

CUnet Family

CU(B)-IOLGW8

DATA SHEET

CUnet

CUnet 対応 IO-Link Gateway

IOLGW CUnet リファレンスマニュアル

目 次

1. はじめに	1-1
2. グローバルメモリー領域(GM)	2-1
2.1 占有データブロック.....	2-1
2.1.1 IOLGW の ProcessInData.....	2-2
2.1.2 IO-Link デバイスの ProcessInData.....	2-3
2.2 データ出力メモリーブロック	2-4
2.2.1 IOLGW の ProcessOutData.....	2-4
2.2.2 IO-Link デバイスの ProcessOutData.....	2-5
3. メール通信機能	3-1
3.1 PortConfig(0x20).....	3-3
3.1.1 OpeMode[0](ItemNo=0)	3-5
3.1.2 OpeMode[1](ItemNo=1)	3-7
3.1.3 OpeMode[2](ItemNo=2)	3-9
3.1.4 CycleTime(ItemNo=3)	3-11
3.2 ProcessDataSize(0x21)	3-13
3.2.1 UProcInByte(ItemNo=0)	3-13
3.2.2 UProcOutByte(ItemNo=1).....	3-15
3.2.3 AutoAlloc(ItemNo=2).....	3-17
3.3 VerifyConfig(0x22)	3-18
3.3.1 VenderID(ItemNo=0)	3-18
3.3.2 DeviceID(ItemNo=1)	3-20
3.3.3 SerialNo(ItemNo=2)	3-22
3.3.4 AutoCfg(ItemNo=3)	3-24
3.4 NoDataOut(0x23).....	3-25
3.5 DataStorage(0x30)	3-27
3.5.1 StorageInformation(ItemNo=0)	3-28
3.5.2 StorageClear(ItemNo=1)	3-29
3.5.3 ManualBackup(ItemNo=2).....	3-30
3.5.4 ManualRestore(ItemNo=3)	3-31
3.6 Diagnosis(0x40)	3-32
3.6.1 UnitStatus(ItemNo=0).....	3-35
3.6.2 UnitDiag(ItemNo=1)	3-36
3.6.3 DeviceDiag(ItemNo=2).....	3-37
3.6.4 DiagPort(ItemNo=3)	3-38
3.6.5 COMStatus(ItemNo=4).....	3-39
3.6.6 COMError(ItemNo=5).....	3-40
3.6.7 FPGATemp(ItemNo=6)	3-41

3.7 Information(0x41)	3-42
3.7.1 VenderName(ItemNo=0)	3-43
3.7.2 ProductName(ItemNo=1)	3-44
3.7.3 ProductType(ItemNo=2)	3-45
3.7.4 SerialNo(ItemNo=3)	3-46
3.7.5 HardRev(ItemNo=4)	3-47
3.7.6 FirmRev(ItemNo=5)	3-48
3.8 NetworkStatus(0x42)	3-49
3.9 CUnet(0x43)	3-50
3.10 DeviceInformation(0x44)	3-51
3.10.1 DProcInByte(ItemNo=0)	3-52
3.10.2 DProcOutByte(ItemNo=1)	3-53
3.10.3 VenderID(ItemNo=2)	3-54
3.10.4 DeviceID(ItemNo=3)	3-55
3.10.5 MinCycle(ItemNo=4)	3-56
3.10.6 SeqType(ItemNo=5)	3-57
3.10.7 SerialNo(ItemNo=6)	3-58
3.11 ISDU 通信(0x5n)	3-59
3.12 ProcessData(0x60)	3-61
3.12.1 ProcInCh(ItemNo=0)	3-61
3.12.2 ProcInALL(ItemNo=1)	3-62
3.13 ConfigReset(0x70)	3-63
3.13.1 UnitReset(ItemNo=0)	3-63
3.13.2 ChReset(ItemNo=1)	3-64
3.13.3 CUReset(ItemNo=2)	3-65
4. 付録	4-1
4.1 メール通信コマンド一覧	4-1
4.2 エラーコード	4-2
4.2.1 メール通信ステータスエラーコード	4-2
4.2.2 IO-Link マスターエラーコード	4-2

1. はじめに

CUnet IO-Link Gateway(以降“IOLGW”と省略)は、IO-Link デバイスを CUnet ネットワーク下においてコントロールできるようにした Gateway ユニットです。

本資料では、この IOLGW を制御する CUnet 通信について記述しています。

この IOLGW の機能及び仕様については別紙のデータシートを参照してください。

本資料は、「CUnet 導入ガイド」「CUnet テクニカルガイド」「MKY43 ユーザーズマニュアル」を既にお読みになっていることを前提に記述させていただいております。まずはこれらの資料を熟読いただきますようお願いいたします。

2. グローバルメモリー領域(GM)

ここでは CUnet のグローバルメモリー領域の使い方について説明します。

IOLGW は、IO-Link デバイスの ProcessInData を格納する占有メモリーブロックと、ProcessOutData として参照するデータ出力メモリーブロックを必要としています。

前者は SW102 で指定された SA (Station Address) を先頭として使用します。後者は SW101 で指定された DOSA (Data Out Station Address) を先頭として使用します。この時使われるメモリーブロックの数は、IO-Link デバイスの ProcessInData ・ ProcessOutData のサイズに依存して変化します。

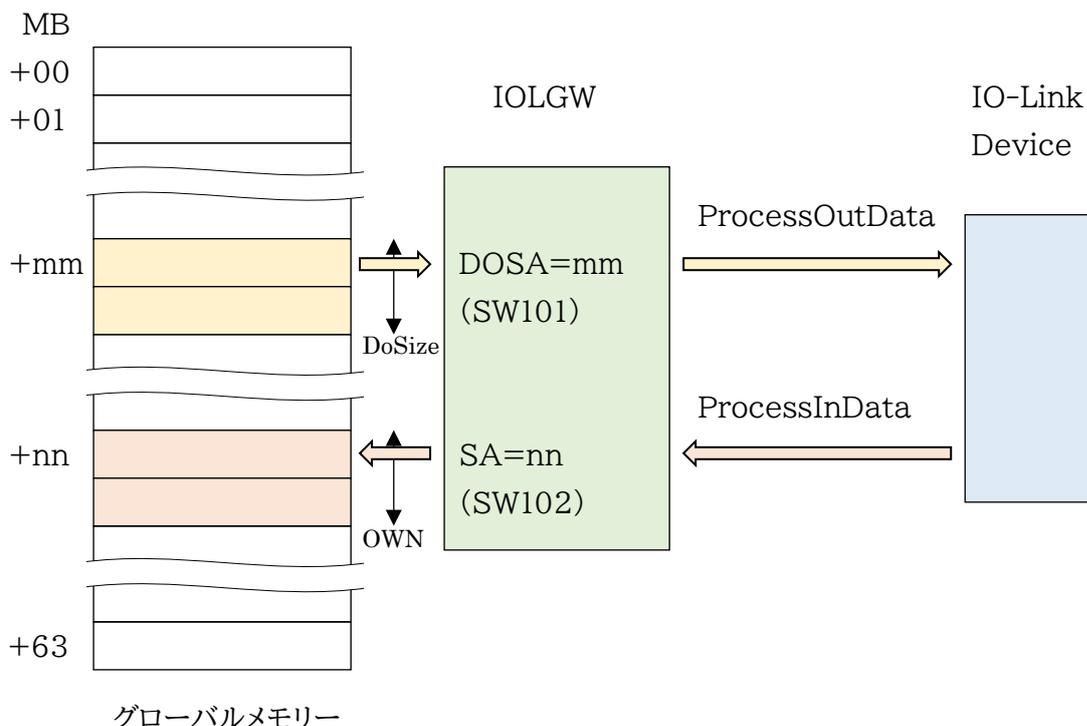


図 2-1 メモリーブロック(グローバルメモリー)

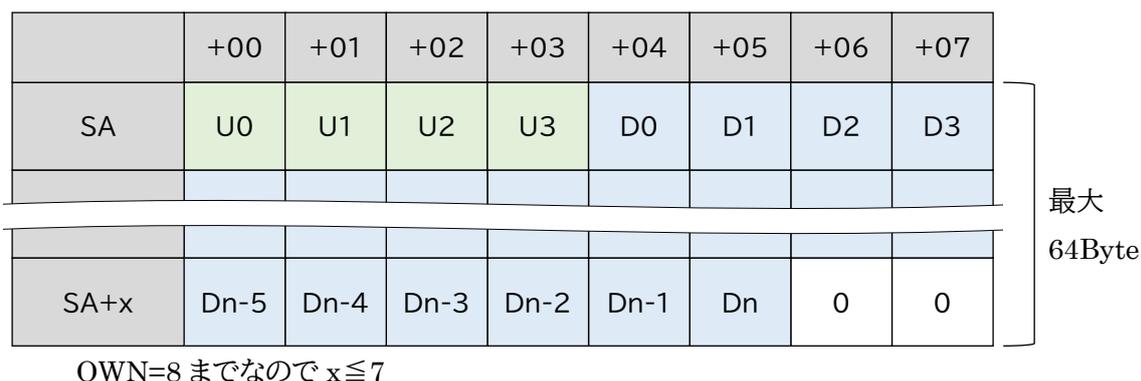
2.1 占有データブロック

SA (Station Address) で指定されたステーションを先頭とし以降のメモリーブロックが、IOLGW で定義する ProcessInData と IO-Link デバイスで使用する ProcessInData を格納する領域となります。

IOLGW の ProcessInData は 4Byte 固定です。IO-Link デバイスの ProcessInData のサイズはそれぞれのデバイスの設定によって可変長となります。

ただし、一つの IO-Link デバイスに割り当てることができるデータサイズは 32Byte までです。

また、全体としては 64Byte まで(メモリーブロックの数としては 8 ブロック[OWN=8])の制限があります。



2.1.1 IOLGW の ProcessInData

IOLGW の ProcessInData 4Byte の内容は次の通りです。

表 2-1 IOLGW が使用する ProccessInData

Byte	ビット配置	名称	
U0	b0	Input1Ch0	Input1(Pin4 C/Q)の信号の状態 0=OFF 1=ON ポートのモードが入力状態でなくとも単に信号の状態を表示している
	b1	Input1Ch1	
	b2	Input1Ch2	
	b3	Input1Ch3	
	b4	Input1Ch4	
	b5	Input1Ch5	
	b6	Input1Ch6	
	b7	Input1Ch7	
U1	b0	Input2Ch0	Input2(Pin2 I/Q)の信号の状態 0=OFF 1=ON ポートのモードが入力状態でなくとも単に信号の状態を表示している
	b1	Input2Ch1	
	b2	Input2Ch2	
	b3	Input2Ch3	
	b4	Input2Ch4	
	b5	Input2Ch5	
	b6	Input2Ch6	
	b7	Input2Ch7	
U2	b3-0	ErrorPort	エラーが発生しているポート番号
	b4	reserved	未使用
	b5	CUFlg	CUnet のエラー状態 0=正常 1=エラー発生
	b6	OCFlg	OverCurrent 状態 0=正常 1=過電流検出
	b7	ErrFlg	エラー状態 0=正常 1=エラー発生
U3	b3-0	EventPort	イベントが発生しているポート番号
	b4	reserved	未使用
	B5	infoRdy	通信でデバイスの情報を取得できた 0=未取得 1=取得済(CMD=0x44 で読み出せる)
	b6	IOLRdy	IO-Link デバイスがレディになっている 0=NotReady 1=オペレーションフェーズになっている
	b7	EvtFlg	イベント発生状態 0=正常 1=イベント発生

2.1.2 IO-Link デバイスの ProcessInData

IO-Link デバイスの ProcessInData は後述するメール通信コマンドで、8 ポート個別にバイト数を変更することができます。

初期値としては各ポートが 2Byte となっていることから次のレイアウトとなっています。

(このため OWN の初期値は 3)

表 2-2 IO-Link デバイスを使用する ProcessInData

Byte	名称	
D0	Port0[0]	Port0 の ProcessInData
D1	Port0[1]	
D2	Port1[0]	Port1 の ProcessInData
D3	Port1[1]	
D4	Port2[0]	Port2 の ProcessInData
D5	Port2[1]	
D6	Port3[0]	Port3 の ProcessInData
D7	Port3[1]	
D8	Port4[0]	Port4 の ProcessInData
D9	Port4[1]	
D10	Port5[0]	Port5 の ProcessInData
D11	Port5[1]	
D12	Port6[0]	Port6 の ProcessInData
D13	Port6[1]	
D14	Port7[0]	Port7 の ProcessInData
D15	Port7[1]	

各ポートは 0~32Byte の領域を指定でき、0 が設定された場合は、次のポートが詰めて配置されます。

IO-Link デバイスで実際に使用されるデータサイズより大きな値を設定して、バウンダリを揃えるなどの対応も可能です。

しかし、各ポートの合計が 60(64-4)Byte を超えることはできません。

2.2 データ出力メモリーブロック

DOSA (Data Out Station Address) を指定されたステーションを先頭として以降のメモリーブロックが、IOLGW の ProcessOutData と IO-Link デバイスの ProcessOutData を参照します。

IOLGW の ProcessOutData は 2Byte 固定です。IO-Link デバイスの ProcessOutData のサイズはそれぞれのデバイスの設定によって可変長となります。

こちらも全体としては 64Byte まで(メモリーブロックの数としては 8 ブロック)の制限があります。

	+00	+01	+02	+03	+04	+05	+06	+07	
DOSA	U0	U1	D0	D1	D2	D3	D4	D5	最大 64Byte
DOSA+x	Dn-5	Dn-4	Dn-3	Dn-2	Dn-1	Dn	0	0	

