

Anybus[®] CompactCom 40 CC-Link

Network Guide

Doc.Id. HMSI-27-330
Rev. 1.00



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、Anybus CompactCom 40 CC-Link 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 40 CC-Link の機能についてのみ説明しています。Anybus CompactCom 40 に関する一般的な情報については、Anybus CompactCom 40 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。通常は、設計を実装するにはこのネットワークガイドの情報だけで十分です。ただし、CC-Link 固有の高度な機能を使用する場合は、CC-Link ネットワーキングの内部詳細、および公式の CC-Link 規格に関する十分な知識が必要となります。本製品の使用者は、CC-Link 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ぬものとします。

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	5
	ドキュメント更新履歴	5
	表記と用語	6
	サポート	6
第 1 章	Anybus CompactCom CC-Link について	
	概要	7
	機能	7
	40 シリーズと 30 シリーズの違い	7
	フィールドバス適合性に関する注意	8
第 2 章	基本動作	
	概要	9
	ソフトウェアの要件	9
	CC-Link ファミリシステムプロファイル (CSP+) ファイル	9
	通信設定	9
	データ交換	10
	ビット領域とワード領域	10
	データサイズ	10
第 3 章	プロセスデータマッピング	
	デフォルトの CC-Link バッファメモリマップ	12
	マッピング	14
第 4 章	CC-Link システム領域の実装	
	システム領域のモード	17
	システム領域のレイアウト	18
	システム領域の位置	19
	診断	20

第 5 章	Anybus モジュールオブジェクト	
	概要	21
	Anybus オブジェクト (01h)	22
	診断オブジェクト (02h)	23
	ネットワークオブジェクト (03h)	25
	ネットワークコンフィグレーションオブジェクト (04h)	27
	ネットワーク CC-Link オブジェクト (08h)	30
	Anybus ファイルシステムインターフェースオブジェクト (0Ah)	32
第 6 章	ホストアプリケーションオブジェクト	
	概要	33
	CC-Link ホストオブジェクト (F7h)	34
Appendix A	機能の分類	
	基本	38
	拡張	38
	高度	38
Appendix B	実装詳細	
	SUP ビットの定義	39
	Anybus ステートマシン	39
	アプリケーションウォッチドッグのタイムアウト処理	39
	CC-Link ハンドシェイクの実装	40
Appendix C	認定情報	
	基本	41
	モデルコード	41
	CC-Link バージョン 2.00	41
Appendix D	例外情報	

Appendix E 技術仕様

前面図 (M40)	43
<i>Run LED</i>	43
<i>Error LED</i>	43
CC-Link コネクタ	43
保護接地 (PE) に関する要件	43
電源	44
環境仕様	44
EMC への準拠	44

Appendix F タイミングと性能

概要	45
内部タイミング	45
<i>起動遅延</i>	45
<i>NW_INIT の処理</i>	45
イベントベースの <i>WrMsg</i> ビジー時間	46
イベントベースのプロセスデータ遅延時間	46

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

関連ドキュメント	ドキュメント作成者
Anybus CompactCom 40 Software Design Guide	HMS
Anybus CompactCom 40 Hardware Design Guide	HMS
Anybus CompactCom 40 Software Driver User Guide	HMS
CC-Link Remote Device Station Conformance Test specification (publication BAP-C0401-012-F)	CLPA
CC-Link Specification (Profile) (publication BAP-05028-H)	CLPA
CC-Link Family System Profile (CSP+) Creation Guideline (publication BCN89000-0756-A)	CLPA

P.2 ドキュメント更新履歴

最近の変更に関する概要 (0.95 ~ 1.00)

変更内容	ページ
コンFORMANCEテストに関するコメントを追加	8
タイミングと性能に関する情報を追加	45
ラッチイベントがサポートされた	17, 20, 23, 40
診断オブジェクトにコマンド詳細を追加	24

改訂一覧

改訂	日付	作成者	章	説明
0.95	2014/12/19	KeL	すべて	リリース準備版
1.00	2015/02/06	KeL	1, 4, 5, F	最初の正式リリース

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。
- " 基本 "、" 拡張 "、" 高度 " を使用してオブジェクト、インスタンス、およびアトリビュートを分類しています。詳細については、38 ページの「機能の分類」を参照してください。

P.4 サポート

お問い合わせとサポートに関する情報は、www.anybus.jp の各ページをご覧ください。

1. Anybus CompactCom CC-Link について

1.1 概要

Anybus CompactCom 40 CC-Link 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ースを通じて、CC-Link スレーブ機能を簡単に実現します。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 40 Hardware / Software Design Guide で定義されたあらゆるホストインターフェースに対応しています。そのため、CC-Link 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、ネットワークソフトウェアを別途用意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

- CC-Link リモートデバイス
- I/O : 最大 128 個のビットポイントと、16 ビットデータの最大 16 個のワードポイント (CC-Link v.1.10、デフォルト)
- I/O : 最大 896 個のビットポイントと、16 ビットデータの最大 128 個のワードポイント (CC-Link v.2.00)¹
- CC-Link システム領域の自動ハンドシェイク²
- HMS Industrial Networks AB によって提供されたコンフィグレーションファイル (CSP+)
- ベンダーコード、モデルコード、およびバージョンをアプリケーションインターフェースを介してカスタマイズ可能
- アプリケーションインターフェースによるボーレートと局番の設定
- ガルバニック絶縁されたバス

このバージョンの Anybus CompactCom 40 CC-Link では、Anybus CompactCom 40 モジュラーデバイスと同期機能はサポートされていないことに注意してください。

1.3 40 シリーズと 30 シリーズの違い

- 新しいデフォルトのプロセスデータ (PD) マッピングスキームが実装されています。
- 特定の CC-Link マッピングコマンドが削除されました。
- 異なる診断機能が実装されています。

-
1. CC-Link v.2.00 を使用するには、CC-Link ホストオブジェクト (F7h) を実装する必要があります。
 2. 自動ハンドシェイク機能を有効にするには、CC-Link ホストオブジェクト (F7h) を実装する必要があります。

1.4 フィールドバス適合性に関する注意

- Anybus CompactCom 40 CC-Link はスタンドアロンでテストされ、CC-Link コンフォーマンステストの仕様書 BAP-C0401-012-F に適合することがわかっています。ただし、最終製品が CC-Link 認定ポリシーに適合するためには、再認定を受ける必要があります。
- CC-Link 仕様書（プロファイル編）BAP-05028-H で定義されたプロファイルとの互換性の維持については、アプリケーションが単独で責任を負います。この仕様書は、登録されたすべての CLPA 会員に無償で提供されます。CLPA に加入する方法については、CLPA にお問い合わせください（www.cc-link.org）。
- コンフォーマンステストに合格するためには、CC-Link に準拠していない誤ったボーレートまたは局番が使用されたときに、アプリケーションがエラーステータスを示す必要があります。Anybus CompactCom 40 CC-Link は、このような場合に Anybus オブジェクトの Setup Complete アトリビュートの設定を許可しません。
- ホストアプリケーションがシステム領域を処理する場合（自動ハンドシェイクを使用しない場合）は、システム領域に関連するコンフォーマンステストのすべての部分についても、ホストアプリケーションが全責任を負う必要があります。
- CC-Link V.2.00 プロトコルは、モジュールでの戻りチェックを行わない方式で実装されています。

2. 基本動作

2.1 概要

2.1.1 ソフトウェアの要件

Anybus CompactCom CC-Link 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、CC-Link ネットワークシステムの実用上、いくつかの制約を考慮する必要があります。

