

Anybus[®] CompactCom PROFINET IO 2-Port

Network Interface Appendix

Doc.Id. HMSI-27-218
Rev. 2.42



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
Web: www.anybus.com

必ずお読みください

本ドキュメントは、PROFINET IO 2-Port の機能を十分理解していただくためのものです。本ドキュメントでは、Anybus CompactCom PROFINET IO 2-Port の機能についてのみ説明しています。Anybus CompactCom に関する一般的な情報については、Anybus CompactCom のデザインガイドを参照してください。

本ドキュメントの読者は、ソフトウェア設計や通信システム全般に関して高い知識を備えていることが求められます。PROFINET IO 2-Port の高度な機能を使用するには、PROFINET IO 2-Port ネットワーキングの内部詳細、および公式の PROFINET IO 2-Port 規格に関する十分な知識が必要です。本製品の使用者は、PROFINET IO 2-Port 規格を入手して十分な知識を得るか、または規格に関する知識を必要としない範囲で本製品を使用してください。

責任の範囲

本マニュアルは細心の注意を払って作成されています。誤字や脱字があった場合は、HMS Industrial Networks AB にお知らせください。本ドキュメントに記載されているデータや図表は、何ら拘束力を持ちません。HMS Industrial Networks AB は、製品開発に継続的に取り組むという自社のポリシーに基づき、製品に変更を加える権利を留保します。本ドキュメントの内容は予告なく変更される場合があります。また、本ドキュメントの内容は、HMS Industrial Networks AB による何らかの保証を表明するものではありません。HMS Industrial Networks AB は、本ドキュメント内の誤りについて一切の責任を負いません。

本製品は様々な用途に応用可能です。本装置の使用者は、必要なあらゆる手段を通じて、本装置の用途が適用される法令、規則、規約、規格の定める性能・安全性に関する要件をすべて満たしていることを検証しなければならないものとします。

HMS Industrial Networks AB は、いかなる場合であっても、本製品のドキュメントに記載されていない機能やタイミング、機能の副作用によって生じた不具合について一切の責任を負いません。本製品のかかる側面を直接または間接に使用したことによって生じる影響（互換性の問題や安定性の問題など）は、本ドキュメントでは定義されていません。

本ドキュメントの例や図表は、説明のみを目的として使用されています。本製品の個々の使用においては様々なバリエーションや要件が存在するため、本ドキュメントの例や図表に基づいて本製品を使用したことに関して、HMS Industrial Networks AB は一切の責任を負いません。

知的所有権

本ドキュメントに記載されている製品に組み込まれた技術に関する知的所有権は HMS Industrial Networks AB に帰属します。この知的所有権には、米国およびその他の国における特許や出願中の特許が含まれます。

商標

Anybus® は、HMS Industrial Networks AB の登録商標です。その他の商標は、各所有者に帰属します。

警告： これはクラス A 製品です。ご家庭でお使いになる場合、電波障害を引き起こす場合があります。その場合は適切な措置をお取りください。

ESD に関する注意事項： 本製品では、ESD（静電気放電）による損傷を受けやすい部品が使用されています。ESD の管理手順に従わない場合、それらの部品が損傷するおそれがあります。本製品を扱う際は、静電気を管理するための予防措置を講じてください。この予防措置を怠った場合、本製品が損傷するおそれがあります。

目次

前書き	本ドキュメントについて	
	関連ドキュメント.....	1
	ドキュメント更新履歴.....	1
	表記と用語.....	2
	サポート.....	2
第 1 章	Anybus-CompactCom PROFINET IO 2-Port について	
	概要.....	3
	機能.....	3
	<i>ABCC-PRT IO</i> モジュールとの互換性.....	3
	前面図.....	4
	ネットワークコネクタ (Brick バージョン).....	6
第 2 章	チュートリアル	
	はじめに.....	7
	フィールドバスの適合性に関するメモ.....	7
	認定.....	7
第 3 章	基本動作	
	概要.....	9
	ソフトウェアの要件.....	9
	電子データシート (GSD).....	10
	ネットワーク ID.....	11
	通信設定.....	12
	ネットワークのデータ交換.....	13
	アプリケーションデータインスタンス (ADI).....	13
	プロセスデータ.....	14
	注意事項.....	14
	診断.....	15
	Web インターフェイス.....	15
	E-mail クライアント.....	15
	ファイルシステム.....	16
	概要.....	16
	システムファイル.....	16
第 4 章	PROFINET IO の実装詳細	
	概要.....	17
	アプリケーションプロセスインスタンス (API).....	18
	アプリケーションリレーションシップ (AR).....	18
	Real Identification (RI).....	19

概要.....	19
設定の不一致.....	20
チャンネル診断とプロセスアラーム.....	20
識別とメンテナンス (I&M).....	21
概要.....	21
I&M のデータ構造.....	22
ファストスタートアップ.....	23
概要.....	23
STEP7 を用いたファストスタートアップの設定.....	24
PROFIenergy プロファイル.....	26
実装.....	26
第 5 章 FTP サーバー	
概要.....	27
ユーザーアカウント.....	27
セッションの例.....	28
第 6 章 Web サーバー	
概要.....	29
デフォルトの Web ページ.....	29
ネットワークの設定.....	30
イーサネットの統計ページ.....	31
サーバーの設定.....	32
概要.....	32
インデックスページ.....	32
デフォルトのコンテンツタイプ.....	33
認証.....	33
第 7 章 E-mail クライアント	
概要.....	34
E-mail メッセージの送信.....	34
第 8 章 Server Side Include (SSI)	
概要.....	35
ファイルのインクルード.....	35
コマンド機能.....	36
概要.....	36
GetConfigItem().....	37
SetConfigItem().....	38
SsiOutput().....	40
DisplayRemoteUser.....	40
ChangeLanguage().....	41
IncludeFile().....	42
SaveDataToFile().....	43
printf().....	44
scanf().....	46
引数関数.....	48

	概要.....	48
	ABCCMessage()	48
	SSI の出力設定.....	52
第 9 章	SNMP エージェント	
	概要.....	53
	管理情報 (MIB)	53
	MIB-II.....	53
	システムグループ変数.....	54
	インターフェイスグループ変数.....	54
第 10 章	Anybus モジュールオブジェクト	
	概要.....	56
	Anybus オブジェクト (01h)	57
	診断オブジェクト (02h).....	58
	ネットワークオブジェクト (03h).....	61
	ネットワークコンフィグレーションオブジェクト (04h).....	62
	ネットワーク PROFINET IO オブジェクト (0Eh)	70
	PROFINET 追加診断オブジェクト (0Fh).....	81
	ソケットインターフェイスオブジェクト (07h).....	86
	SMTP クライアントオブジェクト (09h)	103
	ファイルシステム・インターフェースオブジェクト (0Ah).....	108
	ネットワークイーサネットオブジェクト (0Ch)	121
第 11 章	ホストアプリケーションオブジェクト	
	概要.....	122
	PROFINET IO オブジェクト (F6h)	123
	Ethernet ホストオブジェクト (F9h)	147
	Energy コントロールオブジェクト (F0h).....	150
Appendix A	機能の分類	
	基本.....	155
	拡張.....	155
	高度.....	155
Appendix B	Anybus の実装詳細	
	SUP ビットの定義.....	156
	Anybus のステートマシン	156
	アプリケーションウォッチドッグのタイムアウト処理.....	156
Appendix C	メッセージの分割	
	概要.....	157

コマンドの分割.....	157
応答の分割.....	158
Appendix D フローチャート	
フローチャート - レコードデータへのアクセス	159
フローチャート - 設定の不一致 (RI).....	160
フローチャート - Real Identification (RI) の確立.....	161
Appendix E HICP (ホスト IP 設定プロトコル)	
概要.....	162
操作.....	162
Appendix F 技術仕様	
保護接地 (PE) に関する要件	163
電源.....	163
環境仕様.....	163
EMC への準拠.....	164
Appendix G タイミングと性能	
概要.....	165
プロセスデータ	166
概要.....	166
Anybus におけるリードプロセスデータの遅延時間 (Anybus の遅延時間) ..	166
Anybus におけるライトプロセスデータの遅延時間 (Anybus の遅延時間) ..	166
ネットワークシステムにおけるリードプロセスデータの遅延時間 (ネットワークシステムの遅延時間).....	167
ネットワークシステムにおけるライトプロセスデータの遅延時間 (ネットワークシステムの遅延時間).....	167
Appendix H コンフォーマンステストに関するガイド	
概要.....	168
お客様製品の Identity 情報確定	169
工場出荷時状態へのリセット.....	170
IP アドレス	170
ステーション名	170
Anybus のジェネリックモードにおける認定	171
アドバンスモードにおける認定.....	172
Appendix I 著作権情報	

P. 本ドキュメントについて

より詳しい情報や各種ドキュメントは、HMS の Web サイト www.anybus.com から入手いただけます。

P.1 関連ドキュメント

ドキュメント	作成者
Anybus-CompactCom Software Design Guide	www.anybus.com
Anybus-CompactCom Hardware Design Guide	
Anybus-CompactCom Software Driver User Guide	
Anybus-CompactCom Profinet IO Fieldbus Appendix	
PROFINET IO 規格, rev. 2.2	Profibus International
PROFInergy 技術規格, rev. 1.0	Profibus International

P.2 ドキュメント更新履歴

最近の変更に関する概要 (2.41 ~ 2.42)

変更内容	ページ
Anybus オブジェクトにアトリビュート 16 を追加	57
Anybus のステートマシンテーブルを更新	156
PROFINET ホストオブジェクト, インスタンス #1, アトリビュート #16 のデフォルト値を訂正	124
PROFINET ホストオブジェクトのアトリビュートに関する説明を更新	126

リビジョンリスト

リビジョン	日付	作成者	章	説明
1.00	2009/05/12	KeL	-	初版
1.01	2009/09/17	KeL	E, 3, 10, 1	小規模な訂正
2.00	2010/05/05	KeL	F, 10, 6, 9	コンセプト変更。小規模な訂正
2.10	2010/12/02	KeL	F, 1, 4, 10, 11	エナジープロファイルを追加。小規模な訂正
2.11	2011/01/27	KeL	H, P, 11, 10, B	小規模な訂正
2.20	2011/05/23	KeL	P, H	認定に関する付録を追加。販売 / サポート情報を更新
2.21	2011/05/25	KeL	H	小規模な変更
2.22	2011/06/16	KeL	2	小規模な変更
2.23	2011/08/08	KaD	6, 11	小規模な追加と変更各種訂正
2.24	2011/12/02	KeL	9, F	小規模な訂正
2.25	2012/04/17	KaD	3, 4, F, H	小規模な訂正と追加
2.30	2012/09/12	KeL, KaD	1, 3, 6	M12 コネクタを追加。小規模な変更
2.40	2012/12/04	KeL	1, 10	小規模な変更。Brick バージョンを追加
2.41	2013/03/22	KeL	1, 10, E	小規模な変更
2.42	2013/09/01	KeL	9, B	小規模な変更と訂正

P.3 表記と用語

本マニュアルでは以下の表記を使用します。

- 番号付きリストは手順を表します。
- 番号なしリストは情報を表します。手順ではありません。
- "Anybus" または "モジュール" は、Anybus-CompactCom モジュールを表します。
- "ホスト" または "ホスト アプリケーション" は、Anybus モジュールをホストする機器を表します。
- 16 進数は NNNNh または 0xNNNN の形式で表します。ここで、NNNN は 16 進の値を表します。
- 1 バイトは常に 8 ビットで構成されます。

P.4 サポート

一般的なお問い合わせ情報とサポート窓口については、お問い合わせとサポートのページ (www.hms-networks.com) を参照してください。

1. Anybus-CompactCom PROFINET IO 2-Port について

1.1 概要

Anybus-CompactCom PROFINET IO 2-Port 通信モジュールは、特許技術である Anybus-CompactCom ホストインターフェイスを通じて、PROFINET Real Time 通信を簡単に実現します。本モジュールの機能は、EtherCAT の規格に準拠しているあらゆる機器で利用することができます。これにより、種類を問わず、あらゆるネットワークにおいてシームレスなネットワーク統合が可能となります。

本製品は、Anybus-CompactCom Hardware / Software Design Guide で定義されたアクティブモジュールが使用するあらゆるホストインターフェイスに対応しています。そのため、EtherCAT の規格に準拠したあらゆる機器との間で完全なデータ通信が可能です。通常、新たにネットワークに関するソフトウェアをサポートする必要はありませんが、高度なネットワーク機能を使用する際に専用のソフトウェアが必要となる場合があります。

1.2 機能

- PROFINET IO 通信
- イーサネットコネクタまたは M12 コネクタ
- Brick バージョン
- 最大 2 個の API (API 0 を含む)
- 最大 64 個のモジュール。1 モジュールにつき 8 個のモジュール (最大 128 個のサブモジュール)
- 最大 32767 個の ADI
- 最大 256 バイトのリアルタイム I/O
- 汎用の診断および PROFINET 特有の診断をサポート
- SNMP エージェント
- FTP サーバー
- E-mail クライアント
- Server Side Include (SSI) 機能
- デバイス ID のカスタマイズ
- HMS により提供された汎用の GSD ファイル
- PROFINET ファストスタートアップをサポート
- PROFIenergy プロファイルをサポート

1.2.1 ABCC-PRT IO モジュールとの互換性

ABCC-PRT IO 2-Port のモジュール ID は、ABCC-PRT IO のモジュール ID とは異なります。また、GSD ファイルも異なります。アプリケーションの設計によっては、1 ポートモジュールを 2 ポートモジュールに置き換える際にソフトウェアや設定の変更が必要となる場合があります。

また、2 ポート版は PROFIenergy プロファイルをサポートします。

1.3 前面図

#	項目		コネクタ
1	ネットワークステータス LED ^a		イーサネット (RJ45)
2	モジュールステータス LED ^a		M12 (メス)
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. 起動時に、ネットワークステータス LED およびモジュールステータス LED に従ってテストシーケンスを行います。

ネットワークステータス LED

LED の状態	説明	コメント
オフ	オフライン	- 電源がオフになっています。 - IO コントローラーが接続されていません。
緑	オンライン (RUN)	- IO コントローラーとの接続が確立されていません。 - IO コントローラーが RUN 状態にあります。
緑点滅	オンライン (STOP)	- IO コントローラーとの接続が確立されていません。 - IO コントローラーが STOP 状態にあります。

モジュールステータス LED

LED の状態	説明	コメント
オフ	初期化されていない	電源がオフになっています。または、モジュールが "SETUP" か "NW_INIT" 状態にあります。
緑	通常動作	モジュールが "NW_INIT" 状態から移行しました。
緑 1 回点滅	診断イベント	診断イベントが発生しています。
緑 1 Hz 点滅	DCP 点滅	ネットワーク上のノードを特定するために、エンジニアリングツールで使用されています。
赤	例外エラー	モジュールが "EXCEPTION" 状態にあります。
赤 1 回点滅	設定エラー	期待される ID が実際の ID と異なります。
赤 2 回点滅	IP アドレスエラー	IP アドレスが設定されていません。
赤 3 回点滅	ステーション名エラー	ステーション名が設定されていません。
赤 4 回点滅	内部エラー	モジュール内部で重大なエラーが発生しています。

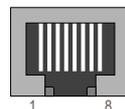
リンク/アクティビティ LED

LED の状態	説明	コメント
オフ	リンクなし	リンクが確立されておらず、通信も行われていません。
緑	リンク検出	イーサネットのリンクが確立していますが、通信は行われていません。
緑点滅	アクティビティ	イーサネットのリンクが確立し、通信が行われています。

イーサネットインターフェイス

イーサネットインターフェイスは、PROFINET で要求される 100 M ビット全二重通信が可能です。

ピン番号	説明
1	TD+
2	TD-
3	RD+
4, 5, 7, 8	直列 RC 回路を介して筐体に接続
6	RD-
ハウジング	ケーブルシールド

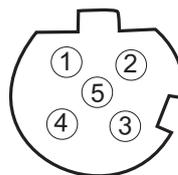


重要:

PROFINET ケーブルの接続についての詳細は、163 ページ「保護接地 (PE) に関する要件」を参照してください。

M12 コネクター (コード D、メス)

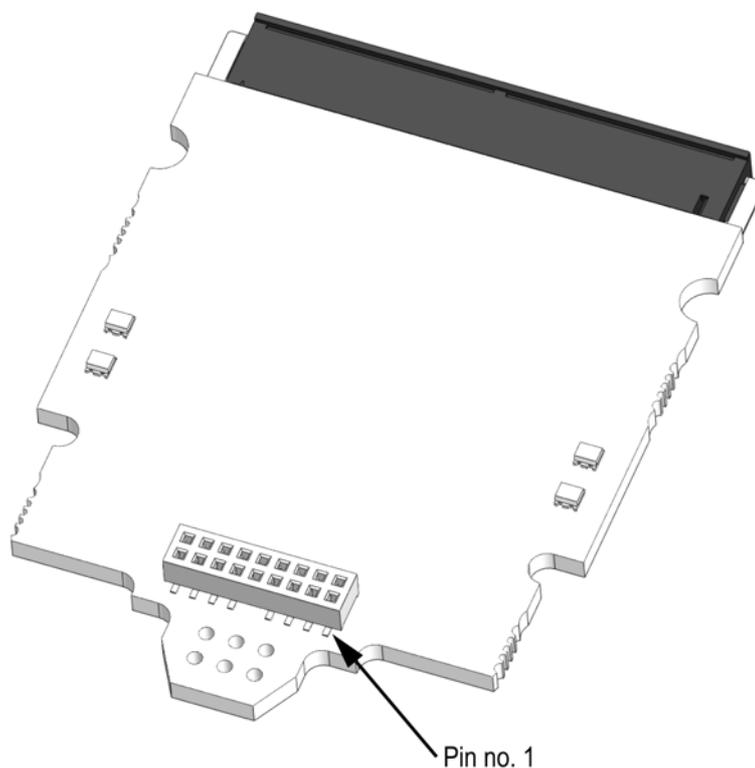
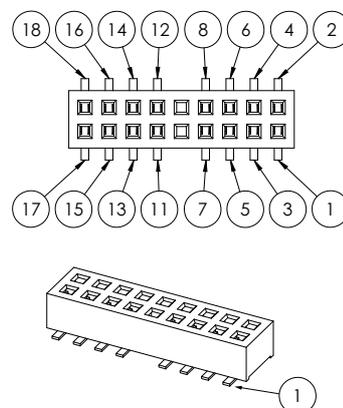
ピン	名前	説明
1	TXD+	送信プラス
2	RXD+	受信プラス
3	TXD-	送信マイナス
4	RXD-	受信マイナス
5 (より線)	シールド	シールド



1.4 ネットワークコネクタ (Brick バージョン)

Anybus CompactCom PROFINET IO 2-Port は、フィールドバスコネクタの代わりにピンコネクタでマザーボード (ホスト機器) に接続する Brick バージョンも用意されています。Brick バージョンのコンセプトとアセンブリは、『Anybus CompactCom Mounting Kit Appendix』(Doc.Id.HMSI-168-30) に記述されています。

ピン番号	信号	ポート番号
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

この章では、Anybus CompactCom (ABCC) 実装チュートリアルの補足事項について説明します。ABCC チュートリアルでは、Anybus CompactCom の簡単な実装例について説明しています。この章では、ホストアプリケーションを動作させ、PROFINET IO ネットワークで使用するための認定を受けるためのネットワーク特有の設定について説明します。

2.2 フィールドバスの適合性に関するメモ

- 本モジュールは、すべてのパラメーターをデフォルトの設定で使用すれば、ネットワークの規格に準拠するようになっています。ただし、HMS が提供する標準の GSD ファイルの変更を伴うパラメーターの変更を行った場合は、再度認定を受ける必要があります。詳細については HMS までお問い合わせください。
- 規格に準拠するため、ホストアプリケーションは、アプリケーションオブジェクト (FFh) にてネットワーク リセット タイプ 02h (パワーオン+工場出荷状態) をサポートする必要があります。

2.3 認定

認定を受けるには以下の手順を行う必要があります。

1. ベンダー ID の変更:

PNO (PROFIBUS Nutzerorganisation e.V.) からベンダー ID を取得済みの場合は、HMS のベンダー ID をその ID に置き換えてください。これを実現するには、PROFINET IO オブジェクト (F6h) のインスタンス 1、アトリビュート 2 を実装し、Get_Attribute を受信したときにそのベンダー ID を返すようにします。

2. デバイス ID の変更:

PROFINET IO オブジェクト (F6h) の HMS デバイス ID を、各自のデバイス ID に置き換えてください。ベンダーの製品は、それぞれ固有のデバイス ID を持つ必要があります。PROFINET IO オブジェクト (F6h) のインスタンス 1、アトリビュート 1 を実装し、Get_Attribute を受信したときにそのデバイス ID を返してください。

3. アトリビュート (ステーションタイプ、I&M オーダー ID、システムの説明) の変更 (推奨):

モジュール ID を変更したら、PROFINET IO オブジェクト (F6h) のインスタンス 1 のステーションタイプ (アトリビュート 3)、I&M オーダー ID (アトリビュート 8)、およびシステムの説明 (アトリビュート 19) を置き換えてください。PROFINET IO オブジェクト (F6h) のインスタンス 1、アトリビュート 3、8、19 を実装してください。Get_Attribute を受信したときに、ステーションタイプ、I&M オーダー ID、システムの説明を返してください。

4. I&M リビジョンの変更 (任意):

顧客に特有の I&M ハードウェア/ソフトウェアリビジョンを表示するには、PROFINET IO オブジェクト (F6h) のインスタンス 1、アトリビュート 10、11 を変更します。PROFINET IO オブジェクト (F6h) のインスタンス 1、アトリビュート

10、11 を実装してください。Get_Attribute を受信したときに、I&M ハードウェア / ソフトウェアリビジョンを返してください。

5. GSD ファイルの変更:

上記の変更に合わせて、PROFINET ABCC GSD ファイルを変更します。

6. MAC アドレスの設定 (任意):

イーサネットホストオブジェクト (F9h) のインスタンス 1、アトリビュート 1、9、10 にて MAC アドレスを設定します。

詳しい情報と説明については、168 ページ「コンフォーマンステストに関するガイド」を参照してください。

3. 基本動作

3.1 概要

3.1.1 ソフトウェアの要件

一般に、Anybus-CompactCom PROFINET IO 2-Port をサポートするために、ネットワーク対応のコードを新たに記述する必要はありません。ただし、PROFINET ネットワークシステムの特性上、いくつかの事項を考慮する必要があります。

- PROFINET では、最大 32767 個の ADI を示すことが可能です。
- ADI の名前、種類、および類似のアトリビュートは、PROFINET からはアクセスできません。これらは、内蔵の Web サーバーを介してネットワーク上に示されます。
- ホストアプリケーションは、最大 5 個の診断インスタンスを作成できます。6 番目のイベントは、重大な障害が発生したときに作成されます。
- 規格に準拠するため、ホストアプリケーションは、アプリケーションオブジェクト (FFh) にてネットワークリセットタイプ 02h (パワーオン + 工場出荷状態) をサポートする必要があります。
- PROFINET 自身は、非周期の要求 (アプリケーションデータオブジェクトへの要求) に関して具体的なタイミングを規定していません。しかし一般に、そのような要求を適切な時間内に処理することが推奨されます (適切な時間が実際にどの程度の長さを意味するかは、実装や実際のインストールにより異なります)。
- ADI がプロセスデータにマッピングされる順序は重要で、ネットワーク通信を設定する際は、IO コントローラーでも同じ順序に設定しなければなりません (マッピングされた ADI と同じ順序、サイズ、方向でモジュールを設定する必要があります)。これらが考慮されないと、ネットワーク接続が確立できず、通信を行うことができません。
- 技術的な理由から、物理的にネットワークに接続されない限り (リンクが検出されない限り)、モジュールは NW_INIT から WAIT_PROCESS に移行しません。
- PROFINET の高度な機能を使用するには、PROFINET ネットワーキングの内部詳細、および公式の PROFINET 規格に関する十分な知識が必要です。本製品の使用者は、規格を入手して十分な知識を得るか、または規格に関する知識を必要としないう範囲で本製品を使用することが求められます。

下記も参照してください。

- 13 ページ 「アプリケーションデータインスタンス (ADI)」
- 58 ページ 「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 「アプリケーションデータオブジェクト (FEh)」

3.1.2 電子データ シート (GSD)

PROFINET では、デバイスの特性は XML データファイルに格納されます。このファイルは "GSD ファイル" と呼ばれ、ネットワーク構成を設定する際にエンジニアリングツールで使用されます。HMS は、汎用の GSD ファイルを提供しています。その内容は、モジュールのデフォルトの設定と同じです。ただし、Anybus-CompactCom コンセプトは柔軟であるため、モジュールの機能を変更した結果、汎用の GSD ファイルの内容が無効になる場合があります。

下記も参照してください。

- 7 ページ 「フィールドバスの適合性に関するメモ」

ビットマップをデバイスアクセスポイントに関連付ける

GSD ファイルを使用すると、ビットマップをデバイスアクセスポイントに関連付けることができます。

デバイスアクセスポイントに関して、以下の情報を追加する必要があります ("`</DeviceAccessPointItem>`" の直前に追加)。

```
<Graphics>
  <GraphicItemRef Type="DeviceSymbol" GraphicItemTarget="X"/>
</Graphics>
```

さらに、画像リストも作成する必要があります。このリストは、"`</DeviceAccessPointList>`" や "`</ValueList>`" といったキーワードの直後に追加できます。なお、上記と下記の "X" は、適切な値に置き換えてください (ビットマップが 1 つだけ使われている場合は、X を 1 に置き換えます)。

```
<GraphicsList>
  <GraphicItem ID="X" GraphicFile="GSDML-VVVV-DDDD-N...N"/>
</GraphicsList>
```

ビットマップ名は上記の形式で指定してください。ここで、VVVV はベンダー ID (例: "010C")、DDDD はデバイス ID (例: "0009")、"N...N" はベンダー固有の拡張文字 (例: "ABCCPRTPIC1") を表します。

初期レコードデータを有効にする

IO 機器と IO コントローラーとの間で IO コネクションを確立している間、IO コントローラーは初期レコードデータを送信できます。この初期レコードデータは、PROFINET IO のレコード書き込みサービスを使用して送信されます。このサービスはいつでも使用可能で、これを使用すると定義された ADI にデータが書き込まれます。初期レコードデータは、GSD ファイルにてモジュールのサブモジュールごとに定義します。デフォルトでは、Anybus CompactCom モジュールは初期レコードデータを使用しませんが、必要に応じて初期レコードデータを有効にできます。

この機能を有効にするには、GSD ファイルを以下のように変更します。この例では、PROFINET IO 接続のスタートアップ時に 2 バイトのデータが ADI 67 (ADI 67 はインデックス 67 に対応) に書き込まれます (データの値はエンドユーザーが設定します)。

```
<RecordDataList>
  <ParameterRecordDataItem Index="67" Length="1" TransferSequence="0">
    <Name TextId="T_ID_EXAMPLE2"/>
    <Ref DataType="Unsigned16" ByteOffset="0" DefaultValue="0"
      AllowedValues="0..65535" TextId="T_ID_EXAMPLE2_PRM_1"/>
  </ParameterRecordDataItem>
</RecordDataList>
```

上記の GSD 情報は、データを関連付けるモジュールの "<IOData>" キーワードの直後に記述することを推奨します。

複数の ADI を設定する必要がある場合は、"ParameterRecordDataItem" キーワードを複製します。

なお、GSD ファイルの "<ExternalTextList>" セクションに、TextId (上記の "T_ID_xxx") を追加する必要があります (定義されている言語ごとに 1 回ずつ)。

```
<Text TextId="T_ID_EXAMPLE2" Value="Config parameter 1"/>
<Text TextId="T_ID_EXAMPLE2_PRM_1" Value="Parameter value description"/>
```

3.2 ネットワーク ID

デフォルトでは、本モジュールは以下に示す Anybus の一般的な実装がなされているものとして自分自身を識別します。

Device ID	0009h (Anybus-CompactCom PROFINET IO)
Vendor ID	010Ch (HMS Industrial Networks)
ステーションタイプ	"ABCC-PRT 2-Port"

Anybus モジュールが汎用の Anybus 製品ではなくベンダー固有の実装に見えるように、ID 情報をカスタマイズできます。ただし、これにより標準の GSD ファイルが無効となるため、その製品は再度認定を受ける必要があります。

下記も参照してください。

- 21 ページ 「識別とメンテナンス (I&M)」
- 123 ページ 「PROFINET IO オブジェクト (F6h)」

3.3 通信設定

ネットワーク関連の通信設定は、ネットワーク設定オブジェクト (04h) にてグループ化されています。Anybus-CompactCom PROFINET IO では、以下の項目が含まれます。

- **イーサネット インターフェイスの設定**
本モジュールは、PROFINET で要求される 100 M ビット全二重通信に固定されています。
- **TCP/IP の設定**
この設定は、モジュールをネットワークに接続するのに必要です。
- **SMTP アカウントの設定**
この設定は、E-mail メッセージを送信するのに必要です。
- **PROFINET ステーション名**
本モジュールを PROFINET に接続するには、ステーション名を割り当てる必要があります。
- **IM1 ~ IM4 に関する各種設定**
この設定は、IM1 ~ IM4 のコンテンツを指定します。

下記も参照してください。

- 21 ページ 「識別とメンテナンス (I&M)」
- 29 ページ 「Web サーバー」
- 62 ページ 「ネットワークコンフィグレーションオブジェクト (04h)」
- 162 ページ 「HICP (ホスト IP 設定プロトコル)」

3.4 ネットワークのデータ交換

3.4.1 アプリケーションデータインスタンス (ADI)

レコードデータ読み出し/書き込みサービスを使用すると、ネットワークから非周期に ADI へアクセスできます。与えられた API とインデックスの範囲でアドレスが指定されると、本モジュールはサービスを変換し、アプリケーションデータオブジェクトへの標準のオブジェクト要求に変換します。その要求に対してホストアプリケーションがエラーを返した場合、そのエラーは PROFINET の規格に変換されます。

以下のパラメーターは、PROFINET 上の ADI のアドレス指定に影響を与えます。

- **アプリケーションプロセスインスタンス (API)**

API 0 (ゼロ) は、アプリケーションデータオブジェクトのデータ (ADI) へのアクセスを提供します。他の API への非周期の要求は、"Get_Record" および "Set_Record" コマンドにより PROFINET IO オブジェクト (F6h) に転送されます。以下、API 0 (ゼロ) を前提に説明します。

- **スロットとサブスロット**

スロットとサブスロットの値は、ADI の実際のアドレス指定には影響を与えません。ただし、実際のスロットとサブスロットをモジュール/サブモジュールに設定する必要があります。これは、スロット #0/サブスロット #1 を占有する DAP (デバイスアクセスポイント) について常に当てはまります。モジュール/サブモジュールに値が設定されていれば、他のスロット/サブスロットの値も使用できます。

- **インデックス**

インデックス番号が 7FFFh 以下の場合、ADI とインデックスの間には 1 対 1 の関係があります。インデックス 0 (ゼロ) は ADI には割り当てられません。このインデックスは使用できません。

API	スロット	サブスロット	インデックス	ADI	コメント
0	0	1	0000h	-	(ADI への割り当てなし)
			0001h	1	デバイスアクセスポイント (DAP)
			0002h	2	
			
			7FFFh	32767	
			8000h ~ FFFFh	-	(ADI への割り当てなし)
	X (>0)	Y	0000h	-	条件付き; X と Y を設定する必要があります
			0001h	1	
			0002h	2	
			
7FFFh			32767		
8000h ~ FFFFh	-	(ADI への割り当てなし)			
>0	-	-	-	-	18 ページ「アプリケーションプロセスインスタンス (API)」を参照

下記も参照してください。

- 14 ページ「注意事項」
- 18 ページ「アプリケーションプロセスインスタンス (API)」
- 123 ページ「PROFINET IO オブジェクト (F6h)」

重要 :0000 ~ 7FFFh において"透過モード"が有効になっている場合、アプリケーションデータオブジェクトには要求は転送されません(この場合、PROFINET IO オブジェクトに転送されます)。

3.4.2 プロセスデータ

ADI をライトプロセスデータにマッピングすると PROFINET の入力データとなり、ADI をリードプロセスデータにマッピングすると PROFINET の出力データになります。ADI に特定のデータタイプの要素が 1 つだけ含まれていれば、各要素が 1 つの識別子 (" モジュール ") となるため、デフォルトでは、このときに限り ADI 全体の整合性が保たれます。

オプションとして、上級ユーザーは、GSD ファイルにてカスタムモジュールやカスタムサブモジュールを定義できます。この場合、複数の要素が存在しても ADI 全体の整合性は保たれます。なお、この場合、ホストアプリケーションはモジュールやサブモジュールの埋め込みに対応する必要があります。

下記も参照してください。

- 19 ページ 「Real Identification (RI)」
- 72 ページ 「コマンド詳細 :Plug_Module」
- 73 ページ 「コマンド詳細 :Plug_Submodule」
- 77 ページ 「コマンド詳細 :API_Add」

重要 :ADI がプロセス データにマッピングされる順序は重要で、ネットワーク通信を設定する際は、IO コントローラーでも同じ順序に設定しなければなりません (マッピングされた ADI と同じ順序、サイズ、方向でモジュールを設定する必要があります)。これらが考慮されないと、ネットワーク接続が確立できず、通信を行うことができません (19 ページ 「Real Identification (RI)」 を参照)。

3.4.3 注意事項

レコードデータ要求におけるパラメーター長は、読み出し / 書き込みバイト数を指定します。

- ADI の実際のサイズより大きなデータを読み出した場合、応答には実際の ADI データのみ含まれます。すなわち、モジュールはデータのパディングを行いません。
- ADI に書き込みを行った場合、モジュールはパラメーター長をチェックしません。すなわち、書き込んだデータのサイズが、要求された ADI の実際のサイズと異なる場合、ホストアプリケーションはエラーを返さなければなりません。

下記も参照してください。

- 18 ページ 「アプリケーションプロセスインスタンス (API)」

3.5 診断

標準の診断オブジェクト (02h) を使用すると、基本的な診断機能を利用できます。修復不可能な重大なイベントが発生すると、本モジュールはネットワークから物理的に切断します。これにより、ネットワークへの接続が阻止されます。その他の重大なイベントが発生した場合、イベントコードに応じて、チャンネル診断エントリ / 警報、または一般の診断エントリ / 警報が発生します。

ホストアプリケーションは、最大 5 個の診断インスタンスを作成できます。6 番目のイベントは、修復不可能な重大な障害が発生したときに作成されます。

下記も参照してください。

- 20 ページ 「チャンネル診断とプロセスアラーム」
- 58 ページ 「診断オブジェクト (02h)」
- 81 ページ 「PROFINET 追加診断オブジェクト (0Fh)」

3.6 Web インターフェイス

内蔵の Web サーバーを使用すると、SSI スクリプトによる内容豊富な動的コンテンツを提供できます。これにより、ファイルシステムにある情報 / 設定や、Anybus-CompactCom オブジェクトモジュールへのアクセスが可能となります。

Web サーバーのコンテンツは、フラッシュメモリのファイルシステム上にあります。そのため、必要に応じて標準の FTP クライアントからコンテンツにアクセスしたりコンテンツをカスタマイズしたりできます。

下記も参照してください。

- 16 ページ 「ファイルシステム」
- 27 ページ 「FTP サーバー」
- 29 ページ 「Web サーバー」
- 35 ページ 「Server Side Include (SSI)」

3.7 E-mail クライアント

内蔵の E-mail クライアントを使用すると、ファイルシステムに保存されている E-mail メッセージや、SMTP クライアントオブジェクト (09h) にて直接定義された E-mail メッセージを、ホストアプリケーションから送信できます。このとき、メッセージ内に SSI コンテンツが含まれているかどうかスキャンされます。そのため、ファイルシステムや Anybus-CompactCom オブジェクトモデルから動的情報を埋め込むことができます。

下記も参照してください。

- 3-16 「ファイルシステム」
- 7-34 「E-mail クライアント」
- 8-35 「Server Side Include (SSI)」
- 10-103 「SMTP クライアントオブジェクト (09h)」

3.8 ファイルシステム

3.8.1 概要

内蔵のファイルシステムは、2 MB の不揮発性ストレージを備えています。このファイルシステムには、HTTP/FTP サーバーや E-mail クライアント、ホストアプリケーションからアクセスできます。

このファイルシステムでは、以下の表記法を使用します。

- "\" (バックslash) はパスの区切り文字として使用します。
- システムのルートからのパスは、\" で始めなければなりません。
- パスは \" で終了することはできません。
- 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 / 終了することはできません。
- 名前には以下の文字を含めることはできません: \ / : * ? < > |
- 名前は 48 文字以内でなければなりません。
- パスは 255 文字以内でなければなりません (ファイル名を含む)。

下記も参照してください。

- 27 ページ 「FTP サーバー」
- 29 ページ 「Web サーバー」
- 34 ページ 「E-mail クライアント」
- 35 ページ 「Server Side Include (SSI)」
- 108 ページ 「ファイルシステム・インターフェースオブジェクト (OAh)」

重要: ファイルシステムはフラッシュメモリ上にあります。技術的な理由により、各フラッシュセグメントは約 100000 回消去すると故障します。そのため、ランダムアクセスのストレージには適していません。

以下の操作を行うと、1 つまたは複数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、名前変更
- 既存のファイルへの上書きまたは追加
- ファイルシステムのフォーマット

3.8.2 システムファイル

ファイルシステムには、システムの設定に使用される一連のファイルが格納されています。" システムファイル " として知られるこれらのファイルは、通常の ASCII ファイルで、標準のテキストエディター (Microsoft Windows™ のメモ帳など) で編集できます。これらのファイルの形式は、いくつかの例外を除き、" キー " という考えに基づいています。それぞれの " キー " には、以下のように値を割り当てることができます。

例:

[キー 1]
キー 1 の値

[キー 2]
キー 2 の値

4. PROFINET IO の実装詳細

4.1 概要

この章では、Anybus の実装における PROFINET に関する詳細について説明します。PROFINET の機能を使用するには、PROFINET ネットワーキングの内部詳細、および公式の PROFINET 規格に関する十分な知識が必要です。本製品のユーザーは、規格を入手して十分な知識を得るか、または規格に関する知識を必要としない範囲で本製品を使用することが求められます。