DOSize=8 までなので $x \leq 7$

また、一つの IO-Link デバイスに割り当てることができるデータサイズは 32Byte までとなっています。

2.2.1 IOLGW の ProcessOutData

IOLGW の ProcessOutData 2Byte の内容は次の通りです。

表 2-3 IOLGW が使用する ProcessOutData

Byte	ビット配置	名称	
U0	b0	OutputCh0	Output1(Pin4 C/Q)の信号の状態 0=OFF 1=ON ポートのモードを出力に設定した時の出力状態を指示する
	b1	OutputCh1	
	b2	OutputCh2	
	b3	OutputCh3	
	b4	OutputCh4	
	b5	OutputCh5	
	b6	OutputCh6	
	b7	OutputCh7	
U1	b2-0	reserved	未使用
	b3	EvtClr	イベント発生状態をクリアする 0→1=クリア
	b6-4	reserved	未使用
	b7	ErrClr	エラー発生状態をクリアする 0→1=クリア

クリア動作は対応するビットが 0→1 に変化することで実行されます。このため再度クリアする場合は、実行後に 0 を書き込むか、実行前に一度 0 を書いてから 1 にする必要があります。

2.2.2 IO-Link デバイスの ProcessOutData

IO-Link デバイスの ProcessOutData は後述するメール通信コマンドで、8ポート個別にバイト数を変更できます。

初期値としては各ポートが 2Byte となっていることから次のレイアウトとなっています。

(このため DOSA で参照するサイズ[DOSIZE]の初期値は 3)

表 2-4 IO-Link デバイスが使用する ProcessOutData

Byte	名称	
D0	Port0[0]	Port0 の ProcessOutData
D1	Port0[1]	
D2	Port1[0]	Port1 の ProcessOutData
D3	Port1[1]	
D4	Port2[0]	Port2 の ProcessOutData
D5	Port2[1]	
D6	Port3[0]	Port3 の ProcessOutData
D7	Port3[1]	
D8	Port4[0]	Port4 の ProcessOutData
D9	Port4[1]	
D10	Port5[0]	Port5 の ProcessOutData
D11	Port5[1]	
D12	Port6[0]	Port6 の ProcessOutData
D13	Port6[1]	
D14	Port7[0]	Port7 の ProcessOutData
D15	Port7[1]	

各ポートは 0~32Byte の領域を指定でき、0 が設定された場合は、次のポートが詰めて配置されます。各ポートの合計が 62(64-2)Byte を超えることはできません。

◇ NoDataOut モード

接続されている全てのデバイスが入力用である場合、NoDataOut モードを有効とすると、CUnet の DOSA を使用せず、ステーションアドレスを消費しなくてよくなります。

ただし、IOLGW の ProcessOutData も使用しないこととなります。

次の条件を満たす場合、DOSA を参照しない NoDataOut モードを選択することができます。

- ・全ポートで ioMode が IO-Link モードもしくは入力ポートに設定されている
- ・全ポートのプロセス出力データサイズがゼロに設定されている
- ・メールコマンドで NoDataOut モードが指定されている

このモード設定がされていても ioMode がデジタル出力ポートであったり、プロセス出力データサイズがゼロ以外であったりすれば、動作は自動的に解除されます。

3. メール通信機能

IOLGWの機能設定、IO-Link デバイスとの通信は、CUnet のメール機能を使って実現します。

リード・ライト共にコマンドデータをメイン・ステーション側から送り、それに対してレスポンスデータを受信するシーケンスとなっています。このためレスポンスがないことも想定し、タイムアウトを設定してリトライやエラー処理を行うことを組み込むようお願いします。

コマンドデータ (メイン・ステーション → IOLGW)

Offset	+0		+1	+2	+3	+4	+5	+6	+7
+00	CMD00		CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CMD07
+08	CMD08		CMD09	CMD0A	CMD0B	CMD0C	CMD0D	CMD0E	CMD0F
+08×n	CMDxx		CMDxx	CMDxx	CMDxx	CMDxx	CMDxx	CMDxx	CHK

レスポンスデータ (IOLGW→ メイン・ステーション)

Offset	+0	+1	+2	+3	+4	+5	+6	+7
+00	RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	RSP07
+08	RSP08	RSP09	RSP0A	RSP0B	RSP0C	RSP0D	RSP0E	RSP0F
+08×n	RSPxx	CHK						

CUnet のメール機能は 8Byte 単位の送受信のため通信するデータサイズは 8 の整数倍としています。この時、最後尾バイトにはチェックサムを配置し、データの品質確保に努めています。

チェックサムの計算方法は、直前までのデータをバイト単位で EXOR を取ったものです。
 例)

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
0xA1	0x02	0x01	0x00	0x00	0x02	0x01	
	0xA3						
		0xA2					
			0xA2				
				0xA2			
					0xA0		
						0xA1	
							0xA1

コマンドデータの先頭バイトでコマンドの内容を示しています。
このうち最上位のビット7を R/W（読み出し・書き込み）の識別として使用しています。

CMD0[7] : R/W '0'=読み出し、'1'=書き込み

CMD0[6:0] : コマンド種別

コマンドによっては、読み出し専用となっているもの・書き込み専用となっているものもあります。
詳細につきましては、それぞれのコマンドの説明をご参照ください。

表 3-1 コマンド種別一覧表

CMD0[6:0]	名称	機能
0x20	PortConfig	I/O ポートの動作条件
0x21	ProcessdataSize	各デバイスのプロセスデータサイズ
0x22	VerifyConfig	照合機能の情報
0x23	NoDataOut	DOSA 未使用モード設定
0x30	DataStorage	データストレージ機能
0x40	Diagnosis	診断情報
0x41	Information	ユニット情報
0x42	NetworkStatus	LCARE/MCARE 情報
0x43	CUnet	ユニット識別情報
0x44	DeviceInformation	デバイス情報
0x5n	ISDU 通信	ISDU 通信
0x60	ProcessData	プロセスデータ
0x70	ConfigReset	設定初期化

3.1 PortConfig(0x20)

各ポートの動作状態を設定・確認するコマンドです。

下表の説明の中で黄色にマーキングされている値は、工場出荷時もしくは IOLGW 初期化時に設定される値を意味しています。

表 3-2 PortConfig コマンドの Item 番号

ItemNo	ビット配置	名称	説明
0	b2-0	ioMode	ポートの動作モード 000=無効モード 001=IO-Link モード 010=PNP 入力モード 011=NPN 入力モード 100=PNP 出力モード 101=NPN 出力モード
	b4-3	Verify	デバイス照合機能 00=照合機能を使用しない 01=VenderID/DeviceID の一致を確認 10=SerialNo の一致まで確認
	b6-5	Backup/Restore	自動バックアップ・リストア機能 00=無効 01=バックアップ有効 10=リストア有効 11=バックアップ・リストア有効
	b7	Sync	ポート間同期 0=無効 (Free-Run) 1=有効
1	b1-0	OutputClear	異常時の出力データ設定 00=出力 OFF 01=出力 ON 10=直前の値を保持
	b2	OverCurPwr	電源過電流検知機能 0=無効 1=有効
	b3	OverCurDout	出力過電流検知機能 0=無効 1=有効
	b4	Recovery	異常時から復帰する時の出力データ設定 0=自動復帰 1=異常時の値を保持
	b7-5	reserved	未使用

(次項へ続く)

表 3-2 PortConfig コマンドの Item 番号 (つづき)

ItemNo	ビット配置	名称	説明
2	b2-0	InputFilter	入力ポートのフィルター時間設定 000 = 0.1 msec 001 = 1 msec 010 = 5 msec 011 = 10 msec 100 = 20 msec
	b3	reserved	未使用
	b5-4	InputHold	入力ホールド時間設定 00 = 1 msec 01 = 15 msec 10 = 100 msec 11 = 200 msec
	b7-6	Reserved	未使用
3	b7-0	CycleTime	ポートのサイクル時間設定 0 = 自動設定 1~3 = 設定禁止 4~255 = × 0.1 msec

有効でない値を設定すると「Out of range(パラメーターの範囲が不適切)」のエラーが返ります。

3.1.1 OpeMode[0](ItemNo=0)

動作モードのうち、ポートの動作モード(ioMode)・デバイス照合機能(Verify)・自動バックアップ・リストア機能(Backup/Restore)・ポート間同期(Sync)に対して設定・確認するコマンドです。

- 書き込み(CMD0[7]='1')
オペレーションモードを先頭から書き込みます。

Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	cLEN	CH	OFS	INUM	wData0	wData1	CHK
0xA0	0x01	0x01	0x00	0x00	0x01	0x00	0x84

cLEN : 書き込みアイテム数

0x01=OpeMode[0]のみ書き込み(1Byte)

0x02=OpeMode[1:0]を書き込み(2Byte)

0x03=OpeMode[2:0]を書き込み(3Byte)

0x04=OpeMode[3:0]を書き込み(4Byte)

CH : 書き込み対象のポート (ポート番号をビットで指定)

OFS : オフセット指定

INUM : ItemNo. = 0

wData : 書き込みデータ

cLEN にアイテム数を入力することで連続同時書き込みが可能となります。この時送信データ数が不足しているとレンジエラーになります。

CH で複数のビットを設定するとそのポートに一括で値が書き込まれます。

OFS はバイト単位での書き込み時は 0x00 を使用します。バイト内に割り当てられている一部の項目だけを書き込む場合にそのオフセット値(ビット)を指定します。

wData0 の 1 バイトデータの中身は次のようなレイアウトとなっています。

	b7	b6	b5	b4	b3	b2	b1	b0
wData0	Sync	Backup/Restore	Verify		ioMode			

cLen を 0x04、INUM を 0x80 に、OFS を 0x80 して 4 バイトのデータを送ると、全ポートに対して一括して書き込むことができます。ただしこの時 CH のデータは 0x00 にしておかないといけません。

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CMD07
CMD	cLEN	CH	OFS	INUM	wData0	wData1	wData2
0xA0	0x04	0x00	0x80	0x80	OpMode[0]	OpMode[1]	OpMode[2]
CMD08	CMD09	CMD0A	CMD0B	CMD0C	CMD0D	CMD0E	CHK
wData3	Rsv0	Rsv1	Rsv2	Rsv3	Rsv4	Rsv5	CHK
OpMode[3]	0x00	0x00	0x00	0x00	0x00	0x00	CHK

Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
0xA0	rLEN	STAT	Rsv0	Rsv1	Rsv2	Rsv3	CHK

rLEN : 返答するデータ長

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
[7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

Rsv : 0x00 が返答されます

- 読み出し(CMD0[7]='0')
 オペレーションモードを先頭から読み出します。

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	cLEN	CH	OFS	INUM	Rsv1	Rsv2	CHK
0x20	0x01	0x01	0x00	0x00	0x00	0x00	0x22

cLEN : 読み出すアイテム数

0x01=OpeMode[0]のみ読み出し(1Byte)

0x02=OpeMode[1:0]を読み出し(2Byte)

0x03=OpeMode[2:0]を読み出し(3Byte)

0x04=OpeMode[3:0]を読み出し(4Byte)

CH : 読み出す対象のポート (ポート番号をビットで指定)

OFS : オフセット指定

INUM : ItemNo. = 0

cLEN にアイテム数を入力することで連続読み込みが可能となります。

CH で複数のビットを設定するとそのポートの値を順に応答します。ただし、この時は複数アイテムを読み出すことはできません。OpeMode[0]のみ複数ポート分読み出すことができます。

OFS はバイト単位での読み出し時は 0x00 を使用します。バイト内に割り当てられている一部の項目だけを読み出す場合にそのオフセット値(ビット)を指定します。

Rsv は無視されますが 0x00 を使用することを推奨します。

Command 送信時の INUM を 0x80 に設定して全ポートを読み出すことは対応していません。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
0x20	rLEN	STAT	rData0	rData1	rData2	rData3	CHK

rLEN : 返答するデータ長

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
 [7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

rData : 返答データ

rData としてデータが返答されます。

3.1.2 OpeMode[1](ItemNo=1)

動作モードのうち、異常時の出力データ設定(OutputClear)・電源過電流検知機能(OverCurPwr)・出力過電流検知機能(OverCurDout)・異常時から復帰する時の出力データ設定(Recovery) に対して設定・確認するコマンドです。

- 書き込み(CMD0[7]='1')
オペレーションモードを2バイト目から書き込みます。

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	cLEN	CH	OFS	INUM	wData0	wData1	CHK
0xA0	0x01	0x01	0x00	0x01	0x08	0x00	0x84

cLEN : 書き込みアイテム数

0x01=OpeMode[1]のみ書き込み(1Byte)

0x02=OpeMode[2:1]を書き込み(2Byte)

0x03=OpeMode[3:1]を書き込み(3Byte)

CH : 書き込み対象のポート (ポート番号をビットで指定)

OFS : オフセット指定

INUM : ItemNo. = 1

wData : 書き込みデータ

cLEN にアイテム数を入力することで連続同時書き込みが可能となります。この時送信データ数が不足しているとレンジエラーになります。

CH で複数のビットを設定するとそのポートに一括で値が書き込まれます。

OFS はバイト単位での書き込み時は 0x00 を使用します。バイト内に割り当てられている一部の項目だけを書き込む場合にそのオフセット値(ビット)を指定します。

wData0 の1バイトデータの中身は次のようなレイアウトとなっています。

	b7	b6	b5	b4	b3	b2	b1	b0
wData0	0	0	0	Recovery	OverCurDout	OverCurPwr	OutputClear	

Command 送信時の INUM を 0x80 に設定して全ポートを書き込むことは対応していません。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
0xA0	rLEN	STAT	Rsv0	Rsv1	Rsv2	Rsv3	CHK

rLEN : 返答するデータ長

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)

[7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

Rsv : 0x00 が返答されます

- 読み出し(CMD0[7]='0')
 オペレーションモードを2バイト目から読み出します。

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	cLEN	CH	OFS	INUM	Rsv1	Rsv2	CHK
0x20	0x01	0x01	0x00	0x01	0x00	0x00	0x22

cLEN : 読み出すアイテム数

0x01=OpeMode[1]のみ読み出し(1Byte)

