

MD1260A
40/100G イーサネット アナライザ
リモートコントロール
取扱説明書

第6版

- ・製品を適切・安全にご使用いただくために、製品をご使用になる前に、本書を必ずお読みください。
- ・本書に記載以外の各種注意事項は、『MD1260A 40/100G イーサネット アナライザ 取扱説明書』に記載の事項に準じますので、そちらをお読みください。
- ・本書は製品とともに保管してください。

アンリツ株式会社

安全情報の表示について

当社では人身事故や財産の損害を避けるために、危険の程度に応じて下記のようなシグナルワードを用いて安全に関する情報を提供しています。記述内容を十分理解して機器を操作するようにしてください。

下記の表示およびシンボルは、そのすべてが本器に使用されているとは限りません。また、外観図などが本書に含まれるとき、製品に貼り付けたラベルなどがその図に記入されていない場合があります。

本書中の表示について



危険

回避しなければ、死亡または重傷に至る切迫した危険状況があることを警告しています。



警告

回避しなければ、死亡または重傷に至る恐れがある潜在的危険について警告しています。



注意

回避しなければ、軽度または中程度の人体の傷害に至る恐れがある潜在的危険、または、物的損害の発生のみが予測されるような危険状況について警告しています。

機器に表示または本書に使用されるシンボルについて

機器の内部や操作箇所の近くに、または本書に、安全上または操作上の注意を喚起するための表示があります。

これらの表示に使用しているシンボルの意味についても十分理解して、注意に従ってください。



禁止行為を示します。丸の中や近くに禁止内容が描かれています。



守るべき義務的行為を示します。丸の中や近くに守るべき内容が描かれています。



警告や注意を喚起することを示します。三角の中や近くにその内容が描かれています。



注意すべきことを示します。四角の中にその内容が書かれています。



このマークを付けた部品がリサイクル可能であることを示しています。

MD1260A

40/100G イーサネット アナライザ

リモートコントロール 取扱説明書

2010年（平成22年）7月14日（初 版）

2012年（平成24年）12月19日（第6版）

- ・予告なしに本書の内容を変更することがあります。
- ・許可なしに本書の一部または全部を転載・複製することを禁じます。

Copyright © 2010-2012, ANRITSU CORPORATION

Printed in Japan

国外持出しに関する注意

1. 本製品は日本国内仕様であり、外国の安全規格などに準拠していない場合もありますので、国外へ持ち出して使用された場合、当社は一切の責任を負いかねます。
2. 本製品および添付マニュアル類は、輸出および国外持ち出しの際には、「外国為替及び外国貿易法」により、日本国政府の輸出許可や役務取引許可を必要とする場合があります。また、米国の「輸出管理規則」により、日本からの再輸出には米国政府の再輸出許可を必要とする場合があります。

本製品や添付マニュアル類を輸出または国外持ち出しする場合は、事前に必ず当社の営業担当までご連絡ください。

輸出規制を受ける製品やマニュアル類を廃棄処分する場合は、軍事用途等に不正使用されないように、破碎または裁断処理していただきますようお願い致します。

はじめに

MD1260A 40/100G イーサネット アナライザ（以下、本器）には3冊の取扱説明書があります。

MD1260A 40/100G イーサネット アナライザ
取扱説明書

本器の設置方法と取扱上の注意、コネクタの接続方法、パネル操作、保守、仕様、各種機能を説明します。

MD1260A 40/100G イーサネット アナライザ
リモートコントロール取扱説明書（本書）

本器をリモート制御するためのコマンドの使用方を説明します。

MD1260A 40/100G イーサネット アナライザ
Add-on 機能 取扱説明書

本器の Add-on 機能の使用方を説明します。

この説明書は、本器をリモート制御する方法について記述したものです。電源や周辺機器の接続、パネル操作および保守については、『MD1260A 40/100G イーサネット アナライザ 取扱説明書』（M-W3395AW）をご利用ください。

本説明書は、次の読者を想定しています。

- ・ 『MD1260A 40/100G イーサネット アナライザ 取扱説明書』を一読している。
- ・ プログラム作成の経験がある。

目次

安全情報の表示について	ii
はじめに	I
第 1 章 概要	1-1
1.1 リモート制御の紹介	1-2
1.2 用途	1-3
1.3 用語	1-4
第 2 章 ご使用になる前に	2-1
2.1 設備を準備する	2-2
2.2 機器を接続する	2-3
2.3 インタフェースを設定する	2-5
2.4 接続を確認する	2-9
2.5 メッセージの書式	2-15
2.6 機器の状態を調べる	2-19
2.7 メッセージの実行状態を確認する	2-23
2.8 マルチポート機能動作時のリモート制御	2-24
第 3 章 メッセージの詳細	3-1
3.1 メッセージの記述方法	3-2
3.2 パネル操作とメッセージの対応	3-4
3.3 メッセージの説明	3-41
第 4 章 操作記録機能	4-1
4.1 操作記録機能の概要	4-2
4.2 操作記録機能の開始と停止	4-3

付録A	測定項目一覧.....	A-1
付録B	エラーコード.....	B-1
付録C	コマンドの変更情報.....	C-1
付録D	コマンドのアプリケーション対応表	D-1
付録E	サンプルプログラム	E-1
索引		索引-1

1
2
3
4
付録
索引

コマンド目次

*CLS [Clear Status]	3-42
*ESE [Event Status Enable]	3-43
*ESR [Standard Event Status Register]	3-44
*IDN [Identification]	3-44
*OPC [Operation Complete]	3-45
*RST [Reset]	3-46
*SRE [Service Request Enable]	3-46
*STB [Status Byte]	3-47
*TRG [Trigger]	3-47
*WAI [Wait to Continue]	3-47
:CALCulate:APS:ERRor:FREE	3-48
:CALCulate:APS:START[:AUNit]	3-48
:CALCulate:APS:START:EVENT[:AUNit]	3-49
:CALCulate:APS:STATus[:AUNit]	3-49
:CALCulate:APS:STOP[:AUNit]	3-50
:CALCulate:APS:TRIGger	3-50
:CALCulate:CAPTure:DATA	3-51
:CALCulate:CAPTure:DATA:RXC	3-51
:CALCulate:CAPTure:DATA:RXD	3-52
:CALCulate:CAPTure:DATA:SIZE	3-53
:CALCulate:CAPTure:DATA:TRIGger:POSition	3-53
:CALCulate:CAPTure:START	3-54
:CALCulate:CAPTure:START:EVENT	3-54
:CALCulate:CAPTure:STATus	3-55
:CALCulate:CAPTure:STOP	3-55
:CALCulate:CAPTure:TRIGger	3-56
:CALCulate:CAPTure:TRIGger:MANual	3-56
:CALCulate:CAPTure:TRIGger:POSition	3-56
:CALCulate:CAPTure:TRIGger:TYPE	3-57
:CALCulate:CAPTure:TYPE	3-58
:CALCulate:COUNter:ETHer:OVERsize	3-59
:CALCulate:COUNter:FLOW:FIELD	3-59
:CALCulate:COUNter:FLOW:FIELD:ID	3-62
:CALCulate:COUNter:FLOW:FIELD:Nfid	3-63
:CALCulate:COUNter:FLOW:TYPE	3-63
:CALCulate:COUNter:GAP	3-64
:CALCulate:COUNter:SERRor	3-64
:CALCulate:COUNter:START[:AUNit]	3-65
:CALCulate:COUNter:START:EVENT[:AUNit]	3-65
:CALCulate:COUNter:STATus[:AUNit]	3-65
:CALCulate:COUNter:STOP[:AUNit]	3-67
:CALCulate:DATA	3-67
:CALCulate:DATA:TYPE	3-68

:CALCulate:DELay:START[:AUNit]	3-68
:CALCulate:DELay:START:EVENT[:AUNit]	3-69
:CALCulate:DELay:STATus[:AUNit]	3-69
:CALCulate:DELay:STOP[:AUNit]	3-70
:CALCulate:EALarm[:AUNit]	3-71
:CALCulate:MONitor:OTU:COLUMN	3-71
:CALCulate:MONitor:OTU:DATA	3-72
:CALCulate:MONitor:OTU:TYPE	3-74
:CALCulate:TRIGger:CONDition	3-75
:MDIO:READ	3-76
:MDIO:WRITe	3-76
:MMEMory:CATalog	3-76
:MMEMory:INITialize	3-78
:MMEMory:LOG:FNAME	3-78
:MMEMory:LOG:ITEM	3-79
:MMEMory:LOG:PREFIX	3-79
:MMEMory:LOG:START	3-80
:MMEMory:LOG:STATus	3-80
:MMEMory:LOG:STOP	3-80
:MMEMory:LOG:TIMing	3-81
:MMEMory:RECall[:AUNit]	3-81
:MMEMory:STORe[:AUNit]	3-82
:MMEMory:STORe:CAPTure	3-83
:MMEMory:STORe:CAPTure:ITEM	3-84
:MMEMory:STReam:RECall	3-84
:PROTOcol:ARPNa:REPLy:ENABLE	3-85
:PROTOcol:GARPns:DURation:TYPE	3-85
:PROTOcol:GARPns:ENABLE	3-86
:PROTOcol:GARPns:INTERval	3-86
:PROTOcol:GARPns:START	3-87
:PROTOcol:GARPns:STATus	3-87
:PROTOcol:GARPns:STOP	3-87
:PROTOcol:GARPns:TYPE	3-88
:PROTOcol:PING:COUNT	3-88
:PROTOcol:PING:DST:IPV4	3-89
:PROTOcol:PING:DST:IPV6	3-89
:PROTOcol:PING:DST:MAC	3-90
:PROTOcol:PING:IPMode	3-90
:PROTOcol:PING:MACResolve	3-91
:PROTOcol:PING:PAYLoad	3-91

1

2

3

4

付録

索引

:PROTOcol:PING:PSIZE:TYPE	3-92
:PROTOcol:PING:PSIZE:VALUE	3-93
:PROTOcol:PING:REPLY:ENABLE	3-93
:PROTOcol:PING:RESULT	3-94
:PROTOcol:PING:SRC:IPV4	3-94
:PROTOcol:PING:SRC:IPV6	3-95
:PROTOcol:PING:SRC:MAC	3-95
:PROTOcol:PING:START	3-96
:PROTOcol:PING:START:EVENT	3-96
:PROTOcol:PING:STATUS	3-97
:PROTOcol:PING:STOP	3-97
:PROTOcol:PING:TIMEout	3-98
:PROTOcol:PING:VLAN	3-98
:ROUTE:BERT	3-99
:ROUTE:ETHER:NEGOTIATION:AUTO	3-99
:ROUTE:FCONTROL	3-100
:ROUTE:LFS:REPLY	3-100
:ROUTE:MODE	3-100
:ROUTE:MODE:THROUGH:OVERWRITE:RANGE	3-101
:ROUTE:MODE:THROUGH:TYPE	3-102
:ROUTE:MPLStp:CWORD	3-102
:ROUTE:VLAN:NUM	3-103
:ROUTE:VLAN:VALUE	3-103
:SENSE:MAPPING	3-104
:SENSE:ODTU:MAIN:DETECT	3-104
:SENSE:ODTU:MAIN:TS	3-105
:SENSE:OTN:FEC	3-106
:SENSE:PLM[:L]:PATTERN	3-106
:SENSE:TIM:{SM PM[:L] TCM{1 2 3 4 5 6}}	3-107
:SENSE:TIM:{SM PM[:L] TCM{1 2 3 4 5 6}}:PATTERN:{DAPI SAPI}	3-108
:SENSE:TPATTERN:INVERT	3-109
:SENSE:TPATTERN:TYPE	3-110
:SENSE:TPATTERN:WORD	3-111
:SENSE:TRANSCEIVER:EQUALIZER:CONTROL	3-111
:SENSE:TRANSCEIVER:EQUALIZER:DCGAIN	3-112
:SOURCE:CFP:OPTICAL:OFF	3-112
:SOURCE:CFP:OPTICAL:ON	3-112
:SOURCE:CFP:OPTICAL:STATUS	3-113
:SOURCE:CLOCK	3-114
:SOURCE:CLOCK:FREQUENCY:OFFSET	3-114
:SOURCE:CLOCK:OUTPUT:DIVIDE	3-115
:SOURCE:CLOCK:OUTPUT:M10	3-115
:SOURCE:CLOCK:PAYLOAD:OFFSET[:L]	3-116

1
2
3
4
付録
索引

:SOURce:EALarm:BIT	3-116
:SOURce:EALarm:FAS:EXCLude	3-117
:SOURce:EALarm:LANE.....	3-118
:SOURce:EALarm:START[:AUNit]	3-118
:SOURce:EALarm:START:EVENT[:AUNit]	3-119
:SOURce:EALarm:STATus[:AUNit].....	3-119
:SOURce:EALarm:STOP[:AUNit].....	3-120
:SOURce:EALarm:SUBRow	3-120
:SOURce:EALarm:TIMing:BURSt	3-121
:SOURce:EALarm:TIMing:ERRor	3-121
:SOURce:EALarm:TIMing:NORMal	3-122
:SOURce:EALarm:TIMing:RATE	3-123
:SOURce:EALarm:TIMing:TYPE.....	3-124
:SOURce:EALarm:TYPE.....	3-125
:SOURce:GFP:CSF:REPLacement	3-128
:SOURce:GFP:{PTI UPI}.....	3-129
:SOURce:MAPPing	3-129
:SOURce:MAPPing:LANE.....	3-131
:SOURce:ODTU:DUMMy:PATtern	3-132
:SOURce:ODTU:MAIN:TP	3-132
:SOURce:ODTU:MAIN:TS	3-133
:SOURce:OTN:FEC	3-133
:SOURce:OTN:OH[:L]	3-134
:SOURce:SKEW:BIT	3-134
:SOURce:SKEW:LANE	3-135
:SOURce:SKEW:NS	3-136
:SOURce:SKEW:TYPE	3-136
:SOURce:STReam:BURSt:CONTRol:VALue.....	3-137
:SOURce:STReam:BURSt:ENABle	3-137
:SOURce:STReam:BURSt:SIZE	3-137
:SOURce:STReam:CONTRol:RANGe	3-138
:SOURce:STReam:CONTRol:TYPE	3-138
:SOURce:STReam:CONTRol:VALue.....	3-139
:SOURce:STReam:COUNt	3-139
:SOURce:STReam:DURation:FRAMes	3-140
:SOURce:STReam:DURation:REPeat:COUNt	3-140
:SOURce:STReam:DURation:TYPE.....	3-141
:SOURce:STReam:ENABle	3-142
:SOURce:STReam:ERRor:TYPE.....	3-142
:SOURce:STReam:FSIZe	3-143
:SOURce:STReam:FSIZe:RANGe.....	3-143
:SOURce:STReam:FSIZe:TYPE.....	3-144
:SOURce:STReam:FSIZe:VALue	3-144

:SOURCE:STReam:HEADer.....	3-145
:SOURCE:STReam:HEADer:ETHer:DA	3-147
:SOURCE:STReam:HEADer:ETHer:SA	3-148
:SOURCE:STReam:HEADer:ETHer:TYPE.....	3-148
:SOURCE:STReam:HEADer:PATtern	3-149
:SOURCE:STReam:HEADer:VARiable1:RANGe	3-150
:SOURCE:STReam:HEADer:VARiable1:TYPE	3-151
:SOURCE:STReam:HEADer:VARiable2:RANGe	3-152
:SOURCE:STReam:HEADer:VARiable2:TYPE	3-153
:SOURCE:STReam:HEADer:VARiable{3 4 5}	3-153
:SOURCE:STReam:ID	3-157
:SOURCE:STReam:RESolve:IP:TARGet	3-158
:SOURCE:STReam:RESolve:IPV4:ROUTer	3-158
:SOURCE:STReam:RESolve:IPV6:ROUTer	3-159
:SOURCE:STReam:RESolve:MAC:RETRy	3-160
:SOURCE:STReam:RESolve:MAC:TIMEout	3-160
:SOURCE:STReam:RESolve:PING:TRY	3-161
:SOURCE:STReam:RESolve:PING:PAYLoad.....	3-161
:SOURCE:STReam:RESolve:PING:TIMEout.....	3-162
:SOURCE:STReam:RESolve:RESult	3-162
:SOURCE:STReam:RESolve:STARt	3-163
:SOURCE:STReam:RESolve:STARt:EVENT	3-163
:SOURCE:STReam:RESolve:STATus.....	3-164
:SOURCE:STReam:RESolve:STOP	3-164
:SOURCE:STReam:RESolve:TYPE	3-165
:SOURCE:STReam:STARt[:AUNit].....	3-166
:SOURCE:STReam:STARt:EVENT[:AUNit].....	3-166
:SOURCE:STReam:STATus[:AUNit]	3-167
:SOURCE:STReam:STOP[:AUNit].....	3-167
:SOURCE:STReam:TFRame:ENABle	3-168
:SOURCE:STReam:TFRame:FID	3-168
:SOURCE:STReam:TYPE	3-169
:SOURCE:TPATtern:INVert	3-169
:SOURCE:TPATtern:TYPE	3-170
:SOURCE:TPATtern:WORD	3-171
:SOURCE:TRANsceiver:EMPHasis:FIRSt.....	3-171
:SOURCE:TRANsceiver:EMPHasis:PRE	3-172
:SOURCE:TRANsceiver:EMPHasis:SECond	3-172
:SOURCE:TRANsceiver:VOD	3-172
:SYSTem:CONFig	3-173
:SYSTem:DATE	3-175
:SYSTem:ERRor	3-175
:SYSTem:PRINt:COPI	3-177
:SYSTem:STATus.....	3-177
:SYSTem:TERMination	3-177
:SYSTem:TIME	3-178

:SYSTem:VERSion	3-178
:UENTry:ID	3-179
:UENTry:LIST	3-179

1
2
3
4
付録
索引

ここでは, リモート制御の用途と本書で使用する用語について説明します。

1.1 リモート制御の紹介 1-2

1.2 用途 1-3

1.3 用語 1-4

1.1 リモート制御の紹介

リモート制御は、制御用コンピュータから通信インタフェースを介して測定器に命令を与え、測定器の設定をしたり、測定結果や測定器の状態を読み取ったりする機能です。

本器の通信インタフェースは、イーサネットを標準で使用します。オプションを追加すると、**GPIO**を使用できます。

どちらのインタフェースを使用する場合でも、本器を別の機器と識別するための番号を設定します。この番号は、イーサネットを使用するときは **IP アドレス**、**GPIO**を使用するときは **GPIO アドレス**です。

本器を制御する文字列をコマンドと呼びます。コマンドはアスキー(ASCII)文字列で構成されます。たとえば、**Counter** 測定を開始するときのコマンドは次のとおりです。

:CALCulate:COUNTER:START

本器からデータを読み取るときのコマンドをクエリと呼びます。クエリは文字列の最後に?(クエスチョンマーク)をつけます。たとえば、本器が受信しているクロック周波数の値を読み取るときは、次のクエリを送信します。

:CALCulate:DATA? RX_FREQ

制御側コンピュータは、本器から次の文字列を受信したとします。

103125000000

このとき、受信クロック周波数は 103,125,000,000 Hz であることがわかります。

本器がリモート制御されると、アプリケーションの操作画面がロックされた状態になります。このときに操作できるキーは、電源スイッチとシステムメニューの **[Local/Panel Unlock]** だけになります。この状態をリモート状態と呼びます。リモート状態を解除するには、システムメニューの **[Local/Panel Unlock]** をタッチします。

1.2 用途

リモート制御の主な用途は次のとおりです。

測定の自動化

タッチパネルを操作するかわりに、プログラムを実行して測定器を制御します。測定器の制御手順をプログラムで記述することにより、測定を自動化することができます。

測定器の遠隔操作

遠隔地に設置した測定器を通信回線経由で制御し、データを収集することができます。

1.3 用語

本書で使用している省略語を表 1.3-1 に示します。

表1.3-1 省略語

省略語	正式名
ASCII	American Standard Code for Information Interchange
CR	Carriage Return
EOI	End or Identity
ESE	Event Status Enable Register
ESR	Event Status Register
GPIB	General Purpose Interface Bus
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronics Engineers
LAN	Local Area Network
LF	Line Feed
MAV	Message Available summery
MSS	Master Summery Status
OSER	Operation Status Enable Register
OSR	Operation Status Register
SCPI	Standard Commands for Programmable Interfaces
SRER	Service Request Enable Register
SRQ	Service Request
STB	Status Byte Register
TR	Transition Filter

第2章 ご使用になる前に

この章では、リモート制御をご使用になる前に準備することを説明します。

2.1	設備を準備する	2-2
2.2	機器を接続する	2-3
2.2.1	イーサネットによる接続	2-3
2.2.2	GPIB による接続	2-4
2.3	インタフェースを設定する	2-5
2.3.1	イーサネットを設定する	2-5
2.3.2	GPIB を設定する	2-8
2.4	接続を確認する	2-9
2.4.1	イーサネットの場合(Windows XP)	2-9
2.4.2	イーサネットの場合(Windows 7/Vista)	2-13
2.4.3	GPIB の場合	2-14
2.5	メッセージの書式	2-15
2.5.1	メッセージの種類	2-15
2.5.2	メッセージの構造	2-16
2.5.3	共通コマンド	2-18
2.5.4	機器固有コマンド	2-18
2.6	機器の状態を調べる	2-19
2.6.1	ステータスバイトレジスタ	2-19
2.6.2	標準イベントステータスレジスタ	2-21
2.7	メッセージの実行を確認する	2-23
2.8	マルチポート機能動作時のリモート制御	2-24

2

ご使用になる前に

2.1 設備を準備する

本器をリモート制御するには、次が必要です。

- コンピュータ
- イーサネットインタフェース
- イーサネットケーブル
- GPIB インタフェース(オプション追加時)
- GPIB ケーブル(オプション追加時)
- プログラム開発ツール

イーサネットインタフェース

次のいずれかの規格に適合したインタフェースを用意してください。

10BASE-T

100BASE-TX

それぞれの規格に適合したケーブルを使用してください。

GPIB インタフェース

IEEE488.2 の規格に適合した GPIB インタフェースを用意してください

プログラム開発ツール

リモート制御のプログラムを作成・実行するためのツールを用意します。プログラム開発ツールの要求仕様については、インタフェースのマニュアルを参照してください。

コンピュータ

GPIB インタフェース、およびプログラム開発ツールの動作環境を満たすコンピュータを用意してください。

2.2 機器を接続する

2.2.1 イーサネットによる接続

本器の左側面にあるイーサネットコネクタ(上側)と外部機器を、LAN ケーブルで接続します。

本器と制御用コンピュータを直接接続する場合は、LAN クロスケーブルで接続します。複数の外部機器と接続する場合は、ネットワークハブを介して LAN ストレートケーブルで接続します。

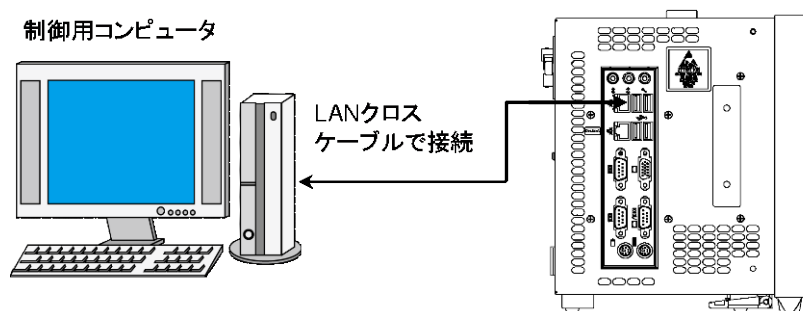


図 2.2.1-1 本器と制御用コンピュータの直接接続例

注:

本器を LAN に接続する場合は、本器のネットワーク設定を確認してから実施してください。

- ・ 本器の IP アドレスがほかの機器の IP アドレスと重複していない。
- ・ 本器のサブネットマスクで設定したアドレス範囲に、制御用コンピュータの IP アドレスが含まれる。

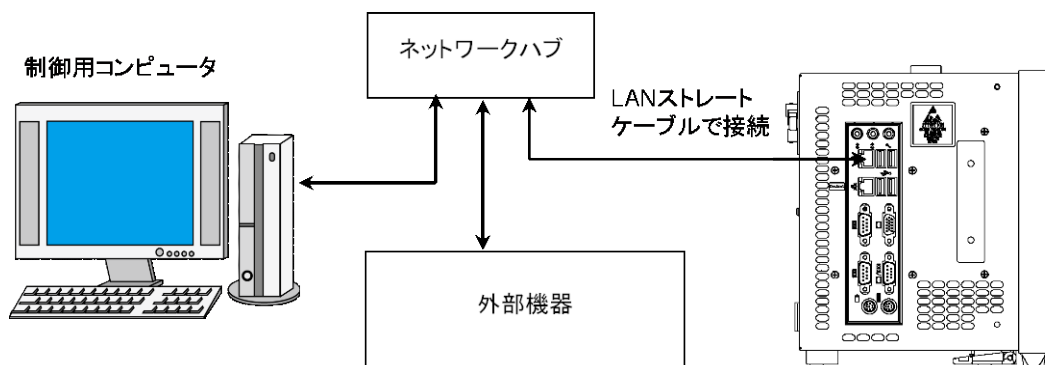


図 2.2.1-2 本器と複数の外部機器の接続例

注:

外部機器同士の通信状態によっては、本器と制御用コンピュータ間の通信がつながりにくくなる場合があります。安定して通信するには本器と制御用コンピュータを直接接続する方法をお勧めします。

2

ご使用になる前に

2.2.2 GPIBによる接続

本器の背面にある GPIB コネクタと外部機器を GPIB ケーブルで接続します。

注意

GPIB ケーブルの接続は、必ず本器の電源を入れる前に行ってください。電源を入れたまま接続すると、内部回路が損傷を受ける場合があります。

1 つの本器に接続できるデバイス台数は、制御用のコンピュータを含めて最大 15 台です。また、以下に示す条件に従って接続してください。

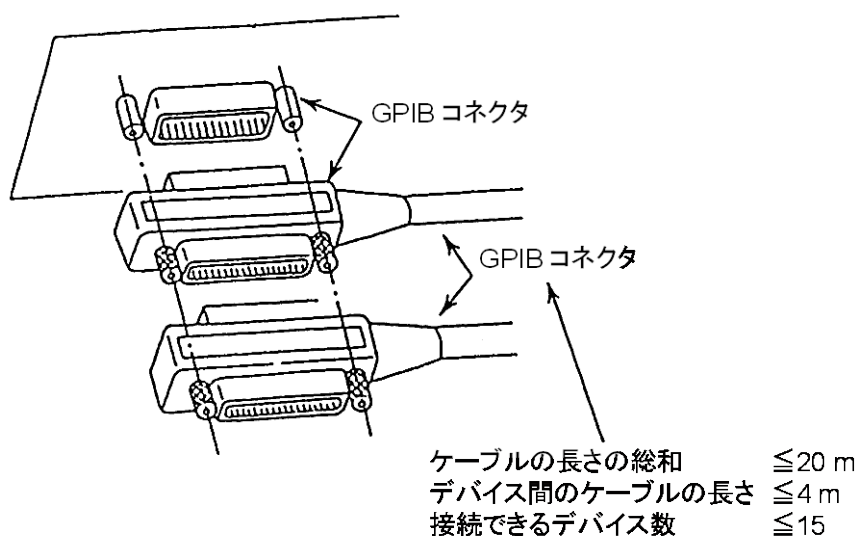


図 2.2.2-1 GPIB ケーブルの接続

ケーブルは、ループにならないように接続してください。

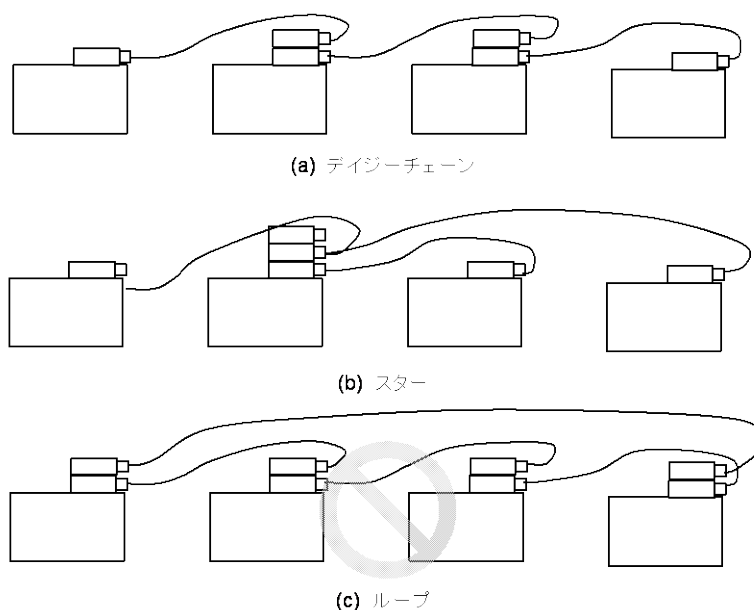


図 2.2.2-2 GPIB ケーブルの接続方法

2.3 インタフェースを設定する

2.3.1 イーサネットを設定する

次の方法でリモート制御インタフェースをイーサネットに設定し、IP アドレスを入力します。

1. 本器の電源を投入します。
2. セレクタ画面の, [Utility]タブをタッチします。
3. [Setup Utility] をタッチします。
4. [Remote Control] をタッチします。
5. Active Interface のボタンをタッチして、ボタンの表示を [Ethernet] にします。

Remote Control

Active Interface

Ethernet

GPIB

Address 1

Ethernet(Windows)

Local Area Connection

IP Address 192 . 168 . 10 . 94

Subnet Mask 255 . 255 . 255 . 0

Gateway Clear

Local Area Connection 2

IP Address 169 . 254 . 0 . 15

Subnet Mask 255 . 255 . 255 . 0

Gateway

Port Number 5001

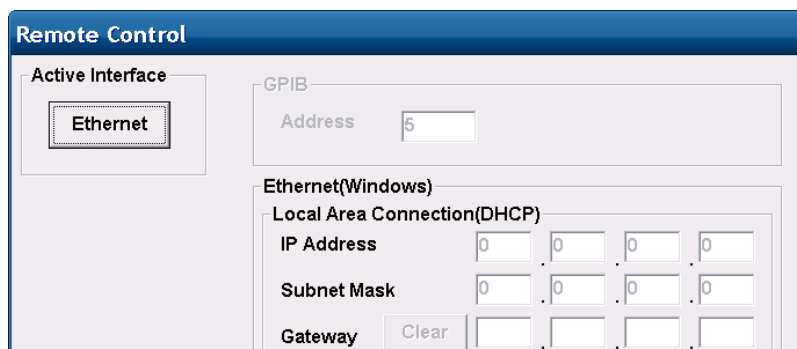
Apply Exit

6. IP アドレス、サブネットマスク、ゲートウェイ、およびポート番号を設定します。
ゲートウェイのアドレスは省略できます。
テキストボックスにタッチすると、数値入力パネルが表示されます。
外付けキーボードを使用して、直接テキストボックスに数値を入力できません
ポート番号は、1024~5001 の範囲で設定します。
7. [Apply] をタッチすると、設定が完了します。
[Exit] をタッチすると、設定した値は取り消されます。

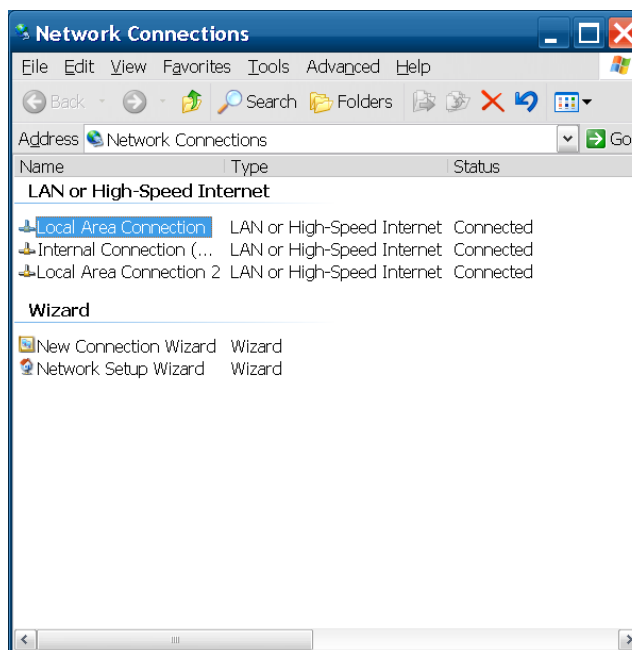
2

ご使用になる前に

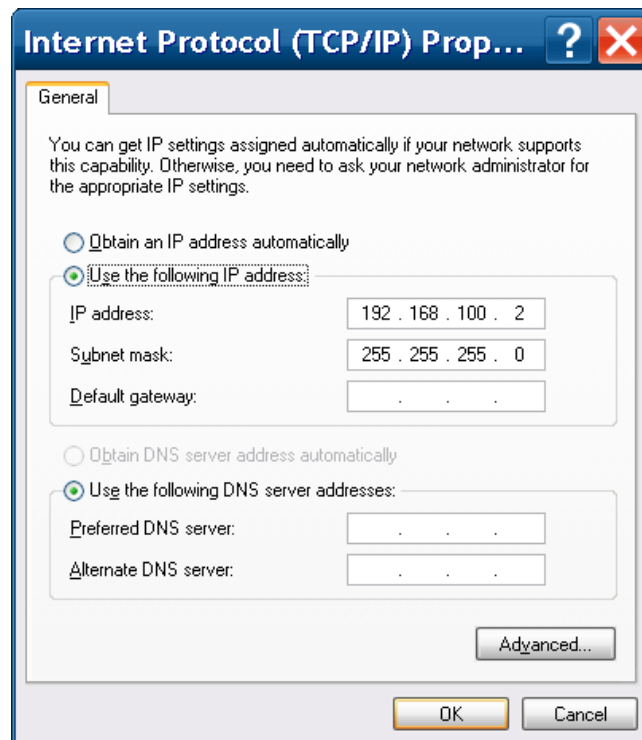
手順5.で画面に Local Area Connection (DHCP)が表示されたときは、次の手順でネットワークアドレスの自動取得を解除します。