実装概要：

- **適合性クラス**
Anybus モジュールは適合性クラス B に準拠しています。
- **性能特性**
 - 100 Mbps、全二重、デフォルトで自動ネゴシエーションが有効
 - リアルタイム (RT) 通信、サイクルタイム 2 ms
- **機器モデル**
 - 1 個の IO 機器インスタンス
 - 各 IO 機器インスタンスには、識別子 (API) で参照されるアプリケーションプロセスが含まれます。デフォルトで API 0 (ゼロ) が実装されており、ホストアプリケーションにより追加の API を実装できます。
 - 各 API には 1 つまたは複数のスロットが実装されています。
 - 各スロットには 1 つまたは複数のサブスロットが実装されています。
 - 各サブスロットには 1 つまたは複数のチャンネルを実装できます。
- **スロットとサブスロット**
最大 64 個のスロット。1 スロットにつき 8 個のサブスロット (最大 128 個のサブスロット)
- **IO データ**
各方向に対応した最大 256 バイトの IO データ

重要 :Anybus-CompactCom コンセプトは柔軟であるため、本製品の動作を変更した結果、標準の GSD ファイルの内容と異なる動作になる場合があります。その場合、カスタムの GSD ファイルを作成し、本製品の認定を再度受ける必要があります。

下記も参照してください。

- 7 ページ「フィールドバスの適合性に関するメモ」
- 10 ページ「電子データ シート (GSD)」

4.2 アプリケーションプロセスインスタンス (API)

先に述べたように、API 0 に対する非周期の要求は、アプリケーションデータオブジェクトに転送されます。他の API への非周期の要求は、"Get_Record" および "Set_Record" コマンドにより PROFINET IO オブジェクト (F6h) に転送されます。

デフォルトでは、周期データ交換は API 0 を介して行われます (Anybus によりモジュールとサブモジュールが API 0 に割り当てられます)。特定のプロファイルをサポートするため、PROFINET 固有のコマンドである "API_Add" を使用して、("SETUP" 状態中に) カスタム API を追加できます。なお、この場合、ホストアプリケーションはモジュールやサブモジュールの埋め込みに対応する必要があります (19 ページ「Real Identification (RI)」を参照)。

API 番号は特定のプロファイルの実装に関連付けられているため、API を任意に選ぶことはできません。

下記も参照してください。

- 19 ページ「Real Identification (RI)」
- 127 ページ「コマンド詳細 :Get_Record」
- 129 ページ「コマンド詳細 :Set_Record」
- 70 ページ「ネットワーク PROFINET IO オブジェクト (0Eh)」(77 ページ「コマンド詳細 :API_Add」)

4.3 アプリケーションリレーションシップ (AR)

PROFINET では、IO コントローラー/スーパーバイザーと I/O 機器 (この場合は Anybus) との接続は "アプリケーションリレーションシップ" (AR) と呼ばれます。Anybus モジュールは複数のアプリケーションリレーションシップを同時にサポートします。そのため、複数の IO スーパーバイザーがそのデータや機能にアクセスできます。

ホスト側の実装では、この機能を完全に無視するか (この場合、Anybus モジュールがこの機能を自動的に処理します)、アプリケーションリレーションシップの確立と処理をホストのファームウェアに組み込むかのいずれかを選択できます。

アプリケーションリレーションシップは以下の機能により管理されます。

- AR_Check_Ind (135 ページ「コマンド詳細 :AR_Check_Ind」を参照)
- AR_Info_Ind (138 ページ「コマンド詳細 :AR_Info_Ind」を参照)
- AR_Offline_Ind (141 ページ「コマンド詳細 :AR_Offline_Ind」を参照)
- AR_Abort_Ind (143 ページ「コマンド詳細 :AR_Abort_Ind」を参照)
- AR_Abort (79 ページ「コマンド詳細 :AR_Abort」を参照)

また、AR の処理では以下のコマンドが付随するため、ホストアプリケーションは、各アプリケーションリレーションシップに関連する要求を追跡できます。

- Get_Record (127 ページ「コマンド詳細 :Get_Record」を参照)
- Set_Record (129 ページ「コマンド詳細 :Set_Record」を参照)
- Cfg_Mismatch_Ind (137 ページ「コマンド詳細 :Cfg_Mismatch_Ind」を参照)
- End_Of_Prm_Ind (140 ページ「コマンド詳細 :End_Of_Prm_Ind」を参照)
- Appl_State_Ready (78 ページ「コマンド詳細 :Appl_State_Ready」を参照)

4.4 Real Identification (RI)

4.4.1 概要

Anybus モジュールへの IO コネクションを確立する際、IO コントローラーから得られた設定 ("Expected Identification") と Anybus モジュールの実際の設定 ("Real Identification" または RI) が比較されます。

- **デフォルト設定**

デフォルトの状態 (アプリケーションが API_Add、Plug_Module、Plug_Submodule を発行していない状態) では、マッピングされたプロセスデータに基づいて、Anybus が以下のようにモジュールとサブモジュールの埋め込みを自動的に処理します。

- DAP はスロット 0 (ゼロ) に埋め込まれます
- モジュールは順番に追加されます (マッピングコマンドの順に追加)
- 要素を 1 つだけ持つ ADI には、モジュールが 1 つだけ追加されます
- 複数の要素を持つ ADI には、要素と同じ数のモジュールが追加されます
- 1 つのモジュールにつき 1 つのサブモジュール
- 全てのモジュールは API 0 (ゼロ) に属します

例:

ADI 番号	種類	オーダー番号	結果として得られる Real Identification
-	-	-	モジュール 000000011h、サブモジュール 00000000h をスロット 0 に追加 (DAP with PDEV) 物理的なサブモジュールは表示されません。
6	SINT16	1	モジュール 000000003h、サブモジュール 00000000h をスロット 1、サブスロット 1 に追加
3	UINT8	2	モジュール 000000002h、サブモジュール 00000000h をスロット 2、サブスロット 1 に追加
9	SINT32	3	モジュール 000000004h、サブモジュール 00000000h をスロット 3、サブスロット 1 に追加

- **カスタム設定 (上級ユーザーのみ)**

オプションとして、PROFINET 固有のコマンドである "API_Add"、"Plug_Module"、"Plug_Submodule" を使用して、"SETUP" 状態中にデフォルトの設定を上書きすることができます。この方法でカスタム モジュールやカスタムサブモジュールを定義することで、ホストアプリケーションは PROFINET 上で ADI がどのように示されるかを正確に定義できます。ただし、該当するモジュールやサブモジュールを含むカスタム GSD ファイルを使う必要があります。

下記も参照してください。

- 18 ページ「アプリケーションプロセスインスタンス (API)」
- 72 ページ「コマンド詳細 :Plug_Module」
- 73 ページ「コマンド詳細 :Plug_Submodule」
- 77 ページ「コマンド詳細 :API_Add」
- 161 ページ「フローチャート - Real Identification (RI) の確立」

4.4.2 設定の不一致

設定の不一致が存在する場合、Anybus は IO コントローラーに対し、サービス応答の "module diff block" によりその不一致を通知します。また、"Cfg_Mismatch_Ind" を発行し、問題のブロックをホストアプリケーションに通知します。

Anybus は、スロット / サブスロットの設定の不一致を無条件に拒絶します (設定の不一致があった場合、本モジュールは "PROCESS ACTIVE" 状態に移行しません)。ただし、ホストアプリケーションは、以下のいずれかの方法で Expected Identification に合わせるすることができます。

- Anybus モジュールを再起動し、"Cfg_Mismatch_Ind" コマンドによりアプリケーションに渡された Expected Identification に一致するように、設定を変更します。
- 後のステージ ("End_Of_Prm_Ind" コマンドと "Appl_State_Ready" コマンドの間が望ましい) において、問題があるモジュール / サブモジュールを削除し、Expected Identification に一致する新しいモジュール / サブモジュールを挿入することで、設定を変更します。

下記も参照してください。

- 137 ページ「コマンド詳細 :Cfg_Mismatch_Ind」
- 140 ページ「コマンド詳細 :End_Of_Prm_Ind」
- 78 ページ「コマンド詳細 :Appl_State_Ready」
- 161 ページ「フローチャート - Real Identification (RI) の確立」

4.5 チャネル診断とプロセスアラーム

PROFINET IO は、アラームを使って IO コントローラーに診断エントリを通知します。Anybus の実装では、診断オブジェクト (02h) または PROFINET の追加診断オブジェクト (0Fh) を使ってアプリケーションが診断エントリを作成できます。前者は基本的な診断のみをサポートするのに対し、後者は詳細なチャネル診断やプロセスアラームを作成できます。

標準の診断オブジェクト (02h) と同様の方法でインスタンスが作成されます。またこのインスタンスには、ソース API やスロット / サブスロット番号、チャネル番号、チャネルタイプを付けることができます。各インスタンスはチャネル診断エントリに対応しており、"delete" コマンドを使用して後で削除できます。

プロセスアラームは、オブジェクト固有の "Process_Alarm" コマンドで作成されます。各アラームには、ソース API やスロット / サブスロット番号、アラームの種類を指定する構造体、アラームに関連するデータを付けることができます。ただし、このコマンドを発行してもインスタンスは作成されないため、"delete" コマンドで削除することはできません。

下記も参照してください。

- 15 ページ「診断」
- 58 ページ「診断オブジェクト (02h)」
- 81 ページ「PROFINET 追加診断オブジェクト (0Fh)」
- 84 ページ「コマンド詳細 :Process_Alarm」

重要 : ユーザーは、この機能のコンセプトや、標準の診断オブジェクト (02h) により提供される機能の使い方を熟知していることが求められます。

4.6 識別とメンテナンス (I&M)

4.6.1 概要

識別とメンテナンス (I&M) は、I/O 機器の情報を収集する標準的な手段を提供します。I&M の情報は、非周期のレコードデータ読み出し / 書き込みサービスを使用して IO コントローラーからアクセスできます。

デフォルトでは、Anybus モジュールは以下の I&M0 をサポートしています。

デフォルトの I&M0 情報：

IM メーカー ID	010Ch (HMS Industrial Networks)
IM オーダー ID	"ABCC-PRT (2-Port)"
IM シリアル番号	(一意のシリアル番号。製造時に設定)
IM ハードウェアリビジョン	(Anybus のハードウェアリビジョン ID。製造時に設定)
IM ソフトウェアリビジョン	(Anybus のソフトウェアリビジョン。製造時に設定)
IM リビジョンカウンター	(リビジョンカウンター)
IM プロファイル ID	F600h (汎用機器)
IM プロファイル固有タイプ	0004h (プロファイルなし)
IM バージョン	0101h
IM サポート	0000h (IM0 をサポート)

オプションとして、ホストアプリケーションはこれらの I&M エントリの情報をカスタマイズできます。または、"Get_IM_Record" コマンドおよび "Set_IM_Record" コマンドを実装し、全てのスロットの全ての I&M レコードをサポートできます。

下記も参照してください。

- 123 ページ「PROFINET IO オブジェクト (F6h)」
- 131 ページ「コマンド詳細 :Get_IM_Record」
- 133 ページ「コマンド詳細 :Set_IM_Record」
- 62 ページ「ネットワークコンフィグレーションオブジェクト (04h)」

4.6.2 I&M のデータ構造

I&M は以下のデータ構造を使用します。

レコード	内容	サイズ	説明
I&M0	メーカー ID	2 バイト	PROFINET IO オブジェクト (F6h)、アトリビュート #2 ("ベンダー ID/I&M ベンダー ID")
	オーダー ID	20 バイト	PROFINET IO オブジェクト (F6h)、アトリビュート #8 ("I&M オーダー ID")
	シリアル番号	16 バイト	PROFINET IO オブジェクト (F6h)、アトリビュート #9 ("I&M シリアル番号")
	ハードウェアリビジョン	2 バイト	PROFINET IO オブジェクト (F6h)、アトリビュート #10 ("I&M ハードウェアリビジョン")
	ソフトウェアリビジョン	4 バイト	PROFINET IO オブジェクト (F6h)、アトリビュート #11 ("I&M ソフトウェアリビジョン")
	リビジョンカウンター	2 バイト	PROFINET IO オブジェクト (F6h)、アトリビュート #12 ("I&M リビジョンカウンター")
	プロファイル ID	2 バイト	PROFINET IO オブジェクト (F6h)、アトリビュート #13 ("I&M プロファイル ID")
	プロファイル固有タイプ	2 バイト	PROFINET IO オブジェクト (F6h)、アトリビュート #14 ("I&M プロファイル固有タイプ")
	IM バージョン	2 バイト	PROFINET IO オブジェクト (F6h)、アトリビュート #15 ("I&M バージョン")
	IM バージョン	2 バイト	PROFINET IO オブジェクト (F6h)、アトリビュート #15 ("I&M バージョン")
I&M1	タグ機能	32 バイト	ネットワークコンフィグレーションオブジェクト (04h)、アトリビュート #16 ("I&M タグ機能")
	タグ位置	22 バイト	ネットワークコンフィグレーションオブジェクト (04h)、アトリビュート #17 ("I&M タグ位置")
I&M2	インストール日	16 バイト	ネットワークコンフィグレーションオブジェクト (04h)、アトリビュート #18 ("I&M インストール日")
I&M3	Descriptor	54 バイト	ネットワークコンフィグレーションオブジェクト (04h)、アトリビュート #19 ("I&M Descriptor")
I&M4 ^a	署名	54 バイト	デフォルト: 全バイトをゼロ (00h) に設定

a. このフィールドのデータは、IO コントローラ / スーパーバイザーによりネットワークからのみアクセスすること。

下記も参照してください。

- 123 ページ「PROFINET IO オブジェクト (F6h)」

4.7 ファストスタートアップ

4.7.1 概要

ファストスタートアップ (FSU) 機能を使用すると、ネットワークに接続されている PROFINET IO 機器を素早く起動できます。この機能は、素早い交換が必要なロボットアプリケーションなどで便利です。Anybus CompactCom モジュールを設定する際は、この機能を有効にする必要があります。

GSD ファイルでは、この機能に関するいくつかのキーワードが使用されています。FSU 時間は、ハードウェアリセット (またはパワーオン) から PROFINET IO 通信が確立されるまでの時間をミリ秒単位で定義します。FSU 時間の測定値が約 1500 ms より長い場合は、この機能を無効にすることを推奨します。

この機能では以下のキーワードが使用されます (デバイスアクセスポイント用)

- **PowerOnToCommReady**

FSU 時間。単位: ミリ秒 (ms) デフォルト値: 0 ms

- **DCP_HelloSupported**

パワーオン時にその機器が "Hello" メッセージを送信するかどうかを指定するキーワードです。デフォルト値: true

FSU を無効にするには、キーワードを以下の値に設定してください。

- **PowerOnToCommReady**

GSD ファイルからこのキーワードを削除してください。

- **DCP_HelloSupported**

値: false

4.7.2 STEP7 を用いたファストスタートアップの設定

以下の例では、Siemens のツールである STEP7 を用いて設定する場合の手順を示します。

ファストスタートアップのアクティベート

1. 設定ツールを起動します。STEP7 の HW Config ウィンドウを以下に示します。ファストスタートアップのアクティベートでは、Module 列の拡大表示した部分を使用します。

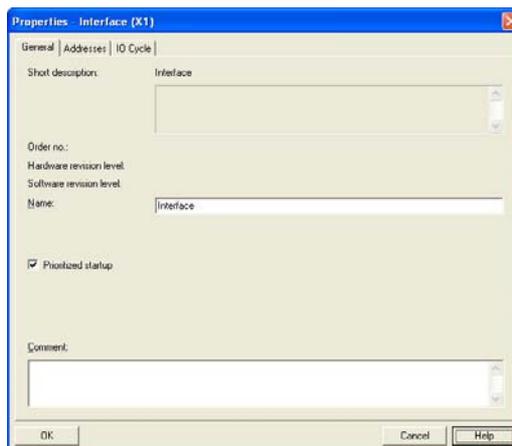
The screenshot shows the HW Config window for a SIMATIC 300 system. The rack configuration is as follows:

Slot	Module	Order Number	I Address	Q Address	Diagnostic address	Comment
1	PS 307 2A					
2	CPU 319-3 PN/DP					
X7	MP/DP					
X2	DP					
X3	PN/IO					
X3 F1	Port 1					
3						
4						
5						
6						

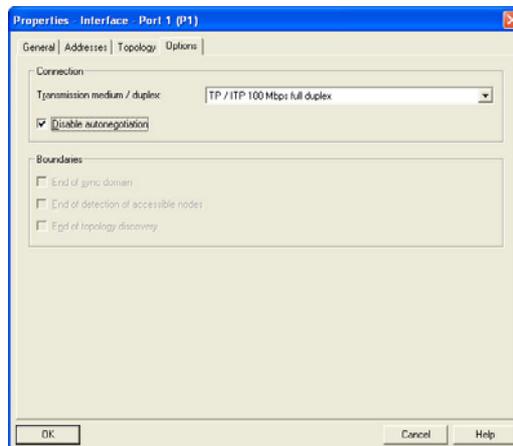
The expanded view of the ABCC-PRT 2-Port module shows the following details:

Slot	Module	Order Number
0	ABCC-PRT 2-Port	ABCC-PRT (2-Port)
X7	Interface	
F1	Port 1	
F2	Port 2	
1		

2. Module 列の "Interface" をダブルクリックします。右側に示すウィンドウが表示されます。"General" タブを選択し、"Prioritized startup" チェックボックスをオンにします。



3. HW Config ウィンドウに戻ります。Module 列の "Port 1" をダブルクリックします。右側に示すウィンドウが表示されます。Options タブを選択します。可能な限り早いスタートアップを設定するには、転送レートとして "100 Mbps, full duplex" を選択し、"Disable autonegotiation" チェックボックスをオンにします。



4. Port 2 についても同じ手順を繰り返します。

4.8 PROFIenergy プロファイル

Anybus CompactCom PROFINET IO 2-Port モジュールは、PROFIenergy 技術規格 rev. 1.0 に基づいて、PROFIenergy プロファイルをサポートしています。このプロファイルを使用すると、昼休みや週末などに、機器を一時的に省電力モードにできます。これにより、機器が積極的に使用されていないときの消費電力を減らすことができます。各機器は、生産が停止している時間に応じて、最適な省電力モードを個別に設定できます。例えば工場のオペレーターは、容易に全ての機器を省電力モードに設定できます。このとき、機器ごとに最適な省電力モードにすることが可能です。

アプリケーションは、各機器をどのくらいの時間だけ省電力モードにするかを設定します。また各機器は、どのモードが最適かを判断します。"Ready to operate" モードと各省電力モードとの間の遷移は必須です。一方、省電力モード間の遷移は任意です。機器の電源の再投入は手動による再起動を意味するため、"Ready to operate" モードから "Power off" モードへの遷移は必須ではありません。

4.8.1 実装

Anybus CompactCom PROFINET IO 2-Port モジュールでは、PROFIenergy 技術規格に記載されているステートマシンに基づいて、PROFIenergy プロファイルが実装されています(この技術規格は PROFIBUS International より入手可能)。ネットワークから送信された PROFIenergy コマンドは、以下に示すように、Anybus CompactCom の実装に変換されます。

PROFIenergy のコマンド	サブコマンド	Anybus CompactCom における実装
Start_Pause	-	Energy コントロールオブジェクトのインスタンス 0 への StartPause コマンドに変換されます。p. 153 を参照。
End_Pause	-	Energy コントロールオブジェクトのインスタンス 0 への EndPause コマンドに変換されます。p. 154 を参照。
Query_Modes ^a	List_Energy_Saving_Modes	Energy オブジェクトのインスタンスが示されます。PROFIenergy プロファイルにおける Energy コントロールオブジェクトのインスタンス / モードの最大数は 254 です。
	Get_Mode	要求されたモード / インスタンスでサポートされているアトリビュートが読み込まれます。サポートされていないアトリビュートはデフォルト値に設定されます。
PEM_Status ^a	-	インスタンス 0、および現在省電力モードにある Energy オブジェクトのインスタンスの状態を読み込みます。
PE_Identify ^a	-	PROFINET IO オブジェクト (F6h) の "PROFIenergy 機能" のアトリビュート (No. 23) を読み込みます。
Query_Measurement	Get_Measurement_List	サポートされていません。
	Get_Measurement_Values	

a. このコマンドは、PROFINET IO オブジェクト (F6h) の PROFIenergy アトリビュート (No. 23) により有効化 / 無効化されます。

PROFIenergy プロファイルは全てのサブスロットに対して有効で、インデックス 80A0h にてアクセスできます。この機能は、PROFINET IO オブジェクト (F6h) のアトリビュート 23 により有効化 / 無効化されます。このアトリビュートには、PROFIenergy 機能の各部分を有効 / 無効にするマスクビットが用意されています。Energy コントロールオブジェクトのインスタンス / モードの最大数は 254 です。

下記も参照してください。

- 150 ページ 「Energy コントロールオブジェクト (F0h)」
- 123 ページ 「PROFINET IO オブジェクト (F6h)」

5. FTP サーバー

5.1 概要

カテゴリ : 拡張

内蔵の FTP サーバーにより、標準の FTP クライアントを使用してファイルシステムを容易に管理できます。

デフォルトでは、FTP 通信では以下のポート番号が使用されます。

- TCP、ポート 20 (FTP データポート)
- TCP、ポート 21 (FTP コマンドポート)

この FTP サーバーは、最大 8 個の同時接続が可能です。

5.2 ユーザーアカウント

ユーザーアカウントは、設定ファイル "\ftp.cfg" に格納されています。このファイルには、ユーザー名、パスワード、および各ユーザーのホームディレクトリが格納されています。ユーザーは、各自のホームディレクトリの外にあるファイルにはアクセスできません。

ファイル形式 :

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

オプションとして、UserN:PasswordN セクションを以下のように置き換えて、ユーザーリストを含むファイルへのパスを指定することができます。

ファイル形式 ("\ftp.cfg"):

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

ユーザーリストを含むファイルは以下の形式で記述します。

ファイル形式 :

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

注 :

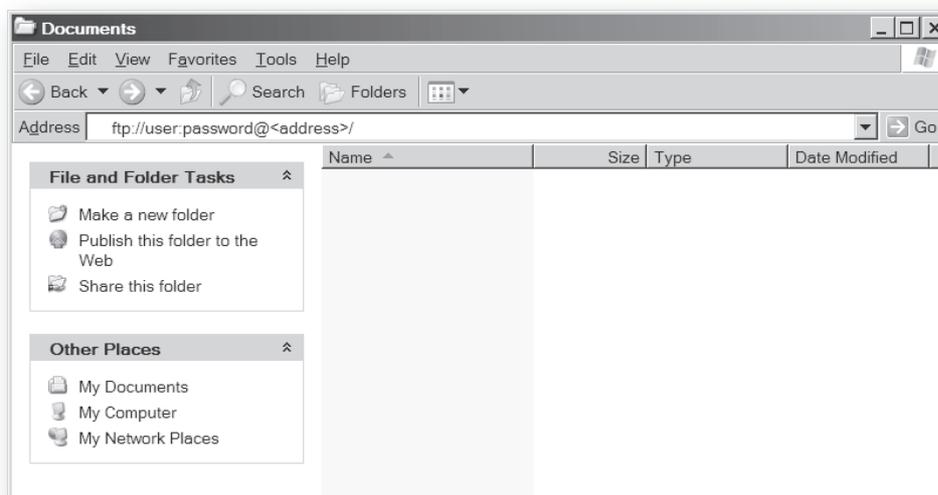
- ユーザー名は 15 文字以内でなければなりません。
- パスワードは 15 文字以内でなければなりません。
- ユーザー名とパスワードには英数文字のみ使用可能です。
- "\ftp.cfg" が見つからない、またはその内容を解釈できない場合、全てのユーザー名 / パスワードの組み合わせが受け付けられ、ホームディレクトリが FTP のルート ("\ftp\") になります。

- ユーザーがログインできるためには、ユーザーのホームディレクトリがファイルシステム上に存在する必要があります。"ftp.cfg" にユーザー情報を追加するだけでは不十分です。
- イーサネットオブジェクトにて "管理者モード" が有効になっている場合、全てのユーザー名 / パスワードの組み合わせが受け付けられ、そのユーザーはファイルシステムに無制限にアクセスできます (ホームディレクトリがシステムのルートになります)。
- ルート ("\\") にアクセスできる権限を持ったユーザーを最低 1 つ用意することを強く推奨します。さもなければ、システムファイル ("ftp.cfg" を含む) を変更する必要があります。必要が生じるたびに、"管理者モード" を有効にする必要があります。

5.3 セッションの例

Windows エクスプローラには FTP クライアントが内蔵されており、以下の手順でファイルシステムに容易にアクセスできます。

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 を起動します。
2. アドレスバーにて FTP://<user>:<password>@<address> と入力します。
 - <address> は Anybus モジュールの IP アドレスに読み替えてください
 - <user> はユーザー名に読み替えてください
 - <password> はパスワードに読み替えてください
3. Enter を押します。すると、エクスプローラは、指定された設定にて Anybus モジュールへの接続を試みます。接続に成功すると、エクスプローラのウィンドウにファイルシステムが表示されます。



6. Web サーバー

6.1 概要

カテゴリ : 拡張

内蔵の Web サーバーは、エンドユーザーへの対話型の環境の提供や、設定を行うための柔軟な環境を提供します。SSI とクライアントサイドスクリプティングの強力な組み合わせにより、オブジェクトやファイルシステムデータへのアクセスが可能となり、高度なグラフィカルユーザーインターフェースの作成が可能です。

Web インターフェイスはファイルシステムに格納されており、FTP サーバーを介してアクセスできます。必要であれば、イーサネットホストオブジェクトにて Web サーバーを完全に無効にできます。

この Web サーバーは、最大 20 個の同時接続をサポートし、ポート 80 経由で通信します。

下記も参照してください。

- 27 ページ「FTP サーバー」
- 35 ページ「Server Side Include (SSI)」
- 147 ページ「Ethernet ホストオブジェクト (F9h)」

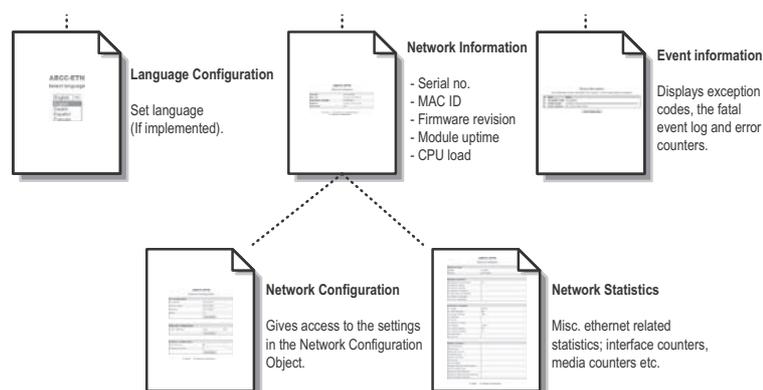
6.2 デフォルトの Web ページ

デフォルトの Web インターフェイスは、一連の仮想ファイルで構成されています。同じ名前のファイルを同じ場所 (Web ルート) に置くことで、これらの仮想ファイルを置き換えることができますが、仮想ファイルを完全には削除できません。

このファイルは、現状のまま使用することも、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。

仮想ファイルは以下のとおりです。

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



注: これらのいずれかのファイルを使用する場合、ファイルシステムインターフェイスオブジェクトにて仮想ファイルシステムを完全に無効にすることを推奨します。

下記も参照してください。

- 16 ページ「ファイルシステム」
- 108 ページ「ファイルシステム・インターフェースオブジェクト (0Ah)」

6.2.1 ネットワークの設定

ネットワーク設定ページでは、ネットワーク設定オブジェクトにおける TCP/IP および SMTP の設定を変更するためのインターフェイスが用意されています。

ABCC-PRT (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"/>
DHCP:	<input type="checkbox"/>
Host name:	<input type="text"/>
Domain name:	<input type="text"/>
<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"/>	

▶ [Main](#) ▶ [Network interface](#)

変更内容を反映するにはモジュールをリセットする必要があります。
編集可能な設定は以下のとおりです。

IP の設定

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
DHCP	DHCP を有効 / 無効にするためのチェックボックス デフォルト値 : 有効
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

SMTP の設定

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

6.2.2 イーサネットの統計ページ

イーサネットの統計情報に関する Web ページでは、以下の情報が表示されます。

イーサネット接続		説明
Port 1	Speed:	現在の接続速度
	Duplex:	現在のデュプレックス設定
Port 2	Speed:	現在の接続速度
	Duplex:	現在のデュプレックス設定

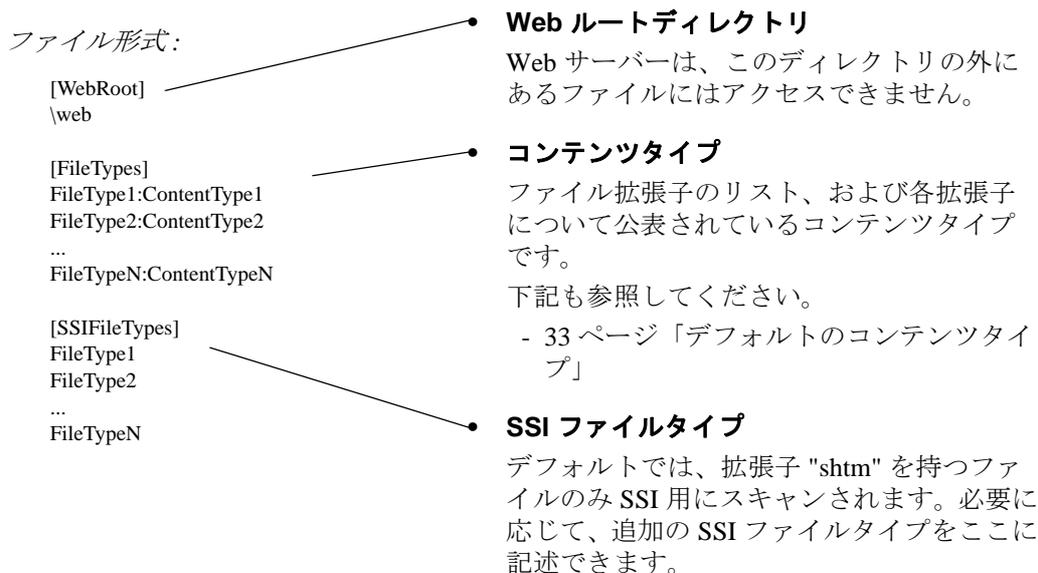
インターフェイス カウンター	説明
In Octets:	受信バイト数
In Ucast Packets:	ユニキャストの受信パケット数
In NUcast packets:	ユニキャスト以外の受信パケット数 (ブロードキャストおよびマルチキャスト)
In Discards:	メモリバッファの空きがないために破棄された受信パケット数
In Errors:	受信エラーのために破棄された受信パケット数
In Unknown Protos:	サポートされていないプロトコルタイプの受信パケット数
Out Octets:	送信バイト数
Out Ucast packets:	ユニキャストの送信パケット数
Out NUcast packets:	ユニキャスト以外の送信パケット数 (ブロードキャストおよびマルチキャスト)
Out Discards:	メモリバッファの空きがないために破棄された送信パケット数
Out Errors:	送信エラー

6.3 サーバーの設定

6.3.1 概要

カテゴリ : 高度

Web サーバーの基本設定は、システムファイル "\http.cfg" に格納されています。このファイルには、Web インターフェイスのルートディレクトリ、コンテンツタイプ、および SSI 用にスキャンされるファイルタイプのリストが格納されています。



Web ルートディレクトリにより、Web インターフェイスに関する全てのファイルの場所が決定されます。このディレクトリの外にあるファイルおよびそのサブディレクトリは、Web サーバーからアクセスできません。

6.3.2 インデックスページ

本モジュールは、以下の順序でインデックスページを検索します。

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

注 1: <WebRoot> は、"\http.cfg" で指定されたルートディレクトリに読み替えてください。

注 2: インデックスページが見つからなかった場合、デフォルトの仮想インデックスファイルが使用されます (仮想インデックスファイルが存在する場合)。

下記も参照してください。

- 29 ページ「デフォルトの Web ページ」

6.3.3 デフォルトのコンテンツタイプ

デフォルトでは、ファイル拡張子により以下のコンテンツタイプが認識されます。

ファイル拡張子	公表されてい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/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

サーバー設定ファイルにコンテンツタイプを追加することで、コンテンツタイプを追加または再定義できます。32 ページ「概要」を参照してください。

6.3.4 認証

ディレクトリ内に "web_accs.cfg" というファイルを置くことで、そのディレクトリを Web からアクセスできないように設定できます。このファイルには、対象となるディレクトリおよびそのサブディレクトリにアクセスできるユーザーのリストを記述します。

ファイル形式：

```

Username1:Password1
Username2:Password2
...
UsernameN:PasswordN

```

アクセスを許可されたユーザーのリスト

```

[AuthName]
(message goes here)

```

オプションとして、[AuthName] キーを記述することでログインメッセージを指定できます。このメッセージは、保護されたディレクトリにアクセスしたときに Web ブラウザーに表示されます。

オプションとして、アクセスを許可されたユーザーのリストを他のファイルにリダイレクトできます。

例：

この例では、アクセスを許可されたユーザーは "here.cfg" および "too.cfg" から読み込まれます。

```

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

```

```

[AuthName]
Howdy.Password, please.

```

7. E-mail クライアント

7.1 概要

カテゴリ : 拡張

内蔵の E-mail クライアントにより、アプリケーションは SMTP サーバーを介して E-mail メッセージを送信できます。メッセージは、SMTP クライアントオブジェクトにて直接指定することも、ファイルシステムから取得することもできます。後者の方法には SSI も含まれますが、技術的な理由により一部のコマンドは使用できません (各 SSI コマンドに対して個別に指定されています)。

このクライアントは、"LOGIN" メソッドによる認証をサポートしています。アカウントの設定などは、ネットワーク設定オブジェクトに格納されています。

下記も参照してください。

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

7.2 E-mail メッセージの送信

E-mail メッセージを送信するには、SMTP アカウントの設定を指定する必要があります。

以下の設定が必要です。

- 有効な SMTP サーバー アドレス
- 有効なユーザー名
- 有効なパスワード

E-mail メッセージを送信するには以下のようにします。

1. "Create" コマンド (03h) を使用して新規 E-mail インスタンスを作成します。
2. E-mail インスタンスにて送信者、受信者、件名、本文を指定します。
3. E-mail インスタンスに対して "Send Instance Email" (10h) コマンドを発行します。
4. オプションとして、"Delete" コマンド (04h) を使用して E-mail インスタンスを削除できます。

ファイルシステム内のファイルを使用してメッセージを送信するには、"Send Email from File" コマンドを使用します。ファイル形式の説明については、106 ページ「コマンド詳細 :Send Email From File」を参照してください。

8. Server Side Include (SSI)

8.1 概要

カテゴリ：高度

Server Side Include (SSI) 機能を使用すると、ファイルやオブジェクトからデータを読み込み、Web ページや E-mail メッセージに出力できるようになります。

SSI は、ソースドキュメントに埋め込まれる特殊なコマンドです。Anybus モジュールは、そのようなコマンドを検出すると、そのコマンドを実行し、指定された動作にてそのコマンドを実行結果で置き換えます(置き換え可能な場合)。

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

8.2 ファイルのインクルード

この機能は、ファイルの内容をインクルードします。この内容は SSI によりスキャンされます。

注：この機能は E-mail メッセージでは使用できません。

構文：

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

filename- 入力元ファイル

デフォルト出力：

シナリオ	デフォルト出力
成功	(ファイルの内容)

8.3 コマンド機能

8.3.1 概要

コマンド機能は、コマンドを実行してその結果をインクルードします。

一般的な構文：

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

command- コマンド機能。下記を参照。

コマンド機能：

コマンド	E-mail メッセージで有効	ページ
GetConfigItem()	Yes	37
SetConfigItem()	No	38
SsiOutput()	Yes	40
DisplayRemoteUser	No	40
ChangeLanguage()	No	41
IncludeFile()	Yes	42
SaveDataToFile()	No	43
printf()	Yes	44
scanf()	No	46

8.3.2 GetConfigItem()

このコマンドは、ファイルシステム内のファイルにある特定の情報を返します。

ファイル形式：

入力元ファイルは以下の形式でなければなりません。

```
[ キー 1 ]
値 1

[ キー 2 ]
値 2
...
[ キー N ]
値 N
```

構文：

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

filename- 読み込み元のファイル
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 (例 : "
")
(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
成功	(指定されたキーの値)
認証エラー	"Authentication error"
ファイルオープンエラー	"Failed to open file "filename""
キーが見つからない	"Tag (key) not found "

例：

以下の SSI を ...

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

... 以下のファイルと組み合わせて使用します ("\fruit.cnf")...

```
[Apple]
Green

[Lemon]
Yellow

[Banana]
Blue
```

... すると、"Yellow" という文字列が返されます。

8.3.3 SetConfigItem()

この関数は、HTML フォームをファイルシステム内のファイルに保存します。

注: この機能は E-mail メッセージでは使用できません。

構文:

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

filename- 保存先ファイル。指定されたファイルが存在しない場合、そのファイルが作成されます (パスが有効な場合)。

Overwrite - オプション。コマンドを発行するたびに新規ファイルを作成します。

デフォルトでは、既存のファイルが変更されます。

ファイル形式:

各フォームオブジェクトは [タグ] として保存され、その後に実際の値が続きます。

[フォームオブジェクト名 1]
フォームオブジェクト値 1

[フォームオブジェクト名 2]
フォームオブジェクト値 2

[フォームオブジェクト名 3]
フォームオブジェクト値 3

...