0x02=OpeMode[2:1]を読み出し(2Byte)

0x03=OpeMode[3:1]を読み出し(3Byte)

CH : 読み出す対象のポート (ポート番号をビットで指定)

OFS : オフセット指定

INUM : ItemNo. = 1

cLEN にアイテム数を入力することで連続読み込みが可能となります。

CH で複数のビットを設定するとそのポートの値を順に応答します。ただし、この時は複数アイテムを読み出すことはできません。OpeMode[1]のみ複数ポート分読み出すことができます。

OFS はバイト単位での読み出し時は 0x00 を使用します。バイト内に割り当てられている一部の項目だけを読み出す場合にそのオフセット値(ビット)を指定します。

Rsvは無視されますが 0x00 を使用することを推奨します。

Command 送信時の INUM を 0x80 に設定して全ポートを読み出すことは対応していません。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
0x20	rLEN	STAT	rData0	rData1	rData2	rData3	CHK

rLEN : 返答するデータ長

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)

[7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

rData : 返答データ

rData としてデータが返答されます。

3.1.3 OpeMode[2](ItemNo=2)

動作モードのうち、入力ポートのフィルター時間設定(InputFilter)・入力ホールド時間設定(InputHold) に対して設定・確認するコマンドです。

- 書き込み(CMD0[7]='1')
オペレーションモードを3バイト目から書き込みます。

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	cLEN	CH	OFS	INUM	wData0	wData1	CHK
0xA0	0x01	0x01	0x00	0x02	0x00	0x00	0x84

cLEN : 書き込みアイテム数

0x01=OpeMode[2]のみ書き込み(1Byte)

0x02=OpeMode[3:2]を書き込み(2Byte)

CH : 書き込み対象のポート (ポート番号をビットで指定)

OFS : オフセット指定

INUM : ItemNo. = 2

wData : 書き込みデータ

cLEN にアイテム数を入力することで連続同時書き込みが可能となります。この時送信データ数が不足しているとレンジエラーになります。

CH で複数のビットを設定するとそのポートに一括で値が書き込まれます。

OFS はバイト単位での書き込み時は 0x00 を使用します。バイト内に割り当てられている一部の項目だけを書き込む場合にそのオフセット値(ビット)を指定します。

wData0 の1バイトデータの中身は次のようなレイアウトとなっています。

	b7	b6	b5	b4	b3	b2	b1	b0
wData0	0	0	InputHold		0	InputFilter		

Command 送信時の INUM を 0x80 に設定して全ポートを書き込むことは対応していません。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
0xA0	rLEN	STAT	Rsv0	Rsv1	Rsv2	Rsv3	CHK

rLEN : 返答するデータ長

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
[7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

Rsv : 0x00 が返答されます

- 読み出し(CMD0[7]='0')
 オペレーションモードを3バイト目から読み出します。

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	cLEN	CH	OFS	INUM	Rsv1	Rsv2	CHK
0x20	0x01	0x01	0x00	0x02	0x00	0x00	0x22

cLEN : 読み出すアイテム数

0x01=OpeMode[2]のみ読み出し(1Byte)

0x02=OpeMode[3:2]を読み出し(2Byte)

CH : 読み出す対象のポート (ポート番号をビットで指定)

OFS : オフセット指定

INUM : ItemNo. = 2

cLEN にアイテム数を入力することで連続読み込みが可能となります。

CH で複数のビットを設定するとそのポートの値を順に応答します。ただし、この時は複数アイテムを読み出すことはできません。OpeMode[2]のみ複数ポート分読み出すことができます。

OFS はバイト単位での読み出し時は 0x00 を使用します。バイト内に割り当てられている一部の項目だけを読み出す場合にそのオフセット値(ビット)を指定します。

Rsv は無視されますが 0x00 を使用することを推奨します。

Command 送信時の INUM を 0x80 に設定して全ポートを読み出すことは対応していません。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
0x20	rLEN	STAT	rData0	rData1	rData2	rData3	CHK

rLEN : 返答するデータ長

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
 [7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

rData : 返答データ

rData としてデータが返答されます。

3.1.4 CycleTime(ItemNo=3)

動作モードのうち、ポートのサイクル時間設定(CycleTime) に対して設定・確認するコマンドです。

- 書き込み(CMD0[7]='1')
 オペレーションモードの4バイト目を書き込む。

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	cLEN	CH	OFS	INUM	wData0	wData1	CHK
0xA0	0x01	0x01	0x00	0x03	0x00	0x00	0x84

cLEN : 書き込みアイテム数

0x01=OpeMode[3]のみ書き込み(1Byte)

CH : 書き込み対象のポート (ポート番号をビットで指定)

OFS : オフセット指定

INUM : ItemNo. = 3

wData : 書き込みデータ

cLEN で指定できるのは 0x01 のみです。

CH で複数のビットを設定するとそのポートに一括で値が書き込まれます。

OFS はバイト単位での書き込みのみとなり 0x00 を使用します。

wData0 に CycleTime データを入れてコマンドを送信します。

Command 送信時の INUM を 0x80 に設定して全ポートを書き込むことは対応していません。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
0xA0	rLEN	STAT	Rsv0	Rsv1	Rsv2	Rsv3	CHK

rLEN : 返答するデータ長

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
 [7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

Rsv : 0x00 が返答されます

- 読み出し(CMD0[7]='0')
 オペレーションモードの4バイト目を読み出す。

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	cLEN	CH	OFS	INUM	Rsv1	Rsv2	CHK
0x20	0x01	0x01	0x00	0x03	0x00	0x00	0x22

cLEN : 読み出すアイテム数

0x01=OpeMode[3]のみ読み出し(1Byte)

CH : 読み出す対象のポート (ポート番号をビットで指定)

OFS : オフセット指定

INUM : ItemNo. = 3

cLEN で指定できるのは 0x01 のみです。

CH で複数のビットを設定するとそのポートの値を順に応答します。ただし、この時は複数アイテムを読み出すことはできません。OpeMode[3]のみ複数ポート分読み出すことができます。

Rsv は無視されますが 0x00 を使用することを推奨します。

Command 送信時の INUM を 0x80 にして全ポートを読み出すことは対応していません。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
0x20	rLEN	STAT	rData0	rData1	rData2	rData3	CHK

rLEN : 返答するデータ長

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
 [7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

rData : 返答データ

rData0 からデータが返ります。

3.2 ProcessDataSize(0x21)

各ポートに接続されているデバイスのプロセスデータ領域のサイズを設定・確認するコマンドです。

表 3-3 ProcessDataSize コマンドの Item 番号

ItemNo	名称	データ長	
0	UProcInByte	1	プロセス入力データのサイズ
1	UProcOutByte	1	プロセス出力データのサイズ
2	AutoAlloc	1	自動割振り指示 (書き込みのみ)

3.2.1 UProcInByte(ItemNo=0)

UProcInByte は、ユニットが持つデバイスの入力データを格納する領域のバイト数を表しています。ブロックメモリの SA から始まる領域で、UnitData に続く DeviceData 部分にポート順にマッピングされます。

- 書き込み(CMD0[7]='1')
- 入力データを格納する領域のバイト数を書き込みます。

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	cLEN	CH	Rsv0	INUM	wData0	wData1	CHK
0xA1	0x01	0x01	0x00	0x00	0x02	0x00	0xA3

cLEN : 書き込みアイテム数

0x01=UProcInByte のみ書き込み(1Byte)

0x02=UProcInByte と UProcOutByte を書き込み(2Byte)

CH : 書き込み対象のポート (ポート番号をビットで指定)

INUM : ItemNo. = 0

wData : 書き込みデータ

cLEN にアイテム数を入力することで連続同時書き込みが可能となります。この時送信データ数が不足しているとレンジエラーになります。

CH で複数のビットを設定するとそのポートに一括で値が書き込まれます。

Rsv0 は無視されますが 0x00 を使用することを推奨します。

INUM を 0x80 にすると全ポートに対して一括して書き込むことができます。ただし、この時 CH を 0x00 にしておかないとエラーとなります。

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	cLEN	CH	Rsv0	INUM	wData0	wData1	CHK
0xA1	0x02	0x00	0x00	0x80	UProcInByte	UProcOutByte	CHK

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
0xA1	rLEN	STAT	Rsv0	Rsv1	Rsv2	Rsv3	CHK

rLEN : 返答するデータ長

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
[7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

Rsv : 0x00 が返答されます

- 読み出し(CMD0[7]='0')
 現在設定されている入力データを格納する領域のバイト数を読み出します。

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	cLEN	CH	OFS	INUM	Rsv0	Rsv1	CHK
0x21	0x01	0x01	0x00	0x00	0x00	0x00	0x21

cLEN : 読み出すアイテム数

0x01=UProcsInByte のみ書き込み(1Byte)

0x02=UProcsInByte と UProcsOutByte を書き込み(2Byte)

CH : 読み出す対象のポート (ポート番号をビットで指定)

OFS : オフセット指定

INUM : ItemNo. = 0

cLEN にアイテム数を入力することで連続読み込みが可能となります。

CH で複数のビットを設定するとそのポートの値を順に応答します。ただし、この時は複数アイテムを読み出すことはできません。UProcInByte のみ複数ポート分読み出すことができます。

Rsv は無視されますが 0x00 を使用することを推奨します。

Command 送信時の INUM を 0x80 に設定して全ポートを読み出すことは対応していません。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
0x21	rLEN	STAT	rData0	rData1	rData2	rData3	CHK

rLEN : 返答するデータ長

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
 [7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

rData : 返答データ

rData としてデータが返答されます。

3.2.2 UProcOutByte(ItemNo=1)

UProcOutByte はユニットが持つデバイスへの出力データを格納する領域のバイト数を表しています。ブロックメモリの DOSA から始まる領域で、UnitData に続く DeviceData 部分にポート順にマッピングされます。

- 書き込み(CMD0[7]='1')
 出力データを格納する領域のバイト数を書き込みます。

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	cLEN	CH	Rsv0	INUM	wData0	wData1	CHK
0xA1	0x01	0x01	0x00	0x01	0x02	0x00	0xA2

cLEN : 書き込みアイテム数

0x01=UProcOutByte のみ書き込み(1Byte)

CH : 書き込み対象のポート (ポート番号をビットで指定)

INUM : ItemNo. = 1

wData : 書き込みデータ

cLEN で指定できるのは 0x01 のみです。

CH で複数のビットを設定するとそのポートに一括で値が書き込まれます。

Rsv0 は無視されますが 0x00 を使用することを推奨します。

Command 送信時の INUM を 0x80 に設定して全ポートを書き込むことは対応していません。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
0xA1	rLEN	STAT	Rsv0	Rsv1	Rsv2	Rsv3	CHK

rLEN : 返答するデータ長

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
 [7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

Rsv : 0x00 が返答されます

- 読み出し(CMD0[7]='0')
 現在設定されている出力データを格納する領域のバイト数を読み出します。

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	cLEN	CH	OFS	INUM	Rsv0	Rsv1	CHK
0x21	0x01	0x01	0x00	0x01	0x00	0x00	0x20

cLEN : 読み出すアイテム数

0x01=UProcOutByte のみ書き込み(1Byte)

CH : 読み出す対象のポート (ポート番号をビットで指定)

OFS : オフセット指定

INUM : ItemNo. = 1

cLEN で指定できるのは 0x01 のみです。

CH で複数のビットを設定するとそのポートの値を順に応答します。ただし、この時は複数アイテムを読み出すことはできません。UProcOutByte のみ複数ポート分読み出すことができます。

Rsv は無視されますが 0x00 を使用することを推奨します。

Command 送信時の INUM を 0x80 にして全ポートを読み出すことは対応していません。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
0x21	rLEN	STAT	rData0	rData1	rData2	rData3	CHK

rLEN : 返答するデータ長

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
 [7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

rData : 返答データ

rData0 からデータが返ります。

3.2.3 AutoAlloc(ItemNo=2)

このコマンドは、現在取り付けられているデバイスから ProcessInData・ProcessOutData のサイズを読み込み、自動的に格納するものです。

このためコマンドとして読み出しは機能せず、書き込みだけが対応しています。

- 書き込み(CMD0[7]='1')

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	cLEN	CH	Rsv0	INUM	EXE	Rsv1	CHK
0xA1	0x01	0x01	0x00	0x02	0x01	0x00	0xA2

cLEN : 書き込みアイテム数(0x01)

CH : 書き込み対象のポート (ポート番号をビットで指定)

INUM : ItemNo. = 2

EXE : 0x01=実行指示

cLEN で指定できるのは 0x01 のみです。

CH で複数のビットを設定するとそのポートに一括で処理を指示できます。

EXE に対応する値を書き込むことで、現在接続されている IO-Link デバイスから ProcessData サイズを取り込みます。

Rsv0 は無視されますが 0x00 を使用することを推奨します。

Command 送信時の INUM を 0x80 に設定して全ポートに処理を指示することは対応していません。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
0xA1	rLEN	STAT	Rsv0	Rsv1	Rsv2	Rsv3	CHK

rLEN : 返答するデータ長

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
 [7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

Rsv : 0x00 が返答されます

3.3 VerifyConfig(0x22)

IO-Link デバイスの照合機能について設定・確認するコマンドです。

照合機能で使用する VenderID・DeviceID・SerialNo を読み書きできます。これを ItemNo で指定して読み書きを行います。

表 3-4 VerifyConfig コマンドの Item 番号

ItemNo	名称	データ長	
0	VenderID	2	デバイスのベンダーコード
1	DeviceID	4	デバイスの製品コード (IO-Link 通信リビジョン)
2	SerialNo	16	デバイスの製造番号
3	AutoCfg	1	デバイスから読み込み指示(書き込みのみ)