1. キーボードとマウスを本器に接続します。
2. 接続したキーボードの Windows キーを押します。
キーボードに Windows キーが無い場合は、デスクトップを表示して [Start] をクリックします。
デスクトップの表示方法は『MD1260A 40/100G イーサネット アナライザ 取扱説明書』の「2.6.1 Windows デスクトップを表示する」を参照してください。
3. [Control Panel] をクリックします。
4. [Network Connections] をダブルクリックします。



5. [Local Area Connection]を右クリックして、[Property] をクリックします。
6. Local Area Connection Properties ウィンドウが開きます。
リストボックスの中の [Internet Protocol (TCP/IP)] をクリックして、[Property] ボタンをクリックします。



2

ご使用になる前に

7. [Use the following IP address] をチェックします。
工場出荷時の設定は次のとおりです。
IP address 192.168.100.101, Subnet mask 255.255.255.0
8. [OK] ボタンをクリックします。
9. [Local Area Connection Properties] の [OK] ボタンをクリックします。

注意

[Internal Connection (Don't Change)]の設定を変更しないでください。
変更すると本器が正常に動作しなくなります。

[Local Area Connection2]の設定を変更しないでください。
変更するとマルチポート機能が使用できなくなります。

2.3.2 GPIBを設定する

次の方法でリモート制御インタフェースを GPIB に設定し、GPIB アドレスを入力します。

1. 本器の電源を投入します。
2. セレクタ画面の, [Utility] タブをタッチします。
3. [Setup Utility] をタッチします。
4. [Remote Control] をタッチします。
5. Active Interface のボタンをタッチして、ボタンの表示を [GPIB] にします。

オプション 030 または 130 が追加されていないときは、Active Interface のボタンは操作できません。

Remote Control

Active Interface

GPIB

GPIB

Address 1

Ethernet(Windows)

Local Area Connection

IP Address 192 168 10 94

Subnet Mask 255 255 255 0

Gateway Clear

Local Area Connection 2

IP Address 169 254 0 15

Subnet Mask 255 255 255 0

Gateway

Port Number 5001

Apply Exit

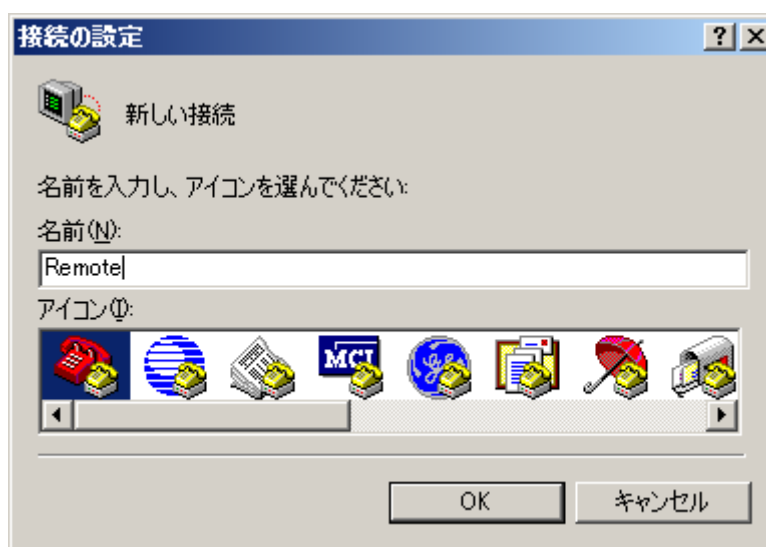
6. GPIB のアドレスのテキストボックスにタッチします。
7. 数値入力パネルで GPIB アドレスを設定します。
外付けキーボードからは、直接テキストボックスに数値を入力できません。
8. [Apply] をタッチすると、設定が完了します。
[Exit] をタッチすると、設定した値は取り消されます。

2.4 接続を確認する

制御側コンピュータが、イーサネット経由で本器を認識できるか確認します。

2.4.1 イーサネットの場合(Windows XP)

1. Windows のスタートメニューから、プログラムをクリックします。
2. [アクセサリ] をクリックします。
3. サブメニューの通信から、[ハイパーターミナル] をクリックします。
4. 次の画面が開きますので、名前を入力します。[OK] をクリックします。



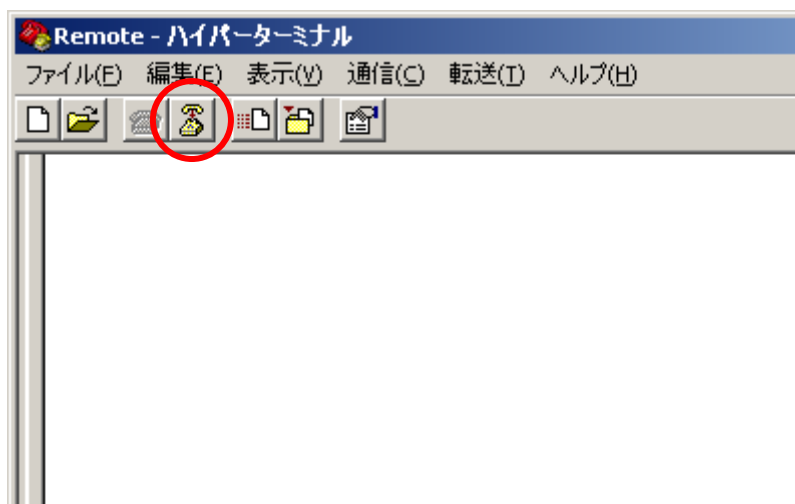
5. 接続方法を TCP/IP にして、ホストアドレスとポート番号を入力します。
「2.3.1 イーサネットを設定する」で設定した本器の IP アドレスが 192.168.100.2, ポート番号が 5001 のときの設定は次のとおりです。



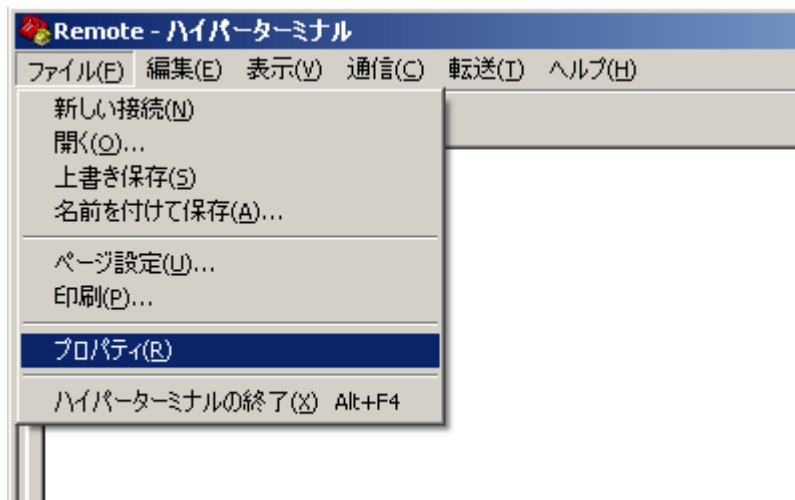
6. 「192.168.100.2 ポート 5001 に接続できません」と表示されたときは、コンピュータが本器を認識できていません。次を確認してください。
 - ・ 本器のアプリケーションが起動しているか。

(『MD1260A 40/100G イーサネット アナライザ 取扱説明書』の「3.1.1 アプリケーションの起動」を参照)

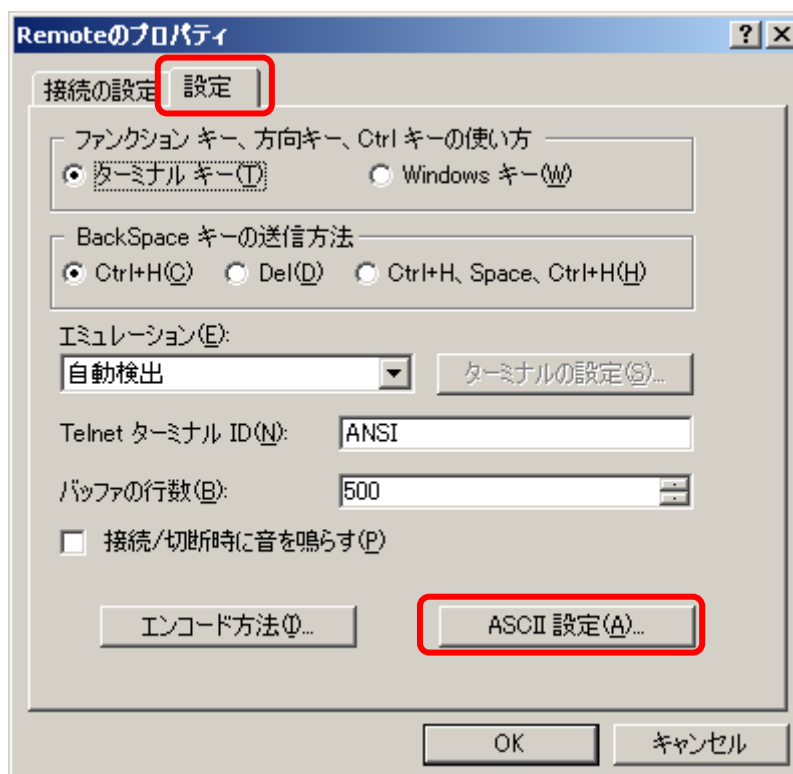
- ・ 本器の IP アドレスおよびポート番号に間違いが無いか。
 - ・ イーサネットケーブルのストレート/クロスの種類は正しいか。
 - ・ 本器左側面に 2 つあるイーサネットコネクタの、正しい方にイーサネットケーブルを接続しているか。
 - ・ ケーブルの断線またはコネクタの破損が無いか。
7. 通信中の表示になれば、制御用コンピュータと本器はイーサネット経由で接続できています。



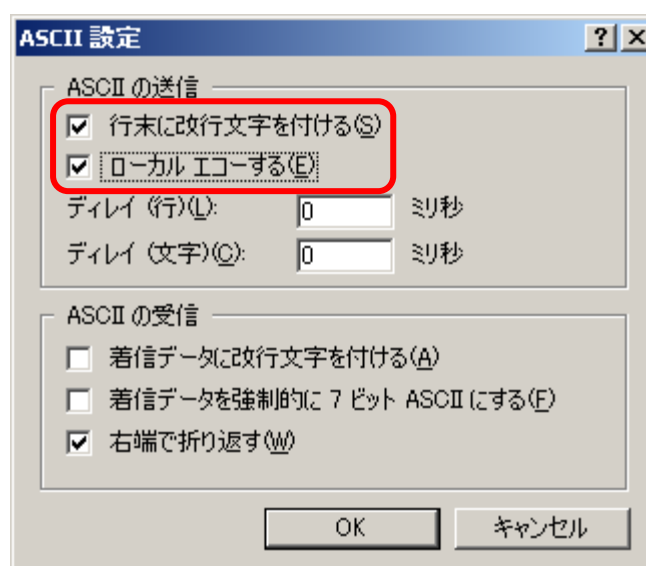
8. 本器とリモートコマンドのやり取りができるようにするために、ハイパーターミナルの設定を変更します。[ファイル]-[プロパティ]を選択します。



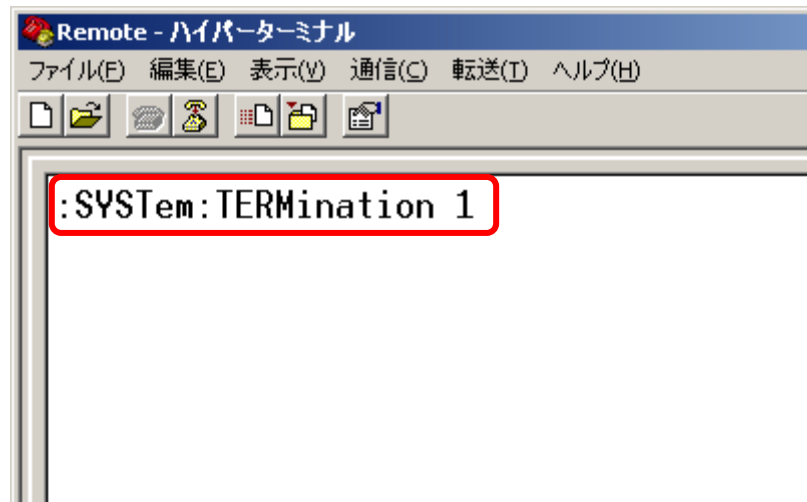
9. 本器とリモートコマンドのやり取りができるようにするために、ハイパーターミナルの設定を変更します。[設定] タブを選択して [ASCII 設定] ボタンをクリックします。



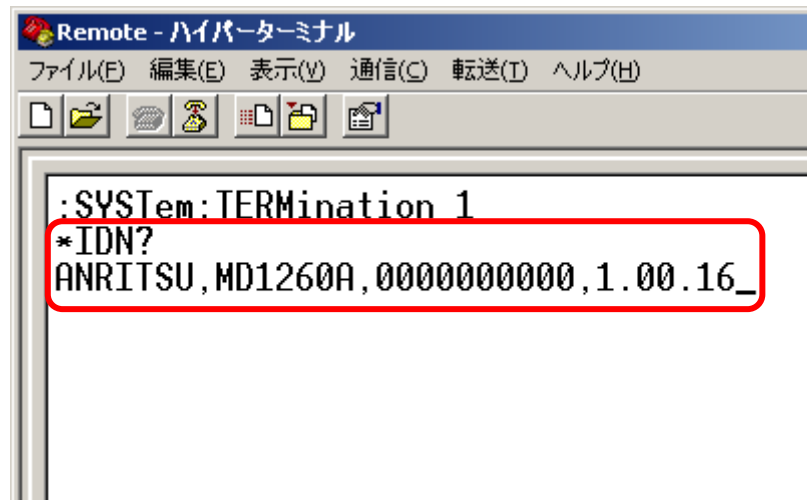
10. [行末に改行文字を付ける] と [ローカルエコーする] にチェックを入れて [OK] ボタンをクリックします。



11. 本器からのレスポンスをハイパーターミナルで正しく表示できるようにするために、本器からのレスポンスの終端文字 (ターミネータ) を CR+LF にします。「:SYSTem:TERMination 1」と入力して Enter キーを押します。コマンドが本器で受信されると、操作画面がロックされた状態 (リモート状態) になります。



12. 本器からのレスポンスが受信できることを確認するために、「*IDN?」と入力して Enter キーを押します。本器からのレスポンスを受信すると「ANRITSU,MD1260A,0000000000,1.00.16」などの文字列が表示されます。



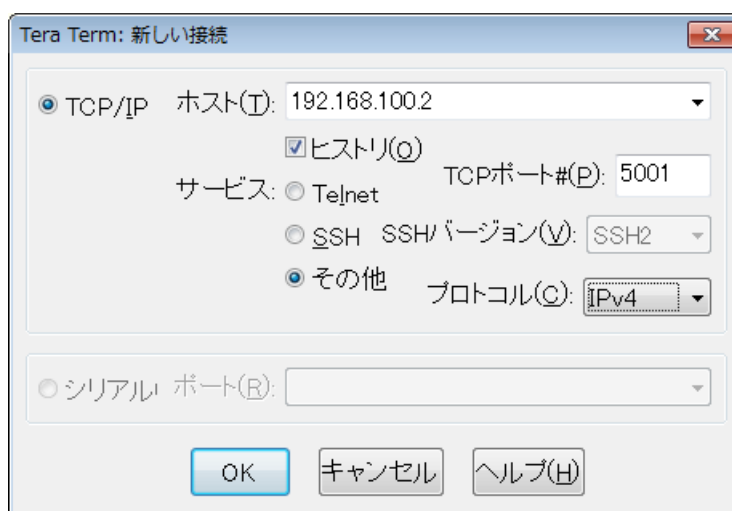
以上で、リモートコマンドによる制御が可能であることが確認できました。以降、同様にコマンドを入力することでリモートコマンドの動作を確認することができます。

2.4.2 イーサネットの場合(Windows 7/Vista)

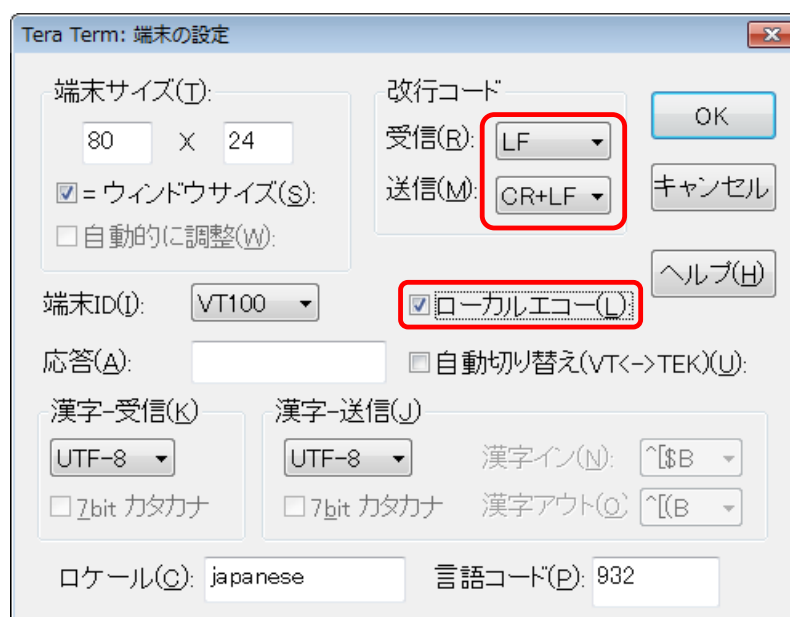
ここでは、フリーソフトウェアの Tera Term Version 4.69 を使用方法を説明します。

1. Tera Term を起動すると、「新しい接続」画面が開きます。
ホストに本機の IP アドレス, TCP ポート番号を入力します。
サービスを [その他], プロトコルを [IPv4] に設定します。
[OK] をクリックします。

「2.3.1 イーサネットを設定する」で設定した本機の IP アドレスが 192.168.100.2, ポート番号が 5001 のときの設定は次のとおりです。



2. 本機を認識できると、通信画面が表示されます。
3. メニューの [設定(S)] - [端末(T)..] をクリックします。
4. 改行コードの受信を [LF], 送信を [CR+LF] にします。
ローカルエコーをチェックします。[OK] をクリックします。



6. *IDN?を送信します。
本器からのレスポンスが表示されることを確認します。

注:

本器のパネル操作でパネルロックを解除すると、本器との通信が切断され Tera Term が終了します。

2.4.3 GPIBの場合

1. 制御用コンピュータに接続する GPIB インタフェースに添付されているソフトウェアをインストールします。
2. ソフトウェアを実行します。
操作方法は、GPIB インタフェースの取扱説明書を参照してください。
3. 本器に設定した GPIB アドレスの機器が表示されることを確認します。

2.5 メッセージの書式

2.5.1 メッセージの種類

メッセージは、メッセージ本体の文字列とメッセージの終了を示す文字列で構成されます。メッセージの終了を示す文字列は、LF(ラインフィード), または CR(キャリッジリターン)+LF です。

メッセージには送信方向によって次の種類があります。

プログラムメッセージ:

制御用コンピュータから測定器に送信されるメッセージです。

プログラムメッセージは、次の 2 つに分類できます。

- コマンド
測定条件の設定や、測定の開始などに使用します。
- クエリ
測定器の状態や設定を問い合わせます。
クエリを送信すると、測定器はクエリに対するレスポンスメッセージを作成します。

レスポンスメッセージ:

測定器から制御用コンピュータに送信されるメッセージです。

2.5.2 メッセージの構造

メッセージには、ヘッダー部分とデータ部分があり、その間には半角スペースを入れます。プログラムメッセージには、必ずヘッダーがありますがデータは無いことがあります。レスポンスメッセージには、必ずデータがありますがヘッダーは無いことがあります。

ヘッダー

コマンドのヘッダーには次の種類があります。

- ・ 共通コマンドヘッダー
英数字とアンダーバーで構成され、先頭文字はアスタリスク(*)です。
例: `*CLS`
- ・ 複合ヘッダー
コロンで結合したヘッダーです。複合ヘッダーは、処理を階層構造化することができます。
例: `:CALCulate:COUNter:STARt`

クエリのヘッダーは、最後にクエスションマークをつけます。

例: `*ESE?`

データ

データの書式には、文字列データ、数値データ、およびバイナリデータがあります。

文字列データは、アスキーコードの文字列をクォーテーション・マークで囲みます。ファイル名に「`TestResult`」と入力するときのプログラムメッセージの例は、次のとおりです。

例: `:MMEMory:STORe "TestResult",RESULT`
`:MMEMory:STORe 'TestResult',RESULT`

文字列にクォーテーション・マークを含むときは、クォーテーション・マークを 2 つ連続させます。

例: `He said "Good product".` → `"He said ""Good Product""."`
`He said 'Good product'.` → `'He said "'Good Product'.'`

または、文字データに使用するクォーテーション・マークと、文字列を囲むクォーテーション・マークとを別の種類にします。

例: `He said "Good product".` → `'He said "Good Product".'`
`He said 'Good product'.` → `"He said 'Good Product'."`

数値データは 10 進数、2 進数、8 進数および 16 進数で記述できます。

2 進数、8 進数、16 進数の場合は、データの前に `#B`、`#O`、`#H` をつけます。

次の例はいずれも同じ値を表します。

例: `10` `#B1010` `#O12` `#HA`

1550 #B11000001110 #03016 #H60E

10 進数のときは、整数、固定小数点または浮動小数点で書くことができます。
次の例はいずれも同じ値を表します。

例: -10	-10.00	-1E1
1250	1250.000	1.25E3
0.0023	2.3E-4	

バイナリデータは、先頭文字が番号記号(#)で始まり、データ長を示す数字の後に
データを続けます。

番号記号(#)の次の文字が 0 以外の数字のときは、データ長の桁数を表します。
データ長を表す数字の後からバイナリデータが続きます。

例: #42002an%*qe4445+¥...

4 桁

↑

2002 バイトのバイナリデータ

番号記号(#)の次の文字が 0 のときは、0 の次からバイナリデータが続きます。

例: #0an%*qe4445+¥...

↑

バイナリデータ

メッセージに複数のデータがあるときは、コンマでデータを区切ります。

例: :CALCulate:DATA? RX_FREQ,RX_FREQ_D

複数のプログラムメッセージを連結するときは、セミコロンでメッセージを区切ります。

例: :MMEMory:STORe "TestResult",RESULT;:SYSTem:ERRor?

2
ご使用になる前に

2.5.3 共通コマンド

GPIBの規格 (IEEE 488.2) では, 機器が対応するコマンドを定義しています。このコマンドを本書では, 「共通コマンド」と呼びます。

共通コマンドには, 対応が必須のコマンドとオプションのコマンドがあります。本器は以下の共通コマンドに対応しています。

表 2.5.3-1 本器の共通コマンド

コマンド	説明
*CLS	標準イベントレジスタと出力キューのクリア
*ESE	標準イベントイネーブルレジスタの設定と問い合わせ
*ESR	標準イベントレジスタの問い合わせ
*IDN	製品情報の問い合わせ
*OPC	処理完了時のステータスバイトのビット設定, ステータスバイトのビット 0 の問い合わせ
*RST	工場出荷時の機器設定に変更
*SRE	サービスリクエストイネーブルレジスタの設定と問い合わせ
*STB	ステータスバイトレジスタの問い合わせ
*TRG	測定の開始
*WAI	実行中の処理が終了するまで, 次のコマンド実行を待機

2.5.4 機器固有コマンド

測定器の機能によって, 機器ごとに異なるコマンドを本書では, 「機器固有コマンド」と呼びます。

2.6 機器の状態を調べる

本器には、エラーの発生やコマンドの実行状況など機器の状態を表示するレジスタがあります。ここではそのレジスタを説明します。

2.6.1 ステータスバイトレジスタ

ステータスバイトレジスタ (STB)は、 GPIB の規格で定義された機器の状態を表示するレジスタです。機器の状態が変化すると、ステータスバイトレジスタの値が変化します。このとき、制御コンピュータに対して割り込みを発生されることかできます。この割り込みをサービスリクエストと呼びます。

ステータスバイトのレジスタに対して、サービスリクエストイネーブルレジスタ(SRER)があります。サービスリクエストイネーブルレジスタは、サービスリクエストを発生するステータスバイトのビットを選択できます。

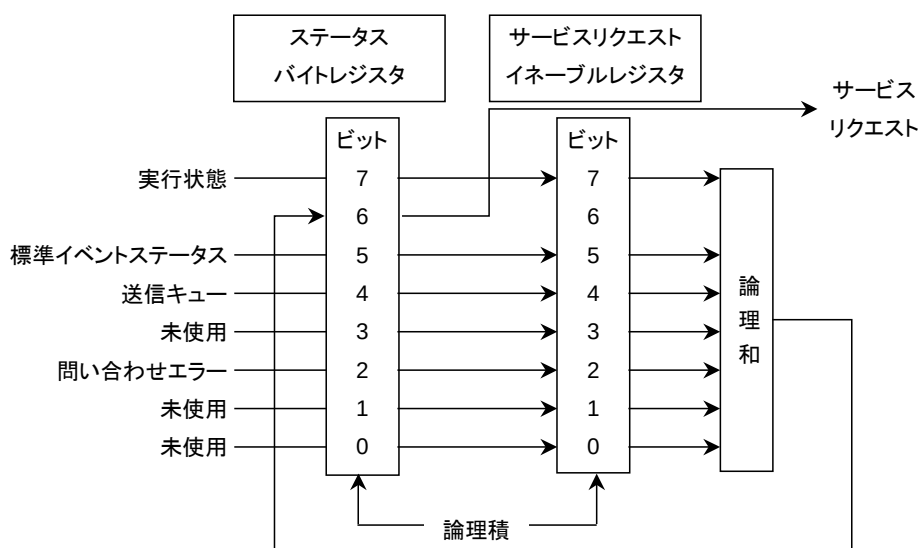


図 2.6.1-1 ステータスバイトレジスタとサービスリクエストイネーブルレジスタの構成

注

サービスリクエストは、 GPIB インタフェースのときに使用できます。

ステータスバイトレジスタは、次の方法で読みとります。

- ・ 共通コマンドの*STB?を使用する
- ・ GPIB のシリアルポールを使用する

シリアルポールの方法は、 GPIB インタフェースの説明書を参照してください。

シリアルポールを使用するときは、ビット 6 が 1 であっても一度読みとると 0 になります。

サービスリクエストイネーブルレジスタの設定と読みとりには、共通コマンドの*SRE、*SRE?を使用します。ステータスバイトレジスタのデータを出力するためには、サービスリクエストイネーブルレジスタの該当するビットを、1 に設定してください。

ステータスバイトレジスタの各ビットの定義を、次の表に示します。

表 2.6.1-1 ステータスバイトレジスタのビット定義

ビット	説明
7	未使用 常に 0 になります。
6	MSS (Master Summary Register) ステータスバイトレジスタとサービスリクエストイネーブルレジスタとの論理積の、ビット 7 とビット 5～0 の論理和です。
5	標準イベントレジスタと、標準イベントイネーブルレジスタとの論理積の、各ビットの論理和です。
4	MAV (Message Available summary) 測定器の出力キューに、レスポンスメッセージがあるときに 1 になります。
3	未使用 常に 0 になります。
2	未使用 常に 0 になります。
1	未使用 常に 0 になります。
0	未使用 常に 0 になります。

ステータスバイトレジスタのビット 6 は、マスターサマリーステータス(MSS)と呼ばれます。MSS が 1 のとき、本器から制御コンピュータへ通知する情報があります。MSS が 0 から 1 に変化すると、サービスリクエストを発生します。

ステータスバイトレジスタのビット 5 は標準イベントレジスタの情報を示します。情報の詳細は、「2.6.2 標準イベントステータスレジスタ」を参照してください。

ステータスバイトレジスタのビット 5 は、共通コマンド*CLS で 0 にできます。コマンドの後に*CLS を送信したとき、または*CLS の後にクエリを送信したときは、送信キューがクリアされてビット 4 が 0 になります。

サービスリクエストイネーブルレジスタは、*CLS で 0 にできませんので*SRE を使用してください。

2.6.2 標準イベントステータスレジスタ

標準イベントステータスレジスタ(ESR)に対して、標準イベントステータスイネーブルレジスタ(ESE)があります。この 2 つのレジスタの論理積をとり、その結果の各ビットの論理和が、ステータスバイトレジスタのビット 5 に出力されます。

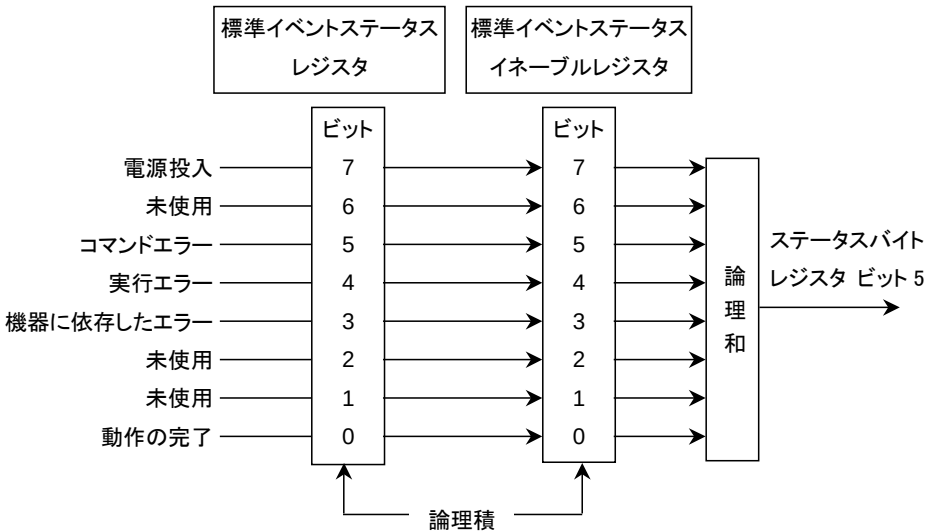


図 2.6.2-1 標準イベントステータスレジスタと標準イベントステータスイネーブルレジスタの構成

標準イベントステータスレジスタの各ビットの定義は、次のとおりです。

表 2.6.2-1 標準イベントレジスタのビット定義

ビット	説明
7	電源投入 電源が投入されると 1 になります。1 回読みとると 0 になります。
6	未使用 常に 0 になります。
5	コマンドエラー 定義されていないプログラムメッセージ、文法に従わないプログラムメッセージ、スペルミスがあるプログラムメッセージを受信したときに 1 になります。
4	実行エラー コマンドエラーにならないが、実行できないプログラムメッセージを受信したときに 1 になります。
3	未使用 常に 0 になります。
2	未使用 常に 0 になります。
1	未使用 常に 0 になります。
0	操作の完了 プログラムメッセージを受信すると 0 になります。 コマンドに対する動作が完了したとき、クエリに対するレスポンスデータの送信が終了したときに 1 になります。

ビット 5 のコマンドエラーは、次の場合に発生します。

- 第 3 章に記載していないメッセージを送信したとき

2
ご使用になる前に

例: メッセージにタイプミスがある。

メッセージに全角文字がある。

メッセージとパラメータを区切るスペースが無い。

複数のメッセージを区切るセミコロンが無い。

メッセージの省略できない文字を省略した。

- パラメータの数が正しくないとき

例: パラメータの数が 1 つと定義されているメッセージに, パラメータを 2 個送信した。

パラメータを区切るコンマが無い。

- パラメータの書式が正しくないとき

例: パラメータとして数値が指定されているメッセージに, 文字を送信した。

文字列データをクォーテーションマークで囲っていない。

ビット 4 の実行エラーは, 次の場合に発生します。

- パラメータの値が設定範囲外のとき

例: ビットレートオフセットを 150 ppm に設定した。

外部減衰量係数を 100 dB にした。

- コマンドは正しいが, 本器の状態により実行できないとき

例: OTU4/3 が無いオプション構成で, OUT フレームの設定をした。

標準イベントステータスレジスタのビット 7 からビット 0 は, *ESR? で読みとることができます。読みとると, 標準イベントレジスタは 0 になります。

標準イベントステータスイネーブルレジスタの設定と読みとりには, *ESE と *ESE? を使用します。標準イベントレジスタのデータを出力するためには, イネーブルレジスタの該当するビットを 1 に設定してください。

ビット 0 は, 共通コマンド *OPC で読みとることができます。

標準レジスタは, 共通コマンド *CLS で 0 にできます。

2.7 メッセージの実行を確認する

イーサネットインタフェースを使用する場合

本器に送信したプログラムメッセージが実行されている間であっても、メッセージを送信できます。ただし、前に送信したメッセージの実行が終了するまで、次のメッセージは処理されません。

このため、イーサネットインタフェースを使用する場合は、先に送信したメッセージの実行終了を確認してから、次のメッセージを送信してください。

メッセージの実行終了を確認するには、クエリを送信します。

前のメッセージの処理が終了すると、クエリが処理されレスポンスを受信します。

使用例:

:MMEMory:STORe "TestResult",RESULT

測定結果を保存します。

:System:Error?