[フォームオブジェクト名 N]
フォームオブジェクト値 N

注: 名前がアンダースコア ("_") で始まる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
成功	"Configuration stored to " <i>filename</i> "
認証エラー	"Authentication error"
ファイルオープンエラー	"Failed to open file " <i>filename</i> "
ファイル書き込みエラー	"Could not store configuration to " <i>filename</i> "

例：

以下に、この関数の使用例を示します。結果として得られるページは、自分自身にフォームを送信します。送信されたフォームは、SetConfigItem コマンドで評価されます。

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

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

<FORM action="test.shtml">
  <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>
```

結果として得られるファイル ("food.txt") は以下のようになります。

```
[Name]
Cliff Barnes

[Food]
Cheese

[Drink]
Beer
```

注：このサンプルを機能させるには、HTML ファイルの名前を "test.shtml" にする必要があります。

8.3.4 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- 成功した場合に使用する文字列

failure - 失敗した場合に使用する文字列

デフォルト出力：

(このコマンド自体は何も出力を生成しません)

例：

以下に、このコマンドの使用例を示します。

```
<!--#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「SSI の出力設定」

8.3.5 DisplayRemoteUser

このコマンドは、認証セッションにあるユーザー名を返します。

注：このコマンドは E-mail 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
成功	(現在のユーザー)

8.3.6 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の設定を変更します。

注: このコマンドは E-mail メッセージでは使用できません。

構文:

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

source - 新規の言語設定を含むフォームオブジェクトの名前
引数には以下に示す 1 桁の値を指定します。

フォームの値	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

デフォルト出力:

シナリオ	デフォルト出力
成功	"Language changed"
エラー	"Failed to change language "

例:

以下に、この関数の使用例を示します。結果として得られるページは、自分自身にフォームを送信します。送信されたフォームは、ChangeLanguage() コマンドで評価されます。

```
<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: このサンプルを機能させるには、HTML ファイルの名前を "test.shtm" にする必要があります。

8.3.7 IncludeFile()

このコマンドは、ファイルの内容をインクルードします。なお、この内容は SSI によりスキャンされません。

構文：

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

filename- 入力元ファイル
separator- オプション。行区切り文字を指定 (例: "
")

デフォルト出力：

シナリオ	デフォルト出力
成功	(ファイルの内容)
認証エラー	"Authentication error"
ファイルオープンエラー	"Failed to open file "filename""

例：

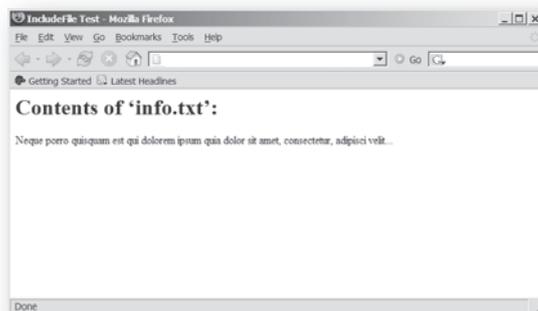
以下に、この関数の使用例を示します。

```
<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" の内容：

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

結果として得られるページをブラウザで開くと、以下のように表示されます。



下記も参照してください。

- 35 ページ「ファイルのインクルード」

8.3.8 SaveDataToFile()

この関数は、HTML フォームからデータをファイルシステム内のファイルに保存します。複数のフォームオブジェクトからの内容は、空行 (2 個の CRLF) により区切られます。

注: このコマンドは E-mail 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- 保存先ファイル。指定されたファイルが存在しない場合、そのファイルが作成されます (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と、特定のフォームオブジェクトのデータのみ保存されます。デフォルトでは、名前がアンダースコア ("_") で始まるものを除く全てのフォームオブジェクトのデータが保存されます。

Overwrite|Append- 既存のファイルにデータを上書きするか追加するかを指定します。

デフォルト出力:

シナリオ	デフォルト出力
成功	"Configuration stored to "filename" "
認証エラー	"Authentication error"
ファイル書き込みエラー	"Could not store configuration to "filename" "

例:

以下に、この関数の使用例を示します。結果として得られるページは、自分自身にフォームを送信します。送信されたフォームは、SaveDataToFile コマンドで評価されます。

```
<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>
```

結果として得られるファイル ("stuff.txt") には、"Meat" というフォームオブジェクトに入力された値が含まれます。

注: このサンプルを機能させるには、HTML ファイルの名前を "test.shtm" にする必要があります。

8.3.9 printf()

この関数は、Anybus モジュールやアプリケーションからのデータを書式付き文字列で返します。ここで使用される書式構文は、C 言語の標準関数 `printf()` と同様です。

この関数は、0 個以上の書式タグとそれに続く引数を含むテンプレート文字列を受け付けます。各書式タグは 1 個の引数と対応しており、引数をどのようにして人間が読める形にするかを決定します。

構文：

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

template- 引数をどのように表現するかを決定するテンプレートです。書式タグを何個でも記述可能です。ここで記述した書式タグは、後ろに続く引数に置き換えられ、指定された書式に変換されます。書式タグの数は引数の数と同じでなければなりません。両者が一致しない場合、結果は不定となります。

書式タグは以下のように記述します。

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

下記も参照してください。

- 45 ページ「書式タグ」

argument- 入力元引数；テンプレート文字列に挿入される実際の入力元データを指定するオプションパラメーター引数の数は書式タグの数と同じでなければなりません。両者が一致しない場合、結果は不定となります。

現時点では、`ABCCMessage()` のみ引数に指定可能です。

下記も参照してください。

- 48 ページ「`ABCCMessage()`」

デフォルト出力：

シナリオ	デフォルト出力
成功	(<code>printf()</code> の結果)
ABCCMessage エラー	ABCCMessage のエラー文字列 (8-51 “エラー”)

例：

下記も参照してください。

- 48 ページ「`ABCCMessage()`」
- 50 ページ「例 (`Get_Attribute()`):」

書式タグ

- 型指定 (必須)

型指定文字列は必ず指定してください。型指定文字列は以下の基本表現を決定します。

型指定文字	表現	例
c	1 個の文字	b
d, i	1 個の 10 進整数	565
e, E	指数表現による浮動小数点数	5.6538e2
f	通常の固定小数点表現による浮動小数点数	565.38
g, G	指数が -4 未満または精度以上の場合、%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。末尾のゼロ / 小数点は出力されません。	565.38
o	符号なし 8 進数	1065
s	文字列	Text
u	符号なし 10 進整数	4242
x, X	16 進整数	4e7f
%	リテラルの % です。値は割り当てられません。	%

- フラグ (任意)

フラグ文字	意味
-	指定された文字幅内で左寄せで出力されます (デフォルトは右寄せ)
+	値が正か負かを示すため、常に "+" または "-" が出力されます。
(空白)	値が "+" または "-" で始まらない場合、先頭に空白が挿入されます。
0 (ゼロ)	空白の代わりにゼロが埋められます。
#	%e、%E、%f の場合、小数部がない場合でも、値には常に小数点が含まれます。%x および %X の場合、それぞれ 0x または 0X が先頭に追加されます。

- 文字幅 (任意)

文字幅	意味
値	出力される最小の文字数を指定します。 出力される文字数がこの値より小さい場合、フィールドの幅まで出力結果に空白が埋められます。出力される文字幅が指定文字幅より長い場合でも、出力結果は切り捨てられません。
*	書式文字列では文字幅を指定しません。文字幅は、書式変換される引数の前に記述した整数値で指定されます。

- . 精度 (任意)

このフィールドの実際の意味は、型指定文字により異なります。

型指定文字	意味
d, i, o, u, x, X	出力される最小の桁数を指定します。出力される桁数がこの値より小さい場合、フィールドの幅まで出力結果に空白が埋められます。なお、出力される桁数が指定桁数より長い場合でも、出力結果は切り捨てられません。
e, E, f	出力される小数桁数を指定します (デフォルトは 6)
g, G	出力される最大の有効桁数を指定します。
s	出力される最大の文字数を指定します。
c	(何も起こりません)

- 修飾子

修飾文字	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。

8.3.10 scanf()

この関数は、先に述べた `printf()` 関数とよく似ていますが、出力ではなく入力で使用される点が異なります。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読み込み、指定されたテンプレート文字列を用いて構文解析し、その結果を指定された引数に渡します。ここで使用される書式構文は、C 言語の標準関数 `scanf()` と同様です。

この関数は、0 個以上の書式タグとそれに続く引数を含む、入力元のテンプレート文字列を受け付けます。各引数は 1 個の書式タグと対応しています。この書式タグは、HTML フォームから読み込んだデータを、出力先の引数に渡す前にどのように解釈するかを決定します。

注：このコマンドは E-mail 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 -** 文字列を抽出する 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構文解析して解釈するかを指定します。後の引数に渡す前にデータをどのように変換するか指定する書式タグを、何個でも記述可能です。書式タグの数は引数の数と同じでなければなりません。両者が一致しない場合、結果は不定となります。
- 書式タグは以下のように記述します。
- ```
%[*][Width][Modifier]type
```
- 下記も参照してください。
- 47 ページ「書式タグ」
- argument-** 解釈されたデータの出力先引数です。引数の数は書式タグの数と同じでなければなりません。両者が一致しない場合、結果は不定となります。
- 現時点では、`ABCCMessage()` のみ引数に指定可能です。
- 下記も参照してください。
- 48 ページ「`ABCCMessage()`」

デフォルト出力：

| シナリオ             | デフォルト出力                           |
|------------------|-----------------------------------|
| 成功               | "Success"                         |
| 構文解析エラー          | "Incorrect data format "          |
| 引数に対してデータが多すぎる   | "Too much data "                  |
| ABCC Message エラー | ABCCMessage のエラー文字列 (51 ページ「エラー」) |

例：

下記も参照してください。

- 48 ページ「`ABCCMessage()`」
- 50 ページ「例 (`Set_Attribute()`):」

## 書式タグ

- 型指定 (必須)

型指定文字列は必ず指定してください。型指定文字列は以下の基本表現を決定します。

| 種類                  | 入力                                                                                                                                                                                     | 引数のデータタイプ                                     |
|---------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------|
| c                   | 1 個の文字                                                                                                                                                                                 | CHAR                                          |
| d                   | 符号付き 10 進整数を受け付けます。                                                                                                                                                                    | SINT8<br>SINT16<br>SINT32                     |
| i                   | 符号付きまたは符号なし 10 進整数を受け付けます。10 進数、16 進数、8 進数で与えることが可能です。どの形式で与えるかは、入力データの先頭文字で指定します。<br>先頭文字: 形式:<br>0x 16 進数<br>0 8 進数<br>1 ~ 9 10 進数                                                   | SINT8/UINT8<br>SINT16/UINT16<br>SINT32/UINT32 |
| u                   | 任意の符号付き 10 進整数を受け付けます。                                                                                                                                                                 | UINT8<br>UINT16<br>UINT32                     |
| o                   | 任意の符号付き 8 進整数を受け付けます。                                                                                                                                                                  | SINT8/UINT8<br>SINT16/UINT16<br>SINT32/UINT32 |
| x, X                | 任意の符号付き 16 進整数を受け付けます。                                                                                                                                                                 | SINT8/UINT8<br>SINT16/UINT16<br>SINT32/UINT32 |
| e, E,<br>f,<br>g, G | 任意の符号付き浮動小数点数を受け付けます。浮動小数点数の入力形式は、文字列<br>- 符号付きの値が可能です<br>- 指数表現された値が可能です。この場合、10 進数の有理数の後ろに、整数値および "E" または "e" で構成された指数フィールドを記述します。                                                   | FLOAT                                         |
| n                   | 入力データを消費しません; 対応する引数は、scanf がオブジェクトの入力から読み込んだ文字数を書き込む整数型の引数です。                                                                                                                         | SINT8/UINT8<br>SINT16/UINT16<br>SINT32/UINT32 |
| s                   | 空白を含まない文字列を受け付けます。                                                                                                                                                                     | STRING                                        |
| [scanset]           | scanlist により指定された集合に含まれる文字で構成された、空白を含まない文字列を受け付けます (例: "[0123456789ABCDEF]"). 集合の最初の文字として、リテラルの "]" を指定できます。最初の "]" の直後にキャレット (^) を記述すると、scanlist が反転されます (リストされた文字以外の全ての文字が受け付けられます)。 | STRING                                        |
| %                   | この入力場所に 1 個の "%" を入力できます; 値の割り当てや変換は行われません。完全な変換仕様は "%%" です。                                                                                                                           | -                                             |

- \*(任意)

データは読み込まれますが無視されます。対応する引数には割り当てられません。

- 文字幅 (任意)

読み込む最大の文字数を指定します。

- 修飾子 (任意)

異なるデータ サイズを指定します。

| 修飾子 | 意味                             |
|-----|--------------------------------|
| h   | SINT8, SINT16, UINT8 or UINT16 |
| l   | SINT32 or UINT32               |

## 8.4 引数関数

### 8.4.1 概要

引数関数は、特定のコマンド関数のパラメーターとして渡されます。

一般的な構文：

(構文は状況により異なります)

引数関数：

| 関数            | 説明 | ページ |
|---------------|----|-----|
| ABCCMessage() | -  | 48  |

### 8.4.2 ABCCMessage()

この関数は、モジュール/ホストアプリケーション内のオブジェクトに対してオブジェクト要求を発行します。

構文：

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

- object - 出力先のオブジェクトを指定します
- instance- 出力先の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- コマンドの MsgData[] サブフィールドにおける実際の内容を指定します
  - データは直接表現で与えます (形式は c\_type によります)
  - 親コマンド (scanf() など) によりデータが与えられる場合、"ARG" キーワードを指定します
- c\_type - コマンドにおけるデータ型を指定します (msgdata)
 

下記も参照してください。

  - 49 ページ「コマンドデータタイプ (c\_type)」
- r\_type - 応答におけるデータ型を指定します (msgdata)
 

下記も参照してください。

  - 49 ページ「応答データタイプ (r\_type)」

以下の形式にて数値入力を与えることができます。

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

下記も参照してください。

- 50 ページ「例 (Get\_Attribute):」
- 50 ページ「例 (Set\_Attribute):」

- コマンドデータタイプ (c\_type)

配列をサポートするデータ型の場合、サフィックス "[n]" を用いて要素の値を指定できます。ここで、"n" は要素の数を表します。各データ要素は空白で区切ります。

| 種類     | サポートする配列 | データ形式 (msgdata で与える形式)                                                                     |
|--------|----------|--------------------------------------------------------------------------------------------|
| BOOL   | Yes      | 1                                                                                          |
| SINT8  | Yes      | -25                                                                                        |
| SINT16 | Yes      | 2345                                                                                       |
| SINT32 | Yes      | -2569                                                                                      |
| UINT8  | Yes      | 245                                                                                        |
| UINT16 | Yes      | 40000                                                                                      |
| UINT32 | Yes      | 32                                                                                         |
| CHAR   | Yes      | A                                                                                          |
| STRING | No       | "abcde"<br>注: バックスラッシュ ("\") を前に置くと、引用符を記述できます。例: "We usually refer to it as \"the Egg\" " |
| FLOAT  | Yes      | 5.6538e2                                                                                   |
| NONE   | No       | コマンドにはデータが含まれないため、データ型はありません。                                                              |

- 応答データタイプ (r\_type)

配列をサポートするデータ型の場合、サフィックス "[n]" を用いて要素の値を指定できます。ここで、"n" は要素の数を表します。

| 種類     | サポートする配列 | コメント                                                                                                                                                             |
|--------|----------|------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| BOOL   | Yes      | 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に置き換えることができます。この場合、関数が実際に返すデータ型は STRING になります。<br>構文: BOOL<true><false><br>配列の場合の形式は BOOL[n]<true><false> となります。 |
| SINT8  | Yes      | -                                                                                                                                                                |
| SINT16 | Yes      | -                                                                                                                                                                |
| SINT32 | Yes      | -                                                                                                                                                                |
| UINT8  | Yes      | オブジェクトから ENUM データ型を読み込む場合もこのデータ型を使用できます。この場合、実際には ENUM 型の値が返されます。                                                                                                |
| UINT16 | Yes      | -                                                                                                                                                                |
| UINT32 | Yes      | -                                                                                                                                                                |
| CHAR   | Yes      | -                                                                                                                                                                |
| STRING | No       | -                                                                                                                                                                |
| ENUM   | No       | このデータ型を使用する場合、最初に ABCCMessage() 関数により ENUM 型の値が読み込まれます。そして、実際の列挙型文字列を取得するために、ABCCMessage() 関数により "Get Enum String" コマンドが発行されます。応答における実際のデータ型は STRING になります。      |
| FLOAT  | Yes      | -                                                                                                                                                                |
| NONE   | No       | 応答にはデータが含まれないため、データ型はありません。                                                                                                                                      |

**重要:** メッセージは出力先のオブジェクトに透過的に渡されることに注意してください。SSI のエンジンは、オブジェクトの指定方法が間違っているかどうかをチェックしません。例えば、(間違った) メッセージデータを持つ Get\_Attribute 要求は、明らかに間違っている場合でも、そのまま修正されずにオブジェクトに渡されます。この注意点を守らない場合、データが消失したり、思わしくない副作用が発生したりする場合があります。

## 例 (Get\_Attribute):

この例では、printf() および ABCCMessage() を用いて IP アドレスを取得する方法を示します。

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

| 変数      | Value    | コメント                      |
|---------|----------|---------------------------|
| オブジェクト  | 4        | ネットワーク設定オブジェクト (04h)      |
| インスタンス  | 3        | インスタンス #3 (IP アドレス)       |
| コマンド    | 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にはデータは含まれません。    |
| r_type  | UINT8[4] | 4 個の符号なし 8 bit 整数を要素に持つ配列 |

下記も参照してください。

- 8-44 “printf()”

## 例 (Set\_Attribute):

この例では、scanf() および ABCCMessage() を用いて IP アドレスを設定する方法を示します。なお、この例では、(scanf() により構文解析された) フォームデータを使用するようモジュールに指示する特殊なパラメーター "ARG" が使われています。

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

| 変数      | Value    | コメント                         |
|---------|----------|------------------------------|
| オブジェクト  | 4        | ネットワーク設定オブジェクト (04h)         |
| インスタンス  | 3        | インスタンス #3 (IP アドレス)          |
| コマンド    | 2        | Set_attribute                |
| ce0     | 5        | アトリビュート #5                   |
| ce1     | 0        | -                            |
| msgdata | ARG      | scanf() により構文解析されたデータを使用します。 |
| c_type  | UINT8[4] | 4 個の符号なし 8 bit 整数を要素に持つ配列    |
| r_type  | NONE     | 応答メッセージにはデータは含まれません。         |

