connect" を参照)  
 Receive ( 105 ページの " コマンドの詳細 : Receive" を参照)  
 Receive\_From ( 107 ページの " コマンドの詳細 : Receive\_From" を参照)  
 Send ( 109 ページの " コマンドの詳細 : Send" を参照)  
 Send\_To ( 110 ページの " コマンドの詳細 : Send\_To" を参照)  
 IP\_Add\_membership  
 ( 111 ページの " コマンドの詳細 : IP\_Add\_Membership" を参照)  
 IP\_Drop\_membership  
 ( 112 ページの " コマンドの詳細 : IP\_Drop\_Membership" を参照)  
 DNS\_Lookup ( 113 ページの " コマンドの詳細 : DNS\_Loopup" を参照)

### オブジェクトの属性 (インスタンス #0)

#	名前	アクセス	データタイプ	値
1	Name	Get	CHAR の配列	'Socket interface'
2	Revision	Get	UINT8	01h
3	Number of instances	Get	UINT16	-
4	Highest instance no.	Get	UINT16	-
11	Max. no. of instances	Get	UINT16	0008h

## インスタンスの属性 (ソケット #1 ~ 8)

### 高度

#	名前	アクセス	タイプ	説明
1	Socket type	Get	UINT8	値: <u>ソケットタイプ:</u> 00h SOCK_STREAM、ノンブロッキング (TCP) 01h SOCK_STREAM、ブロッキング (TCP) 02h SOCK_DGRAM、ノンブロッキング (UDP) 03h SOCK_DGRAM、ブロッキング (UDP)
2	Port	Get	UINT16	ソケットがバインドされるローカルポート
3	Host IP	Get	UINT32	ホスト IP アドレス、または接続されていない場合は 0 (ゼロ)
4	Host port	Get	UINT16	ホストポート番号、または接続されていない場合は 0 (ゼロ)
5	TCP State	Get	UINT8	状態 (TCP ソケットのみ): 値: <u>状態:</u> <u>説明:</u> 00h CLOSED ソケットクローズ済み 01h LISTEN コネクションをリスン中 02h SYN_SENT 接続を確立中、SYN を送信済み 03h SYN_RECEIVED SYN の送信と受信済み 04h ESTABLISHED 接続確立済み 05h CLOSE_WAIT FIN を受信して、ソケットのクローズ待ち 06h FIN_WAIT_1 ソケットをクローズして、FIN 送信済み 07h CLOSING お互いに FIN を送信、ソケットをクローズ、ACK 受信待ち 08h LAST_ACK FIN を送信してソケットをクローズ後、ACK 受信待ち 09h FIN_WAIT_2 ソケットをクローズして、FIN 認証済み 0Ah TIME_WAIT ソケットのクローズ後、コネクションの終了待ち
6	TCP RX bytes	Get	UINT16	RX バッファ内のバイト数 (TCP ソケットのみ)
7	TCP TX bytes	Get	UINT16	TX バッファ内のバイト数 (TCP ソケットのみ)
8	Reuse address	Get/Set	BOOL	ソケットはローカルアドレスを再使用可能 値: <u>意味:</u> 1 有効 0 無効 (デフォルト)
9	Keep alive	Get/Set	BOOL	プロトコルはアイドルコネクションを確認する (TCP ソケットのみ) 値: <u>意味:</u> 1 有効 0 無効 (デフォルト)
10	IP Multicast TTL	Get/Set	UINT8	IP マルチキャスト TTL 値 (UDP ソケットのみ) デフォルト = 1
11	IP Multicast Loop	Get/Set	BOOL	IP マルチキャストループバック (UDP ソケットのみ) <sup>a</sup> 値: <u>意味:</u> 1 有効 (デフォルト) 0 無効



#	名前	アクセス	タイプ	説明
12	Ack delay time	Get/Set	UINT16	遅延 ACK の ms 単位での時間 (TCP ソケットのみ) デフォルト = 200ms <sup>b</sup>
13	TCP No Delay	Get/Set	BOOL	融合パケットの送信を遅延しない (TCP) 値: 意味: 1 遅延する (デフォルト) 0 遅延しない (ソケット上で Nagle アルゴリズムをオフにする)
14	TCP Connect Timeout	Get/Set	UINT16	秒単位での TCP コネクションタイムアウト (デフォルト = 75s)

- a. ループバックされたメッセージを取得するには、グループに属する必要があります。  
b. 分解能は 50ms、つまり、50...99 = 50ms、100...149 = 100ms、199 = 150ms などとなります。

## コマンドの詳細 : Create

### カテゴリ

高度

### 詳細

コマンドコード : 03h

有効な対象 : オブジェクトインスタンス

### 説明

このコマンドは、ソケットを作成します。

**注意 :** このコマンドは、WAIT\_PROCESS 状態、IDLE 状態、および PROCESS\_ACTIVE 状態でのみ許可されます。

- コマンドの詳細

フィールド	内容
CmdExt[0]	(予約、ゼロに設定)
CmdExt[1]	値 :     ソケットタイプ : 00h     SOCK_STREAM、ノンブロッキング (TCP) 01h     SOCK_STREAM、ブロッキング (TCP) 02h     SOCK_DGRAM、ノンブロッキング (UDP) 03h     SOCK_DGRAM、ブロッキング (UDP)

- レスポンスの詳細

フィールド	内容	コメント
Data[0]	インスタンス番号 (ロー)	作成されたソケットのインスタンス番号
Data[1]	インスタンス番号 (ハイ)	

## コマンドの詳細 : Delete

### カテゴリ

高度

### 詳細

コマンドコード : 04h

有効な対象 : オブジェクトインスタンス

### 説明

このコマンドは、以前に作成されたソケットを削除して、コネクションを閉じます (接続されている場合)。

- ソケットが TCP タイプであり、コネクションが確立されている場合、コネクションは RST フラグによって切断されます。
- TCP コネクションを問題なく切断するには、ソケットを削除する前に 'Shutdown' コマンド (101 ページの "コマンドの詳細 : Shutdown" を参照) を使用し、代わりに FIN フラグによってコネクションが閉じられるようにすることをお奨めします。
- コマンドの詳細

フィールド	内容	コメント
CmdExt[0]	削除するインスタンス番号 (ロー)	削除されるソケットのインスタンス番号
CmdExt[1]	削除するインスタンス番号 (ハイ)	

- レスポンスの詳細  
(データなし)

## コマンドの詳細 : Bind

### カテゴリ

高度

### 詳細

コマンドコード : 10h

有効な対象 : インスタンス

### 説明

このコマンドは、ソケットをローカルポートへバインドします。

- コマンドの詳細

フィールド	内容	コメント
CmdExt[0]	要求されたポート番号 (ロー)	任意の空いているポートへバインドするように要求するには、0 (ゼロ) に設定します。
CmdExt[1]	要求されたポート番号 (ハイ)	

- レスポンスの詳細

フィールド	内容	コメント
CmdExt[0]	バインドされたポート番号 (ロー)	ソケットがバインドされる実際のポート
CmdExt[1]	バインドされたポート番号 (ハイ)	

## コマンドの詳細 : Shutdown

### カテゴリ

高度

### 詳細

コマンドコード : 11h  
有効な対象 : インスタンス

### 説明

このコマンドは、FIN フラグを使用して TCP コネクションを閉じます。レスポンスはコネクションが実際にシャットダウンされたかどうかを示さないことに注意してください。このことは、ノンブロッキングソケットをポーリングするためにこのコマンドを使用できず、このコマンドはブロッキングソケットのためにブロックも行わないを意味します。

#### • コマンドの詳細

フィールド	内容
CmdExt[0]	(予約、ゼロに設定)
CmdExt[1]	値 :   モード : 00h   シャットダウン受信チャンネル 01h   シャットダウン送信チャンネル 01h   シャットダウン受信および送信の両チャンネル

#### • レスポンスの詳細

(データなし)

TCP コネクションを問題なくシャットダウンするために推奨されるシーケンスを以下に説明します。

アプリケーションがシャットダウンを開始する場合 :

- 01h に設定された CmdExt[1] を使用してシャットダウンを送信します。これにより、送信チャンネルをシャットダウンしているホストに FIN フラグが送信されます。受信チャンネルは動作可能のままであることを注意してください。
- ホストが受信チャンネルを閉じたこと示している、エラーメッセージオブジェクト特有のエラー (EDESTADDRREQ (14)) が受信されるまで、ソケット上でデータを受信します。ホストが受信チャンネルを閉じない場合、タイムアウトを使用してステップ 3 に進みます。
- ソケットインスタンスを削除します。ステップ 2 にタイムアウトが発生する場合、RST フラグが送信されてソケットが中止されます。

ホストがシャットダウンを開始する場合 :

- ソケット上の受信データは、それをゼロバイトが受信した場合、ホストがソケット上で受信チャンネルを閉じたことを示します。
- 任意の未送信のデータをホストに送信してみます。
- 01h に設定された CmdExt[1] を使用してシャットダウンを送信します。これにより、受信チャンネルをシャットダウンしているホストに FIN フラグが送信されます。
- ソケットインスタンスを削除します。

## コマンドの詳細 : Listen

### カテゴリ

高度

### 詳細

コマンドコード : 12h

有効 : インスタンス

### 説明

このコマンドは、TCP ソケットをリッスン状態にします。バックログキューの長さは、ソケット上で許可された Accept されていない接続の数です。バックログキューが一杯の場合、それ以降の接続は RST フラグを使用して拒否されます。

- コマンドの詳細

フィールド	内容	コメント
CmdExt[0]	(予約)	(ゼロに設定)
CmdExt[1]	値 : <u>バックログキューの長さ</u> : 00h 1 01h 2 02h 4	-

- レスポンスの詳細  
(データなし)

## コマンドの詳細 : Accept

### カテゴリ

高度

### 詳細

コマンドコード : 13h

有効な対象 : インスタンス

### 説明

このコマンドは、リスンの TCP ソケット上で着信のコネクションを受け入れます。受け入れられた各コネクションには、新しいソケットインスタンスが作成されます。新しいソケットはホストに接続され、レスポンスはそのインスタンス番号を返します。

#### ノンブロッキングモード :

このコマンドは、着信のコネクションのために繰り返し発行される（ポーリングされる）必要があります。着信のコネクションの要求が存在しない場合、モジュールはエラーコード 0006h (EWOULDBLOCK) とともにレスポンスを返します。