3.3.1 VenderID(ItemNo=0)

VenderID は 2Byte データの IO-Link デバイスのメーカーを識別するものです。そのコードは IO-Link 協会に登録されているものです。

- 書き込み(CMD0[7]=' 1')
照合に使用する VenderID を書き込みます。

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	cLEN	CH	Rsv0	INUM	wData0	wData1	CHK
0xA2	0x01	0x01	0x00	0x00	0x34	0x12	0x84

cLEN : 書き込みアイテム数

0x01=VenderID のみ書き込み(2Byte)

0x02=VenderID と DeviceID 書き込み(6Byte)

0x03=VenderID・DeviceID と SerialNo 書き込み(22Byte)

CH : 書き込み対象のポート (ポート番号をビットで指定)

INUM : ItemNo. = 0

wData : 書き込みデータ

cLEN にアイテム数を入力することで連続同時書き込みが可能となります。この時送信データ数が不足しているとレンジエラーになります。

CH で複数のビットを設定するとそのポートに一括で値が書き込まれます。

INUM を 0x80 にすると全ポートに対して一括して書き込むことができます。ただし、この時 CH を 0x00 にしておかないとエラーとなります。

wData0 に下位 8 ビット、wData1 に上位 8 ビットの VenderID を入れてコマンドを送信します。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
0xA2	rLEN	STAT	Rsv0	Rsv1	Rsv2	Rsv3	CHK

rLEN : 返答するデータ長

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
[7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

Rsv : 0x00 が返答されます

- 読み出し(CMD0[7]=' 0')
 現在設定されている照合用の VenderID を読み出します。

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	cLEN	CH	Rsv0	INUM	Rsv1	Rsv2	CHK
0x22	0x01	0x01	0x00	0x00	0x00	0x00	0x22

cLEN : 読み出すアイテム数

0x01=VenderID のみ要求(2Byte)

0x02=VenderID と DeviceID 要求(6Byte)

0x03=VenderID ・ DeviceID と SerialNo 要求(22Byte)

CH : 読み出す対象のポート (ポート番号をビットで指定)

INUM : ItemNo. = 0

cLEN にアイテム数を入力することで連続読み込みが可能となります。

CH で複数のビットを設定するとそのポートの値を順に応答します。ただし、この時は複数アイテムを読み出すことはできません。2バイトずつ指定されたポートの順にデータが返答されます。Rsv は無視されますが 0x00 を使用することを推奨します。

Command 送信時の INUM を 0x80 に設定して全ポートを読み出すことは対応していません。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
0x22	rLEN	STAT	rData0	rData1	0x00	0x00	CHK

rLEN : 返答するデータ長

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
 [7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

rData : 返答データ

rData0 に下位 8 ビット、rData1 に上位 8 ビットが返ります。

一度も VenderID が書き込まれていない状態では 0x00 が読み出されます。

3.3.2 DeviceID(ItemNo=1)

この DeviceID は 4Byte のデータで、そのデバイスがデバイスメーカーの中で識別できるように付けられたものです(型式に 1 対 1 で対応するもの)。

IO-Link の DeviceID の仕様は 3Byte ですが、この DeviceID は RevisionID を最上位バイトに追加した 4Byte として一括で扱っています。

- 書き込み(CMD0[7]='1')
 照合に使用する DeviceID を書き込みます。

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CMD07
CMD	cLEN	CH	Rsv0	INUM	wData0	wData1	wData2
0xA2	0x01	0x01	0x00	0x01	0x56	0x34	0x12
CMD08	CMD09	CMD0A	CMD0B	CMD0C	CMD0D	CMD0E	CHK
wData3	Rsv1	Rsv2	Rsv3	Rsv4	Rsv5	Rsv6	CHK
0x11	0x00	0x00	0x00	0x00	0x00	0x00	0xC2

cLEN : 書き込みアイテム数

0x01=DeviceID のみ書き込み(4Byte)

0x02=DeviceID と SerialNo 書き込み(20Byte)

CH : 書き込み対象のポート (ポート番号をビットで指定)

INUM : ItemNo. = 1

wData : 書き込みデータ

cLEN にアイテム数を入力することで連続同時書き込みが可能となります。この時送信データ数が不足しているとリングスエラーになります。

CH で複数のビットを設定するとそのポートに一括で値が書き込まれます。

INUM を 0x80 にすると全ポートに対して一括して書き込むことができます。ただし、この時 CH を 0x00 にしておかないとエラーとなります。

wData0~wData2 が DeviceID、wData3 が RevisionID を設定してコマンドを送信します。

wData0	wData1	wData2	wData3
DeviceID[7:0]	DeviceID[15:8]	DeviceID[24:16]	RevisionID

Rsv0 は無視されますが 0x00 を使用することを推奨します。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
0xA2	rLEN	STAT	Rsv0	Rsv1	Rsv2	Rsv3	CHK

rLEN : 返答するデータ長

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
 [7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

Rsv : 0x00 が返答されます

- 読み出し(CMD0[7]='0')
 現在設定されている照合用の DeviceID(+RevisionID)を読み出します。

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	cLEN	CH	Rsv0	INUM	Rsv1	Rsv2	CHK
0x22	0x01	0x01	0x00	0x01	0x00	0x00	0x23

cLEN : 読み出すアイテム数

0x01=DeviceID のみ要求(4Byte)

0x02= DeviceID と SerialNo 要求(20Byte)

CH : 読み出す対象のポート (ポート番号をビットで指定)

INUM : ItemNo. = 1

cLEN にアイテム数を入力することで連続読み込みが可能となります。

CH で複数のビットを設定するとそのポートの値を順に応答します。ただし、この時は複数アイテムを読み出すことはできません。4バイトずつ指定されたポートの順にデータが返答されます。Rsv は無視されますが 0x00 を使用することを推奨します。

Command 送信時の INUM を 0x80 に設定して全ポートを読み出すことは対応していません。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
0x22	rLEN	STAT	rData0	rData1	rData2	rData3	CHK

rLEN : 返答するデータ長

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
 [7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

rData : 返答データ

rData0~rData2 に DeviceID で、rData3 に RevisionID が返答されます。

一度も DeviceID が書き込まれていない状態では 0x00 が読み出されます。

3.3.3 SerialNo(ItemNo=2)

SerialNo は 16Byte のデータで、同じ型式のデバイスであっても、異なる素子であることを識別するためのものです。

- 書き込み(CMD0[7]='1')
 照合に使用する SerialNo を書き込みます。

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CMD07
CMD	cLEN	CH	Rsv0	INUM	wData0	wData1	wData2
0xA2	0x01	0x01	0x00	0x02	0x30	0x31	0x32
CMD08	CMD09	CMD0A	CMD0B	CMD0C	CMD0D	CMD0E	CMD0F
wData3	wData4	wData5	wData6	wData7	wData8	wData9	wDataA
0x33	0x34	0x35	0x36	0x37	0x38	0x39	0x41
CMD10	CMD11	CMD12	CMD13	CMD14	CMD15	CMD16	CHK
wDataB	wDataC	wDataD	wDataE	wDataF	Rsv1	Rsv2	CHK
0x42	0x43	0x44	0x45	0x46	0x00	0x00	0xA6

cLEN : 書き込みアイテム数

0x01=SerialNo のみ書き込み(4Byte)

CH : 書き込み対象のポート (ポート番号をビットで指定)

INUM : ItemNo. = 2

wData : 書き込みデータ

cLEN で指定できるのは 0x01 のみです。

CH で複数のビットを設定するとそのポートに一括で値が書き込まれます。

INUM を 0x80 にすると全ポートに対して一括して書き込むことができます。ただし、この時 CH を 0x00 にしておかないとエラーとなります。

wData0~wDataF にシリアル番号として 16Byte の文字列データを入れてコマンドを送信します。文字数が 16Byte に満たない時は残りの部分を 0x00 にしてください。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
0xA2	rLEN	STAT	Rsv0	Rsv1	Rsv2	Rsv3	CHK

rLEN : 返答するデータ長

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
 [7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

Rsv : 0x00 が返答されます

- 読み出し(CMD0[7]='0')
 現在設定されている照合用の SerialNo を読み出します。

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	cLEN	CH	Rsv0	INUM	Rsv1	Rsv2	CHK
0x22	0x01	0x01	0x00	0x02	0x00	0x00	0x20

cLEN : 読み出すアイテム数

0x01= SerialNo のみ要求(16Byte)

CH : 読み出す対象のポート (ポート番号をビットで指定)

INUM : ItemNo. = 2

cLEN で指定できるのは 0x01 のみです。

CH で複数のビットを設定するとそのポートの値を順に応答します。ただし、この時は複数アイテムを読み出すことはできません。16 バイトずつ指定されたポートの順にデータが返答されます。

Rsv は無視されますが 0x00 を使用することを推奨します。

Command 送信時の INUM を 0x80 に設定して全ポートを読み出すことは対応していません。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	RSP07
0x22	rLEN	STAT	rData0	rData1	rData2	rData3	rData4
RSP08	RSP09	RSP0A	RSP0B	RSP0C	RSP0D	RSP0E	RSP0F
rData5	rData6	rData7	rData8	rData9	rDataA	rDataB	rDataC
RSP10	RSP11	RSP12	RSP13	RSP14	RSP15	RSP16	CHK
rDataD	rDataE	rDataF	0x00	0x00	0x00	0x00	CHK

rLEN : 返答するデータ長

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
 [7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

rData : 返答データ

rData0~rDataF にデータが返ります。

一度もシリアル番号が書き込まれていない状態では 0x00 が読み出されます。

3.3.4 AutoCfg(ItemNo=3)

このコマンドは、現在取り付けられているデバイスから DeviceID や SerialNo を読み込み、照合用データとして格納するものです。

このためコマンドとして読み出しは機能せず、書き込みだけが対応しています。

- 書き込み(CMD0[7]='1')

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	cLEN	CH	Rsv0	INUM	EXE	Rsv1	CHK
0xA2	0x01	0x01	0x00	0x03	0x01	0x00	0xA0

cLEN : 書き込みアイテム数(0x01)

CH : 書き込み対象のポート (ポート番号をビットで指定)

INUM : ItemNo. = 3

EXE : 書き込みデータ

0x01= SerialNo の自動設定

0x02= VenderID・DeviceID(+RevisionID)の自動設定

cLEN で指定できるのは 0x01 のみです。

CH で複数のビットを設定するとそのポートに一括で処理を指示できます。

INUM を 0x80 にすると全ポートに対して一括して書き込むことができます。この時 CH のデータは 0x00 にしておかないとエラーとなります。

EXE に対応する値を書き込むことで、現在接続されているデバイスの情報を照合用データとして取り込みます。

Rsv0 は無視されますが 0x00 を使用することを推奨します。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
0xA2	rLEN	STAT	Rsv0	Rsv1	Rsv2	Rsv3	CHK

rLEN : 返答するデータ長

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
 [7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

Rsv : 0x00 が返答されます

3.4 NoDataOut(0x23)

DOSA で示す出力用のメモリーブロックを使用しないモードの設定を行うコマンドです。
 モードを設定しても次の条件を満たしていなければ無視され、DOSA の領域を参照します。

- ・全ポートで ioMode が IO-Link モードもしくは入力ポートに設定されている
 - ・全ポートのプロセス出力データサイズがゼロに設定されている
- ・ 書き込み(CMD0[7]='1')

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	cLEN	Rsv0	Rsv1	INUM	wData0	Rsv2	CHK
0xA3	0x01	0x00	0x00	0x00	0x01	0x00	0xA3

cLEN : 書き込みアイテム数(0x01)

INUM : ItemNo. = 0

wData0 : 0x01=NoDataOut モード、0x00=通常割付

cLEN で指定できるのは 0x01 のみです。

Rsv0 は無視されますが 0x00 を使用することを推奨します。

wData0 に 0x01 を書き込むことで NoDataOut モードを有効にできます。0x00 を書き込むと通常割付けとなります。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
0xB0	rLEN	STAT	Rsv0	Rsv1	Rsv2	Rsv3	CHK

rLEN : 返答するデータ長

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
 [7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

Rsv : 0x00 が返答されます

- 読み出し(CMD0[7]='0')

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	cLEN	Rsv0	Rsv1	INUM	Rsv2	Rsv3	CHK
0x23	0x01	0x00	0x00	0x00	0x00	0x00	0x22

cLEN : 読み出すアイテム数(0x01 固定)

INUM : ItemNo. = 0

cLEN で指定できるのは 0x01 のみです。

Rsv は無視されますが 0x00 を使用することを推奨します。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
CMD	rLEN	STAT	rData	Rsv0	Rsv1	Rsv2	CHK
0x23	0x04	0x00	0x00	0x00	0x00	0x00	0x27

rLEN : 返答するデータ長(0x08 固定)

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
 [7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

rData : 返答データ(0x01=NoDataOut モード、0x00=通常割付)

Rsv : 0x00 が返答されます

3.5 DataStorage(0x30)

IOLGW 内部に用意されているストレージ機能を制御するコマンドです。

表 3-5 DataStorage コマンドの Iyem 番号

ItemNo	名称	
0	StorageInformation	ストレージ情報 (データ長 = 1 × ポート分)
1	StorageClear	ストレージデータの消去
2	ManualBackup	手動バックアップ
3	ManualRestore	手動リストア

I0-Link デバイスでは、ストレージ対象となっている設定値などが変更された場合に、デバイスの情報がバックアップされます。これは I0-Link デバイスから DS_UPLOAD_REQ というイベントが発行されることで実行されています。

この時、IOLGW は OpeMode[0] の設定により次の動作となります。

表 3-6 自動バックアップの動作

		OpeMode[0] の b5 (自動バックアップ)	
		無効	有効
OpeMode[0] の b6 (自動リストア)	無効	何も動作しない	ストレージされているデータのチェックサムと不一致であればバックアップされる
	有効	バックアップされず、ストレージデータが上書きされる	先にストレージされているデータのチェックサムと不一致であればバックアップされる

