

OMRON

V460-H

ハンディ DPM コードリーダー

ユーザーズマニュアル 通信設定編



SDNC-769A

— おことわり —

- 本マニュアルの内容の一部または全部を無断で複写、複製、転載することを禁じます。
- 本マニュアルの内容に関しては、改良のため予告なしに仕様などを変更することがあります。あらかじめご了承ください。
- 本マニュアルの内容に関しては、万全を期しておりますが、万一ご不審な点や誤りなど、お気づきのことがありましたら、お手数ですが、弊社支店または営業所までご連絡ください。その際、巻末記載の Man.No.（マニュアル No.）も併せてお知らせください。

— 商標 —

- Sysmac は、オムロン株式会社 FA 機器製品の日本およびその他の国における商標または登録商標です。
 - Microsoft、Windows、Windows Vista、Excel、Visual Basic は、米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。
 - ODVA、CIP、CompoNet、DeviceNet、EtherNet/IP は ODVA の商標です。
 - QR コードは、(株)デンソーウェーブの登録商標です。
- その他、記載されている会社名と製品名などにつきましては、各社の登録商標または商標です。

— 著作権 —

スクリーンショットはマイクロソフトの許諾を得て使用しています。

はじめに

このたびは、V460-H ハンディ DPM コードリーダをお買い上げいただきまして、まことにありがとうございます。

本マニュアルは、V460-H ハンディ DPM コードリーダを使用する上で、必要な情報を記載しています。

お使いになる前に、本マニュアルをよく読んで、機能・性能などを十分ご理解いただき、システムの構築にご活用ください。

また、お読みになった後も、本マニュアルを大切に保管して、いつも手元においてお使いください。

対象となる読者の方々

本マニュアルは、次の方を対象に記述しています。電気の知識（電気工事士あるいは同等の知識）を有する方で

- FA 機器の導入を担当される方
- FA システムを設計される方
- FA 機器を設置、接続される方
- FA 現場を管理される方

対象となる商品

本マニュアルは、以下の製品を対象にしています。

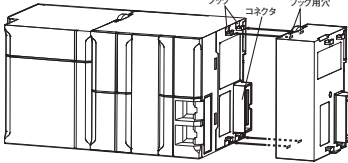
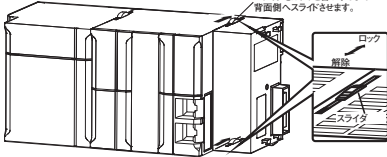
- V460-H ハンディ DPM コードリーダ

各製品の仕様の一部または制約事項が、他のマニュアルに記載されている場合があります。「関連マニュアル (P.14)」をご確認ください。

マニュアルの読み方

ページ構成

このマニュアルの各ページの構成は次のとおりです。

		4 設置と配線		章タイトル
				節タイトル
				項タイトル
				現在のページの章／節／項タイトルをそれぞれ示します。
節タイトル	4-3 ユニットの取り付け			
項タイトル	4-3-1 ユニット同士の接続			
操作手順	ユニット同士は、それぞれのコネクタをかみ合わせ、スライダをロックするだけで接続可能です。右端のユニットには、エンドカバーを接続します。			
操作手順	1 コネクタをきちんとかみ合わせて、ユニット同士を接続します。			
		4-3 ユニットの取り付け		
	2 上下に付いている黄色のスライダを「カチッ」と音がするまでスライドさせ、ロックします。		4	章番号
		4-3-1 ユニット同士の接続		現在のページの章番号を示します。
注記、補足、参照先	安全上の要旨			
	電源ユニットや CPU ユニット、I/O ユニット、高性能 I/O ユニット、CPU 高性能ユニットを接続するときは、ユニット間コネクタを結合後、上下のスライダをカチッと音がするまでスライドさせて、確実にロックしてください。			
注記、補足、参照先などの各項目を、アイコンによって示しています。				
マニュアル名称	NJ シリーズ CPU ユニット ユーザーズマニュアル ハードウェア編 (SBGA-358)		4 - 9	

(注) このページは、説明用の見本です。実際の内容とは異なります。

アイコン

このマニュアルで使われているアイコンには、次のような意味があります。



安全上の要点

製品を安全に使用するために、実施または回避すべきことを示します。



使用上の注意

製品が、動作不能、誤動作、または性能や機能への悪影響を予防するために、実施または回避すべきことを示します。



参考

必要に応じて、読んでいただきたい項目です。
知っておくと便利な情報や、使用する上で参考となる内容について説明しています。

目次構成

		1
1	通信仕様の概要	2
2	イーサネットで制御／出力する	3
3	PROFINETで制御／出力する	A
A	付録	

目次

はじめに.....	1
対象となる読者の方々	1
対象となる商品.....	1
マニュアルの読み方	2
ページ構成.....	2
アイコン	3
目次構成.....	5
ご承諾事項.....	8
安全上の注意.....	10
安全上の要点.....	11
使用上の注意.....	12
法規と規格.....	13
関連マニュアル	14
マニュアル改訂履歴	15

第 1 章 通信仕様の概要

1-1 システム構成の確認	1-2
1-1-1 V460-H シリーズのシステム構成.....	1-2
1-2 外部装置との通信について.....	1-3
1-2-1 コードリーダー制御の基本動作	1-3
1-2-2 V460-H ハンディ DPM コードリーダーと通信可能な通信プロトコル	1-4

第 2 章 イーサネットで制御／出力する

2-1 EtherNet/IP で制御／出力する	2-2
2-1-1 EtherNet/IP の概要.....	2-2
2-1-2 EtherNet/IP 接続時のコードリーダーの通信方法	2-4
2-1-3 PLC とコードリーダーの通信の流れ.....	2-5
2-1-4 通信仕様を設定する（EtherNet/IP）	2-6
2-1-5 タグデータリンクの設定方法	2-8
2-1-6 入力・出力アセンブリ別 状態信号・制御信号一覧	2-10
2-1-7 アセンブリ別タイミングチャート	2-11
2-1-8 サンプルラダープログラム.....	2-13
2-1-9 NJ/NX シリーズコントローラで各通信エリアに変数でアクセスする方法	2-14
2-1-10 EtherNet/IP のメッセージ通信でコントローラと通信する	2-25
2-2 シリアル（TCP）で制御／出力する.....	2-27
2-2-1 シリアル（TCP）の概要	2-27
2-2-2 通信処理の流れ	2-27
2-2-3 通信仕様を設定する（シリアル（TCP））	2-28
2-2-4 読み取り後に出力するデータの設定	2-30
2-2-5 外部からコードリーダーを制御する	2-35
2-2-6 シリアルコマンド一覧	2-38

第 3 章 PROFINET で制御／出力する

3-1	PROFINET の概要	3-2
3-1-1	PROFINET の種類	3-2
3-2	PROFINET 接続による本コードリーダーとの通信	3-5
3-2-1	通信エリアの種類	3-5
3-3	PROFINET の通信設定	3-6
3-3-1	コードリーダーのネットワーク設定	3-6
3-4	アセンブリ別タイミングチャート	3-9
3-4-1	読み取り実行信号 (TRIG 信号) による読み取り実行	3-9
3-5	サンプルラダープログラム	3-11

付録

A-1	コマンド一覧	A-2
A-1-1	コマンド一覧	A-2
A-2	EtherNet/IP 通信仕様詳細	A-3
A-2-1	ファームウェアバージョンごとの EDS ファイル	A-3
A-2-2	メモリ割り当て	A-3
A-3	PROFINET - V460-H 入出力モジュール	A-27
A-3-1	モジュール種別	A-27
A-3-2	データ型	A-29
A-3-3	PROFINET 基本情報	A-30
A-3-4	タイミングチャート	A-31

ご承諾事項

「当社商品」について特別の合意がない場合には、お客様のご購入先にかかわらず、本ご承諾事項記載の条件を適用いたします。

定義

本ご承諾事項中の用語の定義は次のとおりです。

- ・「当社商品」：「当社」の FA システム機器、汎用制御機器、センシング機器、電子・機構部品
- ・「カタログ等」：「当社商品」に関する、ベスト制御機器オムロン、電子・機構部品総合カタログ、その他のカタログ、仕様書、取扱説明書、マニュアル等であって電磁的方法で提供されるものも含みます。
- ・「利用条件等」：「カタログ等」に記載の、「当社商品」の利用条件、定格、性能、動作環境、取り扱い方法、利用上の注意、禁止事項その他
- ・「お客様用途」：「当社商品」のお客様におけるご利用方法であって、お客様が製造する部品、電子基板、機器、設備またはシステム等への「当社商品」の組み込み、または利用を含みます。
- ・「適合性等」：「お客様用途」での「当社商品」の(a)適合性、(b)動作、(c)第三者の知的財産の非侵害、(d)法令の遵守および(e)各種規格の遵守

記載事項のご注意

「カタログ等」の記載内容については次の点をご理解ください。

- ・定格値および性能値は、単独試験における各条件のもとで得られた値であり、各定格値および性能値の複合条件のもとで得られる値を保証するものではありません。
- ・参考データはご参考として提供するもので、その範囲で常に正常に動作することを保証するものではありません。
- ・利用事例はご参考ですので、「当社」は「適合性等」について保証いたしかねます。
- ・「当社」は、改善や当社都合等により、「当社商品」の生産を中止し、または「当社商品」の仕様を変更することがあります。

ご利用にあたってのご注意

ご採用およびご利用に際しては次の点をご理解ください。

- ・定格・性能ほか「利用条件等」を遵守しご利用ください。
- ・お客様ご自身にて「適合性等」をご確認いただき、「当社商品」のご利用の可否をご判断ください。「当社」は「適合性等」を一切保証いたしかねます。
- ・「当社商品」がお客様のシステム全体の中で意図した用途に対して、適切に配電・設置されていることをお客様ご自身で、必ず事前に確認してください。
- ・「当社商品」をご使用の際には、(i)定格および性能に対し余裕のある「当社商品」のご利用、冗長設計などの安全設計、(ii)「当社商品」が故障しても、「お客様用途」の危険を最小にする安全設計、(iii)利用者に危険を知らせるための、安全対策のシステム全体としての構築、(iv)「当社商品」および「お客様用途」の定期的な保守、の各事項を実施してください。
- ・「当社」は DDoS 攻撃（分散型 DoS 攻撃）、コンピュータウイルスその他の技術的な有害プログラム、不正アクセスにより、「当社商品」、インストールされたソフトウェア、またはすべてのコンピュータ機器、コンピュータプログラム、ネットワーク、データベースが感染したとしても、そのことにより直接または間接的に生じた損失、損害その他の費用について一切責任を負わないものとします。

お客様ご自身にて、①アンチウイルス保護、②データ入出力、③紛失データの復元、④「当社商品」またはインストールされたソフトウェアに対するコンピュータウイルス感染防止、⑤「当社商品」に対する不正アクセス防止についての十分な措置を講じてください。

- 「当社商品」は、一般工業製品向けの汎用品として設計製造されています。従いまして、次に掲げる用途での使用は意図しておらず、お客様が「当社商品」をこれらの用途に使用される際には、「当社」は「当社商品」に対して一切保証をいたしません。ただし、次に掲げる用途であっても「当社」の意図した特別な商品用途の場合や特別の合意がある場合は除きます。
 - a) 高い安全性が必要とされる用途（例：原子力制御設備、燃焼設備、航空・宇宙設備、鉄道設備、昇降設備、娯楽設備、医用機器、安全装置、その他生命・身体に危険が及ぶ用途）
 - b) 高い信頼性が必要な用途（例：ガス・水道・電気等の供給システム、24 時間連続運転システム、決済システムほか権利・財産を取扱う用途など）
 - c) 厳しい条件または環境での用途（例：屋外に設置する設備、化学的汚染を被る設備、電磁的妨害を被る設備、振動・衝撃を受ける設備など）
 - d) 「カタログ等」に記載のない条件や環境での用途
- 上記の(a)から(d)に記載されている他、「本カタログ等記載の商品」は自動車（二輪車含む。以下同じ）向けではありません。自動車に搭載する用途には利用しないで下さい。自動車搭載用商品については当社営業担当者にご相談ください。

保証条件

「当社商品」の保証条件は次のとおりです。

- 保証期間 ご購入後 1 年間といたします。
（ただし「カタログ等」に別途記載がある場合を除きます。）
- 保証内容 故障した「当社商品」について、以下のいずれかを「当社」の任意の判断で実施します。
 - a) 当社保守サービス拠点における故障した「当社商品」の無償修理
（ただし、電子・機構部品については、修理対応は行いません。）
 - b) 故障した「当社商品」と同数の代替品の無償提供
- 保証対象外 故障の原因が次のいずれかに該当する場合は、保証いたしません。
 - a) 「当社商品」本来の使い方以外のご利用
 - b) 「利用条件等」から外れたご利用
 - c) 本ご承諾事項「ご利用にあたってのご注意」に反するご利用
 - d) 「当社」以外による改造、修理による場合
 - e) 「当社」以外の者によるソフトウェアプログラムによる場合
 - f) 「当社」からの出荷時の科学・技術の水準では予見できなかった原因
 - g) 上記のほか「当社」または「当社商品」以外の原因（天災等の不可抗力を含む）

責任の制限

本ご承諾事項に記載の保証が、「当社商品」に関する保証のすべてです。

「当社商品」に関連して生じた損害について、「当社」および「当社商品」の販売店は責任を負いません。

輸出管理

「当社商品」または技術資料を、輸出または非居住者に提供する場合は、安全保障貿易管理に関する日本および関係各国の法令・規制を遵守ください。お客様が法令・規則に違反する場合には、「当社商品」または技術資料をご提供できない場合があります。

安全上の注意

安全上の注意の内容については、以下のマニュアルを参照してください。詳細は、『V460-H ハンディ DPM コードリーダー ユーザーズマニュアル (SDNC-768)』の「安全上の注意」を参照してください。

安全上の要点

安全上の要点の内容については、以下のマニュアルを参照してください。詳細は、『V460-H ハンディ DPM コードリーダー ユーザーズマニュアル (SDNC-768)』の「安全上の要点」を参照してください。

使用上の注意

使用上の注意の内容については、以下のマニュアルを参照してください。詳細は、『V460-H ハンディ DPM コードリーダー ユーザーズマニュアル (SDNC-768)』の「使用上の注意」を参照してください。

法規と規格

法規と規格の内容については、以下のマニュアルを参照してください。詳細は、『V460-H ハンディ DPM コードリーダー ユーザーズマニュアル (SDNC-768)』の「法規と規格」を参照してください。

関連マニュアル

関連するマニュアルは、下表のとおりです。あわせてご覧ください。

マニュアル名称	Man.No.	形式	用途	内容
V460-H ハンディ DPM コードリーダー ユーザーズマニュアル	SDNC-768	V460-H ハンディ DPM コードリーダー	V460-H ハンディ DPM コードリーダー の、製品仕様や使用する ための基本設定を 知りたいとき	V460-H ハンディ DPM コードリーダーの仕様、クイックスタート、設定方法、コマンドパラメータについて記載しています。
V460-H ハンディ DPM コードリーダー ユーザーズマニュアル 通信設定編	SDNC-769		V460-H ハンディ DPM コードリーダー を、外部から制御する 方法を知りたいとき	V460-H ハンディ DPM コードリーダーを使用するためのシステム構成、制御方法、入出力仕様、ネットワーク種類、通信設定について記載しています。

マニュアル改訂履歴

マニュアル改訂記号は、表紙と裏表紙の下に記載されている Man. No. の末尾に付記されます。

Man.No. **SDNC-769A**

↑ 改訂記号

改訂記号	改訂年月	改訂理由・改訂ページ
A	2023 年 11 月	初版

通信仕様の概要

V460-H ハンディ DPM コードリーダーと外部装置を通信する前に必要な、通信仕様やコードリーダーの制御方法などについての基本的な概要を説明しています。

1-1	システム構成の確認.....	1-2
1-1-1	V460-H シリーズのシステム構成	1-2
1-2	外部装置との通信について	1-3
1-2-1	コードリーダー制御の基本動作	1-3
1-2-2	V460-H ハンディ DPM コードリーダーと通信可能な通信プロトコル	1-4

1-1 システム構成の確認

本製品は、撮像した 1 次元コード（バーコード）、2 次元コードを読み取り処理するマルチコードリーダーです。

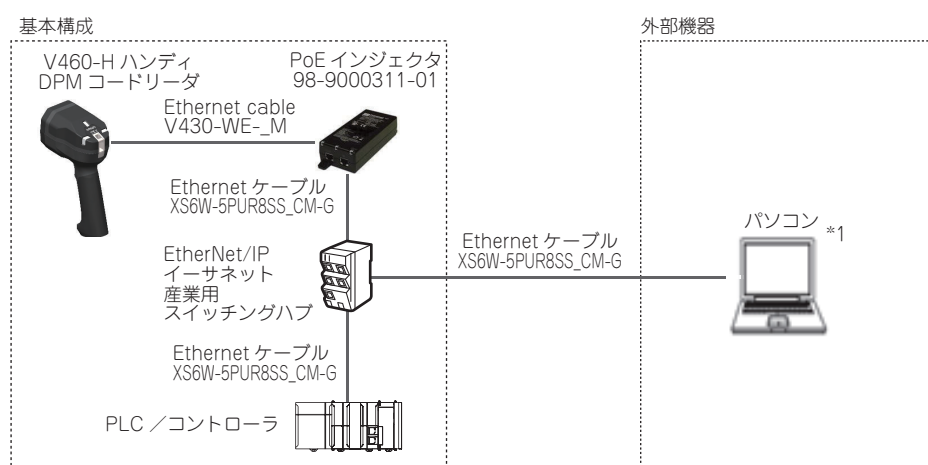
PLC やパソコンといった外部装置を接続して、外部装置からシリアルコマンドを入力したり、読み取り結果を出力することができます。

1-1-1 V460-H シリーズのシステム構成

V460-H を使用するためのシステム構成には以下の種類があります。

イーサネット（EtherNet/IP、シリアル（TCP）、PROFINET）で接続する

イーサネットケーブルにより、各種ネットワークに接続して、トリガや通信コマンドの入力、読み取り結果（判定結果、読み取り内容）の出力を行います。ネットワーク（シリアルは除く）ごとのデータリンク機能により、コードリーダーと外部装置間でデータ転送を周期的に行うことができます。



*1 モニタ表示が必要でない場合は、運用中にパソコンと接続する必要はありません。

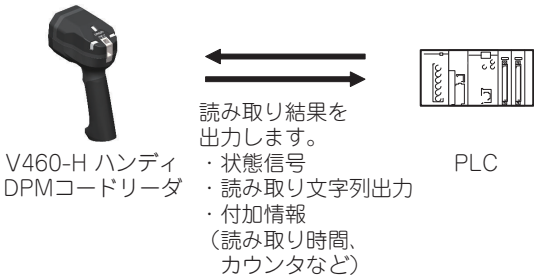
注: 一度リーダーを接続すると、イーサネットケーブル（PoE インジェクタから PLC / コントローラの間）が（両側で）抜かれると、リーダーはピープ音を鳴らしてユーザーに通知します。

1-2 外部装置との通信について

通信仕様、通信時の制御方法、外部装置と通信を開始するまでの設定などについて説明します。

1-2-1 コードリーダー制御の基本動作

外部装置とコードリーダーとの基本的な通信と、信号やデータの流れを説明します。



外部装置とコードリーダーとは、以下の方法でデータのやりとりを行うことができます。

外部装置からコードリーダーに入力できる命令

種類		内容
制御命令	通信コマンド入力	読み取り実行コマンド（トリガ）、マッチコードの有効化、カウンタのクリアなど、各種の命令が実行できます。 通信コマンドは、各通信プロトコルにより異なります。

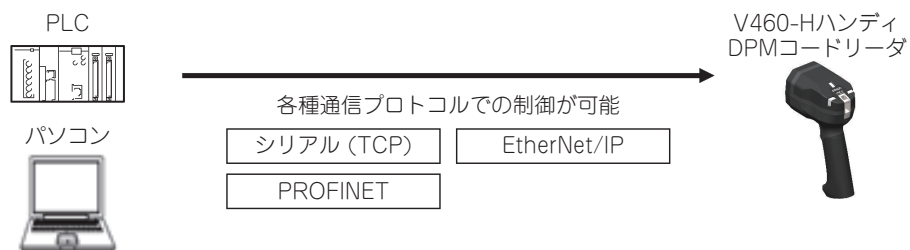
コードリーダーから外部装置に出力されるデータ

種類	内容
状態信号	制御信号や通信コマンド入力を、コードリーダーが確認して読み取り処理を開始すると、コードリーダーの状態（InReadCycle 信号など）、OK/NG の判定信号を外部装置に通知します。
読み取り文字列出力	バーコード、2次元コードから読み取りした文字列を出力できます。
付加情報	読み取り時間、コード位置座標などの付加情報を出力できます。 出力する項目は、あらかじめ WebLink _{HH} の詳細設定から該当項目を有効に設定しておく必要があります。

1-2-2 V460-H ハンディ DPM コードリーダーと通信可能な通信プロトコル

V460-H ハンディ DPM コードリーダーは、PLC やパソコンなどの外部装置からさまざまな通信プロトコルで制御できます。

外部装置から V460-H ハンディ DPM コードリーダー を制御できる通信プロトコルの種類には、以下があります。



使用できる通信プロトコルの種類

○: 使用可能 ×: 使用不可

通信方式	通信プロトコル	概要	通信ケーブル種類
			Ethernet
データ共有	EtherNet/IP	オープン通信プロトコルです。コードリーダとの通信では、タグデータリンクを使用します。PLC 側で、制御信号、コマンド／レスポンス、読み取りデータに対応した構造体型の変数を作成し、それをタグとしてタグデータリンクでの入出力を行うことで、PLC とコードリーダ間のデータのやりとりを行います。	○
	PROFINET	オープン通信プロトコルです。コードリーダとの通信では、ソフト・リアルタイム通信 (SRT) の RT (Real-time) を使用します。制御信号、コマンド／レスポンス、読み取りデータを格納するエリアを、PLC の I/O メモリに割り付け、サイクリックにデータ共有することで、PLC とコードリーダ間のデータのやりとりを行います。	○
フレーム伝送	シリアル (TCP)	特定のプロトコルを使用せず、コマンドフレームをコードリーダに発行し、レスポンスフレームをコードリーダから受信します。 ASCII 形式のデータを送受信することで、PLC や PC などの外部装置とコードリーダ間のデータのやりとりを行います。	○

同時に使用できる接続形態

○: 使用可能 ×: 使用不可 -: 対象外

コードリーダの接続形態	併用する接続方式		
	EtherNet/IP	PROFINET	シリアル (TCP)
EtherNet/IP	-	×	○
PROFINET	×	-	○
シリアル (TCP)	○	○	-



参考

ネットワークのルータ越え接続について

WebLink_{HH} は、ルータを越えて異なるネットワーク上にあるコードリーダに接続できます。

- コードリーダへの接続は、ブラウザからコードリーダの IP アドレスを直接指定して接続してください。
- 接続するコードリーダの IP アドレスは、固定 IP アドレスで設定してください。

イーサネット制御／出力する

2-1	EtherNet/IP で制御／出力する	2-2
2-1-1	EtherNet/IP の概要	2-2
2-1-2	EtherNet/IP 接続時のコードリーダの通信方法	2-4
2-1-3	PLC とコードリーダの通信の流れ	2-5
2-1-4	通信仕様を設定する (EtherNet/IP)	2-6
2-1-5	タグデータリンクの設定方法	2-8
2-1-6	入力・出力アセンブリ別 状態信号・制御信号一覧	2-10
2-1-7	アセンブリ別タイミングチャート	2-11
2-1-8	サンプルラダープログラム	2-13
2-1-9	NJ/NX シリーズコントローラで各通信エリアに変数でアクセスする 方法	2-14
2-1-10	EtherNet/IP のメッセージ通信でコントローラと通信する	2-25
2-2	シリアル (TCP) で制御／出力する	2-27
2-2-1	シリアル (TCP) の概要	2-27
2-2-2	通信処理の流れ	2-27
2-2-3	通信仕様を設定する (シリアル (TCP))	2-28
2-2-4	読み取り後に出力するデータの設定	2-30
2-2-5	外部からコードリーダを制御する	2-35
2-2-6	シリアルコマンド一覧	2-38

2-1 EtherNet/IP で制御／出力する

2-1-1 EtherNet/IP の概要

EtherNet/IP は、Ethernet を使用した産業用のマルチベンダネットワークです。その仕様はオープンな規格として、ODVA (OpenDeviceNet Vendor Association) にて管理されており、さまざまな産業用機器に採用されています。

EtherNet/IP は、標準の Ethernet 技術が使用されているため、さまざまな汎用 Ethernet 機器とも混在させて使用することができます。

EtherNet/IP には、主に以下のような特長があります。

- ・ **タグデータリンク（サイクリック通信）による高速・大容量なデータ交換**

EtherNet/IP 規格標準の Implicit 通信をサポートしており、これにより EtherNet/IP 機器とサイクリック通信（タグデータリンク）が可能です。

- ・ **ノード数に依存せず、アプリケーションごとに指定した周期でのタグデータリンク（サイクリック通信）が可能**

コネクションごとに設定した更新周期で回線をデータ交換するため、ノード数が増加しても通信上の更新周期は増加しません。（コネクション内のデータの同時性は確保されます。）

また、コネクションごとに更新周期を設定できるため、アプリケーションごとに最適な更新周期で通信が可能です。（例：工程間のインタロックは高速、生産指示や工程のステータスマニタは低速など）



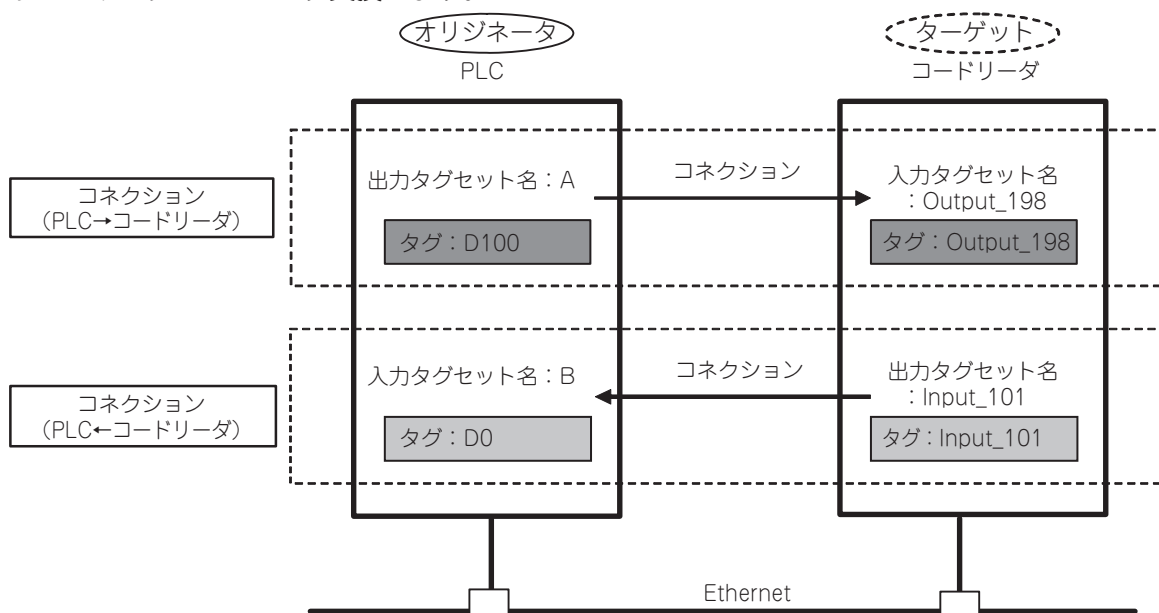
使用上の注意

多数の機器を接続しているネットワークでは、ネットワークに一時的に大きな負荷がかかった場合に、パフォーマンスの低下（レスポンスの遅延やパケットロス）や通信エラーを引き起こすことがあります。

実際に使用する条件で検証の上でご使用ください。

EtherNet/IP でのデータ交換のしくみ

EtherNet/IP ネットワーク上では、Ethernet 機器間のデータを、タグデータリンクによって以下のようにサイクリックにデータ交換します。



・データ交換方法

データ交換するために、2つのEtherNet/IP機器の間でコネクションを開きます。
一方のノードが、コネクションを要求（オープン）し、相手のノードとコネクションを開きます。
コネクションの開設を要求する側を「オリジネータ」、要求される側を「ターゲット」と呼びます。