#### ブロッキングモード :

このコマンドは、コネクションの要求が検出されるまでブロックします。

**注意 :** このコマンドは、受け入れられたコネクションのために使用可能なインスタンスが存在する場合にのみ受け入れられます。ブロッキングコネクションの場合、このコマンドはインスタンスを予約します。

- コマンドの詳細  
(データなし)
- レスポンスの詳細

フィールド	内容
Data[0]	接続されたソケットのインスタンス番号 (ロー)
Data[1]	接続されたソケットのインスタンス番号 (ハイ)
Data[2]	ホスト IP アドレスバイト 3 (ロー)
Data[3]	ホスト IP アドレスバイト 2
Data[4]	ホスト IP アドレスバイト 1
Data[5]	ホスト IP アドレスバイト 0 (ハイ)
Data[6]	ホストポート番号 (ロー)
Data[7]	ホストポート番号 (ハイ)

## コマンドの詳細 : Connect

### カテゴリ

高度

### 詳細

コマンドコード : 14h  
有効な対象 : インスタンス

### 説明

SOCK\_DGRAM ソケットの場合、このコマンドはソケットが関連付けられるピアを指定します。このピアはデータグラムを送信先であり、データグラムの受信元である唯一のアドレスです。

SOCK\_STREAM ソケットの場合、このコマンドはホストへの接続を確立しようとします。

SOCK\_STREAM ソケットは 1 回のみ正常に接続できる一方、SOCK\_DGRAM ソケットは関連付けを変更するためにこのサービスを複数回使用できます。SOCK\_DGRAM ソケットは、IP アドレス 0.0.0.0、ポート 0 (ゼロ) へ接続することで、関連付けを解消できます。

#### ノンブロッキングモード :

このコマンドは、接続が確立、拒否、またはタイムアウトされるまで、繰り返し発行される (ポーリングされる) 必要があります。最初の接続の試みは受け入れられ、それ以降、接続を試みる間はコマンドがポーリング要求に対してエラーコード 22 (EINPROGRESS) を返します。

#### ブロッキングモード :

このコマンドは、接続が確立されるか、タイムアウトあるいは接続エラーにより接続の要求がキャンセルされるまでブロックします。

#### • コマンドの詳細

フィールド	内容	内容
CmdExt[0]	(予約)	(ゼロに設定)
CmdExt[1]		
Data[0]	ホスト IP アドレスバイト 3 (ロー)	-
Data[1]	ホスト IP アドレスバイト 2	
Data[2]	ホスト IP アドレスバイト 1	
Data[3]	ホスト IP アドレスバイト 0 (ハイ)	
Data[4]	ホストポート番号 (ロー)	
Data[5]	ホストポート番号 (ハイ)	

#### • レスポンスの詳細 (データなし)



## コマンドの詳細 : Receive

### カテゴリ

高度

### 詳細

コマンドコード : 15h  
有効な対象 : インスタンス

### 説明

このコマンドは、接続されたソケットからデータを受信します。最大 1472 バイトを受信するために、メッセージのセグメント化が使用される場合があります (153 ページの "メッセージのセグメント化" を参照)。

SOCK-DGRAM ソケットの場合、モジュールは次に受信されたデータグラムから要求された量のデータを返します。データグラムが要求よりも小さい場合、全データグラムがレスポンスメッセージ内で返されます。データグラムが要求よりも大きい場合、超過のバイトは破棄されます。

SOCK-STREAM ソケットの場合、モジュールは受信されたデータストリームから要求された数のバイトを返します。実際のデータサイズが要求よりも少ない場合、使用可能な全てのデータが返されます。

#### ノンブロッキングモード :

ソケット上でデータを使用できない場合、エラーコード 0006h (EWOULDBLOCK) が返されます。

#### ブロッキングモード :

モジュールは、操作が完了するまでレスポンスを発行しません。

モジュールが正常に 0 (ゼロ) バイトのデータのレスポンスを発行する場合、このことはホストが接続を閉じたことを意味します。ただし、送信チャンネルは有効なままであることがあり、'Shutdown' または 'Delete' (あるいはその両方) を使用して閉じる必要があります。

#### • コマンドの詳細

フィールド	内容	注記
CmdExt[0]	(予約)	(ゼロに設定)
CmdExt[1]	セグメント化制御ビット	154 ページの "コマンドのセグメント化" を参照
Data[0]	受信されたデータサイズ (ロー)	最初のセグメント内でのみ使用される
Data[1]	受信されたデータサイズ (ハイ)	

- レスポンスの詳細

注意：レスポンス内のデータはセグメント化される場合があります（153 ページの "メッセージのセグメント化" を参照）。

フィールド	内容	注記
CmdExt[0]	(予約)	(ゼロに設定)
CmdExt[1]	セグメント化制御ビット	155 ページの "レスポンスのセグメント化" を参照
Data[0...n]	受信されたデータ	-

## コマンドの詳細 : Receive\_From

### カテゴリ

高度

### 詳細

コマンドコード : 16h

有効な対象 : インスタンス

### 説明

このコマンドは、接続されていない SOCK\_DGRAM ソケットからデータを受信します。最大 1472 バイトを受信するために、メッセージのセグメント化が使用される場合があります (153 ページの "メッセージのセグメント化" を参照)。

モジュールは次に受信されたデータグラムから要求された量のデータを返します。データグラムが要求よりも小さい場合、全データグラムがレスポンスメッセージ内で返されます。データグラムが要求よりも大きい場合、超過のバイトは破棄されます。

レスポンスメッセージには、送信者の IP アドレスとポート番号が含まれます。

#### ノンブロッキングモード :

ソケット上でデータを使用できない場合、エラーコード 0006h (EWOULDBLOCK) が返されます。

#### ブロッキングモード :

モジュールは、操作が完了するまでレスポンスを発行しません。

#### • コマンドの詳細

フィールド	内容	注記
CmdExt[0]	(予約)	(ゼロに設定)
CmdExt[1]	セグメント化制御ビット	154 ページの "コマンドのセグメント化" を参照
Data[0]	受信されたデータサイズ (ロー)	最初のセグメント内でのみ使用される
Data[1]	受信されたデータサイズ (ハイ)	

#### • レスポンスの詳細

**注意 :** レスポンス内のデータはセグメント化される場合があります (153 ページの "メッセージのセグメント化" を参照)。

フィールド	内容	注記
CmdExt[0]	(予約)	(ゼロに設定)
CmdExt[1]	セグメント化制御ビット	155 ページの "レスポンスのセグメント化" を参照

フィールド	内容	注記
Data[0]	ホスト IP アドレスバイト 3 (ロー)	ホストアドレス / ポート情報は最初のセグメント内にも含まれます。以降の全てのデータは Data[0] で開始されます。
Data[1]	ホスト IP アドレスバイト 2	
Data[2]	ホスト IP アドレスバイト 1	
Data[3]	ホスト IP アドレスバイト 0 (ハイ)	
Data[4]	ホストポート番号 (ロー)	
Data[5]	ホストポート番号 (ハイ)	
Data[6...n]	受信されたデータ	

## コマンドの詳細 : Send

### カテゴリ

高度

### 詳細

コマンドコード : 17h

有効な対象 : インスタンス

### 説明

このコマンドは、接続されたソケットでデータを送信します。最大 1472 バイトを送信するために、メッセージのセグメント化が使用される場合があります（153 ページの "メッセージのセグメント化" を参照）。

#### ノンブロッキングモード :

送信バッファ内に使用可能な十分なバッファスペースが存在しない場合、モジュールはエラーコード 0006h (EWOULDBLOCK) とともにレスポンスを返します。

#### ブロッキングモード :

送信バッファ内に使用可能なバッファスペースが十分に存在しない場合、それが十分存在するまでモジュールはブロックします。

- コマンドの詳細

**注意 :** より大きなデータ量 (255 バイトを超えるデータ) の送信を許容するために、コマンドデータがセグメント化される場合があります（153 ページの "メッセージのセグメント化" を参照）。

フィールド	内容	注記
CmdExt[0]	(予約)	(ゼロに設定)
CmdExt[1]	セグメント化制御	154 ページの "コマンドのセグメント化" を参照
Data[0...n]	送信するデータ	-

- レスポンスの詳細

フィールド	内容	注記
CmdExt[0]	(予約)	(無視する)
CmdExt[1]		
Data[0]	送信されたバイト数 (ロー)	最後のセグメント内でのみ有効
Data[1]	送信されたバイト数 (ハイ)	

## コマンドの詳細 : Send\_To

### カテゴリ

高度

### 詳細

コマンドコード : 18h

有効な対象 : インスタンス

### 説明

このコマンドは、接続されていない SOCK-DGRAM ソケットで指定されたホストにデータを送信します。最大 1472 バイトを送信するために、メッセージのセグメント化が使用される場合があります (153 ページの "メッセージのセグメント化" を参照)。

- コマンドの詳細

**注意 :** より大きなデータ量 (255 バイトを超えるデータ) の送信を許容するために、コマンドデータがセグメント化される場合があります (153 ページの "メッセージのセグメント化" を参照)。

フィールド	内容	注記
CmdExt[0]	(予約)	(ゼロに設定)
CmdExt[1]	セグメント化制御	154 ページの "コマンドのセグメント化" を参照
Data[0]	ホスト IP アドレスバイト 3 (ロー)	ホストアドレス / ポート情報は最初のセグメント内にのみ含まれることとなります。以降の全てのデータは Data[0] で開始される必要があります。
Data[1]	ホスト IP アドレスバイト 2	
Data[2]	ホスト IP アドレスバイト 1	
Data[3]	ホスト IP アドレスバイト 0 (ハイ)	
Data[4]	ホストポート番号 (ロー)	
Data[5]	ホストポート番号 (ハイ)	
Data[6...n]	送信するデータ	

- レスポンスの詳細

フィールド	内容	注記
CmdExt[0]	(予約)	(無視する)
CmdExt[1]		
Data[0]	送信されたバイト数 (ロー)	最後のセグメント内でのみ有効
Data[1]	送信されたバイト数 (ハイ)	

## コマンドの詳細 : IP\_Add\_Membership

### カテゴリ

高度

### 詳細

コマンドコード : 19h

有効な対象 : インスタンス

### 説明

このコマンドは、ソケットに IP マルチキャスト・グループメンバーシップを割り当てます。モジュールは常に '全ホストグループ' へ自動的に参加しますが、このコマンドは最大 20 の追加メンバーシップを指定するために使用されることがあります。