下記も参照してください。

- 46 ページ 「scanf()」

## エラー

オブジェクト要求の結果がエラーとなった場合、応答におけるエラーが評価され、以下のように人間が読める形に変換されます。

| 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「SSI の出力設定」

## 8.5 SSI の出力設定

オプションとして、"output.cfg" ファイルを追加することで SSI の出力を恒久的に変更できます。

ファイル形式：

```
[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 で示さ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「エラー」

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

"%s" を使用してファイル名をインクルードします。

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

"%s" を使用してファイル名をインクルードします。

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

"%s" を使用してファイル名をインクルードします。

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

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

各タグの "X" の値を言語ごとに変えることで、上記の内容をファイルに複数回インクルードできます。その場合、言語設定に基づいて正しい出力が選択されます。選択された言語に関する情報が見つからない場合、デフォルトの SSI 出力が使用されます。

| 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。

- 40 ページ「SsiOutput()」

## 9. SNMP エージェント

### 9.1 概要

シンプルネットワークマネジメントプロトコル (SNMP。RFC1157 規格を参照) は、ネットワークに接続されている機器の状態をネットワーク管理システムにて監視するのに使用されます。管理ステーションに管理エージェントがインストールされ、取得/設定要求によりデータが交換されます。

### 9.2 管理情報 (MIB)

MIB は、SNMP エージェントによりアクセスされる装置データベースです。Anybus-CompactCom Profinet IO 2-Port モジュールは、標準の MIB である LLDP-MIB および MIB-II をサポートします。標準の MIB は RFC 規格で規定されています。この MIB には、グループ分けされた変数が格納されます。ホストアプリケーションは、MIB-II の一部の変数値を変更できます。

### 9.3 MIB-II

ABCC-PRT IO 2-Port モジュールの MIB-II には、システムグループとインターフェイスグループが含まれています。以下の表に、装置の状態を監視するための、MIB-II 規格 (RFC1213) に基づく変数を示しています。アクセス権限は、SNMP プロトコルでアクセスする場合の権限を表します。

### 9.3.1 システムグループ変数

| 変数          | アクセス権限 | 説明                                                                                                                  | 入力元                                                                                 |
|-------------|--------|---------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------|
| sysDescr    | 読み取り専用 | 装置の説明データ型：表示文字列（印刷可能な ASCII 文字のみ）最大 255 文字。工場出荷時の設定："HMS Industrial Networks, Anybus CompactCom PROFINET IO 2-port" | PROFINET オブジェクト；インスタンスアトリビュート 19 - システムの説明 11-124 「インスタンスアトリビュート（インスタンス #1）」を参照     |
| sysObjectID | 読み取り専用 | N/A. 値 = 0                                                                                                          | 内部使用                                                                                |
| sysUpTime   | 読み取り専用 | 最後に電源投入してからの時間（単位：100 分の 1 秒）                                                                                       | 内部使用                                                                                |
| sysContact  | 読み取り専用 | 装置の問い合わせ担当者、および問い合わせ情報。データ型：表示文字列。最大 255 文字。工場出荷時の設定："www.anybus.com"                                               | PROFINET オブジェクト；インスタンスアトリビュート 22 - システムの問い合わせ先 11-124 「インスタンスアトリビュート（インスタンス #1）」を参照 |
| sysName     | 読み取り専用 | 装置名 (Profinet ステーション名) データ型：表示文字列。最大 254 文字。工場出荷時の設定：空の文字列                                                          | ネットワーク設定オブジェクト；インスタンスアトリビュート 15 - ステーション名                                           |
| sysLocation | 読み取り専用 | 装置の物理的な場所 (IM タグロケーション) データ型：(表示文字列)。最大 22 文字。工場出荷時の設定：22 個の空白                                                      | ネットワーク設定オブジェクト；インスタンスアトリビュート 17 - IM タグロケーション                                       |
| sysServices | 読み取り専用 | 装置の機能値 = 74。この値は、装置が OSI モデルの第 2 層 (スイッチ)、第 4 層 (TCP)、第 7 層 (アプリケーション) の機能を持つことを表す。                                 | 内部使用                                                                                |

### 9.3.2 インターフェイスグループ変数

内部使用の値を持つ変数は、全て読み取り専用です。括弧内の値はポート番号を表します (1 - ポート 1、2 - ポート 2、3 - 仮想ポート)。

| 変数                  | Data type | Value <sup>a</sup>                                                                | 説明                                                                                                 |
|---------------------|-----------|-----------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------|
| ifNumber            | integer   | 3                                                                                 | 現在のネットワークインターフェイス数。Constant                                                                        |
| ifIndex(1..3)       | integer   | ifIndex(1) = 1<br>ifIndex(2) = 2<br>ifIndex(3) = 3                                | 各インターフェイスに固有の値 Constant                                                                            |
| ifDescr(1..3)       | 8 進文字列    | ifDescr(1) = "port-001"<br>ifDescr(2) = "port-002"<br>ifDescr(3) = "port-virtual" | インターフェイスに関する情報。STEP7 のトポロジスキャナと互換性を保つため、ifDescr(1) = "port-001"、ifDescr(2) = "port-002" でなければならない。 |
| ifType(1..3)        | integer   | 6 ("Ethernet-csmacd")                                                             | インターフェイスの種類                                                                                        |
| ifMtu(1..3)         | integer   | 1500                                                                              | インターフェイスにて送信 / 受信されたダイアグラムの最大サイズ。オクテットで指定                                                          |
| ifSpeed(1..3)       | gauge     | 0 または 100 000 000                                                                 | ビット / 秒で表した、イーサネットのデータ転送レート接続状態が "up" であるポートの速度のみ示される                                              |
| ifPhysAddress(1..3) | 8 進文字列    |                                                                                   | ポートの MAC アドレス                                                                                      |

| 変数                      | Data type | Value <sup>a</sup>                         | 説明                                                                                                                      |
|-------------------------|-----------|--------------------------------------------|-------------------------------------------------------------------------------------------------------------------------|
| ifAdminStatus(1..3)     | integer   | 1 ("up")                                   | イーサネットポートの目的の状態                                                                                                         |
| ifOperStatus(1..3)      | integer   | 1 ("up") または 2 ("down")                    | イーサネットポートの現在の動作。接続中 = "up"、未接続 = "down"。                                                                                |
| ifLastChange(1..3)      | timeticks | 状態が変更された時間。<br>ifLastChange(3) = 0 の場合を除く。 | ポートが現在の状態に変更された時間 (起動からの時間)。前記の変数を参照。100 分の 1 秒単位で示される                                                                  |
| ifInOctets(1..3)        | counter   | ifInOctets(1..3) = オクテット数                  | インターフェイスで受信されたトータルのオクテット数。フレーミング文字を含む。                                                                                  |
| ifInUcastPkts(1..3)     | counter   | ifInUcastPkts(1..3) = ユニキャストパケット数          | 上位層のプロトコルに渡されたサブネットワークのユニキャストパケット数                                                                                      |
| ifInNUcastPkts(1..3)    | counter   | ifInNUcastPkts(1..3) = ユニキャスト以外のパケット数      | 上位層のプロトコルに渡された、ユニキャスト以外 (サブネットワークのブロードキャストまたはサブネットワークのマルチキャスト) のパケット数。                                                  |
| ifInDiscards(1..3)      | counter   | ifInDiscards(1..3) = 破棄されたパケット数            | 破棄されたインバウンドパケットの数 (エラーは検出されなかったが、上位層のプロトコルに渡されなかったインバウンドパケットの数)。(バッファの空を確保する場合もパケットが破棄される)                              |
| ifInErrors(1..3)        | counter   | ifInErrors(1..3) = エラーパケット数                | エラーが発生したインバウンドパケットの数                                                                                                    |
| ifInUnknownProtos(1..3) | counter   | ifInUnknownProtos(1..3) = 不明なパケット数         | インターフェイス経由で受信されたが、不明または非サポートのプロトコルであったために破棄されたパケットの数                                                                    |
| ifOutOctets(1..3)       | counter   | ifOutOctets(1..3) = オクテット数                 | インターフェイスから送信されたトータルのオクテット数。フレーミング文字を含む。                                                                                 |
| ifOutUcastPkts(1..3)    | counter   | ifOutUcastPkts(1..3) = ユニキャストパケット数         | 上位層のプロトコルにより、サブネットワークのユニキャストアドレスに送信するよう要求されたトータルのパケット数。破棄されたパケット数および送信されなかったパケット数を含む。                                   |
| ifOutNUcastPkts(1..3)   | counter   | ifOutNUcastPkts(1..3) = ユニキャスト以外のパケット数     | 上位層のプロトコルにより、ユニキャスト以外 (サブネットワークのブロードキャストおよびサブネットワークのマルチキャスト) のアドレスに送信するよう要求されたトータルのパケット数。破棄されたパケット数および送信されなかったパケット数を含む。 |
| ifOutDiscards(1..3)     | counter   | ifOutDiscards(1..3) = 破棄されたパケット数           | 破棄されたアウトバウンドパケットの数 (エラーは検出されなかったが、送信されなかったアウトバウンドパケットの数)。(バッファの空を確保する場合もパケットが破棄される)                                     |
| ifOutErrors(1..3)       | counter   | ifOutErrors(1..3) = エラーパケット数               | エラーが発生したため送信できなかった送信パケットの数                                                                                              |
| ifOutQLen(1..3)         | gauge     | ifOutQLen(1..3) = キューに格納されているパケット数         | 出力パケットキューの長さ (単位: パケット数)                                                                                                |
| ifSpecific(1..3)        | objid     | .0.0                                       | 特定のメディアのインターフェイスを実現するため、そのメディアに関する MIB の定義を参照する。ここではそのような参照はできないため、全てのポートに対して固定値を使用する。                                  |

a. 特に指定がない場合、変数の値は 0

## 10. Anybus モジュールオブジェクト

### 10.1 概要

この章では、本モジュールで実装されている Anybus モジュールオブジェクトについて説明します。

標準オブジェクト：

- 57 ページ「Anybus オブジェクト (01h)」
- 58 ページ「診断オブジェクト (02h)」
- 61 ページ「ネットワークオブジェクト (03h)」
- 62 ページ「ネットワークコンフィグレーションオブジェクト (04h)」

ネットワーク固有オブジェクト：

- 86 ページ「ソケットインターフェイスオブジェクト (07h)」
- 103 ページ「SMTP クライアントオブジェクト (09h)」
- 108 ページ「ファイルシステム・インターフェースオブジェクト (0Ah)」
- 121 ページ「ネットワークイーサネットオブジェクト (0Ch)」
- 70 ページ「ネットワーク PROFINET IO オブジェクト (0Eh)」
- 81 ページ「PROFINET 追加診断オブジェクト (0Fh)」

## 10.2 Anybus オブジェクト (01h)

### カテゴリ

基本

### オブジェクトの説明

このオブジェクトには、一般的な Anybus データがすべて組み込まれています。このオブジェクトについては、『Anybus-CompactCom Software Design Guide』で詳しく説明されています。

### サポートされているコマンド

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

### オブジェクトアトリビュート (インスタンス #0)

(詳細については、『Anybus-CompactCom Software Design Guide』を参照してください)

### インスタンスアトリビュート (インスタンス #1)

#### 基本

| #       | 名前                  | アクセス | 種類                                                                  | Value                                                       |
|---------|---------------------|------|---------------------------------------------------------------------|-------------------------------------------------------------|
| 1       | Module type         | Get  | UINT16                                                              | 0401h (標準の Anybus-CompactCom)                               |
| 2 ~ 11  | -                   | -    | -                                                                   | 詳細については、『Anybus-CompactCom Software Design Guide』を参照してください。 |
| 12      | LED colors (LED の色) | Get  | 構造:<br>UINT8(LED1A)<br>UINT8(LED1B)<br>UINT8(LED2A)<br>UINT8(LED2B) | 値:色:<br>01h 緑<br>02h 赤<br>01h 緑<br>02h 赤                    |
| 13 ~ 16 | -                   | -    | -                                                                   | 詳細については、『Anybus-CompactCom Software Design Guide』を参照してください。 |

## 10.3 診断オブジェクト (02h)

### 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、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的な手段を提供します。このオブジェクトについては、『Anybus-CompactCom Software Design Guide』で詳しく説明されています。

### サポートされているコマンド

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 | Value                                                       |
|-------|----------------------|------|--------|-------------------------------------------------------------|
| 1 ~ 4 | -                    | -    | -      | 詳細については、『Anybus-CompactCom Software Design Guide』を参照してください。 |
| 11    | Max no. of instances | Get  | UINT16 | 5+1                                                         |

### インスタンスアトリビュート

#### 拡張

| # | 名前       | アクセス | 種類    | Value                                                       |
|---|----------|------|-------|-------------------------------------------------------------|
| 1 | Severity | Get  | UINT8 | 詳細については、『Anybus-CompactCom Software Design Guide』を参照してください。 |
| 2 | イベントコード  | Get  | UINT8 | Software Design Guide』を参照してください。                            |

#### 高度

| # | 名前               | アクセス | 種類        | Value                                                         |
|---|------------------|------|-----------|---------------------------------------------------------------|
| 3 | NW specific data | Get  | UINT8 の配列 | オプションのネットワーク固有情報。<br>下記も参照してください。<br>- 60 ページ「詳細：ネットワーク固有データ」 |

修復不可能な重大なイベントが発生すると、本モジュールはネットワークから物理的に切断します。これにより、ネットワークへの接続が阻止されます。その他の重大なイベ

ントが発生した場合、イベントコードに応じて、チャンネル診断エントリ/アラーム、または一般の診断エントリ/アラームが発生します。

| イベントコード | 結果                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|---------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 0 ~ FEh | <p>モジュールによりチャンネル診断エントリ/アラームが発行されます。イベントコードは、以下のとおりにチャンネル エラー タイプへ変換/表現されます。</p> <p>コード: イベント (Anybus): チャンネル エラー タイプ (PROFINET):</p> <ul style="list-style-type: none"> <li>10h 一般的なエラー エラー</li> <li>20h 電流 回路ショート</li> <li>21h 電流、装置入力側 回路ショート</li> <li>22h 電流、装置内部 回路ショート</li> <li>23h 電流、装置出力側 回路ショート</li> <li>30h 電圧 過電圧</li> <li>31h 電源電圧 過電圧</li> <li>32h 装置内部の電圧 過電圧</li> <li>33h 出力電圧 過電圧</li> <li>40h 温度 過熱</li> <li>41h 周囲温度 過熱</li> <li>42h 装置温度 過熱</li> <li>50h 装置ハードウェア エラー</li> <li>60h 装置ソフトウェア エラー</li> <li>61h 内部ソフトウェア エラー</li> <li>62h ユーザーソフトウェア エラー</li> <li>63h データセット エラー</li> <li>70h 追加モジュール エラー</li> <li>80h 監視 エラー</li> <li>81h 通信 エラー</li> <li>82h プロトコル エラー エラー</li> <li>90h 外部エラー エラー</li> <li>F0h 追加機能 エラー</li> </ul> |
| FFh     | <p>ネットワーク固有データに基づいて、モジュールにより一般的な診断エントリ/アラームが発行されます。</p> <p>下記も参照してください。<br/>- 60 ページ「詳細: ネットワーク固有データ :」</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |

## 詳細：ネットワーク固有データ：

| オフセット  | 内容                                                                                                                                                                              |
|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 0      | API、下位ワード、下位バイト                                                                                                                                                                 |
| 1      | API、下位ワード、上位バイト                                                                                                                                                                 |
| 2      | API、上位ワード、下位バイト                                                                                                                                                                 |
| 3      | API、上位ワード、上位バイト                                                                                                                                                                 |
| 4      | スロット番号、下位バイト                                                                                                                                                                    |
| 5      | スロット番号、上位バイト                                                                                                                                                                    |
| 6      | サブスロット番号、下位バイト                                                                                                                                                                  |
| 7      | サブスロット番号、上位バイト                                                                                                                                                                  |
| 8      | チャンネル番号、下位バイト                                                                                                                                                                   |
| 9      | チャンネル番号、上位バイト                                                                                                                                                                   |
| 10     | チャンネル特性、データタイプ：<br><u>値：意味：</u><br>00h チャンネル番号が 8000h の場合、または、データタイプが以下のいずれでもない場合<br>01h 1 bit<br>02h 2 bit<br>03h 4 bit<br>04h 8 bit<br>05h 16 bit<br>06h 32 bit<br>07h 64 bit |
| 11     | チャンネル特性、方向：<br><u>値：意味：</u><br>00h メーカー固有<br>01h 入力<br>02h 出力<br>03h 入出力                                                                                                        |
| 12     | UserStructIdent、下位バイト                                                                                                                                                           |
| 13     | UserStructIdent、上位バイト                                                                                                                                                           |
| 14 ~ n | データ                                                                                                                                                                             |

## オブジェクト固有のエラーコード

| コード | エラー                                 |
|-----|-------------------------------------|
| 03h | API が存在しない                          |
| 04h | 指定したスロットにモジュールが挿入されていない             |
| 05h | 指定したスロットにサブモジュールが挿入されていない           |
| 06h | 指定したスロット番号が範囲外                      |
| 07h | 指定したサブスロット番号が範囲外                    |
| 08h | PROFINET IO スタックにチャンネル診断エントリを追加できない |
| 09h | PROFINET IO スタックにチャンネル診断アラームを追加できない |
| 0Ah | チャンネル番号が範囲外                         |
| 0Bh | ChannelPropType が範囲外                |
| 0Ch | ChannelPropDir が範囲外                 |
| 0Dh | ChannelPropAcc が範囲外                 |
| 0Eh | ChannelPropMaintReq が範囲外            |
| 0Fh | ChannelPropMaintDem が範囲外            |
| 10h | UserStructIdent が範囲外                |
| 11h | ChannelErrType が範囲外                 |
| FFh | 不明なエラー                              |

## 10.4 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

このオブジェクトに関する詳細は、『Anybus-CompactCom Software Design Guide』を参照してください。

### サポートされているコマンド

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

### オブジェクトアトリビュート (インスタンス #0)

(詳細については、『Anybus-CompactCom Software Design Guide』を参照してください)

### インスタンスアトリビュート (インスタンス #1)

#### 基本

| #      | 名前                      | アクセス | 種類       | Value                                                                                           |
|--------|-------------------------|------|----------|-------------------------------------------------------------------------------------------------|
| 1      | Network type            | Get  | UINT16   | 0096h                                                                                           |
| 2      | Network type string     | Get  | CHAR の配列 | "PROFINET IO 2-Port"                                                                            |
| 3      | Data format             | Get  | ENUM     | 01h (MSB 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 (単位: バイト)<br>Map_ADI_Write_Area が成功するたびに更新される <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 (単位: バイト)<br>Map_ADI_Read_Area が成功するたびに更新される <sup>a</sup>                     |
| 7      | Exception Information   | Get  | UINT8    | 本モジュールが EXCEPTION 状態になったとき、追加情報がここに設定されます。<br>値: 意味:<br>01h 不正な値<br>02h 間違ったデータサイズ<br>03h 不正な応答 |
| 8 ~ 10 | -                       | -    | -        | 詳細については、『Anybus-CompactCom Software Design Guide』を参照してください。                                     |

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

## 10.5 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

このオブジェクトには、エンドユーザーが設定するネットワーク固有の設定パラメーターが格納されます。このオブジェクトにリセットコマンド(工場出荷状態へのリセット)が発行されると、すべてのインスタンスがデフォルト値になります。

下記も参照してください。

- 12 ページ「通信設定」
- 15 ページ「E-mail クライアント」

**注:** 起動時にホストアプリケーションが以下のインスタンスを設定できるようにすると、適合性試験に合格しなくなる可能性があります。

### サポートされているコマンド

オブジェクト:    Get\_Attribute  
                  Reset

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

### オブジェクトアトリビュート (インスタンス #0)

(詳細については、『Anybus-CompactCom Software Design Guide』を参照してください)

### インスタンスアトリビュート (インスタンス #3、IP アドレス)

値はモジュールがリセットされた後に使用されます。

#### 高度

| # | 名前                 | アクセス    | 種類        | 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 つの要素)                                      |
| 4 | Descriptor         | Get     | UINT8     | 07h (読み出し / 書き込み / 共有アクセス)                        |
| 5 | Value              | Get/Set | UINT8 の配列 | 有効範囲 :0.0.0.0. ~ 255.255.255.255 (デフォルト =0.0.0.0) |

a. Multilingual (多言語)。69 ページ「多言語文字列」を参照。

**注:** 認定に関する問題が生じるため、ホストアプリケーションはパワーオン時にこのアトリビュートを設定すべきではありません。

## インスタンスアトリビュート ( インスタンス #4、サブネット マスク )

値はモジュールがリセットされた後に使用されます。

高度

| # | 名前                 | アクセス    | 種類        | 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 つの要素)                                       |
| 4 | Descriptor         | Get     | UINT8     | 07h ( 読み出し / 書き込み / 共有アクセス )                       |
| 5 | Value              | Get/Set | UINT8 の配列 | 有効範囲 :0.0.0.0. ~ 255.255.255.255 ( デフォルト =0.0.0.0) |

a. Multilingual ( 多言語 )。69 ページ「多言語文字列」を参照。

注 : 認定に関する問題が生じるため、ホストアプリケーションはパワーオン時にこのアトリビュートを設定すべきではありません。

## インスタンスアトリビュート ( インスタンス #5、ゲートウェイ )

値はモジュールがリセットされた後に使用されます。

高度

| # | 名前                 | アクセス    | 種類        | 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 つの要素)                                       |
| 4 | Descriptor         | Get     | UINT8     | 07h ( 読み出し / 書き込み / 共有アクセス )                       |
| 5 | Value              | Get/Set | UINT8 の配列 | 有効範囲 :0.0.0.0. ~ 255.255.255.255 ( デフォルト =0.0.0.0) |

a. Multilingual ( 多言語 )。69 ページ「多言語文字列」を参照。

注 : 認定に関する問題が生じるため、ホストアプリケーションはパワーオン時にこのアトリビュートを設定すべきではありません。

## インスタンスアトリビュート (インスタンス #6、DHCP)

値はモジュールがリセットされた後に使用されます。

### 拡張

| # | 名前                 | アクセス    | 種類       | 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 (要素数 1)                                                             |
| 4 | Descriptor         | Get     | UINT8    | 07h (読み出し / 書き込み / 共有アクセス)                                              |
| 5 | Value <sup>a</sup> | Get/Set | ENUM     | 値:Enum 文字列:意味:<br>00h "Disable" DHCP 無効 (デフォルト)<br>01h "Enable" DHCP 有効 |

a. Multilingual (多言語)。69 ページ「多言語文字列」を参照。

注:DHCP をオンにするようエンドユーザーが明示的に指定しない限り、この設定にしないでください。通常、PROFINET IO コントローラーが IP アドレスを割り当てます。

## インスタンスアトリビュート (インスタンス #8、DNS1)

### 高度

このインスタンスには、プライマリ DNS サーバーのアドレスが格納されます。設定の変更はリセット後に有効になります。

| # | 名前                 | アクセス    | 種類        | 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 (読み出し / 書き込み / 共有アクセス)                      |
| 5 | Value              | Get/Set | UINT8 の配列 | 有効範囲:0.0.0.0. ~ 255.255.255.255 (デフォルト=0.0.0.0) |

a. Multilingual (多言語)。69 ページ「多言語文字列」を参照。

## インスタンスアトリビュート (インスタンス #9、DNS2)

このインスタンスには、セカンダリ DNS サーバーのアドレスが格納されます。設定の変更はリセット後に有効になります。

### 高度

| # | 名前                 | アクセス    | 種類        | 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 (読み出し / 書き込み / 共有アクセス)                      |
| 5 | Value              | Get/Set | UINT8 の配列 | 有効範囲:0.0.0.0. ~ 255.255.255.255 (デフォルト=0.0.0.0) |

a. Multilingual (多言語)。69 ページ「多言語文字列」を参照。

## インスタンスアトリビュート (インスタンス #10、ホスト名)

このインスタンスには、モジュールのホスト名が格納されます。設定の変更はリセット後に有効になります。

### 高度

| # | 名前                 | アクセス    | 種類       | 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 (読み出し / 書き込み / 共有アクセス)   |
| 5 | Value              | Get/Set | CHAR の配列 | ホスト名。64 文字 (最大長までスペースで埋められる) |

a. Multilingual (多言語)。69 ページ「多言語文字列」を参照。

## インスタンスアトリビュート (インスタンス #11、ドメイン名)

このインスタンスにはドメイン名が格納されます。設定の変更はリセット後に有効になります。

### 高度

| # | 名前                 | アクセス    | 種類       | 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    | 30h                           |
| 4 | Descriptor         | Get     | UINT8    | 07h (読み出し / 書き込み / 共有アクセス)    |
| 5 | Value              | Get/Set | CHAR の配列 | ドメイン名。48 文字 (最大長までスペースで埋められる) |

a. Multilingual (多言語)。69 ページ「多言語文字列」を参照。

## インスタンスアトリビュート (インスタンス #12、SMTP サーバー)

このインスタンスには、SMTP サーバーのアドレスが格納されます。設定の変更はリセット後に有効になります。

### 高度

| # | 名前                 | アクセス    | 種類        | 説明                                     |
|---|--------------------|---------|-----------|----------------------------------------|
| 1 | Name <sup>a</sup>  | Get     | CHAR の配列  | "SMTP Server"                          |
| 2 | Data type          | Get     | UINT8     | 07h (= CHAR)                           |
| 3 | Number of elements | Get     | UINT8     | 40h                                    |
| 4 | Descriptor         | Get     | UINT8     | 07h (読み出し / 書き込み / 共有アクセス)             |
| 5 | Value              | Get/Set | UINT8 の配列 | SMTP サーバーのアドレス。64 文字 (最大長までスペースで埋められる) |

a. Multilingual (多言語)。69 ページ「多言語文字列」を参照。

## インスタンスアトリビュート (インスタンス #13、SMTP ユーザー)

このインスタンスには、SMTP アカウントのユーザー名が格納されます。設定の変更はリセット後に有効になります。

### 高度

| # | 名前                 | アクセス    | 種類        | 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 (読み出し / 書き込み / 共有アクセス)               |
| 5 | Value              | Get/Set | UINT8 の配列 | SMTP アカウントのユーザー名。64 文字 (最大長までスペースで埋められる) |

a. Multilingual (多言語)。69 ページ「多言語文字列」を参照。

## インスタンスアトリビュート (インスタンス #14、SMTP パスワード)

このインスタンスには、SMTP アカウントのパスワードが格納されます。設定の変更はリセット後に有効になります。

### 高度

| # | 名前                 | アクセス    | 種類        | 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 (読み出し / 書き込み / 共有アクセス)               |
| 5 | Value              | Get/Set | UINT8 の配列 | SMTP アカウントのパスワード。64 文字 (最大長までスペースで埋められる) |

a. Multilingual (多言語)。69 ページ「多言語文字列」を参照。

## インスタンスアトリビュート (インスタンス #15、ステーション名)

ステーション名は、PROFINET 上にある Anybus モジュールの識別に使用されます。ランタイム中にホストアプリケーションによりこの値が変更された場合、変更内容を反映するにはモジュールをリセットする必要があります。ただし、DCP による変更は直ちに反映されます。

### 拡張

| # | 名前                 | アクセス    | 種類        | 説明                         |
|---|--------------------|---------|-----------|----------------------------|
| 1 | Name <sup>a</sup>  | Get     | CHAR の配列  | "Station name"             |
| 2 | Data type          | Get     | UINT8     | 07h (= CHAR)               |
| 3 | Number of elements | Get/Set | UINT8     | FEh                        |
| 4 | Descriptor         | Get     | UINT8     | 07h (読み出し / 書き込み / 共有アクセス) |
| 5 | Value              | Get/Set | UINT8 の配列 | ステーション名                    |

a. Multilingual (多言語)。69 ページ「多言語文字列」を参照。

**注 1:** 通常、このアトリビュートをアプリケーションで設定しないでください。通常、ステーション名はネットワークを介してエンドユーザーにより設定されます。エンドユー

ザーがアプリケーションを用いてステーション名を変更可能であり、なおかつエンドユーザーがその方法を選択した場合は、ホストアプリケーションがこのアトリビュートを使用する必要があります。

**注 2:** 認定に関する問題が生じるため、ホストアプリケーションはパワーオン時にこのアトリビュートを設定すべきでは有りません。

## インスタンスアトリビュート (インスタンス #16、Function Tag)

このインスタンスには、スロット 0、サブスロット 1 の I&M1 パラメーターである "Function Tag" が格納されます。この値は、ホストアプリケーションとネットワークのいずれから指定でき、不揮発性メモリに保存されます。

### 高度

| # | 名前                 | アクセス    | 種類        | 説明                                              |
|---|--------------------|---------|-----------|-------------------------------------------------|
| 1 | Name <sup>a</sup>  | Get     | CHAR の配列  | "Function tag"                                  |
| 2 | Data type          | Get     | UINT8     | 07h (= CHAR)                                    |
| 3 | Number of elements | Get     | UINT8     | 20h                                             |
| 4 | Descriptor         | Get     | UINT8     | 07h (読み出し / 書き込み / 共有アクセス)                      |
| 5 | Value              | Get/Set | UINT8 の配列 | 装置の機能やタスクを説明する文字列 (最大長までスペースで埋められる。デフォルト = " ") |

a. Multilingual (多言語)。69 ページ「多言語文字列」を参照。

**注:** 認定に関する問題が生じるため、ホストアプリケーションはパワーオン時にこのアトリビュートを設定すべきでは有りません。

## インスタンスアトリビュート (インスタンス #17、Location Tag)

このインスタンスには、スロット 0、スロット 1 の I&M1 パラメーターである "Location Tag" が格納されます。この値は、ホストアプリケーションとネットワークのいずれから指定でき、不揮発性メモリに保存されます。

### 高度

| # | 名前                 | アクセス    | 種類        | 説明                                          |
|---|--------------------|---------|-----------|---------------------------------------------|
| 1 | Name <sup>a</sup>  | Get     | CHAR の配列  | "Location tag"                              |
| 2 | Data type          | Get     | UINT8     | 07h (= CHAR)                                |
| 3 | Number of elements | Get     | UINT8     | 16h                                         |
| 4 | Descriptor         | Get     | UINT8     | 07h (読み出し / 書き込み / 共有アクセス)                  |
| 5 | Value              | Get/Set | UINT8 の配列 | 装置の場所を説明する文字列 (最大長までスペースで埋められる。デフォルト = " ") |

a. Multilingual (多言語)。69 ページ「多言語文字列」を参照。

**注:** 認定に関する問題が生じるため、ホストアプリケーションはパワーオン時にこのアトリビュートを設定すべきでは有りません。

## インスタンスアトリビュート (インスタンス #18、Installation Date)

このインスタンスには、スロット 0、スロット 1 の I&M2 パラメーターである "Installation Date" が格納されます。この値は、ホストアプリケーションとネットワークのいずれからも指定でき、不揮発性メモリに保存されます。

### 高度

| # | 名前                 | アクセス    | 種類        | 説明                                                                                    |
|---|--------------------|---------|-----------|---------------------------------------------------------------------------------------|
| 1 | Name <sup>a</sup>  | Get     | CHAR の配列  | "Install. date"                                                                       |
| 2 | Data type          | Get     | UINT8     | 07h (= CHAR)                                                                          |
| 3 | Number of elements | Get     | UINT8     | 10h                                                                                   |
| 4 | Descriptor         | Get     | UINT8     | 07h (読み出し / 書き込み / 共有アクセス)                                                            |
| 5 | Value              | Get/Set | UINT8 の配列 | "YYYY-MM-DD hh:mm" の形式の文字列<br>例 : "2007-06-15 22:00"<br>(最大長までスペースで埋められる。デフォルト = " ") |

a. Multilingual (多言語)。69 ページ「多言語文字列」を参照。

注：認定に関する問題が生じるため、ホストアプリケーションはパワーオン時にこのアトリビュートを設定すべきではありません。

## インスタンスアトリビュート (インスタンス #19、Description)

このインスタンスには、スロット 0、スロット 1 の I&M3 パラメーターである "Description" が格納されます。この値は、ホストアプリケーションとネットワークのいずれからも指定でき、不揮発性メモリに保存されます。

### 高度

| # | 名前                 | アクセス    | 種類        | 説明                                                          |
|---|--------------------|---------|-----------|-------------------------------------------------------------|
| 1 | Name <sup>a</sup>  | Get     | CHAR の配列  | "Description"                                               |
| 2 | Data type          | Get     | UINT8     | 07h (= CHAR)                                                |
| 3 | Number of elements | Get     | UINT8     | 36h                                                         |
| 4 | Descriptor         | Get     | UINT8     | 07h (読み出し / 書き込み / 共有アクセス)                                  |
| 5 | Value              | Get/Set | UINT8 の配列 | 装置を説明する文字列。エンドユーザーが自由に使用可能<br>(最大長までスペースで埋められる。デフォルト = " ") |

a. Multilingual (多言語)。69 ページ「多言語文字列」を参照。

注：認定に関する問題が生じるため、ホストアプリケーションはパワーオン時にこのアトリビュートを設定すべきではありません。

## 多言語文字列

このオブジェクトのインスタンス名と列挙文字列は複数の言語に対応しており、現在の言語設定に応じて以下のように変換されます。

| 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      |
| 8       | DNS1          | DNS 1        | DNS Primaria  | DNS1          | DNS1           |
| 9       | DNS2          | DNS 2        | DNS Secundia. | DNS2          | DNS2           |
| 10      | Host name     | Host name    | Nombre Host   | Nome Host     | Nom hôte       |
| 11      | Domain name   | Domain name  | Nobre Domain  | Nome Dominio  | Dom Domaine    |
| 12      | SMTP Server   | SMTP Server  | Servidor SMTP | Server SMTP   | SMTP serveurur |
| 13      | SMTP User     | SMTP User    | Usuario SMTP  | Utente SMTP   | SMTP utilizza. |
| 14      | SMTP Pswd     | SMTP PSWD    | Clave SMTP    | Password SMTP | SMTP mt passe  |
| 15      | Station name  | Stationsname | Nom.Estacion  | Nome Stazione | Nom Station    |
| 16      | Function tag  | Funktion     | Tag Funcion   | Descr.Funz.   | Var. Fonction  |
| 17      | Location tag  | Position     | Tag Locacion  | Descr.Locaz.  | Var. Location  |
| 18      | Install. date | Install.Tag  | Fecha Instal. | Data Install. | Date Install.  |
| 19      | Description   | Beschreibung | Descripcion   | Descrizione   | Description    |

## 10.6 ネットワーク PROFINET IO オブジェクト (0Eh)

### 概要

拡張、高度

### オブジェクトの説明

-

### サポートされているコマンド

オブジェクト：   Get\_Attribute  
                   Plug\_Module (72 ページ「コマンド詳細 :Plug\_Module」を参照)  
                   Plug\_Submodule (73 ページ「コマンド詳細 :Plug\_Submodule」を参照)  
                   Pull\_Module (75 ページ「コマンド詳細 :Pull\_Module」を参照)  
                   Pull\_Submodule (76 ページ「コマンド詳細 :Pull\_Submodule」を参照)  
                   API\_Add (77 ページ「コマンド詳細 :API\_Add」を参照)  
                   Appl\_State\_Ready (78 ページ「コマンド詳細 :Appl\_State\_Ready」を参照)  
                   AR\_Abort (79 ページ「コマンド詳細 :AR\_Abort」を参照)

インスタンス：   Get\_Attribute

### オブジェクトアトリビュート (インスタンス #0)

| # | 名前                   | アクセス | データタイプ   | Value                 |
|---|----------------------|------|----------|-----------------------|
| 1 | Name                 | Get  | CHAR の配列 | "Network PROFINET IO" |
| 2 | Revision             | Get  | UINT8    | 01h                   |
| 3 | Number of instances  | Get  | UINT16   | 0001h                 |
| 4 | Highest instance no. | Get  | UINT16   | 0001h                 |

## インスタンスアトリビュート (インスタンス #1)

### 拡張

| #  | 名前                             | アクセス | 種類         | Value                                                                                             |
|----|--------------------------------|------|------------|---------------------------------------------------------------------------------------------------|
| 1  | OnLineTrans                    | Get  | UINT32     | 診断カウンター。オンラインの遷移数を追跡                                                                              |
| 2  | OffLineTrans                   | Get  | UINT32     | 診断カウンター。オフラインの遷移数を追跡                                                                              |
| 3  | Last OffLineInd ReasonCode     | Get  | UINT16     | 最後に指示されたオフラインに関する理由コード<br><br>下記も参照してください。<br>- 141 ページ「コマンド詳細 :AR_Offline_Ind」                   |
| 4  | Last AbortInd ReasonCode       | Get  | UINT16     | 最後に指示されたアボートに関する理由コード<br><br>下記も参照してください。<br>- 141 ページ「コマンド詳細 :AR_Offline_Ind」                    |
| 5  | AddedApis                      | Get  | UINT16     | 追加された API の数を返す (API 0 を含む)                                                                       |
| 6  | ApiList                        | Get  | UINT32 の配列 | 最初の要素は常にゼロで、2 番目の要素に追加の API が格納されます。配列の長さは、"AddedApis" パラメーターにより決定されます。                           |
| 7  | EstablishedArs                 | Get  | UINT16     | 現在確立されているアプリケーションリレーションシップの数                                                                      |
| 8  | ArList                         | Get  | UINT16 の配列 | アプリケーションリレーションシップハンドルの配列です。配列の長さは、"EstablishedArs" パラメーターにより決定されます。                               |
| 9  | ProfinetIoStack Init ErrorCode | Get  | UINT16     | 何らかの理由により PROFINET IO スタックが現在の設定を認識しない場合、返信されたエラーコードをこのアトリビュートで読み込むことができます。                       |
| 10 | Port 1 MAC Address             | Get  | UINT8 の配列  | 6 Byte で表される PROFINET ポート 1 の MAC アドレス<br><br>下記も参照してください。<br>- 147 ページ「Ethernet ホストオブジェクト (F9h)」 |
| 11 | Port 2 MAC Address             | Get  | UINT8 の配列  | 6 Byte で表される PROFINET ポート 2 の MAC アドレス<br><br>下記も参照してください。<br>- 147 ページ「Ethernet ホストオブジェクト (F9h)」 |