また IOLGW の機能として、上位側から通信によって Device の設定値が変更された場合に、自動バックアップが有効であればバックアップを実行します。

I0-Link の仕様としては、あくまでもデバイスの設定値が Local で変更された場合にのみイベントが発生することになっています。

3.5.1 StorageInformation(ItemNo=0)

現在のストレージ機能の状態を返答します（読み出しのみ）。

- 読み出し(CMD0[7]='0')

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	cLEN	CH	Rsv0	INUM	Rsv1	Rsv2	CHK
0x30	0x01	0x00	0x00	0x00	0x00	0x00	0x31

cLEN : 読み出すアイテム数(0x01 固定)

CH : 0x00 固定

INUM : ItemNo. = 0

cLEN で指定できるのは 0x01 のみです。

CH は 0x00 固定です。

Rsv は無視されますが 0x00 を使用することを推奨します。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	RSP07
CMD	rLEN	STAT	rData0	rData1	rData2	rData3	rData4
0x30	0x08	0x00	0x00	0x00	0x00	0x00	0x00
RSP08	RSP09	RSP0A	RSP0B	RSP0C	RSP0D	RSP0E	CHK
rData5	rData6	rData7	Rsv0	Rsv1	Rsv2	Rsv3	CHK
0x00	0x38						

rLEN : 返答するデータ長(0x08 固定)

STAT : 応答結果 [1:0]=Status: '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
[7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

rData : 返答データ

各ポート1バイトのデータで、8ポート分のデータを rData0~rData7 に返答します。

返答データは次のようなレイアウトとなっています。

	b7	b6	b5	b4	b3	b2	b1	b0
rData	Valid	IDerr	reserved				Restore	Backup

b0 : (Backup)自動バックアップ機能の状態

0 =無効

1 =有効

b1 : (Restore)自動リストア機能の状態

0 =無効

1 =有効

b5-2 : (reserved)未使用

b6 : (IDerror)データの有効性

0 =VendeID・DeviceID が一致している

1 =VendeID・DeviceID が一致していない

b7 : (Validity)データの有効性

0 =有効なデータがない

1 =有効なデータを持っている

3.5.2 StorageClear(ItemNo=1)

ストレージしているデータを消去するものです（書き込み専用）。

- 書き込み(CMD0[7]='1')

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	cLEN	CH	Rsv0	INUM	EXE	Rsv1	CHK
0xB0	0x01	0x00	0x00	0x01	0x01	0x00	0xB1

cLEN : 書き込みアイテム数(0x01)
 CH : 書き込み対象のポート（ポート番号を数値で指定）
 INUM : ItemNo. = 1
 EXE : 0x01=実行指示

cLEN で指定できるのは 0x01 のみです。

CH はポート番号を数値で指定します。このため、複数のポートに対して同時に処理することはできません。

Rsv は無視されますが 0x00 を使用することを推奨します。

EXE に対応する値を書き込むことでストレージデータが消去されます。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
0xB0	rLEN	STAT	Rsv0	Rsv1	Rsv2	Rsv3	CHK

rLEN : 返答するデータ長
 STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応（'3'=未使用）
 [7:4]=Error（4.2.1 メール通信ステータスエラーコード参照）
 Rsv : 0x00 が返答されます

3.5.3 ManualBackup(ItemNo=2)

強制的にバックアップを指示します（書き込み専用）。

- 書き込み(CMD0[7]='1')

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	cLEN	CH	Rsv0	INUM	EXE	Rsv1	CHK
0xB0	0x01	0x00	0x00	0x02	0x01	0x00	0xB2

cLEN : 書き込みアイテム数(0x01)
 CH : 書き込み対象のポート（ポート番号を数値で指定）
 INUM : ItemNo. = 2
 EXE : 0x01=実行指示

cLEN で指定できるのは 0x01 のみです。

CH はポート番号を数値で指定します。このため、複数のポートに対して同時に処理することはできません。

Rsv は無視されますが 0x00 を使用することを推奨します。

EXE に対応する値を書き込むことで強制的にデバイスからバックアップします。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
0xB0	rLEN	STAT	Rsv0	Rsv1	Rsv2	Rsv3	CHK

rLEN : 返答するデータ長
 STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応（'3'=未使用）
 [7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)
 Rsv : 0x00 が返答されます

3.5.4 ManualRestore(ItemNo=3)

強制的にリストアを指示します（書き込み専用）。

- 書き込み(CMD0[7]='1')

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	cLEN	CH	Rsv0	INUM	EXE	Rsv1	CHK
0xB0	0x01	0x00	0x00	0x03	0x01	0x00	0xB3

cLEN : 書き込みアイテム数(0x01)

CH : 書き込み対象のポート（ポート番号を数値で指定）

INUM : ItemNo. = 3

EXE : 0x01=実行指示

cLEN で指定できるのは 0x01 のみです。

CH はポート番号を数値で指定します。このため、複数のポートに対して同時に処理することはできません。

Rsv は無視されますが 0x00 を使用することを推奨します。

EXE に対応する値を書き込むことで強制的にデバイスへリストアします。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
0xB0	rLEN	STAT	Rsv0	Rsv1	Rsv2	Rsv3	CHK

rLEN : 返答するデータ長

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
 [7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

Rsv : 0x00 が返答されます

3.6 Diagnosis(0x40)

IOLGW や IO-Link デバイスの状態を確認するコマンドです。
 全て読み出しのみをサポートしています。

表 3-7 DiagnosisコマンドのItem 番号

ItemNo	ビット配置	名称	
0	UnitStatus(データ長=2byte) : IOLGW の状態通知		
	b3-0	reserved	未使用
	b4	CUwarn	CUnet 通信品質低下 0 = 正常 1 = 異常
	b5	reserved	未使用
	b6	TEMPerr	内部温度異常 0 = 正常 1 = 異常
	b7	INITerr	初期化異常 0 = 正常 1 = 異常
	b9-8	reserved	未使用
	b10	MAPerr	データマッピング異常 0 = 正常 1 = 異常
	b12-11	reserved	未使用
	b13	MEMerr	内部不揮発性メモリ異常 0 = 正常 1 = 異常
	b15-14	reserved	未使用

(次項へ続く)

表 3-7 Diagnosis コマンドの Item 番号(つづき)

ItemNo	ビット配置	名称	
1	UnitDiag(データ長=4byte) : IOLGW ポートの診断情報		
	b7-0	ErrType	エラー種別 0x00=詳細情報なし 0x01=エラー詳細情報 0x02=ISDU イベント
	b15-8	ErrQual	IO-Link 仕様準拠 b2-0 : Instance 000=不明 001=物理層 010=データリンク層 011=アプリケーション層 100=アプリケーション 101-=reserved b3 : Source 0=機器アプリケーション (リモート) 1=マスターアプリケーション (ローカル) b5-4 : Type 00=reserved 01=情報 10=警告 11=エラー b7-6 : Mode 00=reserved 01=シングルショット 10=イベント非表示 11=イベント表示
	b23-16	AddErrCode	追加エラーコード (別表参照)
	b31-24	ErrCode	エラーコード (別表参照)
2	DeviceDiag(データ長=4byte) : デバイスの診断情報 (内容は Item#1 と同じ)		
3	DiagPort(データ長=3byte) : 診断情報発生ポート		
	b7-0	ErrCH	エラー発生ポート 各ビットがポートを意味する
	b15-8	EventCH	イベント発生ポート 各ビットがポートを意味する
	b23-16	OverCurCH	過電流発生ポート 各ビットがポートを意味する

(次項へ続く)

表 3-7 Diagnosis コマンドの Item 番号 (つづき)

ItemNo	ビット配置	名称	
4	COMStatus(データ長=8byte) : IO-Link 通信状態 各ポート1バイトで合計 8Byte		
	b1-0	Status	ステータス 00=未接続 01=スタートアップフェーズ 10=プリオペレーションフェーズ 11=オペレーションフェーズ
	b2	Revision	リビジョン 0=Rev. 1.1 1=Rev. 1.0
	b6-3	reserved	
	b7	infoRdy	通信でデバイスの情報を取得できたかを示す 0=未取得 1=取得済(情報は CMD=0x44 で読み出し可能)
5	COMError (データ長=8byte) : IO-Link 通信エラー回数 各ポート1バイトで合計 8Byte		
6	FPGATemp(データ長=2byte) : ユニット内部温度 使用している FPGA の内部温度を 0.1℃単位で返す		

3.6.1 UnitStatus(ItemNo=0)

IO-LGW の状態を取得します。

- 読み出し(CMD0[7]='0')

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	Rsv0	CH	Rsv1	INUM	Rsv2	Rsv3	CHK
0x40	0x00	0x00	0x00	0x00	0x00	0x00	0x40

CH : 0x00(ポート指定無し)

INUM : ItemNo. = 0

Rsvは無視されますが 0x00 を使用することを推奨します。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
0x40	rLEN	STAT	rData0	rData1	Rsv0	Rsv1	CHK

rLEN : 返答するデータ長(0x02)

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
 [7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

Rsv : 0x00 が返答されます

rData0・rData1にUnitStatusの16bitデータが返ります。

	b7	b6	b5	b4	b3	b2	b1	b0
rData0	INITerr	TEMPerr	0	CUwarn	0			
rData1	0		MEMerr	0		MAPerr	0	

3.6.2 UnitDiag(ItemNo=1)

IOLGW のエラー・イベント情報を取得します。

- 読み出し(CMD0[7]='0')

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	Rsv0	CH	Rsv1	INUM	Rsv2	Rsv3	CHK
0x40	0x00	0x00	0x00	0x01	0x00	0x00	0x40

CH : 書き込み対象のポート (ポート番号を数値で指定)
 INUM : ItemNo. = 1

CH はポート番号を数値で指定します。このため、複数のポートに対して同時に処理することはできません。

Rsv は無視されますが 0x00 を使用することを推奨します。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
0x40	rLEN	STAT	rData0	rData1	rData2	rData3	CHK

rLEN : 返答するデータ長(0x04)

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
 [7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

rData0~rData3 に UnitDiag の 32bit データが返ります。

	b7	b6	b5	b4	b3	b2	b1	b0
rData0	0						ErrType	
rData1	Mode		Type		Source	Instance		
rData2	AddErrCode							
rData3	ErrCode							

一度に返答されるエラー・イベントは一つとは限らず、複数の情報(4×nバイト)をまとめて返送されることがあります。

3.6.3 DeviceDiag(ItemNo=2)

IO-Link デバイスのエラー・イベント情報を取得します。

- 読み出し(CMD0[7]='0')

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	Rsv0	CH	Rsv1	INUM	Rsv2	Rsv3	CHK
0x40	0x00	0x00	0x00	0x02	0x00	0x00	0x43

CH : 書き込み対象のポート (ポート番号を数値で指定)
 INUM : ItemNo. = 2

CH はポート番号を数値で指定します。このため、複数のポートに対して同時に処理することはできません。

Rsv は無視されますが 0x00 を使用することを推奨します。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
0x40	rLEN	STAT	rData0	rData1	rData2	rData3	CHK

rLEN : 返答するデータ長(0x04)

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
 [7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

rData0~rData3 に DeviceDiag の 32bit データが返ります。

	b7	b6	b5	b4	b3	b2	b1	b0
rData0	0						ErrType	
rData1	Mode		Type		Source	Instance		
rData2	AddErrCode							
rData3	ErrCode							

一度に返答されるエラー・イベントは一つとは限らず、複数の情報(4×nバイト)をまとめて返送されることがあります。

3.6.4 DiagPort(ItemNo=3)

各ポートの状態を取得します。

- 読み出し(CMD0[7]='0')

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	Rsv0	CH	Rsv1	INUM	Rsv2	Rsv3	CHK
0x40	0x00	0x00	0x00	0x03	0x00	0x00	0x43

CH : 0x00(ポート指定無し)

INUM : ItemNo. = 3

Rsvは無視されますが 0x00 を使用することを推奨します。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
0x40	rLEN	STAT	rData0	rData1	rData2	0x00	CHK

rLEN : 返答するデータ長(0x04)

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
 [7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

rData0 にエラー発生ポート、rData1 にイベント発生ポート、rData2 に過電流発生ポートのデータ(ビット割り当て)が返ります。

	b7	b6	b5	b4	b3	b2	b1	b0
rData0(ErrCH)	CH7	CH6	CH5	CH4	CH3	CH2	CH1	CH0
rData1(EventCH)	CH7	CH6	CH5	CH4	CH3	CH2	CH1	CH0
rData2(OverCurCH)	CH7	CH6	CH5	CH4	CH3	CH2	CH1	CH0

3.6.5 COMStatus(ItemNo=4)

IO-Link の通信状態を取得します。

- 読み出し(CMD0[7]='0')

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	Rsv0	CH	Rsv1	INUM	Rsv2	Rsv3	CHK
0x40	0x00	0x00	0x00	0x04	0x00	0x00	0x44

CH : 0x00(ポート指定無し)

INUM : ItemNo. = 4

Rsvは無視されますが 0x00 を使用することを推奨します。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	RSP07
CMD	rLEN	STAT	rData0	rData1	rData2	rData3	rData4
0x40	0x10	0x00	CH0	CH1	CH2	CH3	CH4
RSP08	RSP09	RSP0A					CHK
rData5	rData6	rData7					CHK
CH5	CH6	CH7	0x00	0x00	0x00	0x00	CHK

rLEN : 返答するデータ長(0x08)

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
 [7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

rData0~rData7 に各ポートのステータスデータが返ります。

	b7	b6	b5	b4	b3	b2	b1	b0
rData	infoRdy	0				Revision	Status	

3.6.6 COMError(ItemNo=5)

IO-Link 通信のエラー回数を取得します。

- 読み出し(CMD0[7]='0')

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	Rsv0	CH	Rsv1	INUM	Rsv2	Rsv3	CHK
0x40	0x00	0x00	0x00	0x05	0x00	0x00	0x45

CH : 0x00(ポート指定無し)