- コマンドの詳細

フィールド	内容	注記
CmdExt[0]	(予約)	(ゼロに設定)
CmdExt[1]		
Data[0]	グループ IP アドレスバイト 3 (ロー)	-
Data[1]	グループ IP アドレスバイト 2	
Data[2]	グループ IP アドレスバイト 1	
Data[3]	グループ IP アドレスバイト 0 (ハイ)	

- レスポンスの詳細  
(データなし)

## コマンドの詳細 : IP\_Drop\_Membership

### カテゴリ

高度

### 詳細

コマンドコード : 1Ah

有効な対象 : インスタンス

### 説明

このコマンドは、IP マルチキャスト・グループメンバーシップからソケットを削除します。

- コマンドの詳細

フィールド	内容	注記
CmdExt[0]	(予約)	(ゼロに設定)
CmdExt[1]		
Data[0]	グループ IP アドレスバイト 3 (ロー)	-
Data[1]	グループ IP アドレスバイト 2	
Data[2]	グループ IP アドレスバイト 1	
Data[3]	グループ IP アドレスバイト 0 (ハイ)	

- レスポンスの詳細  
(データなし)



## コマンドの詳細 : DNS\_Loopup

### カテゴリ

高度

### 詳細

コマンドコード : 1Bh

有効な対象 : オブジェクトインスタンス

### 説明

このコマンドは任意のホスト名を解決し、IP アドレスを返します。

- コマンドの詳細

フィールド	内容	注記
CmdExt[0]	(予約)	(ゼロに設定)
CmdExt[1]		
Data[0...N]	ホスト名	解決するホスト名

- レスポンスの詳細 (成功)

フィールド	内容	注記
CmdExt[0]	(予約)	(ゼロに設定)
CmdExt[1]		
Data[0]	IP アドレスバイト 3 (ロー)	指定されたホストの IP アドレス
Data[1]	IP アドレスバイト 2	
Data[2]	IP アドレスバイト 1	
Data[3]	IP アドレスバイト 0 (ハイ)	

## ソケットインターフェースのエラーコード（オブジェクト特有）

以下のオブジェクト特有のエラーコードは、ソケットインターフェースオブジェクトが使用されるとモジュールによって返されることがあります。

エラーコード	名前	意味
1	ENOBUFS	使用できる内部バッファなし
2	ETIMEDOUT	タイムアウトイベントが発生
3	EISCONN	ソケットがすでに接続されている
4	EOPNOTSUPP	サービスがサポートされていない
5	ECONNABORTED	コネクションが中止された
6	EWOULDBLOCK	ノンブロッキングのソケットタイプであるためソケットがブロックできない
7	ECONNREFUSED	コネクションが拒否された
8	ECONNRESET	コネクションのリセット
9	ENOTCONN	ソケットが接続されていない
10	EALREADY	ソケットはすでに要求されたモードになっている
11	EINVAL	無効なサービスデータ
12	EMSGSIZE	無効なメッセージサイズ
13	EPIPE	パイプ内のエラー
14	EDESTADDRREQ	宛先アドレスが必要
15	ESHUTDOWN	ソケットがすでにシャットダウンされている
16	(予約)	-
17	EHAVEOOB	帯域外データが使用可能
18	ENOMEM	使用できる内部メモリなし
19	EADDRNOTAVAIL	アドレスが使用できない
20	EADDRINUSE	アドレスがすでに使用されている
21	(予約)	-
22	EINPROGRESS	サービスはすでに進行中
28	ETOOMANYREFS	リファレンスが多すぎる
101	Command aborted	コマンドがソケット上でブロックしているためにソケットが Delete コマンドを使用して閉じられる場合、このエラーコードがブロッキングコマンドに返されます。

## 9.7 SMTP クライアントオブジェクト (09h)

### カテゴリ

高度

### オブジェクトの説明

このオブジェクトは、SMTP クライアントに関連した関数をグループ化します。  
下記も参照してください。

- 22 ページの " ファイルシステム "
- 32 ページの " 電子メールクライアント "
- 92 ページの " インスタンスのアトリビュート (インスタンス #13、SMTP サーバ) "
- 93 ページの " インスタンスのアトリビュート (インスタンス #14、SMTP ユーザ) "
- 93 ページの " インスタンスのアトリビュート (インスタンス #15、SMTP パスワード) "

### サポートされるコマンド

オブジェクト :           Get\_Attribute  
                          Create  
                          Delete  
                          Send email from file  
                          ( 118 ページの " コマンドの詳細 : Send Email From File")

インスタンス :           Get\_Attribute  
                          Set\_Attribute  
                          Send email ( 119 ページの " コマンドの詳細 : Send Email")

### オブジェクトのアトリビュート (インスタンス #0)

#	名前	アクセス	データタイプ	値
1	Name	Get	CHAR の配列	'SMTP Client'
2	Revision	Get	UINT8	01h
3	Number of instances	Get	UINT16	-
4	Highest instance no.	Get	UINT16	-
11	Max. no. of instances	Get	UINT16	0006h
12	Success count	Get	UINT16	正常に送信されたメッセージの数を反映
13	Error count	Get	UINT16	送信できなかったメッセージの数を反映

## インスタンスのアトリビュート

### 高度

インスタンスはアプリケーションによって動的に作成されます。

#	名前	アクセス	タイプ	説明
1	From	Get/Set	CHAR の配列	例 "someone@somewhere.com"
2	To	Get/Set	CHAR の配列	例 "someone.else@anywhere.net"
3	Subject	Get/Set	CHAR の配列	例 "Important notice"
4	Message	Get/Set	CHAR の配列	例 "Duck and cover"

## コマンドの詳細 : Create

### カテゴリ

高度

### 詳細

コマンドコード : 03h

有効な対象 : オブジェクト

### 説明

このコマンドは、電子メールインスタンスを作成します。

- コマンドの詳細

フィールド	内容	コメント
CmdExt[0]	(予約、ゼロに設定)	-
CmdExt[1]		

- レスポンスの詳細

フィールド	内容	コメント
CmdExt[0]	(予約、無視する)	-
CmdExt[1]		
MsgData[0]	インスタンス番号	ローバイト
MsgData[1]		ハイバイト

## コマンドの詳細 : Delete

### カテゴリ

高度

### 詳細

コマンドコード : 04h

有効な対象 : オブジェクト

### 説明

このコマンドは、電子メールインスタンスを削除します。

- コマンドの詳細

フィールド	内容	コメント
CmdExt[0]	(予約、無視する)	-
CmdExt[1]		

- レスポンスの詳細

(データなし)

## コマンドの詳細 : Send Email From File

### カテゴリ

高度

### 詳細

コマンドコード : 11h

有効な対象 : オブジェクト

### 説明

このコマンドは、ファイルシステム内のファイルに基づいて電子メールを送信します。

ファイルフォーマット :

ファイルは、以下のフォーマットによるプレーンな ASCII ファイルである必要があります。

```
[To]
recipient

[From]
sender

[Subject]
email subject

[Headers]
extra headers, optional

[Message]
actual email message
```

- コマンドの詳細

フィールド	内容	コメント
CmdExt[0]	(予約、ゼロに設定)	-
CmdExt[1]		
MsgData[0... n]	パス + メッセージファイルのファイル名	-

- レスポンスの詳細

(データなし)

## コマンドの詳細 : Send Email

### カテゴリ

高度

### 詳細

コマンドコード : 10h

有効な対象 : インスタンス

### 説明

このコマンドは、指定された電子メールインスタンスを送信します。

- コマンドの詳細  
(データなし)
- レスポンスの詳細  
(データなし)

## オブジェクト特有のエラーコード

エラーコード	意味
1	SMTP サーバが見つからない
2	SMTP サーバの準備が整っていない
3	認証エラー
4	SMTP ソケットエラー
5	SSI スキャンエラー
6	電子メールファイルを変換できない
255	指定されていない SMTP エラー
(その他)	(予約)

## 9.8 ファイルシステム・インターフェースオブジェクト (0Ah)

### カテゴリ

高度

### オブジェクトの説明

このオブジェクトは、組み込みのファイルシステムにインターフェースを提供します。各インスタンスはファイルストリームへのハンドルを表し、ファイルシステム操作のサービスが含まれています。

### サポートされるコマンド

オブジェクト :	Get_Attribute Create ( 122 ページの " コマンドの詳細 : Create") Delete ( 123 ページの " コマンドの詳細 : Delete") Format Disc ( 132 ページの " コマンドの詳細 : Format Disc")
インスタンス :	Get_Attribute File Open ( 123 ページの " コマンドの詳細 : File Open") File Close ( 124 ページの " コマンドの詳細 : File Close") File Delete ( 124 ページの " コマンドの詳細 : File Delete") File Copy ( 125 ページの " コマンドの詳細 : File Copy") File Rename ( 126 ページの " コマンドの詳細 : File Rename") File Read ( 127 ページの " コマンドの詳細 : File Read") File Write ( 128 ページの " コマンドの詳細 : File Write") Directory Open ( 128 ページの " コマンドの詳細 : Directory Open") Directory Close ( 129 ページの " コマンドの詳細 : Directory Close") Directory Delete ( 129 ページの " コマンドの詳細 : Directory Delete") Directory Read ( 130 ページの " コマンドの詳細 : Directory Read") Directory Create ( 131 ページの " コマンドの詳細 : Directory Create") Directory Change ( 131 ページの " コマンドの詳細 : Directory Change")



## オブジェクトの属性 (インスタンス #0)

#	名前	アクセス	データタイプ	値
1	Name	Get	CHAR の配列	'File System Interface'
2	Revision	Get	UINT8	01h
3	Number of instances	Get	UINT16	-
4	Highest instance no.	Get	UINT16	-
11	Max. no. of instances	Get	UINT16	0004h
12	Disable virtual file system	Get	BOOL	False
13	Total disc size	Get	UINT32 の配列	-
14	Free space	Get	UINT32 の配列	-
15	Disc CRC	Get	UINT32 の配列	-