・データ交換するエリア

コネクションでデータ交換するエリアを、タグとして指定します。
タグには、メモリエリアや変数を指定できます。
これらのタグをまとめたものを、それぞれ出力タグセット、入力タグセットと呼びます。



参考

タグデータリンク通信をサポートしていないPLCとEtherNet/IPで通信する場合には、タグデータリンクではなく、メッセージ通信機能を使用します。「2-1-10 EtherNet/IP のメッセージ通信でコントローラと通信する (P.2-25)」

2-1-2 EtherNet/IP 接続時のコードリーダの通信方法

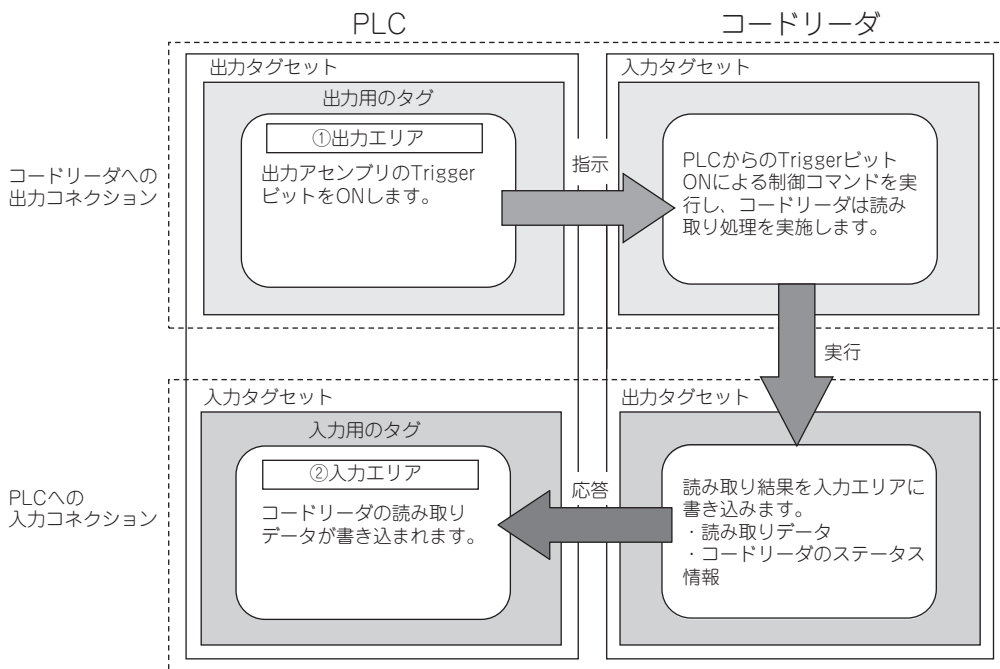
PLC とコードリーダを EtherNet/IP のタグデータリンクで通信して、コマンド／レスポンスによる PLC からの制御、および読み取り後のデータ出力が可能です。

オムロン製コントローラと接続し EtherNet/IP で通信する場合は、タグデータリンクの設定（タグ、タグセット、コネクション設定）を Sysmac Studio または Network Configurator で行います。タグデータリンク設定の詳細については、以下の各マニュアルを参照してください。

- ・『NJ/NX シリーズ CPU ユニット 内蔵 EtherNet/IP ポート ユーザーズマニュアル（SBCD-377）』
- ・『EtherNet/IP ユニット ユーザーズマニュアル（SBCD-342）』
- ・『CJ シリーズ EtherNet/IP ユニット ユーザーズマニュアル NJ シリーズ 接続編（SBCD-355）』

通信エリアの種類について

EtherNet/IP での通信では、PLC 側に入力エリア・出力エリアの 2 つの通信エリアを使用して通信を行います。本コードリーダには、入力エリアに 6 種類、出力エリアに 2 種類のアセンブリがあり、それぞれ 1 つを選択して使用します。



2-1-3 PLC とコードリーダーの通信の流れ

- (1) PLC（ユーザ）は、あらかじめ PLC のメモリエリア（出力エリア）に割り当てた Trigger ビットを OFF→ON にします。
- (2) PLC からの Trigger ビット ON により、コードリーダーは読み取り処理を実施します。
- (3) コードリーダーは、読み取り処理完了後、読み取りデータを指定された PLC のメモリエリア（入力エリア）に格納します。

【出力データ例】

デバイス名	名称	モジュール	変更	表示形式
new_Controller_0	EIOutput.COMMANDS.Run_Mode	False	TRUE FALSE	Boolean
new_Controller_0	EIOutput.COMMANDS.Trigger	True	TRUE FALSE	Boolean
new_Controller_0	EIInput			
	INFO_BITS	01		Hexadecimal
	DIAGNOSTIC_SEQUENCE_CO	00		Hexadecimal
	CONFIGURATION_SEQUENCI	00		Hexadecimal
	RESERVED	00		Hexadecimal
	▶ DEVICE_STATUS			
	FAULT	0		Decimal
	▶ COUNTERS[0-5]			
	▶ READ_CYCLE_REPORT[0-3]			
	▶ DECODE_CYCLE_REPORT[0-3]			
	DECODE_LENGTH	6		Decimal
	▼ DECODE_DATA[0-435]			
	DECODE_DATA[0]	2 (16#)		ASCII
	DECODE_DATA[1]	0 (16#)		ASCII
	DECODE_DATA[2]	3 (16#)		ASCII
	DECODE_DATA[3]	0 (16#)		ASCII
	DECODE_DATA[4]	4 (16#)		ASCII
	DECODE_DATA[5]	0 (16#)		ASCII
	DECODE_DATA[6]	. (16#)		ASCII
	DECODE_DATA[7]	. (16#)		ASCII

Run ModeビットとTriggerビットはTRUE
(通常、リーダーがトリガされる)

文字列はDECODE DATA STRINGエリアに出力される

2-1-4 通信仕様を設定する（EtherNet/IP）

WebLink_{HH} でコードリーダー本体のネットワーク設定をする

PLC などの外部装置のネットワークに合わせて、WebLink_{HH} からコードリーダーの IP アドレスを設定します。

- WebLink_{HH} - [セットアップ] 画面 - [歯車アイコン] - [詳細] - [通信設定] タブ - [イーサネット]

1 PLC などの外部装置のネットワークに合わせて、[IP アドレス]、[サブネットマスク] を設定します。

設定項目	設定値	説明
IP アドレス	a.b.c.d a : 0～255 b : 0～255 c : 0～255 d : 0～255 (初期値 : 192.168.188.2)	コードリーダーの IP アドレスを入力します。
サブネット	a.b.c.d a : 0～255 b : 0～255 c : 0～255 d : 0～255 (初期値 : 255.255.0.0)	サブネットマスクのアドレスを入力します。
ゲートウェイ	a.b.c.d a : 0～255 b : 0～255 c : 0～255 d : 0～255 (初期値 : 0.0.0.0)	ゲートウェイを使用する場合は、ゲートウェイの IP アドレスを入力します。使用しない場合は初期値の 0.0.0.0 にしてください。
IP アドレスモード	- 固定 (初期値) - DHCP	固定モードでは、コードリーダーはユーザ定義の IP アドレスを使用します。 DHCP モードでは、コードリーダーは DHCP サーバから IP アドレス、サブネット、およびゲートウェイを取得します。 PLC 通信の場合、[固定モード] が必須です。DHCP モードを有効にしないでください。

WebLink_{HH} で EtherNet/IP 通信の設定をする

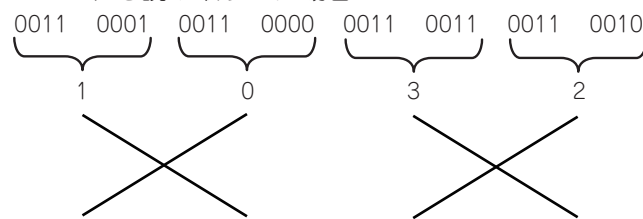
- WebLink_{HH} - [セットアップ] 画面 - [歯車アイコン] - [詳細] - [通信設定] タブ - [イーサネット]

設定項目	設定値	説明
EtherNet/IP	- 有効 - 無効	- 有効：コードリーダーの EtherNet/IP 接続を有効にします。 - 無効：コードリーダーの EtherNet/IP 接続を無効にします。
EtherNet/IP バイトスワップ	- 有効 - 無効	- 有効：読み取りデータのバイトスワップを有効にします。読み取りデータは、リトルエンディアンで Decode Data に格納されます。読み取りデータと CPU アーキテクチャのエンディアンが異なるときに使用します。 - 無効：読み取りデータのバイトスワップを無効にします。読み取りデータは、ビッグエンディアンで Decode Data に格納されます。

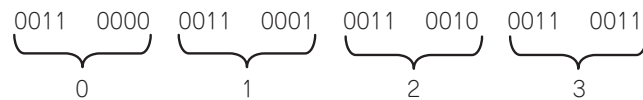
バイトスワップの設定により、出力が以下のように変わります。

例：文字列「0123」のコードを読み取りした場合

バイトスワップ有効



バイトスワップ無効



(注) バイトスワップは、EtherNet/IP の入力アセンブリ Decode Data String (読み取り文字列) のメモリ領域にのみ適用されます。

2-1-5 タグデータリンクの設定方法

EtherNet/IP でデータリンクするための設定方法について説明します。

コードリーダとデータリンクする PLC 側の通信エリアは、タグ（タグセット）として指定して、タグデータリンク通信するためのコネクションを設定します。



使用上の注意

NJ/NX シリーズまたは CJ シリーズの CPU ユニットとコードリーダを接続する場合は、コードリーダのコネクション情報を定義した EDS ファイルを Sysmac Studio にインストールしてください。

EDS ファイルは、当社 HP (www.fa.omron.co.jp) からダウンロードしてください。

タグ、タグセット、コネクションの設定について

コードリーダには、6 種類の入力アセンブリ、2 種類の出力アセンブリがあり、1 つずつ選択して使用します。また、選択したアセンブリによって、データ構造が変化します。

メモリ割り付けと各アセンブリのデータ構造の詳細な説明については、「A-2 EtherNet/IP 通信仕様詳細 (P.A-3)」を参照ください。

アセンブリ一覧

アセンブリ名	コネクション I/O タイプ	入力／出力	ターゲット変数	サイズ (byte)	使用出力	アセンブリ説明	データ構造詳細
小型入力	IO small	入力	100	84	198	小型、軽量な入力アセンブリです。 64byte の読み取りデータを格納できます。	*1
大型入力	IO large	入力	101	176	198	小型入力アセンブリより多くのデバイスステータス情報が確認できます。 128byte の読み取りデータを格納できます。	
MXL/SLC 入力	Input MXLSLC	入力	102	258	197	大型入力アセンブリより詳細なデバイスステータス情報が確認できます。 184byte の読み取りデータを格納できます。	
1 デコード入力	Input 1 Decode	入力	103	500	197	436byte の読み取りデータを格納できます	
4 デコード入力	Input 4 Decode	入力	104	500	197	4 つのシンボル情報と読み取りデータを格納できるよう設計されています。 1 つ目の読み取りデータは 160byte のエリアに格納され、2～4 つ目の読み取りデータは 72byte のエリアに格納されます。	
N デコード入力	Input N Decode	入力	105	500	197	任意の数のシンボル情報と読み取りデータを格納できるよう設計されています。 456byte の読み取りデータを格納できます。	
出力	-	出力	197	4	-	コードリーダに対しコマンドを送信することができます。	
出力（レガシー）	-	出力	198	12	-	コードリーダに対し、固定データのエコーバックやコマンドを送信することができます。	

*1. 「A-2 EtherNet/IP 通信仕様詳細 (P.A-3)」を参照してください。

タグセット設定

設定項目	設定内容
入力	
タグセット名	PLC 側のタグセット名
サイズ	入力アセンブリの設定に依存 • 84、176、248、500 byte
出力	
タグセット名	PLC 側のタグセット名
サイズ	出力アセンブリの設定に依存 • 4、12byte

コネクション設定

設定項目	設定内容
入力	
ターゲット変数	入力アセンブリの設定に依存 • 100、101、102、103、104、105
サイズ	入力アセンブリの設定に依存 • 84、176、248、500 byte
オリジネータ変数	PLC 側で定義した変数
サイズ	入力アセンブリの設定に依存 • 4、12 byte
コネクションタイプ	Point to Point connection
RPI	4.0～65.0ms（初期値：10.0ms）
タイムアウト	RPI × (4～512)（初期値：RPI × 512）
出力	
ターゲット変数	出力アセンブリの設定に依存 • 197、198
サイズ	出力アセンブリの設定に依存 • 4、12byte
オリジネータ変数	PLC 側で定義した変数
サイズ	出力アセンブリの設定に依存 • 4、12byte
コネクションタイプ	Point to Point connection



使用上の注意

- 各通信エリアに I/O メモリアドレスを指定する場合、メモリ保持する CIO メモリエリアを指定しないと、PLC の動作モードが変更された際に、各通信エリアの情報がクリアされます。
- EDS ファイルを使用しないでインスタンスを指定する場合、以下の AssemblyObject の設定が必要です。

AssemblyObject の設定

設定項目	設定値	備考
インスタンス ID	100	小型入力
	101	大型入力
	102	MXL/SLC 入力
	103	1 デコード入力
	104	4 デコード入力
	105	N デコード入力
	197	出力
	198	出力（レガシー）

2-1-6 入力・出力アセンブリ別 状態信号・制御信号一覧

本コードリーダの入力アセンブリには以下の種類があります。

- (1) 小型入力
- (2) 大型入力
- (3) MXL/SLC 入力
- (4) 1 デコード入力
- (5) 4 デコード入力
- (6) N デコード入力

状態信号は以下のとおりです。

コードリーダの状況に応じて自動的に制御される信号です。

○：確認可能 ×：確認不可

状態信号	説明	(1)	(2)	(3)	(4)	(5)	(6)
InReadCycle	読み取りサイクル中のとき、このビットは「1」になります。	×	○	×	×	×	×
Trigger Acknowledged	出力アセンブリの Trigger ビットを受け付けると、このビットが「1」になります。 Trigger ビットが OFF にされたとき、Trigger Acknowledged も「0」になります。	×	×	○	○	○	○
Exposure Done	露光前、または露光が完了したとき、このビットは「1」になります。 露光中である時、このビットは「0」になります。	×	×	○	○	○	○
Decoding	画像を処理しているとき、このビットは「1」になります。 画像処理を完了すると、このビットは「0」になります。	×	×	○	○	○	○
Data is Ready	Read Cycle Report と Data Cycle Report のデータが確定した時、このビットは「1」になります。 次の読み取りが開始されるとこのビットは「0」になります。	×	×	○	○	○	○
Read Cycle Pass	読み取り成功時（マッチコード有効のときは一致時）に、このビットは「1」になります。 次の読み取りが開始されるとこのビットは「0」になります。	×	×	○	○	○	○
Read Cycle Fail	読み取り失敗時（マッチコード有効のときは不一致時）に、このビットは「1」になります。 次の読み取りが開始されるとこのビットは「0」になります。	×	×	○	○	○	○
Decode Data	読み取り文字列が格納されるエリアです。印字品質評価指標などの付加情報を設定している場合は、読み取り文字列に続いて格納されます。	○	○	○	○	○	○

本コードリーダの出力アセンブリには以下の種類があります。

- (1) 出力アセンブリ
- (2) 出力アセンブリ（レガシー）

制御信号は以下のとおりです。

ユーザにより、任意のタイミングで制御することができます。

○：確認可能 ×：確認不可

制御信号	説明	(1)	(2)
Trigger	読み取りを実行します。コードリーダはこのビットの 0→1 をトリガの立ち上がりエッジ、1→0 をトリガの立ち下がりエッジと認識します。	○	○

制御信号	説明	(1)	(2)
New Master	このビットを ON すると、次の読み取り結果をマスタシンボルとして登録します。	○	○

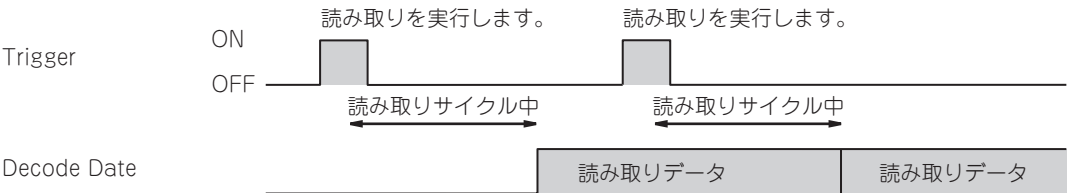
2-1-7 アセンブリ別タイミングチャート

読み取り実行信号（TRIG 信号）による読み取り実行

読み取りデータを、PLC のデータメモリに格納完了した時のタイミング信号は、入力アセンブリによって異なります。

● 小型入力 (100)

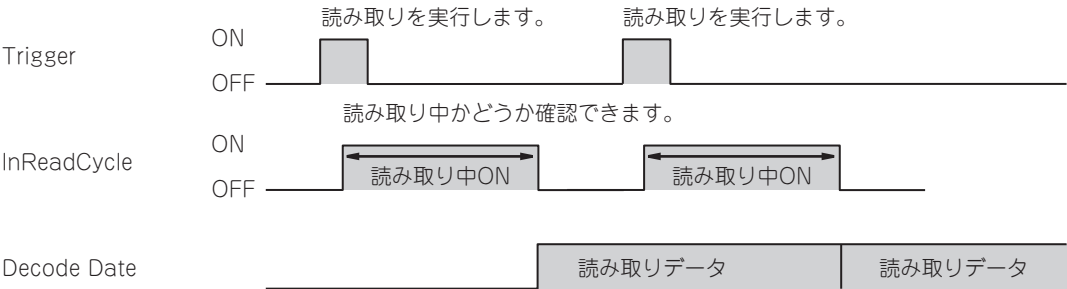
読み取りデータ格納のタイミング信号に該当するものではありません。



- (1) Trigger の立ち上がりエッジで読み取りを開始します。
- (2) 読み取り終了時、読み取りデータが Decode Data に格納されます。

● 大型入力 (101)

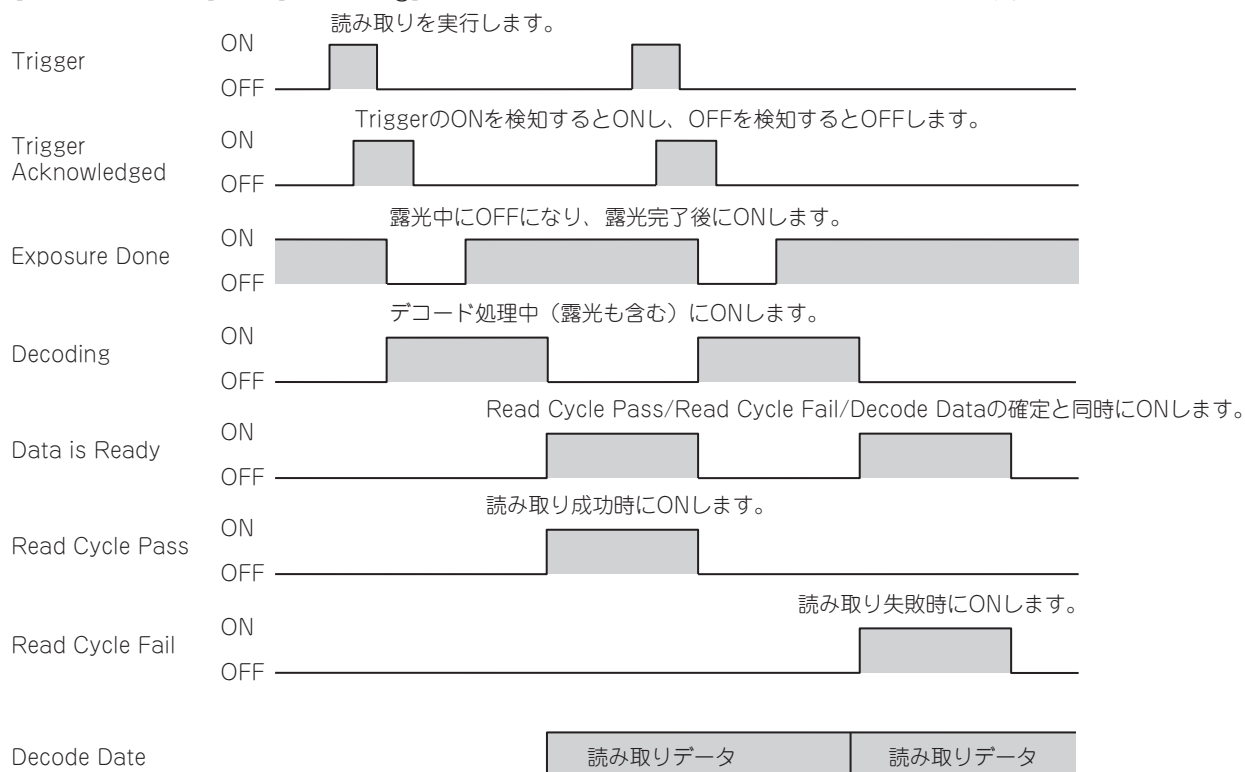
[Device Status] の [InReadCycle] ビットが、ON→OFF になったタイミングで出力しています。



- (1) Trigger の立ち上がりエッジで読み取りを開始します。
- (2) 読み取り開始で InReadCycle が ON になり、Trigger が OFF になります。
- (3) 読み取り終了時、読み取りデータが Decode Data に格納され、同時に InReadCycle が OFF します。

● MXL/SLC 入力 (102)から N デコード入力 (105)

[Device Status] の [Decoding] ビットが、ON→OFF になったタイミングで出力しています。



- (1) Trigger の立ち上がりエッジで読み取りを開始します。
- (2) Trigger Acknowledged は、Trigger の ON を検知すると ON し、Trigger の OFF を検知すると OFF します。
- (3) Exposure Done は、露光処理が開始されると OFF になり、露光完了後に ON になります。
- (4) Decoding は、デコード処理中に ON になります。デコード処理は露光処理と重複します。
- (5) Data is Ready は、Decode Data / Read Cycle Pass / Read Cycle Fail の確定と同時に ON します。
- (6) 読み取り成功時に Read Cycle Pass が ON、読み取り失敗時に Read Cycle Fail が ON になります。また、読み取りデータが Decode Data に格納されます。



参考

出力タイミングは、10ms ほど差があることがあります。

- (7) Data is Ready は、次の Trigger を検知したら OFF します。

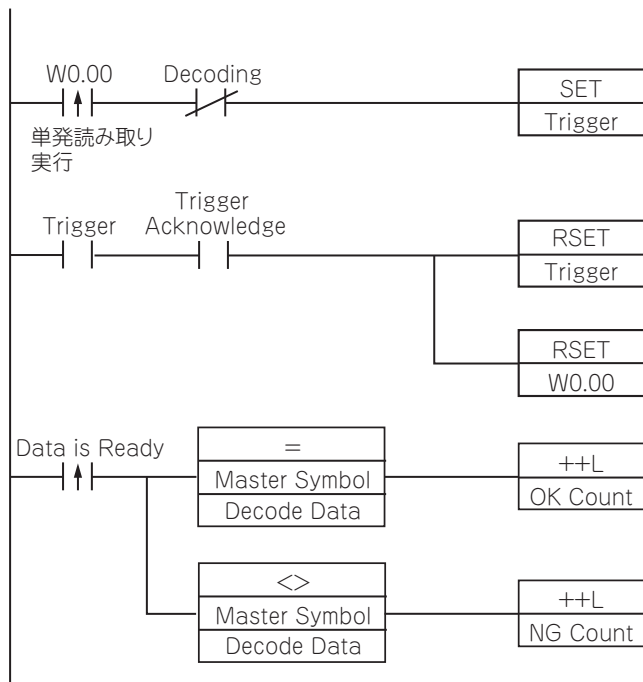
2-1-8 サンプルラダープログラム

以下を実行するサンプルラダープログラムです。

- Trigger 信号を入力して、単発読み取りを実施します。
- 読み取った文字列（Decode Data）と PLC に格納した照合用文字列（Master Symbol）を比較します。
- 一致したら OK、不一致であれば NG とカウントします。

以下の入力・出力アセンブリを使用しています。

- 入力アセンブリ：MXL/SLC 入力 (102)
- 出力アセンブリ：出力 (197)



- (1) 単発読み取り実行フラグが ON になった時、Trigger ビットを ON にします。
- (2) トリガ入力を検知した Trigger Acknowledged ビットが ON になります。
- (3) Trigger Acknowledged ビットの ON を検知したら、Trigger ビットを OFF にします。
- (4) 読み取りが完了し、Data is Ready ビットが ON になります。
- (5) 照合用文字列（Master Symbol）と読み取り文字列（Decode Data）を比較します。
- (6) 2 つの文字列が一致した場合は、OK カウントを+1 します。
- (7) 2 つの文字列が不一致である場合は、NG カウントを+1 します。

2-1-9 NJ/NX シリーズコントローラで各通信エリアに変数でアクセスする方法

NJ/NX シリーズコントローラでは、各通信エリアに割り付けた I/O メモリにユーザプログラムからアクセスするには、変数経由でのみ行うことができます。
ここでは、MXL/SLC 入力 (102) と出力 (197) を使用した例を記載します。
各アセンブリのデータ構造の詳細な説明については、「A-2 EtherNet/IP 通信仕様詳細 (P.A-3)」を参照ください。

ネットワーク変数でアクセスする

センサの各通信用エリアの構造に合わせて、変数をカスタマイズして定義します。
変数の定義は、Sysmac Studio で行います。
Sysmac Studio の操作については、『Sysmac Studio Version1 オペレーティングマニュアル (SBCA-470)』を参照してください。

1 変数のデータ型を定義する

各通信用エリアの構造に合わせて、変数のデータ型を定義します。

1) 制御信号アクセス用のデータ型の定義

まず、制御信号・状態信号にアクセスするための BOOL 型配列のデータ型を定義します。
ここでは、「COMMAND」、「Device_Status」というデータ型を定義します。

制御信号

データ型の名称	データ型
COMMAND	ARRAY[0..31] OF BOOL
Run_Mode	BOOL
Trigger	BOOL
Enable_Matchcode	BOOL
...	
Reserved	ARRAY[0..21] OF BOOL

状態信号

データ型の名称	データ型
Device_Status	ARRAY[0..31] OF BOOL
Run Mode	BOOL
Trigger_Acknowledged	BOOL
Exposure_Done	BOOL
...	
Buffer_Overflow	BOOL
Reserved	ARRAY[0..9] OF BOOL

2) 各通信エリアアクセス用のデータ型の定義

通信エリアにアクセスするためのデータ型を、出力エリアと入力エリアごとに定義します。
ここでは、「S_EIOutput197」と「S_EIInput102」という 2 種類のデータ型を定義します。

- 出力エリアへのアクセス用のデータ型
 - データ型の名称：S_EIOutput197
 - 派生データ型の種類：構造体型

データ型の名称	データ型
S_EIOutput197	STRUCT
COMMANDS	COMMAND

出力エリアに合わせた変数のデータ型の割り当て例：

	ビット															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
+0	Reserved			*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	TRIG	*1
+1	Reserved															

*1. TRIG 以外のビットについては、「出力（インスタンス ID: 197）（P.A-23）」を参照ください。

- 入力エリアへのアクセス用のデータ型
 - データ型の名称：S_EIPIInput102
 - 派生データ型の種類：構造体型

データ型の名称	データ型
S_EIPIInput102	STRUCT
RESERVED	BYTE
RESERVED	BYTE
RESERVED	BYTE
RESERVED	BYTE
DEVICE_STATUS	Device Status
RESERVED	DINT
COUNTERS	ARRAY[0..5] OF DINT
READ_CYCLE_REPORT	ARRAY[0..3] OF INT
DECODE_CYCLE_REPORT	ARRAY[0..3] OF DINT
DECODE_LENGTH	DINT
DECODE_DATA	ARRAY[0..183] OF BYTE

入力エリアに合わせた変数のデータ型の割り当て例：

	ビット															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
+0	RESERVED								INFO_BITS							
+1	RESERVED								RESERVED							
+2	コードリーダーの各信号状態情報（Device_Status）															
+3																
+4	RESERVED															
+5																
+6	読み取りカウンタ情報（COUNTERS）															
...																
...																
+11																
+12	読み取りサイクル情報（READ CYCLE REPORT）															
+13																
+14																
+15																
+16	読み取りデータの文字数情報（DECODE LENGTH）															
+17																
+18	読み取りデータ情報（DECODE DATA）															
...																
...																
+89																

