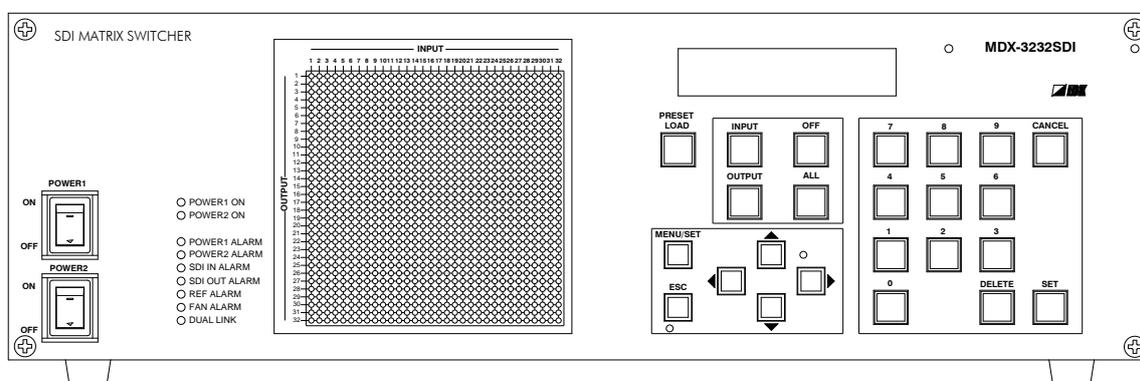




SDI マトリクススイッチャ MDX-3232SDI

<コマンドガイド>

取扱説明書 Ver.1.0.0



- この度は、本製品をお買い上げいただきまして誠にありがとうございます。
- 本製品の性能を十分に引き出してご活用いただくために、ご使用前に必ずこの取扱説明書をお読みください。また、お読みになった後は、本製品近くの見やすい場所に保管してください。

IDK Corporation

商標について

- Microsoft, および 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。
- Microsoft® .NET は、お客様、情報、システムおよびデバイスを繋ぐソフトウェアです。
- その他、記載されている会社名、製品名は、各社の登録商標または商標です。
なお、本文中において、®マークや™マークを省略している場合があります。

この取扱説明書をお読みいただく前に

- この取扱説明書の無断転載を禁じます。
- お客様がお持ちの製品のバージョンによっては、この取扱説明書に記載される外観図や通信コマンドなどが、一部異なる場合がありますのでご了承ください。
- 取扱説明書は改善のため、事前の予告なく変更することがあります。最新の取扱説明書は、弊社のホームページからダウンロードすることができます。

<http://www.idk.co.jp/>

取扱説明書の分冊構成

この取扱説明書は、目的に応じて分冊で提供しています。必要に応じて、各取扱説明書をお読みください。なお、コマンドガイドについては、ホームページからの提供となります。

■ ユーザーズガイド

[目的]

- ・ 簡単な操作方法を知る。
- ・ 設置し、他の機器と接続する。
- ・ 入出力調整や設定などをする。

■ コマンドガイド (本書)

[目的]

- ・ シリアル通信および LAN 通信などによる外部制御をする。

目次

1	シリアル通信仕様.....	5
2	LAN 通信仕様.....	6
2.1	TCP-IP コネクション数の制限と解決策.....	7
3	コマンド仕様.....	11
3.1	コマンド概要.....	11
3.2	コマンド一覧.....	12
3.3	コマンド詳細.....	14
3.3.1	エラーステータス.....	14
3.3.2	基本操作.....	15
3.3.3	プリセットメモリ.....	18
3.3.4	シリアル設定.....	21
3.3.5	LAN 設定.....	22
3.3.6	アラーム.....	26
3.3.7	SNMP 設定.....	30
3.3.8	その他設定.....	33
3.4	互換モード通信コマンド概要.....	35
3.4.1	互換モード通信コマンドの例.....	35
3.5	互換モード通信コマンドコマンド一覧.....	35
3.6	互換モード通信コマンドコマンド詳細.....	36
4	ASCII コード表.....	43

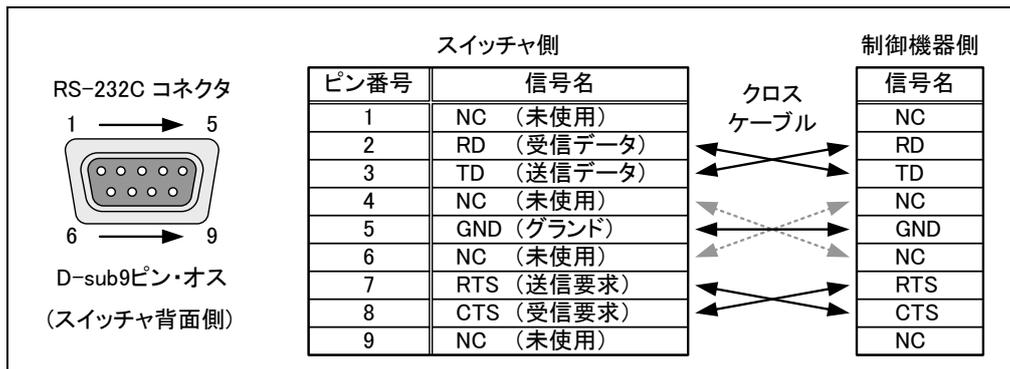
1 シリアル通信仕様

本機は、RS-232C、RS-422 コネクタを使ったシリアル通信による外部制御が可能です。

※注意 : 電源投入後、本機のインシャライズ終了までは約 30 秒かかります。
その間、通信制御はできません。

[表 1.1] シリアル通信仕様

準拠規格	RS-232C, RS-422
通信速度	4800, 9600, 19200, 38400 [bps]
データビット長	8 [bit]
パリティチェック	なし
ストップビット	1 [bit]
Xパラメーター	無効
フロー制御	なし
デリミタ	CRLF (復帰+改行, 16進表記の 0D と 0A)
通信方式	全二重



※フロー制御を行わないため、3線クロス (RD、TD、GND) ケーブルの使用が可能です。

[図 1.1] RS-232C コネクタ・ケーブル仕様



[図 1.2] RS-422 コネクタ・ケーブル仕様

2 LAN 通信仕様

本機は LAN による外部制御が可能です。

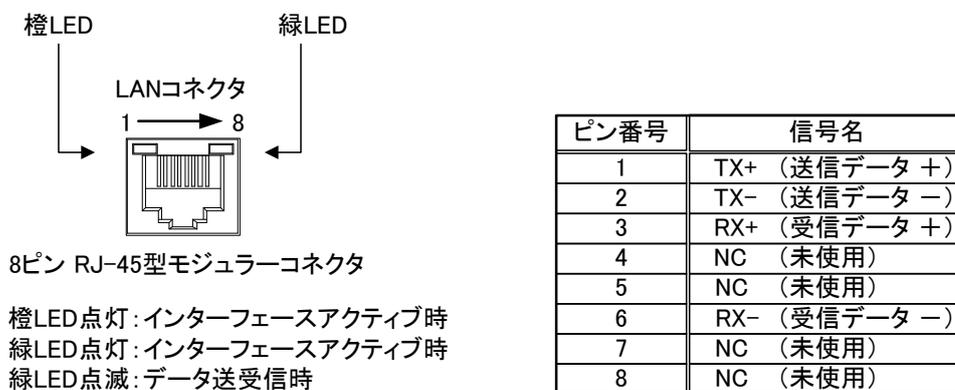
※注意 : 電源投入後、本機のイニシャライズ終了までは約 30 秒かかります。
その間、通信制御はできません。

[表 2.1] LAN 通信仕様

物理層	10Base-T (IEEE802.3i) / 100Base-TX (IEEE802.3u) Auto Negotiation, Auto MDI/MDI-X
ネットワーク層	ARP, IP, ICMP
トランスポート層	TCP, UDP ※UDP は SNMP にて使用。 コマンド制御使用ポート : 23, 1100, 6000~6999 WEB ブラウザ制御 (HTTP) 使用ポート : 80, 5000~5999 SNMP 使用ポート : 161 (固定) SNMP トラップのポートは動的に割り当てられます。
アプリケーション層	HTTP, SNMP
コネクション数	最大 8 個

※同時に使用することができるコネクション数は最大 8 個です。

コネクション接続後、30 秒以上通信がない場合、コネクションは自動的に切断されます。



ストレートケーブル/クロスケーブルの判別・切り換えを自動的に行なうAuto MDI/MDI-Xに対応していますので、本機とパソコンまたはハブなどとの接続の際、意識せずに接続することが可能です。

[図 2.1] LAN コネクタ仕様

2.1 TCP-IP コネクション数の制限と解決策

本機は最大 8 コネクション（8 ポート）まで同時に接続することができます。ただし、同時に使用することができるコネクション数が限られているため、9 台以上のパソコンから制御を行う場合には、マトリクススイッチャとのコネクションができなくなることがあります。

8 コネクションより多くのパソコンからコマンド制御を行う場合は、ユーザ側ソフトで、TCP-IP のコネクションと TCP-IP のクローズを通信コマンド送受信ごとに行うことで、本機側でポートの占有と解放が行われ常時ポートを占有されなくなり、論理的に 8 ポート以上の接続を行うことが可能となります。

[表 2.2] 接続数を増やす手法

ユーザ側パソコンソフト		マトリクススイッチャ
TCP-IP コネクション	→	(1 ポート占有)
コマンド送信 (@xxx)	→	
	←	コマンド返信 (@xxx)
TCP-IP クローズ	→	(1 ポート開放)

注意！

パソコン側からマトリクススイッチャへ 30 秒間コマンドの送信が行われなかった場合、マトリクススイッチャは、コネクション数制限の問題を回避するため、コネクションの切断処理を行います。そのためパソコン側からは再度コネクションを確立しないと通信ができなくなります。再度コネクションを確立するためには、今まで繋いでいたパソコン側のコネクションの切断処理をした後に、再度コネクションの確立処理を行ってください。

※ マトリクススイッチャ側のポート数は 8 ポートのため、パソコン側の電源が OFF にされた場合などは、コネクションが繋がったまま、永久にポートが占有されてしまいます。そのため、パソコン側から通信コマンドが来ない場合、コネクションの切断処理を行っています。

以下に、Microsoft Visual Basic .NET 2008 でのプログラミング例を示します。

105 行目の Button1.Click にて TextBox1 に送信する通信コマンド、TextBox2 に通信先ホスト、TextBox3 にポート番号を取得してマトリクススイッチャからの通信コマンド応答を取得します。