## コマンド詳細 :Plug\_Module

### 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

このコマンドは、起動時に Real Identification を指定するために呼び出されます。また、Real Identification が変更された場合、ランタイム中にもこのコマンドが呼び出されます。この場合、Anybus は IO コントローラーに対して自動的に "Plug" アラームを発行します。

注：このコマンドは、先に "API\_Add" が発行されている場合のみ発行できます。

- コマンド詳細

| フィールド     | 内容                       | コメント                             |
|-----------|--------------------------|----------------------------------|
| CmdExt[0] | ( 予約。ゼロに設定 )             | ( 将来のために予約 )                     |
| CmdExt[1] |                          |                                  |
| Data[0]   | API ( 下位ワード、下位バイト )      | アプリケーションプロセスインスタンス (API)         |
| Data[1]   | API ( 下位ワード、上位バイト )      |                                  |
| Data[2]   | API ( 上位ワード、下位バイト )      |                                  |
| Data[3]   | API ( 上位ワード、上位バイト )      |                                  |
| Data[4]   | SlotNr ( 下位バイト )         | モジュールの宛先スロット                     |
| Data[5]   | SlotNr ( 上位バイト )         | 範囲：<br>0 ~ N(API_Add にて定義)       |
| Data[6]   | ModIdent ( 下位ワード、下位バイト ) | モジュールは GSD ファイルに記述されているとおりに認識される |
| Data[7]   | ModIdent ( 下位ワード、上位バイト ) |                                  |
| Data[8]   | ModIdent ( 上位ワード、下位バイト ) |                                  |
| Data[9]   | ModIdent ( 上位ワード、上位バイト ) |                                  |

- 応答詳細

80 ページ「オブジェクト固有のエラーコード」を参照してください。

下記も参照してください。

- 19 ページ「Real Identification (RI)」(20 ページ「設定の不一致」)
- 73 ページ「コマンド詳細 :Plug\_Submodule」
- 75 ページ「コマンド詳細 :Pull\_Module」
- 76 ページ「コマンド詳細 :Pull\_Submodule」
- 77 ページ「コマンド詳細 :API\_Add」
- 161 ページ「フローチャート - Real Identification (RI) の確立」

## コマンド詳細 :Plug\_Submodule

### 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

このコマンドは、起動時に **Real Identification** を指定するために呼び出されます。また、**Real Identification** が変更された場合、ランタイム中にもこのコマンドが呼び出されます。この場合、Anybus は IO コントローラーに対して自動的に "Return of Submodule" アラームを発行します。

このコマンドに埋め込まれているサブモジュールは、マスターへの IO データ、マスターからの IO データ、または双方向のデータを持つことができます。また、データを全く持たないサブモジュールも埋め込むことができます。

Anybus は、最大 128 個のサブモジュールをサポートします。

**注 1:** コマンドにおけるスロット番号を 0 (ゼロ) に設定した場合、スロット 0 は実際のデータを持っていないため、ADI 番号も 0 (ゼロ) に設定する必要があります。

**注 2:** このコマンドは、先に "API\_Add" が発行されている場合のみ発行できます。

**注 3:** 認定試験に合格するには、"インターフェイス" サブモジュールおよび "ポート" サブモジュールを埋め込む必要があります。

**注 4:** インターフェイスサブモジュールおよびポートサブモジュールは、SETUP 状態でのみ埋め込むことができます。これらのサブモジュールをランタイム中に埋め込もうとすると、エラーが発生します。

- コマンド詳細

| フィールド     | 内容                        | コメント                                                                                                                                                   |
|-----------|---------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------|
| CmdExt[0] | (予約。ゼロに設定)                | (将来のために予約)                                                                                                                                             |
| CmdExt[1] |                           |                                                                                                                                                        |
| Data[0]   | ADI 番号 (下位バイト)、読み取り       | Map_ADI_Read_Area にマッピングされた ADI への参照。                                                                                                                  |
| Data[1]   | ADI 番号 (上位バイト)、読み取り       |                                                                                                                                                        |
| Data[2]   | ADI 要素、読み取り               | 指定した SlotNr の Map_ADI_Read_Area にマッピングされた ADI の要素への参照 (Data[10 ~ 11] を参照)。<br><br>範囲：<br>1 ~ 255 サブモジュールに割り当てられた ADI 要素<br>0 ADI 全体がサブモジュールに割り当てられている  |
| Data[3]   | ADI 番号 (下位バイト)、書き込み       | Map_ADI_Write_Area にマッピングされた ADI への参照。                                                                                                                 |
| Data[4]   | ADI 番号 (上位バイト)、書き込み       |                                                                                                                                                        |
| Data[5]   | ADI 要素、読み取り               | 指定した SlotNr の Map_ADI_Write_Area にマッピングされた ADI の要素への参照 (Data[10 ~ 11] を参照)。<br><br>範囲：<br>1 ~ 255 サブモジュールに割り当てられた ADI 要素<br>0 ADI 全体がサブモジュールに割り当てられている |
| Data[6]   | API (下位ワード、下位バイト)         | アプリケーションプロセスインスタンス (API)                                                                                                                               |
| Data[7]   | API (下位ワード、上位バイト)         |                                                                                                                                                        |
| Data[8]   | API (上位ワード、下位バイト)         | 下記も参照してください。                                                                                                                                           |
| Data[9]   | API (上位ワード、上位バイト)         | - 18 ページ「アプリケーションプロセスインスタンス (API)」                                                                                                                     |
| Data[10]  | SlotNr (下位バイト)            | サブモジュールの宛先スロット                                                                                                                                         |
| Data[11]  | SlotNr (上位バイト)            | 範囲：<br>0 ~ N(API_Add にて定義)                                                                                                                             |
| Data[12]  | SubSlotNr (下位バイト)         | サブモジュールの宛先サブスロット                                                                                                                                       |
| Data[13]  | SubSlotNr (上位バイト)         | 範囲：<br>0 ~ P(API_Add にて定義)<br>8000 ~ 8002h (スロット 0 のみ)                                                                                                 |
| Data[14]  | SubModIdent (下位ワード、下位バイト) | モジュールは GSD ファイルに記述されているとおりに認識される                                                                                                                       |
| Data[15]  | SubModIdent (下位ワード、上位バイト) |                                                                                                                                                        |
| Data[16]  | SubModIdent (上位ワード、下位バイト) |                                                                                                                                                        |
| Data[17]  | SubModIdent (上位ワード、上位バイト) |                                                                                                                                                        |

- 応答詳細

80 ページ「オブジェクト固有のエラーコード」を参照してください。

下記も参照してください。

- 19 ページ「Real Identification (RI)」(20 ページ「設定の不一致」)
- 145 ページ「コマンド詳細 :Plug\_Submodule\_Failed」

- 72 ページ「コマンド詳細 :Plug\_Module」
- 75 ページ「コマンド詳細 :Pull\_Module」
- 76 ページ「コマンド詳細 :Pull\_Submodule」
- 77 ページ「コマンド詳細 :API\_Add」
- 161 ページ「フローチャート - Real Identification (RI) の確立」

## コマンド詳細 :Pull\_Module

### 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

このコマンドは、コンフィグレーションからモジュールを削除します。いつでも発行可能です。

- コマンド詳細

| フィールド     | 内容                | コメント                                               |
|-----------|-------------------|----------------------------------------------------|
| CmdExt[0] | (予約。ゼロに設定)        | (将来のために予約)                                         |
| CmdExt[1] |                   |                                                    |
| Data[0]   | API (下位ワード、下位バイト) | アプリケーションプロセスインスタンス (API)                           |
| Data[1]   | API (下位ワード、上位バイト) |                                                    |
| Data[2]   | API (上位ワード、下位バイト) | 下記も参照してください。<br>- 18 ページ「アプリケーションプロセスインスタンス (API)」 |
| Data[3]   | API (上位ワード、上位バイト) |                                                    |
| Data[4]   | SlotNr (下位バイト)    | モジュールのスロット番号。                                      |
| Data[5]   | SlotNr (上位バイト)    | 範囲:<br>0 ~ N(API_Add にて定義)                         |

- 応答詳細  
80 ページ「オブジェクト固有のエラーコード」を参照してください。

下記も参照してください。

- 72 ページ「コマンド詳細 :Plug\_Module」
- 76 ページ「コマンド詳細 :Pull\_Submodule」

## コマンド詳細 :Pull\_Submodule

### カテゴリ

高度

### 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

このコマンドは、コンフィグレーションからサブモジュールを削除します。いつでも発行可能です。

- コマンド詳細

| フィールド     | 内容                | コメント                                                  |
|-----------|-------------------|-------------------------------------------------------|
| CmdExt[0] | (予約。ゼロに設定)        | (将来のために予約)                                            |
| CmdExt[1] |                   |                                                       |
| Data[0]   | API (下位ワード、下位バイト) | アプリケーションプロセスインスタンス (API)                              |
| Data[1]   | API (下位ワード、上位バイト) | 下記も参照してください。<br>- 18 ページ「アプリケーションプロセスインスタンス (API)」    |
| Data[2]   | API (上位ワード、下位バイト) |                                                       |
| Data[3]   | API (上位ワード、上位バイト) |                                                       |
| Data[4]   | SlotNr (下位バイト)    | サブモジュールのスロット番号。                                       |
| Data[5]   | SlotNr (上位バイト)    | 範囲：<br>0 ~ N(API_Add にて定義)                            |
| Data[6]   | SubSlotNr (下位バイト) | サブモジュールのサブスロット番号。                                     |
| Data[7]   | SubSlotNr (上位バイト) | 範囲：<br>0 ~ P(API_Add にて定義)<br>8000 ~ 8002h(スロット 0 のみ) |

- 応答詳細

80 ページ「オブジェクト固有のエラーコード」を参照してください。

下記も参照してください。

- 73 ページ「コマンド詳細 :Plug\_Submodule」
- 75 ページ「コマンド詳細 :Pull\_Module」

## コマンド詳細 :API\_Add

### カテゴリ

高度

### 詳細

コマンドコード: 14h

有効な対象: オブジェクトインスタンス

### 説明

デフォルトでは、本モジュールは API 0 (ゼロ) のみサポートします。追加の API がサポートされている場合、または、ホストアプリケーションがモジュールやサブモジュールの埋め込み/取り外しを行う必要がある場合、このコマンドを使用して API の実装を指定する必要があります。なお、このコマンドを使用する場合は、他の API の定義や、モジュール/サブモジュールの埋め込み/取り外しを行う前に、API 0 (ゼロ) を宣言する必要があります。API 番号は、(PROFIBUS および PROFINET International (PI)) により割り当てられます。

**注 1:** このコマンドは、Anybus オブジェクトの "Setup Complete" アトリビュートを設定する前にのみ発行できます。

**注 2:** このコマンドは、ADI をプロセスデータにマッピング中に Anybus モジュールにより作成されたデフォルトの Real Identification をクリアします。そのため、このコマンドを発行する場合、"Plug\_Module" コマンドおよび "Plug\_Submodule" コマンドを使用して実際の Real Identification を指定する必要があります。

**注 3:** このコマンドを発行した場合、ADI から プロセスデータへのマッピングはエラー ("Invalid State") になります。

- コマンド詳細

| フィールド     | 内容                  | コメント                                               |
|-----------|---------------------|----------------------------------------------------|
| CmdExt[0] | (予約。ゼロに設定)          | (将来のために予約)                                         |
| CmdExt[1] |                     |                                                    |
| Data[0]   | API (下位ワード、下位バイト)   | アプリケーションプロセスインスタンス (API)                           |
| Data[1]   | API (下位ワード、上位バイト)   | 下記も参照してください。<br>- 18 ページ「アプリケーションプロセスインスタンス (API)」 |
| Data[2]   | API (上位ワード、下位バイト)   |                                                    |
| Data[3]   | API (上位ワード、上位バイト)   |                                                    |
| Data[4]   | MaxSlots (下位バイト)    | API の最大スロット数 (MNS)。                                |
| Data[5]   | MaxSlots (上位バイト)    | 範囲:<br>0 ~ 64                                      |
| Data[6]   | MaxSubSlots (下位バイト) | API のスロット当たりの最大サブスロット数。                            |
| Data[7]   | MaxSubSlots (上位バイト) | 範囲:<br>0 ~ 8                                       |

- 応答詳細

80 ページ「オブジェクト固有のエラーコード」を参照してください。

下記も参照してください。

- 18 ページ「アプリケーションプロセスインスタンス (API)」
- 161 ページ「フローチャート - Real Identification (RI) の確立」

## コマンド詳細 :Appl\_State\_Ready

### カテゴリ

高度

### 詳細

コマンドコード: 15h

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、ホストアプリケーションが "End\_Of\_Prm\_Ind" をサポートし、ホストアプリケーションがデータ交換可能なモジュール (およびその I/O コントローラー) への信号をサポートする場合のみ適用できます。

- コマンド詳細

| フィールド     | 内容                | コメント                                                                  |
|-----------|-------------------|-----------------------------------------------------------------------|
| CmdExt[0] | AR handle (下位バイト) | アプリケーションリレーションシップ (AR) のハンドル<br>(18 ページ「アプリケーションプロセスインスタンス (API)」を参照) |
| CmdExt[1] | AR handle (上位バイト) |                                                                       |

- 応答詳細  
80 ページ「オブジェクト固有のエラーコード」を参照してください。

下記も参照してください。

- 18 ページ「アプリケーションリレーションシップ (AR)」
- 140 ページ「コマンド詳細 :End\_Of\_Prm\_Ind」

## コマンド詳細 :AR\_Abort

### カテゴリ

高度

### 詳細

コマンドコード: 16h

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、ABCC-PRT に対し、現在のアプリケーションリレーションシップを中断してオフラインにするよう指示します。

- コマンド詳細

| フィールド     | 内容                | コメント                                                                          |
|-----------|-------------------|-------------------------------------------------------------------------------|
| CmdExt[0] | AR handle (下位バイト) | アプリケーションリレーションシップ (AR) の<br>ハンドル<br>(18 ページ「アプリケーションプロセスイン<br>スタンス (API)」を参照) |
| CmdExt[1] | AR handle (上位バイト) |                                                                               |

- 応答詳細

80 ページ「オブジェクト固有のエラー コード」を参照してください。

下記も参照してください。

- 18 ページ「アプリケーションリレーションシップ (AR)」
- 135 ページ「コマンド詳細 :AR\_Check\_Ind」
- 138 ページ「コマンド詳細 :AR\_Info\_Ind」
- 141 ページ「コマンド詳細 :AR\_Offline\_Ind」
- 143 ページ「コマンド詳細 :AR\_Abort\_Ind」

## オブジェクト固有のエラーコード

| コード | 意味                                                                    |
|-----|-----------------------------------------------------------------------|
| 01h | Map_ADI_Write_Area により ADI がマッピングされていない                               |
| 02h | Map_ADI_Read_Area により ADI がマッピングされていない                                |
| 03h | ADI の要素が存在しない                                                         |
| 04h | この ADI/ 要素は既にマッピングされている                                               |
| 05h | 始めに API 0 を追加する必要がある                                                  |
| 06h | API が存在しない                                                            |
| 07h | 既に存在する API を追加しようとしている                                                |
| 08h | API を追加するための空きがない                                                     |
| 09h | スロット 0 のモジュールは IO データを持ってない                                           |
| 0Ah | 要求されたモジュール/サブモジュールを埋め込む前に、スロット 0 (のサブスロット 1) にモジュール/サブモジュールを設定する必要がある |
| 0Bh | スロットが占有されている                                                          |
| 0Ch | サブスロットが占有されている                                                        |
| 0Dh | 指定したスロットにモジュールが挿入されていない                                               |
| 0Eh | 指定したスロットにサブモジュールが挿入されていない                                             |
| 0Fh | 指定したスロット番号が範囲外                                                        |
| 10h | 指定したサブスロット番号が範囲外                                                      |
| 11h | 与えられた AR ハンドルが無効                                                      |
| 12h | 保留可能なアプリケーションが存在しない                                                   |
| 13h | 不明なエラー (PROFINET IO スタックにより要求が拒否された)                                  |
| 14h | 既に最大数のサブモジュールが埋め込まれている                                                |

## 10.7 PROFINET 追加診断オブジェクト (0Fh)

### カテゴリ

拡張

### オブジェクトの説明

このオブジェクトは、PROFINET 固有の高度な診断機能をサポートします。このオブジェクトの各インスタンスは、ネットワーク上の診断エントリに対応します。

### サポートされているコマンド

オブジェクト：   Get Attribute  
                   Set Attribute  
                   Create           (81 ページ「コマンド詳細 :Create」を参照)  
                   Delete          (83 ページ「コマンド詳細 :Delete」を参照)  
                   Process\_Alarm   (84 ページ「コマンド詳細 :Process\_Alarm」を参照)

インスタンス：   Get Attribute

### オブジェクトアトリビュート

| # | 名前                   | アクセス | データタイプ   | Value                               |
|---|----------------------|------|----------|-------------------------------------|
| 1 | Name                 | Get  | CHAR の配列 | "Additional diagnostic PROFINET IO" |
| 2 | Revision             | Get  | UINT8    | 02h                                 |
| 3 | Number of instances  | Get  | UINT16   | 保留中のイベントの数と同じ                       |
| 4 | Highest instance no. | Get  | UINT16   | Highest created instance no.        |
| 1 | Max.no of instances  | Get  | UINT16   | 6                                   |

### インスタンスアトリビュート (インスタンス #1 ~ n)

#### 拡張

| # | 名前   | アクセス | データタイプ | 説明                                                  |
|---|------|------|--------|-----------------------------------------------------|
| 1 | Type | Get  | UINT8  | インスタンスの種類：<br>値：意味：<br>00h チャネル診断<br>(その他) 将来のために予約 |

### コマンド詳細 :Create

#### カテゴリ

拡張

#### 詳細

コマンドコード :03h

有効な対象： オブジェクト

## 説明

このコマンドを実行すると、チャンネル診断エントリが作成され、モジュールによりチャンネル診断アラームが発行されます。Anybus CompactCom モジュールが PROCESS\_ACTIVE 状態にあるときのみ、新規インスタンスを作成できます。モジュールが他の状態にあるときにこのコマンドを実行すると、エラーが返されます (0Dh, "The command is not supported in the current state")。

### • コマンド詳細

| フィールド     | 内容                  | コメント                                                                                                                                         |
|-----------|---------------------|----------------------------------------------------------------------------------------------------------------------------------------------|
| CmdExt[0] | (予約。ゼロに設定)          | (将来のために予約)                                                                                                                                   |
| CmdExt[1] | SubCommand 指定子      | 作成するインスタンスの種類：<br>値： 意味：<br>0: チャンネル診断<br>(その他) (将来のために予約)                                                                                   |
| Data[0]   | API (下位ワード、下位バイト)   | アプリケーションプロセスインスタンス (API)<br><br>下記も参照してください。<br>- 18 ページ「アプリケーションプロセス インスタンス (API)」                                                          |
| Data[1]   | API (下位ワード、上位バイト)   |                                                                                                                                              |
| Data[2]   | API (上位ワード、下位バイト)   |                                                                                                                                              |
| Data[3]   | API (上位ワード、上位バイト)   |                                                                                                                                              |
| Data[4]   | SlotNr (下位バイト)      | 診断エントリと関連付けられたスロット番号                                                                                                                         |
| Data[5]   | SlotNr (上位バイト)      |                                                                                                                                              |
| Data[6]   | SubSlotNr (下位バイト)   | 診断エントリと関連付けられたサブスロット番号                                                                                                                       |
| Data[7]   | SubSlotNr (上位バイト)   |                                                                                                                                              |
| Data[8]   | ChannelNr (下位バイト)   | チャンネル番号：<br>値： 意味：                                                                                                                           |
| Data[9]   | ChannelNr (上位バイト)   | 0 ~ 7FFFh: メーカー固有<br>8000h 特定のチャンネルではなくサブモジュール自身を参照                                                                                          |
| Data[10]  | ChannelPropType     | チャンネルの種類<br>値： 意味：<br>0: その他<br>1: 1 bit<br>2: 2 bit<br>3: 4 bit<br>4: バイト<br>5: ワード (2 バイト)<br>6: ダブルワード (4 バイト)<br>7: ロングワード (8 バイト)       |
| Data[11]  | ChannelPropDir      | チャンネルの方向：<br>値： 意味：<br>0: メーカー固有<br>1: 入力 (I/O コントローラーへのデータ)<br>2: 出力 (I/O コントローラーからのデータ)<br>3: 入出力 (I/O コントローラーへのデータ、または I/O コントローラーからのデータ) |
| Data[12]  | ChannelPropAcc      | (現在は非サポート。ゼロが設定される)                                                                                                                          |
| Data[13]  | ChannelPropMaintReq |                                                                                                                                              |
| Data[14]  | ChannelPropMaintDem |                                                                                                                                              |

| フィールド    | 内容                       | コメント                                                                                                                                                                                                                 |
|----------|--------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Data[15] | ChannelErrorType (下位バイト) | チャンネルエラータイプ<br>値:       意味:<br>1:       回路ショート<br>2:       不足電圧<br>3:       過電圧<br>4:       過負荷<br>5:       過熱<br>6:       ライン切断<br>7:       上限を超えた<br>8:       下限を下回った<br>9:       エラー<br>(その他) (PROFINET IO 規格を参照) |
| Data[16] | ChannelErrorType (上位バイト) |                                                                                                                                                                                                                      |

- 応答詳細 (成功)

| フィールド   | 内容               | コメント           |
|---------|------------------|----------------|
| Data[0] | インスタンス番号 (下位バイト) | 作成されたインスタンスの番号 |
| Data[1] | インスタンス番号 (上位バイト) |                |

下記も参照してください。

- Anybus-CompactCom Software Design Guide (コマンド詳細 : 作成、診断オブジェクト)

## コマンド詳細 :Delete

### カテゴリ

拡張

### 詳細

コマンドコード :04h

有効な対象: インスタンス:

### 説明

このコマンドは、以前に作成された診断イベント/インスタンスを削除します。

- コマンド詳細

| フィールド     | 内容                    |
|-----------|-----------------------|
| CmdExt[0] | 削除するインスタンスの番号 (下位バイト) |
| CmdExt[1] | 削除するインスタンスの番号 (上位バイト) |

- 応答詳細

-

## コマンド詳細 :Process\_Alarm

### カテゴリ

拡張

### 詳細

コマンドコード :10h

有効な対象: オブジェクト

### 説明

このコマンドは、ネットワーク上にプロセスアラームを発行します。インスタンスは作成されません。そのため、"delete" を発行してインスタンスを削除することはできません。

- コマンド詳細

| フィールド       | 内容                      | コメント                                                                                                    |
|-------------|-------------------------|---------------------------------------------------------------------------------------------------------|
| CmdExt[0]   | (予約。ゼロに設定)              | (将来のために予約)                                                                                              |
| CmdExt[1]   |                         |                                                                                                         |
| Data[0]     | API (下位ワード、下位バイト)       | アプリケーションプロセスインスタンス (API)                                                                                |
| Data[1]     | API (下位ワード、上位バイト)       |                                                                                                         |
| Data[2]     | API (上位ワード、下位バイト)       | 下記も参照してください。<br>- 18 ページ「アプリケーションプロセス<br>インスタンス (API)」                                                  |
| Data[3]     | API (上位ワード、上位バイト)       |                                                                                                         |
| Data[4]     | SlotNr (下位バイト)          | 診断エントリと関連付けられたスロット<br>番号                                                                                |
| Data[5]     | SlotNr (上位バイト)          |                                                                                                         |
| Data[6]     | SubSlotNr (下位バイト)       | 診断エントリと関連付けられたサブス<br>ロット番号                                                                              |
| Data[7]     | SubSlotNr (上位バイト)       |                                                                                                         |
| Data[8]     | UserStructIdent (下位バイト) | 渡されたアラームデータの特性を指定<br>値: 意味:<br>0 ~ 7FFFh: メーカー固有<br>8000h: チャネル診断データ<br>8001h: 複数のメーカーに固有<br>(その他) (予約) |
| Data[9]     | UserStructIdent (上位バイト) |                                                                                                         |
| Data[10 ~ ] | データ                     | アラームとともに渡すデータ                                                                                           |

- 応答詳細

-

## オブジェクト固有のエラーコード

| コード | エラー                       |
|-----|---------------------------|
| 03h | API が存在しない                |
| 04h | 指定したスロットにモジュールが挿入されていない   |
| 05h | 指定したスロットにサブモジュールが挿入されていない |
| 06h | 指定したスロット番号が範囲外            |

| コード | エラー                                 |
|-----|-------------------------------------|
| 07h | 指定したサブスロット番号が範囲外                    |
| 08h | PROFINET IO スタックにチャンネル診断エントリを追加できない |
| 09h | PROFINET IO スタックにチャンネル診断アラームを追加できない |
| 0Ah | チャンネル番号が範囲外                         |
| 0Bh | ChannelPropType が範囲外                |
| 0Ch | ChannelPropDir が範囲外                 |
| 0Dh | ChannelPropAcc が範囲外                 |
| 0Eh | ChannelPropMaintReq が範囲外            |
| 0Fh | ChannelPropMaintDem が範囲外            |
| 10h | UserStructIdent が範囲外                |
| 11h | ChannelErrType が範囲外                 |
| 12h | PROFINET IO スタックにチャンネル診断エントリを追加できない |
| FFh | 不明なエラー                              |

## 10.8 ソケットインターフェイスオブジェクト (07h)

### カテゴリ

高度

### オブジェクトの説明

このオブジェクトを使用すると、TCP/IP スタックのソケットインターフェイスに直接アクセスできます。これにより、TCP/UDP を用いたカスタムプロトコルの実装が可能となります。

なお、このオブジェクトにアクセスする際、コマンドによっては分割が必要になる場合があります。詳細については、157 ページ「メッセージの分割」を参照してください。

**重要:** このオブジェクトが提供する機能は、ソケットインターフェイスのプログラミングに精通し、TCP/IP のプログラミングに関するコンセプトを十分理解している方のみ使用してください。

### サポートされているコマンド

オブジェクト :Get\_Attribute

Create ( 88 ページ「コマンド詳細 :Create」を参照 )

Delete ( 89 ページ「コマンド詳細 :Delete」を参照 )

インスタンス :Get\_Attribute

Set\_Attribute

Bind ( 90 ページ「コマンド詳細 :Bind」を参照 )

Shutdown ( 91 ページ「コマンド詳細 :Shutdown」を参照 )

Listen ( 92 ページ「コマンド詳細 :Listen」を参照 )

Accept ( 93 ページ「コマンド詳細 :Accept」を参照 )

Connect ( 94 ページ「コマンド詳細 :Connect」を参照 )

Receive ( 95 ページ「コマンド詳細 :Receive」を参照 )

Receive\_From ( 96 ページ「コマンド詳細 :Receive\_From」を参照 )

Send ( 97 ページ「コマンド詳細 :Send」を参照 )

Send\_To ( 98 ページ「コマンド詳細 :Send\_To」を参照 )

IP\_Add\_membership ( 99 ページ「コマンド詳細 :IP\_Add\_Membership」を参照 )

IP\_Drop\_membership ( 100 ページ「コマンド詳細 :IP\_Drop\_Membership」を参照 )

DNS\_Lookup ( 101 ページ「コマンド詳細 :DNS\_Loopup」を参照 )

### オブジェクトアトリビュート

| #  | 名前                    | アクセス | データタイプ   | Value              |
|----|-----------------------|------|----------|--------------------|
| 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              |

## インスタンスアトリビュート (ソケット #1 ~ 8)

### 高度

| #  | 名前                  | アクセス    | 種類     | 説明                                                                                                                                                                                                                                                                                                                                                                                      |
|----|---------------------|---------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1  | Socket type         | Get     | UINT8  | 値:ソケットの種類:<br>00h SOCK_STREAM, NON-BLOCKING (TCP)<br>01h SOCK_STREAM, BLOCKING (TCP)<br>02h SOCK_DGRAM, NON-BLOCKING (UDP)<br>03h SOCK_DGRAM, BLOCKING (UDP)                                                                                                                                                                                                                            |
| 2  | Port                | Get     | UINT16 | ソケットがバインドされているローカルポート                                                                                                                                                                                                                                                                                                                                                                   |
| 3  | Host IP             | Get     | UINT32 | ホストの IP アドレス。接続されていない場合は 0 (ゼロ)                                                                                                                                                                                                                                                                                                                                                         |
| 4  | Host port           | Get     | UINT16 | ホストのポート番号。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):<br>値:状態:説明:<br>00h CLOSED クローズ状態<br>01h LISTEN 接続要求待ち<br>02h SYN_SENT アクティブ状態。SYN 送信完了<br>03h SYN_RECEIVED SYN 送受信完了<br>04h ESTABLISHED 接続確立<br>05h CLOSE_WAIT FIN 受信完了。クローズ待ち<br>06h FIN_WAIT_1 クローズ完了。FIN 送信完了<br>07h CLOSING FIN 送受信完了。FIN ACK 受信待ち<br>08h LAST_ACK FIN 送信とクローズが完了。FIN ACK 受信待ち<br>09h FIN_WAIT_2 クローズ完了。FIN の確認応答を受信完了<br>0Ah TIME_WAIT クローズ後のウェイト |
| 6  | TCP RX bytes        | Get     | UINT16 | RX バッファのバイト数 (TCP ソケットのみ)                                                                                                                                                                                                                                                                                                                                                               |
| 7  | TCP TX bytes        | Get     | UINT16 | TX バッファのバイト数 (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利用可能<br>値:意味:<br>1 有効<br>0 無効 (デフォルト)                                                                                                                                                                                                                                                                                                                                    |
| 9  | Keep alive          | Get/Set | BOOL   | プロトコルにより、アイドル状態になっている接続の生存確認が行われます (TCP ソケットのみ) <sup>a</sup><br>値:意味:<br>1 有効<br>0 無効 (デフォルト)                                                                                                                                                                                                                                                                                           |
| 10 | IP Multicast TTL    | Get/Set | UINT8  | IP マルチキャストの TTL 値 (UDP ソケットのみ)<br>デフォルト = 1                                                                                                                                                                                                                                                                                                                                             |
| 11 | IP Multicast Loop   | Get/Set | BOOL   | IP マルチキャストのループバック (UDP ソケットのみ) <sup>b</sup><br>値:意味:<br>1 有効 (デフォルト)<br>0 無効                                                                                                                                                                                                                                                                                                            |
| 12 | Ack delay time      | Get/Set | UINT16 | ms 単位で表した ACK 遅延時間 (TCP ソケットのみ)<br>デフォルト = 200ms <sup>c</sup>                                                                                                                                                                                                                                                                                                                           |
| 13 | TCP No Delay        | Get/Set | BOOL   | 遅延なしに連結パケットを送信する (TCP)<br>値:意味:<br>1 遅延あり (デフォルト)<br>0 遅延なし (ソケットの Nagle アルゴリズムをオフにする)                                                                                                                                                                                                                                                                                                  |
| 14 | TCP Connect Timeout | Get/Set | UINT16 | 秒単位で表した TCP 接続タイムアウト (デフォルト = 75 秒)                                                                                                                                                                                                                                                                                                                                                     |

- a. Keep alive アトリビュートが設定されると、120 分間アイドル状態になっている接続の生存確認が行われます。生存確認に失敗すると、75 秒ごとにその接続の生存確認が行われます。生存確認が 8 回失敗すると、その接続を終了します。
- b. ループバックされたメッセージを取得するには、グループに属している必要があります。
- c. 分解能 50ms。すなわち、50 ~ 99 = 50ms、100 ~ 149 = 100ms、199 = 150ms など。

## コマンド詳細 :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:** このコマンドは、WAIT\_PROCESS、IDLE、PROCESS\_ACTIVE の状態でのみ使用できます。

- コマンド詳細

| フィールド     | 内容                                                                                                                                                           |
|-----------|--------------------------------------------------------------------------------------------------------------------------------------------------------------|
| CmdExt[0] | (予約。ゼロに設定)                                                                                                                                                   |
| CmdExt[1] | 値:ソケットの種類:<br>00h SOCK_STREAM, NON-BLOCKING (TCP)<br>01h SOCK_STREAM, BLOCKING (TCP)<br>02h SOCK_DGRAM, NON-BLOCKING (UDP)<br>03h SOCK_DGRAM, BLOCKING (UDP) |

- 応答詳細

| フィールド   | 内容            | コメント         |
|---------|---------------|--------------|
| Data[0] | インスタンス番号 (下位) | 作成されたソケットの番号 |
| Data[1] | インスタンス番号 (上位) |              |

## コマンド詳細 :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、作成されたソケットを削除し、コネクションをクローズします (接続されている場合)。

- TCP ソケットによりコネクションが確立されている場合、RST フラグを使用してコネクションを終了します。
- TCP コネクションを緩やかに終了するには、ソケットを削除する前に "Shutdown" コマンドを使うことを推奨します (91 ページ「コマンド詳細 :Shutdown」を参照)。これにより、RST フラグではなく FIN フラグによりコネクションがクローズされます。

- コマンド詳細

| フィールド     | 内容                 | コメント              |
|-----------|--------------------|-------------------|
| CmdExt[0] | 削除するインスタンスの番号 (下位) | 削除す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の番号 (上位) |                   |

- 応答詳細  
(データなし)

## コマンド詳細 :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にバインドします。

- コマンド詳細

| フィールド     | 内容               | コメント                               |
|-----------|------------------|------------------------------------|
| CmdExt[0] | 要求されたポートの番号 (下位) | いずれかの空きポートにバインドを要求するには0(ゼロ)を設定します。 |
| CmdExt[1] | 要求されたポートの番号 (上位) |                                    |

- 応答詳細

| 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 (下位) | ソケットがバインドされた実際のポート |
| CmdExt[1] | バインドされたポート番号 (上位) |                    |

## コマンド詳細 :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 コネクションをクローズします。なお、このコマンドの応答は、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表しません。そのため、このコマンドを使用して非ブロッキングソケットをポーリングしたり、ブロッキングソケットをブロックしたりすることはできません。

- コマンド詳細

| フィールド     | 内容                                                                                  |
|-----------|-------------------------------------------------------------------------------------|
| CmdExt[0] | (予約。ゼロに設定)                                                                          |
| CmdExt[1] | 値：モード：<br>00h 受信チャンネルをシャットダウン<br>01h 送信チャンネルをシャットダウン<br>02h 受信チャンネルと送信チャンネルをシャットダウン |

- 応答詳細

(データなし)

TCP コネクションを緩やかにシャットダウンする場合の推奨手順を以下に示します。

アプリケーションがシャットダウンを開始する場合：

1. CmdExt[1] を 01h に設定してシャットダウンを送信します。すると、ホストに FIN フラグが送信され、送信チャンネルがシャットダウンされます。このとき、受信チャンネルは動作を継続します。
