

# X-gateway Interface Addendum **EtherNet/IP Scanner**

Doc.Id. JCM-1201-033  
Rev. 1.11



---

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

---

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

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



# 目次

はじめに	このマニュアルについて	
	このマニュアルの使い方 .....	P-1
	重要なユーザ情報 .....	P-1
	関連マニュアル .....	P-2
	マニュアル更新履歴 .....	P-2
	慣例と用語集 .....	P-3
	販売およびサポート .....	P-4
<b>第1章</b>	<b>EtherNet/IP スキャナ・インターフェースについて</b>	
	概要 .....	1-1
	特長 .....	1-1
	外観図 .....	1-2
	インターフェース・ステータス LED .....	1-2
	コネクタおよびスイッチ .....	1-3
<b>第2章</b>	<b>ファイルシステム</b>	
	概要 .....	2-1
	構造 .....	2-2
	システム・ファイル .....	2-2
<b>第3章</b>	<b>ネットワークのコンフィグレーション</b>	
	TCP/IP 設定 .....	3-1
	HICP (Anybus IPconfig) .....	3-2
	DHCP .....	3-2
	速度と二重通信方式 .....	3-2
	IP アクセス制御 .....	3-3
<b>第4章</b>	<b>ウェブ・インターフェース</b>	
	概要 .....	4-1
	スキャン・リスト・コンフィグレーション .....	4-2
<b>第5章</b>	<b>データ交換</b>	
	概要 .....	5-1
	Control Word および Status Word の詳細 .....	5-2
	<i>Status Word</i> .....	5-2
	<i>Control Word</i> .....	5-2
	Statistics List/Live List の解釈 .....	5-3

## 第6章

### CIP オブジェクトの実装

概要 .....	6-1
ID オブジェクト、クラス 01h.....	6-2
概要.....	6-2
クラスのアトリビュート.....	6-2
インスタンスのアトリビュート.....	6-2
詳細：ステータスのアトリビュート.....	6-3
メッセージ・ルータ、クラス 02h.....	6-4
概要.....	6-4
クラスのアトリビュート.....	6-4
インスタンスのアトリビュート.....	6-4
アセンブリ・オブジェクト、クラス 04h.....	6-5
概要.....	6-5
クラスのアトリビュート.....	6-6
インスタンス.....	6-6
インスタンスのアトリビュート (インスタンス 101、768..831).....	6-6
インスタンスのアトリビュート (インスタンス 102、1024..1087).....	6-6
接続マネージャ・オブジェクト、クラス 06h.....	6-7
概要.....	6-7
クラスのアトリビュート.....	6-7
インスタンスのアトリビュート、インスタンス 01b.....	6-7
詳細：クラス1 接続.....	6-8
詳細：クラス3 ターゲット接続.....	6-8
診断オブジェクト、クラス AAh.....	6-9
概要.....	6-9
クラスのアトリビュート.....	6-9
インスタンスのアトリビュート、インスタンス 01b.....	6-9
接続コンフィグレーション・オブジェクト、クラス F3h.....	6-10
概要.....	6-10
クラスのアトリビュート.....	6-10
インスタンスのアトリビュート.....	6-10
ポート・オブジェクト、クラス F4h.....	6-13
概要.....	6-13
クラスのアトリビュート.....	6-13
インスタンスのアトリビュート、インスタンス 02b.....	6-13
TCP/IP インターフェース・オブジェクト、クラス F5h.....	6-14
概要.....	6-14
クラスのアトリビュート.....	6-14
インスタンスのアトリビュート.....	6-14
イーサネット・リンク・オブジェクト、クラス F6h.....	6-15
概要.....	6-15
クラスのアトリビュート.....	6-15
インスタンスのアトリビュート.....	6-15

## 第7章

### FTP サーバ

概要 .....	7-1
----------	-----

<b>第 8 章</b>	<b>電子メール・クライアント</b>	
	概要 .....	8-1
	SMTP サーバ設定 .....	8-1
	電子メール定義 .....	8-2
<b>アペンディックス A</b>	<b>SSI (Server Side Include)</b>	
	関数 .....	A-2
	<i>DisplayMacID</i> .....	A-2
	<i>DisplaySerial</i> .....	A-2
	<i>DisplayFWVersion</i> .....	A-2
	<i>DisplayBLVersion</i> .....	A-2
	<i>DisplayIP</i> .....	A-2
	<i>DisplaySubnet</i> .....	A-2
	<i>DisplayGateway</i> .....	A-3
	<i>DisplayDhcpState</i> .....	A-3
	<i>StoreIPConfig</i> .....	A-3
	<i>GetText</i> .....	A-4
	<i>printf</i> .....	A-4
	<i>scanf</i> .....	A-7
	<i>IncludeFile</i> .....	A-8
	<i>SaveToFile</i> .....	A-8
	<i>SaveDataToFile</i> .....	A-8
	<i>DisplayScannerMode</i> .....	A-9
	<i>SetScannerMode</i> .....	A-9
	SSI 出力の変更 .....	A-10
	SSI 出力文字列ファイル .....	A-10
	一時的な SSI 出力変更 .....	A-11
	ゲートウェイ制御 .....	A-11
	動的ゲートウェイ・ステータス情報の更新 .....	A-11
	ゲートウェイの再起動 .....	A-11
<b>アペンディックス B</b>	<b>技術仕様</b>	
	スキャナ・インターフェースの詳細 .....	B-1
	LAN (イーサネット) コネクタのピンアウト (RJ45) .....	B-1

## このマニュアルについて

### このマニュアルの使い方

このマニュアルでは、Anybus X-gateway の EtherNet/IP スキャナ・インターフェースを操作する際に必要な、ネットワーク固有の機能と手順について説明します。Anybus X-gateway の一般的な情報と操作手順については、"Anybus-X Generic Gateway User Manual" を参照してください。

このマニュアルの読者は、ローカル・エリア・ネットワークおよび一般的な通信システムに精通していることを前提にしています。

詳細な情報や資料などについては、HMS のウェブサイト ([www.anybus.com](http://www.anybus.com)) を参照してください。

### 重要なユーザ情報

このマニュアルに含まれるデータや説明には拘束力がありません。HMS Industrial Networks AB は、継続的な製品開発を旨とする当社のポリシーに則って、弊社の製品を改良する権利を留保します。このマニュアルに含まれる情報は予告なく変更される場合があります。ただし、HMS Industrial Networks AB は変更に関して義務を負うものではありません。HMS Industrial Networks AB はこのマニュアルに現れるあらゆるエラーに対して責任を負いません。

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

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

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

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



## 慣例と用語集

このマニュアルでは下記の慣例を使用しています。

- 番号が付いたリストは、連続した手順を示します。
- 黒丸が付いたリストは、手順ではなく情報を示します。
- 'X-gateway' という用語は、Anybus X-gateway を意味します。
- 'スキャナ・インターフェース' という用語は、Anybus-X EtherNet/IP スキャナ・インターフェースを意味します。
- 'ユーザ・マニュアル' という用語は、"Anybus-X Generic Gateway User Manual" を意味します。
- 16 進値は NNNNh というフォーマットでライトされます。NNNN は 16 進法の値です。
- 16/32 ビット値は、特に指示がないかぎり、通常は Motorola (ビッグ・エンディアン) フォーマットで保存されます。
- "バイト" という用語は、8 ビットの文字列を意味します。



## 販売およびサポート

販売		サポート	
<b>HMS Sweden (Head Office)</b>			
E-mail:	sales@hms.se	E-mail:	support@hms-networks.com
Phone:	+46 (0) 35 - 17 29 56	Phone:	+46 (0) 35 - 17 29 20
Fax:	+46 (0) 35 - 17 29 09	Fax:	+46 (0) 35 - 17 29 09
Online:	www.anybus.com	Online:	www.anybus.com
<b>HMS North America</b>			
E-mail:	us-sales@hms-networks.com	E-mail:	us-support@hms-networks.com
Phone:	+1-312 - 829 - 0601	Phone:	+1-312-829-0601
Toll Free:	+1-888-8-Anybus	Toll Free:	+1-888-8-Anybus
Fax:	+1-312-629-2869	Fax:	+1-312-629-2869
Online:	www.anybus.com	Online:	www.anybus.com
<b>HMS Germany</b>			
E-mail:	ge-sales@hms-networks.com	E-mail:	ge-support@hms-networks.com
Phone:	+49 (0) 721-96472-0	Phone:	+49 (0) 721-96472-0
Fax:	+49 (0) 721-96472-10	Fax:	+49 (0) 721-96472-10
Online:	www.anybus.de	Online:	www.anybus.de
<b>HMS Japan</b>			
E-mail:	jp-sales@hms-networks.com	E-mail:	jp-support@hms-networks.com
Phone:	+81 (0) 45-478-5340	Phone:	+81 (0) 45-478-5340
Fax:	+81 (0) 45-476-0315	Fax:	+81 (0) 45-476-0315
Online:	www.anybus.jp	Online:	www.anybus.jp
<b>HMS China</b>			
E-mail:	cn-sales@hms-networks.com	E-mail:	cn-support@hms-networks.com
Phone:	+86 (0) 10-8532-3183	Phone:	+86 (0) 10-8532-3023
Fax:	+86 (0) 10-8532-3209	Fax:	+86 (0) 10-8532-3209
Online:	www.anybus.cn	Online:	www.anybus.cn
<b>HMS Italy</b>			
E-mail:	it-sales@hms-networks.com	E-mail:	it-support@hms-networks.com
Phone:	+39 039 59662 27	Phone:	+39 039 59662 27
Fax:	+39 039 59662 31	Fax:	+39 039 59662 31
Online:	www.anybus.it	Online:	www.anybus.it
<b>HMS France</b>			
E-mail:	fr-sales@hms-networks.com	E-mail:	fr-support@hms-networks.com
Phone:	+33 (0) 3 68 368 034	Phone:	+33 (0) 3 68 368 033
Fax:	+33 (0) 3 68 368 031	Fax:	+33 (0) 3 68 368 031
Online:	www.anybus.fr	Online:	www.anybus.fr
<b>HMS UK &amp; Eire</b>			
E-mail:	uk-sales@anybus.co.uk	E-mail:	support@hms-networks.com
Phone:	+44 (0) 1926 405599	Phone:	+46 (0) 35 - 17 29 20
Fax:	+44 (0) 1926 405522	Fax:	+46 (0) 35 - 17 29 09
Online:	www.anybus.co.uk	Online:	www.anybus.com
<b>HMS Denmark</b>			
E-mail:	info@anybus.dk	E-mail:	support@hms-networks.com
Phone:	+45 (0) 22 30 08 01	Phone:	+46 (0) 35 - 17 29 20
Fax:	+46 (0) 35 17 29 09	Fax:	+46 (0) 35 - 17 29 09
Online:	www.anybus.com	Online:	www.anybus.com
<b>HMS India</b>			
E-mail:	in-sales@anybus.com	E-mail:	in-support@hms-networks.com
Phone:	+91 (0) 20 40111201	Phone:	+46 (0) 35 - 17 29 20
Fax:	+91 (0) 20 40111105	Fax:	+46 (0) 35 - 17 29 09
Online:	www.anybus.com	Online:	www.anybus.com

# EtherNet/IP スキャナ・インターフェースについて

## 概要

EtherNet/IP スキャナ・インターフェースでは、Anybus X-gateway プラットフォームの EtherNet/IP スキャナ機能を提供します。このインターフェースは最大 64 の EtherNet/IP ノード（アダプタ）とデータを交換し、組み込み FTP およびウェブ・サーバや電子メール・クライアント機能などの追加の IT 機能を備えています。

動的コンテンツ機能では、入力 / 出力バッファからのデータをウェブ・ページで監視したり、電子メール・メッセージに含めたりすることができます。

下記も参照してください。

5-1 “データ交換”