エラーの発生を問い合わせます。

> 0, "No Error"

読みとった値が 0:エラー無し

:System:Error?を送信する前のコマンドの処理に時間がかかる場合、レスポンスメッセージが返信されるまでに時間がかかることがあります。レスポンスメッセージを受信することにより、**:MMEMory:STORe**と**:System:Error?**のメッセージが実行されたことを確認できます。

GPIO インタフェースを使用する場合

送信したメッセージが実行されている間は、本器にメッセージを送信できません。このため、実行時間のかかるコマンドを送信する場合は、本器がメッセージを実行している間に通信がタイムアウトにならないように、制御用コンピュータのGPIOインタフェースのタイムアウト時間を調整してください。

2.8 マルチポート機能動作時のリモート制御

マルチポート機能で制御されている MD1260A をリモート制御する場合は、マスタとなっている MD1260A に制御用コンピュータを接続します。

マルチポート機能については、『MD1260A 40/100G イーサネット アナライザ 取扱説明書』の「第7章 マルチポート機能」を参照してください。

マスタと、アプリケーション画面のトップメニューに表示されているスレーブをリモート制御することができます。

MD1260A のユニット ID のリストを取得するには、:UENTry:LIST?で問い合わせできます。

MD1260A を個別に制御するには、:UENTry:ID で MD1260A のユニット ID を指定します。

ヘッダーに[:AUNit]があるコマンドは、:AUNit を追加するとすべての MD1260A でコマンドが実行されます

第3章 メッセージの詳細

この章ではメッセージの詳細を説明します。

3.1	メッセージの記述方法.....	3-2
3.2	パネル操作とメッセージの対応	3-4
3.2.1	共通操作に対応するメッセージ.....	3-4
3.2.2	40GbE/100GbE に対応するメッセージ.....	3-10
3.2.3	OTU3/OTU4 に対応するメッセージ.....	3-26
3.2.4	No Frame に対応するメッセージ	3-38
3.2.5	Add-on 機能.....	3-40
3.2.6	対応するパネル操作が無いメッセージ	3-40
3.2.7	リモート制御できないパネル操作	3-41
3.3	メッセージの説明.....	3-42
3.3.1	共通コマンド.....	3-42
3.3.2	機器固有コマンド	3-48

3.1 メッセージの記述方法

メッセージの説明に使用する記号の説明を次の表に示します。

表3.1-1 コマンドの記述方法

記号	説明
[]	角カッコで囲ったメッセージまたはパラメータは、省略できます。
	複数の選択肢から 1 つを選びます。 A B C D の場合は A, B, C, D のどれか 1 つを選びます。
{ }	選択肢をグループ化します。 A B({C D}) の場合は、A, B(C), B(D) のどれか 1 つを選びます。
<CHAR>	ニーモニックにより選択肢を表します。 IEEE488.2 の<CHARACTER PROGRAM DATA>、<CHARACTER RESPONSE DATA> に相当します。
<BOOLEAN>	設定の有効/無効を示します。 有効を設定する場合は、ON または 1 にします。 無効を設定する場合は、OFF または 0 にします。 クエリに対するレスポンスの場合は、有効ときは 1、無効のときは 0 になります。
<NR1>	整数値を表します。 例 123, -500
<NR2>	固定小数点形式の数値を表します。 例 12.345, -500.0
<NR3>	浮動小数点形式の数値を表します。 例 0.00E-11, 3.05E-11
<HEX>	16 進表記の数値を表します。 #H の後に 16 進数表記の数値を指定します。 例 #H0011EEFF
<BINARY>	2 進表記の数値を表します。 #B の後に 2 進数表記の数値を指定します。 例 #B0011010
<STRING>	文字列データを示します。詳細は「2.5.2 メッセージの構造」のデータの説明を参照してください。

ヘッダーの文字列には、省略できる部分があります。
省略できない部分は大文字、省略できる部分は小文字で記載します。

例: :SYSTem:ERRor?

このヘッダーは、次のとおり記述できます。

:SYSTem:ERRor?

:SYST:ERR?

:SYSTEM:ERROR?

:SYSTem:ERR?

本器は、これらのメッセージを同じ意味に解釈します。

3.2 パネル操作とメッセージの対応

パネル操作に対応するメッセージを説明します。

メッセージが存在しないパネル操作については、「3.2.7 リモート制御できないパネル操作」を参照してください。

3.2.1 共通操作に対応するメッセージ

The screenshot shows a web-based configuration interface for a multi-port system. At the top, there are tabs for 'Ethernet', 'OTN', 'Utility', and 'Multi Port'. The 'Multi Port' tab is selected. Below the tabs is a table with three columns: 'No', 'Unit ID', and 'Application'. The table contains 16 rows. The first row is highlighted in blue and shows 'Unit 2 (This Chassis)' and 'ODTU4.1-ODU0-GbE'. To the right of the table are several buttons: 'Unit 2', 'Go Slave', 'Update', a set of up/down arrows, 'Select', and 'Start'. A label ':SYSTem:CONFig' with a line pointing to the first row of the table indicates the corresponding message for this operation.

No	Unit ID	Application
1	Unit 2 (This Chassis)	ODTU4.1-ODU0-GbE
2	None	
3	None	
4	None	
5	None	
6	None	
7	None	
8	None	
9	None	
10	None	
11	None	
12	None	
13	None	
14	None	
15	None	
16	None	

図3.2.1-1 マルチポート設定に対応するメッセージ

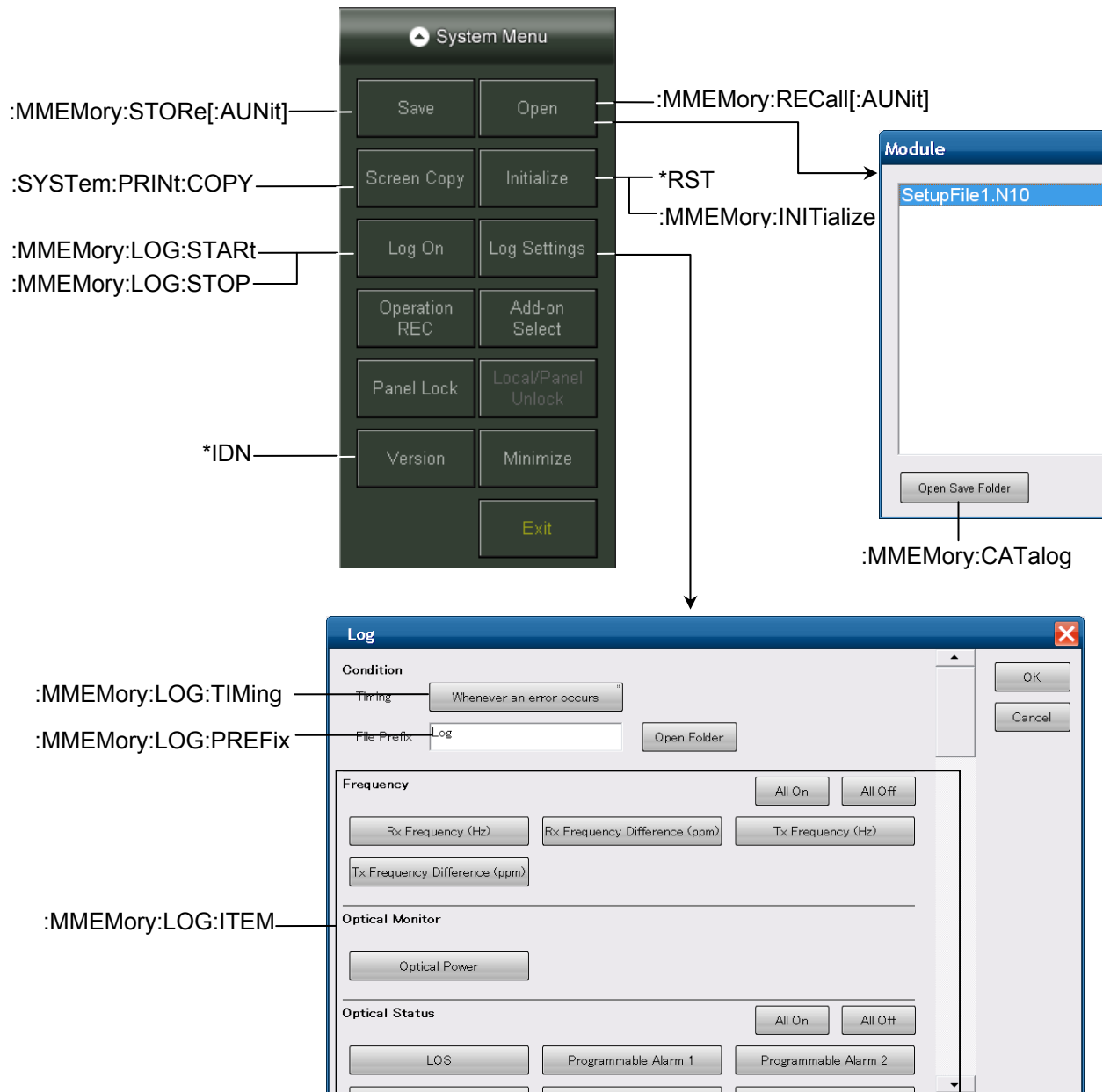


図3.2.1-2 システムメニューに対応するメッセージ

3

メッセージの詳細

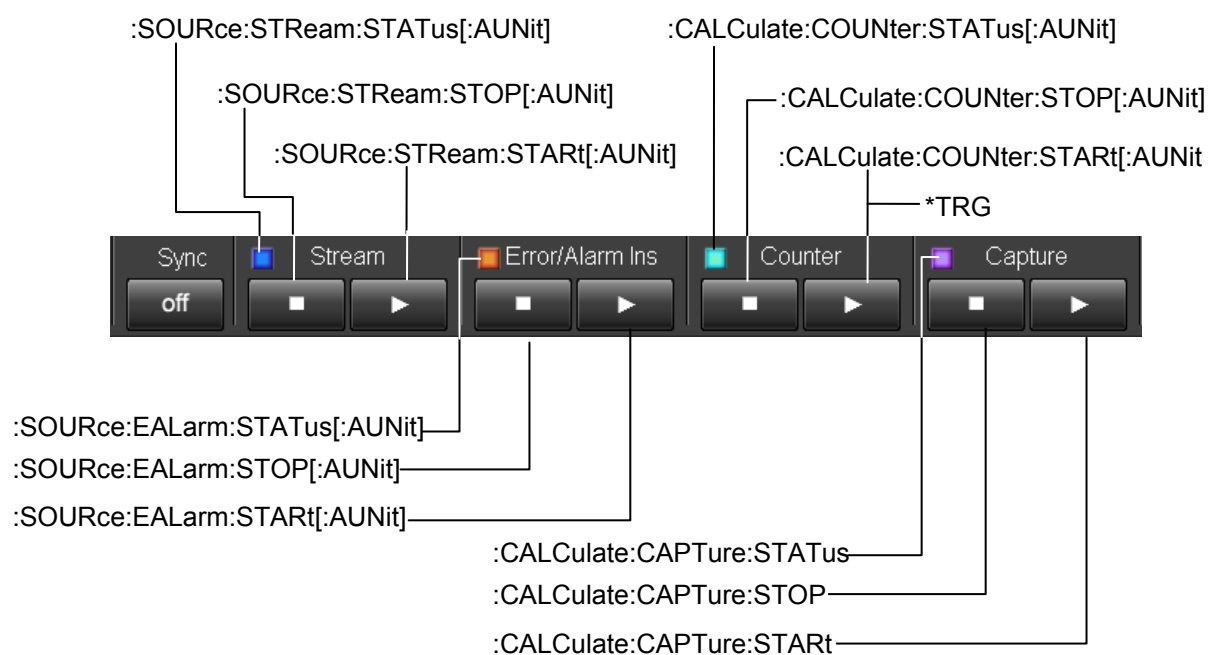


図3.2.1-3 操作エリアに対応するメッセージ

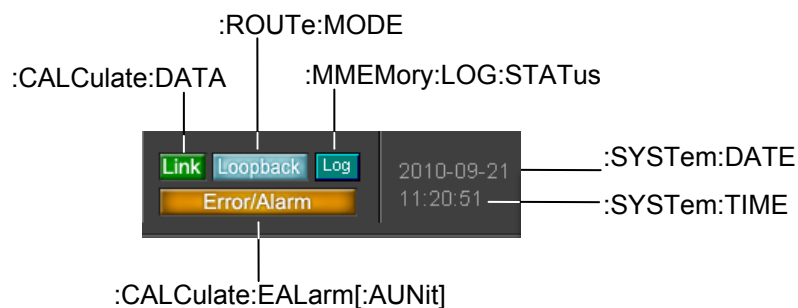


図3.2.1-4 サマリメッセージ・時刻表示エリアに対応するメッセージ

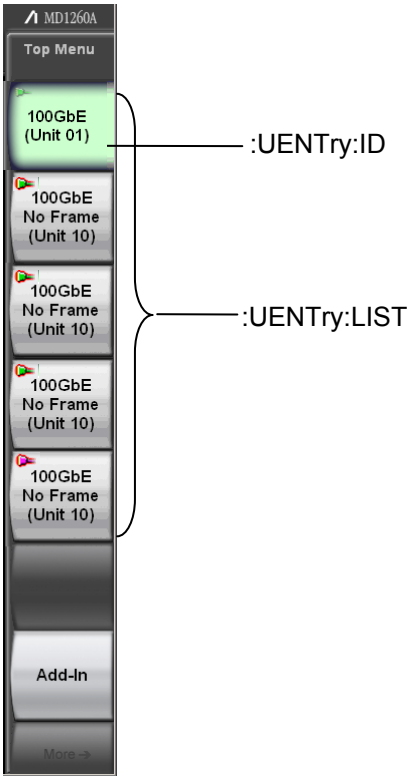


図3.2.1-5 トップメニューに対応するメッセージ

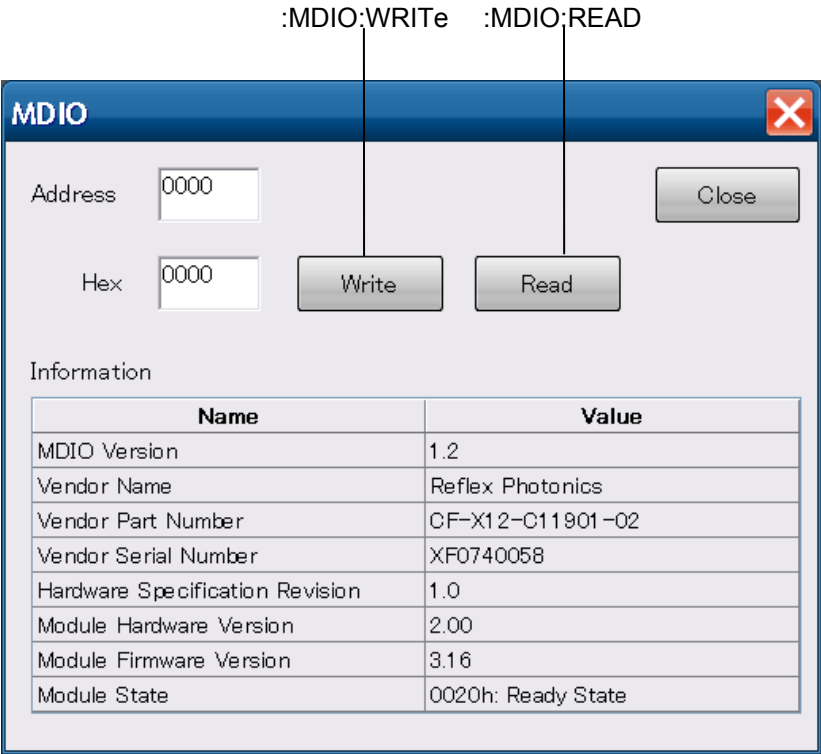


図3.2.1-6 設定エリア[MDIO]に対応するメッセージ

:SENSe:TRANsceiver:EQualizer:DCGain

Transceiver
✖

	Tx				Rx Equalizer	
	VOD	Pre-Emphasis			Control	DC Gain
		First	Pre	Second		
Tracking	Off	Off	Off	Off	Off	Off
Lane 0	0	0	0	0	0	0
Lane 1	0	0	0	0	0	0
Lane 2	0	0	0	0	0	0
Lane 3	0	0	0	0	0	0
Lane 4	0	0	0	0	0	0

:SOURce:TRANsceiver:VOD

:SOURce:TRANsceiver:EMPHasis:FIRSt

:SOURce:TRANsceiver:EMPHasis:PRE

:SOURce:TRANsceiver:EMPHasis:SECond

:SENSe:TRANsceiver:EQualizer:CONTRol

図3.2.1-7 設定エリア[Transceiver]に対応するメッセージ

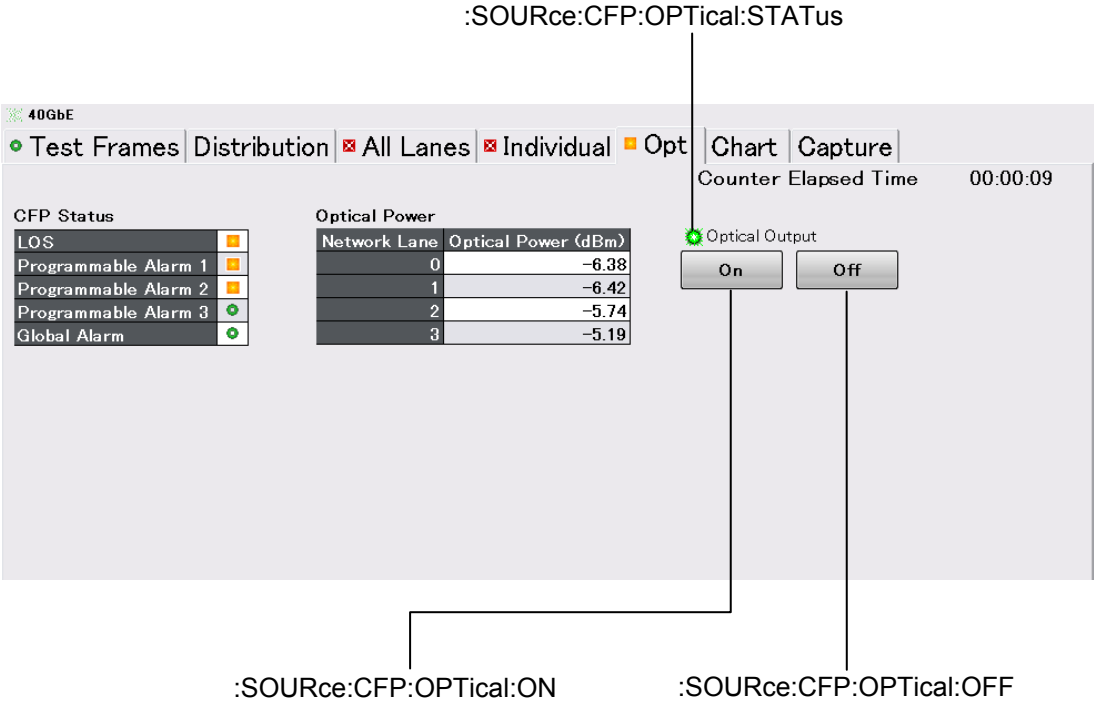


図3.2.1-8 測定結果表示エリア (Opt) に対応するメッセージ

3

メッセージの詳細

3.2.2 40GbE/100GbEに対応するメッセージ

設定エリアの [MDIO], [Transceiver]に対応するメッセージは、次を参照してください。

図3.2.1-6 設定エリア[MDIO]に対応するメッセージ

図3.2.1-7 設定エリア[Transceiver]に対応するメッセージ

The 'Stream' window contains the following fields and controls:

- Control Unit: Gap Size (byte)
- Duration: Repeat, 100,000,000, count, Total 11,117,276.480s, 1,800,000,000 frames
- Test Pattern: Word16, 5555, MAC Resolve...
- Transmission Type: Random
- Buttons: OK, Save, Apply, Load, Cancel
- Table with columns: No., Size (byte), Gap Size (byte), Name, Error, Flow ID
- Buttons on the right: Selection (up/down arrows), Control/Header..., Test Frame..., FCS Error Ins..., Copy/Paste to..., Move to...

Labels and their corresponding message names:

- :SOURCE:STReam:ENABLE (points to the 'Enable/Disable...' button)
- :SOURCE:STReam:DURation:TYPE (points to the 'Repeat' button)
- :SOURCE:STReam:DURation:FRAMES (points to the 'count' field)
- :SOURCE:STReam:DURation:REPeat:COUNT (points to the '100,000,000' value)
- :SOURCE:STReam:TYPE (points to the 'Transmission Type' dropdown)
- :SOURCE:STReam:ID (points to the 'Test Pattern' dropdown)
- :MMEMory:STReam:RECall (points to the 'MAC Resolve...' button)
- :MMEMory:STReam[:AUNit] (points to the 'Test Pattern' field)
- :SOURCE:TPATtern:TYPE (points to the 'Word16' dropdown)
- :SOURCE:TPATtern:WORD (points to the '5555' field)
- :SOURCE:STReam:ERRor:TYPE (points to the 'Error' column in the table)

No.	Size (byte)	Gap Size (byte)	Name	Error	Flow ID
On 1	120-1000	1,500,000-579,999	Ethernet + IPv4 + ICMPv4 (Echo)	-	0
On 2	84	10,000	Ethernet + IPv6 + ICMPv6 (Echo)	-	1
On 3	108	1,200,000	Ethernet + IPv6 + ICMPv6 (NS)	-	2
On 4	64	18,500,000	Ethernet + IPv4 + ICMPv4 (Echo)	-	3
On 5	64	150,000,128	Ethernet + IPv4	-	4
On 6	64	6,000,000	Ethernet + IPv4 + ICMPv4 (Echo)	-	5
On 7	64	85,000	Ethernet + IPv4 + ICMPv4 (Echo)	-	6
On 8	64	12,000,000	Ethernet + IPv4 + ICMPv4 (Echo)	-	7
On 9	64	9,000,000	Ethernet + IPv4 + ICMPv4 (Echo)	-	8
On 10	64	652,000	Ethernet + IPv4 + ICMPv4 (Echo)	-	9
On 11	64	960,000	Ethernet + IPv4 + ICMPv4 (Echo)	-	10
On 12	64	500,000	Ethernet + IPv4 + ICMPv4 (Echo)	-	11
Off 13	64	1,500,000	Ethernet + IPv4 + ICMPv4 (Echo)	-	12
Off 14	64	600,000	Ethernet + IPv4 + ICMPv4 (Echo)	-	13
Off 15	64	300,000	Ethernet + IPv4 + ICMPv4 (Echo)	-	14
Off 16	64	5,500,000	Ethernet + IPv4 + ICMPv4 (Echo)	-	15

図3.2.2-1 設定エリア[Stream]に対応するメッセージ

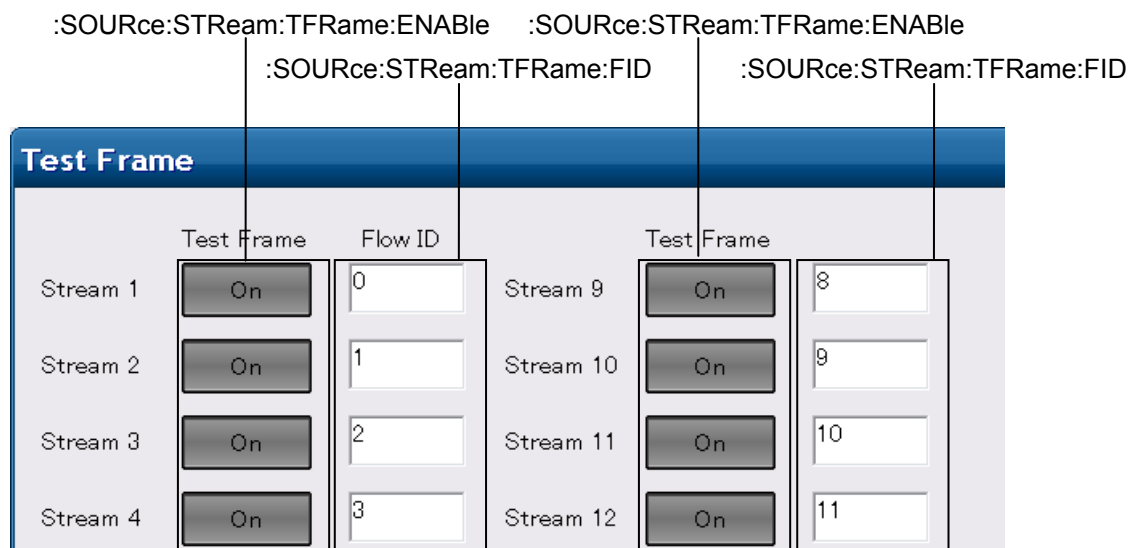


図3.2.2-2 Stream 画面-TestFrame 画面に対応するメッセージ

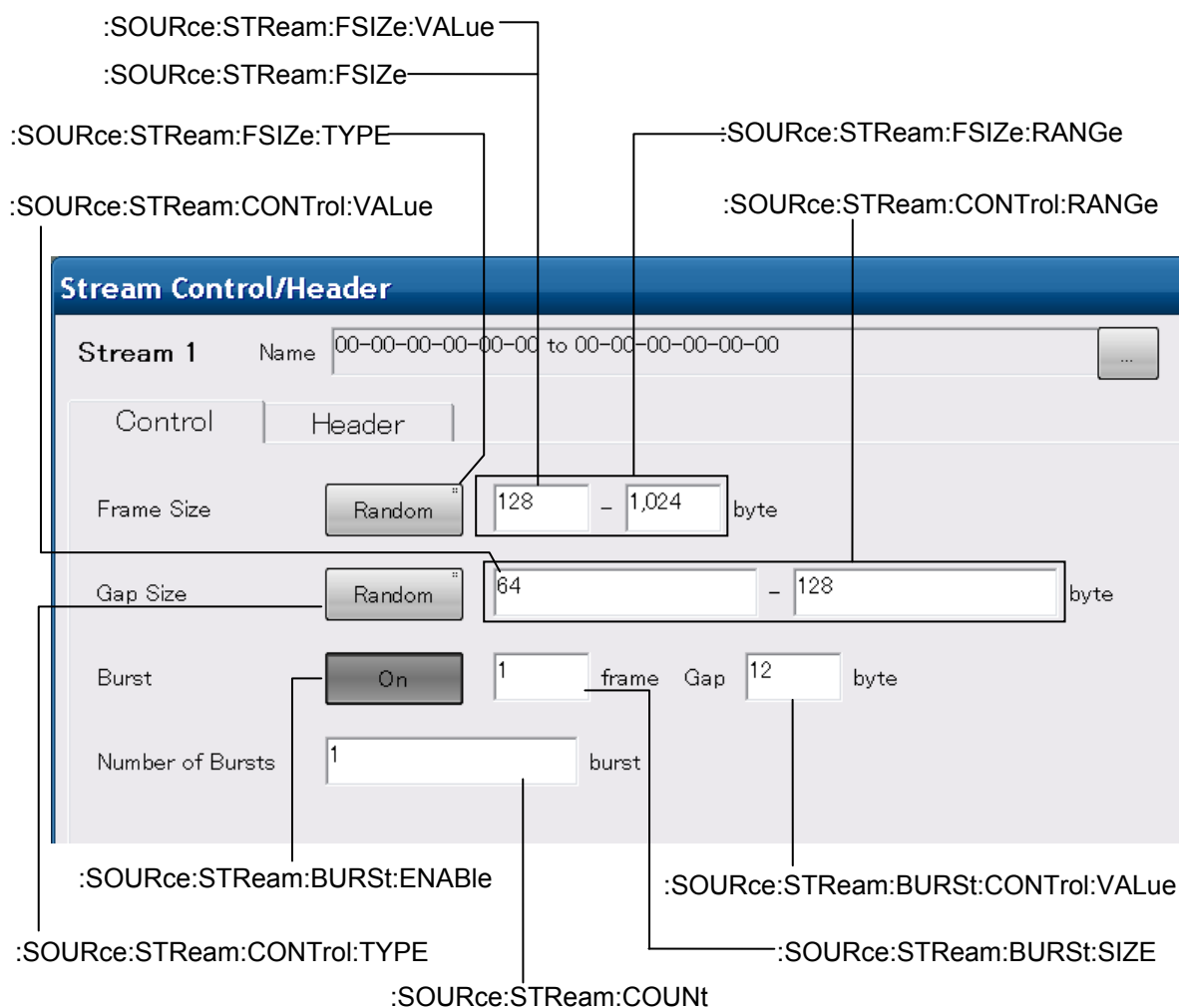


図3.2.2-3 Stream 画面-Control タブに対応するメッセージ

:SOURCE:STReam:HEADer:ETHer:DA

:SOURCE:STReam:HEADer:ETHer:SA :SOURCE:STReam:HEADer

Stream Control/Header

Stream 1

Name

MPLS-TP w/o CW (4) + PBB (B-Tag) + Ethernet + MPLS(3) + IPv4

...

Auto

On

Control

Header

Frame Format

MPLS-TP w/o CW (4) + PBB (B-Tag) + Ethernet + MPLS(3) + IPv4

MPLS-TP

PBB

Ethernet

MPLS

IPv4

Modifiers

Header Pattern

Source Address

192

168

0

0

Fixed

Destination Address

192

168

0

0

Fixed

TOS

bin

00000000

Fixed

TTL

64

Fixed

Protocol

17 - UDP

Fixed

図3.2.2-4 Stream 画面-Header タブに対応するメッセージ 1

:SOURCE:STReam:HEADer:VARiable1:TYPE

:SOURCE:STReam:HEADer:VARiable2:TYPE

:SOURCE:STReam:HEADer:VARiable1:RANGe

:SOURCE:STReam:HEADer:VARiable2:RANGe

Stream Control/Header

Stream 1

Name MPLS-TP w/o CW (1) + PBB (B-Tag) + Ethernet + MPLS(3) + IPv4

...