## インスタンスの属性

### 高度

#	名前	アクセス	タイプ	説明
1	Instance type	Get	UINT8	値:   タイプ: 00h  予約 01h  ファイルインスタンス 02h  ディレクトリインスタンス
2	File size	Get	UINT32	バイト単位でのファイルパス (ディレクトリにはゼロ)
3	Path	Get	CHAR の配列	インスタンスが動作するパス

## コマンドの詳細 : Create

### カテゴリ

高度

### 詳細

コマンドコード : 03h

有効な対象 : オブジェクト

### 説明

このコマンドは、ファイル操作のインスタンスを作成します。

- コマンドの詳細

フィールド	内容	コメント
CmdExt[0]	(予約、ゼロに設定)	-
CmdExt[1]		

- レスポンスの詳細

フィールド	内容	コメント
CmdExt[0]	(予約、無視する)	-
CmdExt[1]		
MsgData[0]	インスタンス番号	ローバイト
MsgData[1]		ハイバイト

## コマンドの詳細 : Delete

### カテゴリ

高度

### 詳細

コマンドコード : 04h

有効な対象 : オブジェクト

### 説明

このコマンドは、ファイル操作のインスタンスを削除します。

- コマンドの詳細

フィールド	内容	コメント
CmdExt[0]	(予約、無視する)	-
CmdExt[1]		

- レスポンスの詳細

(データなし)

## コマンドの詳細 : File Open

### カテゴリ

高度

### 詳細

コマンドコード : 10h

有効な対象 : インスタンス

### 説明

このコマンドは、リード、ライト、または追加のためにファイルを開きます。

- コマンドの詳細

フィールド	内容	コメント
CmdExt[0]	モード	値 : モード : 00h リードモード 01h ライトモード 02h 追加モード
CmdExt[1]	(予約、ゼロに設定)	-
MsgData[0... n]	パス + ファイル名	現在のパスに関連する

- レスポンスの詳細

(データなし)

## コマンドの詳細 : File Close

### カテゴリ

高度

### 詳細

コマンドコード : 11h

有効な対象 : インスタンス

### 説明

このコマンドは、直前に開かれたファイルを閉じます。

- コマンドの詳細  
(データなし)
- レスポンスの詳細

フィールド	内容	コメント
CmdExt[0]	(予約、無視する)	-
CmdExt[1]		
MsgData[0]	ファイルサイズ	ローバイト、ローワード
MsgData[1]		-
MsgData[2]		-
MsgData[3]		ハイバイト、ハイワード

## コマンドの詳細 : File Delete

### カテゴリ

高度

### 詳細

コマンドコード : 12h

有効な対象 : インスタンス

### 説明

このコマンドは、指定されたファイルをファイルシステムから永久に削除します。

- コマンドの詳細

フィールド	内容	コメント
CmdExt[0]	(予約、ゼロに設定)	-
CmdExt[1]		
MsgData[0... n]	パス + ファイル名	現在のパスに関連する

- レスポンスの詳細  
(データなし)

## コマンドの詳細 : File Copy

### カテゴリ

高度

### 詳細

コマンドコード : 13h

有効な対象 : インスタンス

### 説明

このコマンドは、ファイルのコピーを作成します。

- コマンドの詳細

フィールド	内容	コメント
CmdExt[0]	(予約、ゼロに設定)	-
CmdExt[1]		
MsgData[0... n]	ソースパス + ファイル名	現在のパスに関連する、ヌルによって区切られる
	ヌル	
	宛先パス + ファイル名	

- レスポンスの詳細

(データなし)

## コマンドの詳細 : File Rename

### カテゴリ

高度

### 詳細

コマンドコード : 14h

有効な対象 : インスタンス

### 説明

このコマンドは、ファイルの名前を変更するかファイルを移動します。

- コマンドの詳細

フィールド	内容	コメント
CmdExt[0]	(予約、ゼロに設定)	-
CmdExt[1]		
MsgData[0... n]	古いパス + ファイル名	現在のパスに関連する、ヌルによって区切られる
	ヌル	
	新しいパス + ファイル名	

- レスポンスの詳細

(データなし)

## コマンドの詳細 : File Read

### カテゴリ

高度

### 詳細

コマンドコード : 15h

有効な対象 : インスタンス

### 説明

リードを目的として直前に開かれたファイルからデータをリードします。

- コマンドの詳細

フィールド	内容	コメント
CmdExt[0]	バイト	リードするバイト数
CmdExt[1]	(予約、ゼロに設定)	-

- レスポンスの詳細

フィールド	内容	コメント
CmdExt[0]	(予約、無視する)	-
CmdExt[1]		
MsgData[0... n]	データ	ファイルからリードされたデータ

## コマンドの詳細 : File Write

### カテゴリ

高度

### 詳細

コマンドコード : 16h

有効な対象 : インスタンス

### 説明

ライトまたは追加を目的として直前に開かれたファイルヘデータをライトします。

- コマンドの詳細

フィールド	内容	コメント
CmdExt[0]	(予約、ゼロに設定)	-
CmdExt[1]		
Data[0... n]	データ	ファイルにライトするデータ

- レスポンスの詳細

フィールド	内容	コメント
CmdExt[0]	バイト	ライトされたバイト数
CmdExt[1]	(予約、無視する)	-

## コマンドの詳細 : Directory Open

### カテゴリ

高度

### 詳細

コマンドコード : 20h

有効な対象 : インスタンス

### 説明

このコマンドは、ディレクトリを開きます。

- コマンドの詳細

フィールド	内容	コメント
CmdExt[0]	(予約、ゼロに設定)	-
CmdExt[1]		
Data[0... n]	パス + ディレクトリの名前	現在のパスに関連する

- レスポンスの詳細

(データなし)



## コマンドの詳細 : Directory Close

### カテゴリ

高度

### 詳細

コマンドコード : 21h

有効な対象 : インスタンス

### 説明

このコマンドは、直前に開かれたディレクトリを閉じます。

- コマンドの詳細  
(データなし)
- レスポンスの詳細  
(データなし)

## コマンドの詳細 : Directory Delete

### カテゴリ

高度

### 詳細

コマンドコード : 22h

有効な対象 : インスタンス

### 説明

このコマンドは、空のディレクトリをファイルシステムから永久に削除します。

- コマンドの詳細

フィールド	内容	コメント
CmdExt[0]	(予約、ゼロに設定)	-
CmdExt[1]		
MsgData[0... n]	パス + ディレクトリの名前	現在のパスに関連する

- レスポンスの詳細  
(データなし)

## コマンドの詳細 : Directory Read

### カテゴリ

高度

### 詳細

コマンドコード : 23h

有効な対象 : インスタンス

### 説明

このコマンドは、リードを目的として直前に開かれたディレクトリの内容をリードします。

コマンドは単一のディレクトリエントリに関する情報を返しますが、これはディレクトリの完全な内容を取得するためにコマンドを複数回発行する必要があることを意味します。最後のエントリがリードされると、コマンドは "空の" レスポンスを返します。このレスポンスのデータサイズはゼロです。

- コマンドの詳細  
(データなし)
- レスポンスの詳細

フィールド	内容	コメント
CmdExt[0]	(予約、無視する)	-
CmdExt[1]		
MsgData[0]	エントリのサイズ	ローバイト、ローワード
MsgData[1]		-
MsgData[2]		-
MsgData[3]		ハイバイト、ハイワード
MsgData[4]	フラグ	<b>ビット :</b> <b>意味 :</b> 0 エントリはディレクトリ 1 エントリはリードオンリー 2 エントリは非表示 3 エントリはシステムエントリ
MsgData[5... n]	エントリの名前	-

## コマンドの詳細 : Directory Create

### カテゴリ

高度

### 詳細

コマンドコード : 24h

有効な対象 : インスタンス

### 説明

このコマンドは、ディレクトリを作成します。

- コマンドの詳細

フィールド	内容	コメント
CmdExt[0]	(予約、ゼロに設定)	-
CmdExt[1]		
MsgData[0... n]	パス + ディレクトリの名前	現在のパスに関連する

- レスポンスの詳細

(データなし)

## コマンドの詳細 : Directory Change

### カテゴリ

高度

### 詳細

コマンドコード : 25h

有効な対象 : インスタンス

### 説明

このコマンドは、インスタンスの現在のディレクトリ / パスを変更します。

- コマンドの詳細

フィールド	内容	コメント
CmdExt[0]	(予約、ゼロに設定)	-
CmdExt[1]		
MsgData[0... n]	パス + ディレクトリの名前	現在のパスに関連する

- レスポンスの詳細

(データなし)

## コマンドの詳細 : Format Disc

### カテゴリ

高度

### 詳細

コマンドコード : 30h

有効な対象 : オブジェクト

### 説明

このコマンドは、ファイルシステムをフォーマットします。

- コマンドの詳細

フィールド	内容	コメント
CmdExt[0]	(予約、ゼロに設定)	-
CmdExt[1]		

- レスポンスの詳細

(データなし)

## オブジェクト特有のエラーコード

エラーコード	意味
1	ファイルを開くのに失敗した
2	ファイルを閉じるのに失敗した
3	ファイルを削除するのに失敗した
4	ディレクトリを開くのに失敗した
5	ディレクトリを閉じるのに失敗した
6	ディレクトリを作成するのに失敗した
7	ディレクトリを削除するのに失敗した
8	ディレクトリを変更するのに失敗した
9	コピー操作の失敗 (ソースを開くことができなかった)
10	コピー操作の失敗 (宛先を開くことができなかった)
11	コピー操作の失敗 (ライトに失敗した)
12	ファイルの名前を変更できない

## 9.9 ネットワークイーサネットオブジェクト (0Ch)

### カテゴリ

拡張

### オブジェクトの説明

このオブジェクトは、イーサネット特有の情報をアプリケーションに提供します。

### サポートされるコマンド

オブジェクト : Get\_Attribute

インスタンス : Get\_Attribute

### オブジェクトの属性 (インスタンス #0)

#	名前	アクセス	データタイプ	値
1	Name	Get	CHAR の配列	'Network Ethernet'
2	Revision	Get	UINT8	01h
3	Number of instances	Get	UINT16	-
4	Highest instance no.	Get	UINT16	-