- アサイクリックなデータ交換はありません。
- ADI をネットワーク上で表すためには、CC-Link マッピングスキームに従って ADI をプロセスデータとしてマッピングする必要があります。
- ADI の名前、型、およびそれらと同等のアトリビュートにネットワークからアクセスすることはできません。
- ネットワークリセット要求はサポートされていません。
- 通常動作中にホストアプリケーションが作成できる診断インスタンスは最大 5 つです (5-23 「診断オブジェクト (02h)」を参照)。重大な障害が発生した場合には、6 つ目のインスタンスを追加で作成できます。

Anybus CompactCom のソフトウェアインターフェースに関する詳細は、『Anybus CompactCom 40 Software Design Guide』を参照してください。

2.1.2 CC-Link ファミリシステムプロファイル (CSP+) ファイル

CC-Link 上の各機器は、CC-Link ファミリシステムプロファイル (CSP+) ファイルと関連付けられています。このファイルには、機器やその機能の説明が記述されています。

HMS Industrial Networks は標準の CSP+ ファイルを提供しており、このファイルを新たな開発を行うときのベースとして使用できます。このファイルを変更するためのツールを CLPA の Web ページから入手できます (www.cc-link.org、会員登録が必要)。

2.1.3 通信設定

エンドユーザーがアクセスできるネットワーク関連の通信設定は、ネットワークコンフィギュレーションオブジェクト (04h) に集約されています。

CC-Link の場合、これには以下の設定が含まれます。

- **局番**
CC-Link では、ネットワーク上の各機器に一意的局番を割り当てる必要があります。最大の局番は占有局の数によって異なります。
- **ボーレート**
本モジュールは、10Mbps までの一般的な CC-Link ボーレートをすべてサポートします。

2.2 データ交換

2.2.1 ビット領域とワード領域

CC-Link では、データは以下のように 2 つのカテゴリに分割されます。

- **ビット領域**

データはビット単位でアクセスされます。データは一般に RX #nn（スレーブ→マスター）および RY #nn（マスター→スレーブ）と表現されます。ここで、"nn" はビット領域内のアドレス指定可能なポイント（すなわち、単一のビット）を表します。

- **ワード領域**

データは 16 ビットのワードとしてアクセスされます。データは一般に RWr #nn（スレーブ→マスター）および RWw #nn（マスター→スレーブ）と表現されます。ここで、"nn" はワード領域内のアドレス指定可能なポイント（すなわち、ワード）を表します。

方向	ビット領域のポイント (16 進数)	ワード領域のポイント (10 進数)
スレーブ→マスター	RX00 ...RX37F	RWr0...RWr127
マスター→スレーブ	RY00 ...RY37F	RWw0 ...RWw127

2.2.2 データサイズ

CC-Link バージョン 1

デフォルトでは、マッピングされたプロセスデータに基づいて必要な占有局の数が自動的に計算されます。以下のデータサイズが使用できます。

占有局	ビットポイント	ワードポイント	合計（ビット+ワード、バイト単位）
1	32 ビット	4 ワード	12
2	64 ビット	8 ワード	24
3	96 ビット	12 ワード	36
4	128 ビット	16 ワード	48

CC-Link バージョン 2

CC-Link ホストオブジェクト（F7h）の "Network Settings" アトリビュート（#4）を実装することで、CC-Link バージョン 2 の実装をカスタマイズし、拡張サイクルを通じてより大きなデータサイズを使用できます。その場合は、以下のサイズが使用できます。

占有局	1 回の拡張サイクル		2 回の拡張サイクル		4 回の拡張サイクル		8 回の拡張サイクル	
	ポイント	合計	ポイント	合計	ポイント	合計	ポイント	合計
1	32 ビット	12 バイト	32 ビット	20 バイト	64 ビット	40 バイト	128 ビット	80 バイト
	4 ワード		8 ワード		16 ワード		32 ワード	
2	64 ビット	24 バイト	96 ビット	44 バイト	192 ビット	88 バイト	384 ビット	176 バイト
	8 ワード		16 ワード		32 ワード		64 ワード	
3	96 ビット	36 バイト	160 ビット	68 バイト	320 ビット	136 バイト	640 ビット	272 バイト
	12 ワード		24 ワード		48 ワード		96 ワード	

占有局	1回の拡張サイクル		2回の拡張サイクル		4回の拡張サイクル		8回の拡張サイクル	
	ポイント	合計	ポイント	合計	ポイント	合計	ポイント	合計
4	128 ビット	48 バイト	224 ビット	92 バイト	448 ビット	184 バイト	896 ビット	368 バイト
	16 ワード		32 ワード		64 ワード		128 ワード	

注 1 : CC-Link では、ビットデータおよびワードデータの特定の部分が予約されています。予約された部分をデータ交換に使用しないでください。詳細については、17 の「CC-Link システム領域の実装」を参照。

注 2 : CC-Link バージョン 2 のコンFORMANCEテストでは、CC-Link コンFORMANCEテストモードを有効にする必要があります。30 の「ネットワーク CC-Link オブジェクト (08h)」の CCL_Conf_Test_Mode コマンド (31 ページ) を参照してください。

3. プロセスデータマッピング

3.1 デフォルトの CC-Link バッファメモリマップ

Anybus CompactCom 40 CC-Link のデフォルトプロファイルは "Generic Device" です。実装の現実化（対応するメモリへの ADI のマッピングなど）はアプリケーションに委ねられます。これにより、どのようなプロファイルバッファデータのレイアウトにも対応できます。

デフォルトバッファメモリ（ビットポイント）

次の表に、デフォルトバッファメモリのビット領域のレイアウトを示します。これらのアドレスは、ネットワーク全体の絶対アドレスではなく、ユニット相対アドレスです。

スレーブ → マスター		マスター → スレーブ	
ポイント	信号名	ポイント	信号名
RX Q-11h	ユーザー領域	RY0 to RX Q-11h	ユーザー領域
RX Q-10h	予約	RY Q-10h	予約
RX Q-Fh		RY Q-Fh	
RX Q-Eh		RY Q-Eh	
RX Q-Dh		RY Q-Dh	
RX Q-Ch		RY Q-Ch	
RX Q-Bh		RY Q-Bh	
RX Q-Ah		RY Q-Ah	
RX Q-9h		RY Q-9h	
RX Q-8h	イニシャルデータ処理要求フラグ	RY Q-8h	イニシャルデータ処理完了フラグ
RX Q-7h	イニシャルデータ設定完了フラグ	RY Q-7h	イニシャルデータ設定要求フラグ
RX Q-6h	エラー状態フラグ	RY Q-6h	エラーリセット要求フラグ
RX Q-5h	リモート READY	RY Q-5h	予約
RX Q-4h	予約	RY Q-4h	
RX Q-3h		RY Q-3h	
RX Q-2h		RY Q-2h	
RX Q-1h		RY Q-1h	

CC-Link バージョン 1、または拡張サイクル = 1 :

$Q = \text{占有局数} \times 32$

CC-Link バージョン 2、かつ拡張サイクル ≥ 2 :

$Q = (\text{占有局数} \times 32 - 16) \times \text{拡張サイクル}$

デフォルトバッファメモリ（ワードポイント）

次の表に、デフォルトバッファメモリのワード領域のレイアウトを示します。これらのアドレスは、ネットワーク全体の絶対アドレスではなく、ユニット相対アドレスです。

スレーブ → マスター		マスター → スレーブ	
ポイント	信号名	ポイント	信号名
RWr 0	ユーザー領域	RWw 0	ユーザー領域
から		から	
RWr Z-1	ユーザー領域	RWw Z-1	ユーザー領域

CC-Link バージョン 1、または拡張サイクル = 1 :

$Z = \text{占有局数} \times 4$

CC-Link バージョン 2、かつ拡張サイクル ≥ 2 :