2 変数を定義する

EtherNet/IP 通信で、各通信エリアのデータをデータリンクするための変数を定義します。

この変数には、手順 1 で定義したデータ型を使用します。

変数	変数の種類	ネットワーク公開	データ型	用途
EIOutput	グローバル変数	出力	S_EIOutput197	出力エリアのデータリンク用
EIInput	グローバル変数	入力	S_EIInput102	入力エリアのデータリンク用

3 ユーザプログラムからのアクセス

センサの各通信エリアへのアクセスは、定義した変数を使って、以下のように指定します。

出力エリア

信号名	変数名称
Trigger	EIOutput.COMMANDS.Trigger

入力エリア

信号名	変数名称
Run Mode	EIInput.DEVICE_STATUS.Run Mode
Trigger_Acknowledged	EIInput.DEVICE_STATUS.Trigger_Acknowledged
Decoding	EIInput.DEVICE_STATUS.Decoding
DatalsReady	EIInput.DEVICE_STATUS.DatalsReady
Decode_Data	EIInput.DECODE_DATA

コマンド制御の実行例

ここでは、コードリーダーと PLC の EtherNet/IP 通信におけるコマンド制御の実行例を記載します。

● コードを読み取りして、読み取り文字列を PLC に格納する

<タグセット・コネクション設定例>

- ・ 入力アセンブリ：MXL/SXL 入力 (102)
- ・ 出力アセンブリ：出力 (197)

▼ コネクション										
コネクション数/使用可能数: 2 / 32										
ターゲットデバイス	コネクション名	コネクション/タイプ	入力/出力	ターゲット変数	サイズ[Byte]	オリジネータ変数	サイズ[Byte]	コネクションタイプ	IRPI[ms]	タイムアウト値
192.168.188.2 V460-H Rev.1	default_001	Input MXLSLC	入力	102	248	EIInput	248	Point to Point connection	50.0	RPI x 4
			出力	197	4	EIOutput	4	Point to Point connection		

▼ タグセット

タグセット数/使用可能数: 2 / 32 タグ数/使用可能数: 2 / 256

入力 出力

	タグセット名	Bit選択	サイズ (Byte)	サイズ (Bit)	インスタンスID	コントローラステータス
▼	EIInput	☐	248		Auto	含めない
	EIInput	☐	248	0		

▼ タグセット

タグセット数/使用可能数: 2 / 32 タグ数/使用可能数: 2 / 256

入力 [ヒラ](#)

	タグセット名	Bit選択	サイズ (Byte)	サイズ (Bit)	インスタンスID	コントローラステータス(運転停止異常時)
▼	EIOutput	☐	4		Auto	含めない
	EIOutput	☐	4	0		クリア

<変数設定例>

グループフィルタ

グループ設定なし

	名称	データ型	初期値	割付先	保持	コンスタント	ネットワーク公開	コメント
	EIInput	S_EIInput102			☐	☐	入力	
	EIOutput	S_EIOutput197			☐	☐	出力	

<入力アセンブリ構造体(102)>

root	名称	データ型	オフセット種別	バイトオフセット	ビットオフセット	コメント
編成体型	Device_status_102	STRUCT	任意			
共用体型	Run_mode	BOOL		0	0	
列挙型	Trigger_Acknowledge	BOOL		0	1	
	Exposure_Done	BOOL		0	2	
	Decoding	BOOL		0	3	
	Data_Is_Ready	BOOL		0	4	
	Read_Cycle_Pass	BOOL		0	5	
	Read_Cycle_Fail	BOOL		0	6	
	General_Fault	BOOL		0	7	
	New_match_code_acknowledged	BOOL		1	0	
	Match_Code_Enabled	BOOL		1	1	
	Image_Sensor_Calibrating	BOOL		1	2	
	Image_Sensor_Calibration_Complete	BOOL		1	3	
	Training	BOOL		1	4	
	Training_Complete	BOOL		1	5	
	Optimizing	BOOL		1	6	
	Optimization_Complete	BOOL		1	7	
	AutoImage_Photometry_Enabled	BOOL		2	0	
	AutoImage_Photometry_Complete	BOOL		2	1	
	Output1_Status	BOOL		2	2	
	Output2_Status	BOOL		2	3	
	Output3_Status	BOOL		2	4	
	Buffer_Overflow	BOOL		2	5	
	Reserved	ARRAY[0..9] OF BOOL		2	6	
	S_EIPInput102	STRUCT	NJ			
	INFO_BITS	BYTE				
	RESERVED	BYTE				
	CONFIGURATON_CHANGE_DETEC...	BYTE				
	RESERVED2	BYTE				
	DEVICE_STATUS	Device_status_102				
	FAULT_CODE	DINT				
	COUNTERS	ARRAY[0..5] OF DINT				
	READ_CYCLE_REPORT	DECODE_CYCLE_REP...				
	DECODE_CYCLE_REPORT	ARRAY[0..3] OF UINT				
	CODE_TYPE	DINT				
	PIXELS_PER_ELEMENT	REAL				
	DECODE_DATA_LENGTH	DINT				
	DECODE_DATA_STRING	ARRAY[0..183] OF BY...				
	DECODE_CYCLE_REPORT_102	STRUCT	NJ			
	Capture_Time	INT				
	Total_Decode_Time	INT				
	Total_ReadCycle_Time	INT				
	Reserved	INT				
	S_EIOutput197	STRUCT	NJ			
	COMMANDS	COMMAND_197				
	COMMAND_197	STRUCT	任意			
	Run_Mode	BOOL		0	0	
	Trigger	BOOL		0	1	
	Enable_MatchCode	BOOL		0	2	
	Reset_General_Fault	BOOL		0	3	
	Clear_No_Read_ReadCycle_Count	BOOL		0	4	
	Clear_MisMatch_ReadCycle_Count	BOOL		0	5	
	Clear_No_Read_Count	BOOL		0	6	
	Clear_Trigger_Count	BOOL		0	7	
	Clear_Matchcode_Count	BOOL		1	0	
	Clear_MisMatch_Count	BOOL		1	1	
	Output_1	BOOL		1	2	
	Output_2	BOOL		1	3	
	Output_3	BOOL		1	4	
	Reserved	ARRAY[0..17] OF BOOL		1	5	

<出力アセンブリ構造体(197)>

名称	データ型	オフセット	種別	バイトオフセット	ビットオフセット	コメント
COMMANDS	COMMAND_197					
COMMAND_197	STRUCT	任意				
Run_Mode	BOOL		0	0		
Trigger	BOOL		0	1		
Enable_MatchCode	BOOL		0	2		
Reset_General_Fault	BOOL		0	3		
Clear_No_Read_ReadCycle_Count	BOOL		0	4		
Clear_MisMatch_ReadCycle_Count	BOOL		0	5		
Clear_No_Read_Count	BOOL		0	6		
Clear_Trigger_Count	BOOL		0	7		
Clear_Matchcode_Count	BOOL		1	0		
Clear_MisMatch_Count	BOOL		1	1		
Output_1	BOOL		1	2		
Output_2	BOOL		1	3		
Output_3	BOOL		1	4		
Reserved	ARRAY[0..17] OF BOOL		1	5		

<データ格納例>

- 読み取りしたデータ：「1234567890」

new_Controller.0

名称	モジュール	変更	表示形式
Run_Mode	False	TRUE FALSE	Boolean
Trigger	True	TRUE FALSE	Boolean
Enable_MatchCode	False	TRUE FALSE	Boolean
Reset_General_Fault	False	TRUE FALSE	Boolean
Clear_No_Read_ReadCycle_Count	False	TRUE FALSE	Boolean
Clear_MisMatch_ReadCycle_Count	False	TRUE FALSE	Boolean
Clear_No_Read_Count	False	TRUE FALSE	Boolean
Clear_Trigger_Count	False	TRUE FALSE	Boolean
Clear_Matchcode_Count	False	TRUE FALSE	Boolean
Clear_MisMatch_Count	False	TRUE FALSE	Boolean
Output_1	False	TRUE FALSE	Boolean
Output_2	False	TRUE FALSE	Boolean
Output_3	False	TRUE FALSE	Boolean
Reserved[0..17]			
new_Controller.0			
ElPinut			
INFO_BITS	01		Hexadecimal
RESERVED	00		Hexadecimal
CONFIGURATON_CHANGE_DETECTION	00		Hexadecimal
RESERVED2	00		Hexadecimal
DEVICE_STATUS			
FAULT_CODE	0		Decimal
COUNTERS[0..5]			
READ_CYCLE_REPORT			
DECODE_CYCLE_REPORT[0..3]			
CODE_TYPE	2048		Decimal
PIXELS_PER_ELEMENT	16.5		Real
DECODE_DATA_LENGTH	10		Decimal
DECODE_DATA_STRING[0..183]			
DECODE_DATA_STRING[0]	1 (16#31)		ASCII
DECODE_DATA_STRING[1]	2 (16#32)		ASCII
DECODE_DATA_STRING[2]	3 (16#33)		ASCII
DECODE_DATA_STRING[3]	4 (16#34)		ASCII
DECODE_DATA_STRING[4]	5 (16#35)		ASCII
DECODE_DATA_STRING[5]	6 (16#36)		ASCII
DECODE_DATA_STRING[6]	7 (16#37)		ASCII
DECODE_DATA_STRING[7]	8 (16#38)		ASCII
DECODE_DATA_STRING[8]	9 (16#39)		ASCII
DECODE_DATA_STRING[9]	0 (16#30)		ASCII

omron

Data Matrix

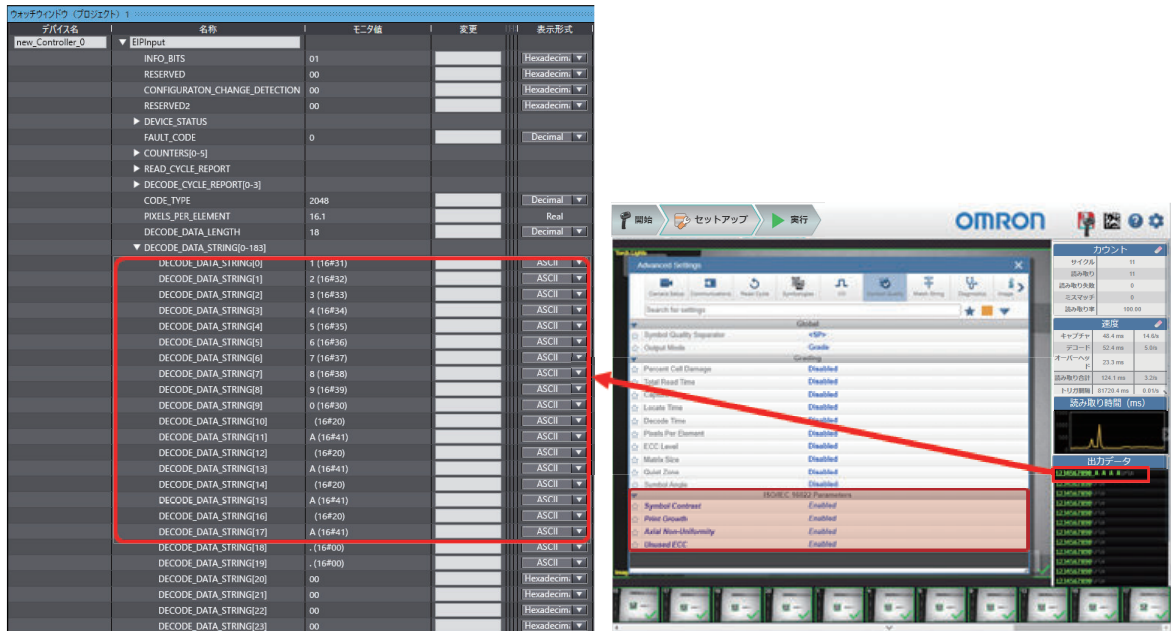
1234567890

読み取りデータ

項目	値
サイクル	11
読み取り	11
読み取り失敗	0
読み取りエラー	0
読み取り時間 (ms)	100.00

<ISO/IEC 16022 パラメータ有効時のデータ格納例>

- 読み取りしたデータ：「1234567890 A A A A」



● 4つのコードを読み取りして、読み取り文字列を PLC に格納する

<タグセット・コネクション設定例>

- ・ 入力アセンブリ：4 デコード入力 (104)
- ・ 出力アセンブリ：出力 (197)

▼ コネクション							
コネクション数/使用可能数: 2 / 32							
ターゲットデバイス	コネクション名	コネクションタイプ	入力/出力	ターゲット変数	サイズ(Byte)	オリジネータ変数	サイズ(Byte)
192.168.188.2 V460-H Rev 1	default_001	Input 4 Decode	入力	104	500	EIPInput	500
			出力	197	4	EIPOutput	4

▼ タグセット						
タグセット数/使用可能数: 2 / 32 タグ数/使用可能数: 2 / 256						
出力						
I	タグセット名	Bit選択	サイズ (Byte)	サイズ (Bit)	インスタンスID	コントロールステータス
▼	EIPInput	☐	500		Auto	含めない
	EIPInput	☐	500	0		

▼ タグセット						
タグセット数/使用可能数: 2 / 32 タグ数/使用可能数: 2 / 256						
入力						
I	タグセット名	Bit選択	サイズ (Byte)	サイズ (Bit)	インスタンスID	コントロールステータス
▼	EIPOutput	☐	4		Auto	含めない
	EIPOutput	☐	4	0		クリア

<変数設定例>

グループフィルタ (グループ設定なし)								
	名称	データ型	初期値	割付先	保持	コンスタント	ネットワーク公開	コメント
■	EIPInput	S_EIPInput104			☐	☐	入力	
■	EIPOutput	S_EIPOutput197			☐	☐	出力	

<入力アセンブリ構造体(104)>

root	名称	データ型	オフセット種別	バイトオフセット	ビットオフセット	コメント
Device_status_104	STRUCT	任意				
Run_mode	BOOL		0	0		
Trigger_Acknowledge	BOOL		0	1		
Exposure_Done	BOOL		0	2		
Decoding	BOOL		0	3		
Data_Is_Ready	BOOL		0	4		
Read_Cycle_Pass	BOOL		0	5		
Read_Cycle_Fail	BOOL		0	6		
General_Fault	BOOL		0	7		
New_match_code_acknowledged	BOOL		1	0		
Match_Code_Enabled	BOOL		1	1		
Image_Sensor_Calibrating	BOOL		1	2		
Image_Sensor_Calibration_Complete	BOOL		1	3		
Training	BOOL		1	4		
Training_Complete	BOOL		1	5		
Optimizing	BOOL		1	6		
Optimization_Complete	BOOL		1	7		
AutoImage_Photometry_Enabled	BOOL		2	0		
AutoImage_Photometry_Complete	BOOL		2	1		
Output1_Status	BOOL		2	2		
Output2_Status	BOOL		2	3		
Buffer_Overflow	BOOL		2	4		
Reserved	ARRAY[0..10] OF BOOL		2	5		
S_EIPInput104	STRUCT	NJ				
INFO_BITS	BYTE					
RESERVED1	BYTE					
RESERVED2	BYTE					
RESERVED3	BYTE					
DEVICE_STATUS	Device_status_104					
FAULT_CODE	DINT					
COUNTERS	ARRAY[0..5] OF UDINT					
READ_CYCLE_REPORT	ARRAY[0..3] OF UINT					
DECODE_1_CYCLE_REPORT	ARRAY[0..3] OF UINT					
CODE_TYPE	DINT					
PIXELS_PER_ELEMENT	REAL					
DECODE_1_LENGTH	DINT					
DECODE_1_DATA	ARRAY[0..159] OF BY...					
DECODE_2_CYCLE_REPORT	ARRAY[0..3] OF UINT					
CODE_TYPE_2	DINT					
PIXELS_PER_ELEMENT_2	REAL					
DECODE_2_LENGTH	DINT					
DECODE_2_DATA	ARRAY[0..71] OF BYTE					
DECODE_3_CYCLE_REPORT	ARRAY[0..3] OF UINT					
CODE_TYPE_3	DINT					
PIXELS_PER_ELEMENT_3	REAL					
DECODE_3_LENGTH	DINT					
DECODE_3_DATA	ARRAY[0..71] OF BYTE					
DECODE_4_CYCLE_REPORT	ARRAY[0..3] OF UINT					
CODE_TYPE_4	DINT					
PIXELS_PER_ELEMENT_4	REAL					
DECODE_4_LENGTH	DINT					
DECODE_4_DATA	ARRAY[0..71] OF BYTE					

<出力アセンブリ構造体(197)>

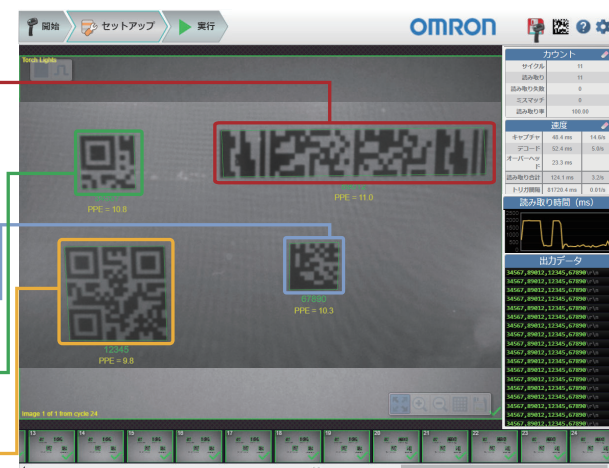
root	名称	データ型	オフセット種別	バイトオフセット	ビットオフセット	コメント
S_EIPOutput197	STRUCT	任意				
Run_Mode	BOOL		0	0		
Trigger	BOOL		0	1		
Enable_MatchCode	BOOL		0	2		
Reset_General_Fault	BOOL		0	3		
Clear_No_Read_ReadCycle_Count	BOOL		0	4		
Clear_MisMatch_ReadCycle_Count	BOOL		0	5		
Clear_No_Read_Count	BOOL		0	6		
Clear_Trigger_Count	BOOL		0	7		
Clear_Matchcode_Count	BOOL		1	0		
Clear_MisMatch_Count	BOOL		1	1		
Output_1	BOOL		1	2		
Output_2	BOOL		1	3		
Output_3	BOOL		1	4		
Reserved	ARRAY[0..17] OF BOOL		1	5		

<4 種類のワークを 1 度に撮像した時のデータ格納例>

- Decode Data String 01: 「34567」 (5 文字、Micro QR Code)
- Decode Data String 02: 「 89012」 (5 文字、MicroPDF417)
- Decode Data String 03: 「12345」 (5 文字、QR Code)
- Decode Data String 04: 「67890」 (5 文字、Data Matrix)

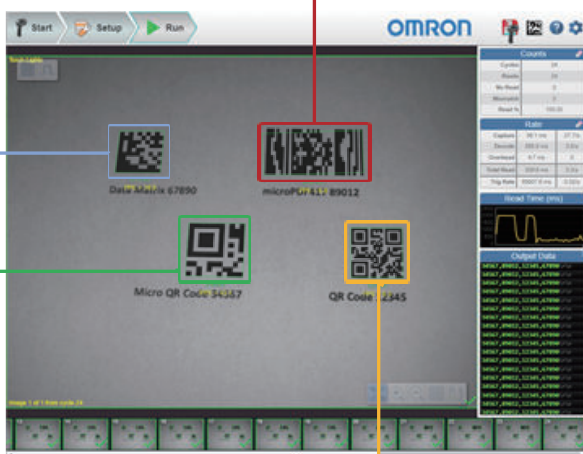
デバイス名	名称	モニタ値	変更	表示形式
new_Controller_0	BIOutput.COMMANDS			
	Run_Mode	False	TRUE FALSE	Boolean
	Trigger	True	TRUE FALSE	Boolean
	Enable_MatchCode	False	TRUE FALSE	Boolean
	Reset_General_Fault	False	TRUE FALSE	Boolean
	Clear_No_Read_ReadCycle_Count	False	TRUE FALSE	Boolean
	Clear_MisMatch_ReadCycle_Count	False	TRUE FALSE	Boolean
	Clear_No_Read_Count	False	TRUE FALSE	Boolean
	Clear_Tripping_Count	False	TRUE FALSE	Boolean
	Clear_MatchCode_Count	False	TRUE FALSE	Boolean
	Clear_MisMatch_Count	False	TRUE FALSE	Boolean
	Output_1	False	TRUE FALSE	Boolean
	Output_2	False	TRUE FALSE	Boolean
	Output_3	False	TRUE FALSE	Boolean
	Reserved[0-17]	False	TRUE FALSE	Boolean
	BIInput			
	INFO_BITS	01		Hexadecimal

各コードのDecode Data Length (読み取り文字列長) (5文字)



- Decode Data String 01: 「34567」 (5文字、Micro QR Code)
- Decode Data String 02: 「89012」 (5文字、MicroPDF417)
- Decode Data String 03: 「12345」 (5文字、QR Code)
- Decode Data String 04: 「67890」 (5文字、Data Matrix)

デバイス名	名称	モニタ値	変更	表示形式
new_Controller_0	DECODE_1_DATA_LENGTH	5		Decimal
	DECODE_1_DATA_STRING[0-159]			
	DECODE_1_DATA_STRING[0]	8 (16#38)		ASCII
	DECODE_1_DATA_STRING[1]	9 (16#39)		ASCII
	DECODE_1_DATA_STRING[2]	0 (16#30)		ASCII
	DECODE_1_DATA_STRING[3]	1 (16#31)		ASCII
	DECODE_1_DATA_STRING[4]	2 (16#32)		ASCII
	DECODE_1_DATA_STRING[5]	(16#00)		ASCII
	DECODE_1_DATA_STRING[6]	(16#00)		ASCII
	DECODE_1_DATA_STRING[7]	(16#00)		ASCII
	DECODE_1_DATA_STRING[8]	00		Hexadecimal
new_Controller_0	DECODE_2_DATA_LENGTH	5		Decimal
	DECODE_2_DATA_STRING[0-71]			
	DECODE_2_DATA_STRING[0]	6 (16#36)		ASCII
	DECODE_2_DATA_STRING[1]	7 (16#37)		ASCII
	DECODE_2_DATA_STRING[2]	8 (16#38)		ASCII
	DECODE_2_DATA_STRING[3]	9 (16#39)		ASCII
	DECODE_2_DATA_STRING[4]	0 (16#30)		ASCII
	DECODE_2_DATA_STRING[5]	(16#00)		ASCII
	DECODE_2_DATA_STRING[6]	(16#00)		ASCII
	DECODE_2_DATA_STRING[7]	(16#00)		ASCII
	DECODE_2_DATA_STRING[8]	(16#00)		ASCII
new_Controller_0	DECODE_3_DATA_LENGTH	5		Decimal
	DECODE_3_DATA_STRING[0-71]			
	DECODE_3_DATA_STRING[0]	3 (16#33)		ASCII
	DECODE_3_DATA_STRING[1]	4 (16#34)		ASCII
	DECODE_3_DATA_STRING[2]	5 (16#35)		ASCII
	DECODE_3_DATA_STRING[3]	6 (16#36)		ASCII
	DECODE_3_DATA_STRING[4]	7 (16#37)		ASCII
	DECODE_3_DATA_STRING[5]	(16#00)		ASCII
	DECODE_3_DATA_STRING[6]	(16#00)		ASCII
	DECODE_3_DATA_STRING[7]	(16#00)		ASCII
	DECODE_3_DATA_STRING[8]	(16#00)		ASCII
new_Controller_0	DECODE_4_DATA_LENGTH	5		Decimal
	DECODE_4_DATA_STRING[0-71]			
	DECODE_4_DATA_STRING[0]	1 (16#31)		ASCII
	DECODE_4_DATA_STRING[1]	2 (16#32)		ASCII
	DECODE_4_DATA_STRING[2]	3 (16#33)		ASCII
	DECODE_4_DATA_STRING[3]	4 (16#34)		ASCII
	DECODE_4_DATA_STRING[4]	5 (16#35)		ASCII
	DECODE_4_DATA_STRING[5]	(16#00)		ASCII
	DECODE_4_DATA_STRING[6]	(16#00)		ASCII
	DECODE_4_DATA_STRING[7]	(16#00)		ASCII
	DECODE_4_DATA_STRING[8]	(16#00)		ASCII



注意: 入力アセンブリ 104 (4 デコード入力) と入力アセンブリ 105 (N デコード入力) についてリーダで出力フォーマットが有効になっていない場合、**DECODE ‘X’ DATA**、**DECODE ‘X’ LENGTH**、および **DECODE ‘X’ CYCLE REPORT** は、すべてグループ化された各コードのデータを反映します。つまり、同じコードに関連する **DECODE 1 DATA**、**DECODE 1 LENGTH**、および **DECODE 1 CYCLE REPORT** です。

リーダで出力フォーマットが有効になっている場合、**DECODE ‘X’ DATA** および **DECODE ‘X’ LENGTH** はユーザが定義したフォーマットを反映します。ただし、**DECODE ‘X’ CYCLE REPORT** は、ユーザが定義したフォーマットの影響を受けることなく、リーダで出力フォーマットが有効になっていない場合と同じように情報を保持します。

要約:

- リーダでフォーマット出力が有効になっている場合、**DECODE ‘X’ CYCLE REPORT** をコードに関連付けることはできません。
- それらが必要な場合は、正しい情報を保証するために、“Output Coordinates” と “Pixel Per Element” を有効にし、“Include Symbology Identifier” オプション (I/O タブ) を出力フォーマットに追加し、**DECODE ‘X’ DATA** および **DECODE ‘X’ LENGTH** で提示された情報だけを参照してください。**DECODE ‘X’ CYCLE REPORT** で提示された情報は無視してください。
- この動作は、入力アセンブリ 104 (4 デコード入力) および入力アセンブリ 105 (N デコード入力) の両方で有効です。

● 出力 (レガシー) で NEW MASTER ビットを使って、マスタシンボルデータを登録する

<タグセット・コネクション設定例>

- 入力アセンブリ：大型入力 (101)
- 出力アセンブリ：出力 (レガシー) (198)

▼ コネクション									
コネクション数/使用可能数: 2 / 32									
ターゲットデバイス	コネクション名	コネクション/Oタイプ	入力/出力	ターゲット変数	サイズ[Byte]	オリジネータ変数	サイズ[Byte]	コネクションタイプ	RPI[ms]
192.168.188.2 V460-H Rev 1	default_001	IO big	入力	101	176	EIPInput	176	Point to Point connection	50.0
			出力	198	12	EIPOutput	12	Point to Point connection	RPI x 4

▼ タグセット						
タグセット数/使用可能数: 2 / 32 タグ数/使用可能数: 2 / 256						
入力 出力						
タグセット名	Bit選択	サイズ (Byte)	サイズ (Bit)	インスタンスID	コントローラステータス	
EIPInput	☐	176		Auto	含めない	
EIPInput	☐	176	0			

▼ タグセット						
タグセット数/使用可能数: 2 / 32 タグ数/使用可能数: 2 / 256						
入力 出力						
タグセット名	Bit選択	サイズ (Byte)	サイズ (Bit)	インスタンスID	コントローラステータス	運転停止異常時
EIPOutput	☐	12		Auto	含めない	
EIPOutput	☐	12	0			クリア

<変数設定例>

グループフィルタ ▼ (グループ設定なし) ▼							
名称	データ型	初期値	割付先	保持	コンスタント	ネットワーク公開	コメント
EIPInput	S_EIPInput101			☐	☐	入力 ▼	
EIPOutput	S_EIPOutput198			☐	☐	出力 ▼	

<入力アセンブリ構造体(101)>

root	名称	データ型	オフセット種別	バイトオフセット	ビットオフセット	コメント
構造体型	Device_status_101	STRUCT	任意			
共用体型	Online	BOOL		0	0	
列挙型	Trigger_Acknowledge	BOOL		0	1	
	Exposure_Done	BOOL		0	2	
	Decoding	BOOL		0	3	
	Data_Is_Ready	BOOL		0	4	
	Read_Cycle_Pass	BOOL		0	5	
	Read_Cycle_Fail	BOOL		0	6	
	General_Fault	BOOL		0	7	
	New_match_code_acknowledged	BOOL		1	0	
	Match_Code_Enabled	BOOL		1	1	
	Image_Sensor_Calibrating	BOOL		1	2	
	Image_Sensor_Calibration_Compl...	BOOL		1	3	
	Training	BOOL		1	4	
	Training_Complete	BOOL		1	5	
	Optimizing	BOOL		1	6	
	Optimization_Complete	BOOL		1	7	
	AutoImage_Photometry_Enabled	BOOL		2	0	
	AutoImage_Photometry_Complete	BOOL		2	1	
	Output1_Status	BOOL		2	2	
	Output2_Status	BOOL		2	3	
	Buffer_Overflow	BOOL		2	4	
	Reserved	ARRAY[0..10] OF BO...		2	5	
	S_EIPIInput101	STRUCT	NJ			
	USER_DEFINED_TAG_ECHO	DINT				
	COMMAND_ECHO	DINT				
	OUTPUT_CONTROL_ECHO	DINT				
	EXTERNAL_INPUT_STATUS	DINT				
	EXTERNAL_OUTPUT_STATUS	DINT				
	DEVICE_STATUS	Device_status_101				
	READ_CYCLE_SEQUENCE_COUNTER	DINT				
	TRIGGER_COUNT	DINT				
	DECODE_MATCH_COUNT	DINT				
	MISMATCH_COUNT	DINT				
	NOREAD_COUNT	DINT				
	DECODE_LENGTH	DINT				
	DECODE_DATA	ARRAY[0..127] OF B...				