Auto

On

Control

Header

Frame Format

MPLS-TP w/o CW (1) + PBB (B-Tag) + Ethernet + MPLS(3) + IPv4

MPLS-TP

PBB

Ethernet

MPLS

IPv4

Modifiers

Header

Pattern

Field 1 (MPLS-TP DA)

Increment

Offset 0

byte

Length 1

byte

Count 1

Field 2 (MPLS-TP SA)

Increment

Offset 0

byte

Length 1

byte

Count 1

Field 3

Random

MPLS - MPLS Tags - Tag 1 - Time to Live (8 bits)

Offset 0

bit

Length 8

bit

Value Range

dec 0

to 255

hex 00 to FF

bin 00000000 to 11111111

Field 4

Random

IPv4 - Protocol (8 bits)

Offset 0

bit

Length 8

bit

Value Range

dec 0

to 255

hex 00 to FF

bin 00000000 to 11111111

Field 5

Increment

Ethernet - Destination Address (48 bits)

Offset 0

bit

Length 32

bit

Value Range

hex 00000000

to FFFFFFFF

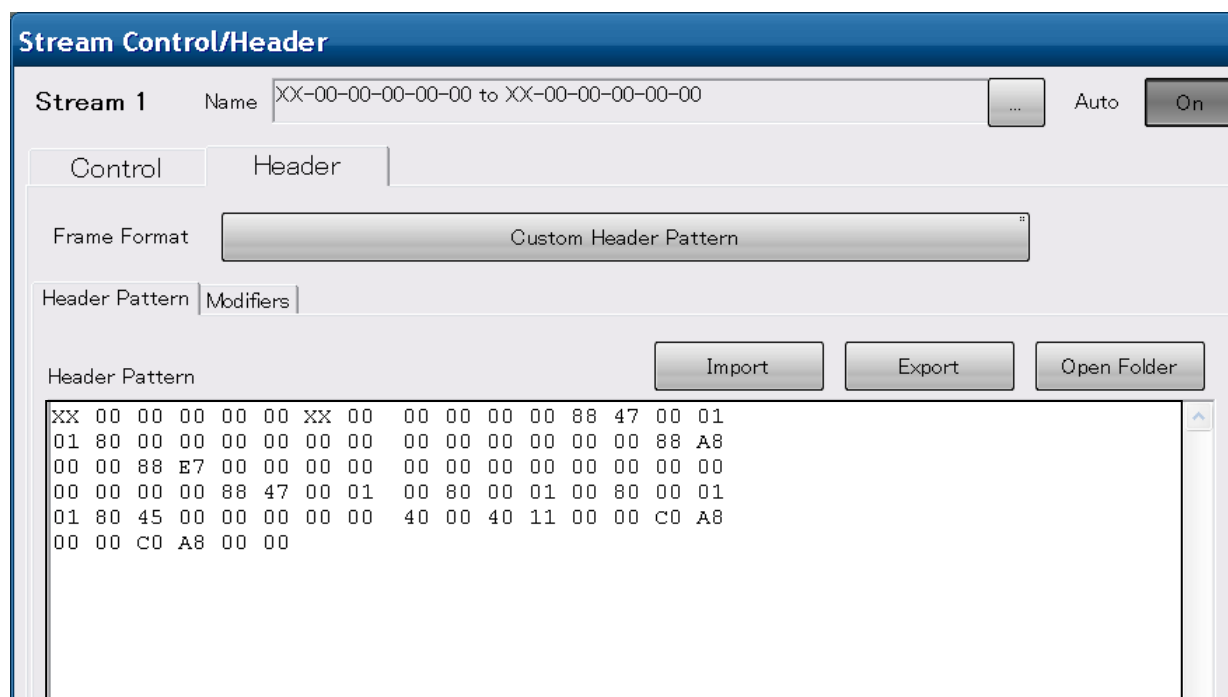
step 1

dec 0 to 4,294,967,295

00000000000000000000000000000000 to 11111111111111111111111111111111

:SOURCE:STReam:HEADer:VARiable{3|4|5}

図3.2.2-5 Stream 画面-Header タブに対応するメッセージ 2



:SOURCE:STReam:HEADer

図3.2.2-6 Stream 画面-Header タブに対応するメッセージ 3

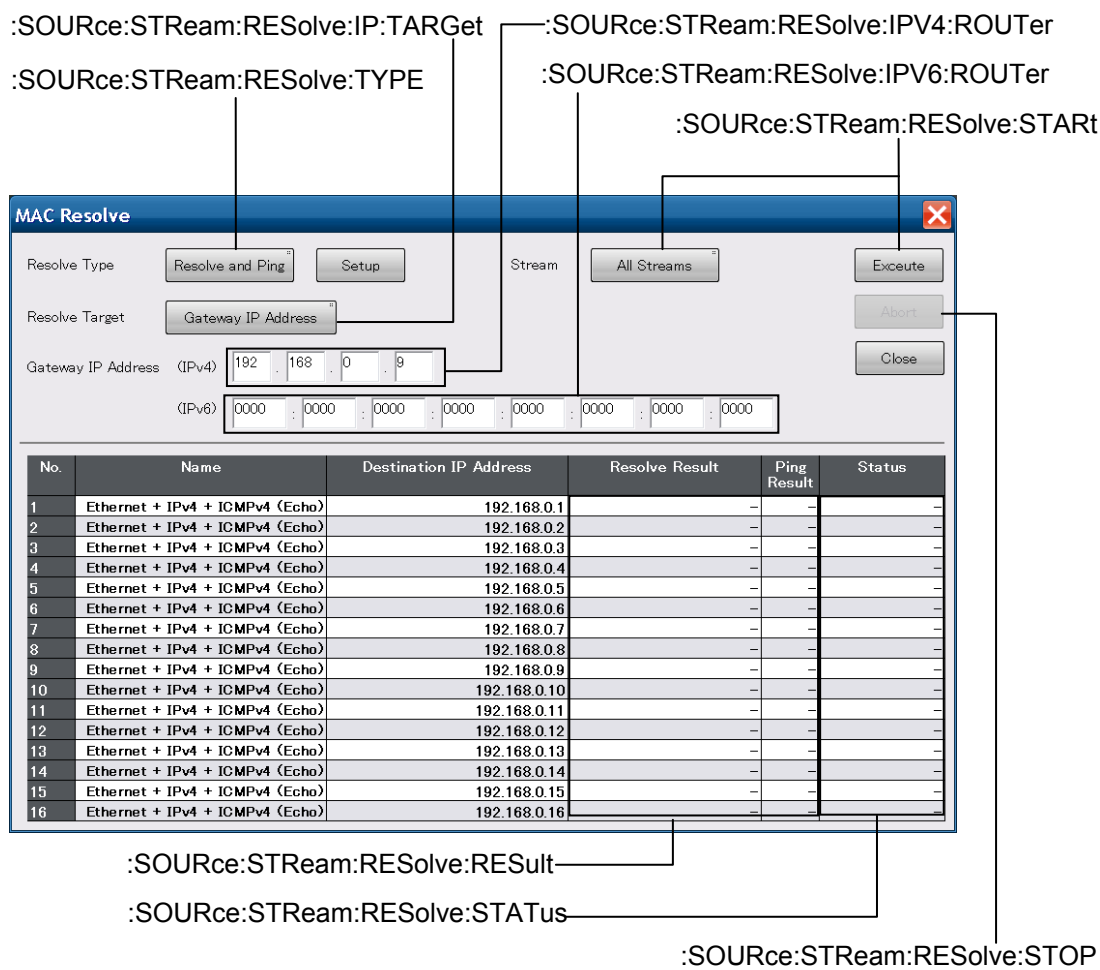


図3.2.2-7 Stream 画面 –MAC Resolve 画面に対応するメッセージ

3
メッセージの詳細

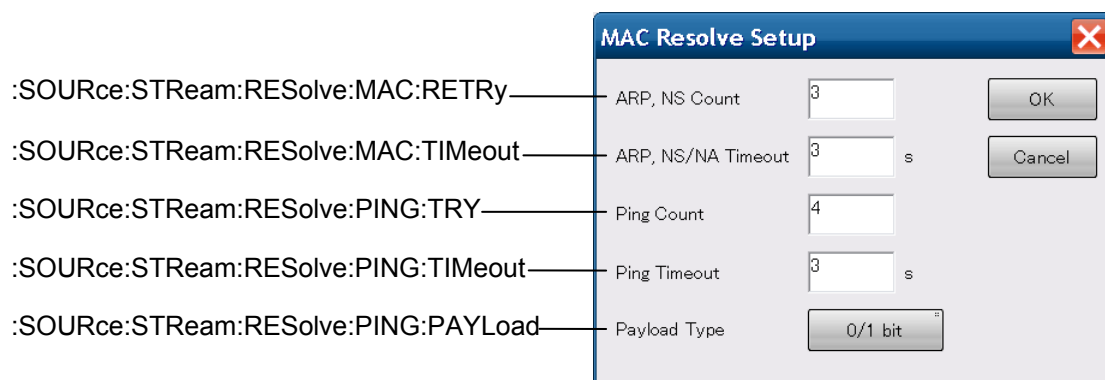


図3.2.2-8 Stream 画面-MAC Setup 画面に対応するメッセージ

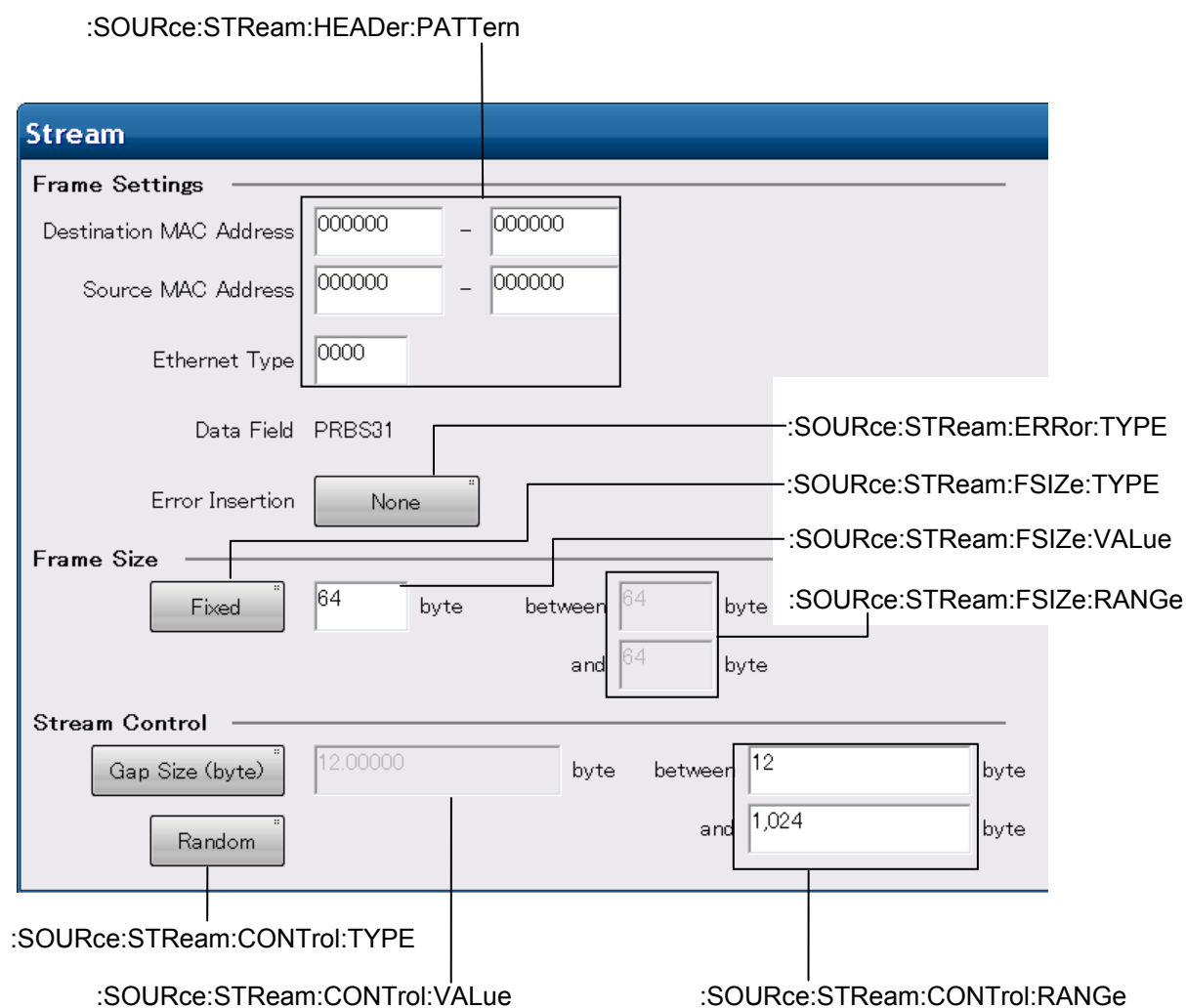


図3.2.2-9 設定エリア[Stream]に対応するメッセージ(Frame BER)

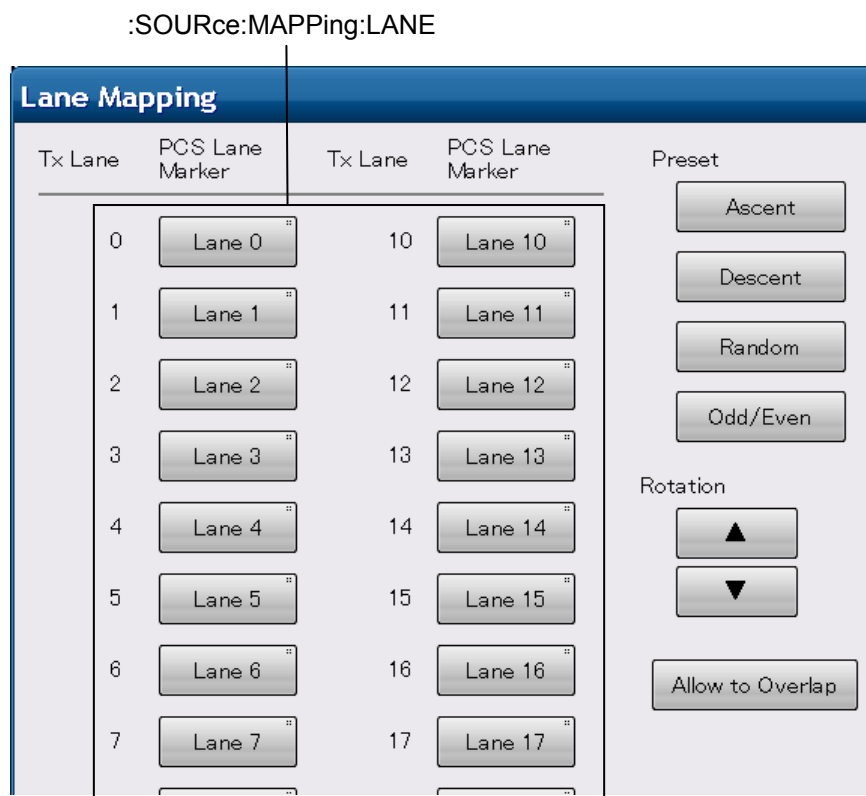


図3.2.2-10 設定エリア[Lane Mapping]に対応するメッセージ

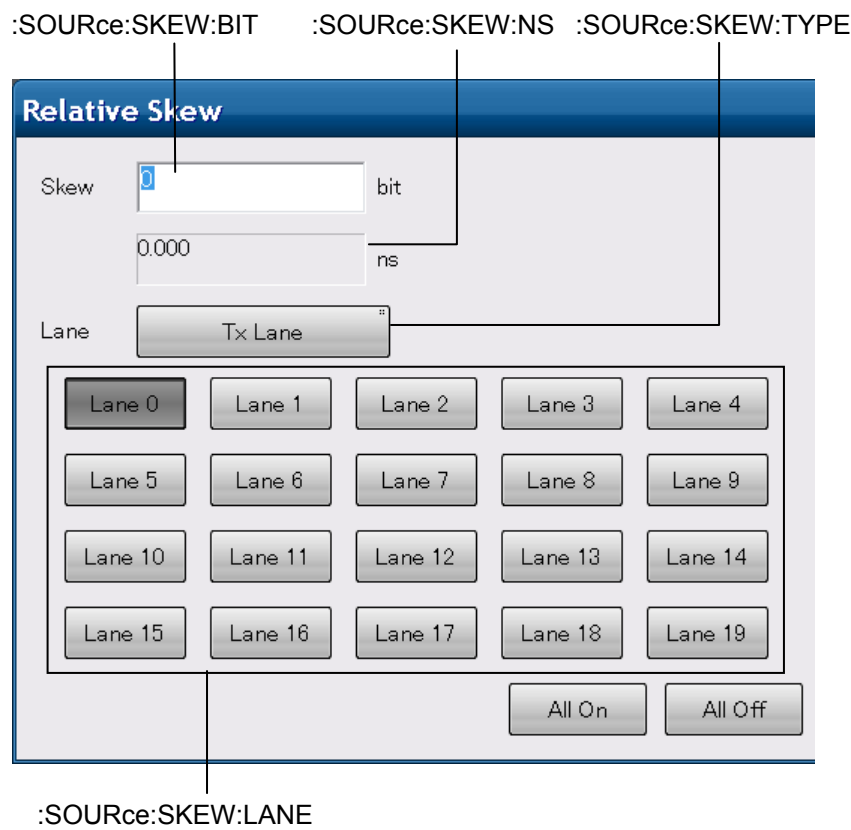


図3.2.2-11 設定エリア[Relative Skew]に対応するメッセージ

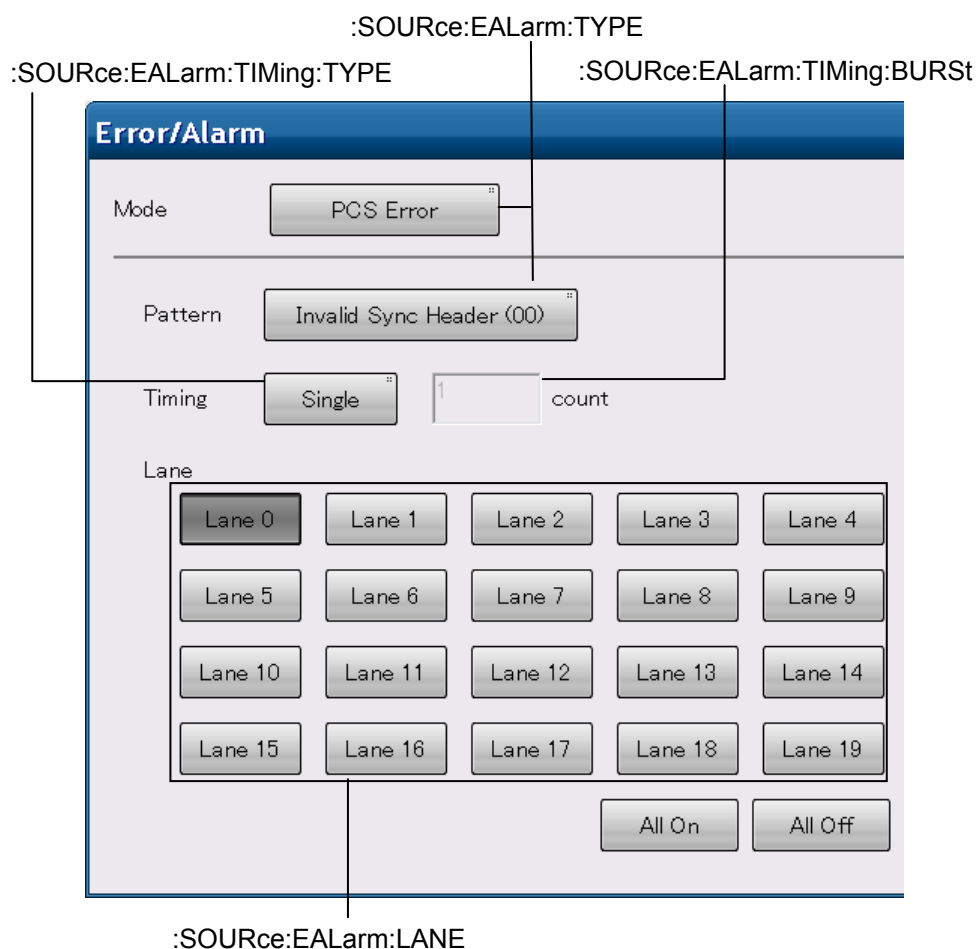


図3.2.2-12 設定エリア[Error/Alarm]に対応するメッセージ

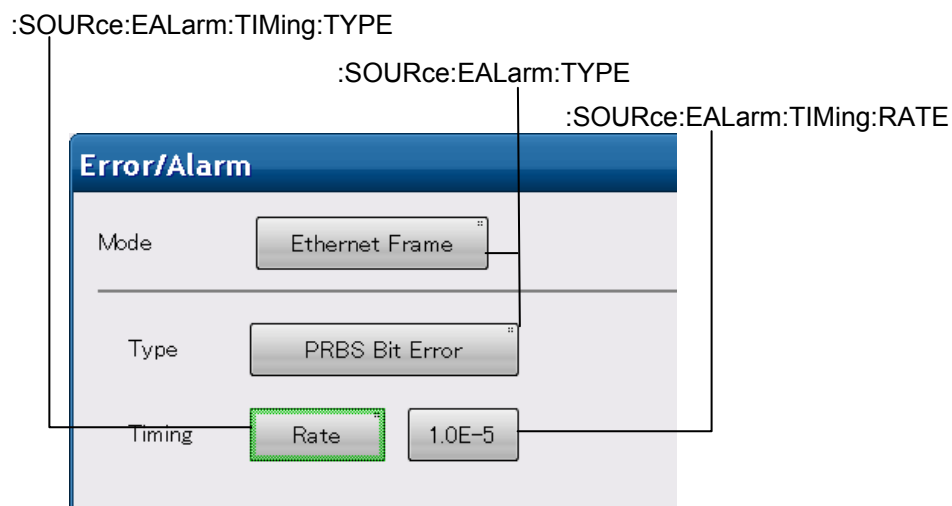


図3.2.2-13 設定エリア[Error/Alarm]に対応するメッセージ (Rate)

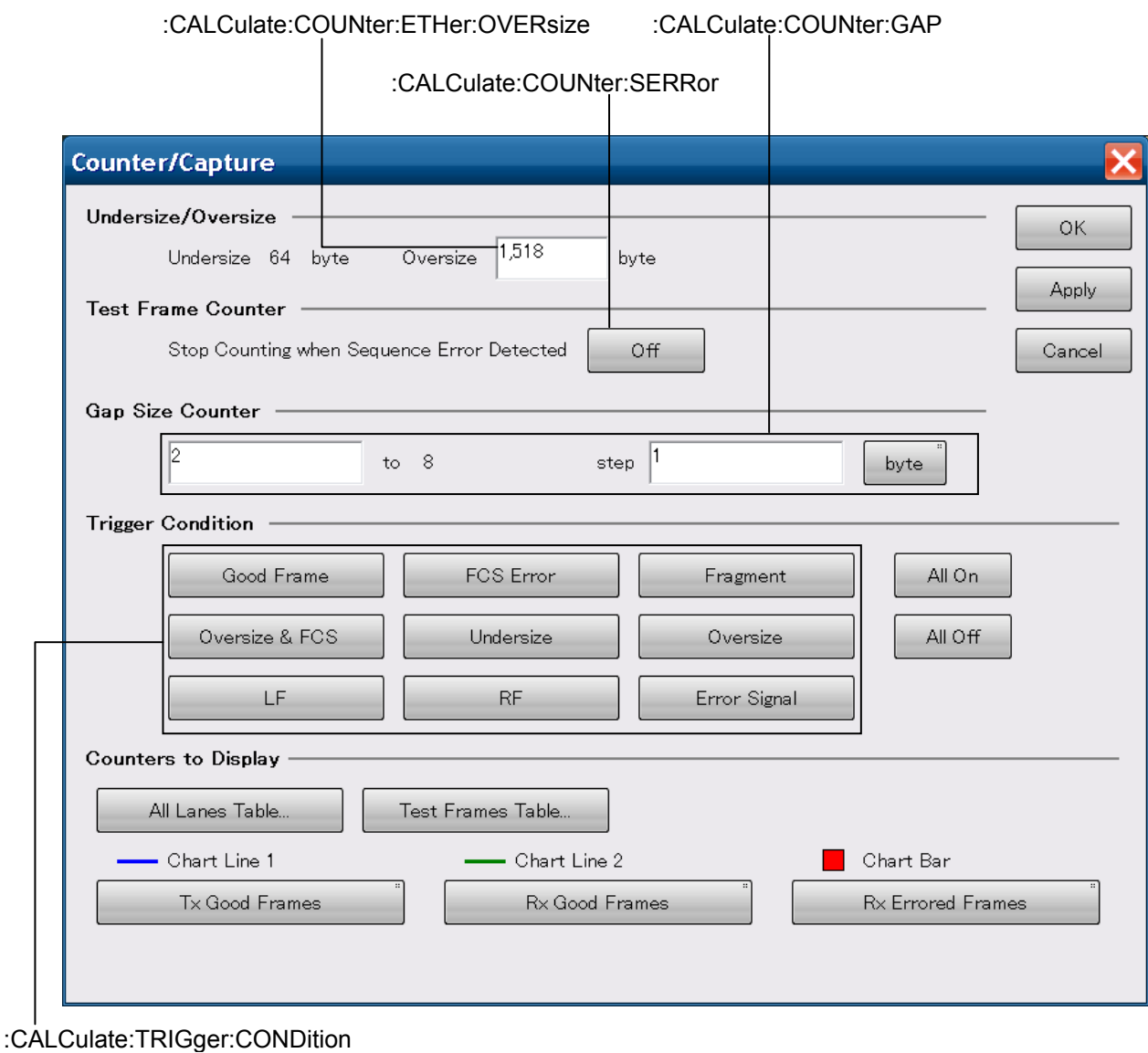


図3.2.2-14 設定エリア[Capture/Counter]に対応するメッセージ

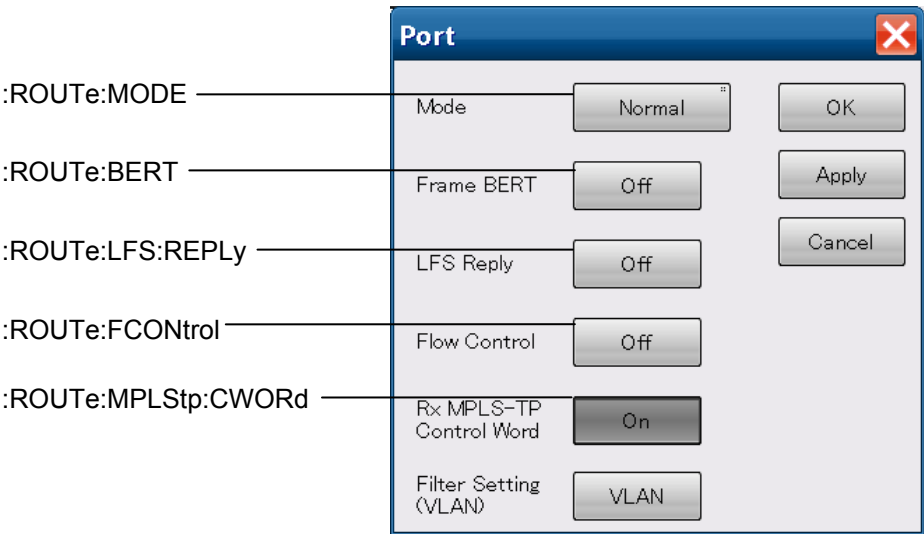


図3.2.2-15 設定エリア[Port]に対応するメッセージ

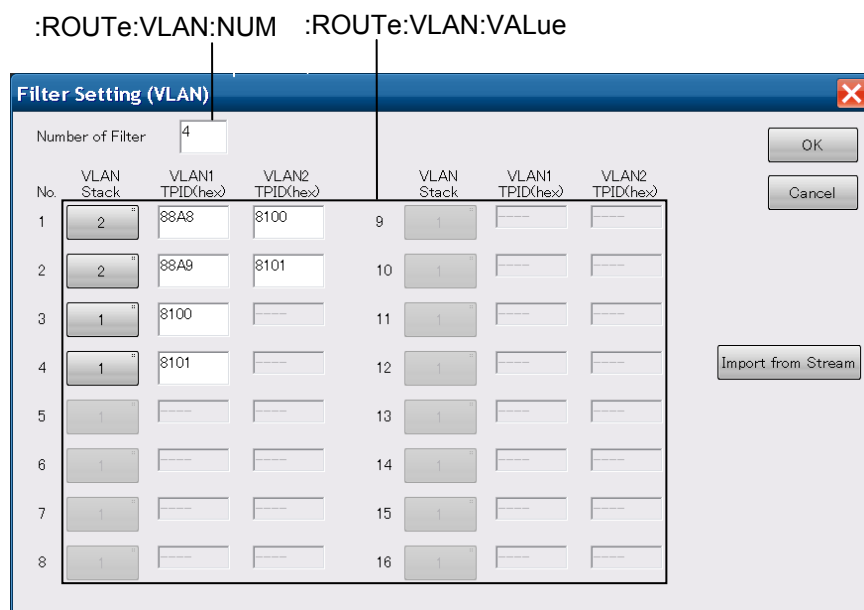


図3.2.2-16 設定エリア[Port] の Filter Setting (VLAN)に対応するメッセージ

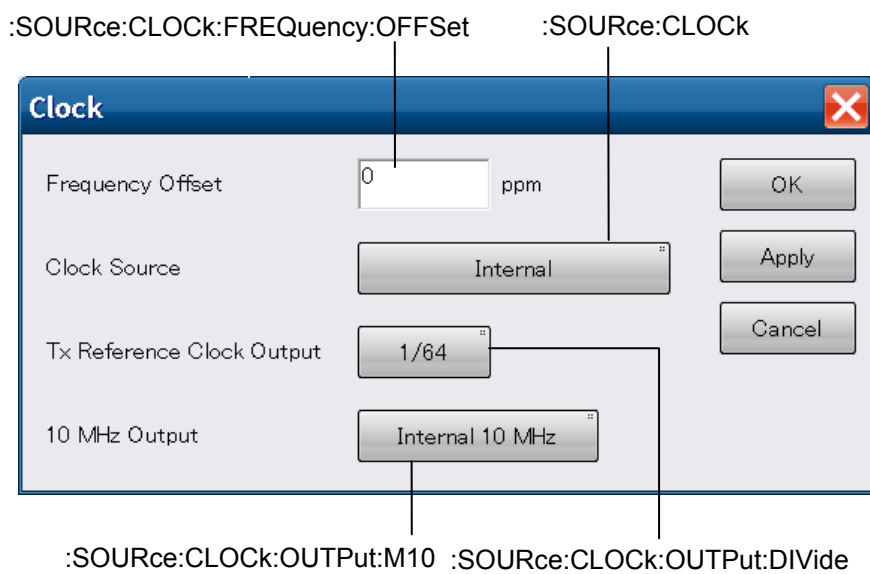


図3.2.2-17 設定エリア[Clock]に対応するメッセージ

:CALCulate:DATA

:CALCulate:DATA:TYPE

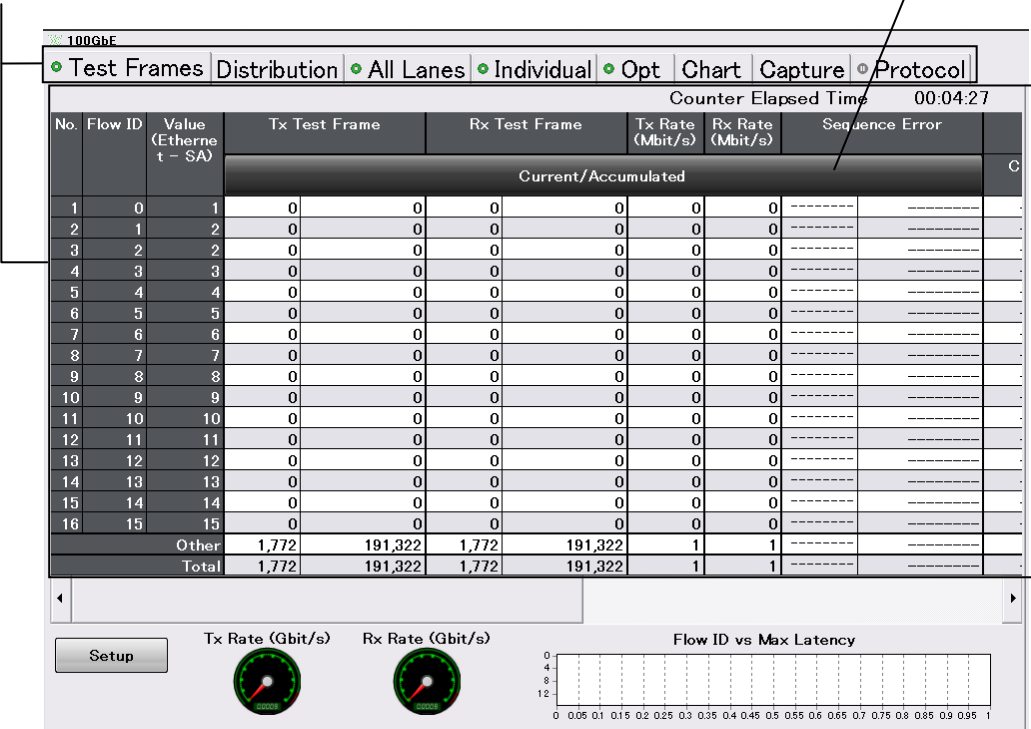


図3.2.2-18 測定結果表示エリアに対応するメッセージ

3
メッセージの詳細

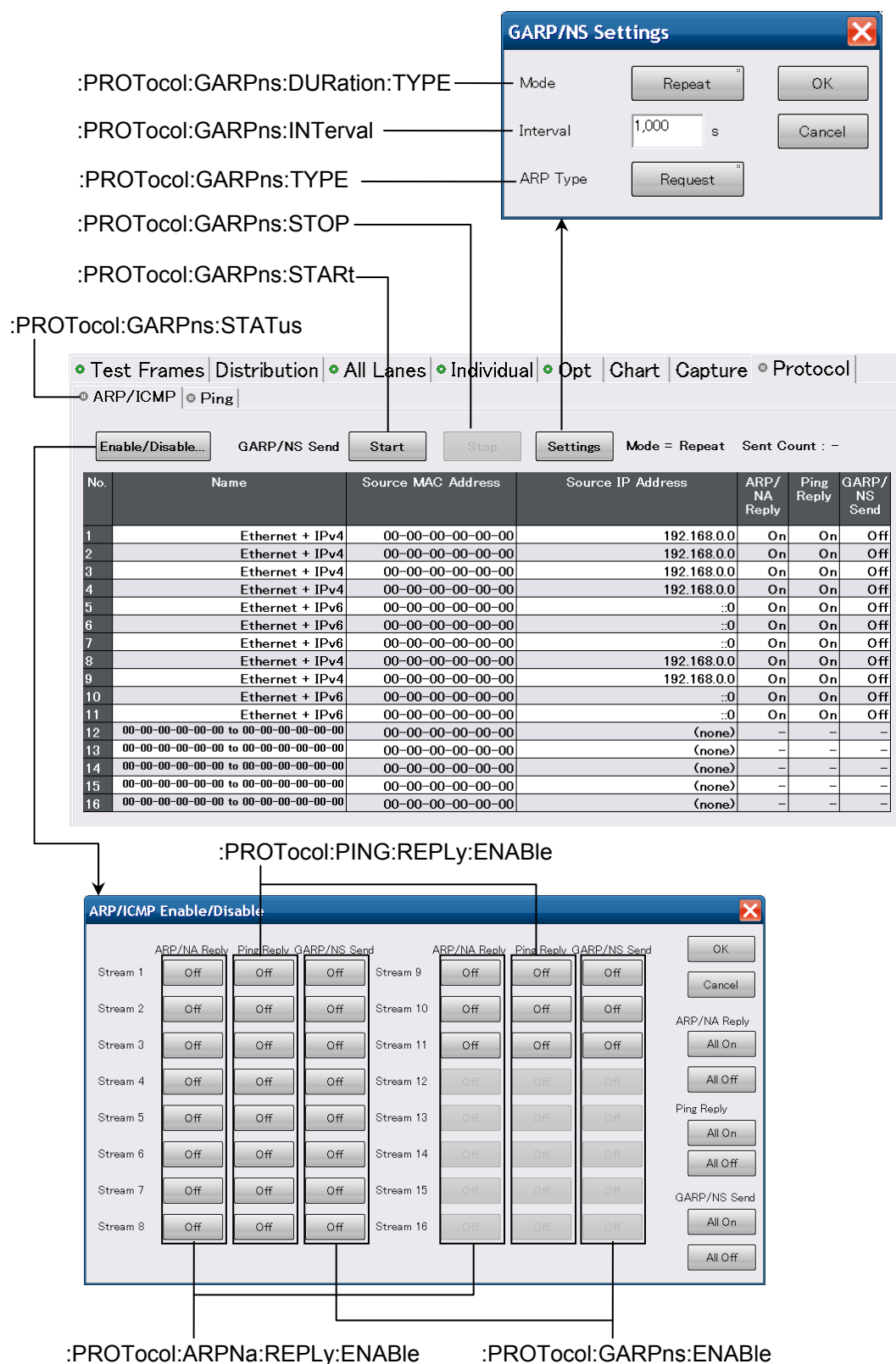


図3.2.2-19 測定結果表示エリアに対応するメッセージ (Protocol-ARP/ICMP)