## 特長

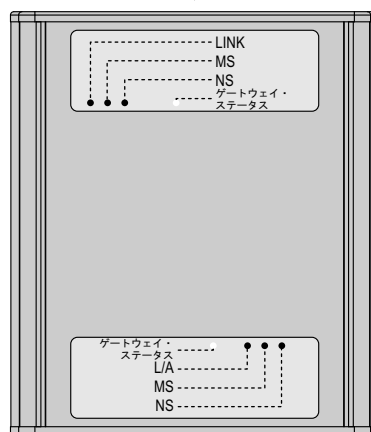
- EtherNet/IP スキャナ
- FTP サーバ
- ウェブ・サーバ
- SMTP クライアント
- 10/100Mbit 動作（全二重または半二重）
- オンボードの IP コンフィグレーション・スイッチ
- シールドまたは非シールド・ケーブル

## 外観図

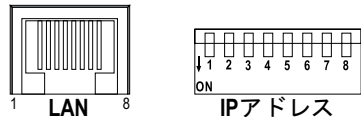
### インターフェース・ステータス LED

LED	色	意味
L/A	グリーン	リンク確立済み
	グリーン、点滅	アクティビティ（データ送受信中）
	オフ	リンクなし、または電源オフ
MS	グリーン	デバイス操作可能 - スキャナが Run 状態にある
	グリーン、点滅	スタンバイ - スキャナが Idle 状態にある - スキャナが設定されていない
	レッド	メジャー障害 - 回復不能なメジャー障害
	レッド、点滅	マイナー障害 - 回復可能なマイナー障害（送信元またはタイムアウト） - 送信元接続を開くことができなかった
	交互にレッド/グリーン	自己テスト - スキャナ起動自己テストの進行中
	オフ	電源オフ
NS	グリーン	接続済み - 少なくとも 1 つの EtherNet/IP 接続が確立されている（送信先または送信元）
	グリーン、点滅	接続なし - EtherNet/IP 接続が確立されていない（クラス 1 またはクラス 3、送信先または送信元）
	レッド	重複 IP - 設定した IP アドレスがすでに使用されている
	レッド、点滅	接続タイムアウト - 1 つまたは複数の EtherNet/IP ターゲット接続がタイムアウトしている - タイムアウトした全てのターゲット接続が再確立された場合、またはゲートウェイがリセットされた場合にのみ、スキャナはこの状態を抜け出すことが可能
	交互にレッド/グリーン	自己テスト - スキャナ起動自己テストの進行中
オフ	電源オフ、または IP アドレスなし（詳細については、ユーザ・マニュアルを参照）	
ゲートウェイ・ステータス	-	

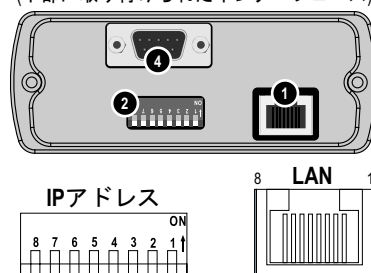
上部に取り付けられたインターフェース



下部に取り付けられたインターフェース



(下部に取り付けられたインターフェース)



## コネクタおよびスイッチ

#	説明
1	LAN（イーサネット）コネクタ B-1 “LAN（イーサネット）コネクタのピンアウト（RJ45）” を参照
2	手動の IP コンフィグレーション・スイッチ 3-1 “ネットワークのコンフィグレーション” を参照
3	ゲートウェイ電源コネクタ （詳細については、ユーザ・マニュアルを参照）
4	ゲートウェイ・コンフィグレーション・コネクタ （詳細については、ユーザ・マニュアルを参照）

# ファイルシステム

## 概要

スキャナ・インターフェースは、組み込みのファイルシステムを備えています。ウェブ・ファイル、ネットワーク通信設定、電子メール・メッセージなどの情報を保存するには、このファイルシステムを使用します。ファイルシステムには、FTP またはウェブ・サーバ経由でアクセスするか、組み込みの電子メール・クライアント（SSI の場合）を介してアクセスできます。

### 慣例と制限事項

- '\ ' (円記号) は、パス区切り文字として使用されます。
- 'パス' はシステム・ルートを開始するため、'\ ' で開始する必要があります。
- 'パス' は '\ ' で終了しないようにしてください。
- 名前にはスペース (' ') を含めることができますが、スペースで開始または終了しないようにしてください。
- 名前には以下の記号は使用しないでください。'\ / : \* ? " < > | '
- 名前は 48 文字 (+ NULL 終端文字) 以下でなければなりません。
- パスは 256 文字 (ファイル名を含む) 以下でなければなりません。
- 同時に開けるファイルの最大数は 40 です。
- 同時に開けるディレクトリの最大数は 40 です。

### 記憶領域

ファイルシステムは、以下の 3 つの異なる記憶領域を備えています。

- **領域 0 (115kByte、不揮発性)**  
この領域は、ウェブ・ファイルなどの静的ファイルに対して使用されます。
- **領域 1 (128kByte、不揮発性)**  
この領域は、コンフィグレーション・ファイルなどに対して使用されます。
- **領域 2 (1024kByte、揮発性)**  
この領域は、一時的な記憶領域として使用されます。

---

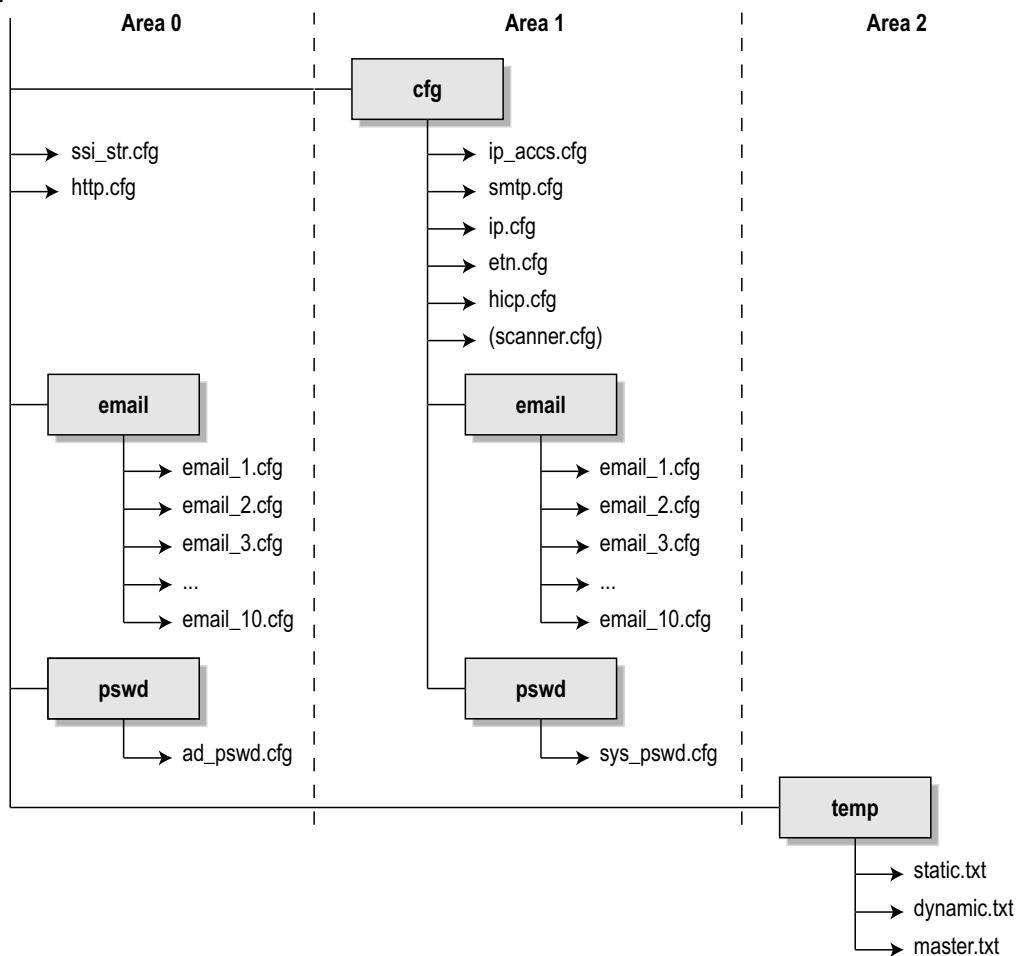
### 重要：

不揮発性の記憶領域はフラッシュ・メモリ内に配置されます。各 FLASH セグメントは、このタイプのメモリの性質上、約 100000 回しか消去できません。

以下の操作を行うと、1 つ以上の FLASH セグメントが消去されます。

- ファイルまたはディレクトリの削除、移動、または名前変更
- 既存ファイルへのデータのライトまたは追加
- ファイルシステムのフォーマット
- スキャナ・コンフィグレーションの保存

## 構造



## システム・ファイル

ファイルシステムには、システム・コンフィグレーションで使用される一連のファイルが含まれています。これらのファイル ("システム・ファイル" と呼ばれる) は、標準テキスト・エディタ (Microsoft Windows™ のメモ帳など) を使用して変更可能な通常の ASCII ファイルです。また、Server Side Include 機能の使用時には、組み込みのウェブ・サーバによって間接的にこれらのファイルを変更することもできます。一般に、システム・ファイルのフォーマットは 'キー' のコンセプトに基づいており、各 'キー' には値を割り当てることができます (下記の例を参照)。

### 例:

```
[Key1]
value of key1

[Key2]
value of key2
```

各システム・ファイルの正確なフォーマット仕様については、このマニュアルの後半で説明します。

### 重要:

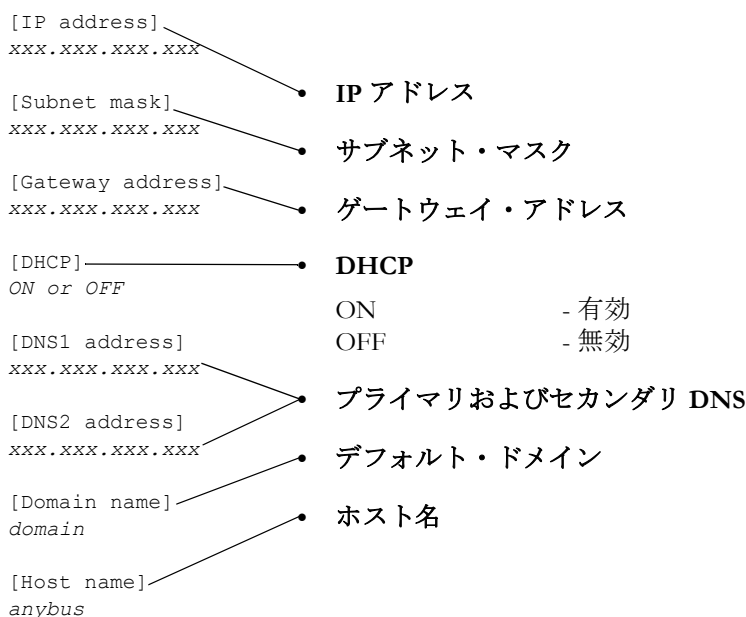
前述の内容に反して、'`\cfg\scanner.cfg`' ファイルはスキャナ・コンフィグレーションをバイナリ・フォーマットで保持します。このファイルはスキャナ・インターフェースによって自動的に作成されますが、手動で変更しないようにしてください。

# ネットワークのコンフィグレーション

## TCP/IP 設定

ネットワーク上で参加できるようにするには、スキャナ・インターフェースは有効な TCP/IP コンフィグレーションを必要とします。これらの設定は、システム・ファイル '\cfg\ip.cfg' に保存されます。

ファイル・フォーマット：



ネットワーク設定は、以下による影響を受ける場合もあります。

- EtherNet/IP (6-14 “TCP/IP インターフェース・オブジェクト、クラス F5h” を参照)
- DHCP (3-2 “DHCP” を参照)
- HICP (3-2 “HICP (Anybus IPconfig)” を参照)
- SSI (A-1 “SSI (Server Side Include)” を参照)
- スイッチ (下記を参照)