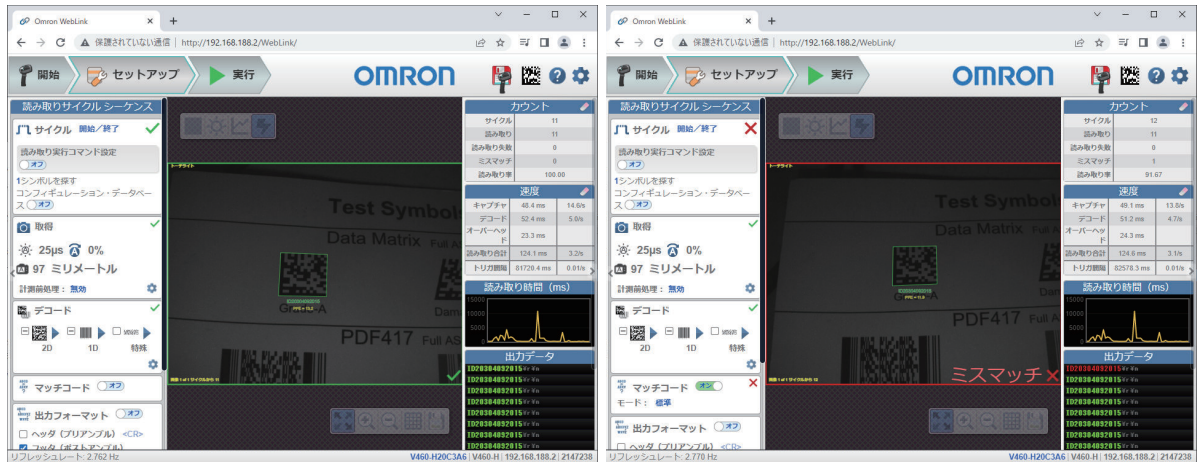
<出力アセンブリ構造体(198)>

root	名称	データ型	オフセット種別	バイトオフセット	ビットオフセット	コメント
構造体型	S_EIPIOutput198	STRUCT	NJ			
共用体型	USER_DEFINED_TAGS	DINT				
列挙型	COMMANDS	COMMAND_198				
	EXTERNAL_OUTPUT	DINT				
	COMMAND_198	STRUCT	任意			
	Trigger	BOOL		0	0	
	New_Master	BOOL		0	1	
	Reserved1	ARRAY[0..5] OF BOOL		0	2	
	Disable_Scanning	BOOL		1	0	
	Reserved2	ARRAY[0..6] OF BOOL		1	1	
	Clear_Read_Cycle	BOOL		2	0	
	Unlatch_Outputs	BOOL		2	1	
	Reserved3	ARRAY[0..13] OF BO...		2	2	

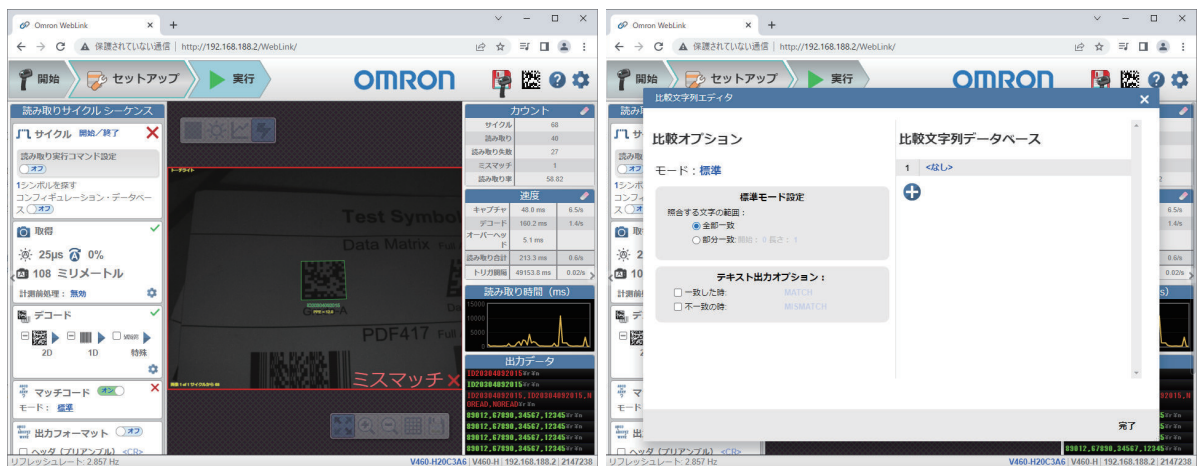
マッチコード=ON に設定します。

- WebLink_{HH} - [セットアップ] 画面

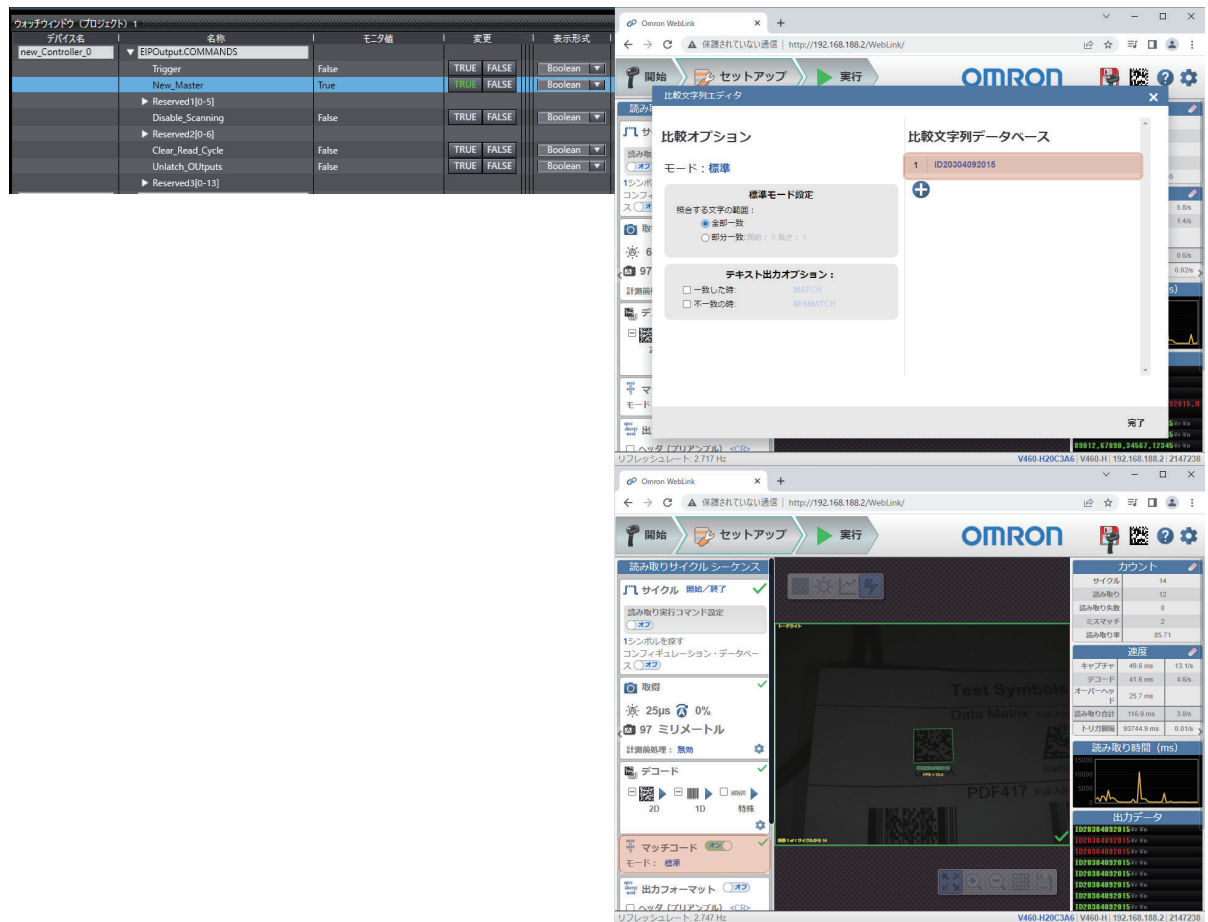
2 イーサネットで制御／出力する



この状態で読み取りを実施した時、マスタシンボルを登録していないため、「文字列不一致（ミスマッチ）」になります。



出力アセンブリに含まれる NewMaster ビットを「1」に設定することで、次に読み取りしたコードを、マスタシンボルデータに登録することができます。



2-1-10 EtherNet/IP のメッセージ通信でコントローラと通信する

EtherNet/IP のメッセージ通信機能を使用して、シリアルコマンドを実行することが可能です。シリアルコマンドについては、以下を参照してください。

「2-2-5 外部からコードリーダを制御する (P.2-35)」

(注) 3 秒以上の動作を引き起こす Explicit メッセージはタイムアウトとなります。このような場合は、Implicit メッセージを使用してください。

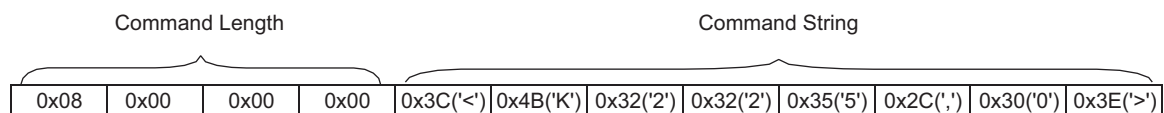
メッセージ通信のオブジェクトの構成は以下のとおりです。

項目	設定値
Class ID	104 (0x68 hex)
Instance ID	1
Attribute ID	1
Service code	69 (0x45 hex)

EtherNet/IP メッセージ(Explicit)フォーマット

EtherNet/IP メッセージは、送受信とも Command Length (コマンド長) と Command String (コマンド文字列) の 2 つの部分から構成されています。

- **Command Length (4 バイト)**
Command String (コマンド文字列) の合計文字数
- **Command String (最大 256 バイト)**
PLC からコードリーダに送信するコマンドの ASCII 文字列



コマンド設定例

メッセージ通信のコマンド文字列設定例を示します。

- PLC→コードリーダの送信データには、シリアルコマンド文字列を設定します。
- K コマンドを使用する場合、一部のコマンドはレスポンスを返しません。

つまり、コマンド送信後に受信するデータはありません。

ただし、<K225?>のような設定値確認コマンドに対してはレスポンスがあるため、受信データが取得されます。

1 回の送信で複数のコマンドを送信できるため、送信されたコマンドが通常レスポンスを生成しない場合、例 3 に示すように、設定確認コマンドを追加で送信できることに注意が必要です。

例 1：「<K225,0>」コマンド文字列を送信した場合の受信データ

(送信データ) 0x08 0x00 0x00 0x00 0x3C('<') 0x4B('K') 0x32('2') 0x32('2') 0x35('5') 0x2C(',') 0x30('0') 0x3E('>') の 12 byte

(受信データ) なし

例 2：「<K225?>」コマンド文字列を送信した場合の受信データ

(送信データ) 0x07 0x00 0x00 0x00 0x3C('<') 0x4B('K') 0x32('2') 0x32('2') 0x35('5') 0x3F('?') 0x3E('>') の 11 byte

(受信データ) 0x08 0x00 0x00 0x00 0x3C('<') 0x4B('K') 0x32('2') 0x32('2') 0x35('5') 0x2C(',') 0x30('0') 0x3E('>') の 12 byte

例 3：「<K225,0><K225?>」コマンド文字列を送信した場合の受信データ

(送信データ) 0x0F 0x00 0x00 0x00 0x3C('<') 0x4B('K') 0x32('2') 0x32('2') 0x35('5') 0x2C(',') 0x30('0') 0x3E('>') 0x3C('<') 0x4B('K') 0x32('2') 0x32('2') 0x35('5') 0x3F('?') 0x3E('>') の 19 byte

(受信データ) 0x08 0x00 0x00 0x00 0x3C('<') 0x4B('K') 0x32('2') 0x32('2') 0x35('5') 0x2C(',') 0x30('0') 0x3E('>') の 12 byte

2-2 シリアル (TCP) で制御／出力する

コードリーダーと外部装置をシリアル (TCP) 方式で通信するときに必要な、通信設定の仕方、入力フォーマットなどについて説明しています。

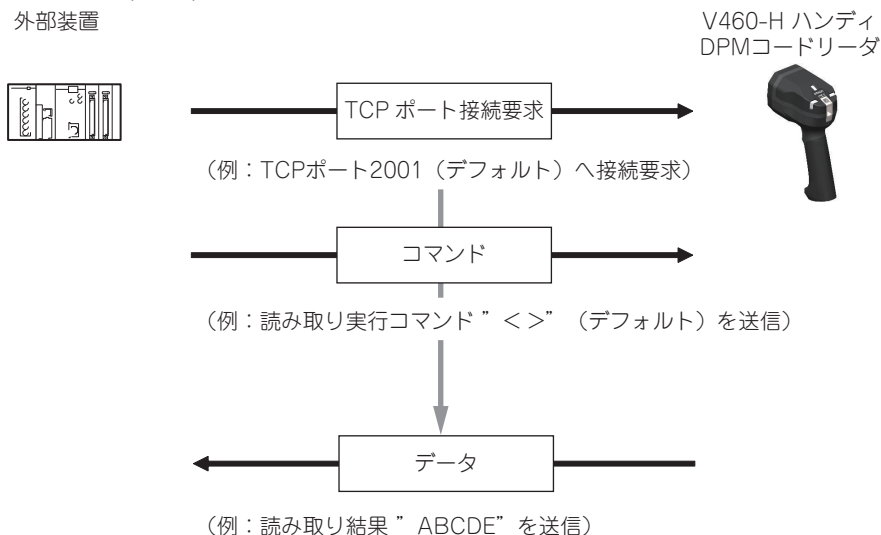
2-2-1 シリアル (TCP) の概要

シリアル (TCP) は、TCP/IP 通信プロトコルに準拠しています。TCP/IP 通信プロトコルに対応したイーサネット通信機器と通信が可能です。本コードリーダーは TCP サーバとして通信しますので、接続する外部機器は TCP クライアントとして、本コードリーダーに接続する必要があります。オムロン製 PLC の場合、ソケットサービス通信 (TCP クライアント) に対応していることをご確認ください。

2-2-2 通信処理の流れ

外部装置 (PLC など) とコードリーダーをシリアル (TCP) で通信して、外部装置 (PLC など) からシリアルコマンドによる制御、および読み取り結果の出力が可能です。

シリアル (TCP) 通信の開始から読み取り実行、読み取り結果出力までの流れは以下です。



2-2-3 通信仕様を設定する（シリアル（TCP））

コードリーダー本体のネットワーク設定をする

PLC など、外部装置のネットワークに合わせて、コードリーダーの IP アドレスを設定します。

- WebLink_{HH} - [セットアップ] 画面 - [歯車アイコン] - [詳細] - [通信設定] タブ - [イーサネット]

- 1** PLC などの外部装置のネットワークに合わせ、[IP アドレス]、[サブネットマスク] を設定します。

設定項目	設定値	説明
IP アドレス	a.b.c.d a : 0~255 b : 0~255 c : 0~255 d : 0~255 (初期値 : 192.168.188.2)	コードリーダーの IP アドレスを入力します。
サブネット	a.b.c.d a : 0~255 b : 0~255 c : 0~255 d : 0~255 (初期値 : 255.255.0.0)	サブネットマスクのアドレスを入力します。
ゲートウェイ	a.b.c.d a : 0~255 b : 0~255 c : 0~255 d : 0~255 (初期値 : 0.0.0.0)	ゲートウェイを使用する場合は、ゲートウェイの IP アドレスを入力します。使用しない場合は初期値の 0.0.0.0 にしてください。
IP アドレスモード	• 固定 (初期値) • DHCP	固定モードでは、コードリーダーはユーザ定義の IP アドレスを使用します。 DHCP モードでは、コードリーダーは DHCP サーバから IP アドレス、サブネット、およびゲートウェイを取得します。 PLC 通信の場合、[固定モード] が必須です。DHCP モードを有効にしないでください。
TCP ポート 1	1024~65536 (初期値 : 2001)	コードリーダーとシリアル（TCP）用の 2 つの TCP ポートの 1 つのポート番号を入力します。
TCP ポート 2	1024~65536 (初期値 : 2003)	コードリーダーとシリアル（TCP）用の 2 つの TCP ポートの 1 つのポート番号を入力します。



参考

本コードリーダーは、同時に 2 つの TCP ポートを使用することで、異なる 2 つの外部機器とシリアル（TCP）通信が可能です。

読み取り実行コマンドを変更する

シリアル通信の読み取り実行コマンドを変更することが可能です。

読み取り実行コマンドには、区切り文字 < > ありと区切り文字なしの 2 種類があります。

- WebLink_{HH} - [セットアップ] 画面 - [歯車アイコン] - [詳細] - [読み取りサイクル] タブ - [読み取り実行コマンド設定]

設定項目	設定値	説明
読み取り実行コマンド文字 (区切りあり)	1 文字の ASCII コード (初期値: スペース (16 進: 20))	読み取りを開始するコマンド文字を指定します。このコマンド実行時は、トリガ文字を山かっこの区切り文字 <> 内に入力する必要があります。 このコマンドを実行するには、トリガモード ([読み取りサイクル] タブ - [トリガ] - [モード]) を「コマンド入力」または「コマンド入力または外部トリガ信号エッジ」にすることが必要です。
読み取り開始コマンド文字 (区切りなし)	最大 2 文字の ASCII コード (初期値: NULL (16 進: 00))	読み取りを開始するコマンド文字、読み取りを終了するコマンド文字を指定します。 開始コマンド文字、終了コマンド文字は異なる文字を設定する必要があります。NULL (16 進: 00) の場合は無効になります。
読み取り終了コマンド文字 (区切りなし)	最大 2 文字の ASCII コード (初期値: NULL (16 進: 00))	トリガモード ([読み取りサイクル] タブ - [トリガ] - [モード]) により動作が変わります。 • 「外部トリガ信号エッジ」の場合、コードリーダーは開始コマンド文字により読み取りを実行します。終了コマンド文字は必要ありません。 • 「外部トリガ信号レベル」または、「コマンド入力または外部トリガ信号エッジ」の場合、開始トリガ文字が読み取りサイクルを開始し、終了コマンド文字が読み取りサイクルを終了します。読み取りが成功した場合でも終了コマンドが送信されるまで読み取りサイクルは終了しません。

● 読み取り実行コマンド文字 (区切りあり) 実行例

- 読み取り文字列: 12345、読み取り実行コマンド文字 (区切りあり): スペース、ヘッダ: なし、フッタ: CRLF の場合

外部装置



コードリーダー



	読み取り実行 コマンド		
文字表記	<		>
16 進表記	3C	20	3E

読み取りサイクル中		読み取り結果						
文字表記	1	2	3	4	5	CR	LF	
16 進表記	31	32	33	34	35	0D	0A	

2-2-4 読み取り後に出力するデータの設定

コードリーダーは読み取り実行後、読み取り結果を接続している TCP ポートに自動的に出力することが可能です。読み取り結果には、シンボルの印字品質評価指標や位置座標などの情報を付加することや、読み取り結果のフォーマットを変更することができます。

読み取り結果出力条件を変更する

読み取り結果の出力条件を変更することができます。

- WebLink_{HH} - [セットアップ] 画面 - [歯車アイコン] - [詳細] - [I/O] タブ - [読み取り結果出力条件]

設定項目	設定値	説明
読み取り結果出力条件	• 無効 • マッチ • ミスマッチ • 読み取り成功（初期値） • 全ての読み取り成功	• 無効：読み取り結果を出力しません。 • マッチ：マッチコード機能でマスタシンボルと一致したときのみ、読み取り結果を出力します。 • ミスマッチ：マッチコード機能でマスタシンボルと一致しなかったときのみ、読み取り結果を出力します。 • 読み取り成功：1 つでも読み取り成功したときに、読み取り結果を出力します。 • 全ての読み取り成功：複数シンボル読み取り機能で指定されたシンボル数の全ての読み取りに成功したときに、読み取り結果を出力します。
出力タイミング [読み取りサイクル終了条件]	• 読み取り成功直後（初期値） • 読み取りサイクル終了条件	• 読み取り成功直後：読み取りが成功した時点で読み取り結果を出力し、読み取りサイクルを終了します。 • 読み取りサイクル終了条件：読み取りサイクル終了条件を満たすまで読み取り結果を出力しません。読み取りサイクル終了条件は [詳細] - [読み取りサイクル] - [読み取りサイクル終了条件] で設定します。

読み取り失敗時の出力内容を設定する

読み取り失敗時の出力内容を変更することができます。

- WebLink_{HH} - [セットアップ] 画面 - [歯車アイコン] - [詳細] - [I/O] タブ - [読み取り失敗時のエラー出力]

設定項目	設定値	説明
読み取り失敗時のエラー出力	• 有効（初期値） • 無効	• 有効：読み取り失敗時にメッセージを出力します。ただし、読み取りサイクルのトリガモードが「連続読み取り」の場合は、この設定にかかわらずメッセージは出力しません。 • 無効：読み取り失敗時にメッセージを出力しません。
メッセージ	NOREAD（初期値）	読み取り失敗時のメッセージ文字列を設定します。任意の ASCII 文字を、最大 64 文字まで設定できます。

ヘッダ、フッタを設定する

読み取り文字列の前後に付加するヘッダ（プリアンブル）、フッタ（ポストアンブル）を変更することができます。

- WebLink_{HH} - [セットアップ] 画面 - [出力フォーマット]

1 必要に応じて、ヘッダ（プリアンブル）を設定します。

読み取り結果にヘッダを付加する場合は、[出力フォーマット] 欄の [ヘッダ] をチェックします。

出力フォーマット ☐ オン

☐ ヘッダ（プリアンブル） <CR>

☒ フッタ（ポストアンブル）
<CR><LF>

入れ替え：0

☐ 制御文字を削除する

☐ 拡張文字を削除する

ヘッダ文字の編集は、[ヘッダ（プリアンブル）] の右の青文字を選択します。テキスト入力ボックスが表示されますので、直接キーボードから文字を入力します。

出力フォーマット ☐ オン

☒ ヘッダ（プリアンブル）
x02

ASCII...

☒ フッタ（ポストアンブル）
<CR><LF>

入れ替え：0

☐ 制御文字を削除する

☐ 拡張文字を削除する

制御文字を入力したい場合は入力ボックスの下に [ASCII...] を選択します。制御文字入力ボードが表示されますので、入力したい制御文字を選択します。

出力フォーマット ☐ オン

☒ ヘッダ（プリアンブル）
x02

ASCII...

SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS
TAB	LF	VT	FF	CR	BSO	SI	DLE
DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN
EM	SUB	ESC	FS	GS	RS	US	SP

☐ 制御文字を削除する

☐ 拡張文字を削除する

2 必要に応じて、フッタ（ポストアンブル）を設定します。

設定手順は、ヘッダ（プリアンブル）と同様です。



参考

ヘッダ、フッタは [詳細] - [通信設定] - [ヘッダ（プリアンブル）]、[フッタ（ポストアンブル）] から設定可能です。

出力フォーマットを設定する

読み取り文字列の出力範囲の指定、固定文字列の挿入など出力フォーマットの設定ができます。

- WebLink_{HH} - [セットアップ] 画面 - [出力フォーマット]

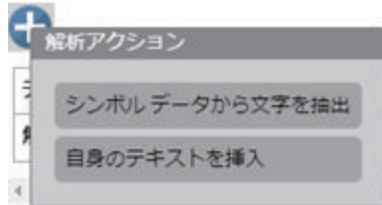
- 1 必要に応じて、出力フォーマットを変更します。
[出力フォーマット] をオンにします。



- 2 出力フォーマットエディタ画面を開きます。
[出力フォーマット] 下の、青文字で表示される [フォーマット:] をクリックすると、出力フォーマットエディタ画面が開きます。



- 3 [シンボルを解析しますか?] にチェックを入れます。
[+] ボタンを押すと、[シンボルデータから文字を抽出] か [自身のテキストを挿入] を選択できます。



- 4** 読み取り文字列の出力範囲を指定する場合は、[シンボルデータから文字を抽出] を選択します。
出力する文字範囲を入力します。下記は、1 文字から 4 文字までを出力する場合の設定例です。



設定を反映する場合は、 ボタンを押します。

- 5** 読み取り文字列に固定文字列を挿入する場合は、[自身のテキストを挿入] を選択します。
初期値の「/r」をクリックすると、テキスト入力ボックスが表示されますので、キーボードから文字を入力します。下記は「ABC-」という 4 文字を挿入する場合の設定例です。



制御文字を入力したい場合は、入力ボックスの下に [ASCII...] を選択します。
制御文字入力ボードが表示されますので、入力したい制御文字を選択します。



入力を終了する場合は、キーボードの Enter キーを入力します。

- 6** 設定した出力フォーマットを削除する場合は、設定した出力フォーマットにカーソルを当てると右上に表示される X ボタンをクリックします。



- 7** [完了] ボタンを押し、出力フォーマットエディタ画面を終了します。

シンボル情報を付加する

読み取り結果には、シンボルの印字品質評価や位置座標などの情報を付加することができます。

● シンボルの位置情報を出力する

- WebLink_{HH} - [セットアップ] 画面 - [歯車アイコン] - [詳細] - [I/O] タブ - [コード情報出力]

1 [コード位置座標出力] を有効にします。



2 必要に応じて、区切り文字を変更します。

WebLink_{HH} - [セットアップ] 画面 - [歯車アイコン] - [詳細] - [シンボル品質] タブ - [グローバル] の、[コード評価指標の区切り文字] を入力します。下記は区切り文字を「,」に設定した例です。



3 読み取り結果にシンボルの位置座標が出力されます。

下記は読み取り文字列「ABCDE」にシンボルの位置座標を付加した例です。
ABCDE,(0867,0708)(0867,0708)(1741,0673)(1741,0673)

● 付加できるシンボル情報の一覧

付加情報	設定箇所 (WebLink _{HH} - [詳細] メニュー)	説明	出力例 (読み取り文字列は「ABCDE」とします。区切り文字は「,」とします。)	出力順序
シンボル識別子	[I/O] タブ - [読み取り結果出力条件]	読み取ったシンボルの種類を示すシンボル識別子（3文字）を、読み取り文字列の前に付与します。	]dlABCDE	読み取り文字列の前に付加
デコード／トリガ	[I/O] タブ - [デコード／トリガ出力]	読み取りサイクル中に読み取りに成功した回数を出力します。	ABCDE,00002	1
コンフィグレーションデータ識別子	[I/O] タブ - [コンフィグレーションデータベース識別子出力]	読み取りに成功したコンフィグレーションデータベースのインデックス番号を出力します。	ABCDE,DB01	2
フレーム番号	[I/O] タブ - [コード情報出力]	シンボルの読み取りに成功したフレーム番号（撮像回数）を3桁の数字で出力します。	ABCDE,F010	3
コード位置座標	[I/O] タブ - [コード情報出力]	読み取ったシンボルの4頂点の座標をピクセル単位で出力します。	ABCDE, (0032,0040) (0287,0056) (0287,0279) (0048,0271)	4
印字品質 (ISO/IEC 16022)	[シンボル品質] タブ - [ISO/IEC 16022 パラメータ]	ISO/IEC 16022 で定義された DataMatrix の印字品質評価指標を出力します。*1	ABCDE,B,A,A,A	5
印字品質 (Omron Microscan)	[シンボル品質] タブ - [Omron Microscan パラメータ]	Omron Microscan 独自の印字品質評価指標を出力します。*1	ABCDE,000,092,1 43,091,001,14,3,2 00,16X16,PASS,3 49	6
読み取り時間	[I/O] タブ - [読み取り時間の出力]	読み取りにかかった時間を ms 単位で出力します。	ABCDE,100	10