$Z = \text{占有局数} \times 4 \times \text{拡張サイクル}$

下記も参照してください。

- 17 ページの「CC-Link システム領域の実装」
- 25 ページの「ネットワークオブジェクト (03h)」

3.2 マッピング

ホストアプリケーションがマッピングコマンドを使用するとき、以下のスキームが使用されます。

- Map_ADI_Write_Area と Map_ADI_Write_Ext_Area は、データを RX（ビット）領域と RWr（ワード）領域にマッピングします。
- Map_ADI_Read_Area と Map_ADI_Read_Ext_Area は、データを RY（ビット）領域と RWw（ワード）領域にマッピングします。
- ADI は、マッピングコマンドが発行された順に、それぞれの領域の連続する位置にマッピングされます。
- すべてのビットデータ型（BITSx、BITx、PADx）は、非ビットデータ型を持つマッピングコマンドが以前に受信されていない限り、CC-Link ビット領域にマッピングされます。非ビットデータ型を持つマッピングコマンドが受信された後、それ以降のビットデータ型を持つマッピングコマンドのデータはすべて CC-Link ワード領域にマッピングされます。以降のページの例を参照してください。
- デフォルトでは、パディングは実行されません。プロファイルレイアウトを満たすためにパディングまたはアライメントの要求を行うことは、アプリケーションの役割です。
- Map_ADI_Write_Area コマンドと Map_ADI_Read_Area コマンドはデータの精度がバイト単位であり、長さが 8 ビット、16 ビット、32 ビット、または 64 ビットのデータ型に対してのみ使用できます。
 - このコマンドを使用すると、マッピング中にバイトアライメントが強制されます。
 - 例えばデータ型 BIT3 を使用すると、その要求への応答として NAK（否定応答）メッセージが生成されます。
 - ホストアプリケーションは、例えばワードアライメントが必要な場合などにパディングを実行しなければならないことがあります。

15 ページのサンプルを参照してください。

- Map_ADI_Write_Ext_Area コマンドと Map_ADI_Read_Ext_Area コマンドはデータの精度がビット単位であり、すべてのデータ型に対して使用できます。
 - ビットデータは、BITSx、BITx、PADx のいずれかのデータ型が使用されている限り、ビット領域に連続的にマッピングされます。
 - 他のデータ型を持つコマンドが受信されると、それ以降のデータはすべてワード領域にマッピングされます。これにより、バイトアライメントが損なわれる場合があります。バイトアライメントは必須条件です。データマッピングのアライメントエラーを回避するために必要なすべてのパディングを実行することは、ホストアプリケーションの役割です。
 - ARRAY または STRUCT 型の ADI については、必ずしもそのすべての要素をマッピングする必要はありません。

16 ページのサンプルを参照してください。

- デフォルトモードでは、CC-Link ホストオブジェクト（F7h）にネットワーク設定は実装されておらず、ビットまたはワードが起動コンフィグレーションに適合しない場合は局数が増加します。
- データ型構造体を持つ ADI は、上記のいずれかのルールが適用されない限り、CC-Link ワード領域に完全なチャンク / ブロブとして配置されます。

注：CC-Link プロファイルの実装によっては、そのプロファイルに必要なメモリバッファレイアウトを得るために、マッピング時にパディングを使用するよう求められる場合があります。その場合、CC-Link ビット領域へのマップではパディングに ADI # 0 または PADx を使用し、CC-Link ワード領域ではパディングに ADI # xx（ダミーバイト）を使用します。これらの領域は、（CC-Link ネットワークに対して）バッファメモリ記述で予約済みとしてマークされています。

誤ったマッピングは NAK（否定応答）を生成します。CC-Link ホストオブジェクトの設定が選択されたマッピングに完全に適合していない場合は、例外が生成されます。例外情報はネットワークオブジェクト（03h）に格納されます。詳細については、以下を参照してください。

- 25 ページの「ネットワークオブジェクト（03h）」
- 42 ページの「例外情報」

標準コマンドを使用したマッピングの例

これは Map_ADI_Read_Area コマンドを使用して Read マッピングコマンドを 9 回（A～I）発行した例です。

コマンド	データタイプ	要素の数
A	BITS8	3
B	BITS16	1
C	BOOL	1
D	UINT8	1
E	UINT16	1
F	BOOL	2
G	BITS32	1
H	BITS16	1
I	UINT32	1

ビット領域の RY マッピングの結果は次のとおりです。

ポイント	内容（コマンド番号 [要素 : ビット]							
RY #7...0	A[0:7]	A[0:6]	A[0:5]	A[0:4]	A[0:3]	A[0:2]	A[0:1]	A[0:0]
RY #15...8	A[1:7]	A[1:6]	A[1:5]	A[1:4]	A[1:3]	A[1:2]	A[1:1]	A[1:0]
RY #23...16	A[2:7]	A[2:6]	A[2:5]	A[2:4]	A[2:3]	A[2:2]	A[2:1]	A[2:0]
RY #31...24	B[0:7]	B[0:6]	B[0:5]	B[0:4]	B[0:3]	B[0:2]	B[0:1]	B[0:0]
RY #39...32	B[0:15]	B[0:14]	B[0:13]	B[0:12]	B[0:11]	B[0:10]	B[0:9]	B[0:8]

ワード領域の RWw マッピングの結果は次のとおりです。

	内容（コマンド番号 [ビット / バイト / ワード]															
ポイント	内容（LSB）								内容（MSB）							
RWw #0	C								D							
RWw #1	E[0]								E[1]							
RWw #2	F[0]								F[1]							
RWw #3	G[LSW]								G[LSW]							
RWw #4	G[MSW]								G[MSW]							
RWw #5	H[7]	H[6]	H[5]	H[4]	H[3]	H[2]	H[1]	H[0]	H[15]	H[14]	H[13]	H[12]	H[11]	H[10]	H[9]	H[8]
RWw #6	I[LSW]								I[LSW]							
RWw #7	I[MSW]								I[MSW]							

拡張マッピングコマンドを使用したマッピングの例

これは Map_ADI_Write_Ext_Area コマンドを使用して Write マッピングコマンドを 3 回 (A ~ C) 発行した例です。

コマンド	マッピングアイテムの数	ADI #	要素の合計数	最初のインデックス要素 ^a	連続する要素の数 ^a	型記述子の数	データ型指定子
A	2	3	1	0	1	1	65(BIT1)
	-	10	1	0	1	1	47(PAD15)
B	1	20	5	0	3	5	70(BIT6) 34(PAD2) 9(BITS8) 36(PAD4) 68(BIT4)
C	1	7	4	1	3	4	9(BITS8) 5(UINT16) 10(BITS16) 0(BOOL)

a. 必ずしも ADI のすべての要素をマッピングする必要はないことに注意してください。マッピングする要素は、「最初のインデックス要素」と「連続する要素の数」の設定によって定義されます。

ビット領域の RX マッピングの結果は次のとおりです。

ポイント	コマンド [マッピングアイテム] : ADI [インデックス要素] : (ビット / バイト / ワード)							
RX #7...0	A[1]:10[0]:(b6)	A[1]:10[0]:(b5)	A[1]:10[0]:(b4)	A[1]:10[0]:(b3)	A[1]:10[0]:(b2)	A[1]:10[0]:(b1)	A[1]:10[0]:(b0)	A[0]:3[0]:(b0)
RX #15...8	A[1]:10[0]:(b14)	A[1]:10[0]:(b13)	A[1]:10[0]:(b12)	A[1]:10[0]:(b11)	A[1]:10[0]:(b10)	A[1]:10[0]:(b9)	A[1]:10[0]:(b8)	A[1]:10[0]:(b7)
RX #23...16	B[0]:20[1]:(b1)	B[0]:20[1]:(b0)	B[0]:20[0]:(b5)	B[0]:20[0]:(b4)	B[0]:20[0]:(b3)	B[0]:20[0]:(b2)	B[0]:20[0]:(b1)	B[0]:20[0]:(b0)
RX #31...24	B[0]:20[2]:(b7)	B[0]:20[2]:(b6)	B[0]:20[2]:(b5)	B[0]:20[2]:(b4)	B[0]:20[2]:(b3)	B[0]:20[2]:(b2)	B[0]:20[2]:(b1)	B[0]:20[2]:(b0)