オンボード・スイッチを 0 (ゼロ) に設定した場合、スキャナ・インターフェースは '\cfg\ip.cfg' に保存されている設定を使用します。0 に設定しなかった場合、スイッチ設定は以下のようにこのファイル内の特定の設定を無効にします。

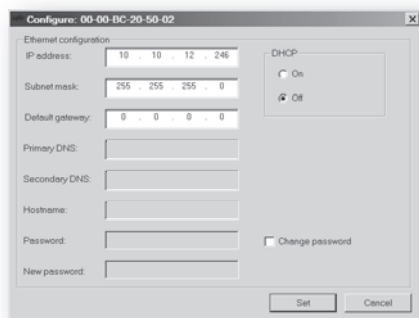
SW1	SW2	SW3	SW4	SW5	SW6	SW7	SW8	DHCP	サブネット	ゲートウェイ	IP
OFF	OFF	OFF	OFF	OFF	OFF	OFF	OFF		'\cfg\ip.cfg' によって決定される		
OFF	OFF	OFF	OFF	OFF	OFF	OFF	ON	Disabled	255.255.255.0	192.168.0.255	192.168.0.1
OFF	OFF	OFF	OFF	OFF	OFF	ON	OFF	Disabled	255.255.255.0	192.168.0.255	192.168.0.2
...	...	...	...	...	...	...	...	...	...	...	...
ON	ON	ON	ON	ON	ON	OFF	ON	Disabled	255.255.255.0	192.168.0.255	192.168.0.253
ON	ON	ON	ON	ON	ON	ON	OFF	Disabled	255.255.255.0	192.168.0.255	192.168.0.254
ON	ON	ON	ON	ON	ON	ON	ON	(無効な設定)			

**注意：**スイッチは起動時に読み取られます。変更内容を有効にするためには、リセットを行う必要があります。

## HICP (Anybus IPconfig)

スキャナ・インターフェースは、Anybus IPconfig ユーティリティ (HMS のウェブサイトから無償でダウンロード可能) によって使用される HICP プロトコルをサポートしています。このユーティリティを使用すると、UDP ポート 3250 を介してネットワークに接続された任意の Anybus 製品のネットワーク設定にアクセスできます。

プログラムの起動時には、Anybus 製品についてネットワークがスキャンされます。ネットワークは、'Scan' をクリックすることでいつでも再スキャンできます。検出されたデバイスのリストでは、スキャナ・インターフェースは 'ABM-EIP' として表示されます。



インターフェースの設定を変更するには、リスト内のエントリをダブル・クリックします。スキャナ・インターフェースに関連する設定が含まれているウィンドウが表示されます。

'Set' をクリックして、新しい設定を確認します。新しい IP コンフィグレーションは、'\cfg\ip.cfg' に保存されます。

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

パスワードは、'\cfg\hicp.cfg' システム・ファイルに保存されます。

### ファイル・フォーマット:

```
[Password]
<password>
```

## DHCP

スキャナ・インターフェースは、TCP/IP 設定を DHCP から取得できます。取得に失敗した場合、スキャナ・インターフェースは現在の設定 ('\cfg\ip.cfg' に現在保存されている設定) に依存します。

現在の設定を使用できない場合 (つまり、0 に設定されている場合)、スキャナ・インターフェースは停止し、オンボード・ステータス LED でエラーを示します (ただし、ネットワーク・コンフィグレーションは HICP 経由で引き続きアクセス可能。3-2 “HICP (Anybus IPconfig)” を参照)。

## 速度と二重通信方式

スキャナ・インターフェースは、全二重または半二重での 10/100Mbit 動作をサポートしています。これらの設定は、システム・ファイル 'cfg\etn.cfg' に保存されます。設定は、イーサネット・リンク・オブジェクト経由で変更することもできます (6-15 “イーサネット・リンク・オブジェクト、クラス F6h” を参照)。

### ファイル・フォーマット:

```
[AutoNeg]
xxx ----- オートネゴシエーション。有効な設定: 'ON' または 'OFF'
```

```
[Speed]
xxx ----- 速度。有効な設定: '100' または '10'
```

```
[Duplex]
xxxx ----- 二重通信方式。有効な設定: 'FULL' または 'HALF'
```



## IP アクセス制御

スキャナ・インターフェースへの接続が許可される IP アドレスを指定できます。この情報は、システム・ファイル '\cfg\ip\_accs.cfg' に保存されます。

### ファイル・フォーマット：

- |                                  |   |   |
|----------------------------------|---|---|
| [Web]<br>xxx.xxx.xxx.xxx         | • | ここにリストされているノードはウェブ・サーバにアクセスできます。                  |
| [FTP]<br>xxx.xxx.xxx.xxx         | • | ここにリストされているノードはFTPサーバにアクセスできます。                   |
| [Ethernet/IP]<br>xxx.xxx.xxx.xxx | • | ここにリストされているノードは、EtherNet/IP 経由でインターフェースに接続できます。   |
| [All]<br>xxx.xxx.xxx.xxx         | • | フォールバック設定。上のキーの1つまたは複数省略された場合にインターフェースによって使用されます。 |

**注意：** '\*' は、IP シリーズを選択するためのワイルドカードとして使用できます。

### 重要：

いかなる状況においても、[All] にはアドレス '0.0.0.0' を入力しないでください。そのように設定すると、スキャナ・インターフェースへの外部アクセスが完全に妨げられます。この注意を守らないと、製品が使用不能になり、HMS サポート・オフィスからのサービスが必要になります。

# ウェブ・インターフェース

## 概要

Ethernet インターフェースは、SSI 機能を備えた高速でフル機能のウェブ・サーバを特徴としています。デフォルトのウェブ・インターフェースは、任意の標準ウェブ・ブラウザを介して最も一般的なオプションへのアクセスを提供します。ただし、ウェブ・インターフェースはカスタマイズ可能であり、特定の製品に合わせて完全に設計することもできます。

デフォルトのウェブ・インターフェースはゲートウェイの反対側のネットワーク・タイプ（スレーブ、マスター、フィールバス・タイプなど）に応じて多少異なりますが、基本機能は実質的に同じです。スキャン・リスト・コンフィグレーションのオプションについては、次のページで説明します。

- **General Status**

このページには、ゲートウェイ初期化パラメータと一般的なゲートウェイ診断の概要が表示されます（これらの値は、ゲートウェイ・コンフィグレーション・インターフェースで設定された値に対応します）。

- **IP Config**

このページには、現在の TCP/IP 設定、DNS コンフィグレーション、および SMTP サーバ設定が保持されます。

- **About**

このページには、ソフトウェア・バージョン・ナンバーとゲートウェイの各種コンポーネントのシリアル・ナンバーが保持されます。また、イーサネット・インターフェースの MAC-ID も保持されます。



開始ページ



'General Status' ページ



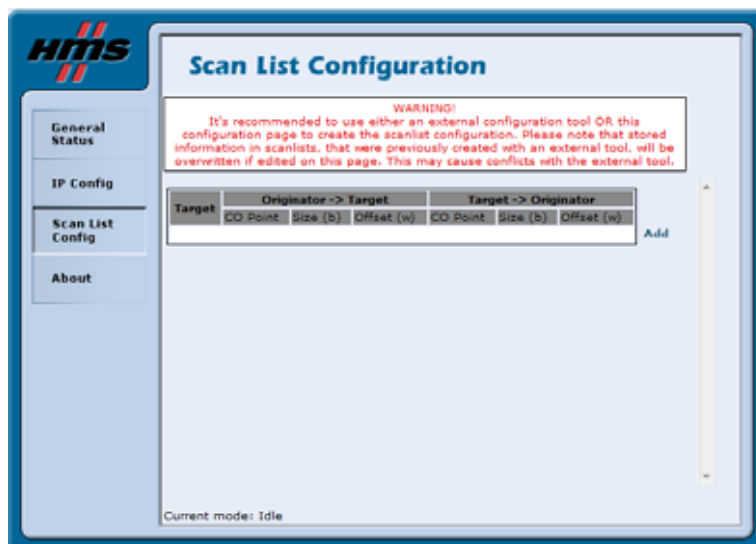
'IP Config' ページ

## スキャン・リスト・コンフィグレーション

イーサネット・ウェブ・インターフェースは、EtherNet/IP ネットワークのスキャン・リストを設定するオプションを提供します。

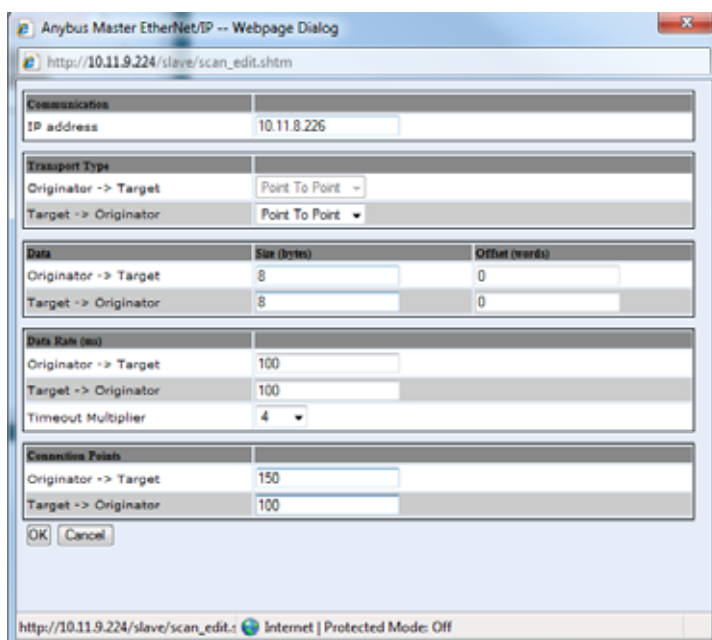
スキャン・リストを保存できるようにするには、スキャナがアイドル・モードになっている必要があります。スキャナをアイドル・モードにするには、HyperTerminal を使用して、ゲートウェイ・コンフィグレーション・インターフェースで操作モードを変更します。

'Scan List Config' メニュー項目を選択して、現在空になっているスキャン・リストを表示します。



左下隅に表示されている現在のモードをチェックして、スキャナがアイドル・モードになっていることを確認します。

ネットワーク接続を追加するには、リストの右側にある 'Add' ボタンを押します。これにより、以下のウィンドウが表示されます。

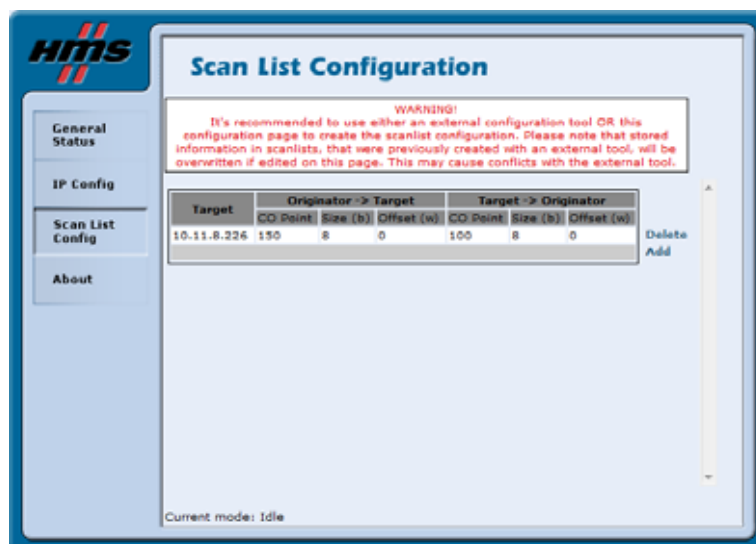


**注意：**オリジネータはスキャナに相当し、ターゲットはアダプタに相当します。ウェブ・インターフェースの項目については、次のページを参照してください。

## 編集可能な設定：

項目	#	説明
Communication	IP Address	アダプタの IP アドレス
Transport Type	Originator -> Target	'Point to Point' (ユニキャスト) が唯一のオプションです。
	Target -> Originator	'Point to Point' (ユニキャスト) または 'Multicast' を選択しません。
Data	Originator -> Target	アダプタに送信されるデータのサイズ (バイト単位) アダプタの入力領域メモリの先頭からのオフセット
	Target -> Originator	スキャナに送信されるデータのサイズ (バイト単位) スキャナの入力領域メモリの先頭からのオフセット
Data Rate (ms)	Originator -> Target	アダプタへのデータの転送間隔 (ミリ秒)
	Target -> Originator	スキャナへのデータの転送間隔 (ミリ秒)
	Timeout Multiplier	データ・レート (要求パケット間隔) に適用されるこの数値は、接続が切れるまでの許容経過時間を示します。
Connection Points	Originator -> Target	接続ポイントはアセンブリ・インスタンスに相当します。これは、製品の製造業者から入手できます。標準 Anybus アダプタ (ターゲット) はアセンブリ・インスタンス 150 を使用します。ドライブ・プロファイル Anybus アダプタ (ターゲット) はアセンブリ・インスタンス 20 を使用します。
	Target -> Originator	接続ポイントはアセンブリ・インスタンスに相当します。これは、製品の製造業者から入手できます。標準 Anybus スキャナ (オリジネータ) はアセンブリ・インスタンス 100 を使用します。ドライブ・プロファイル Anybus スキャナ (オリジネータ) はアセンブリ・インスタンス 70 を使用します。