本例では、コネクションとクローズを繰り返した場合、データ送受信の遅延が問題になったときのために、以下のプログラミングを行っています。プログラミング例の 4、5、14、15、16 行目に相当します。

※ TcpClient.NoDelay

送信バッファまたは受信バッファが設定されているサイズを超えていない場合に遅延を無効にします。既定値は False です。

NoDelay プロパティの変更により、送受信による遅延時間を軽減することができます。

```
'クライアントをオープンします。
Private stClient As TcpClient          'クライアントクラス
Private stns As System.Net.Sockets.NetworkStream 'ストリームクラス
Private portNum As Integer 'ポート番号
Private hostName As String 'ホスト名

stClient = New System.Net.Sockets.TcpClient(hostName, portNum)
stClient.NoDelay = True '遅延を無効にします
stns = stClient.GetStream() 'ストリーム オープン
```

Microsoft Visual Basic .NET 2008 でのプログラミング例

```
1: Imports System
2: Imports System.Net.Sockets
3: Public Class Form1
4: Private stClient As TcpClient          'クライアント
5: Private stns As System.Net.Sockets.NetworkStream 'ストリーム

6: Public Function mOpen(ByVal pHostName As String, ByVal pPortNum As Integer) As Boolean
7:     '*****
8:     * オープン
9:     ' 戻り値 成功:True 失敗:False
10:    '*****
11:    mOpen = False '初期値
12:    Try
13:        'クライアントをオープンします。
14:        stClient = New System.Net.Sockets.TcpClient(pHostName, pPortNum)
15:        stClient.NoDelay = True '送信/受信遅延を無効にします。
16:        stns = stClient.GetStream() 'ストリーム オープン
17:        If stns.CanTimeout Then
18:            stns.ReadTimeout = 1000 'タイムアウト時間(1000ms)
19:        End If
20:        mOpen = True '成功
21:    Catch ex As Exception
22:        Console.WriteLine(ex.Message) '例外処理の表示
23:    End Try
24:
25: End Function
```

(次ページに続く)

```

26: Private Function mSendMessage(ByVal pMsg As String) As String
27: '*****
28: *   メッセージ送信
29: '   pMsg   送信メッセージ
30: '   戻り値  返答文字列
31: '*****
32: Dim dtBirth As DateTime      'タイムアウト時間
33: Dim wNow As DateTime        '現在時間
34: Dim pRecvMsg As String      '返信メッセージ
35: Dim bytes2(1024) As Byte    '返信メッセージ一時格納エリア(Byte型)
36: Dim bytesRead2 As Integer   '返信メッセージ一時格納エリア(Integer型)
37: Dim word As Byte()         'システム出力時の書き出し一時データ格納エリア
38:
39: mSendMessage = ""          '返信値 クリア
40: pRecvMsg = ""             'ワークエリアクリア
41:
42: Try
43: '---送信チェック---
44: If stns.CanWrite Then      '書き込み可能?
45:     '文字エンコード
46:     word = System.Text.Encoding.Default.GetBytes(pMsg + vbCrLf)
47:     'ソケットに出力
48:     stns.Write(word, 0, word.Length)
49: Else
50:     Exit Function
51: End If
52:
53: '---受信---
54: dtBirth = DateTime.Now
55: dtBirth = dtBirth.AddSeconds(3) '3秒でタイムアウト
56: Do
57:     wNow = DateTime.Now      '現在時間と比較
58:     If (wNow > dtBirth) Then
59:         Exit Do              'オーバーした場合処理を中断
60:     End If
61:
62:     If stns.CanRead Then     '読み込み可能状態の場合
63:         'データの読み込み
64:         bytesRead2 = stns.Read(bytes2, 0, bytes2.Length)
65:         'エンコード
66:         pRecvMsg = pRecvMsg & _
67:             System.Text.Encoding.Default.GetString(bytes2, 0, bytesRead2)
68:         If pRecvMsg <> "" Then
69:             '@から CRLF までを抽出
70:             If ((InStr(pRecvMsg, vbCrLf) <> 0) And (InStr(pRecvMsg, "@") <> 0)) Then
71:                 pRecvMsg = Mid(pRecvMsg, _
72:                     InStr(pRecvMsg, "@"), _
73:                     InStr(pRecvMsg, vbCrLf) _
74:                     - InStr(pRecvMsg, "@"))
75:             End Do
76:         End If
77:     End If
78: End Do
79: Loop
80: mSendMessage = pRecvMsg     '受信データを返答
81:
82: Catch ex As Exception
83:     Console.WriteLine(ex.Message) '例外処理の表示
84: End Try
85:
86: End Function

```

```

87: Public Sub mClose()
88:     '*****
89:     * クローズ
90:     '*****
91:     Try
92:         If Not stns Is Nothing Then 'ストリームの存在有無
93:             stns.Close()           'ストリームクローズ
94:         End If
95:
96:         If Not stClient Is Nothing Then 'クライアントの存在有無
97:             stClient.Close()         'クライアントクローズ
98:         End If
99:
100:    Catch ex As Exception
101:        Console.WriteLine(ex.Message) '例外処理の表示
102:    End Try
103:
104: End Sub
105: Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
106:    Dim wRecvMsg As String '受信文字列格納場所
107:    Dim i As Integer
108:    Dim wHostName As String
109:    Dim wPortNum As Integer
110:
111:    If (TextBox2.Text = "") Then 'ホスト名チェック
112:        MsgBox("ホスト名が設定されてません。")
113:        Exit Sub
114:    End If
115:
116:    wHostName = TextBox2.Text
117:
118:    If (TextBox3.Text = "") Then 'ポート番号チェック
119:        MsgBox("ポート番号が設定されてません。")
120:        Exit Sub
121:    End If
122:    wPortNum = Val(TextBox3.Text)
123:
124:    If TextBox1.Text = "" Then '送信文字列チェック
125:        MsgBox("送信文字が設定されてません。")
126:        Exit Sub
127:    End If
128:
129:    Label6.Text = ""
130:
131:    For i = 0 To 2 '3 回リトライを繰り返します (万が一、他のクライアントによりマトリクススイッチャ側のポートが全て使われた
        場合の処理)
132:        If Not mOpen(wHostName, wPortNum) Then
133:            MsgBox("通信オープンエラー")
134:            GoTo Exit_Step
135:        End If
136:
137:        'オープンします
138:        wRecvMsg = mSendMessage(TextBox1.Text) '送信します
139:        Console.WriteLine("wRecvMsg:" & wRecvMsg & Now)
140:        mClose() 'クローズします
141:
142:        If wRecvMsg <> "" Then
143:            Label6.Text = wRecvMsg
144:            GoTo Exit_Step
145:        End If
146:
147:    Next i
148:
149:    MsgBox("送信エラー")
150:
151: Exit_Step:
152:
153: End Sub
154: End Class

```

3 コマンド仕様

3.1 コマンド概要

コマンドは各コマンドを識別する@ (16進表記の40) と3文字の半角英字 (大文字小文字) とそれに続くパラメータ (半角数字) からなります。(コマンドによっては複数のパラメータを指定可能なものやパラメータの必要ないものがあります) コマンドの最後にデリミタを送信することにより処理を実行します。

例 : @IOS, 1, 1☐

「,」は、コマンドとパラメータおよびパラメータ間の区切り文字で、カンマ (16進表記の2C) を表します。
☐は、デリミタ (CR+LFで16進表記の0D+0A) を表します。

※一部のコマンドには、ASCII コードを指定するパラメータがあります。

① 本機の設定を変更するコマンド

コマンドとパラメータをカンマで区切って送信します。(コマンドによっては複数のパラメータを指定可能なものや、パラメータの必要ないものがあります)

コマンドが正常に処理されると、受信したコマンドをそのまま送り返します。

例 : @IOS, 1, 1☐ ←送信コマンド
@IOS, 1, 1☐ ←本機からの返り値

② 本機の設定を受信するコマンド

コマンドを送信します。コマンドが正常に処理されると、受信したコマンドに続き、現在の設定を送り返します。(コマンドによっては複数のパラメータを返すものがあります)

例 : @GCP☐ ←送信コマンド
@GCP, 1, 2, 3, 4☐ ←本機からの返り値

③ エラーコマンド

未定義のコマンドやパラメータに誤りがある場合などは、エラーコマンドとエラーの詳細 (半角数字) を送り返します。

例 : @IOS, 65, 1☐ ←送信コマンド (この例ではパラメータに誤りがあります)
@ERR, 1☐ ←本機からのエラーコマンド

3.2 コマンド一覧

■ エラーステータス

コマンド	機能	詳細ページ
@ERR	エラーステータス	14

■ 基本操作

コマンド	機能	詳細ページ
@IOS	入出力チャンネル切換	15
@IOV	入出力チャンネル切換 (※弊社製品との互換用コマンド)	15
@GCP	入出力チャンネル取得	16
@GVA	入出力チャンネル取得 (※弊社製品との互換用コマンド)	16
@SAO	全チャンネル出力設定	16
@SSC	入出力チャンネル ストレート設定	17
@SSV	入出力チャンネル ストレート設定 (※弊社製品との互換用コマンド)	17

■ プリセットメモリ

コマンド	機能	詳細ページ
@RPM	プリセットメモリの読み出し (入出力チャンネル設定読み出し)	18
@SPM	プリセットメモリの上書き保存 (入出力チャンネル設定保存)	18
@GCM	プリセットメモリの設定取得 (入出力チャンネル設定の取得)	19
@ECM	プリセットメモリの設定編集 (入出力チャンネル設定の編集)	19
@SEM	プリセットメモリの引き継ぎ保存 (入出力チャンネル設定保存)	20
@SMU	電源投入時の状態設定	20
@GMU	電源投入時の状態取得	20

■ シリアル端子

コマンド	機能	詳細ページ
@SCT	シリアル通信端子 通信速度設定	21
@GCT	シリアル通信端子 通信速度取得	21

■ LAN 設定

コマンド	機能	詳細ページ
@SIP	IP アドレス設定	22
@GIP	IP アドレス取得	22
@SSB	サブネットマスク設定	22
@GSB	サブネットマスク取得	22
@SGW	ゲートウェイアドレス設定	23
@GGW	ゲートウェイアドレス取得	23
@SLP	TCP ポート番号設定	23
@GLP	TCP ポート番号取得	23
@GMA	MAC アドレス取得 (10 進数)	24
@GMC	MAC アドレス取得 (16 進数)	24
@SBP	WEB ブラウザパスワードロック機能設定	24
@GBP	WEB ブラウザパスワードロック機能取得	25
@SBR	WEB ブラウザ自動更新時間設定	25
@GBR	WEB ブラウザ自動更新時間取得	25