ワード領域の RWr マッピングの結果は次のとおりです。

	コマンド [マッピングアイテム] : ADI [インデックス要素] : (ビット / バイト / ワード)														
ポイント	LSB							MSB							
RWr #0	C[0]:7[1]:(LSB)							C[0]:7[1]:(MSB)							
RWr #1	C[0]:7[2]:(b7)	C[0]:7[2]:(b6)	C[0]:7[2]:(b5)	C[0]:7[2]:(b4)	C[0]:7[2]:(b3)	C[0]:7[2]:(b2)	C[0]:7[2]:(b1)	C[0]:7[2]:(b0)	C[0]:7[2]:(b15)	C[0]:7[2]:(b14)	C[0]:7[2]:(b13)	C[0]:7[2]:(b12)	C[0]:7[2]:(b11)	C[0]:7[2]:(b10)	C[0]:7[2]:(b9)
RWr #2	C[0]:7[3]:(byte)							-							

4. CC-Link システム領域の実装

4.1 システム領域のモード

CC-Link システム領域は CC-Link 通信に不可欠な部分です。この領域は各種ステータスフラグと診断フラグを保持し、Anybus モジュールによって自動的に処理されるか（デフォルト）、ホストアプリケーションによって処理できます。

- システム領域が Anybus によって処理される場合（デフォルト、基本）

CC-Link ホストオブジェクト（F7h）の "System Area Handler" アトリビュート（#5）を実装しないことによってこの機能が明示的に無効にされている場合を除き、システム領域のすべてのフラグはモジュールによって自動的に処理されます。エラー指示フラグは、診断ラッチイベントによってのみ設定できます。詳細については、20 ページの「診断」を参照してください。

CC-Link ホストオブジェクトの Initial_Data_Setting_Notification コマンドと Initial_Data_Processing_Notification コマンドがサポートされている必要があります。これらのいずれかのコマンドがホストアプリケーションによって受け入れられない場合、ホストアプリケーションは "Unsupported Object" または "Unsupported Command" で応答し、要求をアクノリッジしないことを通知する必要があります。

- システム領域がホストアプリケーションによって処理される場合（高度）

"System Area Handler" アトリビュート（#5）が -1（無効）に設定されている場合は、ホストアプリケーションが単独で、CC-Link 仕様で定義されたいずれかのプロファイルに従って CC-Link ステータスフラグを処理する責任を負います。これを果たすため、ホストアプリケーションは 1 つまたは複数の ADI を CC-Link メモリマップ内の対応する位置にマッピングする必要があります。詳細については、19 ページの「システム領域の位置」を参照してください。

4.2 システム領域のレイアウト

注：このセクションは、システム領域がモジュールによって自動的に処理される場合にのみ適用されます。

スレーブ → マスター		マスター → スレーブ	
ビットオフ セット	内容	ビットオフ セット	内容
0	(予約)	0	(予約)
1		1	
2		2	
3		3	
4		4	
5		5	
6		6	
7		7	
8	イニシャルデータ処理要求	8	イニシャルデータ処理完了
9	イニシャルデータ設定完了	9	イニシャルデータ設定要求
10	エラー状態	10	エラーリセット要求
11	リモート READY	11	(予約)
12	(予約)	12	
13		13	
14		14	
15		15	

上の表に示す各種フラグがどのように処理されるかについては、40 ページの「CC-Link ハンドシェイクの実装」を参照してください。

下記も参照してください。

- 19 ページの「システム領域の位置」
- 20 ページの「診断」
- 40 ページの「CC-Link ハンドシェイクの実装」

4.3 システム領域の位置

注：このセクションは、システム領域がモジュールによって自動的に処理される場合にのみ適用されます。

デフォルトでは、システム領域は次のようにビット領域の末尾にあります。

ポイント	内容	ポイント	内容
RX #0	ユーザー領域 (ライトプロセスデータを保持)	RY #0	ユーザー領域 (リードプロセスデータを保持)
RX #1		RY #1	
...		...	
RX #Q-18		RY #Q-18	
RX #Q-17	(CC-Link システム領域のために 予約)	RY #Q-17	(CC-Link システム領域のために 予約)
RX #Q-16		RY #Q-16	
RX #Q-15		RY #Q-15	
...		...	
RX #Q-2		RY #Q-2	
RX #Q-1		RY #Q-1	

(上の表は、CC-Link マスターから見たときのデータの表現を示します。"Q" は、ABCC のビット領域内のアドレス指定可能なポイントの数を表します。)

システム領域の位置は、CC-Link ホストオブジェクト (F7h) の "System Area Handler" アトリビュート (#5) を実装することによって変更できます。また、このアトリビュートを -1 に設定して完全に無効にすることもできます。その場合はホストアプリケーションが、CC-Link 仕様 (プロファイル編) で定義されたいずれかのプロファイルに従って CC-Link 通信を処理する責任を負います。

下記も参照してください。

- 14 ページの「マッピング」
- 40 ページの「CC-Link ハンドシェイクの実装」

4.4 診断

注：このセクションは、システム領域がモジュールによって自動的に処理される場合にのみ適用されます。

本モジュールは、通常の下況下で最大 5 つの診断エントリをサポートすることに加えて、復旧不可能な重大なイベントの発生時には 6 つ目の追加エントリもサポートします。

診断は "エラー状態" フラグと "リモート READY" フラグによって表されます。

診断イベントの実際の原因に関する情報はネットワークに転送されません。ただし重大度は別で、これは "リモート READY" フラグを監視することによって暗黙的に得られます。重大なイベントが発生すると、このフラグが解除されます（設定されている場合）。

ラッチイベント機能がサポートされており、ラッチイベントによって "エラー状態" フラグが設定されます。イベントの重大度が「重大」である場合は、"リモート READY" フラグが解除されます。"エラー状態" フラグはラッチ診断イベントによってのみ設定されることに注意してください。

- "エラー状態" フラグ

このフラグは、診断オブジェクトの状態を以下のように表します。

- 1：ラッチ診断イベントが存在します。¹
- 0：診断イベントは存在しません。または、"エラーリセット要求" フラグがハイに設定されています。

- "リモート READY" フラグ

- 1：通常動作
- 0：重大度が「重大」の診断イベントが存在します。² または、"イニシャルデータ設定要求" フラグがハイに設定されています。

下記も参照してください。

- 17 ページの「システム領域のモード」
- 40 ページの「CC-Link ハンドシェイクの実装」
- 23 ページの「診断オブジェクト (02h)」

-
1. このフラグは、マスターが "エラーリセット要求" フラグを通じてイベントをアクノリッジするまでハイの状態に留まります。
 2. イベントが解決し（すなわち、ホストアプリケーションが対応する診断インスタンスを削除し）、なおかつマスターが "エラーリセット要求" フラグを通じてイベントをアクノリッジすると、通常動作が再開されます（上記のエラーステータスとエラーコードも参照）。

5. Anybus モジュールオブジェクト

5.1 概要

この章では、Anybus モジュールオブジェクトの実装について説明し、それらのオブジェクトが Anybus CompactCom CC-Link の機能にどのように対応しているかを示します。