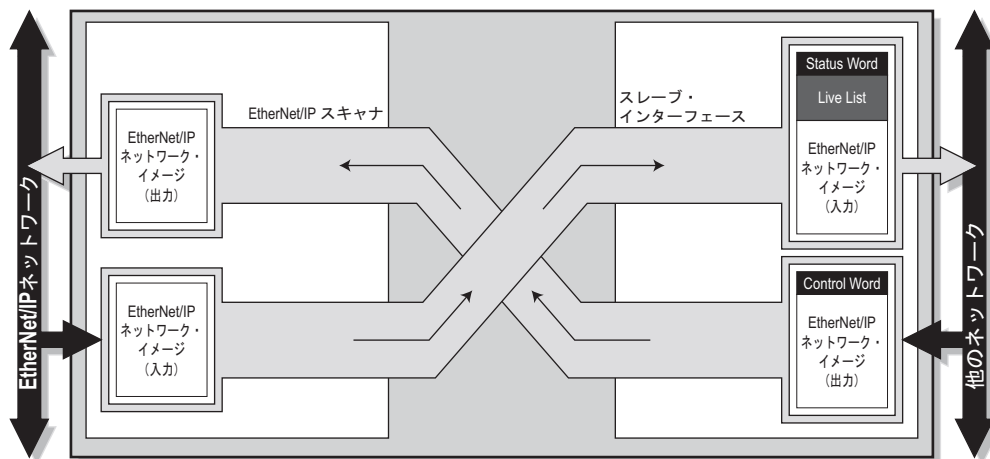
スキャン・リストでの項目の設定が終了したら、'OK' ボタンを押します。新しく設定した項目がリストに表示されます。



## データ交換

### 概要

スキャナ・インターフェースは、以下のように別のネットワークとデータを交換します。



EtherNet/IP ネットワーク I/O の構造は、コンフィグレーション・ツールで作成されたコンフィグレーションによって決まります。

**注意：**スキャナ・インターフェースの入力 / 出力データ・サイズは、コンフィグレーション・ツールを使用して作成された実際のコンフィグレーションのサイズによって決定され、ゲートウェイ・コンフィグレーション・インターフェースで設定することはできません。

## Control Word および Status Word の詳細

### Status Word

ユーザ・マニュアルに記載されているように、Status Word は一般ステータス情報を保持します。

(詳細については、ユーザ・マニュアルを参照)

### Control Word

Control Word (有効にした場合) は、EtherNet/IP ネットワーク上の他のノードへの通信を制御します。現在の状態は、Control Word/Status Word でフィールドバス側から設定するか、または端末インターフェースから設定できます。

Control Word の内容 :



モード (b3、b2)	意味	コメント
00b	アイドル	これらの設定は、フィールドバス側、端末インターフェース、またはゲートウェイ・コンフィグレーション・インターフェースから構成できます。
01b		
10b	実行	
11b		

リセット (b7)	意味	コメント
0b	通常動作	-
1b	ゲートウェイのリセット	このビットをセットすると、ゲートウェイが自己リセットを実行します。

**重要:** Control Word と Status Word は、ゲートウェイ・コンフィグレーション・インターフェースで無効にすることができます。その場合は、起動初期化シーケンスの完了後にマスター・インターフェースが自動的に実行モードに入ります。有効にした場合、スキャナはアイドル・モードで起動し、モードが'実行モード'に変更されるまではデータを交換しません。

## Statistics List/Live List の解釈

開いていない接続はネットワーク上で検出できないため、それらの接続をEtherNet/IPネットワーク・マスターの Live List に追加することはできません。開いている接続のステータスは検出可能であり、この情報はリードされて Address Live List に追加されます。このリストは、EtherNet/IP ネットワーク上で開いている接続（設定済みの接続、アクティブな接続、または障害が発生している接続）の数に関するステータス情報を保持します。この Statistics List は、フィールドバスの特定の領域でアドレス 640h からリードすることができます。

DPRAM アドレス	Address Live List (バイト)	説明
640h - 641h	0 - 1	設定済み接続の数
642h - 643h	2 - 3	アクティブな接続の数
644h - 645h	4 - 5	障害が発生している接続の数
646h - 647h	6 - 7	(予約)

Live List の詳細については、ユーザ・マニュアルを参照してください。

# CIP オブジェクトの実装

## 概要

EtherNet/IP は、CIP (Common Industrial Protocol) に基づいています。CIP は、ノード間でデータを交換するために DeviceNet および ControlNet によって使用されるアプリケーション・レイヤです。

以下の CIP オブジェクトが実装されています。

- ID オブジェクト、クラス 01h
- メッセージ・ルータ、クラス 02h
- アセンブリ・オブジェクト、クラス 04h
- 接続マネージャ・オブジェクト、クラス 06h
- 診断オブジェクト、クラス AAh
- 接続コンフィグレーション・オブジェクト、クラス F3h
- ポート・オブジェクト、クラス F4h
- TCP/IP インターフェース・オブジェクト、クラス F5h
- イーサネット・リンク・オブジェクト、クラス F6h



## ID オブジェクト、クラス 01h

### 概要

#### オブジェクトの説明

-

#### サポートされるサービス

クラス・サービス : Get Attribute All  
Get Attribute Single

インスタンス・サービス : Get Attribute All  
Get Attribute Single  
Set Attribute Single  
Reset

#### クラスのアトリビュート

#	アクセス	名前	タイプ	値
1	Get	Revision	UINT	0001h

#### インスタンスのアトリビュート

#	アクセス	名前	タイプ	値
1	Get	Vendor ID	UINT	005Ah (HMS Industrial Networks AB)
2	Get	Device Type	UINT	000Ch (通信アダプタ)
3	Get	Product Code	UINT	0046h (Anybus-X EtherNet/IP スキャナ)
4	Get	Revision	構成 :	
			USINT	メジャー・フィールドバス・バージョン
			USINT	マイナー・フィールドバス・バージョン
5	Get	Status	WORD	6-3 " 詳細 : ステータスのアトリビュート " を参照
6	Get	Serial Number	UDINT	(シリアル・ナンバー)
7	Get	Product Name	SHORT_STRING	"Anybus-X EtherNet/IP Scanner"
103	Set	Scanner Mode	USINT	0: アイドル・モード
				1: 実行モード

## 詳細：ステータスのアトリビュート

ビット	名前	コメント														
0	Module Owned	-														
1	(予約)	-														
2	Configured	-														
3	(予約)	-														
4...7	Extended Device Status	<table border="0"> <tr> <td>値：</td> <td>意味：</td> </tr> <tr> <td>0000b</td> <td>不明</td> </tr> <tr> <td>0010b</td> <td>失敗した I/O 接続</td> </tr> <tr> <td>0011b</td> <td>I/O 接続未確立</td> </tr> <tr> <td>0100b</td> <td>不揮発性コンフィグレーション不良</td> </tr> <tr> <td>0110b</td> <td>実行モードでの接続</td> </tr> <tr> <td>0111b</td> <td>アイドル・モードでの接続</td> </tr> </table>	値：	意味：	0000b	不明	0010b	失敗した I/O 接続	0011b	I/O 接続未確立	0100b	不揮発性コンフィグレーション不良	0110b	実行モードでの接続	0111b	アイドル・モードでの接続
値：	意味：															
0000b	不明															
0010b	失敗した I/O 接続															
0011b	I/O 接続未確立															
0100b	不揮発性コンフィグレーション不良															
0110b	実行モードでの接続															
0111b	アイドル・モードでの接続															
8	Minor recoverable fault	-														
9	Minor recoverable fault	-														
10	Major recoverable fault	-														
11	Major unrecoverable fault	-														
12...15	(予約)	-														

## メッセージ・ルータ、クラス 02h

### 概要

#### オブジェクトの説明

-

#### サポートされるサービス

クラス・サービス： -

インスタンス・サービス： -

#### クラスのアトリビュート

-

#### インスタンスのアトリビュート

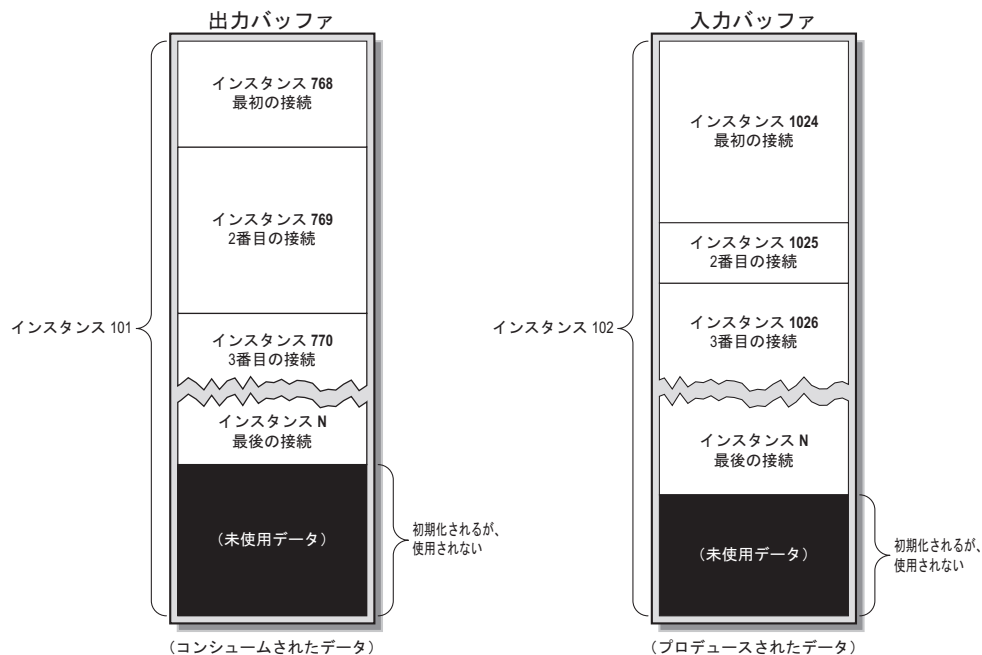
-

## アセンブリ・オブジェクト、クラス 04h

### 概要

#### オブジェクトの説明

入力および出力バッファは、インスタンス 101 および 102 を通して表されます。また、各 EtherNet/IP 接続に関連付けられているデータは、専用インスタンス（768...831 および 1024...1087）として表されます。



#### サポートされるサービス

クラス・サービス : Get Attribute Single  
 インスタンス・サービス : Get Attribute Single  
 Set Attribute Single