INUM : ItemNo. = 5

Rsvは無視されますが 0x00 を使用することを推奨します。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	RSP07
CMD	rLEN	STAT	rData0	rData1	rData2	rData3	rData4
0x40	0x10	0x00	CH0	CH1	CH2	CH3	CH4
RSP08	RSP09	RSP0A					CHK
rData5	rData6	rData7					CHK
CH5	CH6	CH7	0x00	0x00	0x00	0x00	CHK

rLEN : 返答するデータ長(0x08)

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
 [7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

rData0～rData7 に各ポートの通信エラー回数が返ります。
 読み出しを実行すると、値はクリアされます。

3.6.7 FPGATemp(ItemNo=6)

使用している FPGA の内部温度を取得します。

- 読み出し(CMD0[7]='0')

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	Rsv0	CH	Rsv1	INUM	Rsv2	Rsv3	CHK
0x40	0x00	0x00	0x00	0x06	0x00	0x00	0x46

CH : 0x00(ポート指定無し)

INUM : ItemNo. = 6

Rsvは無視されますが 0x00 を使用することを推奨します。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
CMD	rLEN	STAT	rData0	rData1	Rsv0	Rsv1	CHK

rLEN : 返答するデータ長(0x08)

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
 [7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

rData0(LSB)・rData1(MSB)に 0.1℃単位の温度データが返ります。

rData0=0x59, rData1=0x01 の場合、0x0159=345 となり、これは 34.5℃を意味します。

3.7 Information(0x41)

IOLGW の情報を取得するコマンドです。
全て読み出しのみをサポートしています。

表 3-8 Information コマンドの Item 番号

ItemNo	名称	データ長	
0	VenderName	32	ベンダー名(“STEPTECHNICA CO., LTD. ”)
1	ProductName	32	製品名(“IO-Link Master Board ”)
2	ProductType	16	製品型式(“ST-IOLM ”)
3	SerialNo	16	シリアル番号
4	HardRev	4	ハードウェアバージョン
5	FirmRev	4	ファームウェアバージョン

3.7.1 VenderName(ItemNo=0)

ベンダー名を取得します。

- 読み出し(CMD0[7]='0')

Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	cLEN	Rsv0	Rsv1	INUM	Rsv2	Rsv3	CHK
0x41	0x01	0x00	0x00	0x00	0x00	0x00	0x40

cLEN : 書き込みアイテム数

0x01=VenderName のみ要求(32Byte)

0x02=VenderName と ProductName を要求(64Byte)

0x03=VenderName から ProductType まで要求(80Byte)

0x04=VenderName から SerialNo まで要求(96Byte)

0x05=VenderName から HardRev まで要求(100Byte)

0x06=VenderName から FirmRev まで要求(104Byte)

INUM : ItemNo. = 0

cLEN にアイテム数を入力することで連続読み込みが可能となっています。

Rsv は無視されますが 0x00 を使用することを推奨します。

Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	RSP07
CMD	rLEN	STAT	rData0	rData1	rData2	rData3	rData4
0x41	0x20	0x00	0x53(S)	0x54(T)	0x45(E)	0x50(P)	0x54(T)
RSP08	RSP09	RSP0A	RSP0B	RSP0C	RSP0D	RSP0E	RSP0F
rData5	rData6	rData7	rData8	rData9	rData10	rData11	rData12
0x45(E)	0x43(C)	0x48(H)	0x4E(N)	0x49(I)	0x43(C)	0x41(A)	0x20()
RSP10	RSP11	RSP12	RSP13	RSP14	RSP15	RSP16	RSP17
rData13	rData14	rData15	rData16	rData17	rData18	rData19	rData20
0x43(C)	0x4F(O)	0x2E(.)	0x2C(,)	0x20()	0x4C(L)	0x54(T)	0x44(D)
RSP18	RSP19	RSP1A	RSP1B	RSP1C	RSP1D	RSP1E	RSP1F
rData21	rData22	rData23	rData24	rData25	rData26	rData27	rData28
0x2E(.)	0x20()						
RSP20	RSP21	RSP22					CHK
rData29	rData30	rData31					CHK
0x20()	0x20()	0x20()	0x00	0x00	0x00	0x00	0x10

rLEN : 返答するデータ長

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)

[7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

rData : 返答データ(“STEPTECHNICA CO., LTD. ”)

rData に 32 文字分の文字が返答されます。

3.7.2 ProductName(ItemNo=1)

ユニット製品名を取得する。

- 読み出し(CMD0[7]='0')

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	cLEN	Rsv0	Rsv1	INUM	Rsv2	Rsv3	CHK
0x41	0x01	0x00	0x00	0x01	0x00	0x00	0x41

cLEN : 書き込みアイテム数

- 0x01=ProductNameのみ要求(32Byte)
- 0x02=ProductNameとProductTypeを要求(48Byte)
- 0x03=ProductNameからSerialNoまで要求(64Byte)
- 0x04=ProductNameからHardRevまで要求(68Byte)
- 0x05=ProductNameからFirmRevまで要求(72Byte)

INUM : ItemNo. = 1

cLENにアイテム数を入力することで連続読み込みが可能となっています。
 Rsvは無視されますが0x00を使用することを推奨します。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	RSP07
CMD	rLEN	STAT	rData0	rData1	rData2	rData3	rData4
0x41	0x20	0x00	0x49(I)	0x4F(0)	0x2D(-)	0x4C(L)	0x69(i)
RSP08	RSP09	RSP0A	RSP0B	RSP0C	RSP0D	RSP0E	RSP0F
rData5	rData6	rData7	rData8	rData9	rData10	rData11	rData12
0x6E(n)	0x6B(k)	0x20()	0x4D(M)	0x61(a)	0x73(s)	0x74(t)	0x65(e)
RSP10	RSP11	RSP12	RSP13	RSP14	RSP15	RSP16	RSP17
rData13	rData14	rData15	rData16	rData17	rData18	rData19	rData20
0x72(r)	0x20()	0x42(B)	0x6F(o)	0x61(a)	0x72(r)	0x64(d)	0x20()
RSP18	RSP19	RSP1A	RSP1B	RSP1C	RSP1D	RSP1E	RSP1F
rData21	rData22	rData23	rData24	rData25	rData26	rData27	rData28
0x20()							
RSP20	RSP21	RSP22					CHK
rData29	rData30	rData31					CHK
0x20()	0x20()	0x20()	0x00	0x00	0x00	0x00	0x0C

rLEN : 返答するデータ長

STAT : 応答結果 [1:0]=Status: '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
 [7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

rData : 返答データ(“IO-Link Master Board ”)

rDataに32文字分の文字が返答されます。

3.7.3 ProductType(ItemNo=2)

製品タイプを取得する。

- 読み出し(CMD0[7]='0')

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	cLEN	Rsv0	Rsv1	INUM	Rsv2	Rsv3	CHK
0x41	0x01	0x00	0x00	0x02	0x00	0x00	0x42

cLEN : 書き込みアイテム数

0x01=ProductType のみ要求(16Byte)

0x02=ProductType と SerialNo を要求(32Byte)

0x03=ProductType から HardRev まで要求(36Byte)

0x04=ProductType から FirmRev まで要求(40Byte)

INUM : ItemNo. = 2

cLEN にアイテム数を入力することで連続読み込みが可能となっています。

Rsv は無視されますが 0x00 を使用することを推奨します。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	RSP07
CMD	rLEN	STAT	rData0	rData1	rData2	rData3	rData4
0x41	0x10	0x00	0x53(S)	0x54(T)	0x2D(-)	0x49(I)	0x4F(O)
RSP08	RSP09	RSP0A	RSP0B	RSP0C	RSP0D	RSP0E	RSP0F
rData5	rData6	rData7	rData8	rData9	rData10	rData11	rData12
0x4C(L)	0x4D(M)	0x20()					
RSP10	RSP11	RSP12					CHK
rData13	rData14	rData15					CHK
0x20()	0x20()	0x20()	0x00	0x00	0x00	0x00	0x5C

rLEN : 返答するデータ長

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
 [7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

rData : 返答データ(“ST-IOLM ”)

rData に 16 文字分の文字が返答されます。

3.7.4 SerialNo(ItemNo=3)

ユニットのシリアル番号を取得します。

- 読み出し(CMD0[7]='0')

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	cLEN	Rsv0	Rsv1	INUM	Rsv2	Rsv3	CHK
0x41	0x01	0x00	0x00	0x03	0x00	0x00	0x43

cLEN : 書き込みアイテム数

0x01=SerialNo のみ要求(16Byte)

0x02=SerialNo と HardRev を要求(20Byte)

0x03=SerialNo から FirmRev まで要求(24Byte)

INUM : ItemNo. = 3

cLEN にアイテム数を入力することで連続読み込みが可能となっています。
 Rsv は無視されますが 0x00 を使用することを推奨します。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	RSP07
CMD	rLEN	STAT	rData0	rData1	rData2	rData3	rData4
0x41	0x10	0x00	0x32(2)	0x30(0)	0x32(2)	0x31(1)	0x30(0)
RSP08	RSP09	RSP0A	RSP0B	RSP0C	RSP0D	RSP0E	RSP0F
rData5	rData6	rData7	rData8	rData9	rData10	rData11	rData12
0x30(0)							
RSP10	RSP11	RSP12					CHK
rData13	rData14	rData15					CHK
0x30(0)	0x30(0)	0x31(1)	0x00	0x00	0x00	0x00	0x51

rLEN : 返答するデータ長

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
 [7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

rData : 返答データ

rData に 16 文字分の文字が返答されます。

3.7.5 HardRev(ItemNo=4)

ハードウェア(FPGA)のバージョン情報を取得します。

- 読み出し(CMD0[7]='0')

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	cLEN	Rsv0	Rsv1	INUM	Rsv2	Rsv3	CHK
0x41	0x01	0x00	0x00	0x04	0x00	0x00	0x44

cLEN : 書き込みアイテム数

0x01=HardRev のみ要求(16Byte)

0x02=HardRev と FirmRev を要求(20Byte)

INUM : ItemNo. = 4

cLEN にアイテム数を入力することで連続読み込みが可能となっています。

Rsv は無視されますが 0x00 を使用することを推奨します。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
CMD	rLEN	STAT	rData0	rData1	rData2	rData3	CHK
0x41	0x04	0x00	0x30(0)	0x30(0)	0x31(1)	0x36(6)	0x42

rLEN : 返答するデータ長

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
 [7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

rData : 返答データ

rData に 4 文字分の文字が返答されます。

3.7.6 FirmRev(ItemNo=5)

ファームウェア(CPU プログラム)のバージョン情報を取得します。

- 読み出し(CMD0[7]='0')

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	cLEN	Rsv0	Rsv1	INUM	Rsv2	Rsv3	CHK
0x41	0x01	0x00	0x00	0x05	0x00	0x00	0x45

cLEN : 書き込みアイテム数
 0x01=FirmRev のみ要求(16Byte)
 INUM : ItemNo. = 5

cLEN で指定できるのは 0x01 のみです。
 Rsv は無視されますが 0x00 を使用することを推奨します。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
CMD	rLEN	STAT	rData0	rData1	rData2	rData3	CHK
0x41	0x04	0x00	0x36(6)	0x30(0)	0x31(1)	0x30(0)	0x42

rLEN : 返答するデータ長
 STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
 [7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)
 rData : 返答データ

rData に 4 文字分の文字が返答されます。

3.8 NetworkStatus(0x42)

CUnet の通信上で発生した CARE 信号の発生数を取得するコマンドです。
 読み出しのみをサポートしています。
 カウント数の読み出しを実行すると自動的にクリアされます。

- 読み出し (CMD0[7]='0')

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	Rsv0	Rsv1	Rsv2	INUM	Rsv3	Rsv4	CHK
0x42	0x00	0x00	0x00	0x01	0x00	0x00	0x43

INUM : ItemNo. = 1

Rsv は無視されますが 0x00 を使用することを推奨します。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
CMD	rLEN	STAT	rData0	rData1	Rsv0	Rsv1	CHK
0x42	0x02	0x00	LCare	MCare	0x00	0x00	CHK

rLEN : 返答するデータ長

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
 [7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

rData : 返答データ

Rsv : 0x00 が返答されます

rData0 に LCare 発生回数、rData1 に MCare 発生回数が返る。

	b7	b6	b5	b4	b3	b2	b1	b0
rData0	LCare 回数							
rData1	MCare 回数							

発生回数は読み出しを実行した時にクリアされます。

3.9 CUnet(0x43)

メイン・ステーションからサブ・ステーションの品種を問い合わせるコマンドです。
 書き込むことはなく、読み出しだけに対応しています。

- 読み出し(CMD0[7]='0')
- この IOLGW の品種情報・動作状態を取得します。

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CMD07
'C'	'U'	'n'	'e'	't'	SP	'?'	'¥r'
0x43	0x55	0x6E	0x65	0x74	0x20	0x3F	0x0D

“CUnet? ¥r” = 問合せコマンド

このコマンドは MKY44 シリーズと共通仕様とするため、**チェックサムを付加しません。**

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	RSP07
'C'	'U'	'I'	'O'	'L'	'G'	'W'	SP
0x43	0x55	0x49	0x4F	0x4C	0x47	0x57	0x20
RSP08	RSP09	RSP0A	RSP0B	RSP0C	RSP0D	RSP0E	RSP0F
TYPE	HR0	HR1	HR2	HR3	FR0	FR1	FR2
RSP10	RSP11	RSP12	RSP13	RSP14	RSP15	RSP16	CHK
FR3	SA	OWN	DOSA	DOSZ	ERR	EVT	CHK

“CUIOLGW” = この IOLGW の型式名

- TYPE : ユニットタイプ (0x08=8 ポート版)
- HR[3:0] : FPGA バージョン(文字列4文字)
- FR[3:0] : ファームウェアバージョン(文字列4文字)
- SA : DIPSW で設定された入力データを格納する StationAddress
- OWN : ProcessInData のサイズから決まった OWN 値
- DOSA : DIPSW で設定された出力データを参照する StationAddress
- DOSZ : ProcessOutData のサイズから決まった領域サイズ値
- ERR : ユニット GM データの 3byte 目にある Error 情報と同じもの
- EVT : ユニット GM データの 4byte 目にある Event 情報と同じもの