付加情報	設定箇所 (WebLink _{HH} - [詳細] メニュー)	説明	出力例 (読み取り文字列は「ABCDE」とします。区切り文字は「,」とします。)	出力順序
読み取りサイクル ID	[I/O] タブ - [読み取りサイクル ID 出力]	読み取りサイクル ID 番号 (読み取り実行回数) を 16 進値で出力します。	ABCDE,CycleID=0x8	11

*1. 印字品質評価指標についての詳細は、『V460-H ハンディ DPM コードリーダー ユーザーズマニュアル (SDNC-768)』「コード評価指標」のを参照して下さい。



参考

複数シンボル読み取りを有効にしている場合、出力順序は、「シンボル 1 読み取り文字列, シンボル 1 付加情報, シンボル 2 読み取り文字列, シンボル 2 付加情報, . . .」の順になります。

2-2-5 外部からコードリーダーを制御する

シリアルコマンドにより、外部装置からコードリーダーの設定変更や設定値の取得、または制御が可能です。

本コードリーダーには大きくわけて以下の 2 つの種類のシリアルコマンドがあります。

- **シリアルコンフィギュレーションコマンド (K コマンド)**

本コードリーダーの設定変更をおこないます。

- **シリアルユーティリティコマンド**

読み取り率のテスト、コードリーダー状態取得、自動調整などの制御をおこないます。

シリアルコマンドフォーマットについて

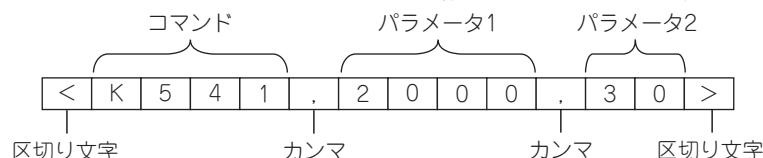
シリアル通信のコマンドフォーマットについて説明します。

- **シリアルコンフィギュレーションコマンドとシリアルユーティリティコマンドで共通のフォーマット**

- ・ < および > のかっこでコマンドを囲みます。
- ・ コマンドとデータは大文字と小文字が区別されます。文字は、指定どおりに大文字または小文字で入力する必要があります。
- ・ シリアルコマンドは連結できます。たとえば、以下のコマンドは、トリガモードを外部トリガエッジに指定し、読み取りサイクル終了条件を新規トリガに設定し、設定保存を実行します。
<K200,3> <K220,1> <Z>

- **シリアルコンフィギュレーションコマンド (K コマンド) のフォーマット**

- ・ K コマンドは以下のように、K と 3 桁の数字、カンマ区切りのパラメータから構成されます。



- ・ 変更する必要のないパラメータが後ろにある場合は、省略できます。
たとえば、露光時間とゲインを変更する K コマンド <K541> で露光時間だけを変更したい場合は、以下のように入力できます。
<K541,1000>
- ・ 変更する必要のないパラメータが前にある場合は、カンマのみを入力できます。

たとえば、露光時間とゲインを変更する K コマンド<K541>でゲインだけを変更したい場合は、以下のように入力できます。

<K541,,30>

- 数字以外の制御文字をパラメータに入力する場合は、16 進値として入力します。パラメータとして文字<、>、カンマ(,)を含める必要がある場合も 16 進値として入力します。16 進値をパラメータに入力する場合は、K コマンドの直後に小文字の h を追加します。

たとえば、フッタ（ポストアンブル）に CR（16 進値：0D）を設定する場合は、以下のように入力できます。

<K142h,,0D>

- K コマンドに対するレスポンスは初期状態ではありません。現在のコードリーダーの状態を取得する場合は、<Knnn?>コマンドを使用します。

たとえば、現在の露光時間とゲインを取得するコマンドとそのレスポンス例は以下になります。

・設定値取得コマンド

<	K	5	4	1	?	>
---	---	---	---	---	---	---

・レスポンス

指定したKコマンドに対する現在の設定値

	<	K	5	4	1	,	2	0	0	0	,	3	0	>	CR	LF
--	---	---	---	---	---	---	---	---	---	---	---	---	---	---	----	----

ヘッダ（プリアンブル） フッタ（ポストアンブル）



参考

レスポンスには、ヘッダ（プリアンブル）、フッタ（ポストアンブル）が付与されます。初期値は、ヘッダ：「なし」、フッタ：「CRLF」（16 進値：0D0A）です。

- シリアル検証機能（デフォルト：無効）を有効にすると、K コマンドのレスポンスとして、現在の設定状態が返ります。K コマンドが正しく反映されたことを確認したい場合は、シリアル検証機能を有効にしてください。

・Kコマンド（シリアル検証機能が有効な場合）

<	K	5	4	1	,	1	0	0	0	>
---	---	---	---	---	---	---	---	---	---	---

・レスポンス

指定したKコマンドに対する現在の設定値

	<	K	5	4	1	,	1	0	0	0	,	3	0	>	CR	LF
--	---	---	---	---	---	---	---	---	---	---	---	---	---	---	----	----

ヘッダ（プリアンブル） フッタ（ポストアンブル）



参考

レスポンスには、ヘッダ（プリアンブル）、フッタ（ポストアンブル）が付与されます。

● シリアルユーティリティコマンドのフォーマット

- シリアルユーティリティコマンドには、レスポンスがあるコマンドとレスポンスがないコマンドがあります。レスポンスのフォーマットはコマンド毎に異なります。

- ・アプリケーションバージョン取得コマンド

<	#	a	>
---	---	---	---

- ・レスポンス

<	#	a	/	3	5	-	9	0	0	0	0	9	7	-	1	.
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

ヘッダ (プリアンブル)

2	.	3	.	3	0	0	8	>	CR	LF
---	---	---	---	---	---	---	---	---	----	----

フッタ (ポストアンブル)

- ・トリガカウンタのクリア

<	U	>
---	---	---

- ・レスポンス

なし

2-2-6 シリアルコマンド一覧

シリアルコマンドの一覧です。

種別	コマンド	説明	レスポンスデータ例 (読み取り文字列は「ABCDE」 とします。)
設定変更 (K コマンド)	<Knnn,パラ メータ> nnn: 各 K コ マンドの 3 桁 の数値	コードリーダーの設定変更をおこないます。各 K コマンドについては、『V460-H ハンディ DPM コードリーダー ユーザーズマニュアル (SDNC-768)』を参照してください。	なし (シリアル検証機能が有効な 場合は、<Knnn?>コマンドと 同じレスポンスデータになり ます。)
読み取り実行	ユーザ定義 (初期値:< >)	読み取り実行コマンド (区切りあり) です。 (参照:「読み取り実行コマンドを変更する (P.2-28)」)	ABCDE
	ユーザ定義 (初期値: 無 効)	読み取り開始コマンド (区切りなし) です。 (参照:「読み取り実行コマンドを変更する (P.2-28)」)	ABCDE
	ユーザ定義 (初期値: 無 効)	読み取り終了コマンド (区切りなし) です。 (参照:「読み取り実行コマンドを変更する (P.2-28)」)	ABCDE
読み取りテス ト	<C>	1 秒あたりのデコード回数をテストします。 レスポンスデータはデコード回数 / sec と読 み取り文字列が出力されます。	5 Decodes / Sec ABCDE *1 (1 秒間に 5 回読み取りに成 功したことがわかります。)
	<Cp>	読み取り率 (%) をテストします。レスポンス データは 100 回読み取りを実行したときの読 み取り成功したパーセンテージと読み取り文 字列が出力されます。	95% ABCDE *1 (100 回読み取り実行中 95 回 読み取りに成功したことがわ かります。)
	<J>	読み取りテストを終了します。	なし
自動調整	<@CAL>	露光時間、フォーカス位置やシンボルタイプを 自動調整します。レスポンスデータには途中 経過とキャリブレーション成功時には Calibration PASSED.が出力されます。キャ リブレーション失敗時には Calibration FAILED.が出力されます。	Prog Exposure Gain Brightness 2 5764 33 24 100 6011 33 37 Calibration PASSED. *1
ティーチング	<TRAIN>	ティーチング操作を開始します。次に読み取 り実行したシンボルをティーチングします。 ティーチングをすることで同じシンボルの読 み取りが安定します。	なし
	<UNTRAIN >	ティーチング操作を解除します。	なし
	<TRAIN?>	ティーチング状態を確認します。レスポンス データは、ティーチング状態に応じて以下のど れかになります。 ・ <TRAIN,0>: デフォルト、ティーチングさ れていない ・ <TRAIN,1>: ティーチング操作が進行中 ・ <TRAIN,2>: シンボルがティーチングされ た	<TRAIN,2>
最適化	<OPT>	最適化操作を開始します。次に読み取り実行 したシンボルを最適化します。最適化をする ことで同じシンボルの読み取り速度が早くな ります。	なし
	<UNOPT>	最適化操作を解除します。	なし

種別	コマンド	説明	レスポンスデータ例 (読み取り文字列は「ABCDE」 とします。)
	<OPT?>	最適化状態を確認します。レスポンスデータは最適化状態に応じて以下のどれかになります。 ・ <OPT,0> : デフォルト、最適化されていない ・ <OPT,1> : 最適化が進行中 ・ <OPT,2> : シンボル最適化	<OPT,0>
	<?>	コードリーダの状態を示す 16 進値を取得します。コードリーダに発生しているエラー情報や読み取りサイクル状態が取得できます。詳細は、V460-H ハンディ DPM コードリーダ ユーザーズマニュアル (SDNC-768) の「付録 A-16 ユーティリティ」を参照して下さい。	<?/02> (コードリーダ上にエラーは発生していないことがわかります。)
	<K?>	全ての K コマンドに対する設定状態を取得します。	全ての K コマンドの設定状態
	<K??>	全ての K コマンドのパラメータ説明を取得します。	全ての K コマンドのパラメータ説明
	<K?#>	全ての K コマンドのパラメータ設定範囲を取得します。	全ての K コマンドのパラメータ設定範囲
	<Knnn?> (nnn:各 K コマンドの 3 桁の数値)	指定した K コマンドの設定状態を取得します。	<K541,2000,30> (<K541?>に対するレスポンスデータ例です。露光時間: 2000us、ゲイン: 30 であることがわかります。)
	<Knnn??> (nnn:各 K コマンドの 3 桁の数値)	指定した K コマンドのパラメータ説明を取得します。	<K541??,Exposure,Gain> (<K541??>に対するレスポンスデータ例です。パラメータ 1 が Exposure(露光時間)、パラメータ 2 が Gain(ゲイン)であることがわかります。)
	<Knnn?#> (nnn:各 K コマンドの 3 桁の数値)	指定した K コマンドのパラメータ設定範囲を取得します。	<K541?#,Value 25-100000:Def=2500,Value 0-100:Def=33> (<K541?#>に対するレスポンスデータ例です。パラメータ 1 の設定範囲が 25 ~ 100000、初期値: 2500、パラメータ 2 の設定範囲が 0 ~ 100、初期値: 33 であることがわかります。)
	<Knnn?*> (nnn:各 K コマンドの 3 桁の数値)	<Knnn?>コマンド、<Knnn??>コマンド、<Knnn?#>コマンドをそれぞれ実行したときと同じレスポンスデータを取得します。	<K541,2000,30> <K541??,Exposure,Gain> <K541?#,Value 25-100000:Def=2500,Value 0-100:Def=33> (<K541?*>に対するレスポンスデータ例です。)
デバイスコントロール	<I1>	ターゲットパターン (青色 LED) を ON します。	なし
	<I0>	ターゲットパターン (青色 LED) を OFF します。	なし
	<I>	読み取りサイクルを無効にします。読み取りサイクルが無効の間はトリガを受け付けません。	なし

種別	コマンド	説明	レスポンスデータ例 (読み取り文字列は「ABCDE」 とします。)
カウンタとカウンタリセット	<H>	読み取りサイクルを有効にします。	なし
	<q>	読み取りサイクル中の読み取り失敗回数を取得します。レスポンスデータは、q/に続いて読み取り失敗回数が9桁の数値で出力されます。	<q/000000005>
	<q0>	読み取りサイクル中の読み取り失敗回数をクリアします。	なし
	<\$>	ミスマッチ回数を取得します。レスポンスデータは、\$/に続いてミスマッチ回数が9桁の数値で出力されます。	<\$/000000002>
	<\$0>	ミスマッチ回数をクリアします。	なし
	<N>	読み取り失敗回数を取得します。レスポンスデータは、N/に続いて読み取り失敗回数が9桁の数値で出力されます。	<N/000000005>
	<O>	ミスマッチ回数をクリアします。	なし
	<T>	トリガ入力回数を取得します。レスポンスデータは、T/に続いてトリガ入力回数が9桁の数値で出力されます。	<T/000000010>
	<U>	トリガ入力回数をクリアします。	なし
	<V>	マッチコード機能をつかった文字列一致(マッチ)回数を取得します。レスポンスデータは、V/に続いて文字列一致回数が9桁の数値で出力されます。	<V/000000010>
	<W>	文字列一致(マッチ)回数をクリアします。	なし
	<X>	マッチコード機能をつかった文字列不一致(ミスマッチ)回数を取得します。レスポンスデータは、X/に続いて文字列不一致回数が9桁の数値で出力されます。	
	<Y>	文字列不一致(ミスマッチ)回数をクリアします。	なし
ファームウェアバージョン確認	<#>	ファームウェアバージョンの全情報を取得します。	<#b/ 35-9000033-122.3021>< #a/ 35-9000097-1.2.3.3008>< #w/ 30-9000079-1.2.3.3006>< #p/N/A><#d/35-xxxxxx- x.x.x.xxxx>
	<#a>	アプリケーションソフトウェアのバージョン情報を取得します。	<#a/ 35-9000097-1.2.3.3008>
	<#b>	ブートソフトウェアのバージョン情報を取得します。	<#b/ 35-9000033-122.3021>
	<#w>	WebLink _{HH} のバージョン情報を取得します。	<#w/ 30-9000079-1.2.3.3006>
	<!>	アプリケーションソフトウェアとブートソフトウェアのチェックサム値を取得します。	<!b/38B7><!a/9555>
	<!a>	アプリケーションソフトウェアのチェックサム値を取得します。	<!a/9555>
	<!b>	ブートソフトウェアのチェックサム値を取得します。	<!b/38B7>

種別	コマンド	説明	レスポンスデータ例 (読み取り文字列は「ABCDE」 とします。)
設定保存／初期化／再起動	<Z>	現在の設定を本体に保存し、再起動します。	<A?/0>
	<Zc>	現在の設定をカスタムデフォルト設定として本体に保存し、再起動します。	<A?/0>
	<Zrc>	コードリーダ設定をカスタムデフォルトに初期化後、再起動します。	<A?/0>
	<Zrd>	コードリーダ設定を工場集荷状態(通信設定やユーザ定義名を除く)に初期化後、再起動します。	<A?/0>
	<Zrdall>	コードリーダ設定を工場出荷状態に初期化し、再起動します。	<A?/0>1>
	<A>	現在の設定を保持したまま再起動します。	<A?/0>
	<Ard>	コードリーダ設定を工場出荷状態(通信設定とユーザ定義名を除く)に戻して再起動します。	<A?/0>
	<Arp>	コードリーダ設定を前回保存された状態に戻して再起動します。	<A?/0>
	<Arc>	コードリーダ設定をカスタムデフォルトに初期化後、再起動します。	<A?/0>
マスターデータベース	<G>	マスターデータベースに登録するデータベース番号を 1 に設定します。	なし
	<Gn> n: マスターデータベースのインデックス番号	マスターデータベースに登録するデータベース番号を n に設定します。	<NEWM/01> (次に読み取り成功した結果がマスターデータベースの 1 番に登録されます。)
	<NEWM>	登録対象のデータベース番号を取得します。登録するデータベース番号が未指定状態の場合は<NEWM/00>が返ります。	なし
バーコードコンフィグレーション	<BCCFG>	K コマンドがデータ化された DataMatrix を読み取り設定が可能になるバーコードコンフィグレーションモードに遷移します。詳細は『V460-H ハンディ DPM コードリーダ ユーザーズマニュアル (SDNC-768)』の「付録 A-16 ユーティリティ」を参照して下さい。	なし

*1. ヘッダ／フッタは、コードリーダの設定に依存せず、ヘッダ：なし、フッタ：CRLF 固定になります。

PROFINET で制御／出力する

本章では、V460-H シリーズのリーダーを NJ/NX シリーズマシンオートメーションコントローラ（以降、コントローラ）と PROFINET IO を介して接続する方法、および装置接続の確認方法について説明します。本章の設定を実行すると、ユーザは、PROFINET の入出力モジュールデータを見たり、出力モジュールに変更を加えたり、それらの変更を入力モジュールで、確認することができるようになります。本章に記載している例には、入出力モジュールを接続すること以外、PLC のプログラム、独自データ構造、または設定については記載していません。V460-H へのデータアクセスが確立されたら、ユーザの責任で、コントローラをプログラムしてください。

3-1	PROFINET の概要	3-2
3-1-1	PROFINET の種類	3-2
3-2	PROFINET 接続による本コードリーダーとの通信	3-5
3-2-1	通信エリアの種類	3-5
3-3	PROFINET の通信設定	3-6
3-3-1	コードリーダーのネットワーク設定	3-6
3-4	アセンブリ別タイミングチャート	3-9
3-4-1	読み取り実行信号（TRIG 信号）による読み取り実行	3-9
3-5	サンプルラダープログラム	3-11

3-1 PROFINET の概要

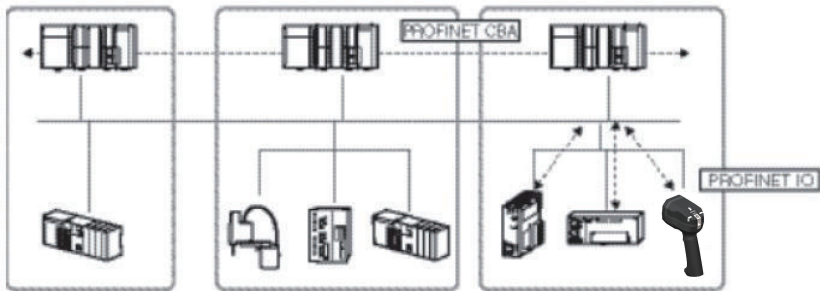
PROFINET は、産業用 Ethernet（100 Mbps、全二重）を、PROFIBUS-DP に適用する産業用ネットワークです。

PROFINET は、オープンな標準で、PI（プロフィバスと PROFINET インターナショナル）により管理されていて、様々な種類の産業用機器に使用されています。PPROFINET は、標準通信仕様である Ethernet を使用しているので、様々な汎用 Ethernet 機器をネットワークに接続することができます。本章では、PROFINET を使用して本コードリーダを使用する概要を説明します。

PROFINET 仕様の詳細については、IEC 61158 と IEC 61784 規格、および PI を参照してください。

3-1-1 PROFINET の種類

- PROFINET には、PROFINET CBA と PROFINET IO という、2 種類の規格があります。
- PROFINET CBA
コンポーネントを使用するデバイス間の通信です。おもに、コントローラ間で使用されます。
 - PROFINET IO
コントローラとデバイス間を I/O データで制御します。



本コードリーダは、PROFINET IO に対応しています。PROFINET は、PROFINET DP として、同じデバイスモデルを使用します。

各装置の情報は、XSL（拡張マークアップ言語）をもとにした GSD（General Station Description）ファイルに記述されています。

PROFINET IO の通信仕様

PROFINET IO の通信仕様を、以下に説明します。

通信仕様	種類	説明	対応／未対応
周期的データ通信方法	RT（リアルタイム）通信	標準の Ethernet ハードウェアを使用し、既存のフィールドバスと同等の性能を提供します。	対応
	IRT（アイソクロナス・リアルタイム）通信	この方法は、RT より高いレベルで通信を提供します。通信は、ある特定の時間内に実行されます。モーションコントロールのような、厳しい実時間性が要求されるシステムで使用されることを想定しています。	未対応

PROFINET IO は、使用目的を考慮して適合クラスにより対応する機能を明記しています。

クラス	概要	対応／未対応
クラス A	RT 通信の基本的な機能です。	対応

クラス	概要	対応／未対応
クラス B	このクラスは、プロセスオートメーションや、その他のアプリケーションで使用するネットワーク診断と冗長性を追加しています。	未対応
クラス C	信頼できる同期制御（IRT 通信）に対応しています。	未対応

以下の機能が、クラス A に定義されています。

機能	概要
周期データ交換	指定した周期の IO コントローラと IO デバイス間のリアルタイムデータ通信。IO データ CR で設定します。
非周期パラメータデータ交換／デバイス ID	パラメータ設定、IO デバイス設定、およびデバイス情報の読み取りに使用します。レコード・データ CR により設定します。
デバイス／ネットワーク診断	IO デバイスから IO コントローラへ、アラームと状態を送信することを目的とする通信。アラーム CR により設定します。

以下の機能は、クラス B で定義されており、クラス A を拡張しています。

機能	概要
SNMP（シンプルネットワーク管理プロトコル）	管理情報ベース 2（Management Information Base 2, MIB2）と下位リンク層検出プロトコル MIB（Lower Link Layer Discovery Protocol-MIB, LLDP-EXT-MIB）を介して、追加のネットワーク診断を許可します。
PDEV（物理デバイス・オブジェクト）	非周期 PROFINET サービスを使用して、診断情報を収集することができます。

PROFINET IO で使用されるデバイスタイプ

以下のデバイスが、PROFINET IO で定義されています。

種別	種別
IO コントローラ	外部および他のデバイス用のコントローラ
IO デバイス	IO コントローラに接続されている本コードリーダー
IO スーパーバイザ	保守と診断に使用される PC、または他のデバイス

IO デバイス

IO デバイスは、DAP と IO モジュールで構成されています。

これらのデバイスの機能と特性は、GSD ファイルに記述されています。

- **DAP（デバイスアクセスポイント）**：これは、Ethernet アクセスポイントで、通信プログラムにより使用されます。
- **IO モジュール**：以下のスロット、サブスロット、およびインデックスで構成されています。ひとつの IO モジュールは、ひとつまたは複数のスロットで構成されます。
- **スロット**：IO デバイス内の、IO モジュールの位置を示しています。
- **サブスロット**：スロット内部の IO インタフェースです。ビットデータ、バイトデータおよびデータ型の意味などのデータ型を定義しています。
- **インデックス**：サブスロット内のデータです。

上記の情報は、本コードリーダーの GSD ファイルに記述してあります。IO コントローラは、システムを構築するために、本コードリーダーの GSD ファイルを使用します。



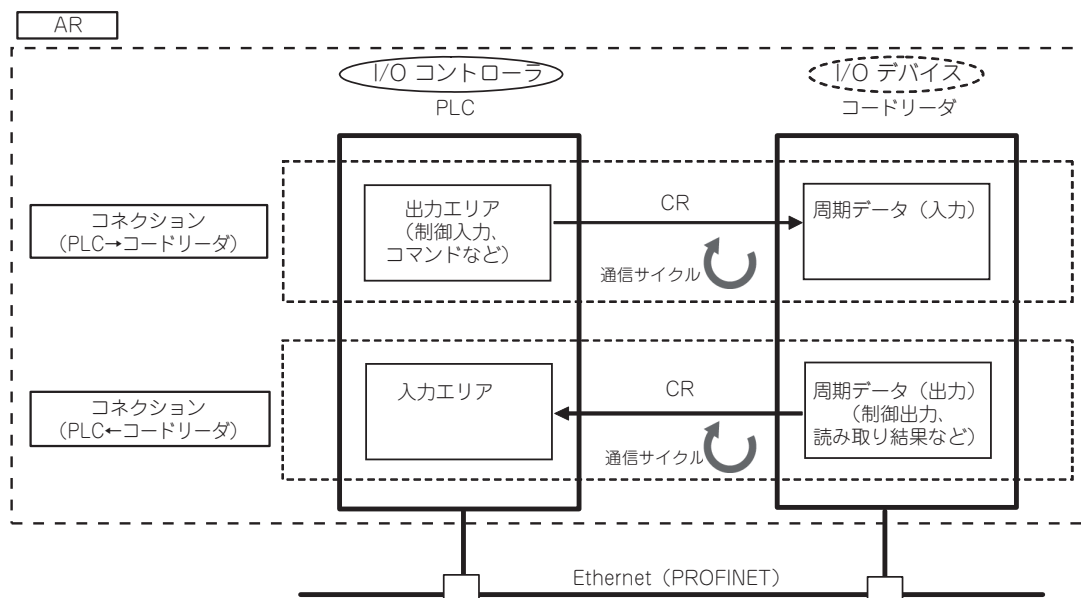
参考

IO デバイスを PROFINET で使用する場合、デバイスの機能と特性を記述してある GSD ファイルを、ネットワーク構成を設定するために使用します。

本コードリーダーを IO デバイスとして PROFINET で使用する場合、本コードリーダーの GSD ファイルが、エンジニアリングツールにインストールされていなければなりません。

PROFINET IO によるデータ通信

IO コントローラと IO デバイスが通信するためには、AR（アプリケーションリレーション）という通信パスが、ふたつのデバイス間で最初に確立されていなければなりません。AR 接続が確立されている場合、IO コントローラと IO デバイス間のデータ通信は、CR（コミュニケーションリレーション）により実行され、データ通信の内容により定義されます。IO デバイスは、複数の通信デバイスと AR を確立することができます。また、複数の CR を、ひとつの AR の内部に定義することもできます。複数の CR を、ひとつの AR の内部に定義することで、複数のプロファイルまたは異なるサブスロットを必要とする AR 通信を実行することができます。また、各 CR や IO 用のサイクル時間を設定することもできます。



CR は、IO データ CR（周期データ交換）、レコードデータ CR（非周期データ交換）、およびアラーム CR に大別されます。IO データ CR 内で、データ通信を各更新タスク期間中に実行します。IO データ CR 以外の CR 内で、通信を周期データ通信の間に実行します。レコードデータ CR 内で、IO コントローラは、いつでも IO デバイスにコマンドを送信できます。IO デバイスは、IO コントローラに応答を返します。

3-2 PROFINET 接続による本コードリーダーとの通信

PLC とコードリーダー間で通信するために、PROFINET の IO データ CR を使用し、コマンド／レスポンス通信で制御を実行したり、計測後にデータを出力したりできます。