## クラスのアトリビュート

#	アクセス	名前	タイプ	値
1	Get	Revision	UINT	0002h
2	Get	Max Instance	UINT	-

## インスタンス

#	内容	説明
3	ハートビート・インスタンス	1つの実際の接続ポイントにのみアクセスする接続のハートビート・インスタンスとして使用されます。このインスタンスに対するアトリビュートは実装されません。
101	コンシュームされたデータ (全ての接続)	出力バッファにマッピングされます。インスタンス 768...831の内容に相当します。
102	プロデュースされたデータ (全ての接続)	入力バッファにマッピングされます。インスタンス 1024...1087の内容に相当します。
768	コンシュームされたデータ (最初の接続)	I/O 接続 #1に関連付けられているコンシュームデータ (コンフィグレーション・オブジェクト・インスタンス 1)
769	コンシュームされたデータ (2番目の接続)	I/O 接続 #2に関連付けられているコンシュームデータ (コンフィグレーション・オブジェクト・インスタンス 2)
...	...	...
831	コンシュームされたデータ (64番目の接続)	I/O 接続 #64に関連付けられているコンシュームデータ (コンフィグレーション・オブジェクト・インスタンス 64)
1024	プロデュースされたデータ (最初の接続)	I/O 接続 #1に関連付けられているプロデュースデータ (コンフィグレーション・オブジェクト、インスタンス 1)
1025	プロデュースされたデータ (2番目の接続)	I/O 接続 #2に関連付けられているプロデュースデータ (コンフィグレーション・オブジェクト、インスタンス 2)
...	...	...
1087	プロデュースされたデータ (64番目の接続)	I/O 接続 #64に関連付けられているプロデュースデータ (コンフィグレーション・オブジェクト、インスタンス 2)

## インスタンスのアトリビュート (インスタンス 101、768...831)

#	アクセス	名前	タイプ	値
3	Get/Set	Consumed data	BYTE の配列	出力バッファにマッピングされます。

## インスタンスのアトリビュート (インスタンス 102、1024...1087)

#	アクセス	名前	タイプ	値
3	Get	Produced data	BYTE の配列	入力バッファにマッピングされます。

## 接続マネージャ・オブジェクト、クラス 06h

### 概要

#### オブジェクトの説明

-

#### サポートされるサービス

クラス・サービス :           Get Attributes All  
                                   Forward Open  
                                   Forward Close  
                                   Unconnected Send

インスタンス・サービス :    Get Attributes All

#### クラスのアトリビュート

#	アクセス	名前	タイプ	値	説明
1	Get	Data	UINT	0001h	リビジョン 1

#### インスタンスのアトリビュート、インスタンス 01h

#	アクセス	名前	タイプ	説明
1	Get	Open Requests	UINT	受信された 'Forward Open' 要求の数
2	Get	Open Format Rejects	UINT	不正なフォーマットが原因で拒否された 'Forward Open' 要求の数
3	Get	Open Resource Rejects	UINT	リソースの欠如が原因で拒否された 'Forward Open' 要求の数
4	Get	Open Other Rejects	UINT	「不正なフォーマット」または「リソースの欠如」以外の理由で拒否された 'Forward Open' 要求の数
5	Get	Close Requests	UINT	受信された 'Forward Close' 要求の数
6	Get	Close Format Rejects	UINT	不正なフォーマットが原因で拒否された 'Forward Close' 要求の数
7	Get	Close Other Rejects	UINT	「不正なフォーマット」以外の理由で拒否された 'Forward Close' 要求の数
8	Get	Connection Timeouts	UINT	接続タイムアウトの数

## 詳細：クラス 1 接続

クラス 1 接続は、I/O データの転送に使用されます。各クラス 1 接続は、2 つのデータ伝送（コンシュームとプロデュース）を確立します。

サポートされる送信元クラス 1 接続の数：	64
最大入力接続サイズ：	509 バイト
最小出力接続サイズ：	504 バイト
サポートされるパケット・レート（API）：	2...3200ms
サポートされるトリガ・タイプ：	サイクリック

- プロデュース・アセンブリ・インスタンス 100、101、102

サポートされるターゲット・クラス 1 接続の数：	インスタンスごとに 20 (同じ伝送を共有)
サポートされる伝送の数：	1
サポートされる伝送タイプ：	ポイント・ツー・ポイント、 マルチキャスト
サポートされるパケット・レート（API）：	2...3200ms
サポートされるトリガ・タイプ：	サイクリック

クラス 1 接続が確立されると、伝送が受信されます。この伝送は、ポイント・ツー・ポイントまたはマルチキャスト・タイプの伝送にすることができます。ポイント・ツー・ポイントの場合、データは UDP ユニキャスト・メッセージを使用して転送されます。他の接続はそのデータにアクセスできません。マルチキャストの場合、データは UDP マルチキャスト・メッセージで転送されます。他の接続は同じ伝送を使用してデータにアクセスできます。

プロデュース・インスタンスには、1 つの伝送のみ割り当てることができます。従って、ポイント・ツー・ポイント接続を使用している場合は、確立できるクラス 1 接続は 1 つのみです。ただし、20 の接続を各マルチキャスト伝送にリンクできるため、それらの接続が全て同じ伝送を使用する場合は 20 のクラス 1 接続の確立が可能になります。

接続で既存の伝送を使用するには、接続データ・サイズが既存の伝送のデータ・サイズと一致している必要があります。一致していないと、エラー応答が返されます。接続 RPI（Requested Packet Interval：要求パケット間隔）が既存の接続の API（Actual Packet Interval：実際のパケット間隔）と一致しない場合は、既存の伝送の API を使用して引き続き接続が確立されます。この API は、'Forward Open' 要求への応答で返されます。

- コンシューム・アセンブリ・インスタンス 150

サポートされるターゲット・クラス 1 接続の数：	インスタンスごとに 1 つ
サポートされる伝送の数：	1
サポートされる伝送タイプ：	ポイント・ツー・ポイントのみ
サポートされるパケット・レート（API）：	無制限

コンシューム・インスタンスは出力の制御に使用されるため、各コンシューム・インスタンスに対して許可される接続は 1 つのみです。接続用の伝送はポイント・ツー・ポイントである必要があります。

## 詳細：クラス 3 ターゲット接続

メッセージ・ルータへの接続を確立するには、クラス 3 接続を使用します。その後、接続は Explicit メッセージ用に使用されます。クラス 3 接続は、TCP 接続を使用します。

メッセージ・ルータへの最大 16 の同時クラス 3 接続がサポートされています。

## 診断オブジェクト、クラス AAh

### 概要

#### オブジェクトの説明

このベンダ特有のオブジェクトは、スキャナ・インターフェースからの診断情報を提供します。

#### サポートされるサービス

クラス・サービス : Get Attribute All

インスタンス・サービス : Get Attribute Single

### クラスのアトリビュート

#	アクセス	名前	タイプ	コメント
1	Get	Revision	UINT	0001h

### インスタンスのアトリビュート、インスタンス 01h

#	アクセス	名前	タイプ	コメント
01h	Get	Serial number	UDINT	シリアル・ナンバー
02h	Get	Vendor ID	UINT	-
03h	Get	Fieldbus Type	UINT	0083h = イーサネット
04h	Get	Software version	UINT	-
0Ah	Get	Interface type	UINT	0201h = マスター
0Fh	Get	Input Buffer size	UINT	入力バッファのサイズ (バイト単位)
12h	Get	Output Buffer size	UINT	出力バッファのサイズ (バイト単位)
18h	Get	MAC ID	USINT の配列	インターフェースのイーサネット MAC ID (6 バイト)
19h	Get	IP Address	UDINT	現在の IP アドレス
1Ah	Get	Subnet mask	UDINT	現在のサブネット・マスク
1Bh	Get	Gateway address	UDINT	現在のゲートウェイ・アドレス
1Ch	Get	SMTP server address	UDINT	SMTP サーバ・アドレス
1Dh	Get	DHCP state	UDINT	0=DHCP 有効、1=DHCP 無効
1Eh	Get	Bootloader version	UDINT	インターフェース・ブートローダ・バージョン
1Fh	Get	Application interface version	UINT	アプリケーション・インターフェース・ソフトウェア・バージョン
20h	Get	Fieldbus software version	UINT	フィールドバス・インターフェース・ソフトウェア・バージョン



## 接続コンフィグレーション・オブジェクト、クラス F3h

### 概要

#### オブジェクトの説明

-

#### サポートされるサービス

クラス・サービス :

- Create
- Delete
- Restore
- Get Attribute All
- Get Attribute Single
- Set Attribute Single
- Kick Timer (4Bh)
- Change Start (4Fh)
- Get Status (50h)
- Change Complete (51h)
- Audit Changes (52h)

インスタンス・サービス :

- Delete
- Restore
- Get Attribute All
- Set Attribute All
- Get Attribute Single

#### クラスのアトリビュート

#	アクセス	名前	タイプ	値
1	Get	Revision	UINT	0002h
2	Get	Max Instance	UDINT	0000 0040h (最大インスタンス数 = 64)
3	Get	No. of instances	UDINT	-
8	Get	Format number	UINT	0065h (インスタンス・アトリビュート9のフォーマット番号)
9	Set	Edit Signature	UDINT	インスタンス・アトリビュート値の変更を検出するためにコンフィグレーション・ツールによってライトされる値

#### インスタンスのアトリビュート

接続ごとに1つのインスタンスが作成されます。インスタンスのアトリビュートは、アトリビュート #1 を除き、全て不揮発性メモリに保存されます。

#	アクセス	名前	タイプ	コメント
1	Get	Connection Status	構成 :	接続が開いていない場合、このアトリビュートは理由を示します。
		Gen_status	USINT	
		(予約)	USINT	
		Ext_status	UINT	
2	Set	Connection Flags	WORD	接続フラグ

#	アクセス	名前	タイプ	コメント
3	Set	Target Device ID	構成 :	このインスタンスに関連付けられているターゲット・デバイスを識別するためにコンフィグレーション・ソフトウェアによって使用されるデバイス ID
		Vendor_id	UINT	
		Product_code	UINT	
		Major_rev	USINT	
		Minor_rev	USINT	
5	Set	Net connection parameters	構成 :	-
		Conn_timeout	USINT	接続タイムアウト乗数
		Xport_class_and_trigger	BYTE	伝送クラスおよびトリガ
		Rpi_OT	UDINT	オリジネータからターゲットへの要求パケット間隔
		Net_OT	UINT	オリジネータからターゲットへのネットワーク接続パラメータ (このアトリビュートでは OT 接続サイズが指定される)
		Rpi_TO	UDINT	ターゲットからオリジネータへの要求パケット間隔
		Net_TO	UINT	ターゲットからオリジネータへのネットワーク・ターゲット接続パラメータ (このアトリビュートでは TO 接続サイズが指定される)
6	Set	Connection Path	構成 :	-
		Open_path_size	USINT	ワード (16 ビット) 単位でのパス・サイズ
		(予約)	USINT	予約、無視する
		Open connection path	埋め込み EPATH	接続マネージャの 'Forward Open' サービスで 사용되는パス
7	Set	Config #1 Data	構成 :	-
		Config_data_size	UINT	Config_data のバイト単位の長さ
		Config_data	オクテットの配列	コンフィグレーション #1 データ
8	Set	Connection name	構成 :	-
		Name_size	USINT	名前の文字数
		(予約)	USINT	予約、無視する
		Connection_name	STRING2	UNICODE で符号化されたユーザ割り当て接続名
9	Set	Implementation Defined Attribute	構成 :	-
		Format_number	UINT	0101h
		Impl_defined_data_size	UINT	000Ah
		Impl_defined_data	配列 : UINT、 UINT、  UINT、 UINT、 UINT、	予約 : (無視する) オフセット OT : メモリ・マップにおけるデータのオフセット  予約 : (無視する) オフセット TO : メモリ・マップにおけるデータのオフセット  予約 : (無視する)
10	Set	Config #2 Data	構成 :	-
		Config_data_size	UINT	Config_data のバイト単位の長さ
		Config_data	オクテットの配列	コンフィグレーション #2 データ