■アラーム

コマンド	機能	詳細ページ
@GAI	アラーム情報取得	26
@GPV	電源電圧値の取得	26
@SID	SDI 入力キャリア検出設定	27
@GID	SDI 入力キャリア検出設定取得	27
@GIC	SDI 入力キャリア検出値取得	27
@SOD	SDI 出力キャリア検出設定	28
@GOD	SDI 出力キャリア検出設定取得	28
@GOC	SDI 出力キャリア検出値取得	28
@SRD	基準同期信号検出設定	29
@GRD	基準同期信号検出設定取得	29
@GRC	基準同期信号検出値取得	29
@GFV	ファン回転数取得	29

■SNMP 設定

コマンド	機能	詳細ページ
@SNN	SNMP コミュニティ名の設定	30
@GNN	SNMP コミュニティ名の取得	30
@SNM	SNMP 有効/無効の設定	30
@GNM	SNMP 有効/無効の取得	30
@STN	SNMP トラップ送信 コミュニティ名の設定	31
@GTN	SNMP トラップ送信 コミュニティ名の取得	31
@STA	SNMP トラップ送信先 IP アドレス設定	31
@GTA	SNMP トラップ送信先 IP アドレス取得	31
@SNT	SNMP トラップ送信 有効/無効の設定	32
@GNT	SNMP トラップ送信 有効/無効の取得	32

■その他設定

コマンド	機能	詳細ページ
@SLM	キーロックモードの設定	33
@GLM	キーロックモードの取得	33
@SBZ	ブザー音設定	33
@GBZ	ブザー音設定取得	33
@GIV	バージョン情報取得	34
@GDL	デュアルリンク SDI モード状態取得	34

3.3 コマンド詳細

3.3.1 エラーステータス

@ERR	エラーステータス		
コマンド書式	返り値のみ		
返り値書式	@ERR, <i>error</i> [↵]		
パラメータ	<i>error</i> : エラーステータス 1 = パラメータの書式、値にエラーがあります。 2 = 未定義のコマンドまたはコマンドの書式に誤りがあります。 3 = パスワード認証エラー。		
実行例	送	@IOS, 999, 1 [↵]	パラメータエラー。
	受	@ERR, 1 [↵]	
	送	@XYZ [↵]	未定義のコマンド。
	受	@ERR, 2 [↵]	

3.3.2 基本操作

@IOS	入出力チャンネル切換		
コマンド書式	@IOS, input_1, output_1 (, input_2, output_2 ...)		
返り値書式	@IOS, input_1, output_1 (, input_2, output_2 ...)		
パラメータ	input_1-32 : 入力チャンネル 0 = OFF ※初期値, 1~32 = INPUT1~32		
	output_1-32 : 出力チャンネル 0 = 全出力, 1~32 = OUTPUT1~32		
実行例	送	@IOS, 1, 3	出力 3 に入力 1 を選択する。
	受	@IOS, 1, 3	正常終了。
	送	@IOS, 0, 3	出力 3 を OFF にする。
	受	@IOS, 0, 3	正常終了。
	送	@IOS, 2, 0	全出力に入力 2 を選択する。
	受	@IOS, 2, 0	正常終了。
	送	@IOS, 4, 1, 3, 2, 2, 3	出力 1 に入力 4、出力 2 に入力 3、出力 3 に入力 2 を選択する。
	受	@IOS, 4, 1, 3, 2, 2, 3	正常終了。
関連項目	@GCP 入出力チャンネル取得 @GVA 入出力チャンネル取得 (※弊社製品との互換用コマンド)		
備考	デュアルリンク SDI モードの場合、チャンネル数は半分になります。 8 系統単位での入出力数をカスタマイズした場合のチャンネル数は、カスタマイズした入出力数になります。		

@IOV	入出力チャンネル切換 (※弊社製品との互換用コマンド)		
コマンド書式	@IOV, input_1, output_1 (, input_2, output_2 ...)		
返り値書式	@IOV, input_1, output_1 (, input_2, output_2 ...)		
パラメータ	input_1-32 : 入力チャンネル 0 = OFF ※初期値, 1~32 = INPUT1~32		
	output_1-32 : 出力チャンネル 0 = 全出力, 1~32 = OUTPUT1~32		
実行例	送	@IOV, 1, 3	出力 3 に入力 1 を選択する。
	受	@IOV, 1, 3	正常終了。
	送	@IOV, 0, 3	出力 3 を OFF にする。
	受	@IOV, 0, 3	正常終了。
	送	@IOV, 2, 0	全出力に入力 2 を選択する。
	受	@IOV, 2, 0	正常終了。
	送	@IOV, 4, 1, 3, 2, 2, 3	出力 1 に入力 4、出力 2 に入力 3、出力 3 に入力 2 を選択する。
	受	@IOV, 4, 1, 3, 2, 2, 3	正常終了。
関連項目	@GCP 入出力チャンネル取得 @GVA 入出力チャンネル取得 (※弊社製品との互換用コマンド)		
備考	デュアルリンク SDI モードの場合、チャンネル数は半分になります。 8 系統単位での入出力数をカスタマイズした場合のチャンネル数は、カスタマイズした入出力数になります。		

@GCP	入出力チャンネル取得	
コマンド書式	@GCP [↵]	
返り値書式	@GCP, <i>ch_1</i> , <i>ch_2</i> , ..., <i>ch_31</i> , <i>ch_32</i> [↵]	
パラメータ	<i>ch_1-32</i> : 入力チャンネル 0 = OFF, 1~32 = INPUT1~32	
実行例	送 @GCP [↵] 受 @GCP, 1, 2, ..., 31, 32 [↵]	入出力チャンネルを取得する。
関連項目	@IOS 入出力チャンネル切換 @IOV 入出力チャンネル切換 (※弊社製品との互換用コマンド)	
備考	デュアルリンク SDI モードの場合、チャンネル数は半分になります。 8 系統単位での入出力数をカスタマイズした場合のチャンネル数は、カスタマイズした入出力数になります。	

@GVA	入出力チャンネル取得 (※弊社製品との互換用コマンド)	
コマンド書式	@GVA [↵]	
返り値書式	@GVA, <i>ch_1</i> , <i>ch_2</i> , ..., <i>ch_31</i> , <i>ch_32</i> [↵]	
パラメータ	<i>ch_1-32</i> : 入力チャンネル 0 = OFF, 1~32 = INPUT1~32	
実行例	送 @GVA [↵] 受 @GVA, 1, 2, ..., 31, 32 [↵]	入出力チャンネルを取得する。
関連項目	@IOS 入出力チャンネル切換 @IOV 入出力チャンネル切換 (※弊社製品との互換用コマンド)	
備考	デュアルリンク SDI モードの場合、チャンネル数は半分になります。 8 系統単位での入出力数をカスタマイズした場合のチャンネル数は、カスタマイズした入出力数になります。	

@SAO	全チャンネル出力設定	
コマンド書式	@SAO, <i>input</i> [↵]	
返り値書式	@SAO, <i>input</i> [↵]	
パラメータ	<i>input</i> : 入力チャンネル 0 = OFF ※初期値, 1~32 = INPUT1~32	
実行例	送 @SAO, 3 [↵] 受 @SAO, 3 [↵]	全出力に入力 3 を選択する。 正常終了。
関連項目	@GCP 入出力チャンネル取得 @GVA 入出力チャンネル取得 (※弊社製品との互換用コマンド)	
備考	デュアルリンク SDI モードの場合、チャンネル数は半分になります。 8 系統単位での入出力数をカスタマイズした場合のチャンネル数は、カスタマイズした入出力数になります。	

@SSC	入出力チャンネル ストレート設定	
コマンド書式	@SSC [↵]	
返り値書式	@SSC [↵]	
パラメータ	なし	
実行例	送 @SSC[↵] 受 @SSC[↵]	入力チャンネルと出力チャンネルをストレート (1:1) に設定する。 (出力 1 に入力 1、出力 2 に入力 2、・・・)
関連項目	@GCP 入出力チャンネル取得 @GVA 入出力チャンネル取得 (※弊社製品との互換用コマンド)	

@SSV	入出力チャンネル ストレート設定 (※弊社製品との互換用コマンド)	
コマンド書式	@SSV [↵]	
返り値書式	@SSV [↵]	
パラメータ	なし	
実行例	送 @SSV[↵] 受 @SSV[↵]	入力チャンネルと出力チャンネルをストレート (1:1) に設定する。 (出力 1 に入力 1、出力 2 に入力 2、・・・)
関連項目	@GCP 入出力チャンネル取得 @GVA 入出力チャンネル取得 (※弊社製品との互換用コマンド)	

3.3.3 プリセットメモリ

@RPM	プリセットメモリの読み出し（入出力チャンネル設定読み出し）	
コマンド書式	@RPM, <i>preset</i> [↵]	
返り値書式	@RPM, <i>preset</i> [↵]	
パラメータ	<i>preset</i> : プリセットメモリ 1~32 = PRESET1~32	
実行例	送 @RPM, 3[↵] 受 @RPM, 3[↵]	プリセットメモリ 3 を読み出す。 正常終了。
関連項目	@SPM プリセットメモリの上書き保存（入出力チャンネル設定保存） @SEM プリセットメモリの引き継ぎ保存（入出力チャンネル設定保存）	

@SPM	プリセットメモリの上書き保存（入出力チャンネル設定保存）	
コマンド書式	@SPM, <i>preset</i> (, <i>name</i>) [↵]	
返り値書式	@SPM, <i>preset</i> (, <i>name</i>) [↵]	
パラメータ	<i>preset</i> : プリセットメモリ 1~32 = PRESET1~32 <i>name</i> : メモリ名 ASCII コードの 20 ~ 7D の中から半角英数最大 10 文字まで設定可能です。 メモリ名は省略可能で、省略した場合は、現在保存されている名前を変更せずに 入出力チャンネルの設定のみ保存します。	
実行例	送 @SPM, 2[↵] 受 @SPM, 2[↵]	現在の入出力チャンネル設定を プリセットメモリ 2 に保存する。 [メモリ名を省略した場合]
	送 @SPM, 2, MEMORY2[↵] 受 @SPM, 2, MEMORY2[↵]	[メモリ名を指定した場合]
関連項目	@RPM プリセットメモリの読み出し（入出力チャンネル設定読み出し） @SEM プリセットメモリの引き継ぎ保存（入出力チャンネル設定保存）	