本コードリーダーは、PROFINET のクラス A に対応しています。

外部デバイスと接続し、PROFINET により通信するためには、エンジニアリングツールで、PROFINET IO データ CR を設定します。

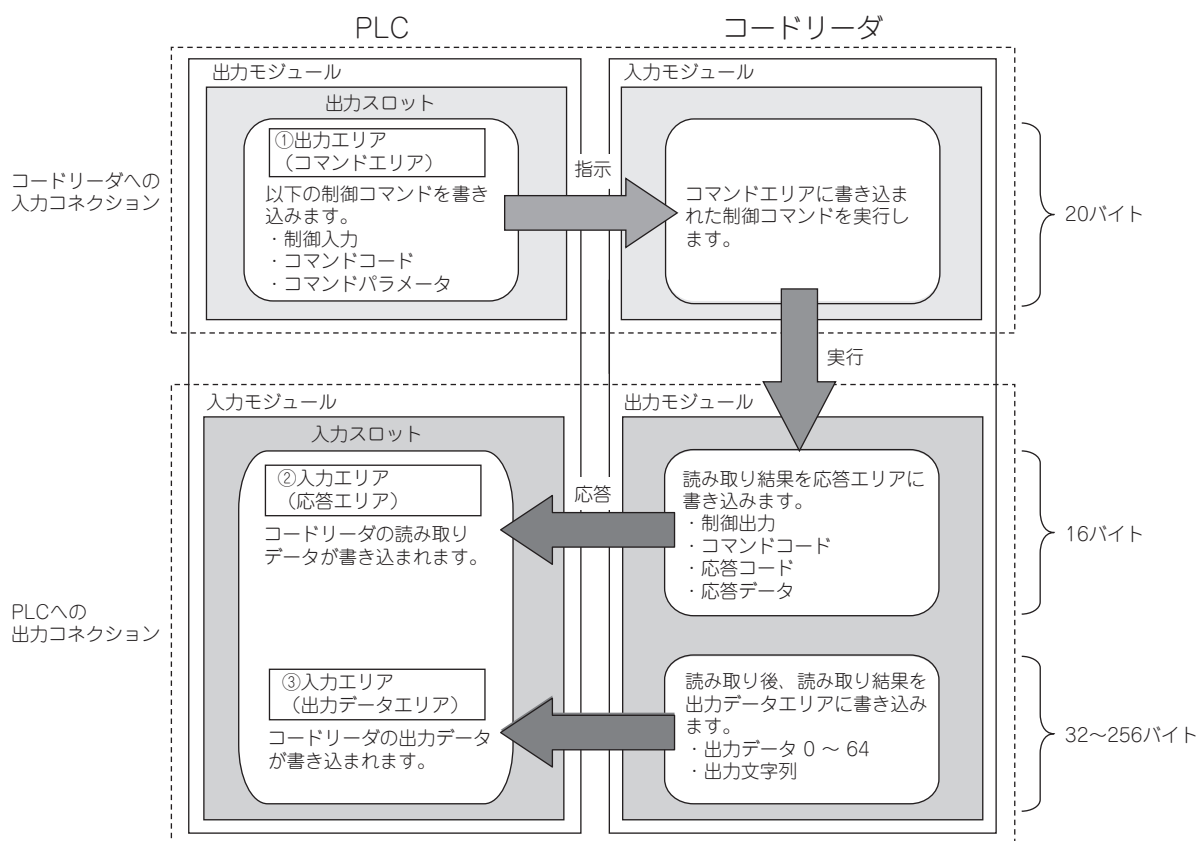
エンジニアリングツールの IO データ CR 設定については、各エンジニアリングツールのマニュアルを参照してください。

3-2-1 通信エリアの種類

PROFINET 通信では、以下の三つの通信エリアを、PLC が使用して通信を行います。

コマンド／レスポンス通信	(1) 出力エリア (コマンドエリア)	本コードリーダーで実行する制御コマンドを、ユーザが入力するエリアです。
	(2) 入力エリア (応答エリア)	コマンドエリアから実行された制御コマンドの結果を、本コードリーダーが入力するエリアです。
計測後データ出力	(3) 入力エリア (出力データエリア)	検査を実行後、計測値の出力データを、本コードリーダーが入力するエリアです。

(2) 入力エリア (応答エリア) と (3) 入力エリア (出力データエリア) は、連続したメモリアドレスまたは変数に割り当てられます。



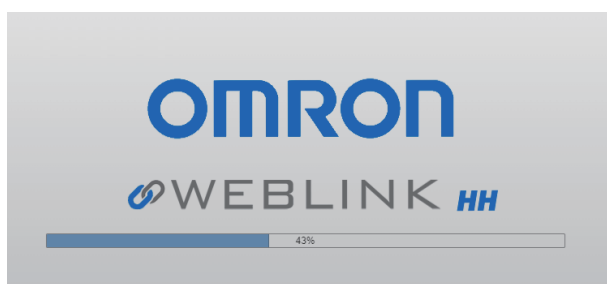
3-3 PROFINET の通信設定

3-3-1 コードリーダーのネットワーク設定

- 1 ブラウザを起動し、<http://192.168.188.2> と入力します。
推奨ブラウザは、Google Chrome です。



- 2 WebLink_{HH} 起動画面が、表示されます。



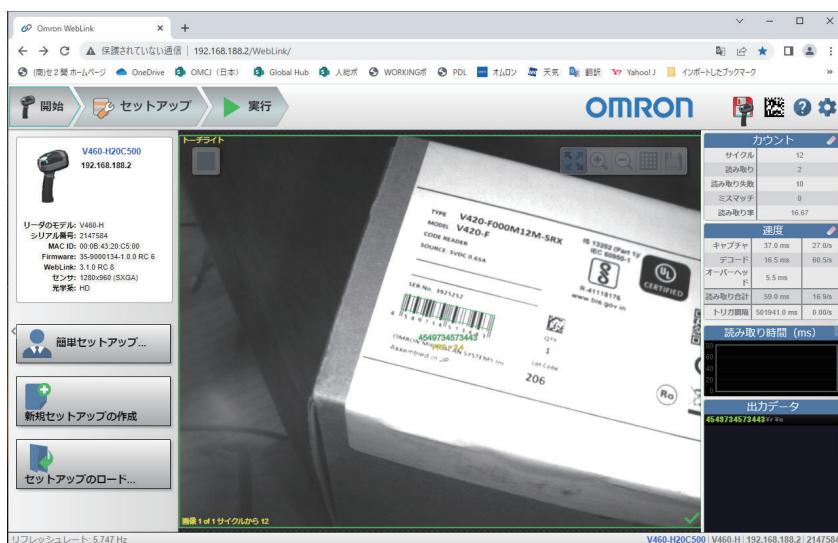
- 3 WebLink_{HH} 起動画面が表示されない場合、コードリーダーとパソコン間の通信が確立していないことが考えられます。以下を確認してください。

- ・コードリーダーとパソコン間が、適切なケーブルで接続されていますか？
- ・パソコンとコードリーダーの IP アドレスが、正しく設定されていますか？

パソコンの IP アドレスを設定し、コードリーダーをハードウェアリセットしてください。

その他の方法については、『V460-H ハンディ DPM コードリーダー ユーザーズマニュアル (SDNC-768)』付録の「WebLink_{HH} に接続できない場合の対処方法」を参照してください。

- 4 以下の WebLink_{HH} 画面が表示されます。



- 5 [セットアップ] タブをクリックし、「サイクル」を「単発読み取り」に設定します。



- 6 画面右上部にある歯車アイコンをクリックし、アプリケーション設定メニューの「詳細」を選択します。

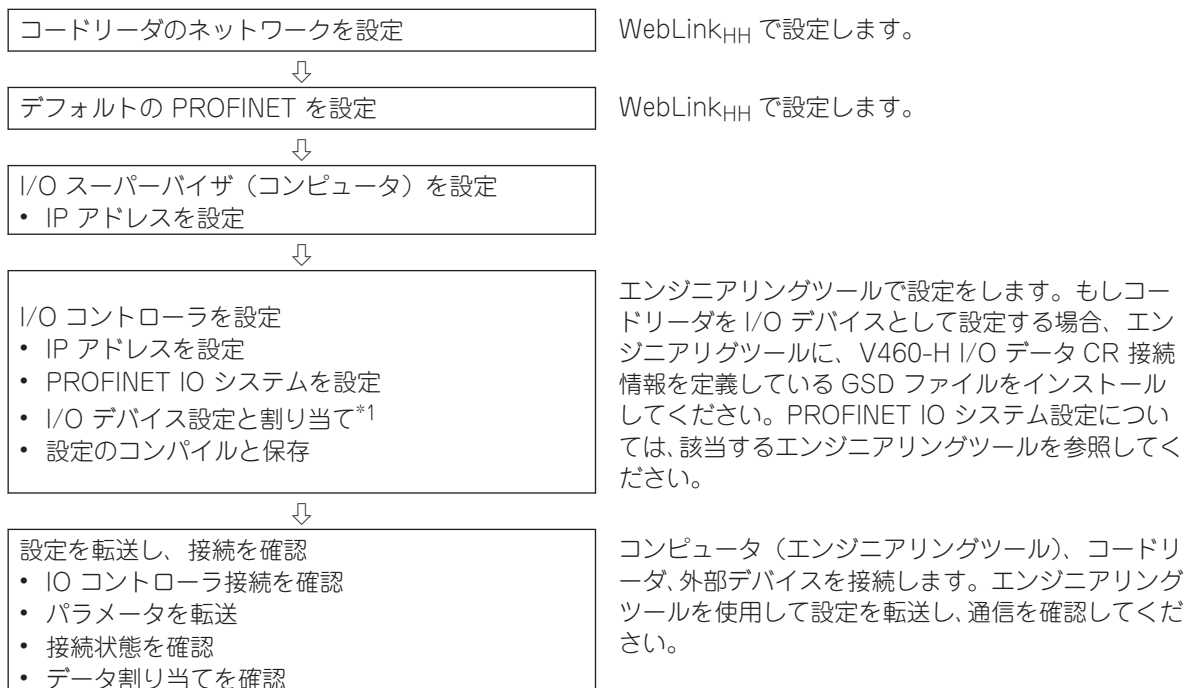


- 7 「詳細設定」画面が、表示されます。赤い四角で囲まれている設定を確認してください。
デフォルトでは、[EtherNet/IP] が [有効] です。EtherNet/IP を無効とし、[PROFINET] を [有効] にします。
複数のコードリーダーを接続するなど、IP アドレスを変更する必要がある場合、アプリケーションで必要な [IP アドレス] を設定します。



通信設定手順

PROFINET による通信を使用するためには、以下の設定を行う必要があります。



- *1. 下記のように、V460-H の IP アドレスが PLC プロジェクトで設定されている場合、V460-H の電源が入っているときに PLC がネットワークに接続されていないと、PLC がネットワークに再接続されるまで、デフォルトの IP アドレス “192.168.188.2” に戻ります。
下図はオムロン製 PLC 用の PROFINET 構成ページを示していますが、他社の PLC でも同様です。



Device: CJ1W-PNT21

Category: dmnSpecific

© Copyright OMRON Corporation 2012-2022. All Rights Reserved.

PROFINET Identification

Network Scan

Configuration

PLC Setup

IO Controller Setup

IO Device Setup

IO Device Area

IO Device Configuration

Watchdog Factor: 3

Data Hold Factor: 3

Device No.	Device Name	Device Type	IP Address	Update Rate (ms)	Watchdog Time (ms)	Data Hold Time (ms)
1	V460	V460-H	192.168.188.50	16	48	48

Diagnosis

Firmware

DTM Information

メモリ割り当て

入出力モジュールの定義については、「A-3 PROFINET - V460-H 入出力モジュール (P.A-27)」を参照してください。

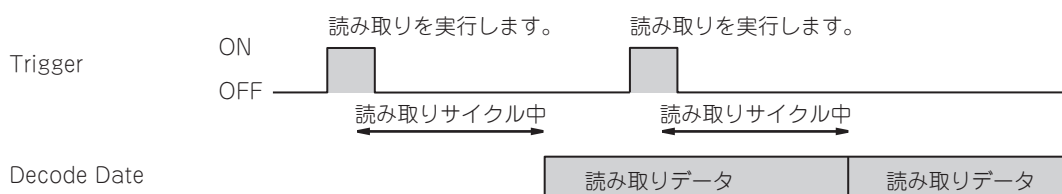
3-4 アセンブリ別タイミングチャート

3-4-1 読み取り実行信号 (TRIG 信号) による読み取り実行

読み取りデータを、PLC のデータメモリに格納完了したときのタイミング信号は、入力モジュール型により異なります。

小型入力モジュール (100)

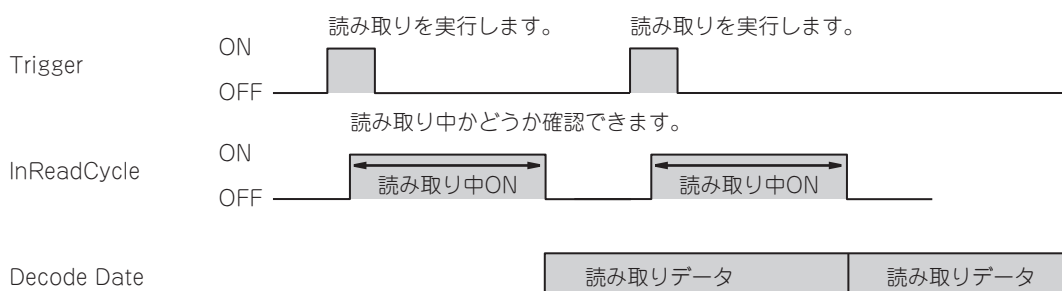
読み取りデータ格納用の、タイミング信号に該当するものではありません。



1. 「Trigger」の立ち上がりエッジで、読み取りを開始します。
2. 読み取り終了時に、読み取りデータが「Decode Data」に格納されます。

大型入力モジュール (101)

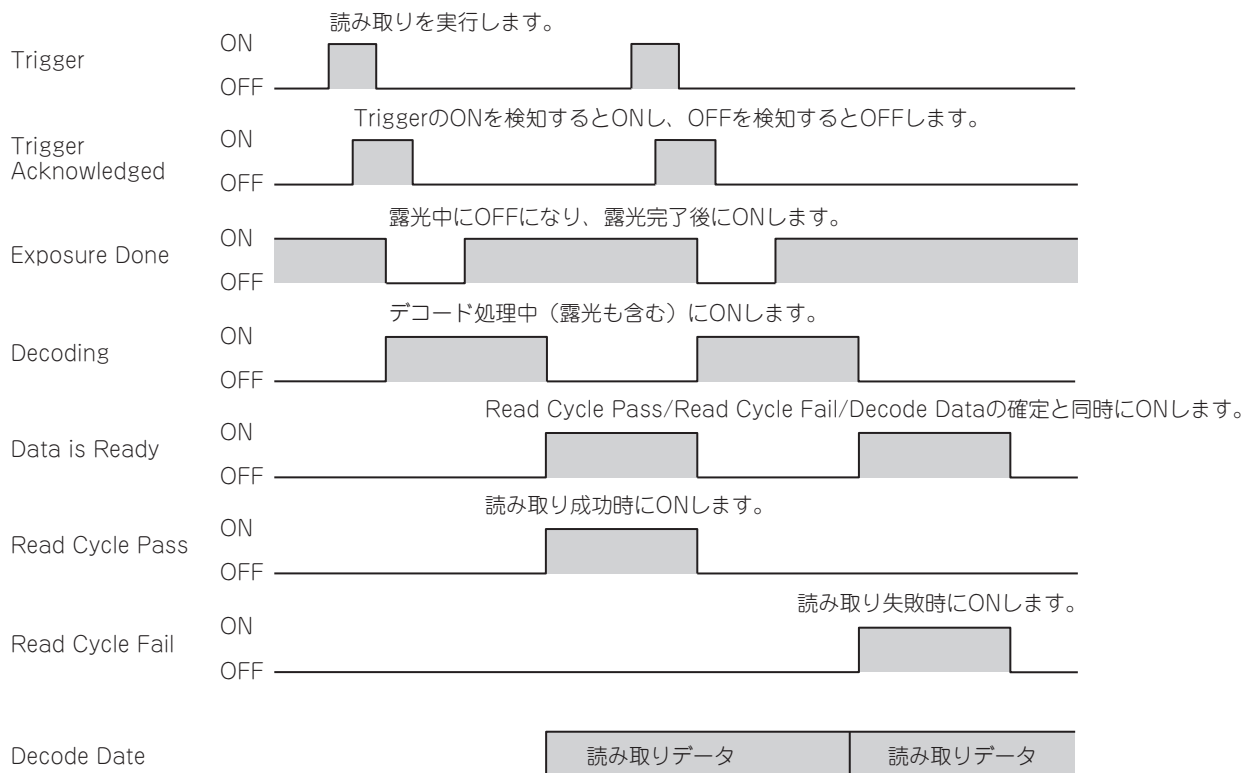
「Device Status」の「InReadCycle」が、ON から OFF に変化したタイミングで出力します。



1. 「Trigger」の立ち上がりエッジで、読み取りを開始します。
2. 読み取り開始時点で、「InReadCycle」が ON になり、「Trigger」が OFF になります。
3. 読み取り終了時に、読み取りデータが「Decode Data」に格納され、「InReadCycle」が OFF になります。

MXL 入力モジュール (102)

「Device Status」の「Decoding」ビットが、ON から OFF になったタイミングで出力します。



1. [Trigger] の立ち上がりエッジで、読み取りを開始します。
2. [Trigger Acknowledged] は、Trigger の ON を検出すると ON になり、Trigger の OFF を検出すると OFF になります。
3. 露光開始で、[Exposure Done] が OFF になり、露光終了で ON になります。
4. [Decoding] は、デコード処理中に ON になります。デコード処理は、露光処理と重複します。
5. Decode Data、Read Cycle Pass もしくは Read Cycle Fail の確定と同時に、[Data is Ready] が ON になります。
6. 読み取りに成功すると、[Read Cycle Pass] が ON になり、読み取りに失敗すると [Read Cycle Fail] が ON になります。読み取りデータが、[Decode Data] に格納されます。



参考

シンボルデータの出力タイミングには、10ms 程度の遅延が発生することがあります。

7. 次の [Trigger] を検出すると、[Data is Ready] は OFF になります。

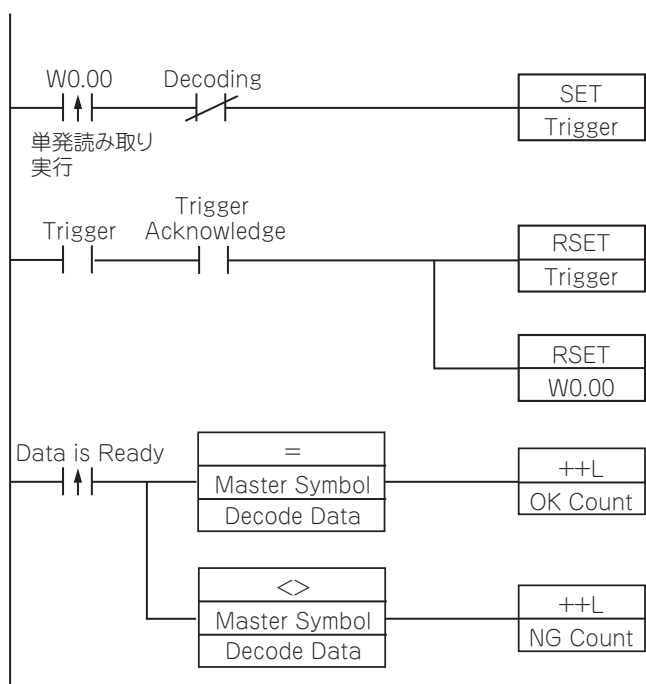
3-5 サンプルラダープログラム

サンプルラダープログラムを、以下に示します。

- Trigger 信号を入力し、単発読み取りを実行します。
- 読み取った文字列 (Decode Data) を、照合用文字列 (Master Symbol) と比較します。
- 一致したら、Match/OK Count に追加し、不一致なら Mismatch/NG Count に追加します。

以下の、入力・出力モジュールを使用します。

- 入力モジュール：MXL 入力 モジュール(102)
- 出力モジュール：出力モジュール (197)



1. 単発読み取り実行フラグが ON になると、Trigger ビットが ON になります。
2. トリガ入力検出用の Trigger Acknowledged ビットが ON になります。
3. Trigger Acknowledged の ON を検出すると、Trigger が OFF になります。
4. 読み取りが完了すると、Data is Ready ビットが ON になります。
5. 読み取り文字列 (Decode Data) を、照合用文字列 (Master Symbol) と比較します。
6. 二つの文字列が一致すると、Match/OK Count を+1 します。
7. 二つの文字列が不一致の場合、Mismatch/NG Count を+1 します。



付録

この章では、コードリーダーで利用できる産業用通信プロトコルについて説明します。

A

A-1	コマンド一覧	A-2
A-1-1	コマンド一覧	A-2
A-2	EtherNet/IP 通信仕様詳細	A-3
A-2-1	ファームウェアバージョンごとの EDS ファイル	A-3
A-2-2	メモリ割り当て	A-3
A-3	PROFINET - V460-H 入出力モジュール	A-27
A-3-1	モジュール種別	A-27
A-3-2	データ型	A-29
A-3-3	PROFINET 基本情報	A-30
A-3-4	タイミングチャート	A-31

A-1 コマンド一覧

A-1-1 コマンド一覧

本コードリーダーで使用可能なコマンドの一覧と、通信プロトコルごとの使用可否を以下に示します。
○：使用できるコマンド、-：使用できないコマンド

機能	シリアル (TCP)	EtherNet/IP
設定を変更する	○	_ *1
読み取りを実行する	○	○
読み取り回数テストを開始する	○	-
読み取り率テストを開始する	○	-
読み取り回数／読み取り率テストを終了する	○	-
キャリブレーションを実行する	○	-
ティーチング操作を行う	○	-
最適化操作を行う	○	-
コードリーダーのエラー情報を取得する	○	○
設定を取得する	○	_ *1
ターゲットパターン（青色 LED）を ON/OFF する	○	○
読み取りサイクルを有効／無効にする	○	○
カウンタ値を取得する	○	○
カウンタ値をリセットする	○	○
バージョン情報を取得する	○	-
設定を本体に保存する	○	-
コードリーダーの設定初期化を実行する	○	-
コードリーダーを再起動する	○	-
読み取り結果をマスタデータベースに登録する	○	○
コード品質評価結果のレポートを取得する	○	-

*1. EtherNet/IP のメッセージ通信によるシリアルコマンド送信では使用可能です。

A-2 EtherNet/IP 通信仕様詳細

A-2-1 ファームウェアバージョンごとの EDS ファイル

シリーズ	ファームウェアバージョン	EDS ファイル	バージョン	Product Code	Device Major Rev	Device Minor Rev
V460-H	1.0.0.xxxx	V460-H_1_0_0_20211119.eds	1.0	3414	1	1

A-2-2 メモリ割り当て

各入力アセンブリ（本コードリーダー→PLC）、出力アセンブリ（PLC→本コードリーダー）のメモリ割り当てを説明します。

小型入力（インスタンス ID: 100）

小型、軽量な入力アセンブリです。読み取り結果に 64 バイトの情報を保持するように設計されています。複数シンボル読み取りの場合の読み取り文字列は、区切り文字で区切られて出力されます。以下は、小型入力アセンブリのメンバ構成を示す表です。

小型入力 メンバ構成

メンバ名	サイズ（バイト）
USER-DEFINED TAG ECHO	4
COMMAND ECHO	4
RESERVED	4
READ CYCLE SEQUENCE COUNTER	4
DECODE DATA LENGTH	4
DECODE DATA STRING	64

合計サイズ：84 バイト

メンバ詳細

- **User-Defined Tag Echo：ユーザ定義タグエコー**
出力（レガシー）の User-Defined Tag エリアに設定された値を返します。
- **Command Echo：コマンドエコー**
出力（レガシー）の Command エリアに設定された値を返します。
- **Read Cycle Sequence Counter：読み取りサイクル回数**
現在の読み取りサイクル回数が格納されます。
- **Decode Data Length：読み取り文字列長**
読み取り文字列の文字数が格納されます。
- **Decode Data String：読み取り文字列**
読み取り文字列が格納されます。印字品質評価指標などの付加情報を設定している場合は、読み取り文字列に続いて格納されます。

メモリ割り当て

	メンバ名	データ型	ビット番号	データ長	バイトオフセット
32 ビット	User Defined Tag Echo	DINT		4 バイト	0
	UserTag_1		0	1 ビット	
	UserTag_2		1	1 ビット	

	メンバ名	データ型	ビット番号	データ長	バイトオフセット
	UserTag_3		2	1 ビット	
	UserTag_4		3	1 ビット	
	UserTag_5		4	1 ビット	
	UserTag_6		5	1 ビット	
	UserTag_7		6	1 ビット	
	UserTag_8		7	1 ビット	
	UserTag_9		8	1 ビット	
	UserTag_10		9	1 ビット	
	UserTag_11		10	1 ビット	
	UserTag_12		11	1 ビット	
	UserTag_13		12	1 ビット	
	UserTag_14		13	1 ビット	
	UserTag_15		14	1 ビット	
	UserTag_16		15	1 ビット	
	UserTag_17		16	1 ビット	
	UserTag_18		17	1 ビット	
	UserTag_19		18	1 ビット	
	UserTag_20		19	1 ビット	
	UserTag_21		20	1 ビット	
	UserTag_22		21	1 ビット	
	UserTag_23		22	1 ビット	
	UserTag_24		23	1 ビット	
	UserTag_25		24	1 ビット	
	UserTag_26		25	1 ビット	
	UserTag_27		26	1 ビット	
	UserTag_28		27	1 ビット	
	UserTag_29		28	1 ビット	
	UserTag_30		29	1 ビット	
	UserTag_31		30	1 ビット	
	UserTag_32		31	1 ビット	
32 ビット	Command Echo	DINT		4 バイト	4
	Trigger_Echo		0	1 ビット	
	New Master Echo		1	1 ビット	
	Reserved		2 - 7	6 ビット	
	Disable Scanning Echo		8	1 ビット	
	Reserved		9 - 15	7 ビット	
	Clear Read Cycle Report and Counters Echo		16	1 ビット	
	Reserved		17 - 31	15 ビット	
32 ビット	Reserved	DINT		4 バイト	8
32 ビット	Read Cycle Sequence Counter	UDINT	0 - 31	4 バイト	12
32 ビット	Decode Data Length	UDINT	0 - 31	4 バイト	16
	Decode Data String	SINT[64]	0 - 512	64 バイト	20

大型入力（インスタンス ID: 101）

大型入力は、小型入力に比べ、より多くのデバイスステータス情報、128 バイトの読み取り結果文字列を保持します。複数シンボル読み取りの場合の読み取り文字列は、区切り文字で区切られて出力されます。

大型入力 メンバ構成

メンバ名	サイズ (バイト)
USER-DEFINED TAG ECHO	4
COMMAND ECHO	4
RESERVED	4
RESERVED	4
RESERVED	4
DEVICE STATUS (Legacy)	4
READ CYCLE SEQUENCE COUNTER	4
TRIGGER COUNT	4
DECODE/MATCH COUNT	4
MISMATCH COUNT	4
NOREAD COUNT	4
DECODE DATA LENGTH	4
DECODE DATA STRING	128

合計サイズ：176 バイト

メンバ詳細

- **User-Defined Tag Echo：ユーザ定義タグエコー**
出力（レガシー）の User-Defined Tag エリアに設定された値を返します。
- **Command Echo：コマンドエコー**
出力（レガシー）の Command エリアに設定された値を返します。
- **Device Status (Legacy)：デバイスステータス（レガシー）**
コードリーダの状態を示します。

ビット	状態	説明
0	Reserved	-
1	New Master Requested	ビット ON 時、次の読み取り結果をマスタシンボルとして登録します。
2 - 7	Reserved	-
8	Scanning Disabled	ビット ON 時、読み取りサイクル無効状態です。
9 - 15	Reserved	-
16	In read cycle	読み取りサイクル中にビットが ON します。
17	Actively Scanning	読み取りサイクル中にビットが ON します。

- **Read Cycle Sequence Counter：読み取りサイクル回数**
現在の読み取りサイクル回数が格納されます。
- **Trigger Counter：トリガ回数**
現在の入力されたトリガの総数が格納されます。
- **Decode/Matchcode Counter：読み取り成功／一致コード回数**
以下のいずれかが格納されます。
(1)読み取り成功した総数（マッチコードが無効である場合）
(2)マスタシンボルに一致した総数（マッチコードが有効である場合）
- **Mismatch Counter：不一致回数**
ミスマッチ（マスタシンボルに一致しなかった）した総数が格納されます。
- **No Read Counter：読み取り失敗回数**
読み取り失敗した総数が格納されます。
- **Decode Data Length：読み取り文字列長**
読み取り文字列の文字数が格納されます。
- **Decode Data String：読み取り文字列**

読み取り文字列が格納されます。印字品質評価指標などの付加情報を設定している場合は、読み取り文字列に続いて格納されます。

メモリ割り当て

	メンバ名	データ型	ビット番号	データ長	バイトオフセット
32 ビット	User Defined Tag Echo	DINT		4 バイト	0
	UserTag_1		0	1 ビット	
	UserTag_2		1	1 ビット	
	UserTag_3		2	1 ビット	
	UserTag_4		3	1 ビット	
	UserTag_5		4	1 ビット	
	UserTag_6		5	1 ビット	
	UserTag_7		6	1 ビット	
	UserTag_8		7	1 ビット	
	UserTag_9		8	1 ビット	
	UserTag_10		9	1 ビット	
	UserTag_11		10	1 ビット	
	UserTag_12		11	1 ビット	
	UserTag_13		12	1 ビット	
	UserTag_14		13	1 ビット	
	UserTag_15		14	1 ビット	
	UserTag_16		15	1 ビット	
	UserTag_17		16	1 ビット	
	UserTag_18		17	1 ビット	
	UserTag_19		18	1 ビット	
	UserTag_20		19	1 ビット	
	UserTag_21		20	1 ビット	
	UserTag_22		21	1 ビット	
	UserTag_23		22	1 ビット	
	UserTag_24		23	1 ビット	
	UserTag_25		24	1 ビット	
	UserTag_26		25	1 ビット	
	UserTag_27		26	1 ビット	
	UserTag_28		27	1 ビット	
	UserTag_29		28	1 ビット	
	UserTag_30		29	1 ビット	
	UserTag_31		30	1 ビット	
	UserTag_32		31	1 ビット	
32 ビット	Command Echo	DINT		4 バイト	4
	Trigger Echo		0	1 ビット	
	New Master Echo		1	1 ビット	
	Reserved		2 - 7	6 ビット	
	Disable Scanning Echo		8	1 ビット	
	Reserved		9 - 15	7 ビット	
	Clear Read Cycle Report and Counters Echo		16	1 ビット	
	Reserved		17 - 31	15 ビット	
32 ビット	Reserved	DINT		4 バイト	8
32 ビット	Reserved	DINT		4 バイト	12
32 ビット	Reserved	DINT		4 バイト	16