#	アクセス	名前	タイプ	コメント
11	Set	Proxy Device ID	構成 :	-
		Vendor_id	UINT	Vendor ID
		Product_type	UINT	Device Type
		Product_code	UINT	Product Code
		Major_rev	USINT	メジャー・リビジョン
		Minor_rev	USINT	マイナー・リビジョン

## ポート・オブジェクト、クラス F4h

### 概要

#### オブジェクトの説明

-

#### サポートされるサービス

クラス・サービス : Get Attribute All  
Get Attribute Single

インスタンス・サービス : Get Attribute All  
Get Attribute Single

### クラスのアトリビュート

#	アクセス	名前	タイプ	値	説明
1	Get	Revision	UINT	0001h	-
2	Get	Max Instance	UINT	0002h	-
3	Get	No. of instances	UINT	0001h	-
8	Get	Entry Port	UINT	0002h	インスタンス #2 (下記を参照)
9	Get	All Ports	構成の配列	0000h 0000h 0000h 0000h 0004h 0002h	各インスタンスのアトリビュート1および2を含む構成の配列。インスタンス1はバイト・オフセット4、インスタンス2はバイト・オフセット8の位置にあります。オフセット0での4バイトは0になります。

### インスタンスのアトリビュート、インスタンス 02h

#	アクセス	名前	タイプ	値	説明
1	Get	Port Type	UINT	0004h	TCP/IP
2	Get	Port Number	UINT	0002h	ポート 2
3	Get	Port Object	構成 :		-
		Path Size	UINT	0002h	パス・サイズ
		Path	埋め込み EPATH	20 F5 24 01h	TCP インターフェース・オブジェクト (クラス F5h、インスタンス 1)
4	Get	Port Name	SHORT_STIRNG	'TCP/IP'	ポートの名前
8	Get	Node Address	埋め込み EPATH	-	TCP/IP アドレスを表す EPATH

## TCP/IP インターフェース・オブジェクト、クラス F5h

### 概要

#### オブジェクトの説明

このオブジェクトは、EtherNet/IP 経由で TCP/IP 設定を構成するためのメカニズムを提供します。このオブジェクトへのライトはコンフィグレーション・ファイル 'ip.cfg' に保存されている設定に影響するので注意してください。

#### サポートされるサービス

クラス・サービス :           Get Attribute All  
                                  Get Attribute Single

インスタンス・サービス :    Get Attribute All  
                                  Get Attribute Single  
                                  Set Attribute Single

#### クラスのアトリビュート

#	アクセス	名前	タイプ	値	説明
1	Get	Revision	UINT	0001h	リビジョン 1
2	Get	Max Instance	UINT	0001h	1 は最も高いインスタンス番号
3	Get	No. of instances	UINT	0001h	1 つのインスタンスが実装される

#### インスタンスのアトリビュート

#	アクセス	名前	タイプ	値	説明
1	Get	Status	DWORD	0000 0001h	1 = インターフェース・コンフィグレーションのアトリビュートには有効なコンフィグレーションが含まれている
2	Get	Configuration Capability	DWORD	0000 0014h	インターフェース・コンフィグレーションのアトリビュートを設定可能。DHCP 経由でネットワーク・コンフィグレーションを取得可能。
3	Get/Set	Configuration Control	DWORD	-	0 - 不揮発性メモリからのコンフィグレーション 2 - DHCP からのコンフィグレーション
4	Get	Port Object	構成 :		物理リンク -> イーサネット・オブジェクト
		Path Size	UINT	0002h	2 ワード
		Path	埋め込み EPATH	20 F6 24 01h	イーサネット・クラス、インスタンス 1
5	Get/Set	Interface Configuration	構成 :		-
		IP Address	UDINT	-	現在使用されている IP アドレス
		Subnet Mask	UDINT	-	現在使用されているサブネット・マスク
		Gateway address	UDINT	-	現在使用されているゲートウェイ・アドレス
		Name Server 1	UDINT	-	プライマリ DNS サーバ
		Name Server 2	UDINT	-	セカンダリ DNS サーバ
		Domain Name	STRING	-	デフォルト・ドメイン名
6	Get/Set	Host Name	STRING	-	ホスト名

## イーサネット・リンク・オブジェクト、クラス F6h

### 概要

#### オブジェクトの説明

このオブジェクトは、イーサネット通信インターフェースに対するリンク特有のカウントとステータス情報を保持します。

#### サポートされるサービス

クラス・サービス : Get Attribute All  
Get Attribute Single

インスタンス・サービス : Get Attribute All  
Get Attribute Single  
Set Attribute Single

### クラスのアトリビュート

#	アクセス	名前	タイプ	値	説明
1	Get	Revision	UINT	0001h	リビジョン 1
2	Get	Max Instance	UINT	0001h	1 は最も高いインスタンス番号
3	Get	No. of instances	UINT	0001h	1 つのインスタンスが実装される

### インスタンスのアトリビュート

#	アクセス	名前	タイプ	値	説明
1	Get	Interface Speed	UDINT	10 または 100	実際の速度 (MBPS)
2	Get	Interface Flags	DWORD	-	インターフェース・フラグ
3	Get	Physical Address	6 USINT の配列	MAC アドレス	イーサネット MAC アドレス
4	Get	Interface Counters	構成 :	-	
		In Octets	UDINT	-	インターフェース上で受信されたオクテット
		In Ucast Packets	UDINT	-	インターフェース上で受信されたユニキャスト・パケット
		In NUcast Packets	UDINT	-	インターフェース上で受信された非ユニキャスト・パケット
		In Discards	UDINT	-	不明なプロトコルを使用したインバウンド・パケット
		In Errors	UDINT	-	エラーが含まれているインバウンド・パケット (In Discards は含まれない)
		In Unknown Protos	UDINT	-	不明なプロトコルを使用したインバウンド・パケット
		Out Octets	UDINT	-	インターフェース上で送信されたオクテット
		Out Ucast Packets	UDINT	-	インターフェース上で送信された非ユニキャスト・パケット
		Out Discards	UDINT	-	不明なプロトコルを使用したアウトバウンド・パケット
		Out Errors	UDINT	-	エラーが含まれているアウトバウンド・パケット (Out Discards は含まれない)

#	アクセス	名前	タイプ	値	説明
5	Get	Media Counters	構成 :		-
		Alignment Errors	UDINT	-	長さが整数のオクテットではない受信フレーム
		FCS Errors	UDINT	-	FCS チェックを通過しない受信フレーム
		Single Collisions	UDINT	-	1つの衝突のみ発生した正常送信済みフレーム
		Multiple Collisions	UDINT	-	複数の衝突が発生した正常送信済みフレーム
		SQE Test Errors	UDINT	-	SQRE テスト・エラー・メッセージがプロデュースされる回数
		Deferred Transmissions	UDINT	-	媒体がビジー状態であるために最初の送信試行が遅延されるフレーム
		Late Collisions	UDINT	-	パケット送信への 512 ビット時間よりも後に衝突が検出される回数
		Excessive Collisions	UDINT	-	過剰な衝突が原因で送信が失敗するフレーム
		MAC Transmit Errors	UDINT	-	内部 MAC サブレイヤ受信エラーが原因で送信が失敗するフレーム
		Carrier Sense Errors	UDINT	-	フレームの送信試行時にキャリア検知条件が失われるか有効状態にならなかった回数
		Frame Too Long	UDINT	-	最大許容フレーム・サイズを超えている受信フレーム
		MAC Receive Errors	UDINT	-	内部 MAC サブレイヤ受信エラーが原因でインターフェース上での受信に失敗するフレーム
6	Get/Set	Interface Control	構成 :		-
		Control Bits	UDINT	-	インターフェース制御ビット
		Force Interface Speed	UDINT	-	インターフェースに対して強制される動作速度。オートネゴシエーションが有効な場合は、'Object state Conflict' を返す。

# FTP サーバ

## 概要

組み込み FTP サーバでは、標準 FTP クライアントを使用してファイルシステムにファイルをアップロードまたはダウンロードすることができます。このサーバは、以下のポート番号を使用します。

- **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

### セキュリティ・レベル

サーバは以下の 2 つのセキュリティ・レベルを備えています。セキュリティ・レベルは、ユーザごとに設定されます。

- **通常ユーザ**  
ルート・ディレクトリは '\cfg' になります。  
**注意:** 有効なユーザ・アカウントが見つからない場合は、全てのユーザが管理ユーザ（下記を参照）として扱われます。
- **管理ユーザ**  
ルート・ディレクトリは '\' になります。つまり、ユーザは制限なしにファイルシステムにアクセスできます。  
**注意:** 工場出荷時のデフォルト管理ログインは 'ABX' (ユーザ名) および 'FTPAccess' (パスワード) です。

### ユーザ・アカウント

ユーザ・アカウントは、以下の 2 つのファイルに保存されます。これらのファイルはウェブ・アクセスから保護されます。

- **'\cfg\pswd\sys\_pswd.cfg'**  
このファイルは、通常ユーザのアカウント詳細を保持します。
- **'\pswd\ad\_pswd.cfg'**  
このファイルは、管理ユーザのアカウント詳細を保持します。

#### ファイル・フォーマット:

ユーザ・アカウントは、以下のフォーマットで保存されます。

```
Username1:Password1  
Username2:Password2  
Username3:Password3
```

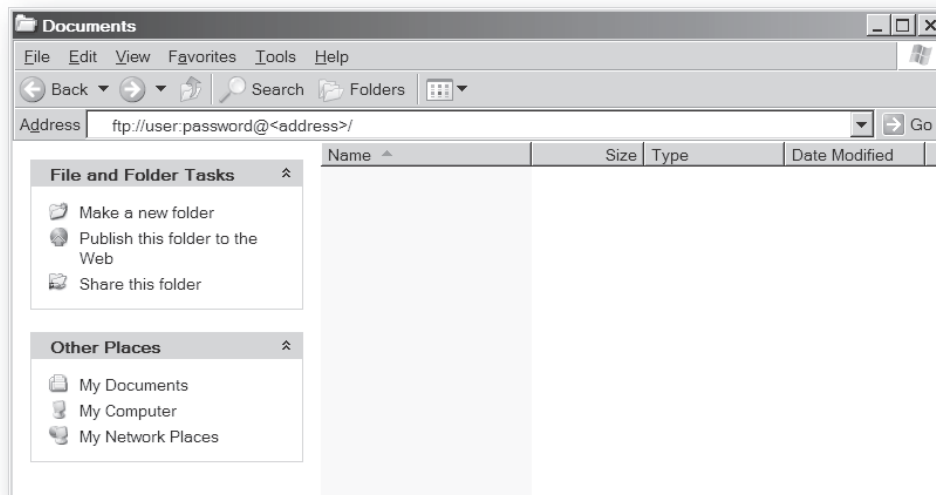


## Windows エクスプローラ™ による FTP サーバへのアクセス

Windows エクスプローラの FTP クライアントでは、以下のように簡単にファイルシステム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 エクスプローラを開きます。
2. アドレス・フィールドで、FTP://<user>:<password>@<address> と入力します。
  - <address> はスキャナ・インターフェースの IP アドレスに置き換えます。
  - <user> はユーザ名に置き換えます。
  - <password> はパスワードに置き換えます。
3. Enter を押します。

エクスプローラは、指定された設定でスキャナ・インターフェースへの接続を試みます。成功した場合は、ファイルシステムがエクスプローラ・ウィンドウに表示されます。



## 電子メール・クライアント

### 概要

組み込みの電子メール・クライアントは、あらかじめ定義された電子メール・メッセージを入力および出力バッファ内のトリガ・イベントに基づいて送信できます。

操作では、これは以下のように機能します。

1. トリガ・ソースが入力または出力バッファから取り出されます。
2. トリガ・ソースとマスク値の間で論理 AND が実行されます。
3. その結果は、指定されたオペランドに従って基準値と比較されます。
4. 最終結果が true の場合、電子メールは指定された受信者へ送信されます。