以下の Anybus モジュールオブジェクトが実装されています。

- 22 ページの「Anybus オブジェクト (01h)」
- 23 ページの「診断オブジェクト (02h)」
- 25 ページの「ネットワークオブジェクト (03h)」
- 27 ページの「ネットワークコンフィグレーションオブジェクト (04h)」
- 30 ページの「ネットワーク CC-Link オブジェクト (08h)」
- 32 ページの「Anybus ファイルシステムインターフェースオブジェクト (0Ah)」

5.2 Anybus オブジェクト (01h)

カテゴリ

基本

オブジェクトの説明

このオブジェクトは、共通の Anybus 情報を集約します。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
 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	値
1	Module type	Get	UINT16	0403h (Anybus CompactCom 40)
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 : UINT8 (LED1A) UINT8 (LED1B) UINT8 (LED2A) UINT8 (LED2B)	値 : 色 : 01h 緑 02h 赤 01h 緑 02h 赤
13 : 18	-	-	-	詳細については、『Anybus CompactCom Software Design Guide』を参照してください。
19	Network time	Get	UINT64	0 (CC-Link によってサポートされていない)

5.3 診断オブジェクト (02h)

カテゴリ

基本

オブジェクトの説明

このオブジェクトは、ホストアプリケーションのイベントと診断を処理する標準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 40 Design Guide』で詳しく説明されています。

CC-Link の場合、診断は CC-Link システム領域のフラグによって表されます。モジュールがシステム領域を処理する場合は、CC-Link コンフォーマンステストに合格するために 1 つの診断インスタンスを作成 / 削除する必要があります。ホストアプリケーションがシステム領域を処理する場合（高度）、診断オブジェクトを実装する必要はありません。

CC-Link ホストオブジェクト (F7h) のインスタンス #1 のアトリビュート #5 でシステム領域ハンドラ機能を有効にした場合（35 ページを参照）、本モジュールはラッチ診断イベントの確認応答をサポートします。ラッチイベントを作成できない場合は、ネットワーク固有のエラー応答が返されます（24 ページを参照）。

本モジュールはこのオブジェクトのインスタンスを 6 個までサポートします。そのうち 1 つは復旧不可能な重大なイベントのために予約されています。診断イベントの実際の原因に関する情報はネットワークに転送されません。得られるのは重大度だけです。

下記も参照してください。

- 17 ページの「システム領域のモード」
- 18 ページの「システム領域のレイアウト」
- 20 ページの「診断」

サポートされているコマンド

オブジェクト : Get_Attribute
 Create (24 ページの「コマンド詳細 : Create」を参照)
 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	値
1	Name	Get	CHAR の配列	"Diagnostic"
2	Revision	Get	UINT8	01h
3	Number of instances	Get	UINT16	『Anybus CompactCom Software Design Guide』を参照。
4	Highest instance no.	Get	UINT16	
11	Max no. of instances	Get	UINT16	5+1
12	Supported functionality	Get	BITS32	0000 0000h (イベントのラッチはサポートされていない)。 または 0000 0001h (イベントのラッチがサポートされている)

インスタンスアトリビュート

#	名前	アクセス	種類	値
1	Severity	Get	UINT8	『Anybus CompactCom 40 Software Design Guide』を参照。
2	Event Code	Get	UINT8	
3-7	(予約)			

コマンド詳細 : Create

カテゴリ

基本

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

このコマンドは新規インスタンスを作成します。ここでいうインスタンスとは、ホストアプリケーションの新規診断イベントのことです。

ラッチイベントを作成できない場合は、以下に説明するネットワーク固有の拡張エラー応答が返されます。

- **コマンド詳細 :**
詳細については、『Anybus CompactCom 40 Software Design Guide』を参照してください。

- **応答詳細 (成功) :**

フィールド	内容
MsgData[0-1]	作成されたインスタンスの番号。

- **応答詳細 (エラー) :**

バイト	項目	説明
MsgData[0]	FFh	オブジェクト固有のエラー
MsgData[1]	FFh	ネットワーク固有のエラー
MsgData[2]	01h 02h	CC-Link 固有のエラーコード、下記を参照。

CC-Link 固有のエラーコード	説明
01h	無効な状態のため（モジュールが WAIT_PROCESS 状態でなかったため）、ラッチイベントを作成できませんでした。
02h	システム領域機能が有効になっていないため、ラッチイベントを作成できませんでした。

5.4 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 40 CompactCom Software Design Guide』を参照してください。

下記も参照してください。

- 30 ページの「ネットワーク CC-Link オブジェクト（08h）」

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
Map_ADI_Write_Ext_Area
Map_ADI_Read_Ext_Area

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	値
1	Name	Get	CHAR の配列	"Network"
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）

基本

#	名前	アクセス	種類	値
1	Network type	Get	UINT16	0090h
2	Network type string	Get	CHAR の配列	'CC-Link'
3	Data format	Get	ENUM	00h (LSB ファースト)
4	Parameter data support	Get	BOOL	False
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） Map_ADI_Write_Area ^a または Map_ADI_Write_Ext_Area ^a が成功するたびに更新される
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） Map_ADI_Read_Area ^a または Map_ADI_Write_Ext_Area ^a が成功するたびに更新される
7	Exception Information	Get	UINT8	42 ページの「例外情報」を参照。

a. 詳細については、『Anybus CompactCom 40 Software Design Guide』を参照してください。

5.5 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

基本

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（NVS ストレージのリセットも含む）。局番は未設定になり、ボーレートは 2（2.5 Mbps）になります。

下記も参照してください。

- 9 ページの「通信設定」
- 34 ページの「CC-Link ホストオブジェクト (F7h)」

注：CC-Link 認定ポリシーに従って最終製品の再認定を受ける場合は、インスタンス #1 と #2 を実装する必要があります。

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）

#	名前	アクセス	データタイプ	値
1	Name	Get	CHAR の配列	"Network configuration"
2	Revision	Get	UINT8	01h
3	Number of instances	Get	UINT16	0002h
4	Highest instance no.	Get	UINT16	0002h

インスタンスアトリビュート（インスタンス #1、"Station number"）

このインスタンスは実際の CC-Link 局番を保持します。

基本

#	名前	アクセス	種類	説明
1	Name ^a	Get	CHAR の配列	'Station number'
2	Data type	Get	UINT8	04h (UINT8)
3	Number of elements	Get	UINT8	01h (要素数 1)
4	Descriptor	Get	UINT8	07h (get/set/shared アクセス)
5	Value ^b	Get/Set	UINT8	<p>値：設定：</p> <p>0：未設定（使用不可）</p> <p>1～64：局番（注を参照）</p> <p>>64：（無効）</p> <p>注：局番と占有局の数の合計が 65 を超えることはできません。</p> <p>値が無効な場合、Setup Complete コマンドはア クノリッジされません。</p>
6	Configured Value	Get	UINT8	<p>設定された値を保持します。これは後でアトリ ビュート #5 に書き込まれます。</p> <p>値：設定：</p> <p>0：未設定（使用不可）</p> <p>1～64：局番（注を参照）</p> <p>>64：（無効）</p>

a. Multilingual（多言語）。29 ページの「多言語文字列」を参照。

b. セットアップの完了後にこのアトリビュートに書き込まれた値は、アトリビュート #5 ではなくアトリビュート #6 に保存されます。電源がオン/オフされた後、アトリビュート #5 はアトリビュート #6 に保存された値で更新されます。