2. オブジェクト指定子エラー (EDESTADDRREQ (14)) のメッセージを受信するまで、ソケットのデータを受信します。このエラーは、ホストが受信チャンネルをクローズしたことを表します。ホストが受信チャンネルをクローズしない場合、タイムアウトを使用して手順 3 に進みます。
3. ソケットのインスタンスを削除します。手順 2 でタイムアウトになった場合、RST フラグが送信されてソケットが終了します。

ホストがシャットダウンを開始する場合：

1. ソケットのデータを受信します。ゼロバイトのデータを受信した場合、ホストがそのソケットの受信チャンネルをクローズしたことを表します。
2. 未送信のデータをホストに送信します。
3. CmdExt[1] を 01h に設定してシャットダウンを送信します。すると、ホストに FIN フラグが送信され、受信チャンネルがシャットダウンされます。
4. ソケットのインスタンスを削除します。

## コマンド詳細 :Listen

### 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、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、ソケットにより許容される受け付けられなかった接続の数になります。バックログキューが一杯になると、それ以降の接続は RST フラグにより拒否されません。

- コマンド詳細

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

- 応答詳細  
( データなし )

## コマンド詳細 :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、リスン中の TCP ソケットへの受信コネクションを受け付けます。受け付けられた各コネクションに対し、新たにソケットインスタンスが作成されます。新規に作成されたソケットはホストに接続され、そのインスタンス番号に対して応答が返されます。

#### NON-BLOCKING モード：

受信コネクションに対し、このコマンドを定期的に発行 (ポーリング) する必要があります。受信接続要求が存在しない場合、本モジュールはエラーコード 0006h (EWOULDBLOCK) を返します。

#### BLOCKING モード：

このコマンドを発行すると、接続要求が検出されるまでブロックされます。

**注：**このコマンドは、受け付けたコネクションに対して使用可能なインスタンスが存在する場合のみ受け付けられます。ブロッキングコネクションの場合、このコマンドがインスタンスを予約します。

- コマンド詳細  
(データなし)
- 応答詳細

| 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 (下位) |
| Data[1] | 接続されたソケットのインスタンス番号 (上位) |
| Data[2] | ホストの IP アドレス。バイト 3 (下位) |
| Data[3] | ホストの IP アドレス。バイト 2      |
| Data[4] | ホストの IP アドレス。バイト 1      |
| Data[5] | ホストの IP アドレス。バイト 0 (上位) |
| Data[6] | ホストのポート番号 (下位)          |
| Data[7] | ホストのポート番号 (上位)          |

## コマンド詳細 :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、このコマンドを発行すると、ソケットを割り当てる相手が指定されます(その相手に対してデータグラムが送信され、その相手のアドレスからのデータグラムのみ受信します)。

SOCK\_STREAM ソケットの場合、このコマンドを発行するとホストへの接続が試みられます。

SOCK\_STREAM ソケットでは1回しか接続できませんが、SOCK\_DGRAM ではこのサービスを何度でも使用して割り当てを変更できます。SOCK-DGRAM ソケットでは、IP アドレス 0.0.0.0、ポート 0 (ゼロ) に接続するとこの割り当てが解除されます。

#### NON-BLOCKING モード：

接続が確立するか、接続が拒否されるか、または、接続がタイムアウトするまで、このコマンドを繰り返し発行 (ポーリング) できます。最初の接続試行が受け付けられた後にこのコマンドを発行すると、接続試行のポーリング要求に対してエラーコード 22 (EINPROGRESS) が返されます。

#### BLOCKING モード：

このコマンドを発行すると、接続が確立するか、または、タイムアウトや接続エラーにより接続要求がキャンセルされるまで、処理がブロックされます。

#### • コマンド詳細

| フィールド     | 内容                      | 内容      |
|-----------|-------------------------|---------|
| CmdExt[0] | (予約)                    | (ゼロに設定) |
| CmdExt[1] |                         |         |
| Data[0]   | ホストの IP アドレス。バイト 3 (下位) | -       |
| Data[1]   | ホストの IP アドレス。バイト 2      |         |
| Data[2]   | ホストの IP アドレス。バイト 1      |         |
| Data[3]   | ホストの IP アドレス。バイト 0 (上位) |         |
| Data[4]   | ホストのポート番号 (下位)          |         |
| Data[5]   | ホストのポート番号 (上位)          |         |

#### • 応答詳細

(データなし)

## コマンド詳細 :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、接続されているソケットからデータを受信します。メッセージの分割を使用して、最大 1472 バイトまで受信します (157 ページ「メッセージの分割」を参照)。

SOCK\_DGRAM ソケットの場合、本モジュールは、次に受信したデータグラムから要求した量のデータを受信します。データグラムのサイズが要求したサイズより小さい場合、データグラム全体が応答メッセージとして返されます。データグラムのサイズが要求したサイズより大きい場合、余分なバイトは捨てられます。

SOCK\_STREAM ソケットの場合、本モジュールは、次に受信したデータストリームから要求したバイト数のデータを受信します。実際のデータサイズが要求したサイズより小さい場合、利用可能な全てのデータが返されます。

#### NON-BLOCKING モード:

利用可能なデータがソケットに存在しない場合、エラー コード 0006h (EWOULDBLOCK) が返されます。

#### BLOCKING モード:

処理が完了するまで、本モジュールは応答メッセージを発行しません。

本モジュールが 0 (ゼロ) バイトのデータを返した場合、ホストにより接続がクローズされたことを表します。ただし、送信チャネルはまだ有効であるため、"Shutdown" コマンドや "Delete" コマンドにて送信チャネルをクローズする必要があります。

#### • コマンド詳細

| フィールド     | 内容            | メモ                  |
|-----------|---------------|---------------------|
| CmdExt[0] | (予約)          | (ゼロに設定)             |
| CmdExt[1] | 分割制御ビット       | 157 ページ「コマンドの分割」を参照 |
| Data[0]   | 受信データサイズ (下位) | 最初のセグメントでのみ使用       |
| Data[1]   | 受信データサイズ (上位) |                     |

#### • 応答詳細

注: 応答内のデータは分割されている場合があります (157 ページ「メッセージの分割」を参照)。

| フィールド       | 内容      | メモ                |
|-------------|---------|-------------------|
| CmdExt[0]   | (予約)    | (ゼロに設定)           |
| CmdExt[1]   | 分割制御ビット | 158 ページ「応答の分割」を参照 |
| Data[0 ~ n] | 受信データ   | -                 |

## コマンド詳細 :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。メッセージの分割を使用して、最大 1472 バイトまで受信します (157 ページ「メッセージの分割」を参照)。

本モジュールは、次に受信したデータグラムから要求した量のデータを受信します。データグラムのサイズが要求したサイズより小さい場合、データグラム全体が応答メッセージとして返されます。データグラムのサイズが要求したサイズより大きい場合、余分なバイトは捨てられます。

応答メッセージには、送信元の IP アドレスとポート番号が含まれます。

#### NON-BLOCKING モード:

利用可能なデータがソケットに存在しない場合、エラー コード 0006h (EWOULDBLOCK) が返されます。

#### BLOCKING モード:

処理が完了するまで、本モジュールは応答メッセージを発行しません。

#### • コマンド詳細

| フィールド     | 内容            | メモ                  |
|-----------|---------------|---------------------|
| CmdExt[0] | (予約)          | (ゼロに設定)             |
| CmdExt[1] | 分割制御ビット       | 157 ページ「コマンドの分割」を参照 |
| Data[0]   | 受信データサイズ (下位) | 最初のセグメントでのみ使用       |
| Data[1]   | 受信データサイズ (上位) |                     |

#### • 応答詳細

注: 応答内のデータは分割されている場合があります (157 ページ「メッセージの分割」を参照)。

| フィールド       | 内容                      | メモ                                                               |
|-------------|-------------------------|------------------------------------------------------------------|
| CmdExt[0]   | (予約)                    | (ゼロに設定)                                                          |
| CmdExt[1]   | 分割制御ビット                 | 158 ページ「応答の分割」を参照                                                |
| Data[0]     | ホストの IP アドレス。バイト 3 (下位) | ホストのアドレス / ポート情報は、最初のセグメントにのみ含まれます。それ以降のデータは、全て Data[0] から始まります。 |
| Data[1]     | ホストの IP アドレス。バイト 2      |                                                                  |
| Data[2]     | ホストの IP アドレス。バイト 1      |                                                                  |
| Data[3]     | ホストの IP アドレス。バイト 0 (上位) |                                                                  |
| Data[4]     | ホストのポート番号 (下位)          |                                                                  |
| Data[5]     | ホストのポート番号 (上位)          |                                                                  |
| Data[6 ~ n] | 受信データ                   |                                                                  |

## コマンド詳細 :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、接続されているソケットにてデータを送信します。メッセージの分割を使用して、最大 1472 バイトまで送信します (157 ページ「メッセージの分割」を参照)。

NON-BLOCKING モード:

送信バッファに十分な空きがない場合、本モジュールはエラー コード 0006h (EWOULDBLOCK) を返します。

BLOCKING モード:

送信バッファに十分な空きがない場合、本モジュールは送信バッファに空きができるまで処理をブロックします。

#### • コマンド詳細

注: 大きなサイズのデータ (255 バイトを超えるデータ) を送信するには、コマンドのデータを分割する必要があります (157 ページ「メッセージの分割」を参照)。

| フィールド       | 内容    | メモ                  |
|-------------|-------|---------------------|
| CmdExt[0]   | (予約)  | (ゼロに設定)             |
| CmdExt[1]   | 分割制御  | 157 ページ「コマンドの分割」を参照 |
| Data[0 ~ n] | 送信データ | -                   |

#### • 応答詳細

| フィールド     | 内容          | メモ            |
|-----------|-------------|---------------|
| CmdExt[0] | (予約)        | (無視)          |
| CmdExt[1] |             |               |
| Data[0]   | 送信バイト数 (下位) | 最後のセグメントでのみ有効 |
| Data[1]   | 送信バイト数 (上位) |               |

## コマンド詳細 :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、指定したホストに対し、接続されていない SOCK-DGRAM ソケットにてデータを送信します。メッセージの分割を使用して、最大 1472 バイトまで送信します (157 ページ「メッセージの分割」を参照)。

- コマンド詳細

**注：**大きなサイズのデータ (255 バイトを超えるデータ) を送信するには、コマンドのデータを分割する必要があります (157 ページ「メッセージの分割」を参照)。

| フィールド       | 内容                      | メモ                                                                      |
|-------------|-------------------------|-------------------------------------------------------------------------|
| CmdExt[0]   | (予約)                    | (ゼロに設定)                                                                 |
| CmdExt[1]   | 分割制御                    | 157 ページ「コマンドの分割」を参照                                                     |
| Data[0]     | ホストの IP アドレス。バイト 3 (下位) | ホストのアドレス / ポート情報は、最初のセグメントにのみ含めてください。それ以降のデータは、全て Data[0] から始める必要があります。 |
| Data[1]     | ホストの IP アドレス。バイト 2      |                                                                         |
| Data[2]     | ホストの IP アドレス。バイト 1      |                                                                         |
| Data[3]     | ホストの IP アドレス。バイト 0 (上位) |                                                                         |
| Data[4]     | ホストのポート番号 (下位)          |                                                                         |
| Data[5]     | ホストのポート番号 (上位)          |                                                                         |
| Data[6 ~ n] | 送信データ                   |                                                                         |

- 応答詳細

| フィールド     | 内容          | メモ            |
|-----------|-------------|---------------|
| CmdExt[0] | (予約)        | (無視)          |
| CmdExt[1] |             |               |
| Data[0]   | 送信バイト数 (下位) | 最後のセグメントでのみ有効 |
| Data[1]   | 送信バイト数 (上位) |               |

## コマンド詳細 :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、IP マルチキャストグループのメンバーシップにソケットを割り当てます。本モジュールは、常に "全ホストグループ" に自動的に参加します。ただし、このコマンドを使用して最大 20 個の追加メンバーシップを指定できます。

- コマンド詳細

| フィールド     | 内容                       | メモ      |
|-----------|--------------------------|---------|
| CmdExt[0] | (予約)                     | (ゼロに設定) |
| CmdExt[1] |                          |         |
| Data[0]   | グループの IP アドレス。バイト 3 (下位) | -       |
| Data[1]   | グループの IP アドレス。バイト 2      |         |
| Data[2]   | グループの IP アドレス。バイト 1      |         |
| Data[3]   | グループの IP アドレス。バイト 0 (上位) |         |

- 応答詳細  
(データなし)

## コマンド詳細 :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のメンバーシップからソケットを削除します。

- コマンド詳細

| フィールド     | 内容                       | メモ      |
|-----------|--------------------------|---------|
| CmdExt[0] | (予約)                     | (ゼロに設定) |
| CmdExt[1] |                          |         |
| Data[0]   | グループの IP アドレス。バイト 3 (下位) | -       |
| Data[1]   | グループの IP アドレス。バイト 2      |         |
| Data[2]   | グループの IP アドレス。バイト 1      |         |
| Data[3]   | グループの IP アドレス。バイト 0 (上位) |         |

- 応答詳細  
(データなし)

## コマンド詳細 :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、与えられたホスト名を解決し、その IP アドレスを返します。

- コマンド詳細

| フィールド       | 内容        | メモ       |
|-------------|-----------|----------|
| CmdExt[0]   | (予約)      | (ゼロに設定)  |
| CmdExt[1]   |           |          |
| Data[0 ~ N] | Host name | 解決するホスト名 |

- 応答詳細 (成功)

| フィールド     | 内容                 | メモ               |
|-----------|--------------------|------------------|
| CmdExt[0] | (予約)               | (ゼロに設定)          |
| CmdExt[1] |                    |                  |
| Data[0]   | IP アドレス。バイト 3 (下位) | 指定したホストの 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 (上位) |                  |

## ソケットインターフェイスのエラーコード (オブジェクト固有)

ソケットインターフェイスオブジェクトを使用する際に、本モジュールにより返されるオブジェクト固有のエラーコードを以下に示します。

| エラーコード | 名前            | 意味                                                                                         |
|--------|---------------|--------------------------------------------------------------------------------------------|
| 1      | ENOBUFS       | 内部バッファが利用できない                                                                              |
| 2      | ETIMEDOUT     | タイムアウトイベントが発生した                                                                            |
| 3      | EISCONN       | ソケットが既に接続されている                                                                             |
| 4      | EOPNOTSUPP    | サービスがサポートされていない                                                                            |
| 5      | ECONNABORTED  | 接続が中断された                                                                                   |
| 6      | EWOULDBLOCK   | 非ブロッキングタイプのソケットであるため、ソケットをブロックできない                                                         |
| 7      | ECONNREFUSED  | 接続が拒否された                                                                                   |
| 8      | ECONNRESET    | 接続がリセットされた                                                                                 |
| 9      | ENOTCONN      | ソケットが接続されていない                                                                              |
| 10     | EALREADY      | ソケットが既に要求されたモードにある                                                                         |
| 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         | パイプでエラーが発生した                                                                               |
| 14     | EDESTADDRREQ  | 送信先アドレスが必要                                                                                 |
| 15     | ESHUTDOWN     | ソケットが既にシャットダウンされている                                                                        |
| 16     | (予約)          | -                                                                                          |
| 17     | EHAVEOOB      | 帯域外のデータが利用可能                                                                               |
| 18     | ENOMEM        | 内部メモリが利用できない                                                                               |
| 19     | EADDRNOTAVAIL | アドレスが利用できない                                                                                |
| 20     | EADDRINUSE    | アドレスが既に使用されている                                                                             |
| 21     | (予約)          | -                                                                                          |
| 22     | EINPROGRESS   | サービスが既に処理中である                                                                              |
| 28     | ETOOMANYREFS  | 参照が多すぎる                                                                                    |
| 101    | コマンドが中断された    | コマンドによりソケット上の処理がブロックされている状態で、そのソケットが Delete コマンドにより削除された場合、ブロックしているコマンドに対してこのエラーコードが返されます。 |

## 10.9 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

このオブジェクトグループは、SMTP クライアントに関する機能を提供します。  
下記も参照してください。

- 16 ページ「ファイルシステム」
- 34 ページ「E-mail クライアント」
- 65 ページ「インスタンスアトリビュート (インスタンス #12、SMTP サーバー)」
- 66 ページ「インスタンスアトリビュート (インスタンス #13、SMTP ユーザー)」
- 66 ページ「インスタンスアトリビュート (インスタンス #14、SMTP パスワード)」

### サポートされているコマンド

オブジェクト :   Get\_Attribute  
                  Create  
                  Delete  
                  Send email from file (106 ページ「コマンド詳細 :Send Email From File」)

インスタンス :   Get\_Attribute  
                  Set\_Attribute  
                  Send email (107 ページ「コマンド詳細 :Send Email」)

### オブジェクトアトリビュート

| #  | 名前                    | アクセス | データタイプ   | Value           |
|----|-----------------------|------|----------|-----------------|
| 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   | 正しく送信されたメッセージの数 |
| 13 | Error count           | Get  | UINT16   | 送信されなかったメッセージの数 |

## インスタンスアトリビュート

### 高度

インスタンスはアプリケーションにより動的に作成されます。

| # | 名前      | アクセス    | 種類       | 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、E-mail インスタンスを作成します。

- コマンド詳細

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

- 応答詳細

| フィールド      | 内容       | コメント  |
|------------|----------|-------|
| CmdExt[0]  | (予約、無視)  | -     |
| CmdExt[1]  |          |       |
| MsgData[0] | インスタンス番号 | 下位バイト |
| MsgData[1] |          | 上位バイト |

## コマンド詳細 :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、E-mail インスタンスを削除します。

- コマンド詳細

| フィールド     | 内容              | コメント  |
|-----------|-----------------|-------|
| CmdExt[0] | E-mail インスタンス番号 | 下位バイト |
| CmdExt[1] |                 | 上位バイト |

- 応答詳細  
(データなし)

## コマンド詳細 :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 E-mail を送信します。

ファイル形式:

ファイルは、以下の形式を持つプレーン ASCII ファイルでなければなりません。

[To]  
recipient

[From]  
sender

[Subject]  
email subject

[Headers]  
extra headers, optional

[Message]  
actual email message

- コマンド詳細

| 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] | メッセージファイルのパス + ファイル名 | -    |

- 応答詳細  
(データなし)

## コマンド詳細 :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、指定した E-mail インスタンスを送信します。

- コマンド詳細  
(データなし)
- 応答詳細  
(データなし)

## オブジェクト固有のエラーコード

| エラーコード | 意味                  |
|--------|---------------------|
| 1      | SMTP サーバーが見つからない    |
| 2      | SMTP サーバーの準備ができていない |
| 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      | E-mail ファイルを解釈できない  |
| 255    | その他の SMTP エラー       |
| (その他)  | (予約)                |

## 10.10 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、内蔵ファイルシステムへのインターフェイスを提供します。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ており、ファイルシステムの操作に関するサービスが用意されています。

### サポートされているコマンド

オブジェクト :   Get\_Attribute  
                  Create (110 ページ「コマンド詳細 :Create」)  
                  Delete (111 ページ「コマンド詳細 :Delete」)  
                  Format Disc (120 ページ「コマンド詳細 :Format Disc」)

インスタンス :   Get\_Attribute  
                  File Open (111 ページ「コマンド詳細 :File Open」)  
                  File Close (112 ページ「コマンド詳細 :File Close」)  
                  File Delete (112 ページ「コマンド詳細 :File Delete」)  
                  File Copy (113 ページ「コマンド詳細 :File Copy」)  
                  File Rename (114 ページ「コマンド詳細 :File Rename」)  
                  File Read (115 ページ「コマンド詳細 :File Read」)  
                  File Write (116 ページ「コマンド詳細 :File Write」)  
                  Directory Open (116 ページ「コマンド詳細 :Directory Open」)  
                  Directory Close (117 ページ「コマンド詳細 :Directory Close」)  
                  Directory Delete (117 ページ「コマンド詳細 :Directory Delete」)  
                  Directory Read (118 ページ「コマンド詳細 :Directory Read」)  
                  Directory Create (119 ページ「コマンド詳細 :Directory Create」)  
                  Directory Change (119 ページ「コマンド詳細 :Directory Change」)

## オブジェクトアトリビュート (インスタンス #0)

| #  | 名前                          | アクセス | データタイプ     | Value                   |
|----|-----------------------------|------|------------|-------------------------|
| 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 | -                       |

## インスタンスアトリビュート

### 高度

| # | 名前            | アクセス | 種類       | 説明                                                    |
|---|---------------|------|----------|-------------------------------------------------------|
| 1 | Instance type | Get  | UINT8    | 値：種類：<br>00h 予約<br>01h ファイルインスタンス<br>02h ディレクトリインスタンス |
| 2 | File size     | Get  | UINT32   | バイト単位で表したファイルサイズ(ディレクトリの場合はゼロ)                        |
| 3 | Path          | Get  | CHAR の配列 | インスタンスにより操作されるパス                                      |

## コマンド詳細 :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、ファイル操作インスタンスを作成します。

- コマンド詳細

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

- 応答詳細

| フィールド      | 内容       | コメント  |
|------------|----------|-------|
| CmdExt[0]  | (予約、無視)  | -     |
| CmdExt[1]  |          |       |
| MsgData[0] | インスタンス番号 | 下位バイト |
| MsgData[1] |          | 上位バイト |

## コマンド詳細 :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、ファイル操作インスタンスを削除します。

- コマンド詳細

| フィールド     | 内容              | コメント  |
|-----------|-----------------|-------|
| CmdExt[0] | ファイル操作インスタンスの番号 | 下位バイト |
| CmdExt[1] |                 | 上位バイト |

- 応答詳細  
(データなし)

## コマンド詳細 :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、読み取り、書き込み、追加を行うファイルを開きます。

- コマンド詳細

| フィールド          | 内容         | コメント                                              |
|----------------|------------|---------------------------------------------------|
| CmdExt[0]      | モード        | 値：モード：<br>00h 読み取りモード<br>01h 書き込みモード<br>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 | 現在のパスからの相対位置                                      |

- 応答詳細  
(データなし)

## コマンド詳細 :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、開いているファイルをクローズします。

- コマンド詳細  
(データなし)
- 応答詳細

| フィールド      | 内容       | コメント        |
|------------|----------|-------------|
| CmdExt[0]  | (予約、無視)  | -           |
| CmdExt[1]  |          |             |
| MsgData[0] | ファイル サイズ | 下位バイト、下位ワード |
| MsgData[1] |          | -           |
| MsgData[2] |          | -           |
| MsgData[3] |          | 上位バイト、上位ワード |

## コマンド詳細 :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、指定したファイルをファイルシステムから恒久的に削除します。

- コマンド詳細

| 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 | 現在のバスからの相対位置 |

- 応答詳細  
(データなし)

## コマンド詳細 :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] | 入力パス + ファイル名 | NULL で区切られた、現在のパスからの相対位置 |
|                | NULL         |                          |
|                | 出力パス + ファイル名 |                          |

- 応答詳細  
(データなし)

## コマンド詳細 :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またはファイルを移動します。

- コマンド詳細

| 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] | 現在のパス + ファイル名 | NULL で区切られた、現在のパスからの相対位置 |
|                | NULL          |                          |
|                | 新規パス + ファイル名  |                          |

- 応答詳細  
(データなし)

## コマンド詳細 :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

読み取り用に開いているファイルからデータを読み取ります。

- コマンド詳細

| フィールド     | 内容         | コメント     |
|-----------|------------|----------|
| CmdExt[0] | バイト数       | 読み取るバイト数 |
| CmdExt[1] | (予約。ゼロに設定) | -        |

- 応答詳細

| フィールド          | 内容      | コメント           |
|----------------|---------|----------------|
| CmdExt[0]      | (予約、無視) | -              |
| CmdExt[1]      |         |                |
| MsgData[0 ~ n] | データ     | ファイルから読み取ったデータ |

## コマンド詳細 :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

書き込み用または追加用を開いているファイルにデータを書き込みます。

- コマンド詳細

| 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        | ファイルに書き込むデータ |

- 応答詳細

| フィールド     | 内容      | コメント     |
|-----------|---------|----------|
| CmdExt[0] | バイト数    | 書き込むバイト数 |
| CmdExt[1] | (予約、無視) | -        |

## コマンド詳細 :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] | ディレクトリのパス + 名前 | 現在のパスからの相対位置 |

- 応答詳細  
(データなし)

## コマンド詳細 :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、開いているディレクトリをクローズします。

- コマンド詳細  
(データなし)
- 応答詳細  
(データなし)

## コマンド詳細 :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恒久的に削除します。

- コマンド詳細

| 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] | ディレクトリのパス + 名前 | 現在のパスからの相対位置 |

- 応答詳細  
(データなし)

## コマンド詳細 :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、読み取り用に開いているディレクトリの内容を読み取ります。

このコマンドは、1個のディレクトリエントリの情報を返します。そのため、ディレクトリの全ての内容を取得するには、このコマンドを複数回発行する必要があります。最後のエントリを読み込むと、このコマンドは"空の"応答(データサイズがゼロの応答)を返します。

- コマンド詳細  
(データなし)
- 応答詳細

| フィールド          | 内容       | コメント                                                                                             |
|----------------|----------|--------------------------------------------------------------------------------------------------|
| CmdExt[0]      | (予約、無視)  | -                                                                                                |
| CmdExt[1]      |          |                                                                                                  |
| MsgData[0]     | エントリのサイズ | 下位バイト、下位ワード                                                                                      |
| MsgData[1]     |          | -                                                                                                |
| MsgData[2]     |          | -                                                                                                |
| MsgData[3]     |          | 上位バイト、上位ワード                                                                                      |
| MsgData[4]     | フラグ      | <u>ビット:意味:</u><br>0 エントリはディレクトリである<br>1 エントリは読み取り専用である<br>2 エントリは隠しエントリである<br>3 エントリはシステムエントリである |
| MsgData[5 ~ n] | エントリ名    | -                                                                                                |

## コマンド詳細 :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] | ディレクトリのパス + 名前 | 現在のパスからの相対位置 |

- 応答詳細  
(データなし)

## コマンド詳細 :Directory Change

### カテゴリ

高度

### 詳細

コマンドコード：

有効な対象： インスタンス：

### 説明

このコマンドは、インスタンスのカレントディレクトリ/パスを変更します。

- コマンド詳細

| 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] | ディレクトリのパス + 名前 | 現在のパスからの相対位置 |

- 応答詳細  
(データなし)

## コマンド詳細 :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。

- コマンド詳細

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

- 応答詳細  
(データなし)

## オブジェクト固有のエラーコード

| エラーコード | 意味                   |
|--------|----------------------|
| 1      | ファイルを開けない            |
| 2      | ファイルを閉じられない          |
| 3      | ファイルを削除できない          |
| 4      | ディレクトリを開けない          |
| 5      | ディレクトリを閉じられない        |
| 6      | ディレクトリを作成できない        |
| 7      | ディレクトリを削除できない        |
| 8      | ディレクトリを変更できない        |
| 9      | コピー操作に失敗した(入力元を開けない) |
| 10     | コピー操作に失敗した(出力先を開けない) |
| 11     | コピー操作に失敗した(書き込みできない) |
| 12     | ファイル名を変更できない         |

## 10.11 ネットワークイーサネットオブジェクト (0Ch)

### カテゴリ

高度

### オブジェクトの説明

このオブジェクトは、イーサネット固有の情報をアプリケーションに提供します。

### サポートされているコマンド

オブジェクト: Get\_Attribute

インスタンス: Get\_Attribute

### オブジェクトアトリビュート

| # | 名前                   | アクセス | データタイプ   | Value              |
|---|----------------------|------|----------|--------------------|
| 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   | -                  |

### インスタンスアトリビュート (インスタンス #1)

高度

| # | 名前                 | アクセス | 種類        | 説明                                                                        |
|---|--------------------|------|-----------|---------------------------------------------------------------------------|
| 1 | MAC Address        | Get  | UINT8 の配列 | 現在の MAC アドレス<br>147 ページ「Ethernet ホストオブジェクト (F9h)」を参照                      |
| 2 | Port 1 MAC Address | Get  | UINT8 の配列 | ポート 1 の MAC アドレス (LLDP プロトコルでは必須)<br>147 ページ「Ethernet ホストオブジェクト (F9h)」を参照 |
| 3 | Port 2 MAC Address | Get  | UINT8 の配列 | ポート 2 の MAC アドレス (LLDP プロトコルでは必須)<br>147 ページ「Ethernet ホストオブジェクト (F9h)」を参照 |

## 11. ホストアプリケーションオブジェクト

### 11.1 概要

この章では、本モジュールで実装されている ホストアプリケーションオブジェクトについて説明します。以下に示すオブジェクトをホストアプリケーションのファームウェアに実装することで、PROFINET の機能を拡張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 (FFh) (『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 (FEh) (『Anybus-CompactCom Software Design Guide』を参照)

ネットワーク固有オブジェクト：

- 123 ページ「PROFINET IO オブジェクト (F6h)」
- 147 ページ「Ethernet ホストオブジェクト (F9h)」
- 150 ページ「Energy コントロールオブジェクト (F0h)」

## 11.2 PROFINET IO オブジェクト (F6h)

### カテゴリ

基本、拡張、高度

### オブジェクトの説明

このオブジェクトを使用すると、ホストアプリケーションに PROFINET IO 関連の設定が実装されます。

このオブジェクトは任意に実装できます。これにより、ホストアプリケーションが以下のアトリビュートを全くサポートしないか、一部のアトリビュートをサポートするか、すべてのアトリビュートをサポートするかを選択できます。このモジュールは、起動時にこれらのアトリビュートの値を取得しようと試みます。値を取得しようとしたアトリビュートがホストアプリケーションに実装されていない場合、エラーメッセージ (06h、"Invalid CmdExt[0]") を返します。その場合、本モジュールはデフォルト値を使用します。

本モジュールが以下に示されていないアトリビュートの値を取得しようとした場合、エラーメッセージ (06h、"Invalid CmdExt[0]") を返します。

下記も参照してください。

- 70 ページ 「ネットワーク PROFINET IO オブジェクト (0Eh)」
- 159 ページ 「フローチャート - レコードデータへのアクセス」
- 『Anybus CompactCom Software Design Guide』の「Error Codes」

### サポートされているコマンド

オブジェクト :Get Attribute

|                       |                                              |
|-----------------------|----------------------------------------------|
| Get_Record            | (127 ページ 「コマンド詳細 :Get_Record」を参照)            |
| Set_Record            | (129 ページ 「コマンド詳細 :Set_Record」を参照)            |
| Get_IM_Record         | (131 ページ 「コマンド詳細 :Get_IM_Record」を参照)         |
| Set_IM_Record         | (133 ページ 「コマンド詳細 :Set_IM_Record」を参照)         |
| AR_Check_Ind          | (135 ページ 「コマンド詳細 :AR_Check_Ind」を参照)          |
| Cfg_Mismatch_Ind      | (137 ページ 「コマンド詳細 :Cfg_Mismatch_Ind」を参照)      |
| AR_Info_Ind           | (138 ページ 「コマンド詳細 :AR_Info_Ind」を参照)           |
| End_Of_Prm_Ind        | (140 ページ 「コマンド詳細 :End_Of_Prm_Ind」を参照)        |
| AR_Offline_Ind        | (141 ページ 「コマンド詳細 :AR_Offline_Ind」を参照)        |
| AR_Abort_Ind          | (143 ページ 「コマンド詳細 :AR_Abort_Ind」を参照)          |
| Plug_Submodule_Failed | (145 ページ 「コマンド詳細 :Plug_Submodule_Failed」を参照) |

インスタンス :Get Attribute

### オブジェクトアトリビュート

| # | 名前                   | アクセス | データタイプ   | Value         |
|---|----------------------|------|----------|---------------|
| 1 | Name                 | Get  | CHAR の配列 | "PROFINET IO" |
| 2 | Revision             | Get  | UINT8    | 01h           |
| 3 | Number of instances  | Get  | UINT16   | 0001h         |
| 4 | Highest instance no. | Get  | UINT16   | 0001h         |

## インスタンスアトリビュート (インスタンス #1)

### 基本

| # | 名前                              | アクセス | 種類       | デフォルト値 <sup>a</sup> | コメント                                    |
|---|---------------------------------|------|----------|---------------------|-----------------------------------------|
| 1 | Device ID                       | Get  | UINT16   | 0009h               | 装置を識別します。(メーカーにより割り当てられます)              |
| 2 | Vendor ID (I&M Manufacturer ID) | Get  | UINT16   | 010Ch               | 装置のメーカーを識別します。(PNOにより割り当てられます)          |
| 3 | Station Type                    | Get  | CHAR の配列 | "ABCC-PRT (2-Port)" | 装置の特性を表します。(メーカーにより割り当てられます); 最大 30 文字。 |

a. アトリビュートが実装されていない場合はこの値が使用されます。

### 拡張

| #  | 名前                              | アクセス | 種類        | デフォルト値 <sup>a</sup> | コメント                                                                                                                                                                 |
|----|---------------------------------|------|-----------|---------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7  | Record Data Mode                | Get  | UINT8     | 00h                 | この設定は、Record Data 要求の処理に影響を与えます。また、以下のようにビットフィールドを規定します：<br>ビット 0: インデックス 0 ~ 7FFFh:<br>0: 通常モード<br>1: 透過モード<br>ビット 1: インデックス AFF0h ~ AFFFh:<br>0: 通常モード<br>1: 透過モード |
| 8  | I&M Order ID                    | Get  | CHAR の配列  | "ABCC-PRT (2-Port)" | I&M0 のパラメーター：<br>装置のオーダー ID; 最大 20 文字。<br>未使用の文字には " " (スペース) が埋められる。                                                                                                |
| 17 | Port 1 MAC address <sup>b</sup> | Get  | UINT8 の配列 | -                   | 6 バイトからなる物理アドレス；<br>あらかじめプログラムされている<br>MAC アドレスが上書きされる。                                                                                                              |
| 18 | Port 2 MAC address <sup>b</sup> | Get  | UINT8 の配列 | -                   | 6 バイトからなる物理アドレス；<br>あらかじめプログラムされている<br>MAC アドレスが上書きされる。                                                                                                              |
| 23 | PROFInergy functionality        | Get  | UINT8     | 00h                 | PROFInergy 機能の各部分を有効/無効にするためのビットマスク。下表を参照。                                                                                                                           |

a. アトリビュートが実装されていない場合はこの値が使用されます。

b. 本モジュールには、有効な MAC アドレスがあらかじめプログラムされています。このアドレスを使用する場合は、このアトリビュートを実装しないでください。

PROFINET 機能の各部分を有効 / 無効にするためのビットマスク (アトリビュート #23) の説明を下表に示します。

| Bit   | Value | 説明                               |
|-------|-------|----------------------------------|
| 0     | 0     | PROFINET 機能は無効。ビット 1 ~ 7 は無視される。 |
|       | 1     | PROFINET 機能は有効                   |
| 1     | 0     | Query_Modes が無効                  |
|       | 1     | Query_modes をサポート                |
| 2     | 0     | PEM_Status が無効                   |
|       | 1     | PEM_Status が有効                   |
| 3     | 0     | PE_Identify が無効                  |
|       | 1     | PE_Identify が有効                  |
| 4 ~ 7 | (予約)  |                                  |

## 高度

| #  | 名前                          | アクセス | 種類       | デフォルト値 <sup>a</sup> | コメント                                                                                                                                                                                                                                                     |
|----|-----------------------------|------|----------|---------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 4  | MaxAr                       | Get  | UINT32   | 0003h               | 同時接続可能な最大の AR 数 (範囲 1 ~ 3)                                                                                                                                                                                                                               |
| 5  | (予約)                        | -    | -        | -                   | (将来のために予約)                                                                                                                                                                                                                                               |
| 6  | (予約)                        | -    | -        | -                   | (将来のために予約)                                                                                                                                                                                                                                               |
| 9  | I&M Serial Number           | Get  | CHAR の配列 | (製造時に割り当て)          | I&M0 のパラメーター: 装置のシリアル番号; 最大 16 文字。未使用の文字には " "(スペース) が埋められる。                                                                                                                                                                                             |
| 10 | I&M Hardware Revision       | Get  | UINT16   | (ハードウェアリビジョン)       | I&M0 のパラメーター: 装置のハードウェアリビジョン; FFFFh は、プロファイル固有の情報が利用可能なことを表す                                                                                                                                                                                             |
| 11 | I&M Software Revision       | Get  | CHAR の配列 | (ソフトウェアリビジョン)       | I&M0 のパラメーター: 装置のソフトウェアリビジョン。<br><u>バイト: 値: 意味:</u><br>0: "V" 正式リリース<br>"R" リビジョン<br>"P" プロトタイプ<br>"U" テスト中<br>"T" テスト装置<br>1: 0 ~ 255 メジャーバージョン<br>2: 0 ~ 255 マイナーバージョン<br>3: 0 ~ 255 内部の変更<br><b>注:</b> バージョン V255.255.255 は、プロファイル固有の情報が利用可能であることを表す。 |
| 12 | I&M Revision Counter        | Get  | UINT16   | 0000h               | I&M0 のパラメーター: 装置のリビジョンカウンター; この値が変更された場合、ハードウェアやそのパラメーターが変更されたことを表す。                                                                                                                                                                                     |
| 13 | I&M Profile ID <sup>b</sup> | Get  | UINT16   | F600h (汎用機器)        | I&M0 のパラメーター: アプリケーションが特定のプロファイルをサポートする場合、ここで指定可能。                                                                                                                                                                                                       |

| #  | 名前                                     | アクセス | 種類                     | デフォルト値 <sup>a</sup>                                            | コメント                                                                                                                                                                                                                                                                                                                         |
|----|----------------------------------------|------|------------------------|----------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 14 | I&M Profile Specific Type <sup>b</sup> | Get  | UINT16                 | 0004h<br>(通信モジュール)                                             | I&M0のパラメーター:<br>アプリケーションが特定のプロファイルをサポートする場合、プロファイル固有の種類をここで指定可能。                                                                                                                                                                                                                                                             |