### インスタンスの属性 (インスタンス #1)

#### 拡張

#	名前	アクセス	タイプ	説明
1	MAC Address	Get	UINT8 の配列	現在の MAC アドレス。 147 ページの "Ethernet ホストオブジェクト (F9h) " も参照してください。

## 9.10 CIP ポートコンフィグレーションオブジェクト (0Dh)

### カテゴリ

高度

### オブジェクトの説明

このオブジェクトは、ネットワーク側で CIP ポートオブジェクト (72 ページの "ポートオブジェクト (F4h) " を参照) にデータ入力し、列挙するために使用されます。基本的に、これはホストアプリケーション内部で CIP ポートを表すことになる、インスタンスとアトリビュートの作成とアップデートに関するものです。このプロセスは、非コネクション型 CIP ルーティングが有効になっている場合に必要です (137 ページの "EtherNet/IP ホストオブジェクト (F8h) "、インスタンスアトリビュート #17 を参照)。

このオブジェクト内部の各インスタンスは、CIP ポートオブジェクト内のインスタンスに対応します。オブジェクトは最大 8 つのインスタンスをサポートしますが、この場合、インスタンス #1 はローカルの TCP ポート専用であるため、ホストアプリケーションは最大 7 つの追加ポートを実装することができます。インスタンス #1 はデフォルト値を使用して自動的にデータ入力されますが、ホストアプリケーションはインスタンスアトリビュート #2 と #4 をカスタマイズすることができます。

アトリビュート #7 を除き、セットアップ中のみインスタンスアトリビュートにライトできます。ホストアプリケーションは、ホストアプリケーション内部に配置された全てのポートのインスタンスアトリビュート #7 を常に最新の状態に維持します。

下記も参照してください。

- 72 ページの "ポートオブジェクト (F4h) " (CIP)
- 137 ページの "EtherNet/IP ホストオブジェクト (F8h) " (インスタンスアトリビュート #7)

---

**重要：**モジュールは、エラー制御に対するホストアプリケーションの責任を引き継がないことに注意してください。モジュールは、ホストアプリケーションによって設定されたデータが正しいことを証明するものではありません。

### サポートされるコマンド

オブジェクト：	Get_Attribute Create Delete
インスタンス：	Get_Attribute Set_Attribute

## オブジェクトの属性 (インスタンス #0)

#	名前	アクセス	データタイプ	値
1	Name	Get	CHAR の配列	'CIP Port Configuration'
2	Revision	Get	UINT8	01h
3	Number of instances	Get	UINT16	-
4	Highest instance no.	Get	UINT16	-
11	Max. no. of instances	Get	UINT16	0008h

## インスタンスの属性

### 高度

#	名前	アクセス	タイプ	説明
1	Port Type	Set	UINT16	ポートの列挙 (CIP 仕様を参照)
2	Port Number	Set	UINT16	このポートに関連付けられた CIP ポート番号
3	Link Path	Set	UINT8 の配列	このポートのオブジェクトを識別する論理パスセグメント
4	Port Name	Set	CHAR の配列	ポートに名前を付ける文字列 (最大文字数は 64)
5	-	-	-	(予約)
6	-	-	-	(予約)
7	Node Address	Set	UINT8 の配列	ポート上のこのデバイスのノード番号。データタイプはポートセグメントに対して範囲を制限します。エンコードされたポート番号は、アトリビュート #2 で指定された値に一致する必要があります。 ポート上にノード番号が設定されていないデバイスは、ポートセグメント内部でゼロの長さのノードアドレスを指定できます (10h 00h など)。 ランタイム中にノードアドレスが変更された場合、ホストアプリケーションはこのアトリビュートのアップデートも処理します。
8	Port Node Range	Set	構成 : UINT16 (最小) UINT16 (最大)	ポート上の最小および最大のノード番号。このアトリビュートのサポートは条件付きであり、ノード番号がデータタイプの範囲内でレポートできる場合 (DeviceNet など) は、アトリビュートはサポートされることとなります。そうでない場合 (4 バイトの IP アドレスを使用する EtherNet/IP などのネットワークの場合など)、アトリビュートはサポートされないこととなります。

下記も参照してください。

- 72 ページの "ポートオブジェクト (F4h)" (73 ページの "インスタンスのアトリビュート (インスタンス #1)")
- 72 ページの "ポートオブジェクト (F4h)" (73 ページの "インスタンスのアトリビュート (インスタンス #2...#8)")

## 10. ホストアプリケーションオブジェクト

### 10.1 概要

この章では、モジュール内のホストアプリケーションオブジェクトの実装を指定します。ここにリストで記載されたオブジェクトは、EtherNet/IP 実装を拡張するために、オプションとしてホストアプリケーションファームウェア内部に実装することができます。

標準オブジェクト：

- アプリケーションオブジェクト ("Anybus-CompactCom Software Design Guide" を参照)
- アプリケーションデータオブジェクト ("Anybus-CompactCom Software Design Guide" を参照)

ネットワーク特有のオブジェクト：

- 137 ページの "EtherNet/IP ホストオブジェクト (F8h) "
- 147 ページの "Ethernet ホストオブジェクト (F9h) "



## 10.2 EtherNet/IP ホストオブジェクト (F8h)

### カテゴリ

基本、拡張、高度

### オブジェクトの説明

このオブジェクトは、ホストアプリケーション内で EtherNet/IP 特有の機能を実装します。このオブジェクトを Ethernet ホストオブジェクトと混同しないように注意してください。147 ページの "Ethernet ホストオブジェクト (F9h) " を参照してください。

このオブジェクトの実装はオプションです。ホストアプリケーションについては、以下に指定されたアトリビュートのどれもサポートしない、一部をサポートする、あるいは全てをサポートすることが可能です。モジュールは、起動中にこれらのアトリビュートの値を取得しようとします。アトリビュートがホストアプリケーション内に実装されていない場合、エラーメッセージ (06h, "Invalid CmdExt[0]") が含まれたレスポンスが返されるだけです。このような場合、モジュールはそのデフォルト値を使用します。

モジュールが以下のリストに記載されていない値を取得しようとすると、エラーメッセージ (06h, "Invalid CmdExt[0]") が含まれたレスポンスが返されます。

このオブジェクトにアクセスする際に使用されるコマンドの一部には、セグメント化が必要であることに注意してください。詳細については、153 ページの "メッセージのセグメント化" を参照してください。

モジュールが EIP QuickConnect 機能を使用するように設定されている場合、EDS ファイルを変更する必要があります。EDS ファイルが変更されるため、モジュールの識別を変更する必要があります。モジュールには認定が必要です。14 ページの "コンフォーマンステストガイド" を参照してください。

下記も参照してください。

- 54 ページの "Identity オブジェクト (01h) " (CIP)
- 58 ページの "アセンブリオブジェクト (04h) " (CIP)
- 72 ページの "ポートオブジェクト (F4h) " (CIP)
- 134 ページの "CIP ポートコンフィグレーションオブジェクト (0Dh) " (Anybus モジュールオブジェクト)
- "Anybus CompactCom Software Design Guide"、"Error Codes"

### サポートされるコマンド

オブジェクト : Process\_CIP\_Object\_Request  
( 142 ページの " コマンドの詳細 : Process\_CIP\_Object\_Request" を参照)

Set\_Configuration\_Data  
( 143 ページの " コマンドの詳細 : Set\_Configuration\_Data" を参照)

Process\_CIP\_Routing\_Request  
( 144 ページの " コマンドの詳細 : Process\_CIP\_Routing\_Request" を参照)

Get\_Configuration\_Data  
( 146 ページの " コマンドの詳細 : Get\_Configuration\_Data" を参照)

インスタンス :-

## オブジェクトの属性 (インスタンス #0)

#	名前	アクセス	データタイプ	値
1	Name	Get	CHAR の配列	'EtherNet/IP'
2	Revision	Get	UINT8	01h
3	Number of instances	Get	UINT16	0001h
4	Highest instance no.	Get	UINT16	0001h

## インスタンスの属性 (インスタンス #1)

### 基本

#	名前	アクセス	タイプ	デフォルト値	コメント
1	Vendor ID	Get	UINT16	005Ah	これらの値は Identity オブジェクト (CIP) に転送されます。
2	Device Type	Get	UINT16	0000h	
3	Product Code	Get	UINT16	002Eh	
4	Revision	Get	構成 : UINT8 メジャー UINT8 マイナー	(ソフトウェアリビジョン)	下記も参照してください。 - 18 ページの " デバイスカスタマイズ " - 54 ページの "Identity オブジェクト (01h) " (CIP オブジェクト)
5	Serial Number	Get	UINT32	(製造時に設定)	<b>注意:</b> これらの属性のいずれかを変更すると、新しい Vendor ID が必要になります。
6	Product Name	Get	CHAR の配列	'Anybus-CC EtherNet/IP (2-Port)'	

### 拡張

#	名前	アクセス	タイプ	デフォルト値	コメント
7	Producing Instance No.	Get	UINT16	0064h	下記も参照してください。 - 60 ページの " インスタンス 64h 属性 (Producing インスタンス) " (CIP インスタンス)
8	Consuming Instance No.	Get	UINT16	0096h	下記も参照してください。 - 60 ページの " インスタンス 96h 属性 (Consuming インスタンス) " (CIP インスタンス)
9	Enable communication settings from Net	Get	BOOL	True	<b>値:</b> <b>意味:</b> True    ネットワークから設定可能 False   ネットワークから設定不可能  下記も参照してください。 - 74 ページの "TCP/IP インターフェースオブジェクト (F5h) " (CIP オブジェクト) - 78 ページの "Ethernet リンクオブジェクト (F6h) " (CIP オブジェクト) - 87 ページの " ネットワークコンフィギュレーションオブジェクト (04h) " (Anybus モジュールオブジェクト)