インスタンスアトリビュート（インスタンス #2、"Baud rate"）

このインスタンスは実際の CC-Link データレートを保持します。

基本

#	名前	アクセス	種類	説明
1	Name ^a	Get	CHAR の配列	'Baud rate'
2	Data type	Get	UINT8	08h (ENUM)
3	Number of elements	Get	UINT8	01h (要素数 1)
4	Descriptor	Get	UINT8	07h (get/set/shared アクセス)
5	Value ^b	Get/Set	UINT8	<p>値：速度 / 文字列：</p> <p>0 : 156 kbps 1 : 625 kbps 2 : 2.5 Mbps (デフォルト) 3 : 5 Mbps 4 : 10 Mbps その他：(無効)</p> <p>注：値が無効な場合、Setup Complete コマンドはアクノリッジされません。</p>
6	Configured Value	Get	UINT8	<p>設定された値を保持します。これは後でアトリビュート #5 に書き込まれます。</p> <p>値：速度 / 文字列：</p> <p>0 : 156 kbps 1 : 625 kbps 2 : 2.5 Mbps 3 : 5 Mbps 4 : 10 Mbps その他：(無効)</p>

a. Multilingual (多言語)。29 ページの「多言語文字列」を参照。

b. セットアップの完了後にこのアトリビュートに書き込まれた値は、アトリビュート #5 ではなくアトリビュート #6 に保存されます。電源がオン/オフされた後、アトリビュート #5 はアトリビュート #6 に保存された値で更新されます。

多言語文字列

このオブジェクトのインスタンス名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
1	Station no	Geräteadresse	Direcc nodo	Indirizzo	Adresse
2	Baud rate	Datenrate	Veloc transf	Velocità dati	Vitesse

5.6 ネットワーク CC-Link オブジェクト (08h)

カテゴリ

基本、拡張

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

インスタンス : Get_Attribute

CCL_Conf_Test_Mode

(31 ページの「コマンド詳細 : CCL_Conf_Test_Mode」を参照)。

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	値
1	Name	Get	CHAR の配列	"Network CC-Link"
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)

基本

#	名前	アクセス	種類	値
1	Network Settings ^a	Get	以下に示す 構造体 : UINT8 UINT8 UINT8	現在のネットワーク設定 : - CC-Link のバージョン。01h = v1.10、02h = v2.00 - 占有局の数 - 拡張サイクルの数
2	System Area Handler ^a	Get	SINT16	システム領域の位置 (システム領域がホストアプリケーションによって処理される場合は -1)
3 ~ 4	(予約)			
5	CC-Link V.2 Conformance test mode	Get	BOOL	特別な CC-Link バージョン 2.00 コンFORMANCE テストモードが有効な場合は TRUE (このモードは CCL_Conf_Mode コマンドによって有効にされます)

a. これらのアトリビュートは、ホストアプリケーションが CC-Link ホストオブジェクト (F7h) で他の値を指定していない限り、モジュールによって自動的に計算されます。アトリビュート値は WAIT_PROCESS への最初の遷移から有効になります。これらの値は、例えば開発中に CC-Link システム領域の位置を確立するために使用できます。

コマンド詳細 : CCL_Conf_Test_Mode

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、特別な CC-Link バージョン 2 コンフォーマンステストモードを有効 / 無効にします。CC-Link バージョン 2 のコンフォーマンステストでは、特別なバージョンのスレーブが、RY/RWw から RX/RWx までの受け入れられるすべてのデータをループ処理する必要があります。このコマンドを使用して、この機能を一時的または恒久的に有効にするか、または恒久的に無効にできます。このコマンドは SETUP 中にのみ発行できます。

• コマンド詳細 :

フィールド	内容								
CmdExt[0]	<p>コンフォーマンステストモード</p> <table border="1"> <thead> <tr> <th>値 :</th><th>説明 :</th></tr> </thead> <tbody> <tr> <td>00h</td><td>コンフォーマンステストモードを恒久的に無効にします。</td></tr> <tr> <td>01h</td><td>コンフォーマンステストモードを一時的に有効にします。リセットまたは電源をオフ / オンした後、コンフォーマンステストモードは無効になります。</td></tr> <tr> <td>02h</td><td>コンフォーマンステストモードを恒久的に有効にします。</td></tr> </tbody> </table>	値 :	説明 :	00h	コンフォーマンステストモードを恒久的に無効にします。	01h	コンフォーマンステストモードを一時的に有効にします。リセットまたは電源をオフ / オンした後、コンフォーマンステストモードは無効になります。	02h	コンフォーマンステストモードを恒久的に有効にします。
値 :	説明 :								
00h	コンフォーマンステストモードを恒久的に無効にします。								
01h	コンフォーマンステストモードを一時的に有効にします。リセットまたは電源をオフ / オンした後、コンフォーマンステストモードは無効になります。								
02h	コンフォーマンステストモードを恒久的に有効にします。								
CmdExt[1]	(予約)								

• 応答詳細 (成功) :

フィールド	内容
Data[0 .. n]	(未使用)

• 応答詳細 (エラー) :

エラー	内容
標準エラーコード	ABCC 機能仕様に準拠

5.7 Anybus 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、Anybus CompactCom 40 CC-Link モジュールに組み込まれたファイルシステムへのインターフェースを提供します。このファイルシステムは、"Firmware" という 1 つのフォルダで構成されています。このフォルダは、モジュールをアップグレードするファームウェアファイルの保存に使用されます。リセット後、モジュールのファームウェアがアップグレードされてファイルが消去されます。

詳細については、『Anybus CompactCom 40 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

(詳細については、『Anybus CompactCom 40 Software Design Guide』を参照してください)

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 40 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

(詳細については、『Anybus CompactCom 40 Software Design Guide』を参照してください)

6. ホストアプリケーションオブジェクト

6.1 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、CC-Link の機能を拡張することができます。

ネットワーク固有オブジェクト：

- 34 ページの「CC-Link ホストオブジェクト (F7h)」

6.2 CC-Link ホストオブジェクト (F7h)

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 CC-Link 固有の機能が実装されます。

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。

サポートされているコマンド

オブジェクト : Get Attribute
 Initial_Data_Setting_Notification
 (36 ページの「コマンド詳細 : Initial_Data_Setting_Notification」を参照)。
 Initial_Data_Processing_Notification
 (37 ページの「コマンド詳細 : Initial_Data_Processing_Notification」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	値
1	Name	Get	CHAR の配列	'CC-Link'
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	コメント
1	Vendor Code	Get	UINT16	0212h ('HMS')	CLPA によって割り当てられます。
2	SW Version	Get	UINT8	(Anybus ファームウェアリリース番号)	有効な設定は 1 ~ 3Fh です。

拡張

#	名前	アクセス	種類	デフォルト値	コメント
3	Model Code	Get	UINT8	7Fh ('Generic Device')	可能な設定については、CC-Link 仕様（プロファイル編）のデバイスタイプコードを参照してください。
4	Network Settings	Get	以下の配列：		
			UINT8	01h	CC-Link のバージョン： 01h : v1.10（デフォルト） 02h : v2.00
			UINT8	(自動的に計算される)	占有局の数： 有効な設定 : 01h, 02h, 03h, 04h
			UINT8	1	拡張サイクルの数 有効な設定、CC-Link v1 : 01h 有効な設定、CC-Link v2 : 01h, 02h, 04h, 08h