スキャナ・インターフェースは電子メール・トリガ・イベントを 0.5 秒ごとに 1 回処理することに注意してください。つまり、トリガ・イベントを正しく検出するためには、トリガ・イベントが 0.5 秒以上存在している必要があります。

特定のメッセージを送信させるイベントは、メッセージごとに別々に指定されます。詳細については、8-2 “電子メール定義” を参照してください。クライアントは SSI をサポートしていますが、電子メール・メッセージ（SSI 機能ごとに別々に指定される）では一部の SSI 機能を使用できないことに注意してください。

下記も参照してください。

- 8-1 “SMTP サーバ設定”
- 8-2 “電子メール定義”
- A-1 “SSI (Server Side Include)”

### SMTP サーバ設定

クライアントで電子メール・メッセージを送信できるようにするには、有効な SMTP サーバ・コンフィグレーションが必要です。これらの設定は、システム・ファイル '\cfg\smtp.cfg' に保存されます。

#### ファイル・フォーマット:

```
[SMTP address]
xxx.xxx.xxx.xxx
```

送信電子メール・サーバ・アドレス

```
[SMTP username]
user
```

SMTP サーバ・ログイン。オプション。

```
[SMTP password]
password
```

## 電子メール定義

電子メール定義は、以下のディレクトリに保存されます。

- '\cfg\email'  
このディレクトリは、通常レベルの FTP ユーザが変更できる最大 10 個のメッセージを保持します。
- '\email'  
このディレクトリは、管理レベルの FTP ユーザが変更できる最大 10 個のメッセージを保持します。

電子メール定義ファイルがクライアントによって正しく認識されるようにするには、それらのファイルに 'email\_1.cfg', 'email\_2.cfg' ... 'email\_10.cfg' という名前を付ける必要があります。

### ファイル・フォーマット :

```
[Register]
Area, Offset, Type

[Register match]
Value, Mask, Operand

[To]
recipient

[From]
sender

[Subject]
subject line

[Headers]
Optional extra headers

[Message]
message body
```

キー	値	SSI 用スキャン	
Area	ソース・バッファ。使用可能な値は、'IN' (入力バッファ) または 'OUT' (出力バッファ) です。	非対応	
Offset	10 進法または 16 進法でライトされたソース・オフセット		
Type	ソース・データ・タイプ。使用可能な値は、'byte'、'word'、'long' です。		
Value	比較の基準値として使用されます。		
Mask	比較の前にトリガ・ソースに適用されるマスク値 (論理 AND)		
Operand	使用可能な値は、'<'、'='、または '>' です。		
To	電子メールの受信者		対応
From	送信者の電子メール・アドレス		
Subject	電子メールの件名。1 行のみ。		
Headers	オプション。追加のヘッダを提供する場合に使用できます。		
Message	実際のメッセージ		

**注意 :** 16 進値がクライアントによって認識されるようにするには、接頭辞 '0x' を付けて 16 進値をライトする必要があります。

# SSI (Server Side Include)

## 概要

Server Side Include (以降、SSI と呼ぶ) 機能は、ウェブ・ページと電子メール・メッセージにおける動的コンテンツを可能にします。

SSI は、ソース・ドキュメント内に埋め込まれた特殊なコマンドです。スキャナ・インターフェースがそのようなコマンドに遭遇すると、コマンドが実行され、該当する場合は出力文字列に置き換えられます。

### 構文

以下の 'X' は、コマンドに関連付けられているコマンド演算コードとパラメータを表します。

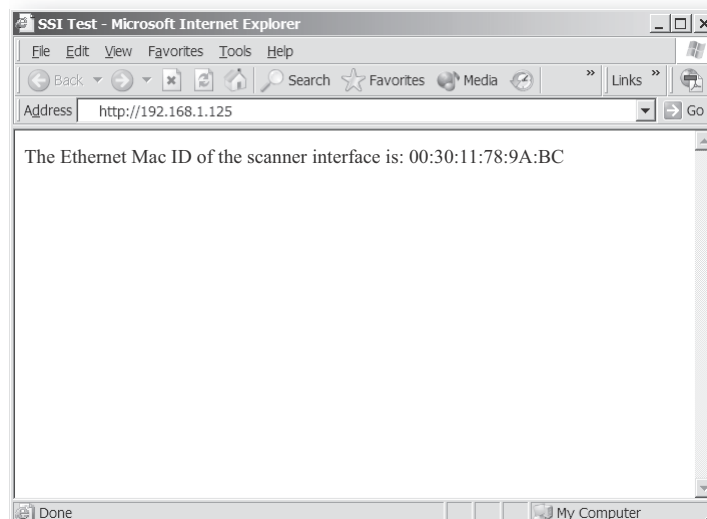
```
<?--#exec cmd_argument='XXXXXXXXXXXXXXXXXXXXXXXXX' -->
```

### 例

以下の例は、インターフェースのイーサネット Mac ID をウェブ・ページに表示します。

```
<HTML>
<HEAD><TITLE>SSI Test</TITLE></HEAD>
<BODY>
The Ethernet Mac ID of the scanner interface is:
<?--#exec cmd_argument='DisplayMacID' -->
</BODY>
</HTML>
```

### 結果のウェブ・ページ



## 関数

### DisplayMacID

この関数は、xx:xx:xx:xx:xx:xx というフォーマットで MAC ID を返します。

**構文:**

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

### DisplaySerial

この関数は、スキャナ・インターフェースのシリアル・ナンバーを返します。

**構文:**

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

### DisplayFWVersion

この関数は、スキャナ・インターフェースのメイン・ファームウェア・リビジョンを返します。

**構文:**

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

### DisplayBLVersion

この関数は、スキャナ・インターフェースのブートローダ・ファームウェア・リビジョンを返します。

**構文:**

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

### DisplayIP

この関数は、現在の IP アドレスを返します。

**構文:**

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

### DisplaySubnet

この関数は、現在のサブネット・マスクを返します。

**構文:**

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

## DisplayGateway

この関数は、現在のゲートウェイ・アドレスを返します。

**構文:**

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

## DisplayDhcpState

この関数は、DHCP が有効か無効かを返します。

**構文:**

```
<?--#exec cmd_argument='DisplayDhcpState( "Output when ON", "Output when OFF" )'-->
```

## StoreIPConfig

**注意:** この関数は、電子メール・メッセージでは使用できません。

この関数は、渡された IP コンフィグレーションをコンフィグレーション・ファイル 'IP.cfg' に保存します。

**構文:**

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

この行を HTML ページに取り込み、新しい IP 設定を備えたフォームをそのページに渡します。

**フォーム内の許容フィールド:**

```
SetIp
SetSubnet
SetGateway
SetDhcpState - value "on" or "off"
SetDNS1
SetDNS2
SetHostName
SetDomainName
```

**デフォルト出力:**

```
Invalid IP address!
Invalid Subnet mask!
Invalid Gateway address!
Invalid IP address or Subnet mask!
Invalid DHCP state!
Invalid DNS1!
Invalid DNS2!
Configuration stored correctly.
Failed to store configuration.
```

## GetText

**注意：**この関数は、電子メール・メッセージでは使用できません。

この関数は、オブジェクトからテキストを取得し、それを OUT 領域に保存します。

### 構文：

```
<?--#exec cmd argument='GetText( "ObjName", OutWriteString ( offset ), n )'-->
```

ObjName - オブジェクトの名前。  
offset - OUT 領域の先頭からのオフセットを指定します。  
n - リードする最大文字数を指定します (オプション)。

### デフォルト出力：

Success - Write succeeded  
Failure - Write failed

## printf

この関数は、フォーマットされた文字列をウェブ・ページ上で取り込みます。この文字列には Anybus IN/OUT 領域からのデータが含まれる場合があります。文字列のフォーマットは、標準 C 関数の printf() と同じです。

### 構文：

```
<?--#exec cmd_argument='printf("String to write", Arg1, Arg2,..., ArgN)'-->
```

標準 C 関数の printf() と同様に、この SSI 関数の "String to write" には 2 つのタイプのオブジェクトが含まれます。1 つは通常の文字です。これらの文字は出力ストリームにコピーされます。もう 1 つは変換指定です。これらの各変換指定によって printf への次の引数の変換とプリントが行われます。各変換指定は、% 文字で始まり、変換文字で終わります。% と変換文字の間には、以下の項目を順番に指定できます。

- 仕様を変更するフラグ (順不同)
  - 変換した引数をフィールド内で左揃えにすることを指定します。
  - + 数値を常に符号付きでプリントすることを指定します。
  - (空白文字) 最初の文字が符号でない場合は、空白文字が前に付きます。
  - 0 数値変換の場合、先行するゼロでフィールドを埋めることを指定します。
  - # 代替出力フォームを指定します。o の場合は、最初の桁がゼロになります。x または X の場合は、0x または 0X がゼロ以外の結果の前に付きます。e、E、f、g、および G の場合は、出力に常に小数点が付きます。g および G の場合は、後続のゼロは削除されません。
- 最小フィールド幅を指定する数値。変換された引数は、少なくともこの幅 (必要に応じて、さらに広い幅) でフィールドにプリントされます。変換された引数の文字数がフィールド幅より少ない場合は、フィールド幅を構成するために引数の左側 (左揃えが要求されている場合は右側) が埋められます。埋め込み文字は通常は空白文字ですが、ゼロ埋め込みフラグがある場合は 0 にすることができます。
- フィールド幅を精度から分離するピリオド。

- 文字列からプリントされる最大文字数、小数点の後にプリントされる桁数 (e、E、または F 変換の場合)、有効桁数 (g または G 変換の場合)、または整数の場合にプリントされる最小桁数 (必要な幅を構成するために先行 0 が追加される) を指定する数値 (精度)。
- 長さ変更子 h、l、または L。"h" は、対応する引数が short または unsigned short としてプリントされることを意味します。"l" は、引数が long または unsigned long であることを意味します。

変換文字とその意味を以下に示します。% の後の文字が変換文字でない場合、挙動は不定です。

文字	引数タイプ、変換後の引数
d、i	byte、short。10 進表記 (符号付き表現の場合。符号付き引数を使用)。
o	byte、short。8 進表記 (先行するゼロなし)。
x、X	byte、short。abcdef (0x の場合) または ABCDEF (0X の場合) を使用した 16 進表記 (先行する 0x または 0X なし)。
u	byte、short。10 進表記。
c	byte、short。unsigned char への変換後は単一文字。
s	char*。"0" に達するまで、または精度によって指定された文字数がプリントされるまで、文字列からの文字がプリントされます。
f	float。[-]mmm.ddd 形式の 10 進表記 (d の数は精度によって指定される)。デフォルトの精度は 6 です。精度 0 にすると、小数点が抑制されます。
e、E	float。[-]m.ddddd e+xx または [-]m.dddddE+xx 形式の 10 進表記 (d の数は精度によって指定される)。デフォルトの精度は 6 です。精度 0 にすると、小数点が抑制されます。
g、G	float。指数が -4 より小さいか精度以上の場合、%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。後続のゼロと小数点はプリントされません。
%	引数は変換されません。% をプリントします。