	メンバ名	データ型	ビット番号	データ長	バイトオフセット
32 ビット	Device Status	DINT		4 バイト	20
	Reserved		0	1 ビット	
	New Master Requested		1	1 ビット	
	Reserved		2 - 7	6 ビット	
	Scanning Disabled		8	1 ビット	
	Reserved		9 - 15	7 ビット	
	In Read Cycle		16	1 ビット	
	Actively Scanning		17	1 ビット	
	Reserved		18 - 31	14 ビット	
32 ビット	Read Cycle Sequence Counter	UDINT	0 - 31	4 バイト	24
32 ビット	Trigger Count	UDINT	0 - 31	4 バイト	28
32 ビット	Decode/Matchcode Count	UDINT	0 - 31	4 バイト	32
32 ビット	Mismatch Count	UDINT	0 - 31	4 バイト	36
32 ビット	No Read Count	UDINT	0 - 31	4 バイト	40
32 ビット	Decode Data Length	UDINT	0 - 31	4 バイト	44
	Decode Data String	SINT[128]	0 - 1024	128 バイト	48

MXL/SLC 入力（インスタンス ID: 102）

MXL/SLC 入力は、大型入力に比べ、より詳細なデバイスステータス情報、184 バイトの読み取り結果文字列を保持します。複数シンボル読み取りの場合の読み取り文字列は、区切り文字で区切られて出力されます。

MXL/SLC 入力 メンバ構成

メンバ名	サイズ (バイト)
RESERVED	1
RESERVED	1
RESERVED	1
RESERVED	1
DEVICE STATUS	4
RESERVED	4
COUNTERS	24
READ CYCLE REPORT	8
DECODE CYCLE REPORT	16
DECODE DATA LENGTH	4
DECODE DATA STRING	184

合計サイズ：248 バイト

メンバ詳細

- **Device Status：デバイスステータス**
コードリーダの現在の状態を示します。

ビット	状態
0	Run Mode
1	Trigger Acknowledged
2	Exposure Done
3	Decoding
4	Data Is Ready
5	Read Cycle Pass
6	Read Cycle Fail

ビット	状態
7	General Fault
8	Matchcode Master Label Trained
9	Matchcode Enabled
10	Image Sensor Calibrating
11	Image Sensor Calibration Complete
12	Training
13	Training Complete
14	Optimizing
15	Optimization Complete
16	AutolImage Photometry Enabled
17	AutolImage Photometry Complete
18	Reserved
19	Reserved
20	Reserved
21	Buffer Overflow
22 - 31	Reserved

・ Run Mode：ランモード

読み取りサイクルの有効、無効を示します。

0 = 読み取りサイクル無効状態。トリガを受け付けない状態です。ただし、コマンドを受領することは可能です。

1 = 読み取りサイクル有効状態。トリガを受け付け可能な状態です。

・ Trigger Acknowledged：トリガ受付

出力アセンブリの Trigger ビットを受け付けると、このビットが「1」になります。

Trigger ビットが OFF にされたとき、Trigger Acknowledged も「0」になります。

・ Exposure Done：露光完了

露光中である時、このビットは「0」になります。

露光が完了したとき、このビットは「1」になります。

・ Decoding：デコード中

画像を処理しているとき、このビットは「1」になります。

画像処理を完了すると、このビットは「0」になります。

・ Data is Ready：データ準備 OK

Read Cycle Report と Data Cycle Report のデータが確定した時、このビットは「1」になります。

次の読み取りが開始されると、このビットは「0」になります。

・ Read Cycle Pass：読み取りサイクル OK

読み取り成功時に、このビットは「1」になります。

次の読み取りが開始されると、このビットは「0」になります。

・ Read Cycle Fail：読み取りサイクル失敗

読み取りなしや不一致などの理由で読み取りサイクルが失敗した場合、このビットは「1」になります。読み取りサイクルの開始時に、このビットは「0」にセットされます。

・ General Fault：エラー

コードリーダーにエラーが発生すると、このビットは「1」になります。ユーザは、エラーコード用の Fault Code エリアを参照して、問題を解決する必要があります。問題を解決した後、出力アセンブリの Control 内の Reset General fault を、ユーザにより「0」にする必要があります。

・ Matchcode Master Label Trained：マッチコードマスタラベルティーチング完了

ユニットが最後のトリガで読み取られたデータおよびマッチコード機能で使用する新しいマスタラベルを受け入れた場合、アクティブになります。

・ Matchcode Enabled：マッチコード有効

マッチコード機能が有効である場合、このビットは「1」になります。

・ Image Sensor Calibrating：キャリブレーション中

ユニットは、以下のキャリブレーションを実行中である場合、このビットは「1」に設定されます。

- 露光
- ゲイン

ユニットのキャリブレーションが完了すると、このビットは「0」にセットされます。

- ・ Image Sensor Calibration Complete：キャリブレーション完了

ユニットは、以下のキャリブレーションが完了した場合、このビットは「1」に設定されます。

- 露光
- ゲイン

- ・ Training：ティーチング中

ティーチング中のとき、このビットは「1」にセットされます。

ティーチングが完了すると、このビットは「0」にセットされます。

- ・ Training Complete：ティーチング完了

このビットはティーチング中に「0」にセットされ、ティーチングが完了すると「1」にセットされます。エラーが発生した場合は、「0」のままです。

- ・ Optimizing：最適化中

最適化中の場合、このビットは「1」にセットされます。

最適化が完了すると、このビットは「0」にセットされます。

- ・ Optimization Complete：最適化完了

最適化処理を完了すると、このビットは「1」にセットされます。エラーが発生した場合は、Fault Code エリアがエラーを出力します。

- ・ Autolmage Photometry Enabled：自動測光中

自動測光中の場合、このビットは「1」にセットされます。

自動側光処理が完了すると、このビットは「0」にセットされます。

- ・ Autolmage Photometry Complete：自動測光完了

自動測光処理を完了すると、このビットは「1」にセットされます。エラーが発生した場合は、Fault Code エリアがエラーを出力します。

- ・ Buffer Overflow：バッファオーバーフロー

読み取り文字列長が Decode Data エリアのサイズを超えた場合、このビットが「1」にセットされます。

• Counters：カウンタ

起動後からの読み取り結果の各種カウンタを出力します。

これらのカウンタは、出力アセンブリの Command エリアからリセットすることができます。

カウンタ名	サイズ (バイト)
No Read Read Cycle Counter	4
Mismatch per Read Cycle Counter	4
No Read Counter	4
Trigger Counter	4
Matchcode Counter	4
Mismatch Counter	4

- ・ No Read Read Cycle Counter：読み取りサイクル失敗カウンタ

読み取りサイクル失敗の総数を出力します。

- ・ Mismatch per Read Cycle Counter：読み取りサイクル不一致カウンタ

読み取りサイクル不一致回数の総数を出力します。

- ・ No Read Counter：読み取り失敗カウンタ

読み取り失敗した総数を出力します。

- ・ Trigger Counter：トリガカウンタ

実行したトリガの総数を出力します。

- ・ Matchcode Counter：マッチコードカウンタ

以下のいずれかを出力します。

(1) マスタシンボルに一致した総数（マッチコード：有効である場合）

(2) 読み取り成功した総数（マッチコード：無効である場合）

・ Mismatch Counter：不一致カウンタ

ミスマッチした（マスタシンボルに一致しなかった）総数を出力します。

• **Read Cycle Report：読み取りサイクルレポート**

読み取りサイクルレポート	サイズ（バイト）
Capture Time	2
Decode Time	2
Total Read Cycle Time	2
Reserved	2

・ Capture Time：読み取り時間

デコードに成功した画像の画像撮像に要した時間です。デコードされた画像がない場合、最初の画像撮像に要した時間となります。（ms 単位）

・ Decode Time：デコード総時間

シンボルをデコードするために要した時間です。デコードされた画像がない場合、最初の画像のデコードに要した時間となります。（ms 単位）

・ Total Read Cycle Time：読み取り総時間

シンボルを読み取りするために要した総時間で、画像撮像、デコード、オーバーヘッドの合計時間です。（ms 単位）

• **Decode Cycle Report：デコードサイクルレポート**

シンボル情報を出力します。

シンボル情報	サイズ（バイト）
Decode Location Top	2
Decode Location Left	2
Decode Location Height	2
Decode Location Width	2
Code Type	4
Pixels per Element	4

・ Decode Location Top：コード位置上

シンボル検出領域の左上 Y 座標です。（pixel 単位）

・ Decode Location Left：コード位置左

シンボル検出領域の左上 X 座標です。（pixel 単位）

・ Decode Location Height：コード位置高さ

シンボル検出領域の高さ（Y サイズ）です。（pixel 単位）

・ Decode Location Width：コード位置幅

シンボル検出領域の幅（X サイズ）です。（pixel 単位）

・ Code Type：コードタイプ

復号されたシンボルタイプを示すビットが出力されます。

ビット	状態
0	Aztec Code
1	Micro QR Code
2	Postal Code
3	Code 39
4	Codabar
5	Interleaved 2 of 5
6	UPC/EAN
7	Code 128/EAN 128

ビット	状態
8	Code 93
9	PDF417
10	Pharma Code
11	Data Matrix
12	QR Code
13	BC412
14	GS1 Databar
15	GS1 Databar Limited
16	GS1 Databar Expanded
17	Micro PDF
18	Composite
19	Dot Code
20 - 31	Reserved

- **Pixels Per Element : 1 セル毎のピクセル数**

画像上の 1 セルサイズをピクセル数で表した数が出力されます。

- **Decode Data Length : 読み取り文字列長**

読み取り文字列の文字数が格納されます。

- **Decode Data String : 読み取り文字列**

読み取り文字列が格納されます。印字品質評価指標などの付加情報を設定している場合は、読み取り文字列に続いて格納されます。

メモリ割り当て

	メンバ名	データ型	ビット番号	データ長	バイトオフセット
32 ビット	Reserved	SINT		1 バイト	0
	Reserved	SINT		1 バイト	
	Reserved	SINT		1 バイト	
	Reserved	SINT		1 バイト	
32 ビット	DeviceStatus	DINT		4 バイト	4
	Run Mode		0	1 ビット	
	Trigger Acknowledged		1	1 ビット	
	Exposure Done		2	1 ビット	
	Decoding		3	1 ビット	
	Data is Ready		4	1 ビット	
	Read Cycle Pass		5	1 ビット	
	Read Cycle Fail		6	1 ビット	
	General Fault		7	1 ビット	
	Matchcode Master Label Trained		8	1 ビット	
	Matchcode Enabled		9	1 ビット	
	Image Sensor Calibrating		10	1 ビット	
	Image Sensor Calibration Complete		11	1 ビット	
	Training		12	1 ビット	
	Training Complete		13	1 ビット	
	Optimizing		14	1 ビット	
	Optimizing Complete		15	1 ビット	
	Auto Image Photometry Enabled		16	1 ビット	
	Auto Image Photometry Complete		17	1 ビット	
	Reserved		18	1 ビット	

	メンバ名	データ型	ビット番号	データ長	バイトオフセット
	Reserved		19	1 ビット	
	Reserved		20	1 ビット	
	BufferOverflow		21	1 ビット	
	Reserved		22 - 31	10 ビット	
32 ビット	Reserved	DINT		4 バイト	8
192 ビット	Counters	UDINT[6]		24 バイト	12
	No Read Read Cycle Counter		0 - 31	4 バイト	
	Mismatch per Read Cycle Counter		0 - 31	4 バイト	
	No Read Counter		0 - 31	4 バイト	
	Trigger Counter		0 - 31	4 バイト	
	Matchcode Counter		0 - 31	4 バイト	
	Mismatch Counter		0 - 31	4 バイト	
64 ビット	Read Cycle Report	UINT[4]		8 バイト	36
	Capture Time		0 - 15	2 バイト	
	Decode Time		0 - 15	2 バイト	
	Total Read Cycle Time		0 - 15	2 バイト	
	Reserved		0 - 15	2 バイト	
128 ビット	Decode Cycle Report			16 バイト	44
	Decode Location Top	UINT	0 - 15	2 バイト	
	Decode Location Left	UINT	0 - 15	2 バイト	
	Decode Location Height	UINT	0 - 15	2 バイト	
	Decode Location Width	UINT	0 - 15	2 バイト	
	Code Type	DINT		4 バイト	52
	Aztec Code		0	1 ビット	
	Micro QR Code		1	1 ビット	
	Postal Code		2	1 ビット	
	Code 39		3	1 ビット	
	Codabar		4	1 ビット	
	Interleaved 2 of 5		5	1 ビット	
	UPC EAN		6	1 ビット	
	Code 128 EAN 128		7	1 ビット	
	Code 93		8	1 ビット	
	PDF417		9	1 ビット	
	Pharma Code		10	1 ビット	
	Data Matrix		11	1 ビット	
	QR Code		12	1 ビット	
	BC412		13	1 ビット	
	GS1 Databar		14	1 ビット	
	GS1 Databar Limited		15	1 ビット	
	GS1 Databar Expanded		16	1 ビット	
	Micro PDF		17	1 ビット	
	Composite		18	1 ビット	
	Dot Code		19	1 ビット	
	Reserved		20 - 31	12 ビット	
	Pixels per Element	REAL	0 - 31	4 バイト	56
32 ビット	Decode Data Length	DINT	0 - 31	4 バイト	60
	Decode Data String	SINT[184]	0 - 1471	184 バイト	64

1 デコード入力（インスタンス ID: 103）

1 デコード入力は、436 バイトの読み取り結果文字列を保持するように設計されています。複数シンボル読み取りの場合の読み取り文字列は、区切り文字で区切られて出力されます。

1 デコード入力 メンバ構成

メンバ名	サイズ（バイト）
RESERVED	1
RESERVED	1
RESERVED	1
RESERVED	1
DEVICE STATUS	4
RESERVED	4
COUNTERS	24
READ CYCLE REPORT	8
DECODE CYCLE REPORT	16
DECODE DATA LENGTH	4
DECODE DATA STRING	436

合計サイズ：500 バイト

メンバ詳細

- **Device Status：デバイスステータス**
「MXL/SLC 入力（インスタンス ID: 102）（P.A-7）」と同じ構造です。
- **Counters：カウンタ**
「MXL/SLC 入力（インスタンス ID: 102）（P.A-7）」と同じ構造です。
- **Read Cycle Report：読み取りサイクルレポート**
「MXL/SLC 入力（インスタンス ID: 102）（P.A-7）」と同じ構造です。
- **Decode Cycle Report：デコードサイクルレポート**
「MXL/SLC 入力（インスタンス ID: 102）（P.A-7）」と同じ構造です。
- **Decode Data Length：読み取り文字列長**
読み取り文字列の文字数が格納されます。
- **Decode Data String：読み取り文字列**
読み取り文字列が格納されます。印字品質評価指標などの付加情報を設定している場合は、読み取り文字列に続いて格納されます。

メモリ割り当て

	メンバ名	データ型	ビット番号	データ長	バイトオフセット
32 ビット	Reserved	SINT		1 バイト	0
	Reserved	SINT		1 バイト	
	Reserved	SINT		1 バイト	
	Reserved	SINT		1 バイト	
32 ビット	DeviceStatus	DINT		4 バイト	4
	Run Mode		0	1 ビット	
	Trigger Acknowledged		1	1 ビット	
	Exposure Done		2	1 ビット	
	Decoding		3	1 ビット	
	Data is Ready		4	1 ビット	
	Read Cycle Pass		5	1 ビット	
	Read Cycle Fail		6	1 ビット	
	General Fault		7	1 ビット	
	Matchcode Master Label Trained		8	1 ビット	

	メンバ名	データ型	ビット番号	データ長	バイトオフセット
	Matchcode Enabled		9	1 ビット	
	Image Sensor Calibrating		10	1 ビット	
	Image Sensor Calibration Complete		11	1 ビット	
	Training		12	1 ビット	
	Training Complete		13	1 ビット	
	Optimizing		14	1 ビット	
	Optimizing Complete		15	1 ビット	
	Auto Image Photometry Enabled		16	1 ビット	
	Auto Image Photometry Complete		17	1 ビット	
	Reserved		18	1 ビット	
	Reserved		19	1 ビット	
	Reserved		20	1 ビット	
	BufferOverflow		21	1 ビット	
	Reserved		22 - 31	10 ビット	
32 ビット	Reserved	DINT		4 バイト	8
192 ビット	Counters	UDINT[6]		24 バイト	12
	No Read Read Cycle Counter		0 - 31	4 バイト	
	Mismatch per Read Cycle Counter		0 - 31	4 バイト	
	No Read Counter		0 - 31	4 バイト	
	Trigger Counter		0 - 31	4 バイト	
	Matchcode Counter		0 - 31	4 バイト	
	Mismatch Counter		0 - 31	4 バイト	
64 ビット	Read Cycle Report	UINT[4]		8 バイト	36
	Capture Time		0 - 15	2 バイト	
	Decode Time		0 - 15	2 バイト	
	Total Read Cycle Time		0 - 15	2 バイト	
	Reserved		0 - 15	2 バイト	
128 ビット	Decode Cycle Report			16 バイト	44
	Decode Location Top	UINT	0 - 15	2 バイト	
	Decode Location Left	UINT	0 - 15	2 バイト	
	Decode Location Height	UINT	0 - 15	2 バイト	
	Decode Location Width	UINT	0 - 15	2 バイト	
	Code Type	DINT		4 バイト	52
	Aztec Code		0	1 ビット	
	Micro QR Code		1	1 ビット	
	Postal Code		2	1 ビット	
	Code 39		3	1 ビット	
	Codabar		4	1 ビット	
	Interleaved 2 of 5		5	1 ビット	
	UPC EAN		6	1 ビット	
	Code 128 EAN 128		7	1 ビット	
	Code 93		8	1 ビット	
	PDF417		9	1 ビット	
	Pharma Code		10	1 ビット	
	Data Matrix		11	1 ビット	
	QR Code		12	1 ビット	
	BC412		13	1 ビット	
	GS1 Databar		14	1 ビット	

	メンバ名	データ型	ビット番号	データ長	バイトオフセット
	GS1 Databar Limited		15	1 ビット	
	GS1 Databar Expanded		16	1 ビット	
	Micro PDF		17	1 ビット	
	Composite		18	1 ビット	
	Dot Code		19	1 ビット	
	Reserved		20 - 31	12 ビット	
	Pixels per Element	REAL	0 - 31	4 バイト	56
32 ビット	Decode Data Length	DINT	0 - 31	4 バイト	60
	Decode Data String	SINT[436]	0 - 3487	436 バイト	64

4 デコード入力（インスタンス ID: 104）

4 デコード入力は、4 つのシンボル情報と読み取り結果を保持するように設計されています。最初の読み取り結果は 160 バイトのエリアに格納されます。残りの読み取り結果は 72 バイトのエリアに格納されます。4 つまでの複数シンボル読み取り設定の場合で、シンボル位置座標などのシンボル情報を各シンボルごとに取得したい場合に使用してください。

注意:

リーダで出力フォーマットが有効になっていない場合、**DECODE 'X' DATA**、**DECODE 'X' LENGTH**、および **DECODE 'X' CYCLE REPORT** は、すべてグループ化された各コードのデータを反映します。つまり、同じコードに関連する **DECODE 1 DATA**、**DECODE 1 LENGTH**、および **DECODE 1 CYCLE REPORT** です。

リーダで出力フォーマットが有効になっている場合、**DECODE 'X' DATA** および **DECODE 'X' LENGTH** はユーザが定義したフォーマットを反映します。ただし、**DECODE 'X' CYCLE REPORT** は、ユーザが定義したフォーマットの影響を受けることはなく、リーダで出力フォーマットが有効になっていない場合と同じように情報を保持します。

要約:

- リーダでフォーマット出力が有効になっている場合、**DECODE 'X' CYCLE REPORT** をコードに関連付けることはできません。
- それらが必要な場合は、正しい情報を保証するために、“Output Coordinates” と “Pixel Per Element” を有効にし、“Include Symbology Identifier” オプション (I/O タブ) を **出力フォーマット** に追加し、**DECODE 'X' DATA** および **DECODE 'X' LENGTH** で提示された情報だけを参照してください。**DECODE 'X' CYCLE REPORT** で提示された情報は無視してください。

4 デコード入力 メンバ構成

メンバ名	サイズ (バイト)
RESERVED	1
RESERVED	1
RESERVED	1
RESERVED	1
DEVICE STATUS	4
RESERVED	4
COUNTERS	24
READ CYCLE REPORT	8
DECODE 1 CYCLE REPORT	16
DECODE 1 LENGTH	4
DECODE 1 DATA	160

メンバ名	サイズ (バイト)
DECODE 2 CYCLE REPORT	16
DECODE 2 LENGTH	4
DECODE 2 DATA	72
DECODE 3 CYCLE REPORT	16
DECODE 3 LENGTH	4
DECODE 3 DATA	72
DECODE 4 CYCLE REPORT	16
DECODE 4 LENGTH	4
DECODE 4 DATA	72

合計サイズ：500 バイト

メンバ詳細

- **Device Status：デバイスステータス**
「MXL/SLC 入力（インスタンス ID: 102）（P.A-7）」と同じ構造です。
- **Counters：カウンタ**
「MXL/SLC 入力（インスタンス ID: 102）（P.A-7）」と同じ構造です。
- **Read Cycle Report：読み取りサイクルレポート**
「MXL/SLC 入力（インスタンス ID: 102）（P.A-7）」と同じ構造です。
- **Decode 1 Cycle Report：デコードサイクルレポート 1**
1 番目のシンボル情報です。
「MXL/SLC 入力（インスタンス ID: 102）（P.A-7）」の Decode Cycle Report と同じ構造です。
- **Decode 1 Data Length：読み取り文字列長 1**
1 番目のシンボルの文字数が格納されます。
- **Decode 1 Data String：読み取り文字列 1**
1 番目のシンボルの読み取り文字列が格納されます。印字品質評価指標などの付加情報を設定している場合は読み取り文字列に続いて格納されます。
- **Decode 2 Cycle Report：デコードサイクルレポート 2**
2 番目のシンボル情報です。
「MXL/SLC 入力（インスタンス ID: 102）（P.A-7）」の Decode Cycle Report と同じ構造です。
- **Decode 2 Data Length：読み取り文字列長 2**
2 番目のシンボルの文字数が格納されます。
- **Decode 2 Data String：読み取り文字列 2**
2 番目のシンボルの読み取り文字列が格納されます。印字品質評価指標などの付加情報を設定している場合は読み取り文字列に続いて格納されます。
- **Decode 3 Cycle Report：デコードサイクルレポート 3**
3 番目のシンボル情報です。
「MXL/SLC 入力（インスタンス ID: 102）（P.A-7）」の Decode Cycle Report と同じ構造です。
- **Decode 3 Data Length：読み取り文字列長 3**
3 番目のシンボルの文字数が格納されます。
- **Decode 3 Data String：読み取り文字列 3**
3 番目のシンボルの読み取り文字列が格納されます。印字品質評価指標などの付加情報を設定している場合は読み取り文字列に続いて格納されます。
- **Decode 4 Cycle Report：デコードサイクルレポート 4**
4 番目のシンボル情報です。
「MXL/SLC 入力（インスタンス ID: 102）（P.A-7）」の Decode Cycle Report と同じ構造です。
- **Decode 4 Data Length：読み取り文字列長 4**
4 番目のシンボルの文字数が格納されます。
- **Decode 4 Data String：読み取り文字列 4**
4 番目のシンボルの読み取り文字列が格納されます。印字品質評価指標などの付加情報を設定している場合は読み取り文字列に続いて格納されます。

メモリ割り当て

	メンバ名	データ型	ビット番号	データ長	バイトオフセット
32 ビット	Reserved	SINT		1 バイト	0
	Reserved	SINT		1 バイト	
	Reserved	SINT		1 バイト	
	Reserved	SINT		1 バイト	
32 ビット	DeviceStatus	DINT		4 バイト	4
	Run Mode		0	1 ビット	
	Trigger Acknowledged		1	1 ビット	
	Exposure Done		2	1 ビット	
	Decoding		3	1 ビット	
	Data is Ready		4	1 ビット	
	Read Cycle Pass		5	1 ビット	
	Read Cycle Fail		6	1 ビット	
	General Fault		7	1 ビット	
	Matchcode Master Label Trained		8	1 ビット	
	Matchcode Enabled		9	1 ビット	
	Image Sensor Calibrating		10	1 ビット	
	Image Sensor Calibration Complete		11	1 ビット	
	Training		12	1 ビット	
	Training Complete		13	1 ビット	
	Optimizing		14	1 ビット	
	Optimizing Complete		15	1 ビット	
	Auto Image Photometry Enabled		16	1 ビット	
	Auto Image Photometry Complete		17	1 ビット	
	Reserved		18	1 ビット	
	Reserved		19	1 ビット	
	Reserved		20	1 ビット	
	BufferOverflow		21	1 ビット	
	Reserved		22 - 31	10 ビット	
32 ビット	Reserved	DINT		4 バイト	8
192 ビット	Counters	UDINT[6]		24 バイト	12
	No Read Read Cycle Counter		0 - 31	4 バイト	
	Mismatch per Read Cycle Counter		0 - 31	4 バイト	
	No Read Counter		0 - 31	4 バイト	
	Trigger Counter		0 - 31	4 バイト	
	Matchcode Counter		0 - 31	4 バイト	
	Mismatch Counter		0 - 31	4 バイト	
64 ビット	Read Cycle Report	UINT[4]		8 バイト	36
	Capture Time		0 - 15	2 バイト	
	Decode Time		0 - 15	2 バイト	
	Total Read Cycle Time		0 - 15	2 バイト	
	Reserved		0 - 15	2 バイト	
128 ビット	Decode 1 Cycle Report			16 バイト	44
	Decode Location Top	UINT	0 - 15	2 バイト	
	Decode Location Left	UINT	0 - 15	2 バイト	
	Decode Location Height	UINT	0 - 15	2 バイト	
	Decode Location Width	UINT	0 - 15	2 バイト	
	Code Type	DINT		4 バイト	52

	メンバ名	データ型	ビット番号	データ長	バイトオフセット
	Aztec Code		0	1 ビット	
	Micro QR Code		1	1 ビット	
	Postal Code		2	1 ビット	
	Code 39		3	1 ビット	
	Codabar		4	1 ビット	
	Interleaved 2 of 5		5	1 ビット	
	UPC EAN		6	1 ビット	
	Code 128 EAN 128		7	1 ビット	
	Code 93		8	1 ビット	
	PDF417		9	1 ビット	
	Pharma Code		10	1 ビット	
	Data Matrix		11	1 ビット	
	QR Code		12	1 ビット	
	BC412		13	1 ビット	
	GS1 Databar		14	1 ビット	
	GS1 Databar Limited		15	1 ビット	
	GS1 Databar Expanded		16	1 ビット	
	Micro PDF		17	1 ビット	
	Composite		18	1 ビット	
	Dot Code		19	1 ビット	
	Reserved		20 - 31	12 ビット	
	Pixels per Element	REAL	0 - 31	4 バイト	56
32 ビット	Decode 1 Data Length	DINT	0 - 31	4 バイト	60
	Decode 1 Data String	SINT[160]	0 - 1279	160 バイト	64
128 ビット	Decode 2 Cycle Report			16 バイト	224
	Decode Location Top	UINT	0 - 15	2 バイト	
	Decode Location Left	UINT	0 - 15	2 バイト	
	Decode Location Height	UINT	0 - 15	2 バイト	
	Decode Location Width	UINT	0 - 15	2 バイト	
	Code Type	DINT		4 バイト	232
	Aztec Code		0	1 ビット	
	Micro QR Code		1	1 ビット	
	Postal Code		2	1 ビット	
	Code 39		3	1 ビット	
	Codabar		4	1 ビット	
	Interleaved 2 of 5		5	1 ビット	
	UPC EAN		6	1 ビット	
	Code 128 EAN 128		7	1 ビット	
	Code 93		8	1 ビット	
	PDF417		9	1 ビット	
	Pharma Code		10	1 ビット	
	Data Matrix		11	1 ビット	
	QR Code		12	1 ビット	
	BC412		13	1 ビット	
	GS1 Databar		14	1 ビット	
	GS1 Databar Limited		15	1 ビット	