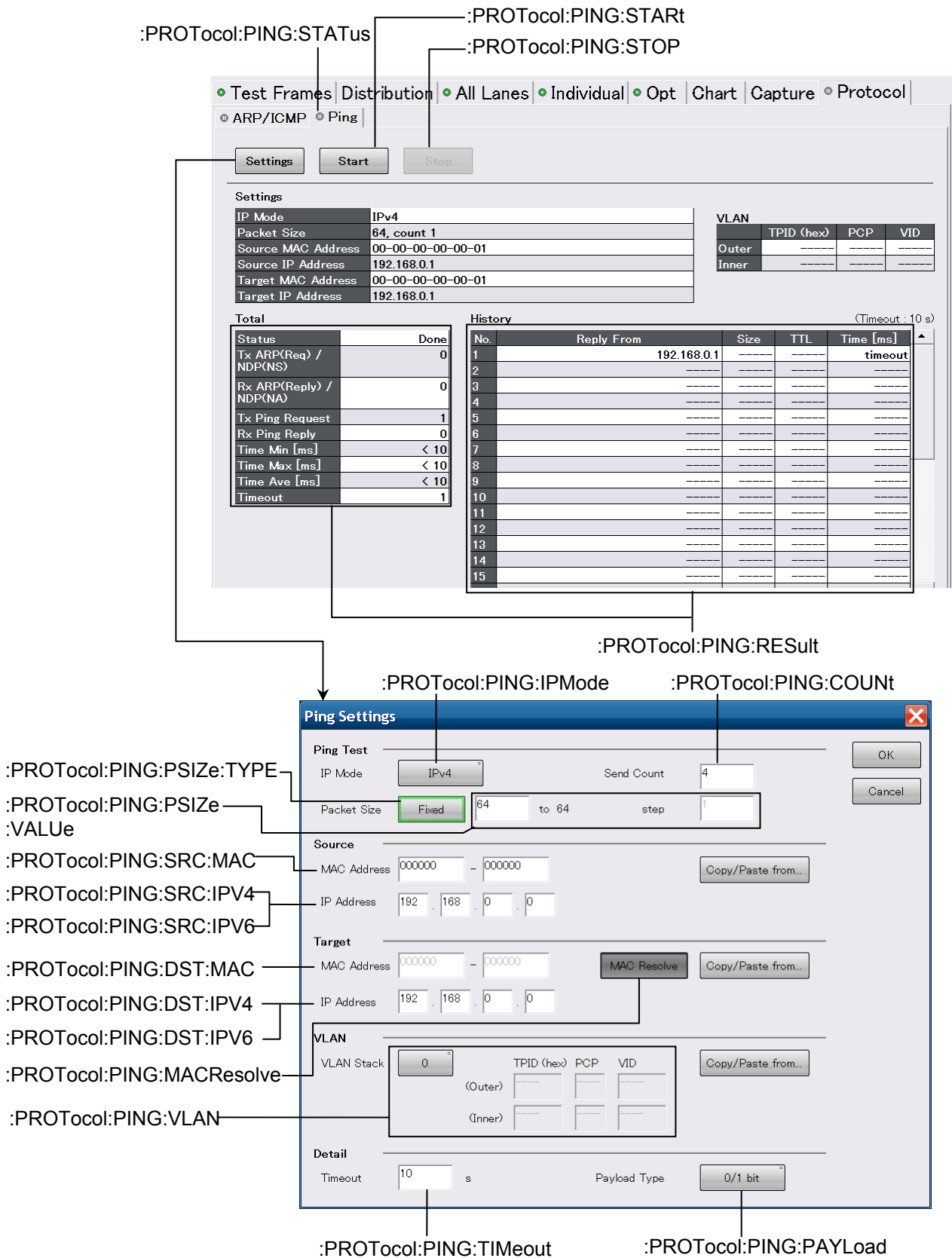


図3.2.2-20 測定結果表示エリアに対応するメッセージ (Protocol-Ping)

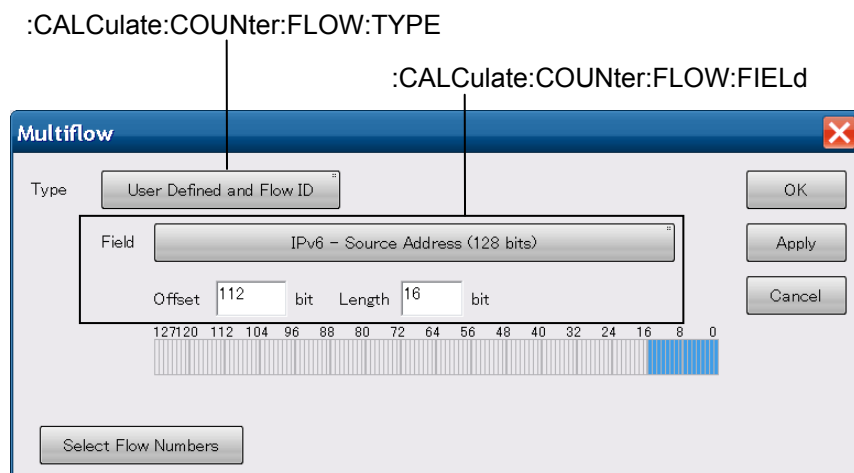


図3.2.2-21 測定結果表示エリア (Test Frames) の MultiFlow 画面に対応するメッセージ

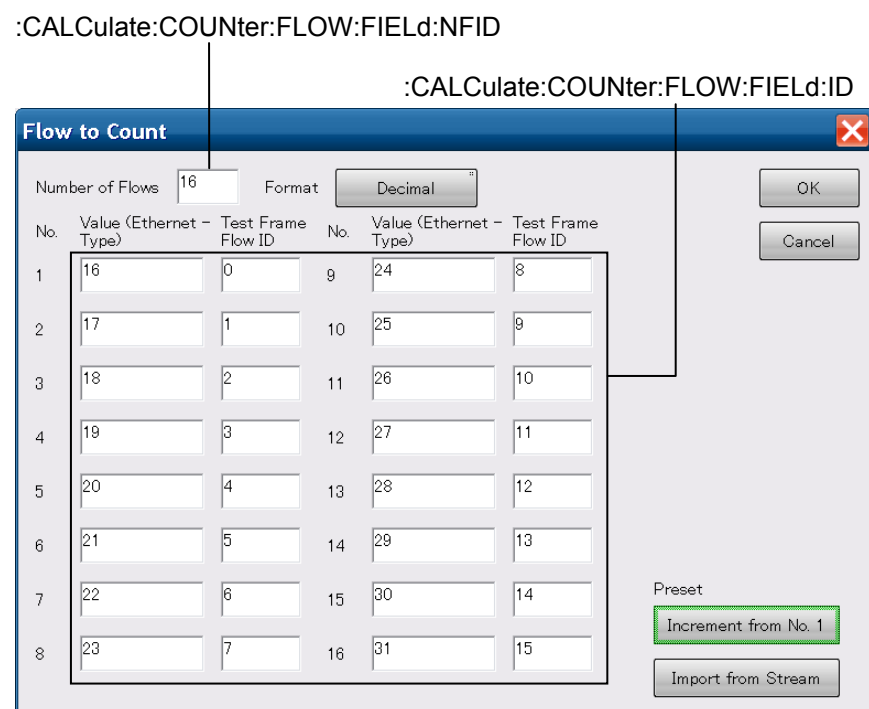


図3.2.2-22 測定結果表示エリア (Test Frames) の Flow to Count 画面に対応するメッセージ

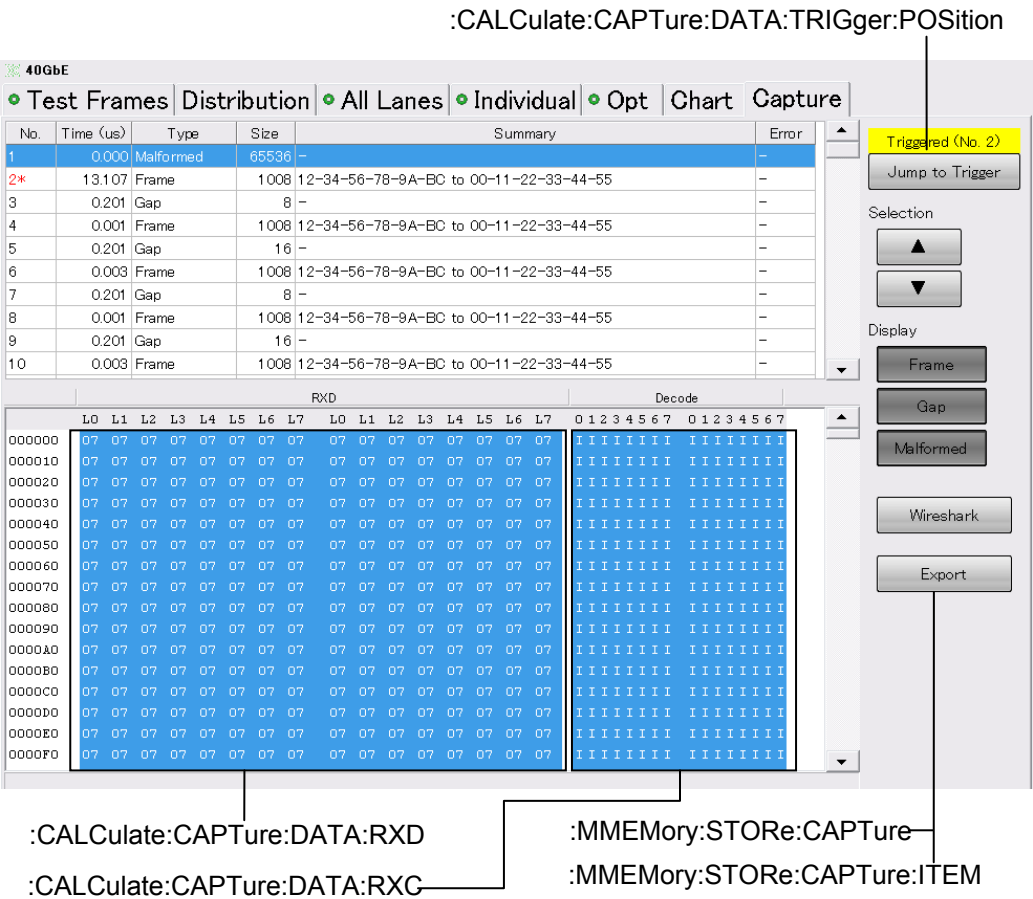


図3.2.2-23 測定結果表示エリアに対応するメッセージ(Capture)

3

メッセージの詳細

3.2.3 OTU3/OTU4に対応するメッセージ

設定エリアの[MDIO], [Transceiver]に対応するメッセージは, 次を参照してください。

図3.2.1-7 設定エリア[Transceiver]に対応するメッセージ

図3.2.2-12 設定エリア[Clock]に対応するメッセージ

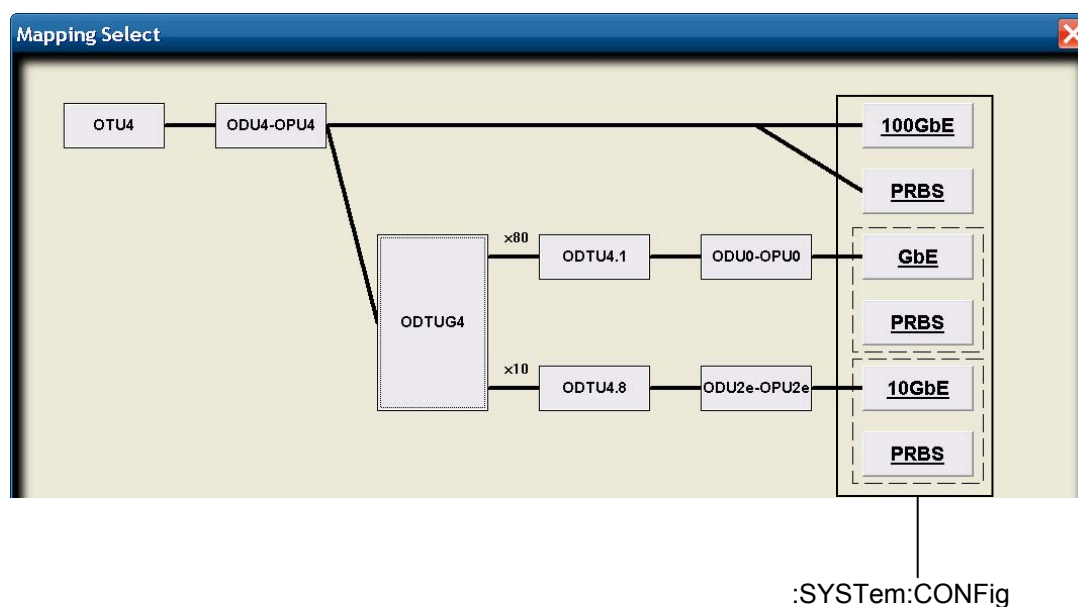


図3.2.3-1 [Mapping Select] に対応するメッセージ (OTU4)

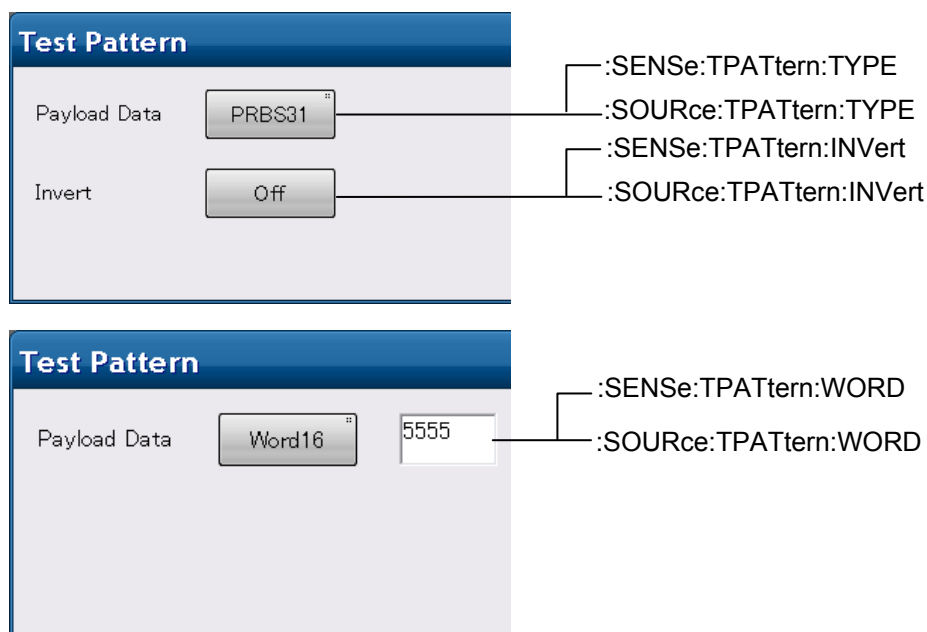


図3.2.3-2 操作エリア[Test Pattern] に対応するメッセージ

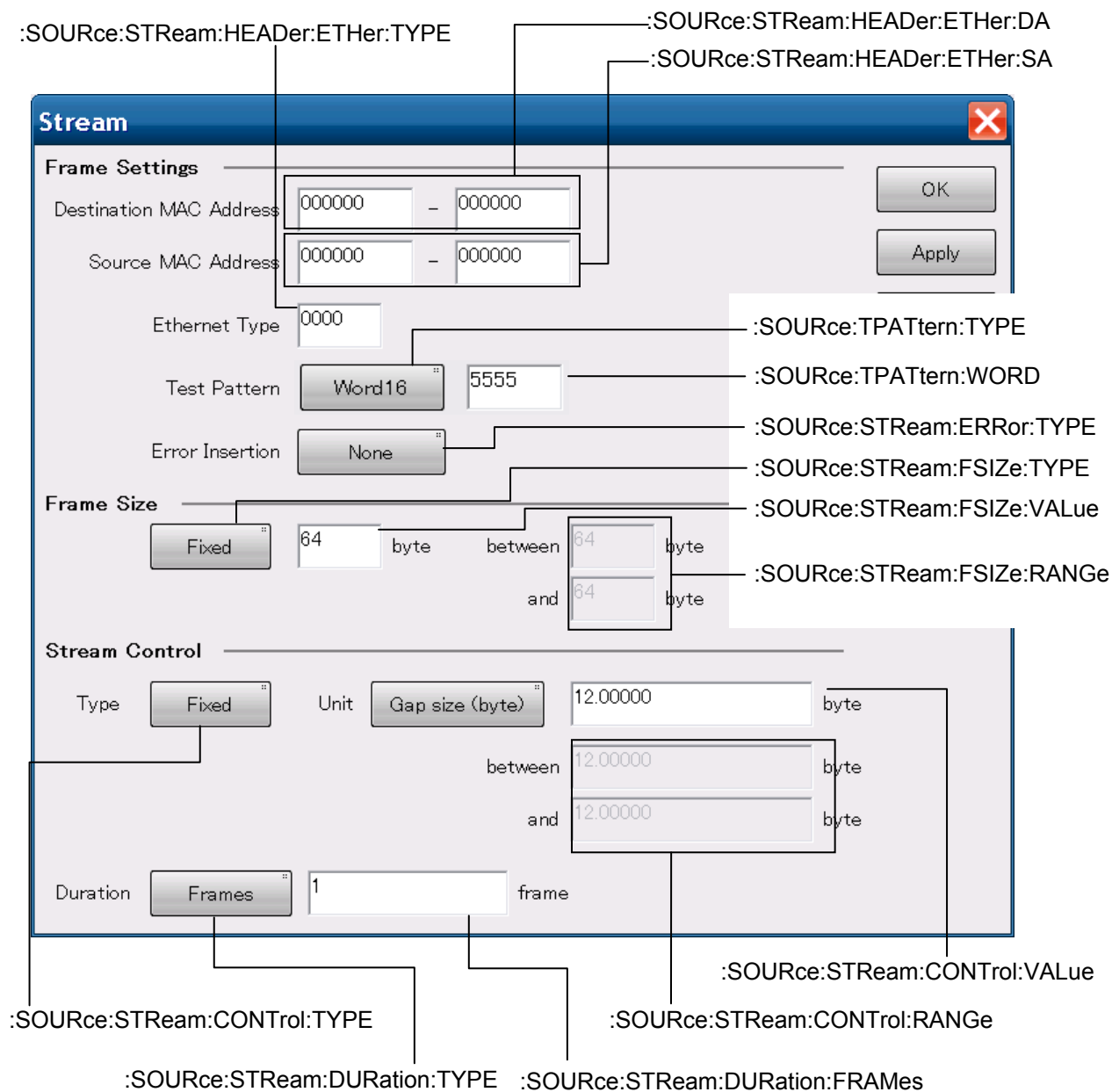


図3.2.3-3 操作エリア[Stream]に対応するメッセージ

:SOURCE:GFP:{PTI|UPI}

Bit	Field
7	PLI=71
6	cHEC
5	EXI=000 PF=0 PTI
4	UPI
3	tHEC
2-0	

:SOURCE:GFP:CSF:REPLacement

図3.2.3-4 設定エリア[GFP-T]に対応するメッセージ

:SOURCE:ODTU:MAIN:TP :SOURCE:ODTU:MAIN:TS :SENSe:ODTU:MAIN:TS :SENSe:ODTU:MAIN:DETeT

TS	+1	+2	+3	+4	+5	+6	+7	+8	+9	+10
0	5	5	5	5	5	5	5	5	1	2
10	3	4	6	7	8	9	10	1	2	3
20	4	6	7	8	9	10	1	2	3	4
30	6	7	8	9	10	1	2	3	4	6
40	7	8	9	10	1	2	3	4	6	7
50	8	9	10	1	2	3	4	6	7	8
60	9	10	1	2	3	4	6	7	8	9
70	10	1	2	3	4	6	7	8	9	10

TS	+1	+2	+3	+4	+5	+6	+7	+8	+9	+10
0	--	--	--	--	--	--	--	--	--	--
10	--	--	--	--	--	--	--	--	--	--
20	--	--	--	--	--	--	--	--	--	--
30	--	--	--	--	--	--	--	--	--	--
40	--	--	--	--	--	--	--	--	--	--
50	--	--	--	--	--	--	--	--	--	--
60	--	--	--	--	--	--	--	--	--	--
70	--	--	--	--	--	--	--	--	--	--

:SOURCE:ODTU:DUMMy:PATtern

図3.2.3-5 設定エリア[TP/TS]に対応するメッセージ

:SOURce:OTN:OH[:L]

OH Preset

OTU4 ODU2e

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	FAS						MFAS	SM			GCC0		RES		JC4	JC1
	F6	F6	F6	28	28	--	--	TTI	--	00	00	00	00	00	--	--
2	RES		PM&TCM	TCM/ACT	TCM6			TCM5			TCM4		FTFL	JC5	JC2	
	00	00	--	00	TTI	--	01	TTI	--	01	TTI	--	01	FTFL	--	--
3	TCM3			TCM2		TCM1			PM			EXP		JC6	JC3	
	TTI	--	01	TTI	--	01	TTI	--	01	TTI	--	00	00	00	--	--
4	GCC1		GCC2		APS/PCC					RES					PSI	RES
	00	00	00	00	00	00	00	00	00	00	00	00	00	00	21	00

OK Apply Cancel Default

図3.2.3-6 操作エリア[OH Preset]に対応するメッセージ

:SOURce:MAPPING:LANE

Lane Mapping

Tx Lane	Logical Lane Marker	Tx Lane	Logical Lane Marker	Preset
0	Lane 2	10	Lane 2	Ascent
1	Lane 3	11	Lane 2	Descent
2	Lane 2	12	Lane 2	Random
3	Lane 2	13	Lane 2	Odd/Even
4	Lane 2	14	Lane 19	Rotation
5	Lane 2	15	Lane 2	▲
6	Lane 2	16	Lane 7	▼
7	Lane 8	17	Lane 2	Allow to Overlap

図3.2.3-7 操作エリア[Lane Mapping]に対応するメッセージ

3

メッセージの詳細

Relative Skew

Skew bit :SOURce:SKEW:BIT

ns :SOURce:SKEW:NS

Lane Tx Lane :SOURce:SKEW:TYPE

Lane 0	Lane 1	Lane 2	Lane 3	Lane 4
Lane 5	Lane 6	Lane 7	Lane 8	Lane 9
Lane 10	Lane 11	Lane 12	Lane 13	Lane 14
Lane 15	Lane 16	Lane 17	Lane 18	Lane 19

:SOURce:SKEW:LANE

図3.2.3-8 設定エリア[Relative Skew]に対応するメッセージ

Error/Alarm

Type Alarm LLD - OOF/LOF :SOURce:EALarm:TYPE

Timing Alternate :SOURce:EALarm:TIMing:TYPE

Alternate Alarm frame :SOURce:EALarm:TIMing:ERRor
(0 us)

Alternate Normal frame :SOURce:EALarm:TIMing:NORMal
(23 us)

Tx Lane

Lane 0	Lane 1	Lane 2	Lane 3	Lane 4
Lane 5	Lane 6	Lane 7	Lane 8	Lane 9
Lane 10	Lane 11	Lane 12	Lane 13	Lane 14
Lane 15	Lane 16	Lane 17	Lane 18	Lane 19

:SOURce:EALarm:LANE

図3.2.3-9 設定エリア[Error/Alarm]に対応するメッセージ (Alternate)

The figure shows a software interface titled "Error/Alarm". It has three main sections: "Type", "Timing", and "Exclude FAS (>2.0E-3)".

- Type:** A dropdown menu is set to "Error OTU4 - Bit All". A line points from this menu to the text `:SOURce:EALarm:TYPE`.
- Timing:** This section contains two dropdown menus. The first is set to "Rate (Random)" and a line points from it to `:SOURce:EALarm:TIMing:TYPE`. The second dropdown is set to "E-2", with a line pointing to `:SOURce:EALarm:TIMing:RATE`. Between these two dropdowns is a text input field containing the value "1.0".
- Exclude FAS (>2.0E-3):** A dropdown menu is set to "Off". A line points from this menu to `:SOURce:EALarm:FAS:EXCLude`.

図3.2.3-10 設定エリア[Error/Alarm]に対応するメッセージ (Rate)

The figure shows a software interface titled "Error/Alarm". It has two main sections: "Type" and "Timing".

- Type:** A dropdown menu is set to "Error OTU4 - SM-BIP8". A line points from this menu to `:SOURce:EALarm:TYPE`.
- Timing:** A dropdown menu is set to "Burst". A line points from this menu to `:SOURce:EALarm:TIMing:TYPE`. Below this dropdown is a text input field labeled "Burst" containing the value "1". A line points from this field to `:SOURce:EALarm:TIMing:BURSt`.

図3.2.3-11 設定エリア[Error/Alarm]に対応するメッセージ (Burst)

The figure shows a software interface titled "Error/Alarm". It has two main sections: "Type" and "Timing".

- Type:** A dropdown menu is set to "Error OTU4 - Uncorrectable Error". A line points from this menu to `:SOURce:EALarm:TYPE`.
- Timing:** A dropdown menu is set to "Single". Below this dropdown is a text input field labeled "Sub-row #" containing the value "1". A line points from this field to `:SOURce:EALarm:SUBRow`.