@GCM	プリセットメモリの設定取得（入出力チャンネル設定の取得）	
コマンド書式	@GCM, <i>preset</i>	
返り値書式	@GCM, <i>preset, ch_1, ch_2, …, ch_31, ch_32</i> 	
パラメータ	<i>preset</i> : プリセットメモリ 1~32 = PRESET1~32	
	<i>ch_1-32</i> : 映像出力 -1 = 制御しない ※初期値, 0 = OFF, 1~32= INPUT1~32	
実行例	プリセットメモリ 2 を読み出すと、出力 1 は入力 1、出力 2 は入力 2、出力 3~32 は制御しない設定になっている。	
	送 受	@GCM, 2  @GCM, 2, 1, 2, -1, -1, …, -1, -1  プリセットメモリ 2 の設定取得
関連項目	@SPM プリセットメモリの上書き保存（入出力チャンネル設定保存） @ECM プリセットメモリの設定編集（入出力チャンネル設定の編集） @SEM プリセットメモリの引き継ぎ保存（入出力チャンネル設定保存）	
備考	デュアルリンク SDI モードの場合、チャンネル数は半分になります。 8 系統単位での入出力数をカスタマイズした場合のチャンネル数は、カスタマイズした入出力数になります。 プリセットメモリ数は最大 32 のままです。	

@ECM	プリセットメモリの設定編集（入出力チャンネル設定の編集）	
コマンド書式	@ECM, <i>preset, ch_1, ch_2, …, ch_31, ch_32</i> 	
返り値書式	@ECM, <i>preset, ch_1, ch_2, …, ch_31, ch_32</i> 	
パラメータ	<i>preset</i> : プリセットメモリ 1~32 = PRESET1~32	
	<i>ch_1-32</i> : 映像出力 -1 = 制御しない ※初期値, 0 = OFF, 1 ~32 = INPUT1~32	
実行例	プリセットメモリ 2 を出力 1 は入力 1、出力 2 は入力 2、出力 3~32 は制御しない設定にする。	
	送 受	@ECM, 2, 1, 2, -1, -1, …, -1, -1  @ECM, 2, 1, 2, -1, -1, …, -1, -1  プリセットメモリ 2 の設定編集
関連項目	@SPM プリセットメモリの上書き保存（入出力チャンネル設定保存） @GCM プリセットメモリの設定取得（入出力チャンネル設定の取得） @SEM プリセットメモリの引き継ぎ保存（入出力チャンネル設定保存）	
備考	デュアルリンク SDI モードの場合、チャンネル数は半分になります。 8 系統単位での入出力数をカスタマイズした場合のチャンネル数は、カスタマイズした入出力数になります。 プリセットメモリ数は最大 32 のままです。	

@SEM	プリセットメモリの引き継ぎ保存（入出力チャンネル設定保存）	
コマンド書式	@SEM, preset(, name) 	
返り値書式	@SEM, preset(, name) 	
パラメータ	<pre> preset : プリセットメモリ 1~32 = PRESET1~32 name : メモリ名 ASCII コードの、20 ~ 7D の中から半角英数最大 10 文字まで設定可能です。メモリ名は省略可能で、省略した場合は、現在保存されている名前を変更せずに入出力チャンネルの設定のみ保存します。 </pre>	
実行例	現在の入出力チャンネル設定をプリセットメモリ 2 に保存する。ただし、チャンネル制御しない設定が保存されている出力チャンネルの設定は保存しない。	
	送	@SEM, 2  [メモリ名を省略した場合]
	受	@SEM, 2 
実行例	送	@SEM, 2, MEMORY2  [メモリ名を指定した場合]
	受	@SEM, 2, MEMORY2 
関連項目	@RPM プリセットメモリの読み出し（入出力チャンネル設定読み出し） @SPM プリセットメモリの上書き保存（入出力チャンネル設定保存）	

@SMU	電源投入時の状態設定	
コマンド書式	@SMU, state 	
返り値書式	@SMU, state 	
パラメータ	<pre> state : 電源投入時の設定 1~32 = プリセットメモリ (PRESET1~PRESET32) 33 = デフォルトチャンネル (LAST CH NO SAVE) 34 = ラストメモリ (LAST CH SAVE ※初期値) </pre>	
実行例	送	@SMU, 3  電源投入時、プリセットメモリ 3 に登録された入出力チャンネルで起動するように設定する。
	受	@SMU, 3  正常終了。
関連項目	@GMU 電源投入時の状態取得	

@GMU	電源投入時の状態取得	
コマンド書式	@GMU 	
返り値書式	@GMU, state 	
パラメータ	<pre> state : 電源投入時の設定 1~32 = プリセットメモリ (PRESET1~PRESET32) 33 = デフォルトチャンネル (LAST CH NO SAVE) 34 = ラストメモリ (LAST CH SAVE ※初期値) </pre>	
実行例	送	@GMU  電源投入時の状態を取得する。
	受	@GMU, 3  プリセットメモリ 3
関連項目	@SMU 電源投入時の状態設定	

3.3.4 シリアル設定

@SCT	シリアル通信端子 通信速度設定	
コマンド書式	@SCT, <i>port</i> , <i>setting</i> ↵	
返り値書式	@SCT, <i>port</i> , <i>setting</i> ↵	
パラメータ	<i>port</i> : シリアル通信端子 0 = 全端子, 1 = RS-232C 端子, 2 = RS-422 端子 <hr/> <i>setting</i> : 通信速度設定 0 = 4800[bps], 1 = 9600[bps] (※初期値), 2 = 19200[bps], 3 = 38400[bps]	
実行例	送 @SCT, 1, 2 ↵ 受 @SCT, 1, 2 ↵	RS-232C 端子の通信速度を 19200[bps] に設定する。 正常終了。
関連項目	@GCT シリアル通信端子 通信速度取得	

@GCT	シリアル通信端子 通信速度取得	
コマンド書式	@GCT ↵	
返り値書式	@GCT, <i>rs-232c</i> , <i>rs-422</i> ↵	
パラメータ	<i>rs-232c</i> : RS-232C 端子 通信速度設定 <i>rs-422</i> : RS-422 端子 通信速度設定 0 = 4800[bps], 1 = 9600[bps] (※初期値), 2 = 19200[bps], 3 = 38400[bps]	
実行例	送 @GCT ↵ 受 @GCT, 2, 1 ↵	シリアルコネクタの通信速度設定を取得する。 ・ RS-232C コネクタ: 19200[bps] ・ RS-422 コネクタ: 9600[bps]
関連項目	@SCT シリアル通信端子 通信速度設定	

3.3.5 LAN 設定

@SIP	IP アドレス設定	
コマンド書式	@SIP, unit_1, unit_2, unit_3, unit_4 [↵]	
返り値書式	@SIP, unit_1, unit_2, unit_3, unit_4 [↵]	
パラメータ	unit_1 : IP アドレス上位 ~ unit_4 : IP アドレス下位 0 ~ 255 = 8 ビット(10 進数表記) ※初期値 192.168.1.199	
実行例	送 @SIP, 192, 168, 3, 2[↵] 受 @SIP, 192, 168, 3, 2[↵]	IP アドレスを 192.168.3.2 に設定する。 正常終了。
関連項目	@GIP IP アドレス取得	

@GIP	IP アドレス取得	
コマンド書式	@GIP [↵]	
返り値書式	@GIP, unit_1, unit_2, unit_3, unit_4 [↵]	
パラメータ	unit_1 : IP アドレス上位 ~ unit_4 : IP アドレス下位 0 ~ 255 = 8 ビット(10 進数表記) ※初期値 192.168.1.199	
実行例	送 @GIP[↵] 受 @GIP, 192, 168, 3, 2[↵]	本機の IP アドレスを取得する。 192.168.3.2
関連項目	@SIP IP アドレス設定	

@SSB	サブネットマスク設定	
コマンド書式	@SSB, unit_1, unit_2, unit_3, unit_4 [↵]	
返り値書式	@SSB, unit_1, unit_2, unit_3, unit_4 [↵]	
パラメータ	unit_1 : サブネットマスク上位 ~ unit_4 : サブネットマスク下位 0 ~ 255 = 8 ビット(10 進数表記) ※初期値 255.255.255.0	
実行例	送 @SSB, 255, 255, 192, 0[↵] 受 @SSB, 255, 255, 192, 0[↵]	サブネットマスクを 255.255.192.0 に設定する。 正常終了。
	送 @SSB, 255, 0, 0, 128[↵] 受 @ERR, 1[↵]	サブネットマスクとして不正な値を指定するとエラーが返されます。
関連項目	@GSB サブネットマスク取得	

@GSB	サブネットマスク取得	
コマンド書式	@GSB [↵]	
返り値書式	@GSB, unit_1, unit_2, unit_3, unit_4 [↵]	
パラメータ	unit_1 : サブネットマスク上位 ~ unit_4 : サブネットマスク下位 0 ~ 255 = 8 ビット(10 進数表記) ※初期値 255.255.255.0	
実行例	送 @GSB[↵] 受 @GSB, 255, 255, 192, 0[↵]	サブネットマスクを取得する。 255.255.192.0
関連項目	@SSB サブネットマスク設定	

@SGW	ゲートウェイアドレス設定	
コマンド書式	@SGW, unit_1, unit_2, unit_3, unit_4 [↵]	
返り値書式	@SGW, unit_1, unit_2, unit_3, unit_4 [↵]	
パラメータ	unit_1 : ゲートウェイアドレス上位 ~ unit_4 : ゲートウェイアドレス下位 0 ~ 255 = 8 ビット(10進数表記) ※初期値 192.168.1.200	
実行例	送 @SGW, 192, 168, 1, 200[↵] 受 @SGW, 192, 168, 1, 200[↵]	ゲートウェイアドレスを 192.168.1.200 に設定する。 正常終了。
関連項目	@GGW ゲートウェイアドレス取得	

@GGW	ゲートウェイアドレス取得	
コマンド書式	@GGW [↵]	
返り値書式	@GGW, unit_1, unit_2, unit_3, unit_4 [↵]	
パラメータ	unit_1 : ゲートウェイアドレス上位 ~ unit_4 : ゲートウェイアドレス下位 0 ~ 255 = 8 ビット(10進数表記) ※初期値 192.168.1.200	
実行例	送 @GGW[↵] 受 @GGW, 192, 168, 1, 200[↵]	ゲートウェイアドレスを取得する。 192.168.1.200
関連項目	@SGW ゲートウェイアドレス設定	

@SLP	TCP ポート番号設定	
コマンド書式	@SLP, port_1, port_2 [↵]	
返り値書式	@SLP, port_1, port_2 [↵]	
パラメータ	port_1 : 通信コマンド制御ポート 23, 1100, 6000 ~ 6999 ※初期値 1100 port_2 : WEB ブラウザ制御ポート 80, 5000 ~ 5999 ※初期値 80	
実行例	送 @SLP, 1100, 80[↵] 受 @SLP, 1100, 80[↵]	ポート番号を設定する。 正常終了。
関連項目	@GLP TCP ポート番号取得	