高度

#	名前	アクセス	種類	デフォルト値	コメント
5	System Area Handler	Get	SINT16	(自動的に計算される。ビット領域の最後の 16 ビット)	システム領域は以下によって処理されます。 -1 : ホストアプリケーション 0...880 ^a : モジュール。値は、ビット領域内のシステム領域のオフセットを示します。偶数の 16 ビット境界に位置する必要があります。
6	Output Hold/Clear	Get	UINT8	1 (保持)	出力アクション： 0 : クリア 1 : 保持 このアトリビュートは、マスター（マスター局のアプリケーション）でアクティブなエラーが発生したときにアプリケーションが出力データをどのように処理するかを CC-Link マスターに指示します。これにより、マスターからスレーブへの出力値の送信が回避されます。この状態は、Anybus の状態 "ERROR" によってアプリケーションに通知されます。

a. 拡張

コマンド詳細 : Initial_Data_Setting_Notification

カテゴリ

基本

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

注 : このセクションは、システム領域がモジュールによって自動的に処理される場合にのみ適用されます。

このコマンドは、マスターがイニシャルデータ設定サイクルを開始するとき（すなわち、"イニシャルデータ設定要求" フラグの立ち上がりエッジで）発行されます。

ホストアプリケーションはこのコマンドを受け入れるか、("Unsupported Object" または "Unsupported Command" で応答することによって）拒絶できます。どちらの場合も、応答が受信されるとモジュールは "イニシャルデータ設定完了" フラグを設定してイニシャルデータ設定サイクルを続行します。

- **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 : Initial_Data_Processing_Notification

カテゴリ

基本

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

注 : このセクションは、システム領域がモジュールによって自動的に処理される場合にのみ適用されます。

このコマンドは、自動ハンドシェイク中に WAIT_PROCESS から PROCESS_ACTIVE への状態遷移が起こったことをアプリケーションに通知します。アプリケーションが CC-Link オブジェクトの "System Area Handler" アトリビュートを使用して自動ハンドシェイクを無効にしている場合、または CC-Link オブジェクトが存在しない場合には、このコマンドは発行されません。コマンドとともに追加情報は送信されず、応答メッセージでも情報は期待されません。

ホストアプリケーションはこのコマンドを受け入れるか、("Unsupported Object" または "Unsupported Command" で応答することによって) 拒絶できます。どちらの場合も、応答が受信されるとモジュールは "イニシャルデータ処理完了" フラグを設定してイニシャルデータ設定サイクルを続行します。

- **コマンド詳細**
(データなし)
- **応答詳細**
(データなし)

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転送だけでなく、産業用ネットワーク固有の機能を利用できるようになります。これにより、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. 実装詳細

B.1 SUP ビットの定義

監視ビット（SUP）は、モジュールが CC-Link マスターとデータを交換していることを示します。

B.2 Anybus ステートマシン

次の表に、Anybus のステートマシンと CC-Link ネットワークとの対応を示します。

Anybus の状態	実装 (システム領域が Anybus モジュールによって処理される)	実装 (システム領域がホストアプリケーションによって処理される)
WAIT_PROCESS	CC-Link の自動ハンドシェイクが進行中	最初のリフレッシュメッセージの受信待ち
ERROR	考えられる原因： <ul style="list-style-type: none"> - タイムアウトエラー（ERR21） - 0 チャネルキャリア検出状態（ERR22） - ホストのデータ数が不十分（ERR30 ～ 32） 問題が解決すると、モジュールは前の状態に戻ります。	
PROCESS_ACTIVE	モジュールは " イニシャルデータ処理完了 " の立ち上がりエッジでこの状態になる	モジュールは最初のリフレッシュメッセージが受信されたときにこの状態になる
IDLE	PLC / マスターが STOP モードになっている PLC / マスターが RUN モードに戻ると、モジュールは前の状態に戻ります。	
EXCEPTION	考えられる原因： <ul style="list-style-type: none"> - コンフィギュレーションエラー - これ以外の方法でアプリケーションに通知できないその他のエラー 例： <ul style="list-style-type: none"> - CC-Link ホストオブジェクト（F7h）内のアトリビュート設定の無効な組み合わせ - CC-Link ホストオブジェクト（F7h）内のマッピングコマンドとアトリビュート設定の無効な組み合わせ 	

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

アプリケーションウォッチドッグのタイムアウトが検出されると、モジュールはネットワークへの参加を中止して "EXCEPTION" 状態に移行します。それ以外のネットワーク固有の動作は起こりません。

B.4 CC-Link ハンドシェイクの実装

注：このセクションは、システム領域がモジュールによって自動的に処理される場合にのみ適用されます。

システム領域が Anybus モジュールによって処理される場合、すべての CC-Link ハンドシェイクは次の表に示すように自動的に実行されます。

フラグ	設定されるとき	クリアされるとき
イニシャルデータ処理要求	状態が NW_INIT から WAIT_PROCESS に遷移するとき	以下のシーケンスが完了したとき： 1. 状態が WAIT_PROCESS から PROCESS_ACTIVE に遷移する 2. ホストアプリケーションが Initial_Data_Processing_Notification ^a に応答した この手順の目的は、モジュールが PROCESS_ACTIVE 状態に移行したことがホストアプリケーションによって確実に検出されるようにすることにあります。
イニシャルデータ設定完了	ホストアプリケーションが Initial_Data_Setting_Notification に応答するとき ^b	"イニシャルデータ設定要求"の立ち下がりエッジのとき
エラー状態 ^c	ラッチイベントが少なくとも1つ存在し ^d 、"エラーリセット要求"が false であるとき	"エラーリセット要求"が設定されているとき
リモート READY ^e	(初期設定) - "イニシャルデータ処理完了"の立ち上がりエッジのとき (実行時) - "エラー状態"と "エラーリセット要求"が false であるとき	"イニシャルデータ設定要求"が設定されているか、重大度が「重大」のラッチイベントが少なくとも1つ存在するとき

a. 37 ページの「コマンド詳細：Initial_Data_Processing_Notification」を参照。

b. 36 ページの「コマンド詳細：Initial_Data_Setting_Notification」を参照。

c. このフラグを通じて追加機能が処理されます（20 ページの「診断」を参照）。

d. 20 ページの「診断」を参照。

e. 設定条件とクリア条件が両方とも満たされている場合は、クリア機能が優先されます。

注：システム領域の初期値は false です。すなわち、起動時にはすべてのフラグがクリアされます。

C. 認定情報

C.1 基本

認定を得るためには以下の手順を実行する必要があります。

1. ベンダーコードを変更する

HMS ベンダー ID を一意のベンダーコード¹に置き換えます。そのためには、CC-Link ホストオブジェクト (F7h) のインスタンス 1 のアトリビュート 1 を実装し、Get_Attribute 要求を受信したときにベンダーコードを返します。

2. ソフトウェアのバージョンのインクリメントする

バージョンを追跡する場合は、CC-Link ホストオブジェクト (F7h) でソフトウェアのバージョンをインクリメントします。この番号は機能を変更するたびにインクリメントする必要があり、これが新たな再認定につながります。CC-Link ホストオブジェクト (F7h) のインスタンス 1 のアトリビュート 2 を実装し、Get_Attribute 要求を受信したときにソフトウェアのバージョンを返します。

3. 保持 / クリアアトリビュートの設定をチェックする

CC-Link ホストオブジェクト (F7h) のインスタンス 1 のアトリビュート 6 (Output Hold / Clear) の設定を、エラーの発生時にホスト (PLC) によって適用される出力フェイルステートに合わせる必要があります。

4. 新しい CSP+ ファイルを生成する

カスタマイズした実装に準拠した CSP+ ファイルを生成する必要があります。

これらの手順は、認定を得るために最低限行わなければならないことを示します。

C.2 モデルコード

モジュールの出荷時には、モデルコード (CC-Link ホストオブジェクト (F7h) のインスタンス 1 のアトリビュート 3) は 7Fh ("Generic device") に設定されています。ホストアプリケーションが既存の CC-Link プロファイルに類似している場合は、そのプロファイルを反映するようにこのコードを変更してください。