SSI 関数の *printf* に渡すことができる引数は以下のとおりです。

引数	説明
InReadSByte( <i>offset</i> )	入力バッファ内の位置 <i>offset</i> から符号付きバイトをリードします。
InReadUByte( <i>offset</i> )	入力バッファ内の位置 <i>offset</i> から符号なしバイトをリードします。
InReadSWord( <i>offset</i> )	入力バッファ内の位置 <i>offset</i> から符号付きワード (short) をリードします。
InReadUWord( <i>offset</i> )	入力バッファ内の位置 <i>offset</i> から符号なしワード (short) をリードします。
InReadSLong( <i>offset</i> )	入力バッファ内の位置 <i>offset</i> から符号付きロングワード (long) をリードします。
InReadULong( <i>offset</i> )	入力バッファ内の位置 <i>offset</i> から符号なしロングワード (long) をリードします。
InReadString( <i>offset</i> )	入力バッファ内の位置 <i>offset</i> から文字列 (char*) をリードします。
InReadFloat( <i>offset</i> )	入力バッファ内の位置 <i>offset</i> から浮動小数点 (float) 値をリードします。
OutReadSByte( <i>offset</i> )	出力バッファ内の位置 <i>offset</i> から符号付きバイトをリードします。
OutReadUByte( <i>offset</i> )	出力バッファ内の位置 <i>offset</i> から符号なしバイトをリードします。
OutReadSWord( <i>offset</i> )	出力バッファ内の位置 <i>offset</i> から符号付きワード (short) をリードします。
OutReadUWord( <i>offset</i> )	出力バッファ内の位置 <i>offset</i> から符号なしワード (short) をリードします。
OutReadSLong( <i>offset</i> )	出力バッファ内の位置 <i>offset</i> から符号付きロングワード (long) をリードします。
OutReadULong( <i>offset</i> )	出力バッファ内の位置 <i>offset</i> から符号なしロングワード (long) をリードします。
OutReadString( <i>offset</i> )	出力バッファ内の位置 <i>offset</i> から NULL 終端文字列 (char*) をリードします。
OutReadFloat( <i>offset</i> )	出力バッファ内の位置 <i>offset</i> から浮動小数点 (float) 値をリードします。
MbReadSWord( <i>id</i> )	ゲートウェイを制御する場合に使用します。A-11 “ゲートウェイ制御” を参照してください。
CipReadSByte( <i>class, inst, attr</i> )	CIP オブジェクトから符号付きバイトをリードします。
CipReadUByte( <i>class, inst, attr</i> )	CIP オブジェクトから符号なしバイトをリードします。
CipReadSWord( <i>class, inst, attr</i> )	CIP オブジェクトから符号付きワードをリードします。
CipReadUWord( <i>class, inst, attr</i> )	CIP オブジェクトから符号なしワードをリードします。
CipReadSLong( <i>class, inst, attr</i> )	CIP オブジェクトから符号付きロングワードをリードします。
CipReadULong( <i>class, inst, attr</i> )	CIP オブジェクトから符号なしロングワードをリードします。
CipReadFloat( <i>class, inst, attr</i> )	CIP オブジェクトから浮動小数点値をリードします。
CipReadShortString( <i>class, inst, attr</i> )	CIP オブジェクトから short 文字列をリードします。
CipReadString( <i>class, inst, attr</i> )	CIP オブジェクトから NULL 終端文字列をリードします。
CipReadUByteArray( <i>class, inst, attr</i> )	CIP オブジェクトから符号なしバイト配列をリードします。
CipReadUWordArray( <i>class, inst, attr</i> )	CIP オブジェクトから符号なしワード配列をリードします。
CipReadULongArray( <i>class, inst, attr</i> )	CIP オブジェクトから符号なしロングワード配列をリードします。

## scanf

**注意：**この関数は、電子メール・メッセージでは使用できません。

この関数は、HTML フォーム内のオブジェクトから渡された文字列をリードし、フォーマットの仕様に従って文字列を解釈し、渡された引数に従って結果を OUT 領域に保存します。文字列のフォーマットは、標準 C 関数呼び出しの scanf() と同じです。

### 構文：

```
<?--#exec cmd_argument='scanf( "ObjName", "format", Arg1, ..., ArgN), ErrVal1, ..., ErrvalN'-->
```

ObjName - 渡されたデータ文字列を持つオブジェクトの名前。  
 format - 渡された文字列のフォーマット方法を指定します。  
 Arg1 - ArgN - どこにデータをライトするのかを指定します。  
 ErrVal1 - ErrValN - エラーが発生した場合にライトする値 / 文字列を指定します (オプション)。

文字	入力、引数タイプ
d	10 進数。byte、short。
i	数値。byte、short。数値は、8 進数 (先行 0 (ゼロ)) または 16 進数 (先行 0x または 0X) にすることができます。
o	8 進数 (先行ゼロあり / なし)。byte、short。
u	符号なし 10 進数。unsigned byte、unsigned short。
x	16 進数 (先行 0x または 0X あり / なし)。byte、short。
c	文字。char*。次の入力文字 (デフォルト 1) は、指定された地点に配置されます。通常のスキップ・オーバー空白は抑制されます。次の非空白文字をリードするには、%1s を使用します。
s	文字列 (引用符なし)。char* (追加される文字列と終端 "\0" に対して十分に大きい文字の配列をポイントする)。
e、f、g	オプションの符号、オプションの小数点、およびオプションの指数を使用した浮動小数点数。float*。
%	リテラル %。代入は行われません。

'byte' または 'short' ではなく 'long' へのポインタが引数リストに含まれていることを示すために、変換文字 d、i、o、u、x の前に l が付く場合があります。

SSI 関数の scanf に渡すことができる引数は以下のとおりです。

引数	説明
OutWriteByte(offset)	出力バッファ内の位置 offset にバイトをライトします。
OutWriteWord(offset)	出力バッファ内の位置 offset にワード (short) をライトします。
OutWriteLong(offset)	出力バッファ内の位置 offset に long をライトします。
OutWriteString(offset)	出力バッファ内の位置 offset に文字列をライトします。
OutWriteFloat(offset)	出力バッファ内の位置 offset に浮動小数点 (float) 値をライトします。
CipWriteByte(class, inst, attr)	CIP オブジェクトにバイト値をライトします。
CipWriteWord(class, inst, attr)	CIP オブジェクトにワード値をライトします。
CipWriteLong(class, inst, attr)	CIP オブジェクトにロングワードをライトします。
CipWriteFloat(class, inst, attr)	CIP オブジェクトに浮動小数点値をライトします。

### デフォルト出力：

```
Write succeeded  
Write failed
```

## IncludeFile

この関数は、ウェブ・ページ上でファイルの内容を取り込みます。

### 構文:

```
<?--#exec cmd_argument='IncludeFile( "File name" )'-->
```

### デフォルト出力:

```
Success      - <File content>
Failure      - Failed to open <filename>
```

## SaveToFile

**注意:** この関数は、電子メール・メッセージでは使用できません。

この関数は、渡されたフォームの内容をファイルに保存します。渡された名前 / 値ペアは、"Separator" 文字列で区切られてファイル "File name" にライトされます。[Append|Overwrite] パラメータでは、指定されたファイルを上書きするかどうか、またはファイル内のデータを追加するかどうかが決まります。

### 構文:

```
<?--#exec cmd_argument='SaveToFile( "File name", "Separator", [Append|Overwrite] )'-->
```

### デフォルト出力:

```
Success      - Form saved to file
Failure      - Failed to save form
```

## SaveDataToFile

**注意:** この関数は、電子メール・メッセージでは使用できません。

この SSI 関数は、渡されたフォームのデータをファイルに保存します。"Object name" パラメータはオプションです。このオプションを指定した場合は、そのオブジェクトからのデータのみ保存されます。指定しなかった場合は、フォーム内の全てのオブジェクトからのデータが保存されます。

[Append|Overwrite] パラメータでは、指定されたファイルを上書きするかどうか、またはファイル内のデータを追加するかどうかが決まります。

### 構文:

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

### デフォルト出力:

```
Success      - Form saved to file
Failure      - Failed to save form
```

## DisplayScannerMode

この関数は、現在のスキャナ・モード（実行またはアイドル状態）を返します。

**構文：**

```
<?--#exec cmd_argument='DisplayScannerMode  
( "Output when Run", "Output when Idle" )'-->
```

## SetScannerMode

**注意：**この関数は、電子メール・メッセージでは使用できません。

EtherNet/IP スキャナを Run または Idle に設定するには、この関数を使用します。  
'scanner\_state' という変数は、値 'run' または 'idle' (他の値は無視される) と共にページに送信されます。

**構文：**

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

**デフォルト出力：**

```
Failure          - Change scanner mode not possible
```

下記も参照してください。

- 6-2 “ID オブジェクト、クラス 01h”

## SSI 出力の変更

SSI 関数からの出力文字列を変更する方法は 2 つあります。

1. システム内の全ての SSI 関数に対する出力文字列が含まれている "\ssi\_str.cfg" というファイルを作成することで、デフォルトの SSI 出力を変更する
2. SSI 関数の "SsiOutput()" を呼び出すことで、一時的に SSI 出力を変更する

### SSI 出力文字列ファイル

ファイル "\ssi\_str.cfg" がファイルシステム内で見つかり、そのファイルが下記の仕様に正しく従っている場合、SSI 関数はデフォルト文字列ではなくこのファイルで指定された出力文字列を使用します。

ファイルのフォーマットは以下のようになります。

```
[StoreEtnConfig]
Success:"String to use on success"
Invalid IP:"String to use when the IP address is invalid"
Invalid Subnet:"String to use when the Subnet mask is invalid"
Invalid Gateway:"String to use when the Gateway address is invalid"
Invalid IP or Subnet:"String to use when the IP address and Subnet mask does
not match"
Invalid DNS1:"String to use when the primary DNS cannot be found"
Invalid DNS2:"String to use when the secondary DNS cannot be found"
Save Error:"String to use when storage fails"
Invalid DHCP state:"String to use when the DHCP state is invalid"

[scanf]
Success:"String to use on success"
Failure:"String to use on failure"

[IncludeFile]
Failure:"String to use when failure"1

[SaveToFile]
Success:"String to use on success"
Failure:"String to use on failure"1

[SaveDataToFile]
Success:"String to use on success"
Failure:"String to use on failure"1

[GetText]
Success:"String to use on success"
Failure:"String to use on failure"
```

'[File path]' を先頭行に配置し、ファイル・パスを 2 行目に配置すると、このファイルの内容をリダイレクトすることができます。

**例:**

```
[File path]
\user\ssi_strings.cfg
```

この例では、上記の設定がファイル 'user\ssi\_strings.cfg' からロードされます。

1. '%s' はファイル名を文字列に含めます。

## 一時的な SSI 出力変更

次に呼び出される SSI 関数の SSI 出力は、SSI 関数 "SsiOutput()" を使用して一時的に変更できます。これは次の SSI 関数の出力文字列にのみ影響することに注意してください。その後、出力文字列はデフォルトに戻るか、またはファイル '\ssi\_str.cfg' で定義された文字列に戻ります。

文字列の最大サイズは 128 バイトです。

### 構文:

```
<?--#exec cmd_argument='SsiOutput( "Success string", "Failure string" )'-->
```

### 例:

この例では、scanf SSI 呼び出しの出力文字列を変更する方法を示します。

```
<?--#exec cmd_argument='SsiOutput ( "Parameter1 updated", "Error" )'-->
<?--#exec cmd_argument="scanf( "Parameter1", "%d", OutWriteByte(0) )'-->
```

## ゲートウェイ制御

### 動的ゲートウェイ・ステータス情報の更新

システム・ファイルの 'dynamic.txt' には、ゲートウェイとオンボード・ネットワーク・インターフェースからの動的ステータス情報が保持されます。最新の情報を提供するには、このファイルを更新してから使用する必要があります。

以下の SSI コマンド・シーケンスは、ファイルの更新をゲートウェイに指示します。

### 構文:

```
<?--#exec cmd_argument='printf( "Data:%u", MbReadSWord( 21 ) )'-->
```

### ゲートウェイの再起動

以下の SSI コマンド・シーケンスを使用してゲートウェイをリセットすることができます。

### 構文:

```
<?--#exec cmd_argument='printf( "Data:%u", MbReadSWord( 1 ) )'-->
```

## 技術仕様

### スキャナ・インターフェースの詳細

- EtherNet/IP スキャナ
- FTP サーバ
- ウェブ・サーバ
- SMTP クライアント
- 10/100Mbit 動作（全二重または半二重）
- オンボードの IP コンフィグレーション・スイッチ
- シールドまたは非シールド・ケーブル
- DHCP 対応
- HICP 対応（HMS の Anybus IPconfig ユーティリティをサポート）
- DNS のサポート

### LAN（イーサネット）コネクタのピンアウト（RJ45）

ピン	信号
1	TD+
2	TD-
3	RD+
4	終端
5	終端
6	RD-
7	終端
8	終端