@GLP	TCP ポート番号取得	
コマンド書式	@GLP [↵]	
返り値書式	@GLP, port_1, port_2 [↵]	
パラメータ	port_1 : 通信コマンド制御ポート 23, 1100, 6000 ~ 6999 ※初期値 1100 port_2 : WEB ブラウザ制御ポート 80, 5000 ~ 5999 ※初期値 80	
実行例	送 @GLP[↵] 受 @GLP, 1100, 80[↵]	ポート番号を取得する。 通信コマンド制御ポートは1100、WEBブラウザ制御ポートは80。
関連項目	@SLP TCP ポート番号設定	

@GMA	MAC アドレス取得 (10 進数)	
コマンド書式	@GMA [↵]	
返り値書式	@GMA, unit_1, unit_2, unit_3, unit_4, unit_5, unit_6 [↵]	
パラメータ	unit_1 : MAC アドレス上位 ~ unit_6 : MAC アドレス下位 0 ~ 255 = 8 ビット(10 進数表記)	
実行例	送 @GMA[↵] 受 @GMA, 0, 8, 229, 64, 0, 0[↵]	MAC アドレス(10 進数)を取得する。 ※MAC アドレスは出荷時に機器単位に個別に設定 されます
関連項目	@GMC MAC アドレス取得 (16 進数)	

@GMC	MAC アドレス取得 (16 進数)	
コマンド書式	@GMC [↵]	
返り値書式	@GMC, unit_1, unit_2, unit_3, unit_4, unit_5, unit_6 [↵]	
パラメータ	unit_1 : MAC アドレス上位 ~ unit_6 : MAC アドレス下位 00 ~ FF = 8 ビット(16 進数表記)	
実行例	送 @GMC[↵] 受 @GMC, 00, 08, E5, 40, 00, 00[↵]	MAC アドレス(16 進数)を取得する。 ※MAC アドレスは出荷時に機器単位に個別に設定 されます
関連項目	@GMA MAC アドレス取得 (10 進数)	

@SBP	WEB ブラウザパスワードロック機能設定	
コマンド書式	@SBP, on/off, pass, new pass [↵]	
返り値書式	@SBP, on/off, pass, new pass [↵]	
パラメータ	on/off : WEB ブラウザパスワードロック機能設定 0 = OFF ※初期値, 1 = ON pass : 現在設定されているパスワード ASCII コードの 20, 30~39, 41~5A, 61~7A の中から最大 10 文字まで設定可能です。 パスワードを OFF から ON に設定する場合は省略可能です。 ※初期値 IDK new pass : 新しく設定するパスワード ASCII コードの 20, 30~39, 41~5A, 61~7A の中から最大 10 文字まで設定可能です。 パスワードを OFF に設定する場合は省略可能です。 ※全てスペース(0x20)での設定はできません。	
実行例	送 @SBP, 1, IDK, IDK1234[↵] 受 @SBP, 1, IDK, IDK1234[↵] 送 @SBP, 1, IDK[↵] 受 @SBP, 1, IDK[↵] 送 @SBP, 0, IDK[↵] 受 @SBP, 0, IDK[↵] 送 @SBP, 0, IDK[↵] 受 @ERR, 3[↵]	パスワードの変更。 現在設定されているパスワード " IDK"。 新しいパスワード " IDK1234"。 ブラウザパスワードロック機能を OFF→ON に設定。 新しいパスワード " IDK"。 ブラウザパスワードロック機能を ON→OFF に設定。 現在設定されているパスワード " IDK"。 パスワードの認証に失敗。
関連項目	@GBP WEB ブラウザパスワードロック機能取得	

@GBP	WEB ブラウザパスワードロック機能取得	
コマンド書式	@GBP [↵]	
返り値書式	@GBP, <i>pass</i> [↵]	
パラメータ	<i>pass</i> : WEB ブラウザパスワードロック機能設定状態 0 = OFF ※初期値, 1 = ON	
実行例	送 @GBP [↵]	WEB ブラウザパスワードロック機能設定状態を取得する。
	受 @GBP, 1 [↵]	WEB ブラウザパスワードロック機能設定状態 ON。
関連項目	@SBP WEB ブラウザパスワードロック機能設定	

@SBR	WEB ブラウザ自動更新時間設定	
コマンド書式	@SBR, <i>time</i> [↵]	
返り値書式	@SBR, <i>time</i> [↵]	
パラメータ	<i>time</i> : WEB ブラウザ自動更新時間 0, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80, 85, 90, 95, 100 [sec] ※初期値 0 [sec]	
実行例	送 @SBR, 30 [↵]	WEB ブラウザ自動更新時間を 30[sec]に設定する。
	受 @SBR, 30 [↵]	正常終了。
関連項目	@GBR WEB ブラウザ自動更新時間取得	

@GBR	WEB ブラウザ自動更新時間取得	
コマンド書式	@GBR [↵]	
返り値書式	@GBR, <i>time</i> [↵]	
パラメータ	<i>time</i> : WEB ブラウザ自動更新時間 0, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80, 85, 90, 95, 100 [sec] ※初期値 0 [sec]	
実行例	送 @GBR [↵]	WEB ブラウザ自動更新時間を取得する。
	受 @GBR, 30 [↵]	30[sec]。
関連項目	@SBR WEB ブラウザ自動更新時間設定	

3.3.6 アラーム

@GAI	アラーム情報取得	
コマンド書式	@GAI [↵]	
返り値書式	@GAI, power1, power2, sdi_i, sdi_o, ref, fan [↵]	
パラメータ	power1 : 電源 1 のアラーム状態 1 = 正常, 2 = 異常 (アラームあり)	
	power2 : 電源 2 のアラーム状態 1 = 正常, 2 = 異常 (アラームあり)	
	sdi_i : SDI 入力キャリア検出アラーム状態 0 = 検出設定なし, 1 = 正常, 2 = 異常 (アラームあり)	
	sdi_o : SDI 出力キャリア検出アラーム状態 0 = 検出設定なし, 1 = 正常, 2 = 異常 (アラームあり)	
	ref : 基準同期信号検出アラーム状態 0 = 検出設定なし, 1 = 正常, 2 = 異常 (アラームあり)	
	fan : ファン回転数のアラーム状態 1 = 正常, 2 = 異常 (アラームあり)	
実行例	送 @GAI [↵]	アラーム情報を取得する。
	受 @GAI, 1, 1, 1, 1, 1, 1 [↵]	アラーム全正常。
関連項目	アラーム関連コマンド	

@GPV	電源電圧値の取得	
コマンド書式	@GPV [↵]	
返り値書式	@GPV, power1, power2 [↵]	
パラメータ	power1 : 電源 1 電圧値 (mV 単位) 0 ~ xxxx	
	power2 : 電源 2 電圧値 (mV 単位) 0 ~ xxxx	
実行例	送 @GPV [↵]	電源電圧値を取得する。
	受 @GPV, 5000, 5000 [↵]	電源電圧値 5000 mV。
関連項目	@GAI アラーム情報取得	

@SID	SDI 入力キャリア検出設定	
コマンド書式	@SID, ch_1, ch_2, …, ch_31, ch_32 [↵]	
返り値書式	@SID, ch_1, ch_2, …, ch_31, ch_32 [↵]	
パラメータ	ch_1-32 : 入力チャンネルごとのキャリア検出設定 0 = キャリア検出なし ※初期値, 1 = キャリア検出あり	
実行例	送 @SID, 1, 1, …, 1, 1 [↵] 受 @SID, 1, 1, …, 1, 1 [↵]	全入力チャンネルをキャリア検出設定ありに設定する。 正常終了。
関連項目	@GID SDI 入力キャリア検出設定取得	
備考	デュアルリンク SDI モードの場合、チャンネル数は半分になります。 8 系統単位での入力数をカスタマイズした場合のチャンネル数は、カスタマイズした入力数になります。	

@GID	SDI 入力キャリア検出設定取得	
コマンド書式	@GID [↵]	
返り値書式	@GID, ch_1, ch_2, …, ch_31, ch_32 [↵]	
パラメータ	ch_1-32 : 入力チャンネルごとのキャリア検出設定 0 = キャリア検出なし ※初期値, 1 = キャリア検出あり	
実行例	送 @GID [↵] 受 @GID, 0, 0, …, 0, 0 [↵]	入力チャンネルのキャリア検出設定を取得する。 全入力チャンネルキャリア検出設定なし。
関連項目	@SID SDI 入力キャリア検出設定	
備考	デュアルリンク SDI モードの場合、チャンネル数は半分になります。 8 系統単位での入力数をカスタマイズした場合のチャンネル数は、カスタマイズした入力数になります。	

@GIC	SDI 入力キャリア検出値取得	
コマンド書式	@GIC [↵]	
返り値書式	@GIC, ch_1, ch_2, …, ch_31, ch_32 [↵]	
パラメータ	ch_1-32 : 入力チャンネルごとのキャリア検出値 0 = キャリア検出設定なし ※初期値, 1 = キャリア検出, 2 = キャリア非検出	
実行例	送 @GIC [↵] 受 @GIC, 1, 1, …, 1, 1 [↵]	入力チャンネルのキャリア検出情報を取得する。 全入力チャンネルキャリア検出。
関連項目	@GAI アラーム情報取得 @SID SDI 入力キャリア検出設定 @GID SDI 入力キャリア検出設定取得	
備考	デュアルリンク SDI モードの場合、チャンネル数は半分になります。 8 系統単位での入力数をカスタマイズした場合のチャンネル数は、カスタマイズした入力数になります。	

@SOD	SDI 出力キャリア検出設定	
コマンド書式	@SOD, <i>ch_1, ch_2, …, ch_31, ch_32</i> [↵]	
返り値書式	@SOD, <i>ch_1, ch_2, …, ch_31, ch_32</i> [↵]	
パラメータ	<i>ch_1-32</i> : 出力チャンネルごとのキャリア検出設定 0 = キャリア検出なし ※初期値, 1 = キャリア検出あり	
実行例	送 @SOD, 1, 1, …, 1, 1 [↵] 受 @SOD, 1, 1, …, 1, 1 [↵]	全出力チャンネルをキャリア検出設定ありに設定する。 正常終了。
関連項目	@GOD SDI 出力キャリア検出設定取得	
備考	デュアルリンク SDI モードの場合、チャンネル数は半分になります。 8 系統単位での出力数をカスタマイズした場合のチャンネル数は、カスタマイズした出力数になります。	