	メンバ名	データ型	ビット番号	データ長	バイトオフセット
	GS1 Databar Expanded		16	1 ビット	236
	Micro PDF		17	1 ビット	
	Composite		18	1 ビット	
	Dot Code		19	1 ビット	
	Reserved		20 - 31	12 ビット	
	Pixels per Element	REAL	0 - 31	4 バイト	
32 ビット	Decode 2 Data Length	DINT	0 - 31	4 バイト	240
	Decode 2 Data String	SINT[72]	0 - 575	72 バイト	244
128 ビット	Decode 3 Cycle Report			16 バイト	316
	Decode Location Top	UINT	0 - 15	2 バイト	324
	Decode Location Left	UINT	0 - 15	2 バイト	
	Decode Location Height	UINT	0 - 15	2 バイト	
	Decode Location Width	UINT	0 - 15	2 バイト	
	Code Type	DINT		4 バイト	
	Aztec Code		0	1 ビット	
	Micro QR Code		1	1 ビット	
	Postal Code		2	1 ビット	
	Code 39		3	1 ビット	
	Codabar		4	1 ビット	
	Interleaved 2 of 5		5	1 ビット	
	UPC EAN		6	1 ビット	
	Code 128 EAN 128		7	1 ビット	
	Code 93		8	1 ビット	
	PDF417		9	1 ビット	
	Pharma Code		10	1 ビット	
	Data Matrix		11	1 ビット	
	QR Code		12	1 ビット	
	BC412		13	1 ビット	
	GS1 Databar		14	1 ビット	
	GS1 Databar Limited		15	1 ビット	
	GS1 Databar Expanded		16	1 ビット	
	Micro PDF		17	1 ビット	
	Composite		18	1 ビット	
	Dot Code		19	1 ビット	
	Reserved		20 - 31	12 ビット	
	Pixels per Element	REAL	0 - 31	4 バイト	
32 ビット	Decode 3 Data Length	DINT	0 - 31	4 バイト	332
	Decode 3 Data String	SINT[72]	0 - 575	72 バイト	336
128 ビット	Decode 4 Cycle Report			16 バイト	408
	Decode Location Top	UINT	0 - 15	2 バイト	416
	Decode Location Left	UINT	0 - 15	2 バイト	
	Decode Location Height	UINT	0 - 15	2 バイト	
	Decode Location Width	UINT	0 - 15	2 バイト	
	Code Type	DINT		4 バイト	
	Aztec Code		0	1 ビット	
	Micro QR Code		1	1 ビット	

	メンバ名	データ型	ビット番号	データ長	バイトオフセット
	Postal Code		2	1 ビット	
	Code 39		3	1 ビット	
	Codabar		4	1 ビット	
	Interleaved 2 of 5		5	1 ビット	
	UPC EAN		6	1 ビット	
	Code 128 EAN 128		7	1 ビット	
	Code 93		8	1 ビット	
	PDF417		9	1 ビット	
	Pharma Code		10	1 ビット	
	Data Matrix		11	1 ビット	
	QR Code		12	1 ビット	
	BC412		13	1 ビット	
	GS1 Databar		14	1 ビット	
	GS1 Databar Limited		15	1 ビット	
	GS1 Databar Expanded		16	1 ビット	
	Micro PDF		17	1 ビット	
	Composite		18	1 ビット	
	Dot Code		19	1 ビット	
	Reserved		20 - 31	12 ビット	
	Pixels per Element	REAL	0 - 31	4 バイト	420
32 ビット	Decode 4 Data Length	DINT	0 - 31	4 バイト	424
	Decode 4 Data String	SINT[72]	0 - 575	72 バイト	428

N デコード入力（インスタンス ID: 105）

N デコード入力は、任意の数の複数シンボル読み取りに対応しています。任意の数の複数シンボル読み取りを実行し、シンボル位置座標などのシンボル情報を各シンボルごとに取得したい場合に使用してください。読み取り結果のデータ構造は最大 456 バイトの可変長になっていますので、ユーザはデータオフセット値を参照して読み取り文字列などのデータにアクセスする必要があります。

注意:

リーダで出力フォーマットが有効になっていない場合、**DECODE 'X' DATA**、**DECODE 'X' LENGTH**、および **DECODE 'X' CYCLE REPORT**（RAW INPUT DATA の“読み取りデータ構造”。次の表を参照）は、すべてグループ化された各コードのデータを反映します。つまり、同じコードに関連する **DECODE 1 DATA**、**DECODE 1 LENGTH**、および **DECODE 1 CYCLE REPORT** です。

リーダで出力フォーマットが有効になっている場合、**DECODE 'X' DATA** および **DECODE 'X' LENGTH** はユーザが定義したフォーマットを反映します。ただし、**DECODE 'X' CYCLE REPORT** は、ユーザが定義したフォーマットの影響を受けることなく、リーダで出力フォーマットが有効になっていない場合と同じように情報を保持します。

要約:

- リーダでフォーマット出力が有効になっている場合、**DECODE 'X' CYCLE REPORT** をコードに関連付けることはできません。
- それらが必要な場合は、正しい情報を保証するために、“Output Coordinates” と “Pixel Per Element” を有効にし、“Include Symbology Identifier” オプション（I/O タブ）を出力フォーマットに追加し、**DECODE 'X' DATA** および **DECODE 'X' LENGTH** で提示された情報だけを参照してください。**DECODE 'X' CYCLE REPORT** で提示された情報は無視してください。

N デコード入力 メンバ構成

メンバ名	サイズ (バイト)
RESERVED	1
RESERVED	1
RESERVED	1
RESERVED	1
DEVICE STATUS	4
RESERVED	4
COUNTERS	24
READ CYCLE REPORT STATIC MEMBERS	8
RAW INPUT DATA	456

合計サイズ：500 バイト

メンバ詳細

- **Device Status：デバイスステータス**
「MXL/SLC 入力（インスタンス ID: 102）（P.A-7）」と同じ構造です。
- **Counters：カウンタ**
「MXL/SLC 入力（インスタンス ID: 102）（P.A-7）」と同じ構造です。
- **Read Cycle Report：読み取りサイクルレポート**

読み取りサイクルレポート	サイズ (バイト)
Capture Time	2
Decode Time	2
Total Read Cycle Time	2
Number of Decodes in Read Cycle	1
Number of Decode Reports	1

- ・ Capture Time：読み取り時間
デコードに成功した画像の画像撮像に要した時間です。デコードされた画像がない場合、最初の画像撮像に要した時間となります。(ms 単位)
- ・ Decode Time：デコード総時間
シンボルをデコードするために要した時間です。デコードされた画像がない場合、最初の画像のデコードに要した時間となります。(ms 単位)
- ・ Total Read Cycle Time：読み取り総時間
シンボルを読み取りするために要した総時間で、画像撮像、デコード、オーバーヘッドの合計時間です。(ms 単位)
- ・ Number of Decodes in Read Cycle：読み取りサイクル中のデコード数
読み取りサイクル中に検出されたシンボルの総数です。
- ・ Number of Decode Reports：デコードレポート数
検出したシンボルに関連したデコード情報の総数です。
読み取りサイクル中に検出されたシンボルの総数に等しいです。

- **Raw Input Data：読み取りデータ配列**
読み取りデータが可変長で格納されます。

読み取りデータ構造	サイズ (バイト)	オフセット
Offset of Report 1	4	オフセット 1
Offset of Report 2	4	
...	...	
Offset of Report N	4	
Decode Cycle Report 1	16	
Decode Length 1	4	
Decode Data 1	可変長	

読み取りデータ構造	サイズ (バイト)	オフセット
Decode Cycle Report 2	16	オフセット 2
Decode Length 2	4	
Decode Data 2	可変長	
...	...	
Decode Cycle Report N*1	16	オフセット N
Decode Data Length N*1	4	
Decode Data String N*1	可変長	

*1. N は、Number of Decodes in Read Cycle に出力される値です。

・ Offset of Report (n) : レポート n のオフセット

Raw Input Data 先頭アドレスから n 番目の読み取り結果が格納されているアドレスまでのオフセット値です。

・ Decode Cycle Report (n) : デコードサイクルレポート n

n 番目のシンボル情報です。

「MXL/SLC 入力 (インスタンス ID: 102) (P.A-7)」の Decode Cycle Report と同じ構造です。

・ Decode Data Length (n) : 読み取り文字列長 n

n 番目のシンボルの文字数が格納されます。

・ Decode Data String (n) : 読み取り文字列 n

n 番目のシンボルの読み取り文字列が格納されます。

メモリ割り当て

	メンバ名	データ型	ビット番号	データ長	バイトオフセット
32 ビット	Reserved	SINT		1 バイト	0
	Reserved	SINT		1 バイト	
	Reserved	SINT		1 バイト	
	Reserved	SINT		1 バイト	
32 ビット	DeviceStatus	DINT		4 バイト	4
	Run Mode		0	1 ビット	
	Trigger Acknowledged		1	1 ビット	
	Exposure Done		2	1 ビット	
	Decoding		3	1 ビット	
	Data is Ready		4	1 ビット	
	Read Cycle Pass		5	1 ビット	
	Read Cycle Fail		6	1 ビット	
	General Fault		7	1 ビット	
	Matchcode Master Label Trained		8	1 ビット	
	Matchcode Enabled		9	1 ビット	
	Image Sensor Calibrating		10	1 ビット	
	Image Sensor Calibration Complete		11	1 ビット	
	Training		12	1 ビット	
	Training Complete		13	1 ビット	
	Optimizing		14	1 ビット	
	Optimizing Complete		15	1 ビット	
	Auto Image Photometry Enabled		16	1 ビット	
	Auto Image Photometry Complete		17	1 ビット	
	Reserved		18	1 ビット	
	Reserved		19	1 ビット	
	Reserved		20	1 ビット	
	BufferOverflow		21	1 ビット	
	Reserved		22 - 31	10 ビット	

	メンバ名	データ型	ビット番号	データ長	バイトオフセット
32 ビット	Reserved	DINT		4 バイト	8
192 ビット	Counters	UDINT[6]		24 バイト	12
	No Read Read Cycle Counter		0 - 31	4 バイト	
	Mismatch per Read Cycle Counter		0 - 31	4 バイト	
	No Read Counter		0 - 31	4 バイト	
	Trigger Counter		0 - 31	4 バイト	
	Matchcode Counter		0 - 31	4 バイト	
	Mismatch Counter		0 - 31	4 バイト	
64 ビット	Read Cycle Report	UINT[4]		8 バイト	36
	Capture Time		0 - 15	2 バイト	
	Decode Time		0 - 15	2 バイト	
	Total Read Cycle Time		0 - 15	2 バイト	
	Number of Decodes in Read Cycle		0 - 7	1 バイト	
	Number of Decode Reports		0 - 7	1 バイト	
	RAW Input Data			456 byte	44 バイト

出力（インスタンス ID: 197）

出力は、コードリーダに対し、いくつかのコマンドを送信することが可能です。
このアセンブリは、MXL/SLC 入力（インスタンス ID: 102）、1 デコード入力（インスタンス ID: 103）、4 デコード入力（インスタンス ID: 104）、N デコード入力（インスタンス ID: 105）で使用されます。

出力 メンバ構成

メンバ名	サイズ（バイト）
COMMANDS	4

合計サイズ：4 バイト

メンバ詳細

・ Commands：コマンド

コードリーダに送信できるコマンドを説明します。

ビット	コマンド
0	Run Mode
1	Trigger
2	Enable Matchcode
3	Reset General Fault
4	Clear No Read Read Cycle Count
5	Clear Mismatch Read Cycle Count
6	Clear No Read Count
7	Clear Trigger Count
8	Clear Matchcode Count
9	Clear Mismatch Count
10 - 31	Reserved

・ Run Mode：ランモード

読み取りサイクルの有効、無効を切り替えます。コードリーダ起動直後は、シリアルコマンドまたは WebLink_{HH} を介して、このコマンド状態に関係なく読み取りサイクル有効状態になります。

0 = 読み取りサイクル無効状態にします。トリガを受付ません。ただし、その他のコマンドを実行することは可能です。

1 = 読み取りサイクル有効状態にします。

・ Trigger：トリガ

読み取りを実行します。コードリーダはこのビットの 0→1 をトリガの立ち上がりエッジ、1→0 をトリガの立ち下がりエッジと認識します。

・ Enabled Matchcode：マッチコード有効

マッチコード機能の有効、無効を切り替えます。コードリーダ起動直後は、このコマンド状態に関係なく前回設定保存時の状態になります。

0 = マッチコード機能を無効にします。

1 = マッチコード機能を有効にします。

・ Reset General Fault：エラーリセット

コードリーダにエラーが発生した場合、問題を解決した後に、このビットを使用して入力アセンブリの Fault Code エリアをリセットできます。

・ Clear No Read Read Cycle Count：読み取りサイクル失敗カウンタクリア

読み取りサイクル失敗カウンタを 0 にリセットします。

・ Clear Mismatch Read Cycle Count：読み取りサイクル不一致カウンタクリア

読み取りサイクル不一致カウンタを 0 にリセットします。

・ Clear No Read Count：読み取り失敗カウンタクリア

読み取り失敗カウンタを 0 にリセットします。

・ Clear Trigger Count：トリガカウンタクリア

トリガカウンタを 0 にリセットします。

・ Clear Matchcode Count：マッチコードカウンタクリア

マッチコードカウンタを 0 にリセットします。

・ Clear Mismatch Count：不一致カウンタクリア

不一致カウンタを 0 にリセットします。

メモリ割り当て

	メンバ名	データ型	ビット番号	データ長	バイトオフセット
32 ビット	Commands	DINT		4 バイト	0
	Run Mode		0	1 ビット	
	Trigger		1	1 ビット	
	Enable Matchcode		2	1 ビット	
	Reset General Fault		3	1 ビット	
	Clear No Read Read Cycle Count		4	1 ビット	
	Clear Mismatch Read Cycle Count		5	1 ビット	
	Clear No Read Count		6	1 ビット	
	Clear Trigger Count		7	1 ビット	
	Clear Matchcode Count		8	1 ビット	
	Clear Mismatch Count		9	1 ビット	
	Reserved		10 - 31	22 ビット	

出力（レガシー）（インスタンス ID: 198）

出力（レガシー）は、コードリーダに対し、固定データのエコーバックやいくつかのコマンドを送信することが可能です。

このアセンブリは、小型入力（インスタンス ID: 100）、大型入力（インスタンス ID: 101）で使用されます。

出力（レガシー）メンバ構成

メンバ名	サイズ（バイト）
USER-DEFINED TAGS	4
COMMANDS	4
RESERVED	4

合計サイズ：12 バイト

メンバ詳細

・ User-Defined Tags：ユーザ定義タグ

このメンバに設定したデータは、小型入力、または大型入力の USER-DEFINED TAG ECHO エリアにエコーバックされます。複数のコードリーダを一意に識別したい場合などに使用します。

・ Commands：コマンド

コードリーダに送信できるコマンドを説明します。

ビット	コマンド
0	Trigger
1	New Master
2 - 7	Reserved
8	Disable Scanning
9 - 15	Reserved
16	Clear Read Cycle Report and Counters
17 - 31	Reserved

・ Trigger：トリガ

読み取りを実行します。コードリーダはこのビットの 0→1 をトリガの立ち上がりエッジ、1→0 をトリガの立ち下がりエッジと認識します。

・ New Master：新規マスタ登録

このビットを ON すると、次の読み取り結果をマスタシンボルとして登録します。

・ Disable Scanning：読み取りサイクル無効

読み取りサイクルの有効、無効を切り替えます。

0 = 読み取りサイクル有効状態にします。

1 = 読み取りサイクル無効状態にします。ただし、その他のコマンドを実行することは可能です。

・ Clear Read Cycle Report and Counters：読み取りサイクルレポートとカウンタクリア

小型入力、または大型入力のカウンタエリアおよび読み取り文字列エリアをリセットします。

Decode Data Length（読み取り文字列長）は 0 に設定され、Decode Data String（読み取り文字列）エリアの最初のバイトは、標準的な終端文字である“\0”（ヌル）に設定されます。

メモリ割り当て

	メンバ名	データ型	ビット番号	データ長	バイトオフセット
32 ビット	User Defined Tag	DINT		4 バイト	0
	UserTag_1		0	1 ビット	
	UserTag_2		1	1 ビット	
	UserTag_3		2	1 ビット	
	UserTag_4		3	1 ビット	
	UserTag_5		4	1 ビット	
	UserTag_6		5	1 ビット	
	UserTag_7		6	1 ビット	
	UserTag_8		7	1 ビット	
	UserTag_9		8	1 ビット	
	UserTag_10		9	1 ビット	
	UserTag_11		10	1 ビット	

	メンバ名	データ型	ビット番号	データ長	バイトオフセット
	UserTag_12		11	1 ビット	
	UserTag_13		12	1 ビット	
	UserTag_14		13	1 ビット	
	UserTag_15		14	1 ビット	
	UserTag_16		15	1 ビット	
	UserTag_17		16	1 ビット	
	UserTag_18		17	1 ビット	
	UserTag_19		18	1 ビット	
	UserTag_20		19	1 ビット	
	UserTag_21		20	1 ビット	
	UserTag_22		21	1 ビット	
	UserTag_23		22	1 ビット	
	UserTag_24		23	1 ビット	
	UserTag_25		24	1 ビット	
	UserTag_26		25	1 ビット	
	UserTag_27		26	1 ビット	
	UserTag_28		27	1 ビット	
	UserTag_29		28	1 ビット	
	UserTag_30		29	1 ビット	
	UserTag_31		30	1 ビット	
	UserTag_32		31	1 ビット	
32 ビット	Commands	DINT		4 バイト	4
	Trigger		0	1 ビット	
	New Master		1	1 ビット	
	Reserved		2 - 7	6 ビット	
	Disable Scanning		8	1 ビット	
	Reserved		9 - 15	7 ビット	
	Clear Read Cycle Report and Counters		16	1 ビット	
	Reserved		17 - 31	15 ビット	
32 ビット	Reserved	DINT		4 バイト	8

A-3 PROFINET - V460-H 入出力モジュール

ここでは、V460-H と PROFINET 産業用プロトコルで使用可能なコマンドを示します。

A-3-1 モジュール種別

入力モジュールが 7 個、出力モジュールが 2 個あります。各モジュールの割り当てとデータ定義を、この付録で説明しています。

モデル項目 ID	名称	サイズ (バイト)	PROFINET 許可スロット	PNT21 対応
100	小型入力モジュール	84	1	○
101	大型入力モジュール	176	1	○
102	MXL 入力モジュール	248	1	○
103	1 デコード入力モジュール	500	1	-
104	4 デコード入力モジュール	500	1	-
105	N デコード入力モジュール	500	1	-
106	Omron デコード入力モジュール	442	1	○
197	出力モジュール (プレミア)	4	2	○
198	出力モジュール (レガシー)	12	2	○



参考

PNT21 用の入力 CR の最大サイズは、450 バイトです。

入力モジュール／出力モジュール

以下の Omron デコード入力モジュールを除く入力モジュールと出力モジュールの詳細については、同じ ID の「A-2 EtherNet/IP 通信仕様詳細 (P.A-3)」と同様です。

● Omron デコード入力モジュール (インスタンス ID: 106)

この入力は、Decode Data String のサイズが 378 バイトであることを除き、1 デコード入力 (インスタンス ID: 103) と同じです。

名称	サイズ (バイト)
MODULE HEADER	
RESERVED	1
RESERVED	1
RESERVED	1
RESERVED	1
DEVICE STATUS	4
RESERVED	4
COUNTERS	24
READ CYCLE REPORT	8
DECODE CYCLE REPORT	
DECODE CYCLE REPORT TABLE	16
DECODE DATA LENGTH	4
DECODE DATA STRING	378

合計サイズ：442 バイト

A-3-2 データ型

入力モジュール／出力モジュール用ユーザデータ型

モジュール名称	ユーザデータ型名称
小型入力モジュール	- Input_Legacy_Small - Legacy_UserTag_Echo - Legacy_Command_Echo
大型入力モジュール	- Input_Legacy_Big - Legacy_UserTag_Echo - Legacy_Command_Echo - Legacy_Device_Status
MXL 入力モジュール	- Input_MXL_Decode - Input_Header - ReadCycle_Report - Input_MXL_Decode_Report
1 デコード入力モジュール	- Input_1_Decode - Input_Header - ReadCycle_Report - Decode_Report_436Bytes
4 デコード入力モジュール	- Input_4_Decode - Input_Header - ReadCycle_Report - Decode_Report_160Bytes - Decode_Report_72Bytes
N デコード入力モジュール	- Input_N_Decode - Input_N_Header - Input_N_ReadCycle_Report - Decode_Report_436Bytes
出力モジュール（レガシー）	- Output_Legacy - Legacy_User_Defined_Tags - Legacy_Cmds
出力モジュール（プレミア）	Premier_Cmds

A-3-3 PROFINET 基本情報

デバイス ID

PROFINET デバイス ID 情報は、以下のとおりです。

- **ベンダ ID**
ベンダ ID は、0x0257 です。
- **デバイス ID**
デバイス ID は、以下の「ファームウェアバージョンごとの PROFINET ファイル (P.A-30)」を参照してください。
- **ベンダ名称**
ベンダ名称は、OMRON MICROSCAN SYSTEMS, INC です。
- **デバイス機能**
デバイス機能は、以下のとおりです。
 - MainFamily = Ident Systems
 - ProductFamily = V460-H

GSDML ファイル

GSDML ファイルは、以下の「ファームウェアバージョンごとの PROFINET ファイル (P.A-30)」を参照してください。

ファームウェアバージョンごとの PROFINET ファイル

シリーズ	ファームウェアバージョン	GSDML ファイル	バージョン	デバイス ID
V460-H	1.0.0.xxxx	GSDML-V2.35-OmronMicroscanSystemsInc-V460-H-20210827.xml	V2.35	0x3414

接続プロパティ: RT Cyclic Messaging

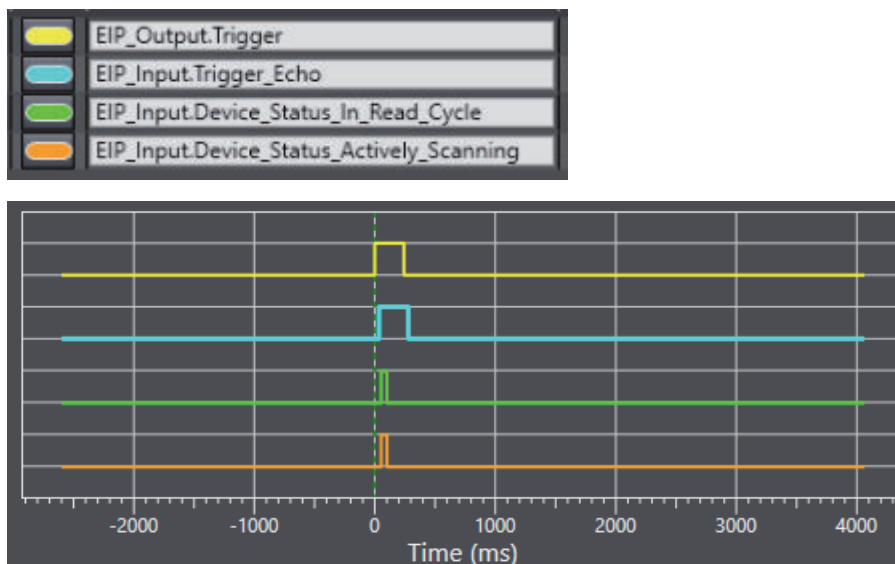
サイクルタイム: 8ms

定義: GSD ファイルの MinDeviceInterval 要素の値の 256 と $31.25 \mu s$ の積がサイクルタイムとなります。詳細は、PROFINET GSDML 仕様を参照してください。

A-3-4 タイミングチャート

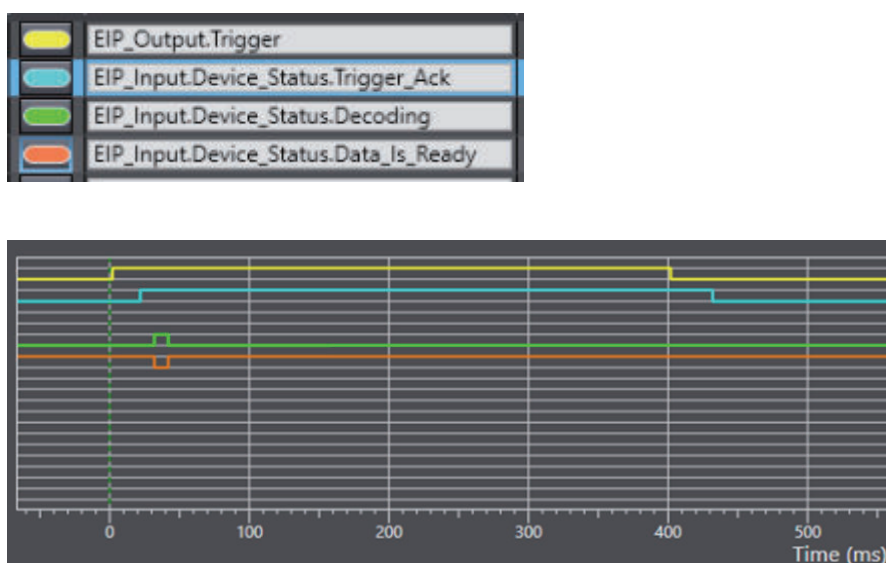
大型入力モジュールタイミング

大型入力モジュールは、ユーザへのフィードバックがほとんどないためタイミングが限られています。



Omron デコード入力モジュールタイミング

- この例では、Trigger は 400ms 間 High で設定されています。
- Trigger が High に設定された後、20ms 後に認識され、TriggerAcknowledgeHigh が High になり、Trigger が Low になった後、30ms 間 TriggerAcknowledgeHigh は High を保持しています。
- 読み取りが完了し、データは Trigger 開始後 39ms で準備が完了しています。



オムロン株式会社 インダストリアルオートメーションビジネスカンパニー

製品に関するお問い合わせ先

お客様
相談室

フリー
通話

0120-919-066

携帯電話・IP電話などではご利用いただけ
ませんので、右記の電話番号へおかけください。

055-982-5015
(通話料がかかります)

受付時間：9:00～19:00 (12/31～1/3を除く)

 オムロンFAクイックチャット
www.fa.omron.co.jp/contact/tech/chat/



技術相談員にチャットでお問い合わせいただけます。(I-Webメンバーズ限定)

受付時間：平日9:00～12:00 / 13:00～17:00 (土日祝日・年末年始・当社休業日を除く)

※受付時間、営業日は変更の可能性があります。最新情報はリンク先をご確認ください。

その他のお問い合わせ：納期・価格・サンプル・仕様書は貴社のお取引先、または貴社担当オムロン販売員にご相談ください。オムロン制御機器販売店やオムロン販売拠点は、Webページでご案内しています。



オムロン制御機器の最新情報をご覧ください。緊急時のご購入にもご利用ください。

www.fa.omron.co.jp

- 本誌に記載のない条件や環境での使用、および原子力制御・鉄道・航空・車両・燃焼装置・医療機器・娯楽機械・安全機器、その他人命や財産に大きな影響が予測されるなど、特に安全性が要求される用途に使用される際には、当社の意図した特別な商品用途の場合や特別の合意がある場合を除き、当社は当社商品に対して一切保証をいたしません。
- 本製品の内、外国為替及び外国貿易法に定める輸出許可、承認対象貨物（又は技術）に該当するものを輸出（又は非居住者に提供）する場合は同法に基づく輸出許可、承認（又は役務取引許可）が必要です。

オムロン商品のご用命は