注意：

電源投入後、このコマンドを送るまではグローバルメモリーの共有は開始されません。
 このコマンドを最初に実行してください。

3.10 DeviceInformation(0x44)

IO-Link デバイスの情報を取得するコマンドです。
 全て読み出しのみをサポートしています。

表 3-9 DeviceInfomation コマンドの Item 番号

ItemNo	名称	データ長	
0	DProcInByte	1	ProcessInData サイズ
1	DProcOutByte	1	ProcessOutData サイズ
2	VenderID	2	ベンダーID
3	DeviceID	4	デバイス ID
4	MinCycle	1	最小サイクルタイム b5-0 : Multiplier b7-6 : TimeBase 00=0.1msec → TimeBase×Multiplier 01=0.4msec → 6.4msec+TimeBase×Multiplier 10=1.6msec → 32.0msec+TimeBase×Multiplier 11=reserved
5	SeqType	1	シーケンスタイプ 0x00=Type0 0x01=Type1_x 0x02=Type2_x
6	SerialNo	16	シリアル番号

IO-Link 通信でステータスが Startup の時、これらの情報が読み出せると preOperate 状態に遷移します。
 この時 CMD=0x40 の ItemNo=4 にある infoRdy がセットされます。すなわち infoRdy が '1' の時は、デバイスから有効な値を読み込んでいることを示しています。

3.10.1DProcInByte(ItemNo=0)

デバイスの ProcessIn のデータサイズを取得します。

- 読み出し(CMD0[7]='0')

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	cLEN	CH	Rsv0	INUM	Rsv1	Rsv2	CHK
0x44	0x00	0x00	0x00	0x00	0x00	0x00	0x44

cLEN : 読み出すアイテム数

0x01=DProcInByte のみ要求(1Byte)

0x02=DProcInByte と DProcOutByte 要求(2Byte)

0x03=DProcInByte から VenderID まで要求(4Byte)

0x04=DProcInByte から DeviceID まで要求(8Byte)

0x05=DProcInByte から MinCycle まで要求(9Byte)

0x06=DProcInByte から SeqType まで要求(10Byte)

0x07=DProcInByte から SerialNo まで要求(26Byte)

CH : 読み出す対象のポート (ポート番号をビットで指定)

INUM : ItemNo. = 0

cLEN にアイテム数を入力することで連続読み込みが可能となっています。

CH で複数のビットを設定するとそのポートの値を順に応答します。ただし、この時は複数アイテムを読み出すことはできません。DProcInByte のみ複数ポート分読み出すことができます。

Rsv は無視されますが 0x00 を使用することを推奨します。

Command 送信時の INUM を 0x80 に設定して全ポートを読み出すことは対応していません。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
0x44	rLEN	STAT	rData0	rData1	rData2	rData3	CHK

rLEN : 返答するデータ長(0x02)

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
 [7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

rData : 返答データ

rData0 からデータが返ります。

3.10.2 DProcOutByte(ItemNo=1)

デバイスの ProcessOut のデータサイズを取得します。

- 読み出し(CMD0[7]='0')

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	cLEN	CH	Rsv0	INUM	Rsv1	Rsv2	CHK
0x44	0x00	0x00	0x00	0x01	0x00	0x00	0x45

cLEN : 読み出すアイテム数

0x01=DProcOutByte のみ要求(1Byte)

0x02=DProcOutByte と VenderID 要求(3Byte)

0x03=DProcOutByte から DeviceID まで要求(7Byte)

0x04=DProcOutByte から MinCycle まで要求(8Byte)

0x05=DProcOutByte から SeqType まで要求(9Byte)

0x06=DProcOutByte から SerialNo まで要求(25Byte)

CH : 読み出す対象のポート (ポート番号をビットで指定)

INUM : ItemNo. = 1

cLEN にアイテム数を入力することで連続読み込みが可能となっています。

CH で複数のビットを設定するとそのポートの値を順に応答します。ただし、この時は複数アイテムを読み出すことはできません。DProcOutByte のみ複数ポート分読み出すことができます。Rsv は無視されますが 0x00 を使用することを推奨します。

Command 送信時の INUM を 0x80 に設定して全ポートを読み出すことは対応していません。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
0x44	rLEN	STAT	rData0	rData1	rData2	rData3	CHK

rLEN : 返答するデータ長(0x02)

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
 [7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

rData : 返答データ

rData0 からデータが返ります。

3.10.3 VenderID(ItemNo=2)

デバイスの VenderID を取得します。

- 読み出し(CMD0[7]='0')

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	cLEN	CH	Rsv0	INUM	Rsv1	Rsv2	CHK
0x44	0x00	0x00	0x00	0x02	0x00	0x00	0x46

cLEN : 読み出すアイテム数

0x01=VenderID のみ要求(2Byte)

0x02=VenderID と DeviceID 要求(6Byte)

0x03=VenderID から MinCycle まで要求(7Byte)

0x04=VenderID から SeqType まで要求(8Byte)

0x05=VenderID から SerialNo まで要求(24Byte)

CH : 読み出す対象のポート (ポート番号をビットで指定)

INUM : ItemNo. = 2

cLEN にアイテム数を入力することで連続読み込みが可能となっています。

CH で複数のビットを設定するとそのポートの値を順に応答します。ただし、この時は複数アイテムを読み出すことはできません。VenderID のみ複数ポート分読み出すことができます。

Rsv は無視されますが 0x00 を使用することを推奨します。

Command 送信時の INUM を 0x80 に設定して全ポートを読み出すことは対応していません。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
0x44	rLEN	STAT	rData0	rData1	rData2	rData3	CHK

rLEN : 返答するデータ長(0x02)

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
 [7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

rData : 返答データ

rData0・rData1 に 16bit 分のデータが返ります。

3.10.4 DeviceID(ItemNo=3)

デバイスの DeviceID(+ RevisionID)を取得します。

- 読み出し(CMD0[7]='0')

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	cLEN	CH	Rsv0	INUM	Rsv1	Rsv2	CHK
0x44	0x00	0x00	0x00	0x03	0x00	0x00	0x47

cLEN : 読み出すアイテム数

0x01=DeviceIDのみ要求(4Byte)

0x02=DeviceIDと MinCycle 要求(5Byte)

0x03=DeviceID から SeqType まで要求(6Byte)

0x04=DeviceID から SerialNo まで要求(22Byte)

CH : 読み出す対象のポート (ポート番号をビットで指定)

INUM : ItemNo. = 3

cLEN にアイテム数を入力することで連続読み込みが可能となっています。

CH で複数のビットを設定するとそのポートの値を順に応答します。ただし、この時は複数アイテムを読み出すことはできません。DeviceIDのみ複数ポート分読み出すことができます。

Rsvは無視されますが0x00を使用することを推奨します。

Command 送信時の INUM を 0x80 に設定して全ポートを読み出すことは対応していません。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
0x44	rLEN	STAT	rData0	rData1	rData2	rData3	CHK

rLEN : 返答するデータ長(0x02)

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
 [7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

rData : 返答データ

rData0~rData2 に DeviceID、rData3 に RevisionID が返答されます。

rData0	rData1	rData2	rData3
DeviceID[7:0]	DeviceID[15:8]	DeviceID[24:16]	RevisionID

3.10.5 MinCycle(ItemNo=4)

デバイスの最小サイクルタイムを取得します。

- 読み出し(CMD0[7]='0')

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	cLEN	CH	Rsv0	INUM	Rsv1	Rsv2	CHK
0x44	0x00	0x00	0x00	0x04	0x00	0x00	0x40

cLEN : 読み出すアイテム数

0x01=MinCycleのみ要求(1Byte)

0x02=MinCycleとSeqType要求(2Byte)

0x03=MinCycleからSerialNoまで要求(18Byte)

CH : 読み出す対象のポート (ポート番号をビットで指定)

INUM : ItemNo. = 4

cLENにアイテム数を入力することで連続読み込みが可能となっています。

CHで複数のビットを設定するとそのポートの値を順に応答します。ただし、この時は複数アイテムを読み出すことはできません。MinCycleのみ複数ポート分読み出すことができます。

Rsvは無視されますが0x00を使用することを推奨します。

Command送信時のINUMを0x80に設定して全ポートを読み出すことは対応していません。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
0x44	rLEN	STAT	rData0	rData1	rData2	rData3	CHK

rLEN : 返答するデータ長(0x02)

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
 [7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

rData : 返答データ

rData0からデータが返ります。

3.10.6 SeqType(ItemNo=5)

デバイスのシーケンスタイプを取得します。

- 読み出し(CMD0[7]='0')

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	cLEN	CH	Rsv0	INUM	Rsv1	Rsv2	CHK
0x44	0x00	0x00	0x00	0x05	0x00	0x00	0x41

cLEN : 読み出すアイテム数

0x01=SeqType のみ要求(1Byte)

0x02=SeqType と SerialNo 要求(17Byte)

CH : 読み出す対象のポート (ポート番号をビットで指定)

INUM : ItemNo. = 5

cLEN にアイテム数を入力することで連続読み込みが可能となっています。

CH で複数のビットを設定するとそのポートの値を順に応答します。ただし、この時は複数アイテムを読み出すことはできません。SeqType のみ複数ポート分読み出すことができます。

Rsv は無視されますが 0x00 を使用することを推奨します。

Command 送信時の INUM を 0x80 に設定して全ポートを読み出すことは対応していません。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
0x44	rLEN	STAT	rData0	rData1	rData2	rData3	CHK

rLEN : 返答するデータ長(0x02)

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
 [7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

rData : 返答データ

rData0 からデータが返ります。

3.10.7 SerialNo(ItemNo=6)

デバイスのシリアル番号を取得します。

- 読み出し(CMD0[7]='0')

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	cLEN	CH	Rsv0	INUM	Rsv1	Rsv2	CHK
0x44	0x00	0x00	0x00	0x06	0x00	0x00	0x42

cLEN : 読み出すアイテム数

0x01=SerialNoのみ要求(1Byte)

CH : 読み出す対象のポート (ポート番号をビットで指定)

INUM : ItemNo. = 6

cLEN で指定できるのは 0x01 のみです。

CH で複数のビットを設定するとそのポートの値を順に応答します。

Rsv は無視されますが 0x00 を使用することを推奨します。

Command 送信時の INUM を 0x80 に設定して全ポートを読み出すことは対応していません。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	RSP07
0x44	rLEN	STAT	rData0	rData1	rData2	rData3	rData4
RSP08	RSP09	RSP0A	RSP0B	RSP0C	RSP0D	RSP0E	RSP0F
rData5	rData6	rData7	rData8	rData9	rDataA	rDataB	rDataC
RSP10	RSP11	RSP12					CHK
rDataD	rDataE	rDataF	0x00	0x00	0x00	0x00	CHK

rLEN : 返答するデータ長

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
 [7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

rData : 返答データ

rData0 から 16 文字分のデータが返ります。

3.11 ISDU 通信(0x5n)

IO-Link デバイスに直接アクセスするコマンドです。

IODD ファイルに定義されている Index・SubIndex を使い、デバイスのパラメーターにアクセスすることが可能です。

- 書き込み(CMD0[7]='1')

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	IND-L	IND-H	SubIND	LEN	wData0	wData1	CHK
0xDn							

CMD : 下位 4bit で書き込み対象のポート番号を数値で指定する

IND-L : Index データの下位 8bit を指定

IND-H : Index データの上位 8bit を指定

SubIND : SubIndex データを指定

LEN : データ長 (バイト数)

wData : 書き込みデータ

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
0xDn	STAT	ERR0	※	※	※	※	CHK

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
[7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

ERR0 : エラーコード

※ : RSP03 以降の内容は、ステータスによって意味が異なります。(次項に記述)

- 読み出し(CMD0[7]='0')

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	IND-L	IND-H	SubIND	LEN	Rsv0	Rsv1	CHK
0x5n				0x00	0x00	0x00	

CMD : 下位 4bit で書き込み対象のポート番号を数値で指定する

IND-L : Index データの下位 8bit を指定

IND-H : Index データの上位 8bit を指定

SubIND : SubIndex データを指定

LEN : データ長 (0x00)

Rsv は無視されますが 0x00 を使用することを推奨します。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
0x5n	STAT	ERR0	※	※	※	※	CHK

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
[7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

ERR0 : エラーコード

※ : RSP03 以降の内容は、ステータスによって意味が異なります。(次項に記述)

• Response 内容

• Status=0 (成功時)

RSP02	ERR0	0x00
RSP03	ERR1	0x00
RSP04	rLen	返答するデータ長
RSP05	rData0	返答データ

• Status=1 (エラー時)

RSP02	ERR0	エラーコード
RSP03	ERR1	追加エラーコード
RSP04	Index(Lo)	エラーとなったインデックス値下位
RSP05	Index(Hi)	エラーとなったインデックス値上位
RSP06	SubIndex	エラーとなったサブインデックス値

• Status=2 (未対応時)

RSP02	ERR0	エラーコード 0x01=デジタル入出力モードになっている 0x02=Operate 状態になっていない 0x04=reserved 0x08=他のポートが ISDU コマンド処理中
RSP03	0x00	
RSP04	0x00	
RSP05	0x00	
RSP06	0x00	

3.12 ProcessData(0x60)

デバイスのプロセス入力データを取得するコマンドです。
全て読み出しのみをサポートしています。

表 3-10 ProcessDataSize コマンドの Item 番号

ItemNo	名称	
0	ProcInCh	ポート単位でのデータ取得
1	ProcInALL	全ポートの一斉取得

3.12.1 ProcInCh(ItemNo=0)

指定ポートのプロセスデータを取得します。

- 読み出し(CMD0[7]='0')