| 15 | I&M Version                            | Get  | 構造:<br>UINT8,<br>UINT8 | 0101h                                                          | I&M0のパラメーター:<br>このパラメーターは、ホストアプリケーションがバージョン 1.1 以外の I&M をサポートする場合のみ使用すること。その場合、不明な I&M レコードへの要求はホストアプリケーションに転送されるため、ホストアプリケーションは Get_IM_Record コマンドおよび Set_IM_Record コマンドを処理できなければならない。                                                                                                                                      |
| 16 | I&M Supported                          | Get  | UINT16                 | 001Eh                                                          | I&M0のパラメーター:<br>スロット 0、サブスロット 1 に対し、装置がどの I&M パラメーターをサポートするかを定義するビットフィールド。<br><b>ビット:意味</b><br>0: プロファイル固有<br>1: I&M1 をサポート<br>2: I&M2 をサポート<br>3: I&M3 をサポート<br>4: I&M4 をサポート<br>5 ~ 15: I&M[5 ~ 15] をサポート<br><b>注:</b> I&M0 は必須です。無効にすることはできません。                                                                           |
| 19 | System Description                     | Get  | CHAR の配列               | "HMS Industrial Networks Anybus-CompactCom PROFINET IO 2-Port" | 最大 255 文字。                                                                                                                                                                                                                                                                                                                   |
| 20 | Interface Description                  | Get  | CHAR の配列               | "PROFINET IO interface"                                        | 最大 255 文字。                                                                                                                                                                                                                                                                                                                   |
| 21 | Module ID Assignment Mode              | Get  | UINT8                  | 0x00                                                           | 0 - デフォルト。モジュールの識別番号は、HMS の標準の GDS ファイルに基づいて生成される。<br>1 - インクリメンタル。モジュールの識別番号はシーケンシャルに生成される。スロット 1 に対しては 0x000000100 から始まる。<br><b>注 1:</b> ホストアプリケーションが Api_Add コマンドを発行した場合、このパラメーターは無効です。<br><b>注 2:</b> スロット 0 のデバイスアクセスポイントは、このパラメーターによって影響を受けません。<br><b>注 3:</b> シーケンシャルな番号生成を選択した場合、GSD ファイルの全てのモジュールが一意に識別されなければなりません。 |

| #  | 名前             | アクセス | 種類       | デフォルト値 <sup>a</sup> | コメント                                                   |
|----|----------------|------|----------|---------------------|--------------------------------------------------------|
| 22 | System Contact | Get  | CHAR の配列 | "www.anybus.com"    | この管理対象ノードの窓口担当者の ID、および窓口担当者への問い合わせ方法に関する情報。最大 255 文字。 |

- a. アトリビュートが実装されていない場合はこの値が使用されます。  
b. ホストアプリケーションが該当する機能を実装する必要があります ; Anybus モジュール自身では、このパラメーターに基づいた動作を変更しません。

## コマンド詳細 :Get\_Record

### カテゴリ

拡張

### 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

本モジュールは、以下の場合にこのコマンドを発行します。

- 0 (ゼロ) 以外の API に対する Record Data Read 要求をモジュールが受信したとき
- 0 (ゼロ) 以外のスロットインデックスに対する Record Data Read 要求をモジュールが受信したとき
- Record Data Read 要求をモジュールが受信し、I&M Version が 1.1 以外に設定されているとき
- API 0 に対する Record Data Read 要求をモジュールが受信し、対象のレコードが透過モードで処理されたとき

このコマンドの実装は必須ではありません。このコマンドが実装されない場合、元のネットワーク要求は拒否され、IO コントローラー / スーパーバイザーにエラーが返されます。

- コマンド詳細

| フィールド       | 内容                | コメント                         |
|-------------|-------------------|------------------------------|
| CmdExt[0]   | AR のハンドル (下位バイト)  | アプリケーションリレーションシップ (AR) のハンドル |
| CmdExt[1]   | AR のハンドル (上位バイト)  |                              |
| MsgData[0]  | API (下位ワード、下位バイト) | アプリケーションプロセスインスタンス (API)     |
| MsgData[1]  | API (下位ワード、上位バイト) |                              |
| MsgData[2]  | API (上位ワード、下位バイト) |                              |
| MsgData[3]  | API (上位ワード、上位バイト) |                              |
| MsgData[4]  | スロット (下位バイト)      |                              |
| MsgData[5]  | スロット (上位バイト)      |                              |
| MsgData[6]  | サブスロット (下位バイト)    | 要求のサブスロット番号                  |
| MsgData[7]  | サブスロット (上位バイト)    |                              |
| MsgData[8]  | インデックス (下位バイト)    | 要求のインデックス                    |
| MsgData[9]  | インデックス (上位バイト)    |                              |
| MsgData[10] | 長さ                | 範囲 : 1 ~ 244                 |

- 応答詳細 (成功)

| フィールド          | 内容               | コメント                          |
|----------------|------------------|-------------------------------|
| CmdExt[0 ~ 1]  | (予約)             | (ゼロに設定)                       |
| MsgData[0 ~ n] | データ (最大 244 バイト) | Record Data Read の応答にて返されるデータ |

- 応答詳細 (エラー)

| フィールド         | 内容       | コメント                           |
|---------------|----------|--------------------------------|
| CmdExt[0 ~ 1] | (予約)     | (ゼロに設定)                        |
| MsgData[0]    | FFh      | オブジェクト固有のエラー                   |
| MsgData[1]    | エラーコード 1 | 146 ページ「詳細: エラーコード 1」を参照       |
| MsgData[2]    | エラーコード 2 | ユーザー固有のエラーコード                  |
| MsgData[3]    | 追加データ 1  | API 固有。追加データ 1 を定義しない場合はゼロを設定。 |
| MsgData[4]    | 追加データ 2  | ユーザー固有。追加データ 2 を定義しない場合はゼロを設定。 |

下記も参照してください。

- 129 ページ 「コマンド詳細 :Set\_Record」
- 131 ページ 「コマンド詳細 :Get\_IM\_Record」
- 133 ページ 「コマンド詳細 :Set\_IM\_Record」
- 159 ページ 「フローチャート - レコードデータへのアクセス」

## コマンド詳細 :Set\_Record

### カテゴリ

拡張

### 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

本モジュールは、以下の場合にこのコマンドを発行します。

- 0 (ゼロ) 以外の API に対する Record Data Write 要求をモジュールが受信したとき
- 0 (ゼロ) 以外のスロットインデックスに対する Record Data Write 要求をモジュールが受信したとき
- Record Data Write 要求をモジュールが受信し、I&M Version が 1.1 以外に設定されているとき
- API 0 に対する Record Data Write 要求をモジュールが受信し、対象のレコードが透過モードで処理されたとき

このコマンドの実装は必須ではありません。このコマンドが実装されない場合、元のネットワーク要求は拒否され、IO コントローラー/スーパーバイザーにエラーが返されます。

- コマンド詳細

| フィールド           | 内容                | コメント                         |
|-----------------|-------------------|------------------------------|
| CmdExt[0]       | AR のハンドル (下位バイト)  | アプリケーションリレーションシップ (AR) のハンドル |
| CmdExt[1]       | AR のハンドル (上位バイト)  |                              |
| MsgData[0]      | API (下位ワード、下位バイト) | アプリケーションプロセスインスタンス (API)     |
| MsgData[1]      | API (下位ワード、上位バイト) |                              |
| MsgData[2]      | API (上位ワード、下位バイト) |                              |
| MsgData[3]      | API (上位ワード、上位バイト) |                              |
| MsgData[4]      | スロット (下位バイト)      | 要求のスロット番号                    |
| MsgData[5]      | スロット (上位バイト)      |                              |
| MsgData[6]      | サブスロット (下位バイト)    | 要求のサブスロット番号                  |
| MsgData[7]      | サブスロット (上位バイト)    |                              |
| MsgData[8]      | インデックス (下位バイト)    | 要求のインデックス                    |
| MsgData[9]      | インデックス (上位バイト)    |                              |
| MsgData[10]     | (予約)              | 予約;ゼロに設定。                    |
| MsgData[11 ~ n] | データ (最大 244 バイト)  | Record Data Write 要求からのデータ。  |

- 応答詳細 (成功)

(データなし)

- 応答詳細 (エラー)

| フィールド         | 内容       | コメント                           |
|---------------|----------|--------------------------------|
| CmdExt[0 ~ 1] | (予約)     | (ゼロに設定)                        |
| MsgData[0]    | FFh      | オブジェクト固有のエラー                   |
| MsgData[1]    | エラーコード 1 | 146 ページ「詳細:エラーコード 1」を参照        |
| MsgData[2]    | エラーコード 2 | ユーザー固有のエラーコード                  |
| MsgData[3]    | 追加データ 1  | API 固有。追加データ 1 を定義しない場合はゼロを設定。 |
| MsgData[4]    | 追加データ 2  | ユーザー固有。追加データ 2 を定義しない場合はゼロを設定。 |

下記も参照してください。

- 127 ページ 「コマンド詳細 :Get\_Record」
- 131 ページ 「コマンド詳細 :Get\_IM\_Record」
- 133 ページ 「コマンド詳細 :Set\_IM\_Record」
- 159 ページ 「フローチャート - レコードデータへのアクセス」

## コマンド詳細 :Get\_IM\_Record

### カテゴリ

拡張

### 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

このコマンドは、ホストアプリケーションから I&M の情報を取得します。このコマンドは、以下の場合に発行されます。

- 不明な I&M インデックスに対する要求をモジュールが受信したとき
- I&M 要求をモジュールが受信し、透過モードが有効になっているとき (インデックス AFF0h ~ AFFFh)

このコマンドの実装は必須ではありません。このコマンドが実装されない場合、元のネットワーク要求は拒否され、IO コントローラー/スーパーバイザーにエラーが返されます。

- コマンド詳細

| フィールド      | 内容             | コメント                                                                                                     |
|------------|----------------|----------------------------------------------------------------------------------------------------------|
| CmdExt[0]  | IM レコードインデックス  | 有効範囲 :0 ~ 15<br>値 : 意味 :<br>0: I&M0<br>1: I&M1<br>2: I&M2<br>3: I&M3<br>4: I&M4<br>5 ~ 15: I&M の追加機能用に予約 |
| CmdExt[1]  | (予約)           | (無視)                                                                                                     |
| MsgData[0] | スロット (下位バイト)   | 要求のスロット番号                                                                                                |
| MsgData[1] | スロット (上位バイト)   |                                                                                                          |
| MsgData[2] | サブスロット (下位バイト) | 要求のサブスロット番号                                                                                              |
| MsgData[3] | サブスロット (上位バイト) |                                                                                                          |

- 応答詳細 (成功)

| フィールド            | 内容      | コメント                   |
|------------------|---------|------------------------|
| CmdExt[0]        | (予約)    | (ゼロに設定)                |
| CmdExt[1]        |         |                        |
| MsgData [0 ~ 9]  | (予約)    | -                      |
| MsgData [10 ~ n] | I&M データ | 22 ページ 「I&M のデータ構造」を参照 |

- 応答詳細 (エラー)

| フィールド      | 内容       | コメント                           |
|------------|----------|--------------------------------|
| CmdExt[0]  | (予約)     | (ゼロに設定)                        |
| CmdExt[1]  |          |                                |
| MsgData[0] | FFh      | オブジェクト固有のエラー                   |
| MsgData[1] | エラーコード 1 | 146 ページ 「詳細: エラーコード 1」を参照      |
| MsgData[2] | エラーコード 2 | ユーザー固有のエラーコード                  |
| MsgData[3] | 追加データ 1  | API 固有。追加データ 1 を定義しない場合はゼロを設定。 |
| MsgData[4] | 追加データ 2  | ユーザー固有。追加データ 2 を定義しない場合はゼロを設定。 |

下記も参照してください。

- 127 ページ 「コマンド詳細 :Get\_Record」
- 129 ページ 「コマンド詳細 :Set\_Record」
- 133 ページ 「コマンド詳細 :Set\_IM\_Record」
- 159 ページ 「フローチャート - レコードデータへのアクセス」

## コマンド詳細 :Set\_IM\_Record

### カテゴリ

拡張

### 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

このコマンドは、ホストアプリケーションに対して I&M の情報を指定します。このコマンドは、以下の場合に発行されます。

- 不明な I&M インデックスに対する要求をモジュールが受信したとき
- I&M 要求をモジュールが受信し、透過モードが有効になっているとき (インデックス AFF0h ~ AFFFh)

このコマンドの実装は必須ではありません。このコマンドが実装されない場合、元のネットワーク要求は拒否され、IO コントローラ/スーパーバイザーにエラーが返されます。

- コマンド詳細

| フィールド           | 内容             | コメント                                                                                                     |
|-----------------|----------------|----------------------------------------------------------------------------------------------------------|
| CmdExt[0]       | IM レコードインデックス  | 有効範囲 :0 ~ 15<br>値 : 意味 :<br>0: I&M0<br>1: I&M1<br>2: I&M2<br>3: I&M3<br>4: I&M4<br>5 ~ 15: I&M の追加機能用に予約 |
| CmdExt[1]       | (予約)           | (無視)                                                                                                     |
| MsgData[0]      | スロット (下位バイト)   | 要求のスロット番号                                                                                                |
| MsgData[1]      | スロット (上位バイト)   |                                                                                                          |
| MsgData[2]      | サブスロット (下位バイト) | 要求のサブスロット番号                                                                                              |
| MsgData[3]      | サブスロット (上位バイト) |                                                                                                          |
| MsgData[4 ~ 13] | (予約)           | (無視)                                                                                                     |
| MsgData[14 ~ n] | I&M データ        | 22 ページ 「I&M のデータ構造」を参照                                                                                   |

- 応答詳細 (成功)

(データなし)

- 応答詳細 (エラー)

| フィールド      | 内容       | コメント                           |
|------------|----------|--------------------------------|
| CmdExt[0]  | (予約)     | (ゼロに設定)                        |
| CmdExt[1]  |          |                                |
| MsgData[0] | FFh      | オブジェクト固有のエラー                   |
| MsgData[1] | エラーコード 1 | 146 ページ 「詳細: エラーコード 1」を参照      |
| MsgData[2] | エラーコード 2 | ユーザー固有のエラーコード                  |
| MsgData[3] | 追加データ 1  | API 固有。追加データ 1 を定義しない場合はゼロを設定。 |
| MsgData[4] | 追加データ 2  | ユーザー固有。追加データ 2 を定義しない場合はゼロを設定。 |

下記も参照してください。

- 127 ページ 「コマンド詳細 :Get\_Record」
- 129 ページ 「コマンド詳細 :Set\_Record」
- 131 ページ 「コマンド詳細 :Get\_IM\_Record」
- 159 ページ 「フローチャート - レコードデータへのアクセス」

## コマンド詳細 :AR\_Check\_Ind

### カテゴリ

高度

### 詳細

コマンドコード: 14h

有効な対象: オブジェクトインスタンス

### 説明

本モジュールは、このコマンドを発行することで、アプリケーションリレーションシップ (AR) が確立されたことをホストアプリケーションに通知します。このコマンドの実装は必須ではありません。

- コマンド詳細

| フィールド      | 内容                    | コメント                                                                                                                                                   |
|------------|-----------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------|
| CmdExt[0]  | AR のハンドル (下位バイト)      | アプリケーションリレーションシップ (AR) のハンドル                                                                                                                           |
| CmdExt[1]  | AR のハンドル (上位バイト)      |                                                                                                                                                        |
| MsgData[0] | IP アドレス (下位ワード、下位バイト) | リモートステーションの IP アドレス (IO コントローラー / スーパーバイザー)                                                                                                            |
| MsgData[1] | IP アドレス (下位ワード、上位バイト) |                                                                                                                                                        |
| MsgData[2] | IP アドレス (上位ワード、下位バイト) |                                                                                                                                                        |
| MsgData[3] | IP アドレス (上位ワード、上位バイト) |                                                                                                                                                        |
| MsgData[4] | AR の種類 (下位バイト)        | 以下のように AR の種類を表す:<br><u>値:意味:</u><br>1: IO_AR_SINGLE<br>3: IO_AR_CIR<br>4: IO_AR_REDUNDANT_CONTROLLER<br>5: IO_AR_REDUNDANT_DEVICE<br>6: SUPERVISOR_AR |
| MsgData[5] | AR の種類 (上位バイト)        |                                                                                                                                                        |

| フィールド               | 内容                    | コメント                                                                                                                                                                                                                                                                                                               |
|---------------------|-----------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| MsgData[6]          | ARのプロパティ(下位ワード、下位バイト) | 以下のようにARのプロパティを表すビットフィールド:                                                                                                                                                                                                                                                                                         |
| MsgData[7]          | ARのプロパティ(下位ワード、上位バイト) | <u>ビット0~2: 状態:</u><br>0: バックアップ<br>1: プライマリ                                                                                                                                                                                                                                                                        |
| MsgData[8]          | ARのプロパティ(上位ワード、下位バイト) | <u>ビット3: スーパーバイザーによる操作:</u><br>0: 禁止<br>1: 許可                                                                                                                                                                                                                                                                      |
| MsgData[9]          | ARのプロパティ(上位ワード、上位バイト) | <u>ビット4: サーバーのパラメーター化:</u><br>0: EXTERNAL_PRM_SERVER<br>1: CM_INITIATOR<br><u>ビット5~6: データレート:</u><br>0: AT_LEAST_100 Mbps<br>1: 100 Mbps<br>2: 1 Gbps<br>3: 10 Gbps<br><u>ビット8: デバイスアクセス:</u><br>0: AR_CONTEXT<br>1: DEVICE_CONTEXT<br><u>ビット9~10: コンパニオンAR:</u><br>0: SINGLE_AR<br>1: FIRST_AR<br>2: COMPANION_AR |
| MsgData[10]         | リモートステーション名の長さ        | バイト単位で表したリモートステーション名の長さ                                                                                                                                                                                                                                                                                            |
| MsgData<br>[11 ~ n] | リモートステーション名           | リモートステーション名 (IO コントローラー / スーパーバイザー)                                                                                                                                                                                                                                                                                |

- 応答詳細  
(データなし)

## コマンド詳細 :Cfg\_Mismatch\_Ind

### カテゴリ

高度

### 詳細

コマンドコード: 15h

有効な対象: オブジェクトインスタンス

### 説明

本モジュールは、このコマンドを発行することで、IO コントローラーにおける設定 (Expected Identification) がホストアプリケーションで定義された設定 (Real Identification) と一致しないことをホストアプリケーションに通知します。

このコマンドの実装は必須ではありません。

- コマンド詳細

| フィールド       | 内容                                | コメント                                                      |
|-------------|-----------------------------------|-----------------------------------------------------------|
| CmdExt[0]   | AR のハンドル (下位バイト)                  | アプリケーションリレー<br>ションシップ (AR) のハン<br>ドル                      |
| CmdExt[1]   | AR のハンドル (上位バイト)                  |                                                           |
| MsgData[0]  | API (下位ワード、下位バイト)                 | アプリケーションプロセス<br>インスタンス (API)                              |
| MsgData[1]  | API (下位ワード、上位バイト)                 |                                                           |
| MsgData[2]  | API (上位ワード、下位バイト)                 |                                                           |
| MsgData[3]  | API (上位ワード、上位バイト)                 |                                                           |
| MsgData[4]  | スロット (下位バイト)                      | 不一致が生じているスロ<br>ット番号                                       |
| MsgData[5]  | スロット (上位バイト)                      |                                                           |
| MsgData[6]  | サブスロット (下位バイト)                    | 不一致が生じているサブ<br>スロット番号                                     |
| MsgData[7]  | サブスロット (上位バイト)                    |                                                           |
| MsgData[8]  | 期待されるモジュール ID (下位ワード、下位バイト)       | IO コントローラーの設定よ<br>り得られたモジュール ID<br>(GSD ファイルの記述と同<br>じ)   |
| MsgData[9]  | 期待されるモジュール ID (下位ワード、上位バイト)       |                                                           |
| MsgData[10] | 期待されるモジュール ID (上位ワード、下位バイト)       |                                                           |
| MsgData[11] | 期待されるモジュール ID (上位ワード、上位バイト)       |                                                           |
| MsgData[12] | 期待されるサブモジュール ID (下位ワード、下位バ<br>イト) | IO コントローラーの設定よ<br>り得られたサブモジュール<br>ID (GSD ファイルの記述と<br>同じ) |
| MsgData[13] | 期待されるサブモジュール ID (下位ワード、上位バ<br>イト) |                                                           |
| MsgData[14] | 期待されるサブモジュール ID (上位ワード、下位バ<br>イト) |                                                           |
| MsgData[15] | 期待されるサブモジュール ID (上位ワード、上位バ<br>イト) |                                                           |

- 応答詳細  
(データなし)

下記も参照してください。

- 19 ページ 「Real Identification (RI)」 (20 ページ 「設定の不一致」)
- 160 ページ 「フローチャート - 設定の不一致 (RI)」

## コマンド詳細 :AR\_Info\_Ind

### カテゴリ

高度

### 詳細

コマンドコード: 16h

有効な対象: オブジェクトインスタンス

### 説明

本モジュールは、このコマンドを発行することで、IO コントローラーが AR の確立に使用する Expected Identification (モジュール/サブモジュールのリスト) をホストアプリケーションに通知します。

この情報は複数のセグメントに分割される場合があります。そのため、本モジュールによりこのコマンドが複数回発行されます。このとき、コマンドを発行するたびに、分割された設定の各部分が送信されます。

このコマンドの実装は必須ではありません。

- コマンド詳細

| フィールド           | 内容                | コメント                                                                                                                                                           |
|-----------------|-------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------|
| CmdExt[0]       | AR のハンドル (下位バイト)  | アプリケーションリレーションシップ (AR) のハンドル                                                                                                                                   |
| CmdExt[1]       | AR のハンドル (上位バイト)  |                                                                                                                                                                |
| MsgData[0]      | 現在のセグメント (下位バイト)  | 現在のセグメント番号; このワードの値が "トータルセグメント" (下記) に等しい場合、全てのデータが転送されたことを意味する。                                                                                              |
| MsgData[1]      | 現在のセグメント (上位バイト)  |                                                                                                                                                                |
| MsgData[2]      | トータルセグメント (下位バイト) |                                                                                                                                                                |
| MsgData[3]      | トータルセグメント (上位バイト) |                                                                                                                                                                |
| MsgData [4 ~ n] | データフィールド          | データフィールドのイニシャルブロックにおける最初の 2 バイトは、設定に含まれるモジュールの数を表す。各モジュールは、モジュールブロックおよびそれに続くサブモジュールブロックにより表される (対象のモジュールがサブモジュールを含む場合)。データフィールドのコーディングについては 139 ページ「データ形式」を参照。 |

- 応答詳細  
(データなし)

## データ形式

データを全て受信すると、そのデータは以下のように解釈されます。

| 種類     | 名前             | 説明                                            |
|--------|----------------|-----------------------------------------------|
| UINT16 | iNbrApi        | 設定に含まれる API の数                                |
| UINT32 | iApiNbr        | API 番号および API のモジュールブロック数が格納されたイニシャルモジュールブロック |
| UINT16 | iNbrMod        |                                               |
| UINT16 | iSlotNbr       | モジュールブロック (8 バイト)。下記を参照。                      |
| UINT16 | iNbrSubMod     |                                               |
| UINT32 | lModIdent      |                                               |
| UINT16 | iSubSlotNbr    | サブモジュールブロック (10 バイト)。下記を参照。                   |
| UINT32 | lSubModIdent   |                                               |
| UINT16 | iInDataLength  |                                               |
| UINT16 | iOutDataLength |                                               |

イニシャル API ブロック (iNbrApi) は、設定に含まれるモジュールの数を定義します。

各 API はイニシャルモジュールブロックを持っており、このブロックには、API 番号 (iApiNbr) および API に含まれるモジュール (またはスロット) の数に関する情報が格納されます。

各モジュールはモジュールブロックで始まります。このブロックには、スロット番号、サブモジュール (またはサブスロット) の数、モジュールの ID 番号が格納されます。

最後に、各サブモジュールブロックには、サブスロット番号、サブモジュールの ID 番号、入出力データ長が格納されます。

例:

この例では、以下のプロパティを持つ 2 つの API が設定に含まれます。

- API #1 は 2 つのモジュールを持ち、1 つ目のモジュールは 2 つのサブモジュールを、2 つ目のモジュールは 1 つのサブモジュールを持ちます。
- API #2 は、1 つのサブモジュールを持つ 1 つのモジュールを持ちます。

|                               |                    |              |
|-------------------------------|--------------------|--------------|
| Initial API Block             | No. of APIs        | 0002h        |
| Initial Module Block (API #0) | API no.            | 00 00 00 00h |
|                               | No. of Modules     | 0002h        |
| Module Block (Module #1)      | Slot no.           | 0001h        |
|                               | No. of Submodules  | 0003h        |
|                               | Module ID          | 4A 6F 48 62h |
| Submodule Block (Module #1)   | Subslot no.        | 0001h        |
|                               | Submodule ID       | 65 6C 69 65h |
|                               | Input Data Length  | 0004h        |
|                               | Output Data Length | 0010h        |
| Submodule Block (Module #1)   | Subslot no.        | 0002h        |
|                               | Submodule ID       | 76 65 73 69h |
|                               | Input Data Length  | 0008h        |
|                               | Output Data Length | 0002h        |
| Submodule Block (Module #1)   | Subslot no.        | 0003h        |
|                               | Submodule ID       | 6E 53 61 6Eh |
|                               | Input Data Length  | 0008h        |
|                               | Output Data Length | 0002h        |
| Module Block (Module #2)      | Slot no.           | 0002h        |
|                               | No. of Submodules  | 0001h        |
|                               | Module ID          | 74 61 43 6Ch |
| Submodule Block (Module #2)   | Subslot no.        | 0001h        |
|                               | Submodule ID       | 61 75 73 21h |
|                               | Input Data Length  | 0010h        |
|                               | Output Data Length | 0001h        |
| Initial Module Block (API #2) | API no.            | 00 00 00 02h |
|                               | No. of Modules     | 0001h        |
| Module Block (Module #1)      | Slot no.           | 0001h        |
|                               | No. of Submodules  | 0002h        |
|                               | Module ID          | 4A 6F 48 82h |
| Submodule Block (Module #1)   | Subslot no.        | 0001h        |
|                               | Submodule ID       | 65 6C 67 65h |
|                               | Input Data Length  | 0004h        |
|                               | Output Data Length | 0010h        |
| Submodule Block (Module #1)   | Subslot no.        | 0002h        |
|                               | Submodule ID       | 76 65 74 69h |
|                               | Input Data Length  | 0008h        |
|                               | Output Data Length | 0002h        |

## コマンド詳細 :End\_Of\_Prm\_Ind

### カテゴリ

高度

### 詳細

コマンドコード: 17h

有効な対象: オブジェクトインスタンス

### 説明

本モジュールは、このコマンドを発行することで、パラメーター化のフェーズが完了したことをホストアプリケーションに通知します。このコマンドの実装は必須ではありません。

このコマンドを実装した場合、ホストアプリケーションは、このコマンドへの応答に基づいて、後のステージで "Appl\_State\_Ready" を発行してデータ交換の準備ができたことを示す必要があります。このコマンドを実装しない場合、この処理は本モジュールにより自動的に行われます。

#### • コマンド詳細

| フィールド      | 内容                | コメント                                                                                                 |
|------------|-------------------|------------------------------------------------------------------------------------------------------|
| CmdExt[0]  | AR のハンドル (下位バイト)  | アプリケーションリレーションシップ (AR) のハンドル                                                                         |
| CmdExt[1]  | AR のハンドル (上位バイト)  |                                                                                                      |
| MsgData[0] | API (下位ワード、下位バイト) | アプリケーションプロセスインスタンス (API)<br>- サブスロット > 0 のときのみ有効                                                     |
| MsgData[1] | API (下位ワード、上位バイト) |                                                                                                      |
| MsgData[2] | API (上位ワード、下位バイト) |                                                                                                      |
| MsgData[3] | API (上位ワード、上位バイト) |                                                                                                      |
| MsgData[4] | スロット (下位バイト)      | コマンドにより影響を受けるスロットの番号<br>- サブスロット > 0 のときのみ有効                                                         |
| MsgData[5] | スロット (上位バイト)      |                                                                                                      |
| MsgData[6] | サブスロット (下位バイト)    | コマンドにより影響を受けるサブスロットの番号<br>値:意味:<br>0: コマンドは設定における全てのモジュールに適用される<br>その他: コマンドは指定したスロット/サブスロットにのみ適用される |
| MsgData[7] | サブスロット (上位バイト)    |                                                                                                      |

#### • 応答詳細

| フィールド      | 内容          | コメント                                                                                       |
|------------|-------------|--------------------------------------------------------------------------------------------|
| CmdExt[0]  | (予約)        | (ゼロに設定)                                                                                    |
| CmdExt[1]  |             |                                                                                            |
| MsgData[0] | アプリケーションの状態 | 値:意味:<br>0: データ交換の準備ができていない<br>1: データ交換の準備ができていない<br>(後のステージで Appl_State_Ready を発行する必要がある) |

下記も参照してください。

- 78 ページ 「コマンド詳細 :Appl\_State\_Ready」

## コマンド詳細 :AR\_Offline\_Ind

### カテゴリ

高度

### 詳細

コマンドコード: 18h

有効な対象: オブジェクトインスタンス

### 説明

本モジュールは、このコマンドを発行することで、モジュールがオフライン状態になったことをホストアプリケーションに通知します。このコマンドの実装は必須ではありません。

- コマンド詳細

| フィールド     | 内容                | コメント                       |
|-----------|-------------------|----------------------------|
| CmdExt[0] | AR handle (下位バイト) | アプリケーションリレーションシップ (AR) のハン |
| CmdExt[1] | AR のハンドル (上位バイト)  | ドル                         |

| フィールド      | 内容            | コメント                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|------------|---------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| MsgData[0] | 理由コード (下位バイト) | オフラインに遷移した理由コード                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| MsgData[1] | 理由コード (上位バイト) | <p>値:理由:</p> <p>0: 理由なし (不明な理由)</p> <p>3: メモリ不足</p> <p>4: プロバイダーまたはコンシューマーの追加に失敗した</p> <p>5: ミス (コンシューマー)</p> <p>6: Cmi タイムアウト</p> <p>7: Alarm のオープンに失敗した</p> <p>8: Alarm-send.cnf(-)</p> <p>9: Alarm-ack-send.cnf(-)</p> <p>10: Alarm データが長すぎる</p> <p>11: Alarm.ind(err)</p> <p>12: Rpc-client call.cnf(-)</p> <p>13: Ar-abort.req</p> <p>14: Re-run アボートが存在する</p> <p>15: release.ind を受け取った</p> <p>16: 装置がパッシブになった</p> <p>17: 装置 /Ar が削除された</p> <p>18: プロトコル違反</p> <p>19: NARE エラー</p> <p>20: RPC バインドエラー</p> <p>21: RPC 接続エラー</p> <p>22: RPC 読み取りエラー</p> <p>23: RPC 書き込みエラー</p> <p>24: RPC 制御エラー</p> <p>25: check.rsp の後、in-data.ind の前は、取り外しまたは埋め込みはできない</p> <p>26: AP が削除された</p> <p>27: リンクダウン</p> <p>28: マルチキャスト MAC を登録できない</p> <p>29: 同期していない (コンパニオン AR を開始できない)</p> <p>30: トポロジーが間違っている (コンパニオン AR を開始できない)</p> <p>31: DCP、ステーション名が変更された</p> <p>32: DCP、工場出荷状態にリセットされた</p> <p>33: パラメーターにエラーがあるため、コンパニオン AR を開始できない</p> |

- 応答詳細  
(データなし)

## コマンド詳細 :AR\_Abort\_Ind

### カテゴリ

高度

### 詳細

コマンドコード: 19h

有効な対象: オブジェクトインスタンス

### 説明

本モジュールは、このコマンドを発行することで、(アプリケーションなどにより)アプリケーションリレーションシップ (AR) が中断されたことをホストアプリケーションに通知します。

このコマンドの実装は必須ではありません。

- コマンド詳細

| フィールド     | 内容                | コメント                       |
|-----------|-------------------|----------------------------|
| CmdExt[0] | AR handle (下位バイト) | アプリケーションリレーションシップ (AR) のハン |
| CmdExt[1] | AR のハンドル (上位バイト)  | ドル                         |

| フィールド      | 内容            | コメント                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|------------|---------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| MsgData[0] | 理由コード (下位バイト) | オフラインに遷移した理由コード                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| MsgData[1] | 理由コード (上位バイト) | <p>値:理由:</p> <p>0: 理由なし (不明な理由)</p> <p>3: メモリ不足</p> <p>4: プロバイダーまたはコンシューマーの追加に失敗した</p> <p>5: ミス (コンシューマー)</p> <p>6: Cmi タイムアウト</p> <p>7: Alarm のオープンに失敗した</p> <p>8: Alarm-send.cnf(-)</p> <p>9: Alarm-ack-send.cnf(-)</p> <p>10: Alarm データが長すぎる</p> <p>11: Alarm.ind(err)</p> <p>12: Rpc-client call.cnf(-)</p> <p>13: Ar-abort.req</p> <p>14: Re-run アボートが存在する</p> <p>15: release.ind を受け取った</p> <p>16: 装置がパッシブになった</p> <p>17: 装置 /Ar が削除された</p> <p>18: プロトコル違反</p> <p>19: NARE エラー</p> <p>20: RPC バインドエラー</p> <p>21: RPC 接続エラー</p> <p>22: RPC 読み取りエラー</p> <p>23: RPC 書き込みエラー</p> <p>24: RPC 制御エラー</p> <p>25: check.rsp の後、in-data.ind の前は、取り外しまたは埋め込みはできない</p> <p>26: AP が削除された</p> <p>27: リンクダウン</p> <p>28: マルチキャスト MAC を登録できない</p> <p>29: 同期していない (コンパニオン AR を開始できない)</p> <p>30: トポロジーが間違っている (コンパニオン AR を開始できない)</p> <p>31: DCP、ステーション名が変更された</p> <p>32: DCP、工場出荷状態にリセットされた</p> <p>33: パラメーターにエラーがあるため、コンパニオン AR を開始できない</p> |

- 応答詳細  
(データなし)

## コマンド詳細 :Plug\_Submodule\_Failed

### カテゴリ

高度

### 詳細

コマンドコード: 1Ah

有効な対象: オブジェクトインスタンス

### 説明

本モジュールは、このコマンドを発行することで、ネットワーク PROFINET IO オブジェクトへの Plug\_Submodule が失敗したことを通知します。

このコマンドの実装は必須ではありません。

- コマンド詳細

| フィールド      | 内容                | コメント                     |
|------------|-------------------|--------------------------|
| CmdExt[0]  | (予約)              | (無視)                     |
| CmdExt[1]  |                   |                          |
| MsgData[0] | API (下位ワード、下位バイト) | アプリケーションプロセスインスタンス (API) |
| MsgData[1] | API (下位ワード、上位バイト) |                          |
| MsgData[2] | API (上位ワード、下位バイト) |                          |
| MsgData[3] | API (上位ワード、上位バイト) |                          |
| MsgData[4] | スロット (下位バイト)      | 要求のスロット番号                |
| MsgData[5] | スロット (上位バイト)      |                          |
| MsgData[6] | サブスロット (下位バイト)    | 要求のサブスロット番号              |
| MsgData[7] | サブスロット (上位バイト)    |                          |

- 応答詳細  
(データなし)

下記も参照してください。

- 75 ページ 「コマンド詳細 :Pull\_Module」
- 73 ページ 「コマンド詳細 :Plug\_Submodule」
- 75 ページ 「コマンド詳細 :Pull\_Module」
- 76 ページ 「コマンド詳細 :Pull\_Submodule」

## 詳細：エラーコード 1

以下のコマンドに対してエラー応答を返す場合、下記のエラーコードを使用します。

- Get\_Record
- Set\_Record
- Get\_IM\_Record
- Set\_IM\_Record

| 上位ニブル (ビット 4 ~ 7) |          | 下位ニブル (ビット 0 ~ 3) |                  | コメント |
|-------------------|----------|-------------------|------------------|------|
| エラークラス            | 意味       | エラーコード            | 意味               |      |
| 0 ~ 9             | 予約       | (予約)              | (予約)             | -    |
| 10                | アプリケーション | 0                 | 読み取りエラー          | -    |
|                   |          | 1                 | 書き込みエラー          | -    |
|                   |          | 2                 | モジュールエラー         | -    |
|                   |          | 3 ~ 6             | (予約)             | -    |
|                   |          | 7                 | ビジー              | -    |
|                   |          | 8                 | バージョンが一致しない      | -    |
|                   |          | 9                 | 機能がサポートされていない    | -    |
|                   |          | 10 ~ 15           | ユーザー固有           | -    |
| 11                | アクセス     | 0                 | 無効なインデックス        | -    |
|                   |          | 1                 | 書き込み長エラー         | -    |
|                   |          | 2                 | 無効なスロット / サブスロット | -    |
|                   |          | 3                 | 種類が一致しない         | -    |
|                   |          | 4                 | 無効な領域            | -    |