@GOD	SDI 出力キャリア検出設定取得	
コマンド書式	@GOD [↵]	
返り値書式	@GOD, <i>ch_1, ch_2, …, ch_31, ch_32</i> [↵]	
パラメータ	<i>ch_1-32</i> : 出力チャンネルごとのキャリア検出設定 0 = キャリア検出なし ※初期値, 1 = キャリア検出あり	
実行例	送 @GOD [↵] 受 @GOD, 0, 0, …, 0, 0 [↵]	出力チャンネルのキャリア検出設定を取得する。 全出力チャンネルキャリア検出設定なし。
関連項目	@SOD SDI 出力キャリア検出設定	
備考	デュアルリンク SDI モードの場合、チャンネル数は半分になります。 8 系統単位での出力数をカスタマイズした場合のチャンネル数は、カスタマイズした出力数になります。	

@GOC	SDI 出力キャリア検出値取得	
コマンド書式	@GOC [↵]	
返り値書式	@GOC, <i>ch_1, ch_2, …, ch_31, ch_32</i> [↵]	
パラメータ	<i>ch_1-32</i> : 出力チャンネルごとのキャリア検出値 0 = キャリア検出設定なし ※初期値, 1 = キャリア検出, 2 = キャリア非検出 ※出力チャンネルに入力 OFF が設定されている場合は「キャリア検出設定なし」を返します。	
実行例	送 @GOC [↵] 受 @GOC, 1, 1, …, 1, 1 [↵]	出力チャンネルのキャリア検出情報を取得する。 全出力チャンネルキャリア検出。
関連項目	@GAI アラーム情報取得 @SOD SDI 出力キャリア検出設定 @GOD SDI 出力キャリア検出設定取得	
備考	デュアルリンク SDI モードの場合、チャンネル数は半分になります。 8 系統単位での出力数をカスタマイズした場合のチャンネル数は、カスタマイズした出力数になります。	

@SRD	基準同期信号検出設定	
コマンド書式	@SRD, <i>ref</i> [↵]	
返り値書式	@SRD, <i>ref</i> [↵]	
パラメータ	<i>ref</i> : 基準同期信号検出設定 0 = 検出なし ※初期値, 1 = 検出あり	
実行例	送 @SRD, 1 [↵] 受 @SRD, 1 [↵]	基準同期信号検出設定ありに設定する。 正常終了。
関連項目	@GRD 基準同期信号検出設定取得	

@GRD	基準同期信号検出設定取得	
コマンド書式	@GRD [↵]	
返り値書式	@GRD, <i>ref</i> [↵]	
パラメータ	<i>ref</i> : 基準同期信号検出設定状態 0 = 検出なし ※初期値, 1 = 検出あり	
実行例	送 @GRD [↵] 受 @GRD, 0 [↵]	基準同期信号検出設定を取得する。 検出設定なし。
関連項目	@SRD 基準同期信号検出設定	

@GRC	基準同期信号検出値取得	
コマンド書式	@GRC [↵]	
返り値書式	@GRC, <i>ref</i> [↵]	
パラメータ	<i>ref</i> : 基準同期信号検出値 0 = 検出設定なし 1 = 基準同期信号検出 (ブラックバースト (BB) 信号検出) 2 = 基準同期信号検出 (3 値同期信号検出) 3 = 基準同期信号非検出	
実行例	送 @GRC [↵] 受 @GRC, 1 [↵]	基準同期信号検出値を取得する。 基準同期信号検出 (BB 検出)。
関連項目	@GAI アラーム情報取得 @SRD 基準同期信号検出設定 @GRD 基準同期信号検出設定取得	

@GFV	ファン回転数取得	
コマンド書式	@GFV [↵]	
返り値書式	@GFV, <i>fan</i> [↵]	
パラメータ	<i>fan</i> : ファン回転数 0 ~ xxxx	
実行例	送 @GFV [↵] 受 @GFV, 3100 [↵]	ファン回転数を取得する。 3100 回転
関連項目	@GAI アラーム情報取得	

3.3.7 SNMP 設定

@SNN	SNMP コミュニティ名の設定
コマンド書式	@SNN, name ↵
返り値書式	@SNN, name ↵
パラメータ	name : コミュニティ名 半角英数最大 15 文字まで設定可能です。 ※初期値 : public
実行例	送 @SNN, public ↵ コミュニティ名を設定する。 受 @SNN, public ↵ 正常終了。
関連項目	@GNN SNMP コミュニティ名の取得

@GNN	SNMP コミュニティ名の取得
コマンド書式	@GNN ↵
返り値書式	@GNN, name ↵
パラメータ	name : コミュニティ名 半角英数最大 15 文字まで取得可能です。 ※初期値 : public
実行例	送 @GNN ↵ コミュニティ名を取得する。 受 @GNN, public ↵ コミュニティ名 “public”。
関連項目	@SNN SNMP コミュニティ名の設定

@SNM	SNMP 有効/無効の設定
コマンド書式	@SNM, on/off ↵
返り値書式	@SNM, on/off ↵
パラメータ	on/off : SNMP 有効/無効 0 = 無効 ※初期値, 1 = 有効
実行例	送 @SNM, 1 ↵ SNMP を有効に設定する。 受 @SNM, 1 ↵ 正常終了。
関連項目	@GNN SNMP 有効/無効の取得

@GNN	SNMP 有効/無効の取得
コマンド書式	@GNN ↵
返り値書式	@GNN, on/off ↵
パラメータ	on/off : SNMP 有効/無効 0 = 無効 ※初期値, 1 = 有効
実行例	送 @GNN ↵ SNMP 有効/無効を取得する。 受 @GNN, 0 ↵ SNMP 無効状態。
関連項目	@SNM SNMP 有効/無効の設定

@STN	SNMP トラップ送信 コミュニティ名の設定	
コマンド書式	@STN, name [↵]	
返り値書式	@STN, name [↵]	
パラメータ	name : コミュニティ名 半角英数最大 15 文字まで設定可能です。 ※初期値 : public	
実行例	送 @STN, public [↵] 受 @STN, public [↵]	コミュニティ名を設定する。 正常終了。
関連項目	@GTN SNMP トラップ送信 コミュニティ名の取得	

@GTN	SNMP トラップ送信 コミュニティ名の取得	
コマンド書式	@GTN [↵]	
返り値書式	@GTN, name [↵]	
パラメータ	name : コミュニティ名 半角英数最大 15 文字まで取得可能です。 ※初期値 : public	
実行例	送 @GTN [↵] 受 @GTN, public [↵]	SNMP トラップコミュニティ名を取得する。 コミュニティ名 “public”。
関連項目	@STN SNMP トラップ送信 コミュニティ名の設定	

@STA	SNMP トラップ送信先 IP アドレス設定	
コマンド書式	@STA, id, unit_1, unit_2, unit_3, unit_4 [↵]	
返り値書式	@STA, id, unit_1, unit_2, unit_3, unit_4 [↵]	
パラメータ	id : トラップ送信先番号 1 = SNMP トラップ 1、 2 = SNMP トラップ 2 ----- unit_1 : IP アドレス上位 ~ unit_4 : IP アドレス下位 0 ~ 255 = 8 ビット(10 進数表記) ※初期値 0.0.0.0 ※0.0.0.0 は SNMP トラップを送信しない。	
実行例	送 @STA, 1, 192, 168, 3, 2 [↵] 受 @STA, 1, 192, 168, 3, 2 [↵]	SNMP トラップ 1 送信先の IP アドレスを 192.168.3.2 に設定する。 正常終了。
関連項目	@GTA SNMP トラップ送信先 IP アドレス取得	

@GTA	SNMP トラップ送信先 IP アドレス取得	
コマンド書式	@GTA, id [↵]	
返り値書式	@GTA, id, unit_1, unit_2, unit_3, unit_4 [↵]	
パラメータ	id : トラップ送信先番号 1 = SNMP トラップ 1、 2 = SNMP トラップ 2 ----- unit_1 : IP アドレス上位 ~ unit_4 : IP アドレス下位 0 ~ 255 = 8 ビット(10 進数表記) ※初期値 0.0.0.0	
実行例	送 @GTA, 1 [↵] 受 @GTA, 1, 192, 168, 3, 2 [↵]	SNMP トラップ 1 送信先の IP アドレスを取得する。 SNMP トラップ 1 送信先の IP アドレスは 192.168.3.2。
関連項目	@STA SNMP トラップ送信先 IP アドレス設定	

@SNT	SNMP トラップ送信 有効/無効の設定
コマンド書式	@SNT, <i>on/off</i> <input type="checkbox"/>
返り値書式	@SNT, <i>on/off</i> <input type="checkbox"/>
パラメータ	<i>on/off</i> : SNMP トラップ送信有効/無効 0 = 無効 ※初期値, 1 = 有効
実行例	送 @SNT, 1 <input type="checkbox"/> SNMP トラップ送信を有効に設定する。 受 @SNT, 1 <input type="checkbox"/> 正常終了。
関連項目	@GNT SNMP トラップ送信 有効/無効の取得

@GNT	SNMP トラップ送信 有効/無効の取得
コマンド書式	@GNT <input type="checkbox"/>
返り値書式	@GNT, <i>on/off</i> <input type="checkbox"/>
パラメータ	<i>on/off</i> : SNMP トラップ送信有効/無効 0 = 無効 ※初期値, 1 = 有効
実行例	送 @GNT <input type="checkbox"/> SNMP トラップ送信有効/無効を取得する。 受 @GNT, 0 <input type="checkbox"/> SNMP トラップ送信、無効状態。
関連項目	@SNT SNMP トラップ送信 有効/無効の設定

3.3.8 その他設定

@SLM	キーロックモードの設定	
コマンド書式	@SLM, in/out, menu, preset load ↵	
返り値書式	@SLM, in/out, menu, preset load ↵	
パラメータ	in/out : 入出力選択キー menu : メニューキー preset load : プリセットメモリ読み出しキー 0 = キーロック OFF (※初期値), 1 = キーロック ON	
実行例	送 @SLM, 1, 1, 1 ↵	全てをキーロック状態にする。
	受 @SLM, 1, 1, 1 ↵	正常終了。
関連項目	@GLM キーロックモードの取得	

@GLM	キーロックモードの取得	
コマンド書式	@GLM ↵	
返り値書式	@GLM, in/out, menu, preset load ↵	
パラメータ	in/out : 入出力選択キー menu : メニューキー preset load : プリセットメモリ読み出しキー 0 = キーロック OFF (※初期値), 1 = キーロック ON	
実行例	送 @GLM ↵	キーロック設定の状態を取得する。
	受 @GLM, 1, 1, 1 ↵	全てのキーがキーロック状態。
関連項目	@SLM キーロックモードの設定	