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	Rsv0	CH	Rsv1	INUM	Rsv2	Rsv3	CHK
0x60	0x00	0x00	0x00	0x00	0x00	0x00	0x60

CH : 読み出す対象のポート (ポート番号を数値で指定)

INUM : ItemNo. = 0

CH はポート番号を数値で指定します。このため、複数のポートに対して同時に処理することはできません。

Rsvは無視されますが0x00を使用することを推奨します。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
0x60	rLEN	STAT	rData0	rData1	rData2	rData3	CHK

rLEN : 返答するデータ長

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
[7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

rData : 返答データ

rData0 からデータが返る。

rData0	rData1	rData2	rDataN
CH NUM	LEN	LSB	MSB

rData0 : ポート番号

rData1 : データサイズ

rData2~ : 読み出しデータ

3.12.2 ProcInALL(ItemNo=1)

全てのポートのプロセスデータを取得します。

- 読み出し(CMD0[7]='0')

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	Rsv0	CH	Rsv1	INUM	Rsv2	Rsv3	CHK
0x60	0x00	0x00	0x00	0x01	0x00	0x00	0x61

CH : 0x00

INUM : ItemNo. = 1

Rsvは無視されますが 0x00 を使用することを推奨します。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
0x60	rLEN	STAT	rData0	rData1	rData2	rData3	CHK

rLEN : 返答するデータ長

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
 [7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

rData : 返答データ

rData0 からデータが返ります。

1ポートのデータ構成は ProcInCh と同じです。

rData0	rData1	rData2	rDataN
CH NUM	LEN	LSB	MSB

グローバルメモリー領域と同じように、IO-Link モードでないポートは次のポートが詰めて返答されます。

3.13 ConfigReset(0x70)

IOLGW およびポートの初期化を行うことができます。

表 3-11 ConfigReset コマンドの Item 番号

ItemNo	名称	
0	UnitReset	IOLGW の初期化
1	ChReset	ポートの初期化
2	CUReset	CUnet 通信の再起動

3.13.1 UnitReset(ItemNo=0)

IOLGW 全体の初期化を指示するものです。
 このためコマンドは書き込みだけが対応しています。
 全ての設定がデフォルトに戻るので注意が必要です。

- 書き込み(CMD0[7]=' 1')

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	Rsv0	CH	Rsv1	INUM	EXE	Rsv2	CHK
0xF0	0x00	0x00	0x00	0x00	0x01	0x00	0xF1

CH : 0x00
 INUM : ItemNo. = 0
 EXE : 書き込みデータ
 0x01= 初期化指示

EXE に対応する値を書き込むことで IOLGW が初期化されます。
 Rsv は無視されますが 0x00 を使用することを推奨します。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
0xF0	rLEN	STAT	Rsv0	Rsv1	Rsv2	Rsv3	CHK

rLEN : 返答するデータ長
 STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
 [7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)
 Rsv : 0x00 が返答されます

3.13.2 ChReset(ItemNo=1)

ポート単位での初期化を指示するものです。IO-Link モードであれば STARTUP フェーズから再始動することになります。

このためコマンドは書き込みだけが対応しています。

- 書き込み(CMD0[7]='1')

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	Rsv0	CH	Rsv1	INUM	EXE	Rsv2	CHK
0xF0	0x00	0x00	0x00	0x01	0x01	0x00	0xF0

CH : 初期化する対象のポート (ポート番号をビットで指定)

INUM : ItemNo. = 1

EXE : 書き込みデータ

0x01= 初期化指示

CH で複数のビットを設定するとそのポートに対し一括で処理を指示できます。

EXE に対応する値を書き込むことで指定された IO-Link ポートが初期化されます。

Rsv は無視されますが 0x00 を使用することを推奨します。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
0xF0	rLEN	STAT	Rsv0	Rsv1	Rsv2	Rsv3	CHK

rLEN : 返答するデータ長

STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
 [7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)

Rsv : 0x00 が返答されます

3.13.3 CUnetReset(ItemNo=2)

CUnet 通信を一旦切断し、新たな設定で再接続するためのものです。

ProcessData サイズを変更したり、NoDataOut モードの設定を変更した場合、パラメーターは変更されますが、実際の CUnet ステーションとしての変更はされません。このため、電源のオフ・オンをせずに再参入する場合に使用します。

このためコマンドは書き込みだけが対応しています。

- 書き込み(CMD0[7]='1')

➤ Command

CMD00	CMD01	CMD02	CMD03	CMD04	CMD05	CMD06	CHK
CMD	Rsv0	CH	Rsv1	INUM	EXE	Rsv2	CHK
0xF0	0x00	0x00	0x00	0x02	0x01	0x00	0xF3

CH : 0x00
 INUM : ItemNo. = 2
 EXE : 書き込みデータ
 0x01= 初期化指示

Rsv は無視されますが 0x00 を使用することを推奨します。

➤ Response

RSP00	RSP01	RSP02	RSP03	RSP04	RSP05	RSP06	CHK
0xF0	rLEN	STAT	Rsv0	Rsv1	Rsv2	Rsv3	CHK

rLEN : 返答するデータ長
 STAT : 応答結果 [1:0]=Status : '0'=成功、'1'=エラー、'2'=未対応 ('3'=未使用)
 [7:4]=Error (4.2.1 メール通信ステータスエラーコード参照)
 Rsv : 0x00 が返答される

再参入した後でレスポンスが返答されます。

この後 0x43 コマンド(CUnet?)を用い、変更後のパラメーターを確認するようにしてください。

4. 付録

4.1 メール通信コマンド一覧

	CMD	ITEM#	LEN	bit	R?W	Name	機能	範囲/データ量			
Port Config	0x20	#0	1	b2-0	R/W	ioMode	ポート動作モード	0~5			
				b4-3	R/W	Verify	デバイス照合機能	0~2			
				b6-5	R/W	Backup/Restore	自動バックアップ機能	0~3			
				b7	R/W	Sync	ポート間同期機能	0~1			
	#1	1	b1-0	R/W	OutputClear	異常時出力設定	0~2				
			b2	R/W	PwrOverCur	電源過電流検知機能	0~1				
			b3	R/W	OverCurrent	過電流検知機能	0~1				
			b4	R/W	Recovery	復帰時挙動設定	0~1				
			b7-5								
			#2	1	b2-0	R/W	InputFilter	入力フィルター設定	0~4		
					b3						
	b5-4	R/W			InputHold	入力保持時間設定	0~3				
	#3	1			R/W	CycleTime	ポートサイクルタイム	0,4~255			
0x21					#0	1	R/W	UProcInByte	ポート入力データ割付	0~32	
							R/W	UProcOutByte	ポート出力データ割付	0~32	
							W	AutoAlloc	ポート自動割付け	1	
Verify Config	0x22	#0	2	R/W	VendeID	照合用ベンダーID設定	2byte				
		#1	4	R/W	DeviceID	照合用デバイスID設定	4byte				
		#2	16	R/W	SerialNo	照合用シリアル番号設定	16character				
		#3	1	W	AutoConfig	照合データ自動設定	1~2				
Data Strage	0x23	#0	1		R/W	NoDataOut	DOSA領域未使用モード設定	0~1			
					0x30	#0	1	R	StrageInfo	ストレージ情報	
								W	StorageClear	ストレージデータ消去	1
								W	ManualBackup	手動バックアップ	1
W	ManualRestore	手動リストア	1								
Diagnosis	0x40	#0	2	R	UnitStatus	ユニットの状態	2byte				
		#1	4	R	UnitDiag	ポートの診断情報	4byte				
		#2	4	R	DeviceDiag	デバイスの診断情報	3byte				
		#3	3	R	DiagPort	発生ポート情報	3byte				
		#4	8	R	COMStatus	デバイス通信状態	8byte				
		#5	8	R	COMError	通信エラー回数	8byte				
		#6	2	R	FPGATemp	ユニット内部温度情報	2byte				
Unit Information	0x41	#0	32	R	VenderName	ユニットベンダー名	32character				
		#1	32	R	ProductName	ユニット製品名	32character				
		#2	16	R	ProductType	ユニット製品型式	16character				
		#3	16	R	SerialNo	ユニットシリアル番号	16character				
		#4	4	R	HardRev	ハードウェアリビジョン	4character				
		#5	4	R	FirmRev	ファームウェアリビジョン	4character				
	0x42	#0	2	R	CareInfo	CUNetCAREカウンタ値	2byte				
	0x43	-		R	CUNetInfo	CUNetユニット情報	固定長				
Device Information	0x44	#0	1	R	DProcInByte	デバイス入力データサイズ	1byte				
		#1	1	R	DProcOutByte	デバイス出力データサイズ	1byte				
		#2	2	R	VenderID	デバイスベンダーID	2byte				
		#3	4	R	DeviceID	デバイスデバイスID	4byte				
		#4	1	R	MinCycleTime	デバイス最小サイクルタイム	1byte				
		#5	1	R	SequenceType	デバイスシーケンスタイプ	1byte				
		#6	16	R	SerialNo	デバイスシリアル番号	16character				
	0x50	-		R/W	ISDU	ISDU通信コマンド					
Proc Data	0x60	#0	n	R	ProcInCh	個別プロセスデータ読み出し					
		#1	n	R	ProcInAll	一括プロセスデータ読み出し					
Reset	0x70	#0	1	W	UnitReset	ユニットリセット	1				
		#1	1	W	PortReset	ポートリセット	1				
		#2	1	W	CUReset	CUNetリセット	1				

4.2 エラーコード

4.2.1 メール通信ステータスエラーコード

Error Code	Name	意味
0x01	Invalid Parameter	不正なパラメーター
0x02	Out of range	パラメーターの範囲が不適切
0x03	Invalid Check Coce	シリアル番号書き込みの値が不正
0x04	reserved	
0x05	Invalid communiction	IO-Link通信エラー
0x06	Invalid Pdout Size	プロセス出力データサイズがゼロでない
0x07	No IO-Link Device	IO-Linkデバイスが接続されていない
0x08	Invalid Port Num	ポート番号が不適切
0x09	Invalid Mapping	プロセスデータのマッピング不適切
0x0A	Invalid Port	対象ポートが不適切
0x0B	Invalid Offset	オフセット指定値が不適切
0x0C	Invalid Item Number	アイテム番号が不適切
0x0D	Invalid Length	項目指定長が不適切
0x0E	Invalid Combination	アイテム番号と項目指定長の組み合わせが不適切
0x0F	Port No Service	対象ポートがIO-Linkモードでない

4.2.2 IO-Link マスターエラーコード

Error Code	意味
0x1000	ISDU非対応orIO-Linkモードでない
0x1100	ISDUタイムアウト
0x1800	接続されたデバイスのプロセスデータサイズが設定値より大きい
0x1801	LCARE/MCARE検出回数超過
0x1802	IO-Link通信なし
0x5600	ISDUチェックサムエラー
0x5700	ISDU応答不正
0x8011	ISDUアクセス禁止INDEXが指定された
0x8033	ISDUデータ長超過
0xFF22	IO-Linkデバイス通信異常
0xFF23	ID照合なしで有効ストレージデータが存在
0xFF24	ストレージデータ長超過
0xFF25	ストレージのアクセスが拒否された
0xFF91	ストレージデータのバックアップ要求された
0xFFFF3	照合用のデバイスが登録されていない
0xFFFF6	内部温度が高温になっている
0xFFFFB	IO-Linkデバイスが未接続
0xFFFFC	シリアル番号照合エラー
0xFFFFD	リストア用のバックアップデータが存在しない
0xFFFFE	接続されたデバイスのIDが異なる
0xFFFFF	リストア先のデバイスIDが異なる

➤ 改訂履歴

Rev.	発行日	改訂内容
1.00	2023/02/06	初版作成
1.01	2023/03/08	誤記訂正 ・ 3-13 : Command 例、CMD04 : 0x00→0x80 ・ 3-28 : 不要行削除 ・ 3-41 : Command 表、CHK : 0x45→0x46 ・ 3-59 : 説明文先頭に“※:” 追加
1.02	2023/12/21	誤記訂正 ・ 3-36 : 3.6.2 UnitDiag(ItemNo=1) ・ 3-37 : 3.6.3 DeviceDiag(ItemNo=2) Command 例 : CH : 0x01→0x00 CH 説明 : 「ビットで指定」 → 「数値で指定」 CH 説明コメント追加
1.03	2025/01/28	ドキュメント名称を「リファレンスマニュアル」へ変更 カタカナ表記を含め表現を変更 「メモリ」 → 「メモリー」や「スレーブ」を使わないように変更 「である」 → 「です」などの変更

ご注意

- 1) 本資料に記載された内容は、将来予告なしに変更する場合があります。本製品をご使用になる際には、本資料が最新の版であるかをご確認ください。
- 2) 本資料において記載されている説明や回路例などの技術情報は、お客様が用途に応じて本製品を適切にご利用いただくための参考資料です。
- 3) 実際に本製品をご使用になる際には、基板上における本製品の周辺回路条件や環境を考慮の上、お客様の責任においてシステム全体を十分に評価し、お客様の目的に適合するようシステムを設計してください。当社は、お客様のシステムと本製品との適合可否に対する責任を負いません。
- 4) 本資料に記載された情報、製品および回路等の使用に起因する損害または特許権その他権利の侵害に関して、当社は一切その責任を負いません。
- 5) 本製品および本資料の情報や回路などをご使用になる際、当社は第三者の工業所有権、知的所有権およびその他権利に対する保証または実施権を許諾致しません。
- 6) 本製品は、人命に関わる装置用としては開発されておりません。人命に関わる用途への採用をご検討の際は、当社までご相談ください。
- 7) 本資料の一部または全部を、当社に無断で転載および複製することを禁じます。

➤ お問い合わせ先

株式会社ステップテクニカ
〒207-0021 東京都東大和市立野1丁目1-15
TEL 042-569-8577 / E-Mail: info@steptechnica.com