図3.2.3-12 設定エリア[Error/Alarm]に対応するメッセージ (Subrow)

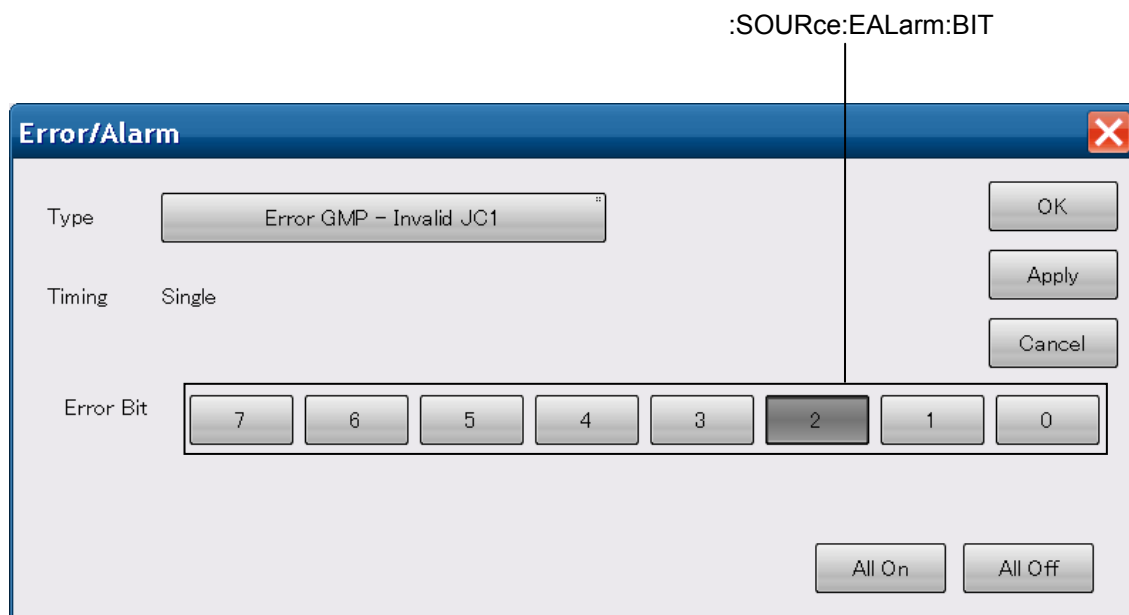


図3.2.3-13 設定エリア[Error/Alarm]に対応するメッセージ (Bit)

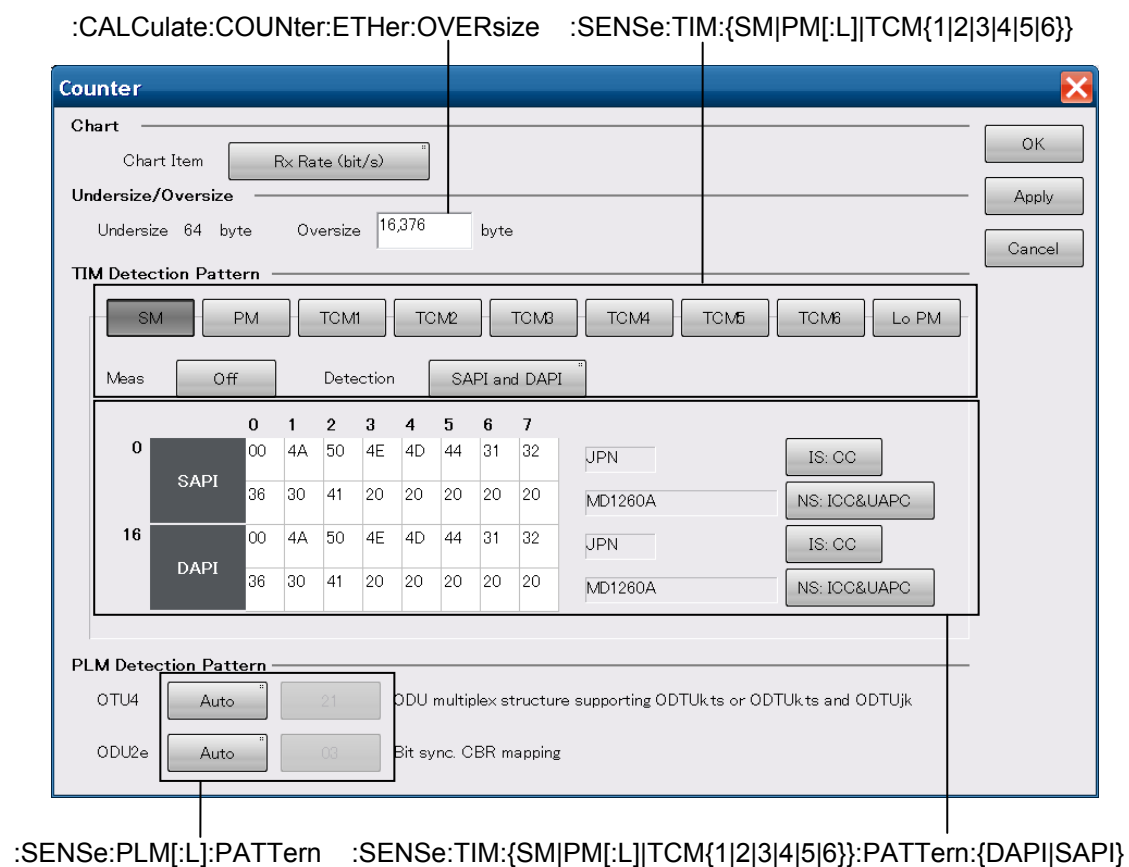


図3.2.3-14 設定エリア[Counter]に対応するメッセージ

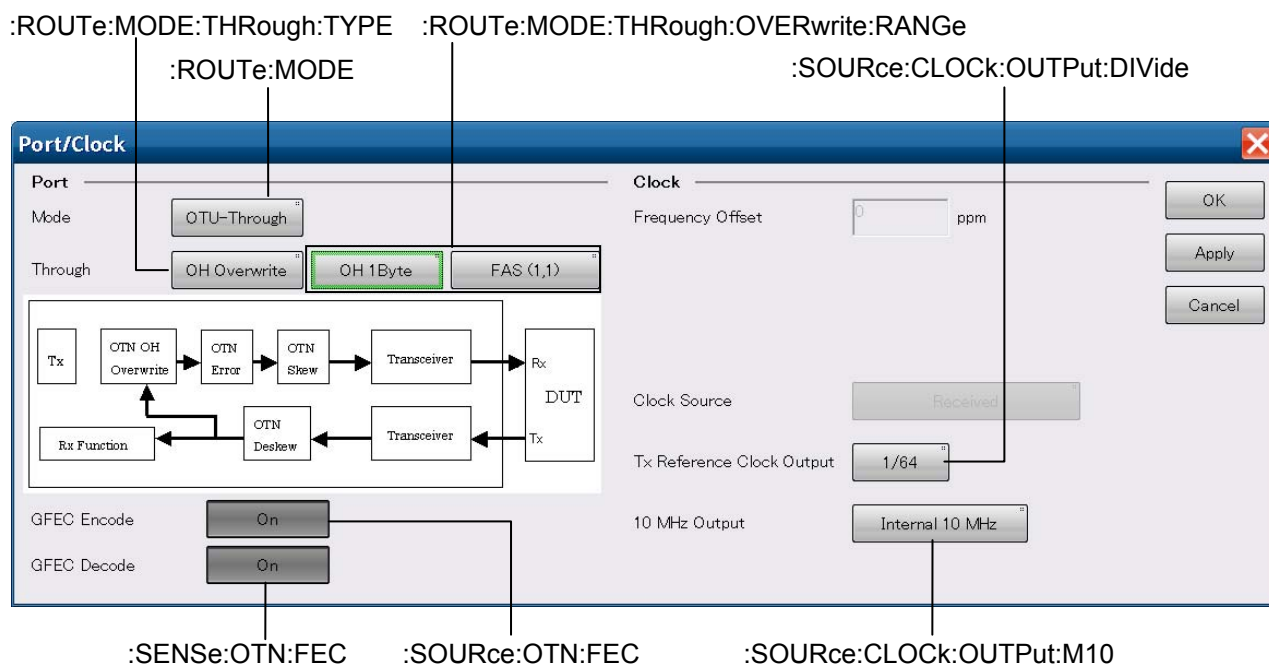


図3.2.3-15 設定エリア[Port/Clock]に対応するメッセージ 1

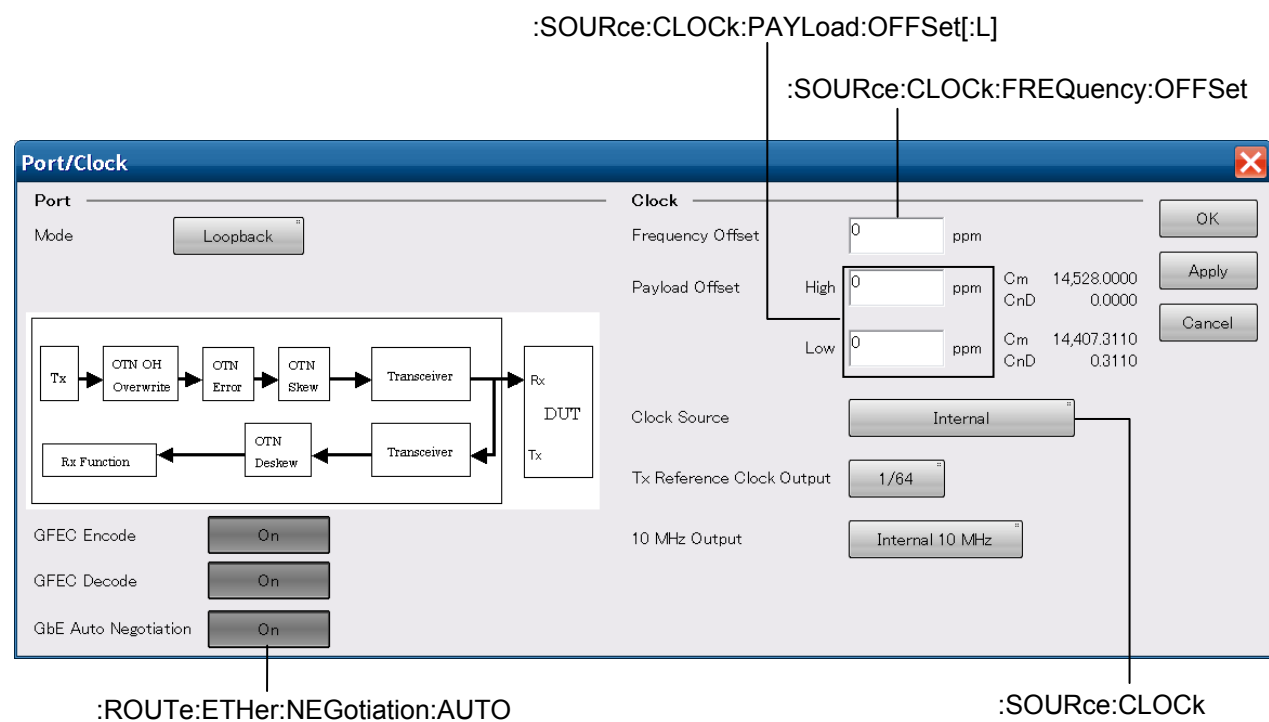


図3.2.3-16 設定エリア[Port/Clock]に対応するメッセージ 2

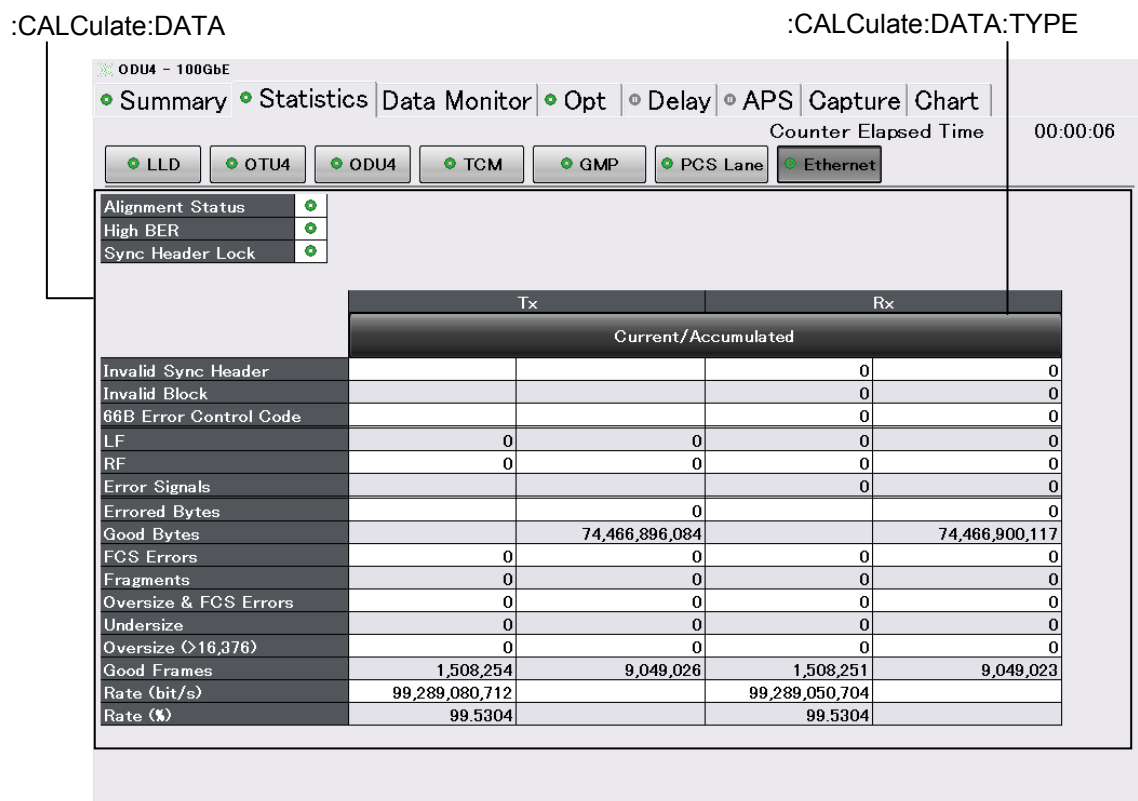


図3.2.3-17 測定結果表示エリア (Statistics)に対応するメッセージ

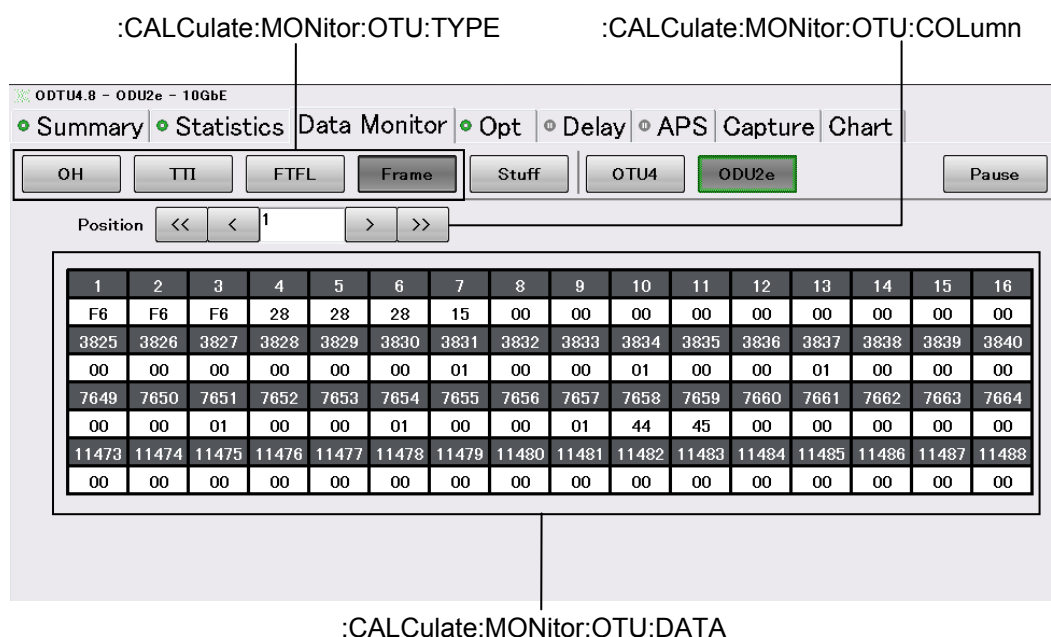


図3.2.3-18 測定結果表示エリア (Data Monitor)に対応するメッセージ

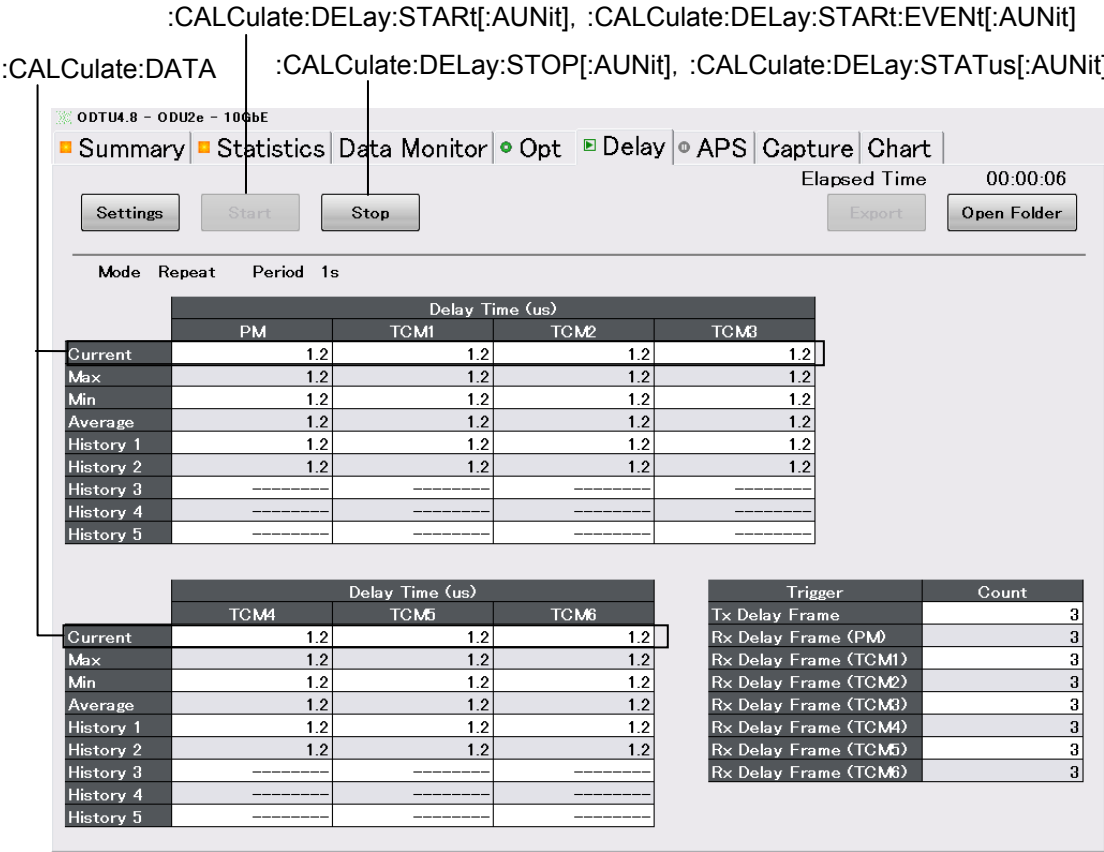


図3.2.3-19 測定結果表示エリア(Delay)に対応するメッセージ

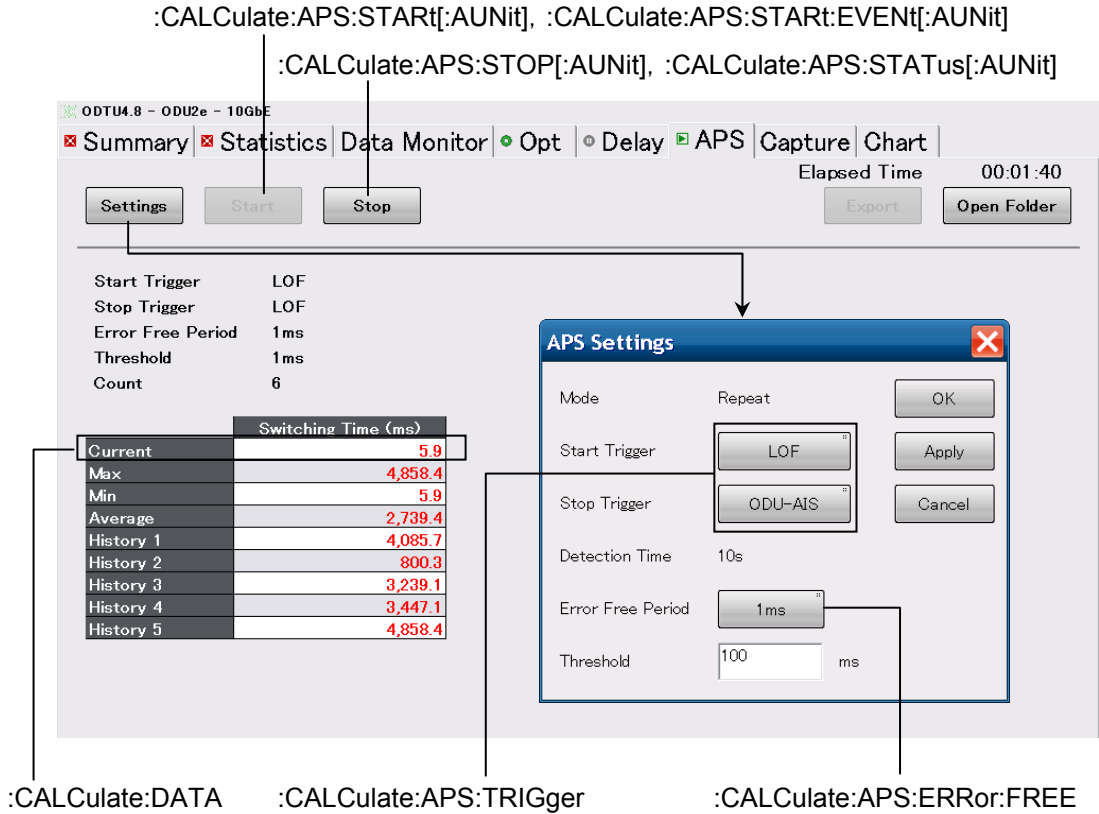


図3.2.3-20 測定結果表示エリア(APS)に対応するメッセージ

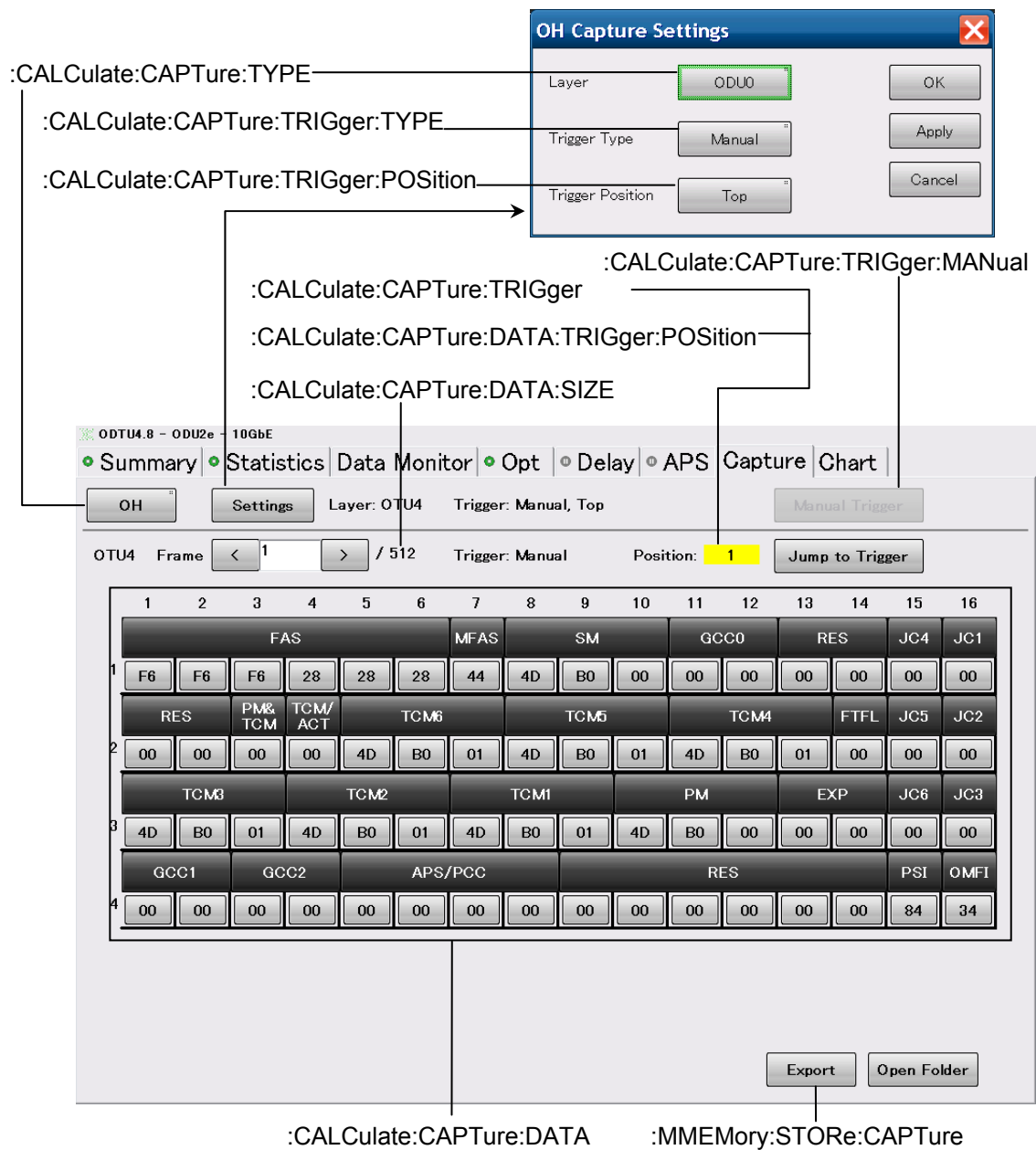


図3.2.3-21 測定結果表示エリア(Capture)に対応するメッセージ 1

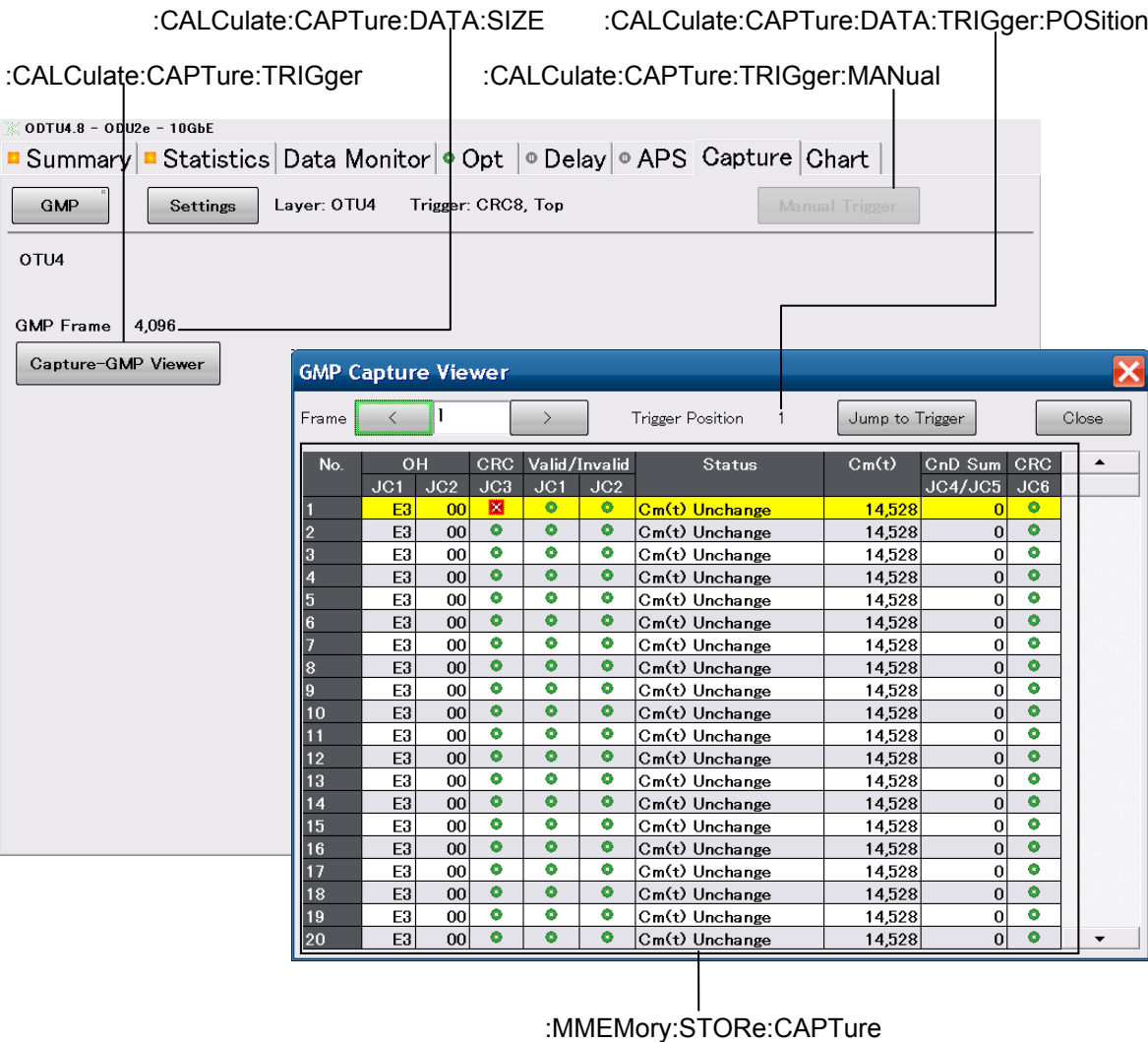


図3.2.3-22 測定結果表示エリア(Capture)に対応するメッセージ 2

3

メッセージの詳細

3.2.4 No Frameに対応するメッセージ

設定エリアの[MDIO], [Clock], [Transceiver]に対応するメッセージは、次を参照してください。

図3.2.1-6 設定エリア[MDIO]に対応するメッセージ

図3.2.1-7 設定エリア[Transceiver]に対応するメッセージ

図3.2.2-12 設定エリア[Clock]に対応するメッセージ

設定エリアの[Counter]に対応するメッセージはありません。

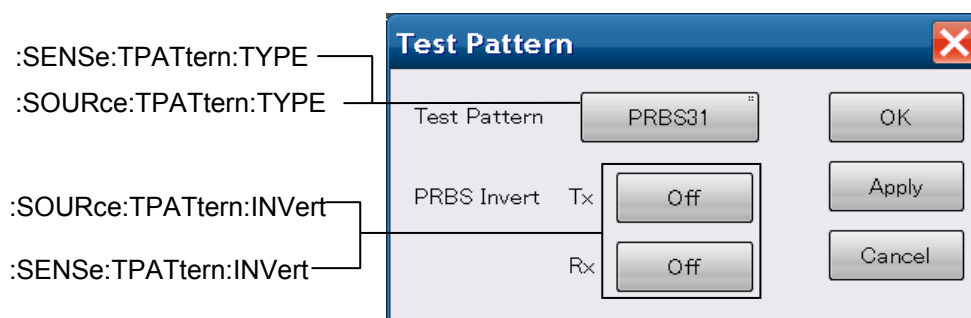


図3.2.4-1 設定エリア[Test Pattern]に対応するメッセージ

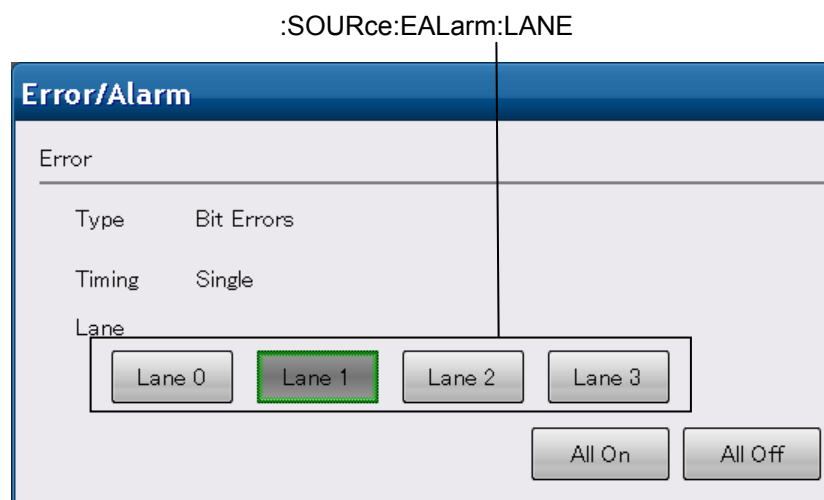


図3.2.4-2 設定エリア[Error/Alarm]に対応するメッセージ

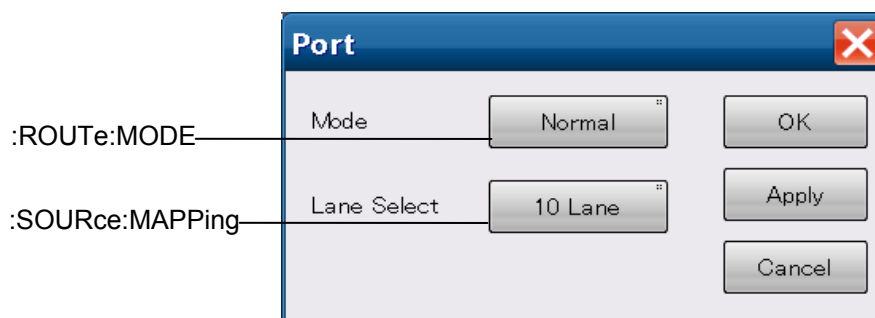


図3.2.4-3 設定エリア[Port]に対応するメッセージ(100GbE No Frame/OTU4 No Frame)

3.2.5 Add-on機能

以下の Add-on 機能の画面操作に対応するメッセージはありません。

- RFC2544
- CFP MDIO Analysis
- Lambda Measure
- Service Disruption

また、Add-on 機能の測定実行中はリモート制御できません。

3.2.6 対応するパネル操作が無いメッセージ

対応するパネル操作が無いメッセージは次のとおりです。

クエリだけのメッセージは、クエションマークを省略して記載しています。

コマンドとクエリの両方があるメッセージは、コマンドのみを記載しています。

表3.2.6-1 パネル操作が無いメッセージ(共通コマンド)

コマンド	説明
*CLS	標準イベントレジスタと出力キューのクリア
*ESE	標準イベントイネーブルレジスタの設定と問い合わせ
*ESR	標準イベントレジスタの問い合わせ
*OPC	処理完了時のステータスバイトのビット設定、ステータスバイトのビット 0 の問い合わせ
*RST	工場出荷時の機器設定に変更
*SRE	サービスリクエストイネーブルレジスタの設定と問い合わせ
*STB	ステータスバイトレジスタの問い合わせ
*WAI	実行中の処理が終了するまで、次のコマンド実行を待機

表3.2.6-2 パネル操作が無いメッセージ

コマンド	説明
:CALCulate:APS:START:EVENT[:AUNit]	APS 測定開始イベントの問い合わせ
:CALCulate:CAPture:DATA:SIZE	キャプチャしたデータサイズの問い合わせ
:CALCulate:CAPture:START:EVENT	キャプチャ開始イベントの問い合わせ
:CALCulate:COUNter:START:EVENT[:AUNit]	カウンタ開始イベントの問い合わせ
:CALCulate:DELay:START:EVENT[:AUNit]	遅延時間測定開始イベントの問い合わせ
:MMEMory:LOG:FNAME	最新の Log ファイル名の問い合わせ
:PROTOCOL:PING:START:EVENT	Ping 試験開始イベントの問い合わせ
:SOURCE:EALarm:START:EVENT[:AUNit]	エラー/アラーム開始イベントの問い合わせ
:SOURCE:STReam:RESolve:START:EVENT	ARP 測定開始イベントの問い合わせ
:SOURCE:STReam:START:EVENT[:AUNit]	遅延時間測定開始イベントの問い合わせ
:SYSTem:ERRor	コマンドエラー発生の問い合わせ
:SYSTem:STATus	アプリケーションの起動状態の問い合わせ
:SYSTem:TERMination	終端文字(ターミネータ)の設定と問い合わせ
:SYSTem:VERSion	SCPI バージョンの問い合わせ

3.2.7 リモート制御できないパネル操作

「3.2.1 共通操作に対応するメッセージ」から「3.2.5 No Frame に対応するメッセージ」までの画面に、リモートコマンドが表示されていないボタン操作、およびデータの読み取りはリモート制御できません。

このほかにリモート操作できないパネル操作は次のとおりです。

- Selector 画面の Utility タブのすべての操作
- Selector 画面の Multi Port タブの[Go Slave]ボタン
- Selector 画面の [Shut Down]ボタン
- アプリケーション画面のタブ操作による表示の切り替え
- システムメニュー、設定エリアの設定画面表示
- アプリケーション画面の[Chart]タブの波形データ取得
- Add-on 機能のすべての操作
- OTU4 アプリケーション Data Monitor タブの[GMP]画面操作、およびデータ取得
- OTU4 アプリケーション Capture タブ(GMP)の[GMP Capture Viewer]画面の取得

ただし、測定結果のファイル保存はできます。

3.3 メッセージの説明

3.3.1 共通コマンド

*CLS [Clear Status]

機能

- (1) *CLS 共通コマンドは、次をクリアします。

- 標準イベントステータスレジスタ
- 出力キュー

この結果、ステータスバイトレジスタのビット 5 が 0 になります。

*CLS によって、次のレジスタの設定値は変化しません。

- 標準イベントステータスイネーブルレジスタ
- サービスリクエストイネーブル
- 実行状態レジスタ
- 機器固有ステータスレジスタ

- (2) *CLS 共通コマンドは、プログラムメッセージのターミネータの後で、クエリの前に*CLS コマンドを送出したときに、ステータスバイトレジスタをクリアします。このときは、出力キューにあるすべての未読み取りメッセージをクリアします。

文法

*CLS

ESE [Event Status Enable]*機能**

*ESE は標準イベントステータス「イネーブル」レジスタの値を設定します。標準イベントステータスレジスタでマスクしたいビットを0にします。*ESE? は標準イベントステータスイネーブルレジスタの値を問い合わせます。

文法

*ESE <integer>

*ESE?