@SBZ	ブザー音設定	
コマンド書式	@SBZ, bz ↵	
返り値書式	@SBZ, bz ↵	
パラメータ	bz : ブザー音設定 0 = OFF, 1 = ON (※初期値)	
実行例	送 @SBZ, 1 ↵	ブザー音を ON に設定する。
	受 @SBZ, 1 ↵	正常終了。
関連項目	@GBZ ブザー音設定取得	

@GBZ	ブザー音設定取得	
コマンド書式	@GBZ ↵	
返り値書式	@GBZ, bz ↵	
パラメータ	bz : ブザー音設定 0 = OFF, 1 = ON (※初期値)	
実行例	送 @GBZ ↵	ブザー音設定状態を取得する。
	受 @GBZ, 1 ↵	ブザー音 ON。
関連項目	@SBZ ブザー音設定	

@GIV		バージョン情報取得	
コマンド書式	@GIV [↵]		
返り値書式	@GIV, <i>id, ver, in_ch, out_ch, mode</i> [↵]		
パラメータ	<i>id</i> : 製品型番 MDX-32SDI Series <hr/> <i>ver</i> : ファームウェアバージョン <hr/> <i>in_ch</i> : 入力チャンネル数 ※デュアルリンク SDI モードのときはチャンネル数が半分になります。 <hr/> <i>out_ch</i> : 出力チャンネル数 ※デュアルリンク SDI モードのときはチャンネル数が半分になります。 <hr/> <i>mode</i> : デュアルリンク SDI モード ON/OFF 0 = デュアルリンク SDI モード OFF 1 = デュアルリンク SDI モード ON		
実行例	送	@GIV[↵]	バージョン情報を取得する。
	受	@GIV, MDX-32SDI Series, 1.00, 32, 32, 0[↵]	返り値 (製品型番 MDX-32SDI Series, バージョン 1.00、入力チャンネル数 32、出力チャンネル数 32、デュアルリンク SDI モード OFF)。
備考	デュアルリンク SDI モードの場合、チャンネル数は半分になります。 8 系統単位での入出力数をカスタマイズした場合のチャンネル数は、カスタマイズした入出力数になります。		

@GDL		デュアルリンク SDI モード状態取得	
コマンド書式	@GDL [↵]		
返り値書式	@GDL, <i>mode</i> [↵]		
パラメータ	<i>mode</i> : デュアルリンク SDI モード状態 0 = デュアルリンク SDI モード OFF ※初期値 1 = デュアルリンク SDI モード ON		
実行例	送	@GDL[↵]	デュアルリンク SDI モード状態を取得する。
	受	@GDL, 0[↵]	デュアルリンク SDI モード OFF。

3.4 互換モード通信コマンド概要

互換モード通信コマンドは、コマンドを意味する半角英字(大文字あるいは小文字)またはパラメータ(半角数字)からなります。(コマンドによっては複数のパラメータを指定可能なものや、パラメータの必要ないものがあります。)

コマンドの最後にデリミタを送信することにより処理を実行します。

なお、無効なコマンドおよびパラメータを入力した場合、コマンドは破棄されます。

例：s,3 \square

「,」は、コマンドとパラメータおよびパラメータ間の区切り文字でカンマ(16進表記の2C)を表します。

\square は、デリミタ(CRで16進表記の0D)を表します。

互換モード通信コマンド操作を有効にするには、本機のメニュー操作から、「通信コマンドの形式設定」を「互換モード通信コマンド形式(OPTION)」(参照：ユーザーズガイド)に設定してください。

3.4.1 互換モード通信コマンドの例

① 単一のクロスポイントを切り換えるコマンド

パラメータをカンマで区切って送信します。

例：1,3 \square ←出力3に入力1を選択します。

例：q,3 \square ←出力3をOFFにします。

② 複数のクロスポイントを切り換えるコマンド

パラメータをカンマとセミコロンで区切って送信します。

例：1,3;2,4 \square ←出力3に入力1、出力4に入力2を選択します。

例：1,r \square ←全出力に入力1を選択します。

3.5 互換モード通信コマンドコマンド一覧

コマンド	機能	詳細ページ
なし	入出力チャンネル切換	36
w	入出力チャンネル取得	36
z	指定した出力の入出力チャンネル取得	37
t	プリセットメモリの読み出し(入出力チャンネル設定読み出し)	37
s	プリセットメモリの書き保存(入出力チャンネル設定保存)	37
y	プリセットメモリの設定取得(入出力チャンネル設定の取得)	37
mem	現在のクロスポイント状態と同一内容のプリセットメモリ番号取得	38
kl	フロントパネルキーロック	38
ku	フロントパネルキーロック解除	38
ky	フロントパネルキーロック状態取得	38
alarm	アラーム情報取得	39
i	SDI入力キャリア検出有効設定	39
j	SDI入力キャリア検出無効設定	39
m	SDI入力キャリア検出設定取得	40
u	SDI入力キャリア検出値取得	40
k	SDI出力キャリア検出有効設定	41
l	SDI出力キャリア検出無効設定	41
n	SDI出力キャリア検出設定取得	41
v	SDI出力キャリア検出値取得	42

3.6 互換モード通信コマンド詳細

なし	入出力チャンネル切換		
コマンド書式	<i>input_1, output_1</i> (; <i>input_2, output_2</i> …) <input type="checkbox"/>		
返り値書式	なし		
パラメータ	<i>input_1-32</i> : 入力チャンネル q = OFF ※初期値, 1~32 = INPUT1~32 <hr/> <i>output_1-32</i> : 出力チャンネル r = 全出力, 1~32 = OUTPUT1~32		
実行例	送	1, 1 <input type="checkbox"/>	出力 1 に入力 1 を選択する。
	受	q, 001 <input type="checkbox"/>	出力 1 を OFF にする。001 や 01 でも動作可能。
	送	1, r <input type="checkbox"/>	全出力に入力 1 を選択する。
	受	4, 1; 3, 2; <input type="checkbox"/>	セミコロン";"で連結し複数のチャンネルを一括で変更します。
備考	デュアルリンク SDI モードの場合、チャンネル数は半分になります。 8 系統単位での入出力数をカスタマイズした場合のチャンネル数は、カスタマイズした入出力数になります。		

w	入出力チャンネル取得		
コマンド書式	w <input type="checkbox"/>		
返り値書式	<i>ch_1; ch_2; … ; ch_31; ch_32</i> <input type="checkbox"/>		
パラメータ	<i>ch_1-32</i> : 入力チャンネル 000 = OFF ※初期値, 001~032 = INPUT1~32		
実行例	送	w <input type="checkbox"/>	入出力チャンネル取得。
	受	001; 002; … ; 031; 032 <input type="checkbox"/>	
備考	デュアルリンク SDI モードの場合、チャンネル数は半分になります。 8 系統単位での入出力数をカスタマイズした場合のチャンネル数は、カスタマイズした入出力数になります。		

z	指定した出力の入出力チャンネル取得	
コマンド書式	z output []	
返り値書式	input []	
パラメータ	<i>output</i> : 出力チャンネル 1~32 = OUTPUT1~32 <hr/> <i>input</i> : 入力チャンネル 000 = OFF ※初期値, 001~032 = INPUT1~32	
実行例	送 z1 [] 受 001 [] <hr/> 送 z001 [] 受 001 []	出力1に選択されている入力を取得。 入力1が選択されている。 <hr/> z001 や z01 でも動作可能。
備考	デュアルリンク SDI モードの場合、チャンネル数は半分になります。 8 系統単位での入出力数をカスタマイズした場合のチャンネル数は、カスタマイズした入出力数になります。	

t	プリセットメモリの読み出し（入出力チャンネル設定読み出し）	
コマンド書式	t, preset []	
返り値書式	なし	
パラメータ	<i>preset</i> : プリセットメモリ 1~32 = PRESET1~32	
実行例	送 t, 1 [] 受 なし	プリセットメモリ 1 の読み出し

s	プリセットメモリの上書き保存（入出力チャンネル設定保存）	
コマンド書式	s, preset []	
返り値書式	なし	
パラメータ	<i>preset</i> : プリセットメモリ 1~32 = PRESET1~32	
実行例	送 s, 1 [] 受 なし	プリセットメモリ 1 の上書き保存

y	プリセットメモリの設定取得（入出力チャンネル設定の取得）	
コマンド書式	y, preset []	
返り値書式	output_1 ; output_2 ; ... ; output_31 ; output_32 []	
パラメータ	<i>preset</i> : プリセットメモリ 1~32 = PRESET1~32 <hr/> <i>output_1-32</i> : 出力チャンネル 000 = OFF, 001~032 = INPUT1~32, -01 = 制御しない ※初期値	
実行例	送 y, 1 [] 受 001;002; ... ;031;032 []	プリセットメモリ 1 の設定取得
備考	デュアルリンク SDI モードの場合、チャンネル数は半分になります。 8 系統単位での入出力数をカスタマイズした場合のチャンネル数は、カスタマイズした入出力数になります。	

mem	現在のクロスポイント状態と同一内容のプリセットメモリ番号取得	
コマンド書式	mem []	
返り値書式	preset []	
パラメータ	preset : プリセットメモリ 000 = なし 001~032 = クロスポイントと一致するプリセットメモリ	
実行例	送 mem []	コマンド送信。
	受 000 []	クロスポイントと一致するプリセットメモリがありません。
	送 mem []	コマンド送信。
	受 001 []	クロスポイントとプリセットメモリ 1 のクロスポイントが一致。
備考	複数のメモリ番号が一致する場合、小さい方の番号を返します。	

kl	フロントパネルキーロック	
コマンド書式	kl [] ※ l = 小文字のL	
返り値書式	なし	
パラメータ	なし	

ku	フロントパネルキーロック解除	
コマンド書式	ku []	
返り値書式	なし	
パラメータ	なし	

ky	フロントパネルキーロック状態取得	
コマンド書式	ky []	
返り値書式	lock []	
パラメータ	lock : キーロック状態 FREE = キーロックなし ※初期値 LOCK = キーロック中	
実行例	送 ky []	コマンド送信。
	受 FREE []	状態取得 (キーロックなし)。
	送 ky []	コマンド送信。
	受 LOCK []	状態取得 (キーロック中)。