C.3 CC-Link バージョン 2.00

CC-Link ホストオブジェクト (F7h) の "Network Settings" アトリビュート (#4) を実装することで、CC-Link バージョン 2.00 の実装をカスタマイズし、拡張サイクルを通じてより大きなデータサイズを使用できます。

CC-Link バージョン 2.00 のコンフォーマンステストでは、CC-Link コンフォーマンステストモードを有効にする必要があります。30 ページの「ネットワーク CC-Link オブジェクト (08h)」の CCL_Conf_Test_Mode コマンド (31 ページ) を参照してください。

-
1. ベンダーコードを入手するには、CLPA 協会に加入する必要があります。ベンダーコードは、CLPA への加入時に発行される CLPA ID 番号に含まれる 5 ～ 8 桁のコードです。

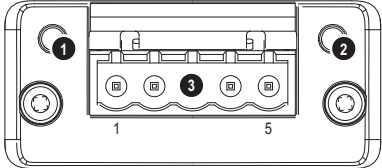
D. 例外情報

モジュールが EXCEPTION 状態になったとき、ネットワークオブジェクト（03h）の "Exception Information" アトリビュート（#7）から、問題に関する CC-Link 固有の詳細情報を取得できます。

Value	意味
00h	情報なし。
01h	CC-Link オブジェクトの SW Version アトリビュートの値が範囲外。
02h	CC-Link オブジェクトの Network Settings アトリビュートに設定された CC-Link バージョンの値が範囲外。
03h	CC-Link オブジェクトの Network Settings アトリビュートに設定された局数の値が範囲外。
04h	CC-Link オブジェクトの Network Settings アトリビュートに設定された拡張サイクルの値が無効。
05h	CC-Link オブジェクトの System Area Handler アトリビュートの値が無効。
06h	CC-Link バージョンと拡張サイクルが適合していない。
07h	CC-Link バージョン 1 の CC-Link マップで、高すぎるオフセットにデータがマッピングされた。
08h	システム領域が自動的に配置されるオフセットにデータがマッピングされた。
09h	CC-Link バージョン 1 で、自動的に配置されるシステム領域用のスペースがない。
0Ah	ネットワーク設定が指定された CC-Link マップで、高すぎるオフセットにデータがマッピングされた。
0Bh	ネットワーク設定で指定されたシステム領域の位置が範囲外。
0Ch	CC-Link バージョン 1 の CC-Link マップで、高すぎるオフセットにシステム領域が配置された。
0Dh	システム領域の位置にデータがマッピングされた。
0Eh	(未使用)
0Fh	(未使用)
10h	設定されたデバイスアドレスが、使用されている占有局の数に対して高すぎる。
11h	CC-Link ホストオブジェクトの Hold / Clear ステータスアトリビュート（#6）の値が範囲外。

E. 技術仕様

E.1 前面図（M40）

#	項目	
1	Run LED	
2	Error LED	
3	CC-Link コネクタ	

E.1.1 Run LED

State	意味
オフ	- ネットワークに参加していない、タイムアウトステータス（電源がオフになっている）
緑	- ネットワークに参加中、通常動作
赤	- 重大な障害（FATAL エラー）

E.1.2 Error LED

State	意味
オフ	- エラーは検出されていない（電源がオフになっている）
赤	- 重大な障害（例外または FATAL イベント）
赤、ちらつき	- CRC エラー（一時的なちらつき）
赤、点滅	- 局番またはボーレートが起動後に変更された（点滅）

E.1.3 CC-Link コネクタ

ピン	信号	コメント
1	DA	ポジティブ RS485 RxD/TxD
2	DB	ネガティブ RS485 RxD/TxD
3	DG	信号グラウンド
4	SLD	ケーブルシールド
5	FG	保護接地

E.2 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。

E.3 電源

電源電圧

本モジュールを使用するには、『Anybus CompactCom Hardware Design Guide』で指定された 3.3 V の安定化電源が必要です。

消費電力

Anybus CompactCom CC-Link 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 280mA を消費します¹。

注：ホストアプリケーションにおける電源の設計は、ある 1 つの製品の消費電力に関する厳密な要件ではなく、『Anybus CompactCom Hardware Design Guide』で記述された消費電力の分類に基づいて行うことを強く推奨します。

E.4 環境仕様

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

E.5 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。

-
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 CC-Link はクラス B モジュールであることを維持します。

F. タイミングと性能

F.1 概要

この章では、Anybus CompactCom 40 CC-Link 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	ページ
起動遅延	T1, T2	45
NW_INIT の処理	T100	45
イベントベースの WrMsg ビジー時間	T103	46
イベントベースのプロセスデータ遅延時間	T101, T102	46

詳細については、『Anybus CompactCom 40 Software Design Guide』を参照してください。

F.2 内部タイミング

F.2.1 起動遅延

以下のパラメータは、/RESET が解放されてから指定のイベントが発生するまでの測定時間として定義されています。

パラメータ	説明	最大	単位
T1	Anybus CompactCom 40 CC-Link モジュールが最初のアプリケーション割り込みを生成（パラレルモード）。	11	ms
T2	Anybus CompactCom 40 CC-Link モジュールが最初のアプリケーションテレグラムを受信 / 処理可能となった（シリアルモード）。	11	ms

F.2.2 NW_INIT の処理

このテストは、Anybus CompactCom 40 CC-Link モジュールが NW_INIT 状態で必要な動作を実行するために要する時間を測定します。

パラメータ	条件
ネットワーク固有のコマンドの数。	最大
各方向における、プロセスデータにマッピングされた ADI (UINT8 × 1) の数。	32 ^a
イベントベースで処理する場合の、アプリケーションのメッセージ応答時間。	> 1 ms
ピンポン方式で処理する場合の、アプリケーションの応答時間。	> 10 ms
アプリケーションが同時に処理可能な、Anybus の未処理コマンドの数。	1

a. または、ネットワーク固有の最大数がそれより小さい場合は、その最大数。

パラメータ	説明	通信	最大	単位
T100	NW_INIT の処理	イベントベースのすべてのモード	14	ms

F.2.3 イベントベースの WrMsg ビジー時間

アプリケーションがメッセージをポストしてから、モジュールが H_WRMSG 領域をアプリケーションに返すまでに要する時間として、イベントベースの WrMsg ビジー時間が定義されています。

パラメータ	説明	最大	単位
T103	H_WRMSG 領域のビジー時間	16	μs

F.2.4 イベントベースのプロセスデータ遅延時間

"リードプロセスデータ遅延時間" は、ネットワークフレームの最後のビットがネットワークインターフェースによって受信されてから、アプリケーションに対して RDPDI 割り込みがアサートされるまでの時間として定義されます。

"ライトプロセスデータ遅延時間" は、アプリケーションがライトプロセスデータバッファを交換してから、新規プロセスデータフレームの最初のビットがネットワークに送出されるまでの時間として定義されます。

このテストは 16 ビットパラレルイベントモードで、新規プロセスデータイベントに対してのみ割り込みをトリガーするという条件で実施されました。遅延時間は CC-Link バージョン 1.10 のみで測定されました。CC-Link バージョン 2.00 は、いくつかの CC-Link サイクル (2、4、8) を使用してモジュールとの間でデータを転送します。これは、モジュールに起因する遅延よりもサイクルタイムの方が重要な要素となることを意味します。

トランシーバやその他のハードウェアによって増加した遅延は、プロセスデータ遅延時間全体に比べるとわずかな量なので、結果に含まれていません。

パラメータ	説明	遅延 (最大)	単位
T101	リードプロセスデータ遅延時間	33	μs
T102	ライトプロセスデータ遅延時間	35	μs