<integer> = bit0 + bit1 + bit2 + bit3 + bit4 + bit5 + bit6 + bit7

レジスタのビットと 10 進数の対応は以下のようになります。

bit7 : $2^7 = 128$	電源投入
bit6 : $2^6 = 64$	未使用
bit5 : $2^5 = 32$	コマンドエラー
bit4 : $2^4 = 16$	実行エラー
bit3 : $2^3 = 8$	デバイス固有エラー
bit2 : $2^2 = 4$	未使用
bit1 : $2^1 = 2$	未使用
bit0 : $2^0 = 1$	操作の完了
範囲	0～255

レスポンスデータ

<integer>

*ESE?: 標準イベントステータスイネーブルレジスタのビットの合計値 0～255

使用例

ビット 0～2, 6～7 をマスクして, ビット 3～5 を許可するときの例を示します。

*ESE 56

*ESE?

>56

*ESR [Standard Event Status Register]

機能

*ESR は標準イベントステータスレジスタの値を問い合わせます。この値は*ESEで設定した 8 ビットと論理積をとった値となります。レジスタの値を読み取ると、標準イベントステータスレジスタはクリアされます。

文法

*ESR?

レスポンスデータ

<integer>

*ESR?: 標準イベントステータスレジスタのビットの合計値 0～255

使用例

標準イベントステータスレジスタの値を読み取り、実行エラーとコマンドエラーが発生したことを示すときの例を示します。

*ESR?

>48

どのようなエラーが発生したかの詳細は、:SYSTem:ERRor?コマンドで調べることができます。

*IDN [Identification]

機能

本器のメーカー名、形名、シリアル番号、バージョン番号を問い合わせます。

文法

*IDN?

レスポンスデータ

ANRITSU,MD1260A,<serial_number>,<version>

<serial_number>

本器のシリアル番号（10 桁の数字）が返ります。

<version>

本器にインストールされているソフトウェアのバージョン番号が X.XX.XX の形式で返ります。

使用例

*IDN?

>ANRITSU,MD1260A,6200123456,1.00.16

*OPC [Operation Complete]

機能

*OPC は、実行中のすべての処理が完了したときに、標準イベントステータスレジスタの操作完了ビット（ビット 0）を 1 に設定します。

*OPC?は、標準イベントステータスレジスタの bit0（OPCビット）を問い合わせます。OPCビットは、処理が実行中のときは 0 に、処理が完了したときは 1 になります。以下の項目が発生した後は、*OPC と*OPC?による操作完了待ちは無効になります。

- ・ 電源が投入されたとき
- ・ IEEE488.2 インタフェース上で DCL または SCL を受信したとき
- ・ *CLS コマンドを受信したとき
- ・ *RST コマンドを受信したとき
- ・ すべての実行中の処理が完了したとき

注:

本器は、メッセージの処理の途中で、別のメッセージを処理できません。前に送信したメッセージが実行中のときは、*OPC?の処理が待たされます。前に送信したメッセージが完了した後に*OPC?が処理されるため、レスポンスデータは常に 1 となります。メッセージの実行終了を確認する方法については「2.6 メッセージの実行を確認する」も参照してください。

文法

*OPC

*OPC?

レスポンスデータ

1

使用例

*OPC?

>1

RST [Reset]*機能**

本器の設定を工場出荷時の状態に戻します。

ただし、以下の内容は工場出荷時の状態に戻りません。

- GPIB アドレス, IP アドレスなどの Windows の設定
- トランシーバの設定 (:SOURce:TRANsceiver:EMPHasis:FIRSt
:SOURce:TRANsceiver:EMPHasis:PRE
:SOURce:TRANsceiver:EMPHasis:SECond
:SOURce:TRANsceiver:VOD
:SENSe:TRANsceiver:EQUalizer:CONTRol
:SENSe:TRANsceiver:EQUalizer:DCGain)
- レスポンスターミネータの設定 (:SYSTem:TERMination)

注:

複数の MD1260A が接続されている場合に、*RST を送信するとすべての MD1260A が初期化されます。

MD1260A を個別に初期化する場合は、:MMEMory:INITializeを使用します。

文法

*RST

SRE [Service Request Enable]*機能**

*SRE はサービスリクエストイネーブルレジスタの値を設定します。設定する値は 8 ビット 2 進数に相当する 0～255 です。ステータスバイトレジスタをマスクするビットを 0 にします。

文法

*SRE <integer>

*SRE?

<integer>= bit0 + bit1 + bit2 + bit3 + bit4 + bit5 + bit6 + bit7

bit7 : $2^7 = 128$	未使用
bit6 : $2^6 = 64$	サービスリクエストイネーブルレジスタの場合は常に 0
bit5 : $2^5 = 32$	イベントステータスレジスタサマリ
bit4 : $2^4 = 16$	出力キューが空でないことを示す
bit3 : $2^3 = 8$	未使用
bit2 : $2^2 = 4$	エラー、イベントキューが空でないことを示す
bit1 : $2^1 = 2$	未使用
bit0 : $2^0 = 1$	未使用

範囲 0～255

STB [Status Byte]*機能**

*SRE?は、サービスリクエストイネーブルレジスタの値を問い合わせます。

*STB?は、ステータスバイトレジスタを読み取ります。

文法

*STB?

レスポンスデータ

<integer>

*STB?: ステータスバイトレジスタのビットの合計値 0～255

使用例

ビット 2 と 5 以外をマスクするときの例を示します。

*SRE 36

*SRE?

> 36

TRG [Trigger]*機能**

:CALCulate:COUNter:STARt[:AUNit]と同じ動作をします。

複数ユニットを接続している場合は、すべてのユニットのカウンタを開始します。

文法

*TRG

WAI [Wait to Continue]*機能**

*WAI の前に送信したメッセージの処理が完了するまで、次のメッセージの実行を待たせます。

注:

本器は、メッセージの処理の途中に、別のメッセージを処理できません。

本器のメッセージは、前に送信したメッセージの処理が完了した後に実行されるため、*WAIを使用する必要はありません。

文法

*WAI

3.3.2 機器固有コマンド

:CALCulate:APS:ERRor:FREE

機能

OTU3, OTU4 アプリケーションの APS 測定のエラーフリー判断条件を設定/問い合わせます。

文法

:CALCulate:APS:ERRor:FREE <NR1>

:CALCulate:APS:ERRor:FREE?

<NR1>: Error Free Period (ms)

1 | 10 | 100 | 200 | 300 | 400 | 500 | 600 | 700 | 800 | 900 | 1000

レスポンスデータ

<NR1>

使用例

:CALCulate:APS:ERRor:FREE 200

:CALCulate:APS:ERRor:FREE?

> 200

:CALCulate:APS:STARt[:AUNit]

機能

OTU3, OTU4 アプリケーションの APS 測定を開始します。

すでに APS を測定している場合は、再スタートとなります。

:AUNit を追加すると、接続しているすべてのユニットの APS 測定を同時に開始します。

文法

:CALCulate:APS:STARt[:AUNit]

使用例

エラー・アラーム付加の開始を待つ場合は以下のようにします。

1. **:CALCulate:APS:STARt:EVENT?** を送信します。
開始イベントがクリアされます。
2. **:CALCulate:APS:STARt** を送信します。
APS 測定が開始します。
3. **:CALCulate:APS:STARt:EVENT?** を送信します。
レスポンスデータが 1 (開始) になるまで送信を繰り返します。

:CALCulate:APS:START:EVENT[:AUNit]**機能**

OTU3, OTU4 アプリケーションの APS 測定の開始イベントを問い合わせます。

文法

:CALCulate:APS:START:EVENT[:AUNit]?

レスポンスデータ

0|1

0: APS 測定の開始前

本コマンドで読み取った後にレスポンスデータが 0 になります。

1: APS 測定が開始済み

ヘッダーに:AUNit を付けた場合は、接続しているすべてのユニットの APS 測定が開始すると 1 となります。

レスポンスデータ 1 を読み取ると、イベントの値は 0 にクリアされます。

使用例

:SOURce:EALarm:START[:AUNit]の使用例を参照してください。

:CALCulate:APS:STATUS[:AUNit]**機能**

OTU3, OTU4 アプリケーションの APS 測定の状態を問い合わせます。

文法

:CALCulate:APS:STATUS[:AUNit]?

レスポンスデータ

0|1

0: APS 測定停止中

1: APS 測定中

ヘッダーに:AUNit を付けた場合は、接続したユニットのどれか 1 つ以上が APS 測定を開始すると 1、すべてのユニットが停止すると 0 になります。

使用例

:SOURce:EALarm:STOP[:AUNit]の使用例を参照してください。

:CALCulate:APS:STOP[:AUNit]**機能**

OTU3, OTU4 アプリケーションの APS 測定を停止します。

:AUNit を追加すると、接続しているすべてのユニットの APS 測定を停止します。

文法

:CALCulate:APS:STOP[:AUNit]

使用例

APS 測定の停止を確認するには、次のようにします。

1. **:CALCulate:APS:STOP** を送信します。
APS 測定が停止します。
2. **:CALCulate:COUNTER:STATus?** を送信します。
レスポンスデータが 0 (停止) になるまで送信を繰り返します。

:CALCulate:APS:TRIGger**機能**

APS 測定のトリガを設定/問い合わせます。

文法

:CALCulate:APS:TRIGger <start>,<stop>

<start>,<stop>=<CHAR>

トリガ	<CHAR>
LOF	LOF
OOF	OOF
ODU-AIS	AIS_ODU
ODU-OCI	OCI
ODU-LCK	LCK
SM-BIP8	SM_BIP8
PM-BIP8	PM_BIP8
Any Error (すべての項目の OR)	ANY_ERROR

使用例

APS 測定の停止を確認するには、次のようにします。

1. **:CALCulate:APS:STOP** を送信します。
APS 測定が停止します。
2. **:CALCulate:COUNTER:STATus?** を送信します。
レスポンスデータが 0 (停止) になるまで送信を繰り返します。

:CALCulate:CAPTure:DATA**機能**

OTU3, OTU4 アプリケーションのキャプチャデータ を問い合わせます。
 キャプチャ対象が GMP の場合は、本コマンドでデータを問い合わせできません。
 :MMEMory:STORe:CAPTureを使用してください。

文法

```
:CALCulate:CAPTure:DATA? {<row>,<column>} |  
{<frame_num>[,<offset>,<length>]}
```

<row>,<column>,<frame_num>,<offset>,<length> = <NR1>

キャプチャ対象が OH の場合

<row>: 行番号

<column>: 列番号

キャプチャ対象が Frame の場合

<frame_num>: フレーム番号

<offset>: オフセット(バイト)

<length>: 長さ(バイト)

レスポンスデータ

<STRING>

カンマ区切りの 16 進数文字列です。データが無効のバイトは..で表示されます。
 一度に取得できるレスポンスデータの最大バッファサイズは 65536 バイトです。
 キャプチャ対象が OH の場合、指定したバイト位置のマルチフレームシーケンス
 データです。データ長は:CALCulate:CAPTure:DATA:SIZEで指定した値で
 す。

:CALCulate:CAPTure:DATA:RXC**機能**

40GbE, 100GbE アプリケーションの制御信号(RXC)のキャプチャデータ を問い
 合わせます。
 オフセットと長さをビット数で問い合わせるデータの位置を指定します。

文法

```
:CALCulate:CAPTure:DATA:RXC? <offset>,<length>
```

<offset>,<length> = <NR1>

<offset>: オフセット(バイト)

<length>: 長さ(バイト)

レスポンスデータ

<BINARY>

以下の場合には:SYSTem:ERRorコマンドに対して、-220,"Parameter Error"を
 返します。

- ・ バッファサイズ*を超えた<length>を指定した場合

- ・ :CALCulate:CAPTure:DATA:SIZE を超える範囲を<offset>, <length>で指定した場合

*: バッファサイズ (レスポンスデータとして返すことができる最大文字数) は, 65536 バイトです。このため, <length>で指定できる RXC データの最大数は $65536 \cdot 2 = 65534$ ビットとなります。

RXC の詳細は, IEEE802.3ba Clause 81 を参照してください。

:CALCulate:CAPTure:DATA:RXD

機能

40GbE, 100GbE アプリケーションの通信データ(RXD)の キャプチャデータを問い合わせます。

オフセットと長さをビット数で問い合わせるデータの位置を指定します。

文法

:CALCulate:CAPTure:DATA:RXD? <offset>, <length>

<offset>, <length> = <NR1>

<offset>: オフセット(バイト)

<length>: 長さ(バイト) 1~32767

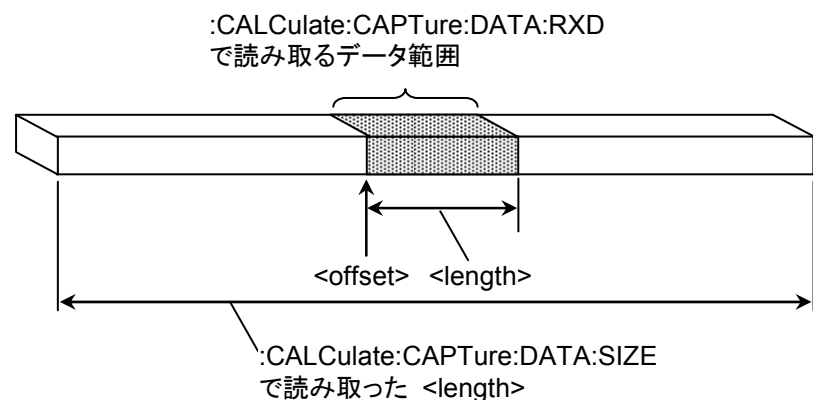
レスポンスデータ

<HEX>

以下の場合には:SYSTem:ERRorコマンドに対して, -220, "Parameter Error"を返します。

- ・ バッファサイズ*を超えた<length>を指定した場合
- ・ :CALCulate:CAPTure:DATA:SIZE を超える範囲を<offset>, <length>で指定した場合

*: バッファサイズ (レスポンスデータとして返すことができる最大文字数) は, 65536 バイトです。このため, <length>で指定できる RXD データの最大数は $(65536 \cdot 2) / 2 = 32767$ バイトとなります。



RXD の詳細は, IEEE802.3ba Clause 81 を参照してください。

使用例

1000～1015 バイトを読み取ります。

```
:CALCulate:CAPTure:DATA:RXD? 1000,16
```

```
> #H002523FE0A0B55230000000000000111A
```

:CALCulate:CAPTure:DATA:SIZE**機能**

40GbE, 100GbE アプリケーションの場合、通信データ(RXD)のキャプチャデータサイズを RXD のバイト数で問い合わせます。

OTU3, OTU4 アプリケーションの場合、キャプチャデータのサイズをフレーム数で問い合わせます。

文法

```
:CALCulate:CAPTure:DATA:SIZE?
```

レスポンスデータ

<NR1>: キャプチャデータのサイズ(0~131072)

キャプチャ実行中は 0 になります。

:CALCulate:CAPTure:DATA:TRIGger:POSition**機能**

40GbE, 100GbE アプリケーションの場合、キャプチャデータ中のトリガ位置を、キャプチャデータの先頭からのオフセット（バイト）で問い合わせます。トリガ条件がフレーム・シーケンスの場合、トリガ位置はフレーム・シーケンスの最終バイトです。

OTU3, OTU4 アプリケーションの場合、キャプチャデータのトリガ位置を、キャプチャデータの先頭からのフレーム数で問い合わせます。

文法

```
:CALCulate:CAPTure:DATA:TRIGger:POSition?
```

レスポンスデータ

<NR1>: トリガ位置

:CALCulate:CAPTure:START

機能

キャプチャを開始します。

文法

:CALCulate:CAPTure:START

注:

複数の MD1260A に対してキャプチャの同時開始はできません。

使用例

キャプチャの開始を待つ場合は以下のようにします。

1. **:CALCulate:CAPTure:START:EVENT?** を送信して、開始イベントをクリアします。
2. **:CALCulate:CAPTure:START** を送信して、キャプチャを開始します。
3. **:CALCulate:CAPTure:START:EVENT?** をレスポンスデータが 1 (開始) になるまで繰り返し送信します。

:CALCulate:CAPTure:START:EVENT

機能

実際にキャプチャが開始したかを問い合わせます

文法

:CALCulate:CAPTure:START:EVENT?

レスポンスデータ

0|1

- 1: キャプチャ開始すると 1 になります。
 読み取り後, 0 にクリアされます。
- 0: キャプチャ開始前

注:

複数の MD1260A に対してキャプチャ開始状況を問い合わせできません。

使用例

:CALCulate:CAPTure:STARTの使用例を参照してください。

:CALCulate:CAPTure:STATus

機能

キャプチャデータを読み取り可能か問い合わせます。

文法

:CALCulate:CAPTure:STATus?

レスポンスデータ

0|1

0: キャプチャデータ読み取り可能

1: キャプチャ処理中

「キャプチャ処理中」とは、トリガ待ちやキャプチャ停止後のデータ取得中など、キャプチャを開始してからデータ取得処理が完了していない状態です。

このときにキャプチャデータを読み取ると、前にキャプチャしたデータを読み取ってしまうことがあり、期待するキャプチャデータを取得できません。

注:

複数の MD1260A に対してキャプチャデータの存在を問い合わせできません。

使用例

:CALCulate:CAPTure:STOPの使用例を参照してください。

:CALCulate:CAPTure:STOP

機能

トリガ待ちの状態で強制的にキャプチャを停止したいときに送信します。

キャプチャは、トリガ発生後に開始した後は自動で停止します。このため、キャプチャ開始後はこのコマンドを送信する必要はありません。

文法

:CALCulate:CAPTure:STOP

注:

複数の MD1260A のキャプチャを同時に停止することはできません。

使用例

キャプチャの停止を待つ場合は次のようにします。

1. :CALCulate:CAPTure:STOP を送信してキャプチャを停止します。
2. :CALCulate:CAPTure:STATus? をレスポンスデータが 0 (停止) になるまで繰り返し送信します。
3. :CALCulate:CAPTure:DATA:SIZE? をレスポンスデータが、1 以上になるまで繰り返し送信します。

:CALCulate:CAPTure:TRIGger

機能

キャプチャを開始してからトリガが発生したかどうかを問い合わせます。

文法

:CALCulate:CAPTure:TRIGger?

レスポンスデータ

0|1

0: トリガ未発生

1: トリガ発生

キャプチャを開始すると、0 にクリアされます。

注:

複数の MD1260A に対してトリガの発生を問い合わせできません。

:CALCulate:CAPTure:TRIGger:MANual

機能

OTU3, OTU4 アプリケーションのキャプチャトリガが Manual の場合に、マニュアルトリガを発生します。

文法

:CALCulate:CAPTure:TRIGger:MANual

:CALCulate:CAPTure:TRIGger:POSition

機能

OTU3, OTU4 アプリケーションのキャプチャトリガ位置を設定/問い合わせます。

文法

:CALCulate:CAPTure:TRIGger:POSition TOP|MIDDLE

:CALCulate:CAPTure:TRIGger:POSition?

TOP: メモリ先頭がトリガ位置

MIDDLE: メモリ中央がトリガ位置

レスポンスデータ

TOP|MIDDLE

:CALCulate:CAPTure:TRIGger:TYPE

機能

OTU3, OTU4 アプリケーションのキャプチャトリガの種類を設定/問い合わせます。

文法

:CALCulate:CAPTure:TRIGger:TYPE <CHAR>

:CALCulate:CAPTure:TRIGger:TYPE?

トリガ	<CHAR>	キャプチャ対象				
		FRAME	GMP	GMP(ローオーダー)	OH	OH(ローオーダー)
Manual	MANUAL	✓	✓	✓	✓	✓
MFAS = 0	MFAS0	—	—	—	✓	✓
MFAS	MFAS<NR1> *1	✓	—	—	—	—
FAS	FAS	—	—	—	✓	✓
OMFI	OMFI<NR1> *2	✓	—	—	—	—
SM-BIP8	SM_BIP8	✓	—	—	✓	—
PM-BIP8	PM_BIP8	✓	—	—	✓	✓
OOF	OOF	—	—	—	✓	✓
OOM	OOM	✓	—	—	✓	✓
ODU-AIS	AIS_ODU	✓	—	—	✓	✓
ODU-OCI	OCI	✓	—	—	✓	✓
ODU-LCK	LCK	✓	—	—	✓	✓
MSIM	MSIM *3	✓	—	—	✓	—
Lock->Unlock	LOCK_UNLOCK	—	✓	✓	—	—
Unlock->Lock	UNLOCK_LOCK	—	✓	✓	—	—
CRC8 Error	GMP_CRC8	—	✓	✓	—	—
CRC5 Error	GMP_CRC5	—	✓	—	—	—

*1: <NR1>=0~255

*2: OTU3 の場合 <NR1>=0~31, OTU4 の場合 <NR1>=0~79

*3: 次のマッピングで設定可能

ODTU4.8-ODU2e-PRBS, ODTU4.8-ODU2e-10GbE,

ODTU4.1-ODU0-PRBS, ODTU4.1-ODU0-GbE,

レスポンスデータ

<CHAR>

:CALCulate:CAPTure:TYPE

機能

OTU3, OTU4 アプリケーションのキャプチャ対象を設定/問い合わせます。

文法

:CALCulate:CAPTure:TYPE FRAME|GMP|GMP_L|OH|OH_L

:CALCulate:CAPTure:TYPE?

FRAME:	フレーム
GMP:	(ハイオーダーの)Cm(t), デコード
GMP_L:	ローオーダーの Cm(t), デコード
OH:	(ハイオーダーの)オーバーヘッド
OH_L:	ローオーダーのオーバーヘッド

レスポンスデータ

FRAME|GMP|GMP_L|OH|OH_L

使用例

キャプチャ条件を次のとおり設定します。

キャプチャ対象 FRAME

トリガ位置 TOP

トリガの種類 MFAS=16

:CALCulate:CAPTure:TYPE FRAME

:CALCulate:CAPTure:TRIGger:POSition TOP

:CALCulate:CAPTure:TRIGger:TYPE MFAS16

:CALCulate:COUNTER:ETHER:OVERsize**機能**

Counter の Oversize を設定します。

文法

```
:CALCulate:COUNTER:ETHER:OVERsize <NR1>
:CALCulate:COUNTER:ETHER:OVERsize?
```

<NR1>

Oversize 判定のしきい値を byte 単位で設定します。

レスポンスデータ

<NR1>

使用例

```
:CALCulate:COUNTER:ETHER:OVERsize 1522
:CALCulate:COUNTER:ETHER:OVERsize?
> 1522
```

:CALCulate:COUNTER:FLOW:FIELD**機能**

Test Frame カウンタのフィルタ方法が User Defined, または User Defined and Flow ID の場合に、フィールドとビット範囲を設定します。

文法

```
:CALCulate:COUNTER:FLOW:FIELD
<format>,<field>,<offset>,<length>
:CALCulate:COUNTER:FLOW:FIELD?
```

<format>=<STRING>

<field>=<CHAR>

<offset>,<length>=<NR1>

<format>の文字列の説明は、:SOURCE:STREAM:HEADERを参照してください。

<field>には、次の表の文字列を指定します。この表に無いフィールドは設定できません。

表3.3.2-3 フィールドを指定する文字列

<field>	フィールド名
MPLS_TP_DA	MPLS-TP Destination Address
MPLS_TP_SA	MPLS-TP Source Address
MPLS_TP_LABEL{1 2 3 4 5}	MPLS-TP Label
MPLS_TP_EXP{1 2 3 4 5}	MPLS-TP Experimental Use
MPLS_TP_BTM{1 2 3 4 5}	MPLS-TP Bottom of Stack
MPLS_TP_TTL{1 2 3 4 5}	MPLS-TP Time to Live
CW_NIB	PWMCW First nibble
CW_FLAG	PWMCW Flag
CW_FRG	PWMCW FRG
CW_LENGTH	PWMCW Length
CW_SEQ	PWMCW Sequence Number
PBB_DA	PBB Destination Address
PBB_SA	PBB Source Address
PBB_{B I}_TPID	PBB TPID
PBB_{B I}_PCP	PBB PCP
PBB_{B I}_DEI	PBB DEI
PBB_B_VID	PBB VID
PBB_I_SID	PBB SID
PBB_I_RSV	PBB reserved
ETH_DA	Ethernet Destination Address
ETH_SA	Ethernet Source Address
ETH_TYPE	Ethernet Type
VLAN_{1 2}_TPID	Ethernet VLAN TPID
VLAN_{1 2}_PCP	Ethernet VLAN PCP
VLAN_{1 2}_CFI	Ethernet VLAN CFI
VLAN_{1 2}_VID	Ethernet VLAN DEI
MPLS_LABEL{1 2 3}	MPLS Label
MPLS_EXP{1 2 3}	MPLS Experimental Use
MPLS_TTL{1 2 3}	MPLS Time to Live

表3.3.2-3 フィールドを指定する文字列 (続き)

<field>	フィールド名
IPV4_VER	IPv4 Version
IPV4_HLEN	IPv4 Header Length
IPV4_TOS	IPv4 ToS
IPV4_PLEN	IPv4 Packet Length
IPV4_ID	IPv4 Identification
IPV4_FLAG	IPv4 Flags
IPV4_OFFSET	IPv4 Fragment Offset
IPV4_TTL	IPv4 TTL
IPV4_PROT	IPv4 Protocol
IPV4_CHKSUM	IPv4 Header Checksum
IPV4_SA	IPv4 Source Address
IPV4_DA	IPv4 Destination Address
IPV6_VER	IPv6 Version
IPV6_TCLASS	IPv6 Traffic Class
IPV6_FLABEL	IPv6 Flow Label
IPV6_PLEN	IPv6 Packet Length
IPV6_NEXT	IPv6 Next Header
IPV6_HOP	IPv6 Hop Limit
IPV6_SA	IPv6 Source Address
IPV6_DA	IPv6 Destination Address
ARP_SMAC	ARP Source MAC Address
ARP_SIP	ARP Source IP Address
ARP_TMAC	ARP Target MAC Address
ARP_TIP	ARP Target IP Address
ARP_OPE	ARP Operation
ICMPV4_CODE	ICMPv4 Code
ICMPV4_EC_ID	ICMPv4 Identifier
ICMPV4_EC_SEQ	ICMPv4 Sequence No.
ICMPV6_CODE	ICMPv6 Code
ICMPV6_EC_ID	ICMPv6 Identifier
ICMPV6_EC_SEQ	ICMPv6 Sequence No.
ICMPV6_NSNA_TADDR	ICMPv6 Target Address
ICMPV6_NSNA_SADDR	ICMPv6 Source Link Layer Address

レスポンスデータ

<format>,<field>,<offset>,<length>

使用例

```
:CALCulate:COUNter:FLOW:FIELD "NONE",ETH_SA,0,32
:CALCulate:COUNter:FLOW:FIELD?
> "NONE",ETH_SA,0,32
```

:CALCulate:COUNter:FLOW:FIELD:ID**機能**

Test Fframe カウンタのフィルタの値を設定します。
フィルタ方法, テストフレーム数によって, パラメータの数が変わります。

文法

```
:CALCulate:COUNter:FLOW:FIELD:ID
MONITOR,<filter>,<num>,<id>,...,<id>
:CALCulate:COUNter:FLOW:FIELD:ID?
```

<filter>=TEST_FRAME|TF_AND_UD|USER_DEFINED

TEST_FRAME: Test Frame Flow ID

TF_AND_UD: User Defined and Flow ID

USER_DEFINED: User Defined

<num>=<NR1>

<filter>=TEST_FRAME の場合 <id>=<flow_id>

<filter>= USER_DEFINED の場合 <id>=<user_id>

<filter>= TF_AND_UD の場合 <id>=<flow_id>,<user_id>

<flow_id>,<user_id>=<NR1>|<BINARY>|<HEX>

<flow_id>と<user_id>は 16 進数形式, 2 進数形式でも設定できます。

レスポンスデータ

MONITOR,<filter>,<num><id>,...,<id>

<id>は 16 進数形式です。

使用例

```
:CALCulate:COUNter:FLOW:FIELD:ID MONITOR,USER_DEFIND,
8,#H10,#H20,#H30,#H40,#H50,#H60,#H70,#H80
:CALCulate:COUNter:FLOW:FIELD:ID?
> MONITOR,USER_DEFIND,8,#H10,#H20,#H30,#H40,#H50,#H60,
#H70,#H80
```


:CALCulate:COUNTER:FLOW:FIELD:NFID**機能**

Test Frame カウンタのテストフレーム数を問い合わせます。

文法

:CALCulate:COUNTER:FLOW:FIELD:NFID?

<num>=<NR1>

レスポンスデータ

<NR1>

使用例

```
:CALCulate:COUNTER:FLOW:FIELD:NFID?  
> 15
```

:CALCulate:COUNTER:FLOW:TYPE**機能**

Test Frame カウンタのフィルタ方法を設定します。

文法

:CALCulate:COUNTER:FLOW:TYPE <filter>
:CALCulate:COUNTER:FLOW:TYPE?

<filter>=TEST_FRAME|TF_AND_UD|USER_DEFINED

TEST_FRAME: Test Frame Flow ID

TF_AND_UD: User Defined and Flow ID

USER_DEFINED: User Defined

レスポンスデータ

TEST_FRAME|TF_AND_UD|USER_DEFINED

使用例

```
:CALCulate:COUNTER:FLOW:TYPE TEST_FRAME  
:CALCulate:COUNTER:FLOW:TYPE?  
> TEST_FRAME
```

:CALCulate:COUNter:GAP

機能

Gap Size カウンタの測定範囲を設定します。

文法

:CALCulate:COUNter:GAP <min>,<step>

<min>,<step>=<NR1>

:CALCulate:COUNter:SERRor

機能

シーケンスエラーでテストフレームカウンタを停止するかどうか設定します。

文法

:CALCulate:COUNter:SERRor 0|1|ON|OFF

:CALCulate:COUNter:SERRor?

ON|1: シーケンスエラーでテストフレームカウンタを停止する。

OFF|0: シーケンスエラーでテストフレームカウンタを停止しない。

レスポンスデータ

0|1

:CALCulate:COUNter:START[:AUNit]**機能**

カウンタを開始します。

すでにカウンタが動作している場合は、再スタートとなります。

:AUNit を追加すると、接続しているすべてのユニットのカウンタを同時に開始します。

文法

:CALCulate:COUNter:START[:AUNit]

使用例

カウンタの開始を待つ場合は以下のようにします。

使用例のように EVENT コマンドを使用したほうが、カウンタを停止して開始するよりも実行時間を短くすることができます。

1. :CALCulate:COUNter:START:EVENT? を送信して開始イベントをクリアします。
2. :CALCulate:COUNter:START を送信してカウンタを開始します。
3. :CALCulate:COUNter:START:EVENT? をレスポンスデータが 1（開始）になるまで繰り返し送信します。

:CALCulate:COUNter:START:EVENT[:AUNit]**機能**

カウンタの開始イベントを問い合わせます。

文法

:CALCulate:COUNter:START:EVENT[:AUNit]?