#	名前	アクセス	タイプ	デフォルト値	コメント
12	Enable Parameter Object	Get	BOOL	True	値: 意味: True CIP パラメータオブジェクトの有効化 False CIP パラメータオブジェクトの無効化
13	Input-Only heartbeat instance number	Get	UINT16	0003h	下記も参照してください。 - 58 ページの "インスタンス 03h アトリビュート (ハートビート、Input-Only) " (CIP インスタンス)
14	Listen-Only heartbeat instance number	Get	UINT16	0004h	下記も参照してください。 - 59 ページの "インスタンス 04h アトリビュート (ハートビート、Listen-Only) " (CIP インスタンス)
15	Assembly object Configuration instance number	Get	UINT16	0005h	下記も参照してください。 - 59 ページの "インスタンス 05h アトリビュート (コンフィグレーションデータ) " (CIP インスタンス)
16	Disable Strict IO Match	Get	BOOL	False	True の場合、モジュールは設定された IO サイズ以下のサイズの Class 1 コネクション要求を受け入れます。
18	Input-Only extended heartbeat instance number	Get	UINT16	0006h	下記も参照してください。 - 59 ページの "インスタンス 06h アトリビュート (ハートビート、拡張 Input-Only) " (CIP インスタンス)
19	Listen-Only extended heartbeat instance number	Get	UINT16	0007h	下記も参照してください。 - 60 ページの "インスタンス 07h アトリビュート (ハートビート、拡張 Listen-Only) " (CIP インスタンス)
20	Interface label port 1	Get	CHAR の配列	Port 1	このアトリビュートの値は、イーサネットリンクオブジェクトのインスタンス #1 のインターフェースラベルを変更するために使用されます。
21	Interface label port 2	Get	CHAR の配列	Port 2	このアトリビュートの値は、イーサネットリンクオブジェクトのインスタンス #2 のインターフェースラベルを変更するために使用されます。
22	Interface label internal port	Get	CHAR の配列	Internal	このアトリビュートの値は、イーサネットリンクオブジェクトのインスタンス #3 のインターフェースラベルを変更するために使用されます。
26	Enable EtherNet/IP QuickConnect a	Get	BOOL	False	値: 意味: True EtherNet/IP QuickConnect 機能が有効 False EtherNet/IP QuickConnect 機能が無効

- a. モジュールが EIP QuickConnect 機能を使用するように設定されている場合、EDS ファイルを変更する必要があります。EDS ファイルが変更されるため、モジュールの識別を変更する必要があります。14 ページの "パフォーマンステストガイド" を参照してください。

## 高度

#	名前	アクセス	タイプ	デフォルト値	コメント
11	Enable CIP forwarding	Get	BOOL	False	<p>値： 意味：</p> <p>True CIP フォワーディングの有効化</p> <p>False CIP フォワーディングの無効化</p> <p>下記も参照してください。 - 142 ページの " コマンドの詳細 : Process_CIP_Object_Request"</p>
17	Enable unconnected routing	Get	BOOL	False	<p>True の場合、モジュールは非コネクション型 CIP ルーティングを有効にします。これは CIP ポートマッピングオブジェクトのコンテンツの初期アップロードもトリガします。</p>
27	Producing Instance Map	Get	UINT の配列		<p>生成するアセンブリインスタンスおよびそのサイズのリストです。このリストは、最大 6 つのエントリを保持できます。このアトリビュートを使用すると、このアトリビュートによって定義されるように、プロセスデータ領域が各領域に分割されます。アプリケーションは、プロセスデータマップのマッピングと維持の責任を負います。</p> <p>このアトリビュートは、アトリビュート #7 が実装される場合、無視されます。</p>
28	Consuming Instance Map	Get	UINT の配列		<p>消費するアセンブリインスタンスおよびそのサイズのリストです。このリストは、最大 6 つのエントリを保持できます。このアトリビュートを使用すると、このアトリビュートによって定義されるように、プロセスデータ領域が各領域に分割されます。アプリケーションは、プロセスデータマップのマッピングと維持の責任を負います。</p> <p>このアトリビュートは、アトリビュート #8 が実装される場合、無視されます。</p>

#	名前	アクセス	タイプ	デフォルト値	コメント
29	Ignore Sequence Count Check	Get	BOOL	False	<p>このアトリビュートを "True" に設定すると、モジュールは消費された Class 1 データの Sequence Count Check を無視するようになります。</p> <p>このことは、変更されたデータや新しいデータだけではなく、発信元から受信された全てのデータがアプリケーションにコピーされることを意味します。</p> <p>変更されたデータだけではなく、全てのデータをコピーすることは、CIP 仕様に違反します。これはモジュールの性能にも影響を与えます。</p> <p>このフラグを "True" に設定する場合は、慎重に行ってください。</p> <p>全てのデータをコピーする場合、HMS は性能測定を行わず、どのように性能に影響が及ぶのかについて一切保証いたしません。</p>

## コマンドの詳細 : Process\_CIP\_Object\_Request

### カテゴリ

高度

### 詳細

コマンドコード : 10h

有効な対象 : オブジェクトインスタンス

### 説明

'Enable CIP Request Forwarding' アトリビュート (#11) を設定することで、実装されていない CIP オブジェクトおよび不明なアセンブリオブジェクトインスタンスへの全ての要求はこのコマンドを通してホストアプリケーションに転送されます。次に、アプリケーションは要求を評価して、適切なレスポンスを返す必要があります。モジュールは、最大 6 つの保留中の CIP 要求をサポートします。追加の要求はモジュールによって却下されます。

ホストインターフェース上のテレグラムの長さは制限されているため、要求データサイズは 255 バイトを超えないようにします。これを超える場合、モジュールは要求の発信元に 'resource unavailable' を送信し、メッセージはホストアプリケーションに転送されません。

**注意 :** このコマンドは、Anybus-CompactCom DeviceNet 内の 'Process\_CIP\_Request' コマンドに類似していますが、同一ではありません。

#### • コマンドの詳細

フィールド	内容	注記
CmdExt[0]	CIP サービスコード	元の CIP 要求からの CIP サービスコード
CmdExt[1]	要求パスサイズ	'Request Path' フィールド内の 16 ビット文字の数
MsgData[0... m]	要求パス	CIP Padded EPATH (クラス、インスタンス、アトリビュートなど)
MsgData[m... n]	要求データ	サービス特有のデータ

#### • レスポンスの詳細

フィールド	内容	注記
CmdExt[0]	CIP サービスコード	(応答ビットセット)
CmdExt[1]	00h	(予約、ゼロに設定)
MsgData[0]	一般ステータス	CIP 一般ステータスコード
MsgData[1]	追加ステータスのサイズ	追加ステータスの配列内の 16 ビット文字の数
MsgData[2... m]	追加ステータス	該当する場合、追加ステータス
MsgData[m... n]	レスポンスデータ	該当する場合、実際のレスポンスデータ

**重要 :** この機能を使用する場合、ホストアプリケーションファームウェア内で全てのオブジェクトに汎用の CIP クラスアトリビュート (アトリビュート #1, 'Revision') を実装していることを確認してください。これを行わないと、モジュールはコンFORMANCE テストに合格できません。

## コマンドの詳細 : Set\_Configuration\_Data

### カテゴリ

高度

### 詳細

コマンドコード : 11h

有効な対象 : オブジェクトインスタンス

### 説明

CIP 'Forward\_Open' サービス内のデータセグメントにコンフィグレーションデータが含まれている場合、これはこのコマンドを通してホストアプリケーションに転送されます。実装されている場合、ホストアプリケーションは要求を評価して、適切なレスポンスを返す必要があります。

ホストインターフェース上のテレグラムの長さは制限されているため、255 バイトを超えるデータサイズにはセグメント化が必要です。モジュールによって受け入れられるコンフィグレーションデータの最大合計量は 458 バイトです。

**注意 :** コンフィグレーションデータをサポートするには、このコマンドを実装する必要があります。実装されない場合、CIP'Forward\_Open' 要求はモジュールによって却下されます。

- コマンドの詳細

フィールド	内容	注記
CmdExt[0]	-	(予約、無視する)
CmdExt[1]	セグメント化制御ビット	153 ページの " メッセージのセグメント化 " を参照
MsgData[0... n]	データ	実際のコンフィグレーションデータ

- レスポンスの詳細 (成功)

フィールド	内容	注記
CmdExt[0]	00h	(予約、ゼロに設定)
CmdExt[1]	00h	(予約、ゼロに設定)
MsgData[0]	ステータス	00h : 許容

- レスポンスの詳細 (エラー)

フィールド	内容	注記
CmdExt[0]	00h	(予約、ゼロに設定)
CmdExt[1]	00h	(予約、ゼロに設定)
MsgData[0]	ステータス	Anybus プロトコルエラーコード
MsgData[1]	インデックス	MsgData[0] が FFh に等しい場合 (オブジェクト特有エラー)、このフィールドのコンテンツは誤ったアトリビュートをポイントします。

下記も参照してください。

- 61 ページの " コネクションマネージャ (06h) " (CIP)
- 153 ページの " メッセージのセグメント化 "

## コマンドの詳細 : Process\_CIP\_Routing\_Request

### カテゴリ

高度

### 詳細

コマンドコード : 12h

有効な対象 : オブジェクトインスタンス

### 説明

モジュールは、Unconnected\_Send サービス内部で最初のパスを削除し、ルーティングを使用して継続することが可能かどうかを評価します (要求されたポートがポートオブジェクト内部に存在することを確認するなど)。削除されたパスがコンテンツによってアプリケーションに送信された最後のパスである場合、アプリケーションは宛先ノードに送信された CIP 要求になります。そうでない場合、アップデートされたルートパス情報が含まれた Unconnected\_Send サービスになります。

モジュールは、最大 6 つの保留中の要求をサポートします。追加の要求はモジュールによって却下されます。

**注意 :** ホストインターフェース上のテレグラムの長さは制限されているため、データは 255 バイトの長さを超えないようにします。これを超える場合、モジュールは要求の発信元を却下し ('Resource unavailable')、このコマンドはホストアプリケーションに対して発行されません。

- コマンドの詳細

フィールド	内容	注記
CmdExt[0]	-	(予約、無視する)
CmdExt[1]	-	(予約、無視する)
MsgData[0... n]	宛先パス	EPATH としてエンコードされた宛先パス
MsgData[n+1]	Time_tick	タイムアウトパラメータがアップデートされた後に有効
MsgData[n+2]	Time-out_ticks	タイムアウトパラメータがアップデートされた後に有効
MsgData[n+3... m]	CIP メッセージ	ルーティングするための CIP メッセージ

- レスポンスの詳細

フィールド	内容	注記
CmdExt[0]	00h	(予約、ゼロに設定)
CmdExt[1]	00h	(予約、ゼロに設定)
MsgData[0]	CIP サービス	実際の CIP サービスコード、レスポンスビットセット
MsgData[1]	00h	(予約、ゼロに設定)
MsgData[2]	一般ステータス	実際の CIP 一般ステータスコード
MsgData[3]	追加ステータスのサイズ	追加ステータスの配列内の 16 ビット文字の数
MsgData[4... n]	追加ステータスの配列	該当する場合、追加ステータス
MsgData[n+1... m]	レスポンスデータ	実際のレスポンスデータ