alarm	アラーム情報取得	
コマンド書式	alarm []	
返り値書式	PA=status; SDI=status; REF=status ; FAN=status []	
パラメータ	PA : 電源 1 および電源 2 のアラーム状態 ok = 正常, ng = 異常 (アラームあり) SDI : SDI 入力キャリア検出および SDI 出力キャリア検出アラーム状態 off = 検出設定なし, ok = 正常, ng = 異常 (アラームあり) REF : 基準同期信号検出アラーム状態 off = 検出設定なし, ok = 正常, ng = 異常 (アラームあり) FAN : ファンのアラーム状態 ok = 正常, ng = 異常 (アラームあり)	
実行例	送 alarm[] 受 PA=ok;SDI=ok;REF=ok;FAN=ok[]	コマンド送信。 アラーム異常検出なし。

i	SDI 入力キャリア検出有効設定	
コマンド書式	i , input []	
返り値書式	なし	
パラメータ	input : 入力チャンネルのキャリア検出有効設定 r = 全入力, 1~32 = INPUT1~32	
実行例	送 i,1[] 受	入力 1 のキャリア検出する。
	送 i,001[] 受	入力 1 のキャリア検出する。001 や 01 でも動作可能。
備考	デュアルリンク SDI モードの場合、チャンネル数は半分になります。 8 系統単位での入力数をカスタマイズした場合のチャンネル数は、カスタマイズした入力数になります。	

j	SDI 入力キャリア検出無効設定	
コマンド書式	j , input []	
返り値書式	なし	
パラメータ	input : 入力チャンネルのキャリア検出無効設定 r = 全入力, 1~32 = INPUT1~32	
実行例	送 j,1[] 受	入力 1 のキャリア検出を解除する。
	送 j,001[] 受	入力 1 のキャリア検出を解除する。001 や 01 でも動作可能。
備考	デュアルリンク SDI モードの場合、チャンネル数は半分になります。 8 系統単位での入力数をカスタマイズした場合のチャンネル数は、カスタマイズした入力数になります。	

m	SDI 入力キャリア検出設定取得	
コマンド書式	m [↵]	
返り値書式	cd_1;cd_2; … ;cd_31;cd_32 [↵]	
パラメータ	cd_1-32 : 入力チャンネルごとの設定値 0 = キャリア検出しない ※初期値, 1 = キャリア検出する	
実行例	送 m [↵] 受 1;1; … ;1;1 [↵]	入力チャンネルのキャリア検出設定を取得する。 全入力チャンネルキャリア検出設定あり。
備考	デュアルリンク SDI モードの場合、チャンネル数は半分になります。 8 系統単位での入力数をカスタマイズした場合のチャンネル数は、カスタマイズした入力数になります。	

u	SDI 入力キャリア検出値取得	
コマンド書式	u [↵]	
返り値書式	err_1;err_2; … ;err_31;err_32 [↵]	
パラメータ	err_1-32 : 入力チャンネルごとのキャリア検出値 0 = エラーなし, 1 = エラーあり	
実行例	送 u [↵] 受 0;0; … ;0;0 [↵]	入力チャンネルのキャリア検出情報を取得する。 全入力チャンネルキャリア検出。
備考	デュアルリンク SDI モードの場合、チャンネル数は半分になります。 8 系統単位での入力数をカスタマイズした場合のチャンネル数は、カスタマイズした入力数になります。 アラーム出力設定の無効/有効に関係なく、検出した結果を取得します。	

k	SDI 出力キャリア検出有効設定	
コマンド書式	k, output <input type="checkbox"/>	
返り値書式	なし	
パラメータ	output : 出力チャンネルのキャリア検出有効設定 r = 全出力, 1~32 = OUTPUT1~32	
実行例	送	k, 1 <input type="checkbox"/>
	受	
実行例	送	k, 001 <input type="checkbox"/>
	受	
備考	デュアルリンク SDI モードの場合、チャンネル数は半分になります。 8 系統単位での出力数をカスタマイズした場合のチャンネル数は、カスタマイズした出力数になります。	

l	SDI 出力キャリア検出無効設定	
コマンド書式	l, output <input type="checkbox"/> ※ l = 小文字の L	
返り値書式	なし	
パラメータ	output : 出力チャンネルのキャリア検出無効設定 r = 全出力, 1~32 = OUTPUT1~32	
実行例	送	l, 1 <input type="checkbox"/>
	受	
実行例	送	l, 001 <input type="checkbox"/>
	受	
備考	デュアルリンク SDI モードの場合、チャンネル数は半分になります。 8 系統単位での出力数をカスタマイズした場合のチャンネル数は、カスタマイズした出力数になります。	

n	SDI 出力キャリア検出設定取得	
コマンド書式	n <input type="checkbox"/>	
返り値書式	cd_1; cd_2; ...; cd_31; cd_32 <input type="checkbox"/>	
パラメータ	cd_1-32 : 出力チャンネルごとの設定値 0 = キャリア検出しない ※初期値, 1 = キャリア検出する	
実行例	送	n <input type="checkbox"/>
	受	0:0; ... ;0:0 <input type="checkbox"/>
実行例	送	n <input type="checkbox"/>
	受	0:0; ... ;0:0 <input type="checkbox"/>
備考	デュアルリンク SDI モードの場合、チャンネル数は半分になります。 8 系統単位での出力数をカスタマイズした場合のチャンネル数は、カスタマイズした出力数になります。	

v	SDI 出力キャリア検出値取得
コマンド書式	v [↵]
返り値書式	err_1;err_2;…;err_31;err_32 [↵]
パラメータ	err_1-32 : 出力チャンネルごとのキャリア検出値 0 = エラーなし, 1 = エラーあり
実行例	送 v[↵] 出力チャンネルのキャリア検出情報を取得する。 受 0;0;…;0;0[↵] 全出力チャンネルキャリア検出。
備考	デュアルリンク SDI モードの場合、チャンネル数は半分になります。 8 系統単位での入力数をカスタマイズした場合のチャンネル数は、カスタマイズした入力数になります。 アラーム出力設定の無効/有効に関係なく、検出した結果を取得します。

4 ASCII コード表

[表 4.1] ASCII コード表(1/2)

文字	16 進	文字	16 進	文字	16 進	文字	16 進
NUL	00	SP	20	@	40	`	60
SOH	01	!	21	A	41	a	61
STX	02	"	22	B	42	b	62
ETX	03	#	23	C	43	c	63
EOT	04	\$	24	D	44	d	64
ENQ	05	%	25	E	45	e	65
ACK	06	&	26	F	46	f	66
BEL	07	'	27	G	47	g	67
BS	08	(28	H	48	h	68
HT	09)	29	I	49	i	69
LF	0A	*	2A	J	4A	j	6A
VT	0B	+	2B	K	4B	k	6B
FF	0C	,	2C	L	4C	l	6C
CR	0D	-	2D	M	4D	m	6D
SO	0E	.	2E	N	4E	n	6E
SI	0F	/	2F	O	4F	o	6F
DLE	10	0	30	P	50	p	70
DC1	11	1	31	Q	51	q	71
DC2	12	2	32	R	52	r	72
DC3	13	3	33	S	53	s	73
DC4	14	4	34	T	54	t	74
NAK	15	5	35	U	55	u	75
SYN	16	6	36	V	56	v	76
ETB	17	7	37	W	57	w	77
CAN	18	8	38	X	58	x	78
EM	19	9	39	Y	59	y	79
SUB	1A	:	3A	Z	5A	z	7A
ESC	1B	;	3B	[5B	{	7B
FS	1C	<	3C	¥	5C		7C
GS	1D	=	3D]	5D	}	7D
RS	1E	>	3E	^	5E	~	7E
US	1F	?	3F	_	5F	DEL	7F

[表 4.2] ASCII コード表(2/2)

文字	16 進	コントロールコード詳細
<i>NUL</i>	00	NULI(ヌル)
<i>SOH</i>	01	Start Of Heading(ヘッダ開始)
<i>STX</i>	02	Start of TeXt(テキスト開始)
<i>ETX</i>	03	End of TeXt(テキスト終了)
<i>EOT</i>	04	End Of Transmission(転送終了)
<i>ENQ</i>	05	ENQuiry(問合せ)
<i>ACK</i>	06	ACknowledge(肯定応答)
<i>BEL</i>	07	BELI(ベル)
<i>BS</i>	08	Back Space(後退)
<i>HT</i>	09	Horizontal Tabulation(水平タブ)
<i>LF</i>	0A	Line Feed(改行)
<i>VT</i>	0B	Vertical Tabulation(垂直タブ)
<i>FF</i>	0C	Form Feed(改ページ)
<i>CR</i>	0D	Carriage Return(復帰)
<i>SO</i>	0E	Shift Out(シフトアウト)
<i>SI</i>	0F	Shift In(シフトイン)
<i>DLE</i>	10	Data Link Escape(伝送制御拡張)
<i>DC1</i>	11	Device Control 1(装置制御 1)
<i>DC2</i>	12	Device Control 2(装置制御 2)
<i>DC3</i>	13	Device Control 3(装置制御 3)
<i>DC4</i>	14	Device Control 4(装置制御 4)
<i>NAK</i>	15	Negative AcKnowledge(否定応答)
<i>SYN</i>	16	SYNchronous idle(同期信号)
<i>ETB</i>	17	End of Transmission Block(転送ブロック終了)
<i>CAN</i>	18	CANcel(取消)
<i>EM</i>	19	End of Medium(媒体終端)
<i>SUB</i>	1A	SUBstitute(置換)
<i>ESC</i>	1B	ESCape(拡張)
<i>FS</i>	1C	File Separator(ファイル分離)
<i>GS</i>	1D	Group Separator(グループ分離)
<i>RS</i>	1E	Record Separator(レコード分離)
<i>US</i>	1F	Unit Separator(ユニット分離)
<i>SP</i>	20	SPace(空白)
<i>DEL</i>	7F	DELete(削除)

MDX-3232SDI 取扱説明書 <コマンドガイド>

Ver.1.0.0

発行日 2014 年 05 月 07 日



株式会社 アイ・ディ・ケイ

本 社 〒242-0021 神奈川県大和市中央 7-9-1
TEL (046) 200-0764 FAX (046) 200-0765

関西営業所 〒564-0063 大阪府吹田市江坂町 1-23-5 大同生命江坂第 2 ビル 5 階
TEL (06) 6192-0764 FAX (06) 6192-0906

九州営業所 〒812-0011 福岡県福岡市博多区博多駅前 4-9-2 八百治センタービル 3 階
TEL (092) 431-0764 FAX (092) 431-0906

E メールアドレス info@idk.co.jp **ホームページ** <http://www.idk.co.jp/>