|                   |          | 5                 | 状態が一致しない         | -    |
|                   |          | 6                 | アクセスが拒否された       | -    |
|                   |          | 7                 | 無効な範囲            | -    |
|                   |          | 8                 | 無効なパラメーター        | -    |
|                   |          | 9                 | 無効な種類            | -    |
|                   |          | 10                | バックアップ           | -    |
|                   |          | 11 ~ 15           | ユーザー固有           | -    |
| 12                | リソース     | 0                 | 読み込みの制約が一致しない    | -    |
|                   |          | 1                 | 書き込みの制約が一致しない    | -    |
|                   |          | 2                 | リソースがビジー         | -    |
|                   |          | 3                 | リソースが利用できない      | -    |
|                   |          | 4 ~ 7             | (予約)             | -    |
|                   |          | 8 ~ 15            | ユーザー固有           | -    |
| 13 ~ 15           | ユーザー固有   | (ユーザー固有)          | ユーザー固有           | -    |

下記も参照してください。

- 127 ページ 「コマンド詳細 :Get\_Record」
- 129 ページ 「コマンド詳細 :Set\_Record」
- 131 ページ 「コマンド詳細 :Get\_IM\_Record」
- 133 ページ 「コマンド詳細 :Set\_IM\_Record」

## 11.3 Ethernet ホストオブジェクト (F9h)

### カテゴリ

拡張、高度

### オブジェクトの説明

このオブジェクトを使用すると、ホストアプリケーションにイーサネットの機能が実装されます。

### サポートされているコマンド

オブジェクト:    Get\_Attribute

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

## オブジェクトアトリビュート

| # | 名前                   | アクセス | データタイプ   | Value      |
|---|----------------------|------|----------|------------|
| 1 | Name                 | Get  | CHAR の配列 | "Ethernet" |
| 2 | Revision             | Get  | UINT8    | 02h        |
| 3 | Number of instances  | Get  | UINT16   | 0001h      |
| 4 | Highest instance no. | Get  | UINT16   | 0001h      |

## インスタンスアトリビュート (インスタンス #1)

### 拡張

| # | 名前                    | アクセス | 種類     | デフォルト <sup>a</sup> | コメント                                                                 |
|---|-----------------------|------|--------|--------------------|----------------------------------------------------------------------|
| 2 | Enable HICP           | Get  | BOOL   | True               | HICP を有効 / 無効にする <sup>b</sup><br>(162 ページ「HICP (ホスト IP 設定プロトコル)」を参照) |
| 3 | Enable Web Server     | Get  | BOOL   | True               | Web サーバーを有効 / 無効にする <sup>b</sup><br>(29 ページ「Web サーバー」を参照)            |
| 5 | Enable Web ADI access | Get  | BOOL   | True               | Web ADI アクセスを有効 / 無効にする <sup>b</sup><br>(29 ページ「Web サーバー」を参照)        |
| 6 | Enable FTP server     | Get  | BOOL   | True               | FTP サーバーを有効 / 無効にする <sup>b</sup><br>(27 ページ「FTP サーバー」を参照)            |
| 7 | Enable admin mode     | Get  | BOOL   | False              | FTP の管理者モードを有効 / 無効にする <sup>b</sup><br>(27 ページ「FTP サーバー」を参照)         |
| 8 | Network Status        | Set  | UINT16 | -                  | 149 ページ「ネットワークステータス」を参照                                              |

a. アトリビュートが実装されていない場合、モジュールはこの値を使用します。

b. True= 有効 / False= 無効

### 高度

| # | 名前                              | アクセス | 種類        | デフォルト <sup>a</sup> | コメント                                                                                                                        |
|---|---------------------------------|------|-----------|--------------------|-----------------------------------------------------------------------------------------------------------------------------|
| 1 | MAC address <sup>b</sup>        | Get  | UINT8 の配列 | -                  | 6 バイトからなる物理アドレス；<br>あらかじめプログラムされている<br>MAC アドレスが上書きされる。<br>新規の MAC アドレスは IEEE より<br>入手する必要がある。                              |
| 9 | Port 1 MAC address <sup>b</sup> | Get  | UINT8 の配列 | -                  | 6 バイトからなるポート 1 の<br>MAC アドレス (LLDP プロトコル<br>では必須)<br>注：この設定は、ホストの<br>PROFINET IO オブジェクトにお<br>ける任意のポートの MAC アドレ<br>スを上書きします。 |

| #  | 名前                              | アクセス | 種類        | デフォルト <sup>a</sup> | コメント                                                                                                       |
|----|---------------------------------|------|-----------|--------------------|------------------------------------------------------------------------------------------------------------|
| 10 | Port 2 MAC address <sup>b</sup> | Get  | UINT8 の配列 | -                  | 6 バイトからなるポート 2 の MAC アドレス (LLDP プロトコルでは必須)<br>注: この設定は、ホストの PROFINET IO オブジェクトにおける任意のポートの MAC アドレスを上書きします。 |

a. アトリビュートが実装されていない場合、モジュールはこの値を使用します。

b. 本モジュールには、有効な MAC アドレスがあらかじめプログラムされています。このアドレスを使用する場合は、このアトリビュートを実装しないでください。

## ネットワークステータス

このアトリビュートには、以下に示すように、ネットワーク全体の状態を表すビットフィールドが用意されています。

| ビット    | 内容        | 説明                                             |
|--------|-----------|------------------------------------------------|
| 0      | リンク       | 現在の全体的なリンク状態<br>True= リンク検出 / False= リンク未検出    |
| 1      | IP 確立     | True=IP アドレスが確立された / False= IP アドレスが確立されていない   |
| 2      | (予約)      | (マスクして無視)                                      |
| 3      | リンク ポート 1 | ポート 1 の現在のリンク状態<br>True= リンク検出 / False= リンク未検出 |
| 4      | リンク ポート 2 | ポート 2 の現在のリンク状態<br>True= リンク検出 / False= リンク未検出 |
| 5 ~ 15 | (予約)      | (マスクして無視)                                      |

## 11.4 Energy コントロールオブジェクト (F0h)

### カテゴリ

拡張

### オブジェクトの説明

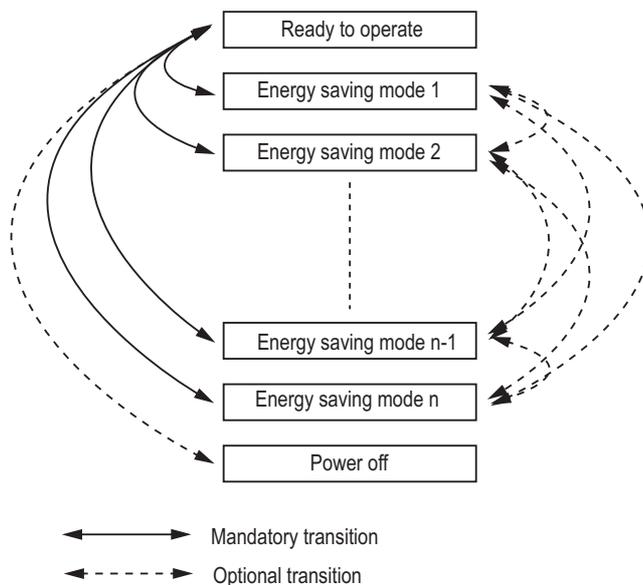
このオブジェクトを使用すると、ホストアプリケーションに **Energy** コントロール機能 (電力固有の設定) が実装されます。このオブジェクトは任意に実装できます。これにより、ホストアプリケーションが以下のアトリビュートを全くサポートしないか、一部のアトリビュートをサポートするか、すべてのアトリビュートをサポートするかを選択できます。

オブジェクトで利用可能な各インスタンスは、省電力モードに対応しています。利用可能なモードの数は装置によって異なります。それらのモードはアプリケーションにて定義する必要があります。インスタンスの番号が大きいほど、より多くの電力を節約できます。最も番号が大きいインスタンスは "電源オフ" に相当します。すなわち、装置は基本的にシャットダウンされます。オブジェクトのインスタンス 1 は "動作可能" を表します。すなわち、装置は完全に機能し、電力は全く節約されません。そのため、意味のある実装では、常に 2 つのインスタンス、すなわち、省電力用のインスタンスと動作用のインスタンスを持ちます。9 番より上のインスタンスは使用しないことを推奨します。

なお、これらの状態は常に存在しており、動的に作成されたり削除されたりすることはありません。

下記も参照してください。

- 123 ページ 「PROFINET IO オブジェクト (F6h)」
- 26 ページ 「PROFInergy プロファイル」



## サポートされているコマンド

オブジェクト : Get Attribute  
StartPause  
EndPause

インスタンス : Get Attribute

## オブジェクトアトリビュート

| #  | 名前                                    | アクセス | データタイプ   | Value                          | 説明                                                                    |
|----|---------------------------------------|------|----------|--------------------------------|-----------------------------------------------------------------------|
| 1  | Name                                  | Get  | CHAR の配列 | "Energy"                       | オブジェクト名                                                               |
| 2  | Revision                              | Get  | UINT8    | 01h                            | オブジェクトの最初のリビジョン                                                       |
| 3  | Number of instances <sup>a</sup>      | Get  | UINT16   | -                              | オブジェクトに含まれるインスタンスの数                                                   |
| 4  | Highest instance no. <sup>a</sup>     | Get  | UINT16   | -                              | 作成されたインスタンスの番号のうち最も大きい番号 (最大 65534)                                   |
| 11 | Current Energy Saving mode            | Get  | UINT16   | -                              | 現在使用されている省電力モードのインスタンス番号 "動作可能" は 1 に相当し、"電源オフ" は最も大きいインスタンス番号に相当します。 |
| 12 | RemaingTimeToDestination <sup>b</sup> | Get  | UINT32   | FFFFFFFFh (デフォルト)              | このパラメータにより、モードの切り替えが完了するまでの実際の時間 (単位: ミリ秒) が規定されます。 <sup>c</sup>      |
| 13 | EnergyConsumption ToDestination       | Get  | FLOAT    | 0.0 (デフォルト。値が未定義の場合もこの値が使用される) | このパラメータにより、モードの切り替えが完了するまでの実際の消費電力 (単位: kWh) が規定されます。 <sup>c</sup>    |

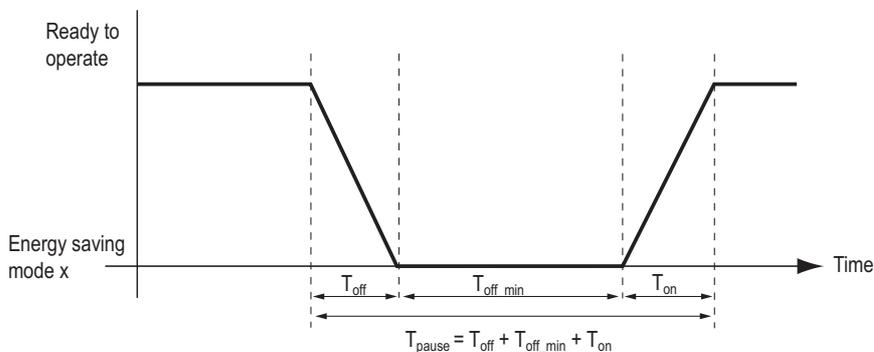
- a. 作成されたインスタンスの番号のうち最も大きい番号は、インスタンスの数と常に等しくなるため、アトリビュート 3 と 4 は同じ値となります。
- b. このアトリビュートの値が無限大または不明な場合、最大値 FFFFFFFFFh が使用されます。この値がゼロの場合は 00000000h が使用されます。
- c. 値を動的に生成できない場合、モードの遷移元から遷移先への移行に関する静的な値が使用されます。

## インスタンスアトリビュート (インスタンス #n)

### 拡張

| # | 名前                               | アクセス | 種類     | デフォルト値 <sup>a</sup> | コメント                                                                                                                |
|---|----------------------------------|------|--------|---------------------|---------------------------------------------------------------------------------------------------------------------|
| 1 | ModeAttributes                   | Get  | UINT16 | 0                   | 静的な値と動的な値のどちらが使用可能かを定義するビットフィールド。<br>ビット 0: 意味<br>0: 時間と電力に関する静的な値のみ使用可能<br>1: 時間と電力に関する動的な値を使用可能<br>ビット 1 ~ 15: 予約 |
| 2 | TimeMinPause <sup>b</sup>        | Get  | UINT32 | 0                   | 最小休止時間。 $t_{\text{pause}}$ (ms) <sup>c</sup>                                                                        |
| 3 | TimeToPause <sup>b</sup>         | Get  | UINT32 | 0                   | この省電力状態に移行するまでの予定時間。 $t_{\text{off}}$ (ms) <sup>c</sup>                                                             |
| 4 | TimeToOperate <sup>b</sup>       | Get  | UINT32 | 0                   | "動作可能" 状態に移行するのに必要な時間。 $t_{\text{on}}$ (ms) <sup>c</sup>                                                            |
| 5 | TimeMinLengthOfStay <sup>b</sup> | Get  | UINT32 | 0                   | 装置がこの状態を維持しなければならない最小時間。<br>$t_{\text{off\_min}}$ (ms) <sup>c</sup>                                                 |
| 6 | TimeMaxLengthOfStay <sup>b</sup> | Get  | UINT32 | FFFFFFFFh           | この状態を維持できる最大時間 (ms)                                                                                                 |
| 7 | ModePowerConsumption             | Get  | FLOAT  | 0.0 <sup>d</sup>    | この状態における消費電力 (kW)                                                                                                   |
| 8 | EnergyConsumptionToPause         | Get  | FLOAT  | 0.0 <sup>d</sup>    | この状態に移行するのに必要な消費電力 (kWh)                                                                                            |
| 9 | EnergyConsumptionToOperate       | Get  | FLOAT  | 0.0 <sup>d</sup>    | この状態から "動作可能" 状態に移行するのに必要な消費電力 (kWh)                                                                                |

- a. アトリビュートが実装されていない場合はこの値が使用されます。  
b. このアトリビュートの値が無限大または不明な場合、最大値 FFFFFFFFh が使用されます。  
この値がゼロの場合は 00000000h が使用されます。  
c.  $t_{\text{off}} + t_{\text{off\_min}} + t_{\text{on}} = t_{\text{pause}}$  下図を参照。  
d. 値 0.0 は、アトリビュートの値が定義されていない場合にも使用されます。



## コマンド詳細 :StartPause

### カテゴリ

拡張

### 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

本モジュールは、このコマンドを発行することで、システムの休止を開始することをホストアプリケーションに要求します。休止時間はミリ秒単位で指定します。コマンドに対する応答には、移行先のモード、すなわち、選択した省電力状態のインスタンス番号が含まれます。このコマンドは、常にオブジェクトそのものに発行されます。

- コマンド詳細

| フィールド      | 内容                 | コメント      |
|------------|--------------------|-----------|
| CmdExt[0]  |                    | (未使用)     |
| CmdExt[1]  |                    |           |
| MsgData[0] | 休止時間 (下位ワード、下位バイト) | 休止時間 (ms) |
| MsgData[1] | 休止時間 (下位ワード、上位バイト) |           |
| MsgData[2] | 休止時間 (上位ワード、下位バイト) |           |
| MsgData[3] | 休止時間 (上位ワード、上位バイト) |           |

- 応答詳細

| フィールド         | 内容               | コメント               |
|---------------|------------------|--------------------|
| CmdExt[0 ~ 1] | (予約)             | (ゼロに設定)            |
| MsgData[0]    | インスタンス番号 (下位バイト) | 選択した電力モードのインスタンス番号 |
| MsgData[1]    | インスタンス番号 (上位バイト) |                    |

アプリケーションがモードを変更しない場合、エラーコード "ABP\_ERR\_OUT\_OF\_RANGE" (0Ch) が返されます。

下記も参照してください。

- 154 ページ 「コマンド詳細 :EndPause」

## コマンド詳細 :EndPause

### カテゴリ

拡張

### 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

本モジュールは、このコマンドを発行することで、システムを休止モードから "動作可能" モードに戻すようホストアプリケーションに要求します。このコマンドに対する応答では、"動作可能" モードに戻るまでの時間をミリ秒単位で指定する必要があります。このコマンドは、常にオブジェクトそのものに発行されます。

- コマンド詳細

| フィールド     | 内容 | コメント  |
|-----------|----|-------|
| CmdExt[0] | -  | (未使用) |
| CmdExt[1] |    |       |

- 応答詳細

| フィールド          | 内容                      | コメント                       |
|----------------|-------------------------|----------------------------|
| CmdExt [0 ~ 1] | (予約)                    | (ゼロに設定)                    |
| MsgData[0]     | 動作するまでの時間 (下位ワード、下位バイト) | "動作可能" モードに切り替わるまでの時間 (ms) |
| MsgData[1]     | 動作するまでの時間 (下位ワード、上位バイト) |                            |
| MsgData[2]     | 動作するまでの時間 (上位ワード、下位バイト) |                            |
| MsgData[3]     | 動作するまでの時間 (上位ワード、上位バイト) |                            |

アプリケーションが休止状態を終了できない場合、エラーコード "ABP\_ERR\_INV\_STAT" (0Dh) が返されます。

## A. 機能の分類

Anybus CompactCom とアプリケーションのオブジェクトやアトリビュート、サービスは、3つのカテゴリ、すなわち、基本機能、高度な機能、拡張機能に分類されます。

### A.1 基本

このカテゴリには、必ず実装または使用しなければならないオブジェクトやアトリビュート、サービスが含まれます。Anybus CompactCom を起動し、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、このカテゴリで十分です。産業用ネットワークの基本機能が使用されます。

製品の認定を可能とする追加オブジェクトなども、このカテゴリに属します。

### A.2 拡張

このカテゴリのオブジェクトを使用すると、アプリケーションの機能を拡張できます。ネットワークとの間の基本的なデータ交換だけでなく、産業用ネットワーク固有の機能を利用できるようになるため、アプリケーションの価値が高まります。

### A.3 高度

このカテゴリのオブジェクトやアトリビュート、サービスを使用すると、特殊な機能やあまり使用されない機能が利用できるようになります。提供されているほとんどのネットワーク機能が有効となり、利用可能になります。通常、産業用ネットワークの仕様の確認が必要となります。

## B. Anybus の実装詳細

### B.1 SUP ビットの定義

監視ビット (SUP) は、他のネットワーク機器によってネットワークへの参加が監視されていることを表します。PROFINET IO の場合、いずれかの IO 接続が確立されたときにこのビットがセットされます。

### B.2 Anybus のステートマシン

以下の表に、Anybus のステートマシンと PROFINET IO ネットワークとの対応を示します。

| Anybus の状態     | 実装                                                                                | コメント                                                                               |
|----------------|-----------------------------------------------------------------------------------|------------------------------------------------------------------------------------|
| WAIT_PROCESS   | Anybus は、IO コントローラーとの IO 接続がオープンするまでこの状態を維持します。                                   | なお、物理的にネットワークに接続されない限り (リンクが検出されない限り)、モジュールは NW_INIT から WAIT_PROCESS に移行しません。      |
| ERROR          | 設定データの不一致、または初期パラメーターのエラー。                                                        | -                                                                                  |
| PROCESS_ACTIVE | IO コントローラーとの IO 接続が確立し、有効な出力データが 1 回以上送信された。                                      | -                                                                                  |
| IDLE           | IO 接続を確立した IO コントローラーが STOP モードにある。または、IO コントローラーから有効な出力データが 1 度も送信されていない。        | IO コントローラーによっては、接続成功後の最初のサイクルにて有効な出力データを送信しないものがあります。                              |
| EXCEPTION      | モジュールのステータス LED が赤く点灯し、重大な障害が発生したことを示す。ネットワーク ステータス LED は消灯。イーサネットの MAC がリセットされる。 | ウォッチドッグのタイムアウト等、何らかの予期せぬ動作が発生した。61 ページ「ネットワークオブジェクト (03h)」のインスタンス 1、アトリビュート 7 を参照。 |

### B.3 アプリケーションウォッチドッグのタイムアウト処理

本モジュールは、アプリケーションウォッチドッグのタイムアウトを検出すると、ネットワークへの接続を中止して "EXCEPTION" 状態に移行します。ネットワークに関するその他の処理は行われません。

## C. メッセージの分割

### C.1 概要

カテゴリ : 高度

Anybus-CompactCom では、最大 255 バイトのメッセージをサポートします。(ソケットインターフェイスを使用する際に必要な) さらに長いメッセージをサポートするには、分割プロトコルを使用します。

分割プロトコルはメッセージ層で実装します。ホストのシリアルインターフェイスで使用されるフラグメンテーションと混同しないでください。詳細については、『Anybus-CompactCom Software Design Guide』を参照してください。

本モジュールは、1つのインスタンスにつき、同時に分割されたメッセージを1つだけサポートします。

### C.2 コマンドの分割

コマンドメッセージが分割された場合、コマンドの発行元は、同じコマンドヘッダーを複数回送信します。各メッセージにおいて、データフィールドは次のデータセグメントにて送信されます。

なお、コマンドによっては、インスタンスの分割バッファにアクセスする必要があるため、同じインスタンスにて同時に使用できないものがあります。

コマンドの分割は以下のコマンドで使用されます。

- Send (97 ページ「コマンド詳細 :Send」を参照)
- Send To (98 ページ「コマンド詳細 :Send\_To」を参照)

#### 分割制御ビット (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である場合にセットされる |
| 1     | LS   | 現在のセグメントが最後のセグメントである場合にセットされる |
| 2     | AB   | 分割が中断されたときにセットされる             |
| 3 ~ 7 | (予約) | 0 (ゼロ) に設定                    |

#### 分割制御ビット (応答)

| ビット   | 内容   | 意味 |
|-------|------|----|
| 0 ~ 7 | (予約) | 無視 |

分割されたコマンドを発行する場合、以下の規則が適用されます。

- 最初のセグメントを発行するとき、FS をセットする必要があります。
- それに続くセグメントを発行するとき、FS と LS 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とき、LS をセットする必要があります。

- 1つのセグメントから成るコマンド (255 バイト以下のコマンド) の場合、FS と LS をセットする必要があります。
- 最後の応答メッセージには、実際の操作結果が含まれます。
- コマンドの発行元は、AB がセットされたメッセージを発行することで、操作をいつでも中断できます。
- 送信中に分割エラーが検出されると、エラーメッセージが返され、現在の分割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ます。既に送信されたセグメントは有効です。

## C.3 応答の分割

応答が分割された場合、コマンドの発行元は、同じコマンドを複数回送信して次のセグメントを要求します。各応答において、データフィールドは次のデータセグメントにて送信されます。

応答の分割は、以下のコマンドに対する応答で使用されます。

- Receive ( オブジェクト固有。95 ページ「コマンド詳細 :Receive」を参照 )
- Receive From ( オブジェクト固有。96 ページ「コマンド詳細 :Receive\_From」を参照 )

### 分割制御ビット ( コマンド )

| ビット   | 内容     | 意味                |
|-------|--------|-------------------|
| 0     | ( 予約 ) | ( ゼロに設定 )         |
| 1     |        |                   |
| 2     | AB     | 分割が中断されたときにセットされる |
| 3 ~ 7 | ( 予約 ) | ( ゼロに設定 )         |

### 分割制御ビット ( 応答 )

| 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である場合にセットされる |
| 1     | LS     | 現在のセグメントが最後のセグメントである場合にセットされる |
| 2 ~ 7 | ( 予約 ) | ( ゼロに設定 )                     |

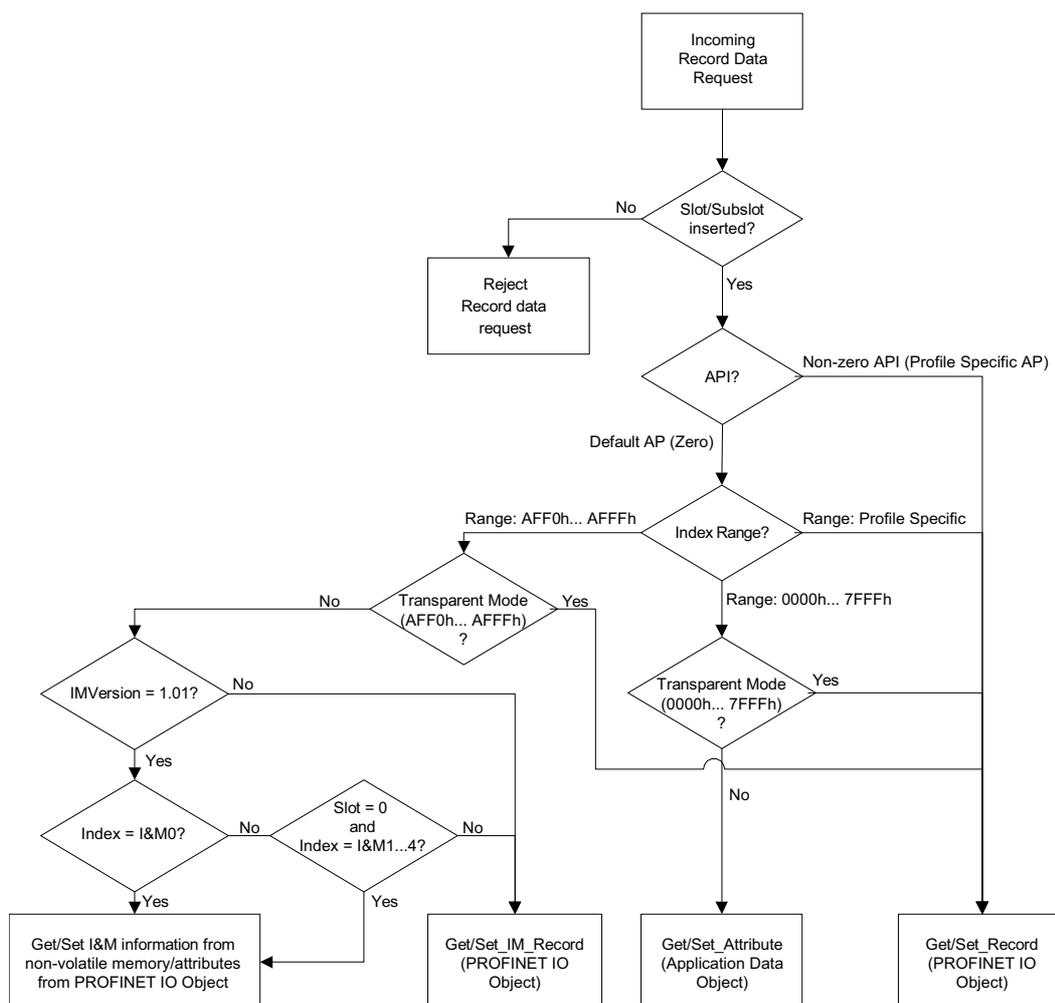
分割された応答を受信する場合、以下の規則が適用されます。

- 最初のセグメントでは FS がセットされます。
- それに続く全てのセグメントでは、FS と LS がクリアされます。
- 最後のセグメントでは LS がセットされます。
- 1つのセグメントから成る応答 (255 バイト以下の応答) の場合、FS と LS がセットされます。
- コマンドの発行元は、AB がセットされたメッセージを発行することで、操作をいつでも中断できます。

## D. フローチャート

### D.1 フローチャート - レコードデータへのアクセス

このフローチャートでは、Anybus モジュールが Record Data 要求をどのように処理するかを示します。

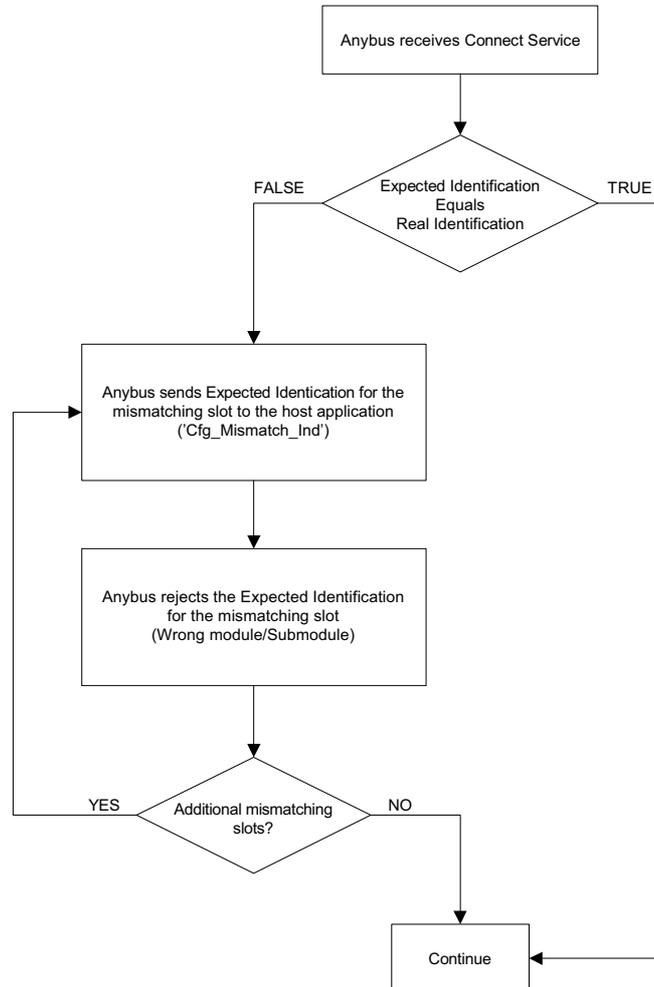


下記も参照してください。

- 13 ページ 「アプリケーションデータインスタンス (ADI)」
- 123 ページ 「PROFINET IO オブジェクト (F6h)」
- 127 ページ 「コマンド詳細 :Get\_Record」
- 129 ページ 「コマンド詳細 :Set\_Record」
- 131 ページ 「コマンド詳細 :Get\_IM\_Record」
- 133 ページ 「コマンド詳細 :Set\_IM\_Record」

## D.2 フローチャート - 設定の不一致 (RI)

このフローチャートでは、Anybus モジュールが設定の不一致をどのように処理するかを示します。

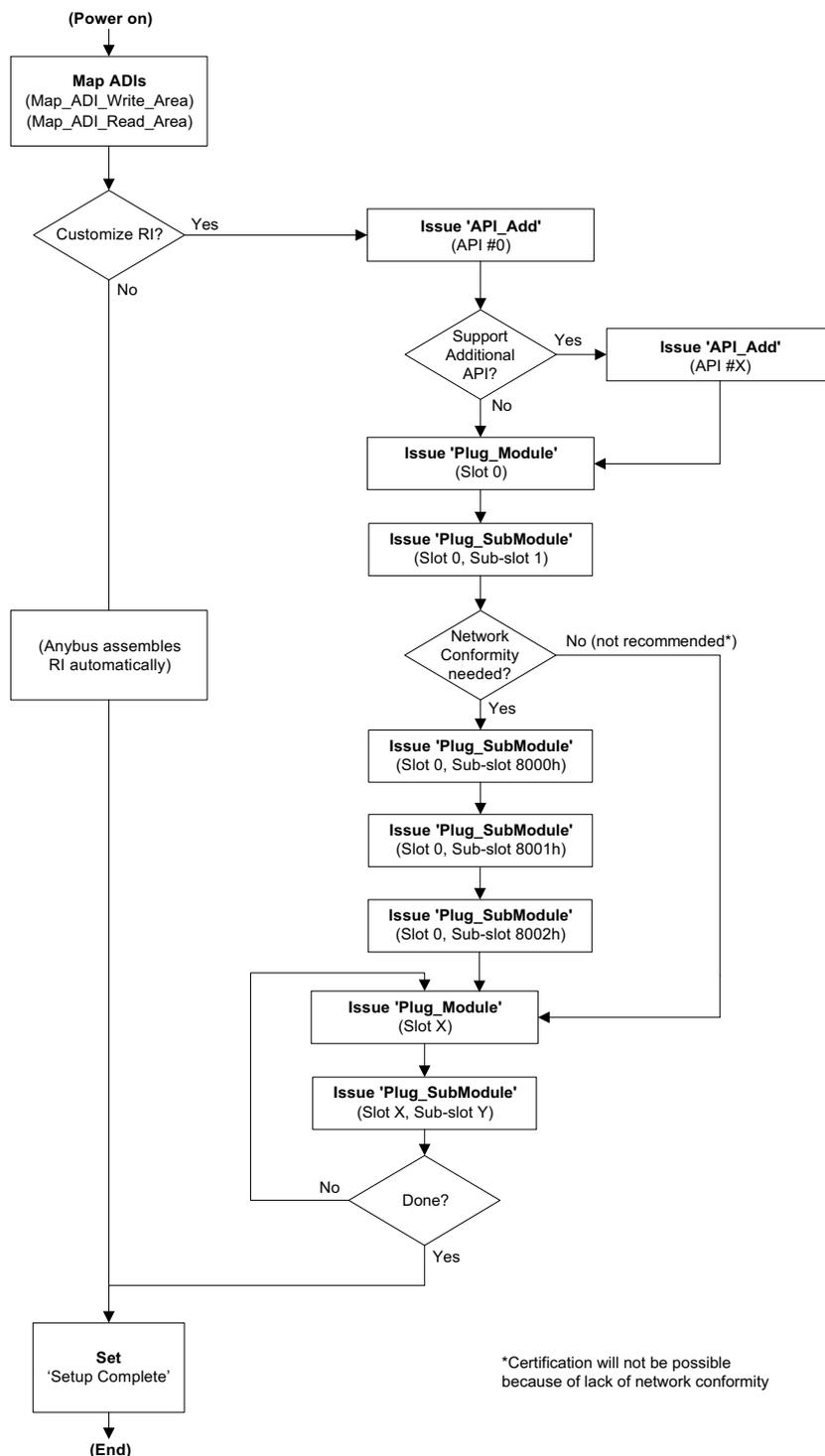


下記も参照してください。

- 14 ページ 「プロセスデータ」
- 20 ページ 「設定の不一致」
- 137 ページ 「コマンド詳細 :Cfg\_Mismatch\_Ind」
- 161 ページ 「フローチャート - Real Identification (RI) の確立」

## D.3 フローチャート - Real Identification (RI) の確立

このフローチャートでは、Real Identification の確立を示します。



下記も参照してください。

- 14 ページ 「プロセスデータ」
- 19 ページ 「Real Identification (RI)」
- 160 ページ 「フローチャート - 設定の不一致 (RI)」

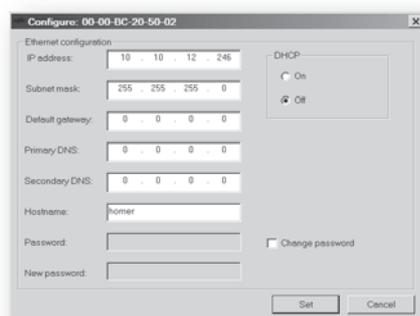
## E. HICP (ホスト IP 設定プロトコル)

### E.1 概要

本モジュールは、IP アドレスやサブネットマスク、DHCP の有効/無効などの設定を変更するために、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](http://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 のポート 3250 を経由して、ネットワークに接続されている Anybus 製品のネットワーク設定にアクセスできます。

### E.2 操作

このプログラムを起動すると、ネットワーク上に Anybus の製品があるかどうかスキャンされます。[Scan] をクリックすることで、ネットワークをいつでもスキャン可能です。



モジュールのネットワーク設定を変更するには、リストにあるエントリをダブルクリックします。すると、モジュールの設定を含むウィンドウが表示されます。

[Set] をクリックすると新規の設定が有効になります。また、[Cancel] をクリックすると全ての変更がキャンセルされます。

オプションとして、パスワードを用いて不正なアクセスから設定を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] にパスワードを入力します。

## F. 技術仕様

### F.1 保護接地 (PE) に関する要件

適切な EMC 動作を保証するには、『Anybus-CompactCom Hardware Design Guide』に記述されている PE パッド / PE メカニズムに従って、本モジュールを適切に保護接地する必要があります。

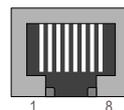
PE に関する要件が満たされていない場合、HMS Industrial Networks は適切な EMC 動作を保証しません。

**注** :RJ45 コネクタのシールドは、PE に直接接続されていません。PROFINET ネットワーク上の全てのノードは同じ筐体接地に接続する必要があるため、ネットワーク上の各ノードにおいて、PROFINET ケーブルのシールドを筐体接地に接続する必要があります。

詳細については、『PROFINET Installation Guideline for Cabling and Assembly』を参照してください (注文番号 8.072。www.PROFINET.com からダウンロード可能)。

#### PROFINET 用 RJ45 コネクタ

| ピン番号       | 説明                |
|------------|-------------------|
| 1          | TD+               |
| 2          | TD-               |
| 3          | RD+               |
| 4, 5, 7, 8 | 直列 RC 回路を介して筐体に接続 |
| 6          | RD-               |
| ハウジング      | ケーブルシールド          |



### F.2 電源

#### 電源電圧

本モジュールを使用するには、『Anybus-CompactCom Hardware Design Guide』が指定する安定化された 3.3 V の電源が必要です。

#### 消費電力

Anybus CompactCom PROFINET IO 2-Port は、クラス B モジュールの要件を満たすように設計されています。Anybus-CompactCom プラットフォームにおける消費電力の分類についての詳細は、『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では、最大 380 mA を消費します<sup>1</sup>。

### F.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。

1. HMS Industrial Networks は、製品開発に継続的に取り組むという HMS のポリシーに従って、本製品の消費電力に関する厳密な要件を予告なく変更する権利を留保します。ただし、いかなる場合であっても、Anybus CompactCom PROFINET IO 2-Port はクラス B モジュールであることを維持します。

## F.4 EMC への準拠

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

## G. タイミングと性能

### G.1 概要

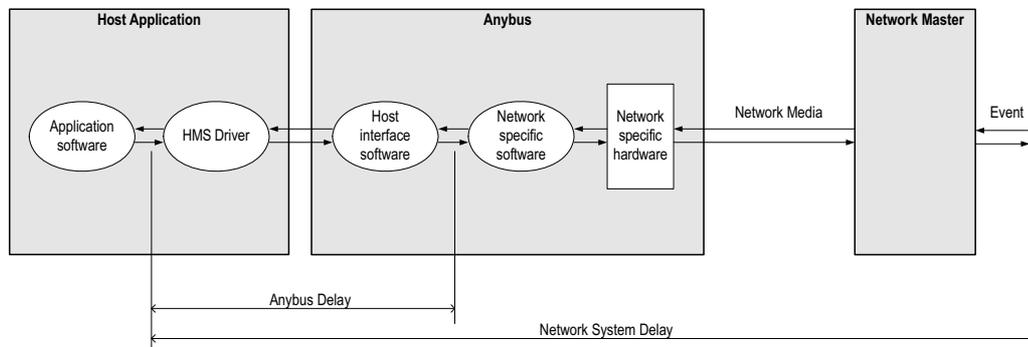
この章では、Anybus CompactCom PROFINET IO 2-Port について検証 / 文書化された、タイミングと性能に関するパラメーターについて説明します。

以下のタイミングが測定されています。

| カテゴリ                                            | パラメーター        | ページ                                                        |
|-------------------------------------------------|---------------|------------------------------------------------------------|
| 起動時の遅延時間                                        | T1, T2        | 『Anybus CompactCom Software Design Guide』の「付録 B」を参照してください。 |
| NW_INIT の遅延時間                                   | T3            |                                                            |
| テレグラムの遅延時間                                      | T4            |                                                            |
| コマンドの遅延時間                                       | T5            |                                                            |
| Anybus におけるリードプロセスデータの遅延時間 (Anybus の遅延時間)       | T6, T7, T8    |                                                            |
| Anybus におけるライトプロセスデータの遅延時間 (Anybus の遅延時間)       | T12, T13, T14 |                                                            |
| ネットワークシステムにおけるリードプロセスデータの遅延時間 (ネットワークシステムの遅延時間) | T9, T10, T11  | 167                                                        |
| ネットワークシステムにおけるライトプロセスデータの遅延時間 (ネットワークシステムの遅延時間) | T15, T16, T17 | 167                                                        |

## G.2 プロセスデータ

### G.2.1 概要



### G.2.2 Anybus におけるリードプロセスデータの遅延時間 (Anybus の遅延時間)

リードプロセスデータの遅延時間 (上記の図で "Anybus の遅延時間" と表記) は、新規データがバッファに入力されて Anybus のホストインターフェイスソフトウェアで利用可能になる直前から、そのデータがホストアプリケーションに利用可能になるまで (新規データがドライバーに読み込まれた直後まで) の時間で定義されます。

詳細については、『Anybus CompactCom Software Design Guide』の「付録 B」を参照してください。

### G.2.3 Anybus におけるライトプロセスデータの遅延時間 (Anybus の遅延時間)

ライトプロセスデータの遅延時間 (上記の図で "Anybus の遅延時間" と表記) は、データがホストアプリケーションから利用可能になったとき (ホストアプリケーションからドライバーにデータが書き込まれる直前) から、Anybus のホストインターフェイスソフトウェアによって新規データがネットワークバッファに転送されたときまでの時間で定義されます。

詳細については、『Anybus CompactCom Software Design Guide』の「付録 B」を参照してください。

## G.2.4 ネットワークシステムにおけるリードプロセスデータの遅延時間 (ネットワークシステムの遅延時間)

ネットワークシステムにおけるリードプロセスデータの遅延時間(上記の図で"ネットワークシステムの遅延時間"と表記)は、ネットワークのマスターにてイベントが生成されてから、該当するデータがホストアプリケーションで利用可能になるまで(該当するデータがドライバーに読み込まれた直後まで)の時間で定義されます。

| パラメーター | 説明                                                 | 平均  | 最大   | 単位 |
|--------|----------------------------------------------------|-----|------|----|
| T9     | ネットワークシステムにおけるリードプロセスデータの遅延時間。ADI × 8 (UINT8 × 1)  | 3.3 | 4.64 | ms |
| T10    | ネットワークシステムにおけるリードプロセスデータの遅延時間。ADI × 16 (UINT8 × 1) | 5.0 | 7.5  | ms |
| T11    | ネットワークシステムにおけるリードプロセスデータの遅延時間。ADI × 32 (UINT8 × 1) | 4.5 | 8.0  | ms |

条件:

| パラメーター                                    | 条件           |
|-------------------------------------------|--------------|
| アプリケーションの CPU                             | -            |
| タイマーシステムの呼び出し間隔                           | 1 ms         |
| ドライバーの呼び出し間隔                              | 0.2 ~ 0.3 ms |
| 各方向について、ADI の数 (UINT8 × 1) をプロセスデータにマッピング | 8, 16, 32    |
| 通信                                        | パラレル         |
| 測定期間中のテレグラムタイプ                            | プロセス データのみ   |
| バスの負荷、ノード数、ボーレートなど                        | 通常           |

## G.2.5 ネットワークシステムにおけるライトプロセスデータの遅延時間 (ネットワークシステムの遅延時間)

ネットワークシステムにおけるライトプロセスデータの遅延時間(上記の図で"ネットワークシステムの遅延時間"と表記)は、新規データがホストアプリケーションから利用可能になってから(新規データがドライバーに書き込まれる直前から)、そのデータによってネットワークのマスターにて該当するイベントが生成されるまでの時間で定義されます。

| パラメーター | 説明                                                 | 平均   | 最大  | 単位 |
|--------|----------------------------------------------------|------|-----|----|
| T15    | ネットワークシステムにおけるライトプロセスデータの遅延時間。ADI × 8 (UINT8 × 1)  | 3.3  | 5.0 | ms |
| T16    | ネットワークシステムにおけるライトプロセスデータの遅延時間。ADI × 16 (UINT8 × 1) | 5.1  | 7.2 | ms |
| T17    | ネットワークシステムにおけるライトプロセスデータの遅延時間。ADI × 32 (UINT8 × 1) | 4.25 | 7.5 | ms |

条件: 167 ページ「ネットワークシステムにおけるリードプロセスデータの遅延時間(ネットワークシステムの遅延時間)」と同じ。

---

## H. コンフォーマンステストに関するガイド

### H.1 概要

Anybus CompactCom PROFINET IO モジュールは、すべてのパラメーターをデフォルトの設定で使用すれば、ネットワークの規格に準拠するようになっています。これは、お客様の製品での適合性認定の取得を可能とするものですが、お客様の製品が認定を受ける必要がないことを意味するものではありません。

HMS が提供する GSD ファイルのパラメーターを変更した場合、認定を受ける必要があります。ベンダー ID は PNO より入手できます。認定を受けるにはベンダー ID が必要です。この章では、Anybus CompactCom PROFINET IO を含む製品のコンフォーマンステストにおいて、PNO が定めるネットワーク規格の認定に関する要求を満たすためのガイドを示します。

認定プロセスでは、選択した動作モードにかかわらず、この付録で説明する作業を考慮する必要があります。また、メーカーと機器に応じて製品 ID を変更する必要があります。

---

**重要：**この付録では、認定に必要なガイドラインと例を示します。各自のアプリケーションの機能によっては、手順の追加が必要となる場合があります。詳細については、*HMS Industrial Networks* ([www.anybus.com](http://www.anybus.com)) にお問い合わせください。

## H.2 お客様製品の Identity 情報確定

Anybus オブジェクト (01h) の "Setup Complete" (セットアップ完了) アトリビュートの設定が完了すると、Anybus モジュールは、ホストの PROFINET IO オブジェクト (F6h) に対して ID データを問い合わせます。そのため、以下のアトリビュートを実装し、正しい値を返す必要があります。

| 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                                       | 顧客の例                            | コメント                                                                                                                                                                                                          |
|-----------------------------------|----------------------------|--------------------------------------------|---------------------------------------------|---------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| PROFINET IO オブジェクト (F6h)、インスタンス 1 | #1, Device ID              | このアトリビュートを使用して、機器の機器 ID を設定します。            | Device ID:0009h                             | Device ID:YYYYh                 | この情報は、GSD ファイルの "DevicelDentity" のキーと一致しなければなりません。                                                                                                                                                            |
| PROFINET IO オブジェクト (F6h)、インスタンス 1 | #2, Vendor ID              | このアトリビュートを使用して、機器のベンダー ID を設定します。          | Vendor ID:010Ch (HMS)                       | Vendor ID:XXXh                  | なお、GSD ファイルの "VendorName" キーワードは、Vendor ID の値と一致しなければなりません。                                                                                                                                                   |
| PROFINET IO オブジェクト (F6h)、インスタンス 1 | #3, Station Type           | このアトリビュートを使用して、機器のステーションタイプを設定します。         | "ABCC-PRT (2-Port)"                         | "Cust-PNIO-Dev"                 | Anybus CompactCom PROFINET IO の場合、この情報は GSD の "DNS_CompatibleName" キーワードおよび "OrderNumber" キーワードと一致します。ステーションタイプは "DNS_CompatibleName" と同じでなければなりません。"OrderNumber" は全く異なっていても構いません。以下の I&M Order ID を参照してください。 |
| PROFINET IO オブジェクト (F6h)、インスタンス 1 | #8, I&M Order ID           | このアトリビュートを使用して、I&M データで使用するオーダー ID を設定します。 | "ABCC-PRT (2-Port)"                         | "Cust-PNIO-Dev"                 | この情報は、GSD ファイルの "OrderNumber" のキーと一致しなければなりません。                                                                                                                                                               |
| PROFINET IO オブジェクト (F6h)、インスタンス 1 | #10, I&M Hardware Revision | このアトリビュートを使用して、I&M ハードウェアリビジョンを設定します。      | (ハードウェアリビジョン)                               | "0002h"                         | この情報は、GSD ファイルの "HardwareRelease" のキーと一致しなければなりません。                                                                                                                                                           |
| PROFINET IO オブジェクト (F6h)、インスタンス 1 | #11, I&M Software Revision | このアトリビュートを使用して、I&M ソフトウェアリビジョンを設定します。      | (ソフトウェアリビジョン)                               | "V2.5.3"                        | この情報は、GSD ファイルの "SoftwareRelease" のキーと一致しなければなりません。                                                                                                                                                           |
| PROFINET IO オブジェクト (F6h)、インスタンス 1 | #19, System Description    | このアトリビュートを使用して、システムの説明を設定します。              | "HMS Industrial Networks Anybus-CompactCom" | "Customer HMI Interface Module" | この情報は、SNMP を用いてネットワークから読み取り可能です。                                                                                                                                                                              |
| PROFINET IO オブジェクト (F6h)、インスタンス 1 | #20, Interface Description | このアトリビュートを使用して、インターフェイスの説明を設定します。          | "PROFINET IO Interface"                     | "PROFINET IO Interface"         |                                                                                                                                                                                                               |
| PROFINET IO オブジェクト (F6h)、インスタンス 1 | #22, System Contact        | このアトリビュートを使用して、システムの問い合わせ情報を設定します。         | "www.anybus.com"                            | "www.customer.com"              | この情報は、SNMP を使用して、MIB-II を介してネットワークから読み取り可能です。                                                                                                                                                                 |

### GSD ファイルの追加情報

GSD ファイルの "ProductFamily" キーワードには、装置のベンダー名を記述します。

GSD ファイルの "MainFamily" キーワードには、製品がリストされる装置の種類を記述します。GSD 規格 v2.25 では、以下の "ファミリー" を使用できます：

"General"、"Drives"、"Switching Devices"、"I/O"、"Valves"、"Controllers"、"HMI"、"Encoders"、"NC/RC"、"Gateway"、"PLCs"、"Ident Systems"、"PA Profiles"、"Network Components"、"Sensors"

## H.3 工場出荷時状態へのリセット

### アプリケーションオブジェクト (FFh) に対するリセットコマンドをサポート

PROFINET IO モジュールは、納入時に "工場出荷時" 状態でなければなりません。PROFINET 機器の場合、これは、ステーション名が空 ("") で、IP スイートが割り当てられていない (IP 0.0.0.0) 状態を意味します。Anybus モジュールは、ネットワークから工場出荷時状態へのリセットコマンドを受信すると、IP とステーション名に関する全ての情報を消去し、ホストアプリケーションに対して Anybus モジュールのハードウェア/ソフトウェアのリセットが必要であることを通知します。工場出荷時状態へのリセットを行うには、ホストのアプリケーションオブジェクト (FFh) にリセットコマンドを送信します (パワーオン + 工場出荷状態へのリセット)。詳細については、『Anybus CompactCom Software Design Guide』を参照してください。

## H.4 IP アドレス

通常、PROFINET IO 機器の IP 番号は、DCP (Discovery and Configuration Protocol) を使用して PROFINET ネットワークから割り当てます。PROFINET モジュールの初期化フェーズ中は、ユーザーにより IP アドレスを特定の値に設定するよう要求されない限り、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 ~ #6) を使用しないことを推奨します。これは、(DCP を介して) PROFINET ネットワークから工場出荷時状態へのリセットコマンドを受信したとき、ハードウェアまたはソフトウェアをデフォルトの IP アドレス (0.0.0.0) にリセットした後も、ノードが利用可能でなければならないためです。

## H.5 ステーション名

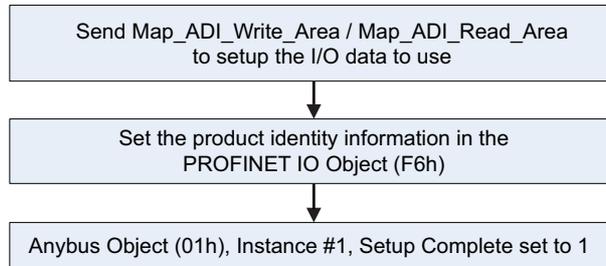
通常、PROFINET 機器のステーション名は、PROFINET DCP を用いてエンドユーザーが割り当てます。PROFINET モジュールの初期化フェーズ中は、ネットワークコンフィグレーションオブジェクトのインスタンス #15 を使用しないことを推奨します。このアトリビュートを使用する場合、エンドユーザーがキーパッドなどでステーション名を変更したときに、このアトリビュートを明示的に送信することを推奨します。これは、(DCP を介して) PROFINET ネットワークから工場出荷時状態へのリセットコマンドを受信したとき、ハードウェアまたはソフトウェアをデフォルトのステーション名 ("") にリセットした後も、ノードが利用可能でなければならないためです。

---

**重要** :Anybus モジュールは、IO コントローラーとの間で確立されている接続に関するあらゆる情報を、PROFINET IO オブジェクト (F6h) に対してコマンドとして転送します。ホストアプリケーションがこの情報を必要としない場合でも、常に応答を生成する必要があります (05h、"Unsupported command" など)

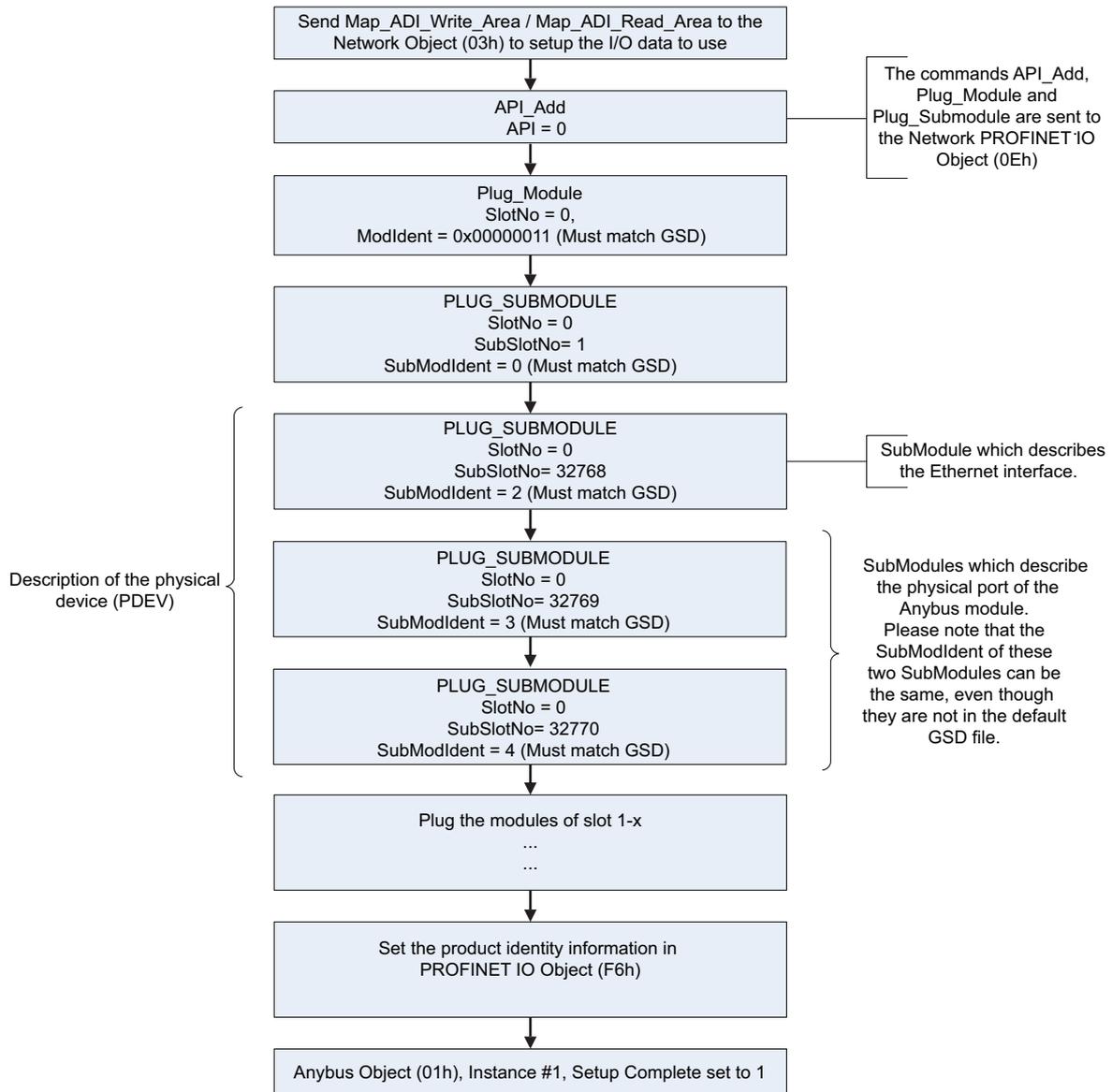
## H.6 Anybus のジェネリックモードにおける認定

Anybus のジェネリックモード ( ネットワーク PROFINET IO オブジェクト (0Eh) の API\_add コマンドを使用しない場合 ) では、通常、この付録で既に述べた内容以外に考慮する必要はありません。製品の ID に関して、HMS が提供するデフォルトの GSD ファイルを変更する必要があります。このとき、製品の認定が必要となります。



## H.7 アドバンストモードにおける認定

アドバンストモード(ネットワーク PROFINET IO Object (0Eh) を使用する場合)において最も重要なことは、PROFINET IO 規格 v2.0 以降に適合するデバイスアクセスポイント(DAP2)を使用するという事です。バージョン 2.0 の規格から、専用のメカニズムを用いてイーサネットの物理インターフェイスとそのポート(PDEV、物理デバイス)を記述できるようになりました。これは、スロット 0 の専用サブモジュールを用いて行います(スロット 0 のモジュールが装置のアクセスポイントになります)。DAP2 を設定する場合、以下の流れで行うことを推奨します。

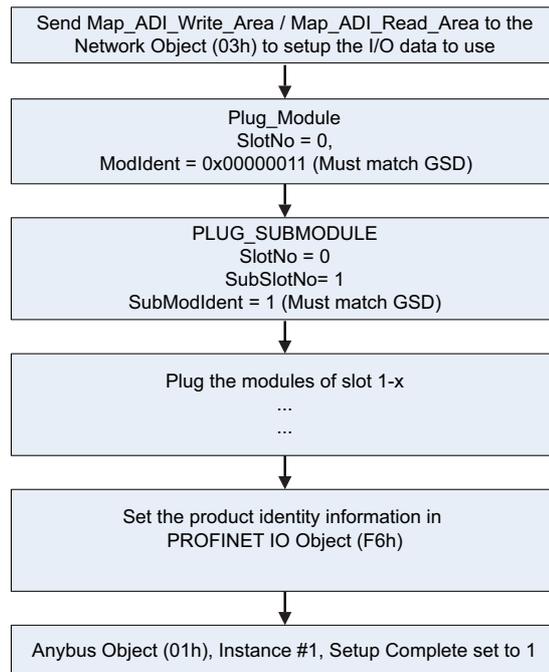


この図では、PROFINET に準拠した DAP を設定する方法が示されています。なお、一部のコマンドでは、関連するパラメーターのみ示してあります。

上記のフローチャートにおける "SubModIdent" の値は、HMS が提供するデフォルトの GSD ファイルの値です。この値は必要に応じて変更できますが、変更する真の必要性はありません。重要なのは、GSD ファイルの値と一致させることです。PROFINET の適合性試験に合格するには、"DAP2" が必ず必要です。市場では、PROFINET IO 規格 v2.0 以降をサポートしていない PROFINET IO コントローラーがまだ存在します。これらのコントローラーでは、DAP2 は使用できません。そのため、PDEV を含まない DAP (最後の 3

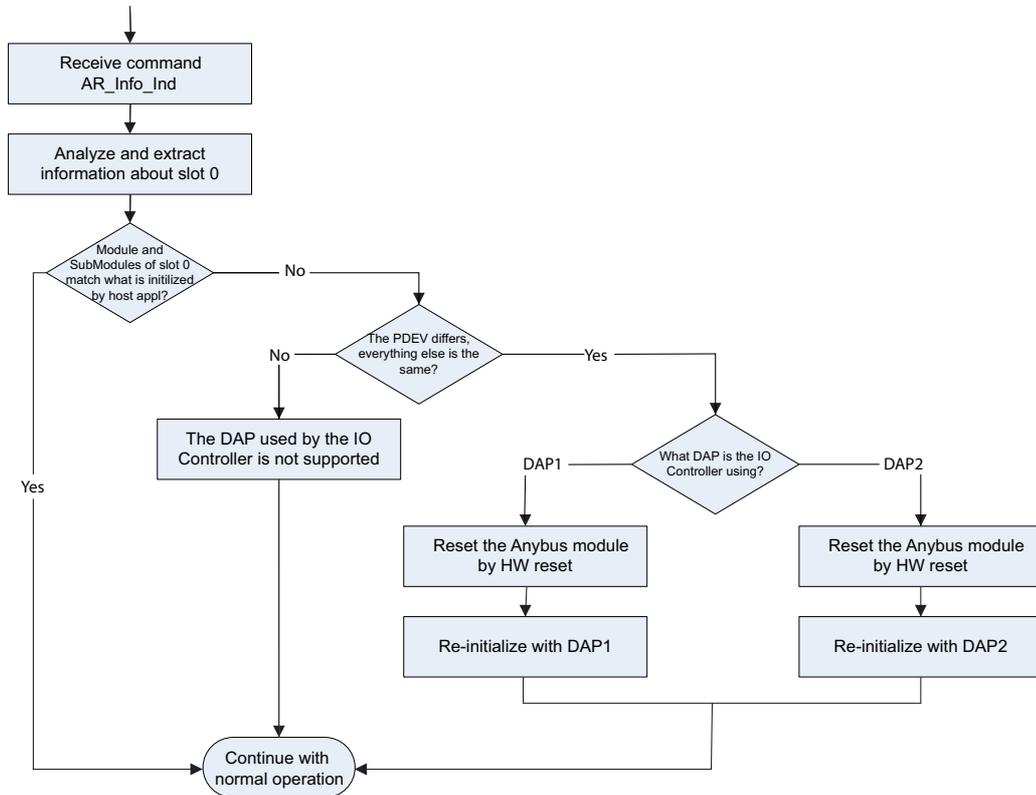
つの PLUG\_SUBMODULE コマンドを発行しない DAP) もサポートする必要があります。これは、"マイグレーション" DAP と呼ばれます。デフォルトの GSD ファイルでは、この機能が用意されています。アドバンスモードでは、2つの方法でこれを実装できます。

1. エンドユーザーが後方互換性の必要性があると判断し、ハンドパネル上のパラメーターなどを用いて後方互換性を選択する場合。このとき、ホストアプリケーションにより Anybus モジュールのハードウェア/ソフトウェアがリセットされ、p.172 の図に示すように、最後の3つの Plug\_Submodules がスキップされます。その結果、以下に示すフローのようになります。



この図では、後方互換性を実現するのに必要な、PDEV を含まない DAP が示されています (一部のコマンドでは、関連するパラメーターのみ示してあります)。

2. ホストアプリケーションが、Anybus モジュールからホストの PROFIBUS IO オブジェクト (F6h) に送信された AR\_Info\_Ind コマンドを使用する場合。この場合、IO コントローラーにより確立されている接続を解析できます。IO コントローラーが、まだ埋め込まれていない DAP を使おうとした場合、ホストアプリケーションは、Anybus モジュールのハードウェア/ソフトウェアをリセットし、正しい DAP (PDEV を含む DAP、または PDEV を含まない DAP) を用いて Anybus モジュールを再度初期化できます。



Note: The command AR\_Info\_Ind is always sent to the host PROFIBUS IO Object (F6h), if it is not needed/used the host application responds with 05h, "Unsupported command".

この図では、IO コントローラーが何を扱うかに応じて DAP を切り替える機能のフローチャートが示されています。

DAP がスロット 0 に埋め込まれると、その他のスロットも設定可能となります。これらの他のモジュールについて重要なのは、モジュール識別番号によりモジュールの種類を一意に識別できなければならないということです(例えば、デジタル入力モジュールは、デジタル出力モジュールと同じ ID 番号を持つことはできません)。DAP については、この規則に関して例外が 1 つあります。DAP は、PDEV を含む場合も PDEV を含まない場合も、同じモジュール識別番号を持つことができます。

ホストアプリケーションは、可能であれば、最後に使用した DAP を不揮発性メモリに保存し、次の電源オフ/オン時にその DAP を使用することを推奨します。これは、接続確立に要する時間を短縮するためです。DAP が保存されていない場合、DAP2 が使用されます。最後に使用した DAP を保存できない場合、ホストアプリケーションは、最初に必ず DAP2 を埋め込む必要があります。

## I. 著作権情報

本製品には、Carnegie Mellon、Massachusetts Institute of Technology、University of California、RSA Data Security により開発されたソフトウェアが含まれています。

\*\*\*\*\*

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.

本コードは、Stanford University の Stephen Deering により Berkeley に寄贈されたソフトウェアに由来しています。

以下の条件を満たす場合、変更の有無にかかわらず、ソース形式およびバイナリ形式での再配布と使用を認めます。

- ソースコードの再配布においては、上記の著作権情報、この条件一覧、および後述の免責事項を記述すること。
- バイナリ形式の再配布においては、上記の著作権情報、この条件一覧、および後述の免責事項を、配布物に付属する文書および/またはその他の資料にて再現させること。
- 大学の名前または貢献者の名前を、書面による事前の許可なしに、本ソフトウェアより派生した製品の保証または販売促進のために使用しないこと。

本ソフトウェアは、リージェントおよび貢献者により "現状のまま" 提供されます。商品性に関する暗示的な保証、および特定の目的に対する適合性などについては、明示的または暗示的を問わず、一切保証しません。いかなる場合でも、リージェントまたは貢献者は、損害の可能性について報告を受けていたとしても、責任の法理、契約、厳格責任、または本ソフトウェアの使用に起因する不法行為 (過失かどうかは関係なく) において、いかなる直接的、間接的、偶発的、特殊的、典型的、または必然的な損害 (代替品または代替サービスの調達、効用、データ、または利益の喪失、業務の中断など) に関する責めを負わないものとします。

\*\*\*\*\*

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

本ソフトウェアをコピーおよび使用するためのライセンスは、本ソフトウェアまたは本機能について記述または参照している全ての資料において、"RSA Data Security, Inc. MD4 Message-Digest Algorithm" と明記されている場合に与えられます。

派生物について記述または参照している全ての資料において、"derived from the RSA Data Security, Inc. MD4 Message-Digest Algorithm" と明記されている場合、派生物の作成および使用に対してもライセンスが与えられます。

RSA Data Security, Inc. は、本ソフトウェアの商品性または特定の目的に対する本ソフトウェアの適合性に関し、いかなる表明も行いません。本ソフトウェアは "現状のまま" 提供され、明示的または暗示的を問わず、いかなる保証も行いません。

この告知は、本文書および/または本ソフトウェアの全ての部分の全てのコピーに記述されなければなりません。

\*\*\*\*\*

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

本ソフトウェアをコピーおよび使用するためのライセンスは、本ソフトウェアまたは本機能について記述または参照している全ての資料において、"RSA Data Security, Inc. MD5 Message-Digest Algorithm" と明記されている場合に与えられます。

派生物について記述または参照している全ての資料において、"derived from the RSA Data Security, Inc. MD5 Message-Digest Algorithm" と明記されている場合、派生物の作成および使用に対してもライセンスが与えられます。

RSA Data Security, Inc. は、本ソフトウェアの商品性または特定の目的に対する本ソフトウェアの適合性に関し、いかなる表明も行いません。本ソフトウェアは "現状のまま" 提供され、明示的または暗示的を問わず、いかなる保証も行いません。

この告知は、本文書および/または本ソフトウェアの全ての部分の全てのコピーに記述されなければなりません。