下記も参照してください。

- 72 ページの "ポートオブジェクト (F4h) " (CIP)
- 134 ページの "CIP ポートコンフィグレーションオブジェクト (0Dh) "

## コマンドの詳細 : Get\_Configuration\_Data

### 詳細

コマンドコード : 13h

有効な対象 : オブジェクトインスタンス

### 説明

コンフィグレーションデータがネットワークから要求されると、Anybus はアプリケーションにこのコマンドを発行します。アプリケーションは、レスポンスメッセージ内に保存されたコンフィグレーションデータを送信することになります。

ホストインターフェース上のテレグラムの長さは制限されているため、255 バイトを超えるデータサイズにはセグメント化が必要です。モジュールによって受け入れられるコンフィグレーションデータの最大合計量は 458 バイトです。

**注意 :** コンフィグレーションデータをサポートするには、このコマンドを実装する必要があります。実装されない場合、要求は Anybus モジュールによって却下されます。

- コマンドの詳細

フィールド	内容	注記
CmdExt[0]	00h	-
CmdExt[1]	00h	-
MsgData[0... n]	-	拡張メッセージデータなし

- レスポンスの詳細 (成功)

フィールド	内容	注記
CmdExt[0]	00h	(予約、ゼロに設定)
CmdExt[1]	セグメント化制御ビット	153 ページの "メッセージのセグメント化" を参照
MsgData[0 - n]	ステータス	アプリケーションからのコンフィグレーションデータ

- レスポンスの詳細 (エラー)

フィールド	内容	注記
CmdExt[0]	00h	(予約、ゼロに設定)
CmdExt[1]	セグメント化制御ビット	153 ページの "メッセージのセグメント化" を参照
MsgData[0]	ステータス	Anybus プロトコルエラーコード

下記も参照してください。

153 ページの "メッセージのセグメント化"

## 10.3 Ethernet ホストオブジェクト (F9h)

### カテゴリ

基本、拡張

### オブジェクトの説明

このオブジェクトは、ホストアプリケーション内にイーサネット機能を実装します。

### サポートされるコマンド

オブジェクト :           Get\_Attribute  
 インスタンス :           Get\_Attribute  
                               Set\_Attribute

### オブジェクトのアトリビュート (インスタンス #0)

#	名前	アクセス	データタイプ	値
1	Name	Get	CHAR の配列	'Ethernet'
2	Revision	Get	UINT8	01h
3	Number of instances	Get	UINT16	0001h
4	Highest instance no.	Get	UINT16	0001h

## インスタンスの属性 (インスタンス #1)

### 基本

#	名前	アクセス	タイプ	デフォルト <sup>a</sup>	コメント
1	MAC Address <sup>b</sup>	Get	UINT8 の配列	-	6 バイトの物理アドレス値。事前にプログラムされた Mac アドレスを上書きします。新しい Mac アドレス値は、IEEE から取得される必要があることに注意してください。

- a. 属性が実装されない場合、モジュールは代わりにこの値を使用します。  
 b. モジュールは、有効な Mac アドレスを使用して事前にプログラムされます。このアドレスを使用するには、この属性を実装しないでください。

### 拡張

#	名前	アクセス	タイプ	デフォルト <sup>a</sup>	コメント
2	Enable HICP	Get	BOOL	True	値: 意味: True HICP が有効 False HICP が無効 (156 ページの "HICP (ホスト IP コンフィグレーションプロトコル)" を参照)
3	Enable Web Server	Get	BOOL	True	値: 意味: True Web サーバが有効 False Web サーバが無効 (25 ページの "Web サーバ" を参照)
5	Enable Web ADI access	Get	BOOL	True	値: 意味: True Web ADI アクセスが有効 False Web ADI アクセスが無効 (25 ページの "Web サーバ" を参照)
6	Enable FTP server	Get	BOOL	True	値: 意味: True FTP サーバが有効 False FTP サーバが無効 (23 ページの "FTP サーバ" を参照)
7	Enable admin mode	Get	BOOL	False	値: 意味: True FTP 管理モードが有効 False FTP 管理モードが無効 (23 ページの "FTP サーバ" を参照)
8	Network Status	Set	UINT16	-	149 ページの "ネットワークステータス" を参照

- a. 属性が実装されない場合、モジュールは代わりにこの値を使用します。

## ネットワークステータス

このアトリビュートは、以下のように全体的なネットワークステータスを示すビットフィールドを保持します。

ビット	内容	説明						
0	リンク	<table><tr><td>値:</td><td>意味:</td></tr><tr><td>True</td><td>リンク検出済み</td></tr><tr><td>False</td><td>リンクなし</td></tr></table>	値:	意味:	True	リンク検出済み	False	リンクなし
値:	意味:							
True	リンク検出済み							
False	リンクなし							
1	IP 確立済み	<table><tr><td>値:</td><td>意味:</td></tr><tr><td>True</td><td>IP アドレス確立済み</td></tr><tr><td>False</td><td>IP アドレス未確立</td></tr></table>	値:	意味:	True	IP アドレス確立済み	False	IP アドレス未確立
値:	意味:							
True	IP アドレス確立済み							
False	IP アドレス未確立							
2...15	(予約)	(マスクオフして無視)						

## A. 機能のカテゴリ化

アトリビュートやサービスなど、Anybus CompactCom とアプリケーションのオブジェクトは、基本、高度、拡張の3つのカテゴリに分類されます。

### A.1 基本

このカテゴリには、実装時または使用時に必要なオブジェクト、アトリビュート、およびサービスが含まれます。これらは Anybus CompactCom を起動して選択したネットワークプロトコルでデータを送受信するには十分な情報です。産業用ネットワークの基本的な機能が使用されます。

製品を認証できるようにする追加オブジェクトなどもこのカテゴリに属しています。

### A.2 拡張

このカテゴリにあるオブジェクトを使用すると、アプリケーションの機能が拡張されます。ネットワークとの間でのデータの基本的な移動だけでなく、産業用ネットワークのより具体的な特徴にアクセスします。アプリケーションには付加価値が与えられます。

### A.3 高度

このグループに属するオブジェクト、アトリビュート、およびサービスは、特殊な使用頻度の低い機能を提供します。使用できるネットワーク機能のほとんどは有効で、アクセス可能です。通常、産業用ネットワークの仕様へのアクセスが必要です。

## B. 実装の詳細

### B.1 拡張 LED 機能

Anybus CompactCom EtherNet/IP 2-Port モジュールでは、モジュールの 2 つのポートでリンク / アクティビティを示すために、GIP1 および GOP1 を使用する可能性があります。

拡張 LED 機能を有効にするには、アプリケーションが Anybus オブジェクトのインスタンス 1 アトリビュート 16 (GPIO コンフィグレーション) を SETUP 状態の間に 0x0001 に設定する必要があります。

"Anybus-CompactCom Hardware Design Guide" の "Host Interface Signals" を参照してください。

#### GPIO モードの説明

		信号	
		GIP1	GOP1
GPIO コンフィグレーション	値 : 0x0000 (デフォルト)	汎用目的の入力	汎用目的の出力
	値 : 0x0001 (拡張 LED 機能)	ポート 1 の 10/100MHz の表示は 1 つの信号に組み合わせられて緑色の LED に接続されますが、これはリンクが存在する場合は点灯し、アクティビティの場合は点滅します。	ポート 2 の 10/100MHz の表示は 1 つの信号に組み合わせられて緑色の LED に接続されますが、これはリンクが存在する場合は点灯し、アクティビティの場合は点滅します。

**注意 1 :** 拡張 LED 機能を有効にすると、GIP[0..1] と GOP[0..1] の両方が出力として機能するようになります。

**注意 2 :** 拡張 LED 機能を有効にすると、GIP[0..1] と GOP[0..1] の両方がアクティブローとして定義されます。これは対応するピンがローの場合に LED が点灯することを意味します。

**注意 3 :** LED の挙動については、第 1 章で説明されています。11 ページの "ネットワークステータス LED" を参照

### B.2 SUP ビットの定義

監視ビット (SUP) は、ネットワーク参加が別のネットワークデバイスによって監視されていることを示します。EtherNet/IP の場合、これは 1 つ以上の CIP (Class 1 または Class 3) の接続がモジュールに対して開かれた場合に SUP ビットが設定されることを意味します。

## B.3 Anybus ステートマシン

以下の表では、Anybus ステートマシンが EtherNet/IP ネットワークにどのように関係するのかを説明します。

Anybus の状態	実装	コメント
WAIT_PROCESS	モジュールは Class 1 コネクションが開かれるまでこの状態のままになります。	-
ERROR	1. Class 1 コネクションエラー 2. 重複 IP アドレスが検出された	-
PROCESS_ACTIVE	エラーのない Class 1 コネクションがアクティブ (Exclusive-Owner コネクションの 32 ビット Run/Idle ヘッダ内に設定された RUN ビット)	消費する接続のみに有効
IDLE	Class 1 コネクションがアイドル	
EXCEPTION	予期されないエラー (ウォッチドッグタイムアウトなど)	MS LED が赤色に点灯 (メジャーフォールトを示すため) NS LED は消灯

## B.4 アプリケーション・ウォッチドッグタイムアウトの処理

アプリケーション・ウォッチドッグタイムアウトを検出すると、モジュールはネットワーク参加を中止し、'EXCEPTION' 状態に切り替わります。他のネットワーク特有のアクションは実行されません。



## C. メッセージのセグメント化

### C.1 概要

カテゴリ：高度

Anybus-CompactCom でサポートされるメッセージの最大サイズは 255 バイトです。ソケットインターフェースを使用している場合に必要なこれよりも長いメッセージのサポートを提供するために、セグメント化プロトコルが使用されます。

セグメント化プロトコルはメッセージレイヤ内に実装されるため、シリアルホストインターフェースで使用されるフラグメンテーションと混同しないようにします。詳細については、一般の "Anybus-CompactCom Software Design Guide" を参照してください。

モジュールは、インスタンスあたり 1 つのセグメント化された一連のメッセージをサポートします。

## C.2 コマンドのセグメント化

コマンドメッセージがセグメント化されると、コマンドイニシエータは複数回同じコマンドヘッダを送信します。各メッセージでは、データフィールドが次のデータセグメントと交換されます。

同じインスタンスにおいて、一部のコマンドを同時に使用できないことに注意してください。そのインスタンスのセグメント化されたバッファにどちらのコマンドもアクセスする必要があるからです。

コマンドセグメント化は、以下のコマンドに使用されます。

- Set\_Configuration\_Data (143 ページの "コマンドの詳細: Set\_Configuration\_Data" を参照)
- Get\_Configuration\_Data (146 ページの "コマンドの詳細: Get\_Configuration\_Data" を参照)
- Send (109 ページの "コマンドの詳細: Send" を参照)
- Send To (110 ページの "コマンドの詳細: Send\_To" を参照)

### セグメント化制御ビット (コマンド)

ビット	内容	意味
0	FS	現在のセグメントが最初のセグメントの場合に設定
1	LS	現在のセグメントが最後のセグメントの場合に設定
2	AB	セグメントが中止される場合に設定
3...7	(予約)	0 に設定 (ゼロ)

### セグメント化制御ビット (レスポンス)

ビット	内容	意味
0...7	(予約)	無視する

セグメントコマンドを発行する場合、以下のルールが適用されます。

- 最初のセグメントを発行する場合、FS を設定する必要があります。
- 続くセグメントを発行する場合、FS および LS の両方をクリアする必要があります。
- 最後のセグメントを発行する場合、LF ビットを設定する必要があります。
- 単一のセグメントコマンド (つまり 255 バイト以下のサイズ) の場合、FS および LS の両方を設定する必要があります。
- 最後のレスポンスメッセージには、操作の実際の結果が含まれます。
- AB が設定されたメッセージを発行することで、コマンドイニシエータはいつでも操作を中止できます。
- セグメント化エラーが送信中に検出された場合、エラーメッセージが返され、現在のセグメント化メッセージは破棄されます。ただし、これは現在のセグメントにのみ適用され、以前に送信されたセグメントは有効なままであることを注意してください。

## C.3 レスポンスのセグメント化

レスポンスがセグメント化されると、コマンドイニシエータは複数回同じコマンドを送信することで次のセグメントを要求します。各レスポンスでは、データフィールドが次のデータセグメントと交換されます。

レスポンスセグメント化は、以下のコマンドへのレスポンスに使用されます。

- Receive (オブジェクト特有、105 ページの " コマンドの詳細 : Receive" を参照)
- Receive From (オブジェクト特有、107 ページの " コマンドの詳細 : Receive\_From" を参照)

### セグメント化制御ビット (コマンド)

ビット	内容	意味
0	(予約)	(ゼロに設定)
1		
2	AB	セグメントが中止される場合に設定
3...7	(予約)	(ゼロに設定)

### セグメント化制御ビット (レスポンス)

ビット	内容	意味
0	FS	現在のセグメントが最初のセグメントの場合に設定
1	LS	現在のセグメントが最後のセグメントの場合に設定
2...7	(予約)	(ゼロに設定)

セグメントレスポンスを受信する場合、以下のルールが適用されます。

- 最初のセグメント内では、FS が設定されます。
- 続く全てのセグメント内では、FS および LS の両方がクリアされます。
- 最後のセグメント内では、LS が設定されます。
- 単一のセグメントレスポンス (つまり 255 バイト以下のサイズ) の場合、FS および LS の両方が設定されます。
- AB が設定されたメッセージを発行することで、コマンドイニシエータはいつでも操作を中止できます。

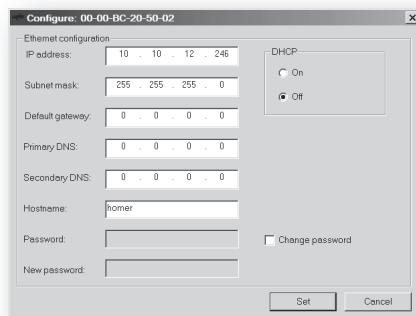
## D. HICP (ホスト IP コンフィグレーションプロトコル)

### D.1 概要

モジュールは、設定 (IP アドレス、サブネットマスク、DHCP の有効化 / 無効化など) を変更するために、Anybus IPconfig ユーティリティによって使用される HICP プロトコルをサポートします。Anybus IPconfig は、HMS の Web サイト (www.anybus.com) から無償でダウンロードできます。このユーティリティを使用すると、UDP ポート 3250 を通してネットワークに接続された任意の Anybus 製品のネットワーク設定にアクセスできます。

### D.2 操作

プログラムの起動時には、Anybus 製品についてネットワークがスキャンされます。ネットワークは、'Scan' をクリックすることでいつでも再スキャンできます。



モジュールのネットワーク設定を変更するには、リスト内でそのエントリをダブルクリックします。モジュールの設定が含まれたウィンドウが表示されます。

'Set' をクリックして新しい設定を確認するか、'Cancel' をクリックして全ての変更をキャンセルします。

オプションとして、コンフィグレーションをパスワードによって不正アクセスから保護することができます。パスワードを入力するには、'Change password' チェックボックスをクリックし、'New password' の下にパスワードを入力します。

## E. 技術仕様

### E.1 保護接地（PE）要件

適切な EMC 挙動を保証するためには、一般の "Anybus-CompactCom Hardware Design Guide" に説明されている PE パッド /PE メカニズムを使用して、モジュールを保護接地に適切に接続する必要があります。

これらの PE 要件が満たされない限り、HMS Industrial Networks は適切な EMC 挙動を保証しません。

### E.2 電源供給

#### 供給電圧

一般の "Anybus-CompactCom Hardware Design Guide" で指定されているように、モジュールには 3.3V の安定化電源が必要です。

#### 電力消費量

Anybus-CompactCom EtherNet/IP は、クラス B モジュールの要件を満たす設計です。Anybus-CompactCom プラットフォームで使用される電力消費量の分類の詳細については、一般の "Anybus-CompactCom Hardware Design Guide" を参照してください。

現在のハードウェア設計は、最大 380mA<sup>1</sup> を消費します。

### E.3 環境仕様

詳細については、"Anybus-CompactCom Hardware Design Guide" を参照してください。

### E.4 EMC 適合性

詳細については、"Anybus-CompactCom Hardware Design Guide" を参照してください。

- 
1. 継続的な製品開発を旨とする HMS のポリシーに則って、弊社は事前に通知することなく、本製品の厳密な電源要件を変更する権利を留保することに注意してください。ただし、どのような場合であっても、Anybus-CompactCom EtherNet/IP はクラス B モジュールのままです。

## F. 著作権表示

本製品には、Carnegie Mellon、Massachusetts Institute of Technology、University of California、および RSA Data Security によって開発されたソフトウェアが含まれます。

\*\*\*\*\*

Copyright 1986 by Carnegie Mellon.

\*\*\*\*\*

Copyright 1983,1984,1985 by the Massachusetts Institute of Technology

\*\*\*\*\*

Copyright (c) 1988 Stephen Deering.

Copyright (c) 1982, 1985, 1986, 1992, 1993

The Regents of the University of California. All rights reserved.

このコードは、Stanford University の Stephen Deering によって Berkeley に提供されたソフトウェアから派生したものです。

ソースおよびバイナリ形式での再配布および使用は、変更の有無に関わらず、以下の条件を満たす場合に許可されます。

- ソースコードの再配布では、上記の著作権表示、本条件一覧、および下記の免責事項を保持する必要があります。
- バイナリ形式での再配布では、配布に付属する文書または他の資料（あるいはその両方）に上記の著作権表示、本条件一覧、および下記の免責事項を転載する必要があります。
- 書面による事前の許可なしに、大学の名前やコントリビュータの名前を本ソフトウェアから派生した製品の宣伝または販売促進に使用することはできません。

本ソフトウェアは評議員およびコントリビュータによって "現状のまま" 提供され、商品性および特定用途に対する適合性の暗黙的保証を含め（ただし、それに限定されることなく）、いかなる明示的または暗黙的保証も放棄されます。いかなる場合においても、評議員またはコントリビュータは、原因のいかんを問わず、かつ責任の根拠が契約、厳格責任、または不法行為（過失などを含む）であるかどうかを問わず、本ソフトウェアの使用によって生じた直接的損害、間接的損害、偶発的損害、特別損害、懲罰的損害、または結果的損害（代替品またはサービスの調達、使用機会、データ、または利益の喪失、あるいは業務の中断を含むが、それらに限定されない）について、仮にそのような損害の可能性が知らされていたとしても責任を負わないものとします。

\*\*\*\*\*

Copyright (C) 1990-2, RSA Data Security, Inc. All rights reserved.

本ソフトウェアまたは本機能について言及または参照している全ての資料で本ソフトウェアが "RSA Data Security, Inc. MD4 Message-Digest Algorithm" として明記されている場合は、本ソフトウェアを複製および使用するライセンスが付与されます。

また、派生物について言及または参照している全ての資料で当該派生物が "RSA Data Security, Inc. MD4 Message-Digest Algorithm から派生したもの" として明記されている場合は、派生物を作成および使用するライセンスが付与されます。

RSA Data Security, Inc. は、本ソフトウェアの商品性と任意の特定用途に対する本ソフトウェアの適合性に関して一切の表明を行いません。本ソフトウェアは、いかなる種類の明示的または暗黙的保証もなく "現状のまま" 提供されます。

これらの通知は、本マニュアルまたはソフトウェア（あるいはその両方）のいかなる部分の複製でも保持されている必要があります。

\*\*\*\*\*

Copyright (C) 1991-2, RSA Data Security, Inc. Created 1991. All rights reserved.

本ソフトウェアまたは本機能について言及または参照している全ての資料で本ソフトウェアが "RSA Data Security, Inc. MD5 Message-Digest Algorithm" として明記されている場合は、本ソフトウェアを複製および使用するライセンスが付与されます。

また、派生物について言及または参照している全ての資料で当該派生物が "RSA Data Security, Inc. MD5 Message-Digest Algorithm から派生したもの" として明記されている場合は、派生物を作成および使用するライセンスが付与されます。

RSA Data Security, Inc. は、本ソフトウェアの商品性と任意の特定用途に対する本ソフトウェアの適合性に関して一切の表明を行いません。本ソフトウェアは、いかなる種類の明示的または暗黙的保証もなく "現状のまま" 提供されます。

これらの通知は、本マニュアルまたはソフトウェア（あるいはその両方）のいかなる部分の複製でも保持されている必要があります。