レスポンスデータ

0|1

0: カウンタが開始前

1: カウンタが開始 本コマンドで読み取った後は 0 にクリアされます。
ヘッダーに:AUNit を付けた場合は、接続しているすべてのユニットのカウンタが開始すると 1 となります。

使用例

:CALCulate:COUNter:START[:AUNit]の使用例を参照してください。

:CALCulate:COUNter:STATus[:AUNit]**機能**

カウンタの動作状態を確認します。

文法

:CALCulate:COUNTER:STATUS[:AUNit]?

レスポンスデータ

0|1

0: 動作停止中

1: 動作中

ヘッダーに:AUNitを付けた場合は接続したユニットのどれか1つ以上のカウンタが開始すると1, すべてのユニットのカウンタが停止すると0になります。

使用例

:CALCulate:COUNTER:STOP[:AUNit]の使用例を参照してください。

:CALCulate:COUNTER:STOP[:AUNit]**機能**

カウンタを停止します。

:AUNit を追加すると、接続しているすべてのユニットのカウンタを停止します。

文法

```
:CALCulate:COUNTER:STOP[:AUNit]
```

ポート個別コマンドとの関係

本 Event の値は、各ポートの Event の論理積です。本コマンドの読み値が 1 だった直後に、各ポートの Event 値が 0 にクリアされます。

使用例

カウンタの停止を待つ場合は次のようにします。

1. :CALCulate:COUNTER:STOP を送信してカウンタを停止します。
2. :CALCulate:COUNTER:STATus? を送信してレスポンスデータが 0 (停止) になるまで、問い合わせを繰り返します。

:CALCulate:DATA**機能**

測定結果を問い合わせます。

文法

```
:CALCulate:DATA? <item>[,<item>,...]
```

<item>=<CHAR>

取得する測定項目の ID を設定します。

設定できる ID は「付録 A 測定項目一覧」を参照してください。

ID はコンマ区切りで複数設定することができます。

ただし、レスポンスデータの文字数がバッファサイズ (65536 文字) を超える場合はエラー (-310,"System error") になります。

レスポンスデータ

```
<value>[,<value>,...]|-----
```

設定した ID の測定データです。

データの書式は、設定した項目によって異なります。

測定データが無効の場合は、「-----」になります。

Lane 単位の測定項目は、Lane 数分の測定データを「,」で区切って問い合わせます。

カウンタ値を取得する場合は、:CALCulate:DATA:TYPE で取得するカウンタデータの種類 (Current または Accumulated) を選択します。

使用例

受信クロック周波数 (Hz と ppm) を問い合わせます。

```
:CALCulate:DATA? RX_FREQ,RX_FREQ_D  
> 103125000000,0.0
```

:CALCulate:DATA:TYPE

機能

カウンタデータの種類を設定します。
カウンタデータの種類の設定を問い合わせます。

文法

```
:CALCulate:DATA:TYPE ACCUM|CURRENT  
:CALCulate:DATA:TYPE?
```

CURRENT : Current
ACCUM : Accumulated

レスポンスデータ

ACCUM|CURRENT

使用例

Counter の経過時間とビットエラーカウント (Accumulated 値) と受信クロック周波数を問い合わせます。

```
:CALCulate:DATA:TYPE ACCUM  
:CALCulate:DATA? ELAPSED,BER_CNT,RX_FREQ  
> 20,1,103135312400
```

:CALCulate:DElay:STARt[:AUNit]

機能

OTU3, OTU4 アプリケーションの遅延時間測定を開始します。
[Delay Settings] の Mode は [Single] になります。
:AUNit を追加すると、接続しているすべてのユニットの遅延時間測定を同時に開始します。

文法

```
:CALCulate:DElay:STARt
```

使用例

遅延時間測定の開始を待つ場合は以下のようにします。

1. :CALCulate:DElay STARt を送信して、キャプチャを開始します。
2. :CALCulate:DElay:STARt:EVENT? をレスポンスデータが 1 (開始) になるまで繰り返し送信します。
3. :CALCulate:DElay:STARt:STATus? をレスポンスデータが 0 (測定結果を読み取り可能) になるまで繰り返し送信します。
4. :CALCulate:DATA? PM_DELAYなどで測定結果を読み取ります。

:CALCulate:DElay:STARt:EVENT[:AUNit]**機能**

OTU3, OTU4 アプリケーションの遅延時間測定が開始したかを問い合わせます

文法

:CALCulate:DElay:STARt:EVENT?

レスポンスデータ

0|1

- 1: 遅延時間測定を開始すると 1 になります。
読み取り後, 0 にクリアされます。
- 0: 遅延時間測定を開始していません。

使用例

:CALCulate:DElay:STARt[:AUNit]の使用例を参照してください。

:CALCulate:DElay:STATus[:AUNit]**機能**

OTU3, OTU4 アプリケーションの遅延時間測定結果を読み取り可能か問い合わせます。

文法

:CALCulate:DElay:STATus?

レスポンスデータ

0|1

- 0: 遅延時間測定結果を読み取り可能
- 1: 遅延時間測定中
「遅延時間測定中」とは, 測定時間が経過していないため, 測定結果が用意されていない状態です。
このときに遅延時間測定結果を読み取ると, 前回測定した遅延時間を読み取ってしまうことがあり, 期待する測定結果を取得できません。

使用例

:CALCulate:DElay:STOP[:AUNit]の使用例を参照してください。

:CALCulate:DElay:STOP[:AUNit]

機能

OTU3, OTU4 アプリケーションの遅延時間測定を停止したいときに送信します。遅延時間測定は、測定時間が経過すると自動で停止します。このため、測定時間経過後にこのコマンドを送信する必要はありません。

:AUNit を追加すると、接続しているすべてのユニットの遅延時間測定を停止します。

文法

:CALCulate:DElay:STOP

:CALCulate:EALarm[:AUNit]**機能**

サマリステータス・時刻表示エリアの Error/Alarm LED の状態を問い合わせます。

文法

:CALCulate:EALarm[:AUNit]?

レスポンスデータ

0|1|2

AUNit を省略した場合は, :UENTry:ID で設定したユニットのみの情報を表します。

- 0: 正常(緑)
- 1: 異常(赤)
- 2: ヒストリ(オレンジ)

ヘッダーに AUNit を付けた場合は, 次の状態を表します。

- 0: 接続しているすべてのユニットが正常(緑)
- 1: 接続しているユニットの 1 つ以上に異常(赤)が発生
- 2: 接続しているユニットの 1 つ以上にヒストリ(オレンジ)が発生
接続しているすべてのユニットで異常(赤)が発生していない。

使用例

エラー・アラームが発生していないとき

:CALCulate:EALarm?

> 0

:CALCulate:MONitor:OTU:COLumn**機能**

OTU データモニタが Frame 表示の場合, 読み取り先頭カラムを設定/問い合わせます。

文法

:CALCulate:MONitor:OTU:COLumn <NR1>

:CALCulate:MONitor:OTU:COLumn?

<NR1>: カラム番号 (1 ~ 4065)

レスポンスデータ

<NR1>

使用例

:CALCulate:MONitor:OTU:DATAの使用例を参照してください。

:CALCulate:MONitor:OTU:DATA**機能**

:CALCulate:MONitor:OTU:TYPE で指定した種類のモニタ値を問い合わせます。

文法

:CALCulate:MONitor:OTU:DATA? [<offset>,<length>]

<offset>=<NR1>: カラム番号

<length>=<NR1>: データ長

<offset>,<length>を省略すると、レスポンスデータには最大のバイト数のデータになります。

:CALCulate:MONitor:OTU:TYPEの設定によって、データ長に指定できる最大値が異なります。

FRAME, FRAME_L: 64, FTFL, FTFL_L: 255, MSI: 80, TTI: 64

レスポンスデータ

レスポンスデータは、:CALCulate:MONitor:OTU:TYPEの設定によって異なります。

モニタデータの 種類	レスポンス データ	説明
FRAME, FRAME_L	<STRING>	", "区切りで 1 バイトずつ最大 64 バイト分の 16 進数 取得する OTU フレームモニタデータの先頭カラムは、 :CALCulate:MONitor:OTU:COLu mnコマンドで設定します。
FTFL, FTFL_L	<STRING>	", "区切りで 1 バイトずつ最大 256 バイト分の 16 進数
MSI	<STRING>	", "区切りで 1 バイトずつ最大 80 バイト分の 16 進数
TTI_PM, TTI_PM_L, TTI_SM, TTI_TCM {1 2 3 4 5 6}	<STRING>	", "区切りで 1 バイトずつ最大 64 バイト分の 16 進数

注:

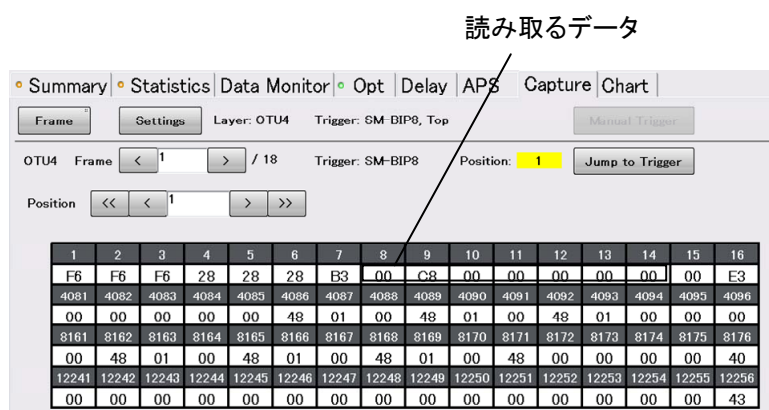
以下の場合、データを取得する前に、1 秒以上待ってください。

:CALCulate:MONitor:OTU:COLumn で、取得するフレームのカラムを変更した。

使用例

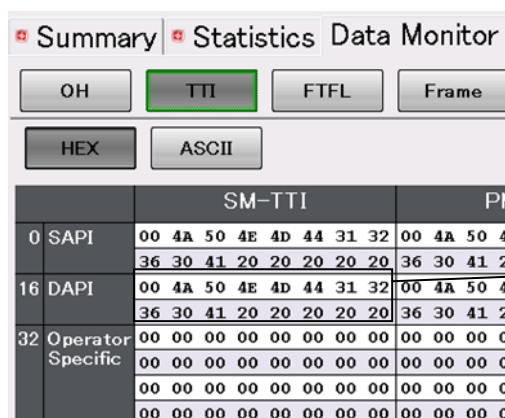
OTU フレームの 1 列目, 8~14 カラムのデータを取得します。

```
:CALCulate:MONitor:OTU:TYPE FRAME
:CALCulate:MONitor:OTU:COLumn 1
:CALCulate:MONitor:OTU:DATA? 7,7
>"00,C8,00,00,00,00,00"
```



SM-TTI のデータを 16 バイト取得します。

```
:CALCulate:MONitor:OTU:TYPE TTI_SM
:CALCulate:MONitor:OTU:DATA? 16,16
>"00,4A,50,4E,4D,44,31,32,36,30,41,20,20,20,20,20"
```



:CALCulate:MONitor:OTU:TYPE**機能**

OTU フレームのモニタデータの種類を設定/問い合わせます。

文法

:CALCulate:MONitor:OTU:TYPE <CHAR>

:CALCulate:MONitor:OTU:TYPE?

<CHAR>

<CHAR>	モニタデータの種類
FRAME	0OTU フレーム *1
FRAME_L	ローオーダーの OTU フレーム
FTFL	FTFL *1
FTFL_L	ローオーダーの FTFL
MSI	MSI
TTI_PM	PM-TTI *1
TTI_PM_L	ローオーダーの PM-TTI
TTI_SM	SM-TTI
TTI_TCM{1 2 3 4 5 6}	TCM{1 2 3 4 5 6}-TTI

*1: マッピングが ODTU4.1, ODTU4.8 の場合は、ハイオーダーのデータを指定します。

*2: Stuff モニタのデータはリモートコマンドで取得できません。

レスポンスデータ**<CHAR>****使用例**

:CALCulate:MONitor:OTU:TYPE FTFL

:CALCulate:MONitor:OTU:TYPE?

> FTFL

:CALCulate:MONitor:OTU:TYPE TTI_TCM3

:CALCulate:MONitor:OTU:TYPE?

> TTI_TCM3

:CALCulate:TRIGger:CONDition**機能**

100GbE, 40GbE アプリケーションのキャプチャのトリガ条件を設定します。
キャプチャのトリガ条件を問い合わせます。

文法

```
:CALCulate:TRIGger:CONDition <BINARY>  
:CALCulate:TRIGger:CONDition?
```

<BINARY>には、トリガ条件を有効にするビットを 1 として値を設定します。
先頭ビットが bit0 になります。

bit	トリガ条件
0	Good Frame
1	LF(Local Fault)
2	RF(Remote Fault)
3	Error Signal
4	FCS Error
5	Undersize
6	Fragment
7	Oversize
8	Oversize & FCS
9-31	予約

レスポンスデータ

<<BINARY>>

ビット長は 32 ビットです。

使用例

トリガ条件に Any Frame を設定します。

```
:CALCulate:TRIGger:CONDition #B100011111  
:CALCulate:TRIGger:CONDition?  
> #B10001111100000000000000000000000
```

トリガ条件に Any Frame Error を設定します。

```
:CALCulate:TRIGger:CONDition #B000011111
```

トリガ条件に Any MII Error を設定します。

```
:CALCulate:TRIGger:CONDition #B0111
```

トリガ条件に Any Error を設定します。

```
:CALCulate:TRIGger:CONDition #B011111111
```

:MDIO:READ

機能

CFP の MDIO レジスタの値を問い合わせます。

文法

:MDIO:READ? <address>

<address>=<HEX>

レジスタのアドレス(#H0000～#HFFFF)

レスポンスデータ

<value>=<HEX>

読み取ったレジスタの値(#H0000～#HFFFF)

使用例

MDIO レジスタ 0xA016 の内容を読み取ります。

:MDIO:READ? #HA016

> #H0020

:MDIO:WRITe

機能

CFP の MDIO レジスタの値を設定します。

文法

:MDIO:WRITe <address>,<value>

<address>=<HEX>

レジスタのアドレス(#H0000～#HFFFF)

<value>=<HEX>

レジスタの値(#H0000～#HFFFF)

:MMEMory:CATalog

機能

以下の場所に保存されている設定ファイルの情報を問い合わせます。

C:\Documents and Settings\Administrator

\My Documents¥Anritsu\MD1260A\UserData\Setting

文法

:MMEMory:CATalog?

レスポンスデータ

<NR1> [,<file_entry>,...]

<NR1>: 保存されている設定ファイルの数
ファイル拡張子が MD1260A の設定ファイルの数です。

<file_entry>

設定ファイルごとに以下の情報をコンマ区切りで出力します。

<file_name>: <STRING>形式のファイル名
ファイル名に拡張子は含みません。

<file type>: 設定ファイルの種類

E40G:	40GbE
E40G_N:	40GbE No Frame
E100G:	100GbE
E100G_N:	100GbE No Frame
OTU3:	OTU3
OTU3_N:	OTU3 No Frame
OTU4:	OTU4
OTU4_E:	ODU4-100GbE
OTU4_ODU2E:	ODTU4.8-ODU2e-PRBS
OTU4_ODU2E_E:	ODTU4.8-ODU2e-10GbE
OTU4_ODU0:	ODTU4.1-ODU0-PRBS
OTU4_ODU0_E:	ODTU4.1-ODU0-GbE
OTU4_N:	OTU4 No Frame

例: フォルダに以下のファイルがある場合のレスポンスデータ

1. "Setup100GbE"という名前の 100GbE 設定ファイル
 2. "SetupOTU4"という名前の OTU4 設定ファイル
- > 2, "Setup100GbE", E100G, "SetupOTU4", OTU4

例: フォルダ内にファイルが 1 つも無い場合のレスポンスデータ

> 0

使用例

1. 現在起動しているアプリケーションを調べます。

```
:SOURce:MAPPing?
```

```
> E100G
```

2. 保存されているファイルを調べます。

```
:MMEMory:CATalog?
```

```
> 2, "Setup100GbE", E100G, "SetupOTU4", OTU4
```

3. 現在起動しているアプリケーションの設定ファイルを読み取ります。

```
:MMEMory:RECall "Setup100GbE"
```

4. 現在の測定結果を保存します。

```
:MMEMory:STORe "Result100GbE", RESULT
```

:MMEMory:INITialize

機能

:UENTry:IDで指定したユニット ID の MD1260A の設定を工場出荷時の状態に戻します。

ただし、以下の内容は工場出荷時の状態に戻りません。

- GPIB アドレス, IP アドレスなどの Windows の設定
- トランシーバの設定
 - (:SOURce:TRANsceiver:EMPHasis:FIRSt
 - :SOURce:TRANsceiver:EMPHasis:PRE
 - :SOURce:TRANsceiver:EMPHasis:SECond
 - :SOURce:TRANsceiver:VOD
 - :SENSe:TRANsceiver:EQUalizer:CONTrol
 - :SENSe:TRANsceiver:EQUalizer:DCGain)
- レスポンスターミネータの設定 (:SYSTem:TERMination)

文法

:MMEMory:INITialize

使用例

ユニット ID 2 のユニットを初期化します。

```
:UENTry:ID 2
:MMEMory:INITialize
```

:MMEMory:LOG:FNAME

機能

最新の Log ファイル名を問い合わせます。

複数ユニット接続時は、:UENTry:ID で設定したユニットの Log ファイル名を問い合わせます。

文法

:MMEMory:LOG:FNAME?

レスポンスデータ

<STRING>: 次のファイル名, または""です。

- Log 動作中の場合は, 現在出力している Log ファイル名
- Log 動作していない場合は, 直前に出力した Log ファイル名
- アプリケーションを起動してから一度も Log ファイルを出力していない場合は, 空文字列 ""

使用例

```
:MMEMory:LOG:FNAME?
> "Test20100301T123456_U010000_0.csv"
```


:MMEMory:LOG:ITEM

機能

Log に出力する項目を設定します。

複数ユニット接続時は、:UENTry:ID で設定したユニットに対して Log に出力する項目を設定します。

文法

```
:MMEMory:LOG:ITEM <id1>[,<id2>,...]
```

```
:MMEMory:LOG:ITEM?
```

<id>

Log に出力する項目の ID を設定します。

設定できる ID については「付録 A 測定項目一覧」を参照してください。

ID はコンマ区切りで複数設定することができます。

レスポンスデータ

```
<id1>[,<id2>,...]|OFF
```

出力する項目が 1 つも無い場合は、OFF になります。

使用例

```
:MMEMory:LOG:ITEM?
```

```
> OFF
```

:MMEMory:LOG:PREFIX

機能

Log ファイル名の先頭に付ける文字列を設定します。

複数ユニット接続時は、すべてのユニットに対して同じ文字列を設定します。

文法

```
:MMEMory:LOG:PREFIX <STRING>
```

```
:MMEMory:LOG:PREFIX?
```

<STRING>

Log ファイル名の先頭に付ける文字列を Test に設定します。

例: "Test"

レスポンスデータ

```
<STRING>
```

使用例

```
:MMEMory:LOG:PREFIX "Test"
```

```
:MMEMory:LOG:PREFIX?
```

```
"Test"
```

:MMEMory:LOG:START

機能

Log 出力を開始します。すでに Log 出力が動作している場合は、新しいログファイルを作成して Log 出力を開始します。

複数ユニット接続時は、すべてのユニットが Log 出力を開始します。

文法

:MMEMory:LOG:START

使用例

:MMEMory:LOG:STATUsの使用例を参照してください。

:MMEMory:LOG:STATUs

機能

Log 出力の動作状態を問い合わせます。

文法

:MMEMory:LOG:STATUs?

レスポンスデータ

0|1

0: Log 出力動作停止中

1: Log 出力動作中

複数ユニット接続時は、次を表します。

0: すべてのユニットが Log 出力動作停止中

1: 1 つ以上のユニット Log 出力動作中

使用例

:MMEMory:LOG:START

:MMEMory:LOG:STATUs?

> 1

:MMEMory:LOG:STOP

機能

Log 出力を停止します。

複数ユニット接続時は、すべてのユニットが Log 出力を停止します。

文法

:MMEMory:LOG:STOP

使用例

:MMEMory:LOG:STOP

:MMEMory:LOG:STATUs?

> 0

:MMEMory:LOG:TIMing**機能**

Log 出力のタイミングを設定します。

複数ユニット接続時は、すべてのユニットに対して同じ Log 出力のタイミングを設定します。

Log 出力のタイミング設定を問い合わせます。

文法

:MMEMory:LOG:TIMing <CHAR>

:MMEMory:LOG:TIMing?

<CHAR>

以下から選択します。

E1S : Every 1s

E10S : Every 10s

E1MIN : Every 1min

ERROR : Whenever an error occurs

レスポンスデータ

<CHAR>

使用例

```
:MMEMory:LOG:TIMing E1S
```

```
:MMEMory:LOG:TIMing?
```

```
> E1S
```

:MMEMory:RECall[:AUNit]**機能**

アプリケーションの設定をファイルから読み取ります。

:AUNit を追加すると、接続しているすべてのユニットがアプリケーションの設定をファイルから読み取ります。

文法

:MMEMory:RECall[:AUNit] <filename>

<filename>=<STRING>

読み込むファイル名を設定します。ファイル名は、ディレクトリ名・拡張子を含みません。

このコマンドは、現在起動しているアプリケーションと一致した設定ファイルを読み込みます。

ファイル内に保存されたユニット構成情報と、現在の構成に不一致を検出した場合は-310 (System error) が返ります。

使用例

:UENTry:ID で設定したユニットに"Setup100GbE"というファイル名の設定ファイルを読み込む場合

```
:MMEMory:RECall "Setup100GbE"
```

全ユニットに"SetupAll"というファイル名の設定ファイルを読み込む場合

```
:MMEMory:RECall:AUNit "SetupAll"
```

:MMEMory:STORe[:AUNit]

機能

アプリケーションの設定を保存します。

:AUNit を追加すると、接続しているすべてのユニットがアプリケーションの設定を保存します。

測定結果のレポートを出力します。

:AUNit を追加すると、接続しているすべてのユニットが測定結果のレポートを出力します。

文法

```
:MMEMory:STORe[:AUNit] <filename>,RESULT|SETUP
```

<filename>

保存ファイル名を<STRING>形式で指定します。

ファイル名は、ディレクトリ名・拡張子を含みません。

保存内容を以下から選択します。

SETUP: 測定条件

RESULT: 測定結果

使用例

:MMEMory:CATalog の使用例を参照してください。

例: :UENTry:ID で指定したユニットの測定条件を"Setup100GbE"というファイル名で保存する場合

```
:MMEMory:STORe "Setup100GbE",SETUP
```

例: :UENTry:ID で指定したユニットの測定結果を"Result100GbE"というファイル名で保存する場合

```
:MMEMory:STORe "Result100GbE",RESULT
```

例: 全ユニットの測定条件を"SetupAll"というファイル名で保存する場合

```
:MMEMory:STORe:AUNit "SetupAll",SETUP
```

例: 全ユニットの測定結果を"ResultAll"というファイル名で保存する場合

```
:MMEMory:STORe:AUNit "ResultAll",RESULT
```

:MMEMory:STORe:CAPTure

機能

現在表示されているキャプチャデータをファイルに保存します。

文法

:MMEMory:STORe:CAPTure <filename>

<filename>

保存するファイル名 (ディレクトリ名・拡張子は含めない) を<STRING>形式で指定します。

40GbE, 100GbE アプリケーションの場合, ファイルに出力する内容を:MMEMory:STORe:CAPTure:ITEMで指定します。

キャプチャデータが表示されていないときは「-310, System error」となります。

接続しているすべてのユニットに対して, キャプチャデータのファイル保存はできません。

:MMEMory:STORe:CAPture:ITEM

機能

キャプチャデータのファイル出力項目を設定します。

文法

:MMEMory:STORe:CAPture:ITEM <id>

<id>

以下のどれかを設定します。

LIBPCAP - nsec libpcap 形式(*.pcap)のファイルを作成する。

TEXT_TABLE - 一覧エリアの内容をテキストファイルに出力する。

TEXT_BYTES - 詳細エリアの全データの内容をテキストファイルに出力する。

接続しているすべてのユニットに対して、キャプチャデータのファイル出力項目設定はできません。

:MMEMory:STReam:RECall

機能

アプリケーションの設定ファイルからストリームの情報だけを読み取ります。

本コマンドは 40GbE, 100GbE アプリケーションで使用できます。

文法

:MMEMory:STReam:RECall <filename>[,<file_type>]

<filename>=<STRING>

読み込むファイル名を設定します。ファイル名は、ディレクトリ名・拡張子を含みません。

<file_type>=E100G|E40G

E100G: 100GbE の測定条件ファイル

E40G: 40GbE の測定条件ファイル

<file_type>を省略すると、現在起動しているアプリケーションの測定条件ファイルが設定されます。

指定したファイルが Frame BERT が[On]の状態では保存されている場合は、-310 (System error) が返ります。

使用例

:UENTry:ID で設定したユニットに"MPLS_IPv4_100GbE"というファイル名の設定ファイルを読み込む場合

:MMEMory:STReam:RECall "MPLS_IPv4_100GbE",E100G

:PROTOCOL:ARPNa:REPLY:ENABLE**機能**

ストリーム 1～16 に対して、ARP/NA 応答の送信の有無を設定します。
ストリーム 1～16 の ARP/NA 応答の送信の有無を問い合わせます。

文法

```
:PROTOCOL:ARPNa:REPLY:ENABLE <NR1>,<NR1>,...,<NR1>
:PROTOCOL:ARPNa:REPLY:ENABLE?
```

<NR1>=0|1

0: ARP/NA 応答を送信しない。
1: ARP/NA 応答を送信する。

レスポンスデータ

<NR1>,<NR1>,...,<NR1>

使用例

ストリーム 1～8 には ARP/NA 応答を送信し、ストリーム 9～16 には ARP/NA 応答を送信しない設定にします。

```
:PROTOCOL:ARPNa:REPLY:ENABLE
1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0
:PROTOCOL:ARPNa:REPLY:ENABLE?
> 1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0
```

:PROTOCOL:GARPns:DURATION:TYPE**機能**

GARP, または NS の送信方法を設定します。

文法

```
:PROTOCOL:GARPns:DURATION:TYPE <char>
:PROTOCOL:GARPns:DURATION:TYPE?
```

<char>=REPEAT|SINGLE

REPEAT: Repeat
SINGLE: Single

レスポンスデータ

REPEAT|SINGLE

使用例

```
:PROTOCOL:GARPns:DURATION:TYPE SINGLE
:PROTOCOL:GARPns:DURATION:TYPE?
> SINGLE
```

:PROTocol:GARPns:ENABle**機能**

ストリーム 1～16 に対して, GARP, または NS 送信の有無を設定します。
ストリーム 1～16 の GARP, または NS 送信の有無を問い合わせます。

文法

```
:PROTocol:GARPns:ENABle <NR1>,<NR1>,...,<NR1>  
:PROTocol:GARPns:ENABle?
```

<NR1>=0|1

0: GARP/NS を送信しない。
1: GARP/NS を送信する。

レスポンスデータ

<NR1>,<NR1>,...,<NR1>

使用例

ストリーム 1～8 には GARP または NS を送信し, ストリーム 9～16 には GARP または NS を送信しない設定にします。

```
:PROTocol:GARPns:ENABle 1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0  
:PROTocol:GARPns:ENABle?  
> 1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0
```

:PROTocol:GARPns:INTerval**機能**

ARP, または NS パケットの送信時間間隔を設定, または問い合わせます。

文法

```
:PROTocol:GARPns:INTerval <NR1>  
:PROTocol:GARPns:INTerval?
```

<NR1>=1～100

レスポンスデータ

<NR1>

使用例

ARP, または NS パケットの送信時間間隔を 10 に設定します。

```
:PROTocol:GARPns:INTerval 10  
:PROTocol:GARPns:INTerval?  
> 10
```


:PROTocol:GARPns:START**機能**

GARP, または NS の送信を開始します。

文法

:PROTocol:GARPns:START

:PROTocol:GARPns:STATus**機能**

GARP, または NS の送信状態を問い合わせます。

文法

:PROTocol:GARPns:STATus?

レスポンスデータ

0|1

0: GARP/NS 送信終了

1: GARP/NS 送信中

使用例

:PROTocol:GARPns:START

:PROTocol:GARPns:STATus?

> 1

:PROTocol:GARPns:STOP**機能**

GARP, または NS の送信を停止します。

文法

:PROTocol:GARPns:STOP

使用例

:PROTocol:GARPns:STOP

:PROTocol:GARPns:STATus?

> 0

:PROTOCOL:GARPns:TYPE

機能

GARP の種類を設定, または問い合わせます。

文法

:PROTOCOL:GARPns:TYPE <CHAR>

:PROTOCOL:GARPns:TYPE?

<CHAR>=REPLY|REQUEST

REPLY: 応答

REQUEST: 要求

レスポンスデータ

REPLY|REQUEST

使用例

:PROTOCOL:GARPns:TYPE REQUEST

:PROTOCOL:GARPns:TYPE?

> REQUEST

:PROTOCOL:PING:COUNT

機能

Ping 試験の実行回数を設定, または問い合わせます。

文法

:PROTOCOL:PING:COUNT <NR1>

:PROTOCOL:PING:COUNT?

<NR1>=1~32637

レスポンスデータ

<NR1>

使用例

:PROTOCOL:PING:COUNT 20

:PROTOCOL:PING:COUNT?

>20

:PROTocol:PING:DST:IPV4

機能

Ping 試験のターゲット IP アドレス (IPv4) を設定, または問い合わせます。

文法

:PROTocol:PING:DST:IPV4 <HEX>

:PROTocol:PING:DST:IPV4?

<HEX>:Ping 試験のターゲット IP アドレス (4 バイトの 16 進数)

使用例

Ping 試験のターゲット IPv4 アドレスを 192.168.1.2 (0xC0A80102) に設定します。

:PROTocol:PING:DST:IPV4 #HC0A80102

:PROTocol:PING:DST:IPV4?

>#HC0A80102

:PROTocol:PING:DST:IPV6

機能

Ping 試験のターゲット IP アドレス (IPv6) を設定, または問い合わせます。

文法

:PROTocol:PING:DST:IPV6 <HEX>

:PROTocol:PING:DST:IPV6?

<HEX>:Ping 試験のターゲット IP アドレス (16 バイトの 16 進数)

使用例

Ping 試験のターゲット IPv6 アドレスを 2001::10 に設定します。

:PROTocol:PING:DST:IPV6

#H2001000000000000000000000000000010

:PROTocol:PING:DST:IPV6?

>#H2001000000000000000000000000000010

:PROTOCOL:PING:DST:MAC

機能

Ping 試験のターゲット MAC アドレス を設定, または問い合わせます。

文法

:PROTOCOL:PING:DST:MAC <HEX>

:PROTOCOL:PING:DST:MAC?

<HEX>: Ping 試験のターゲット MAC アドレス (6 バイトの 16 進数)

使用例

Ping 試験のターゲット MAC アドレスを 0x0A0000123456 に設定します。

```
:PROTOCOL:PING:DST:MAC #H0A0000123456
```

```
:PROTOCOL:PING:DST:MAC?
```

```
>#H0A0000123456
```

:PROTOCOL:PING:IPMODE

機能

Ping 試験の IP バージョンを設定, または問い合わせます。

文法

:PROTOCOL:PING:IPMODE IPV4|IPV6

:PROTOCOL:PING:IPMODE?

IPV4: IPv4

IPV6: IPv6

レスポンスデータ

IPV4|IPV6

使用例

```
:PROTOCOL:PING:IPMODE IPV6
```

```
:PROTOCOL:PING:IPMODE?
```

```
> IPV6
```

:PROTocol:PING:MACResolve**機能**

Ping 試験の MAC アドレス解決のオン/オフを設定, または問い合わせます。
MAC アドレス解決をオンに設定すると, 宛先 MAC アドレスがターゲット IP アドレスから自動解決されます。

文法

```
:PROTocol:PING:MACResolve 0|1|ON|OFF  
:PROTocol:PING:MACResolve?
```

ON|1: MAC Resolve をオンにする。
OFF|0: MAC Resolve をオフにする。

レスポンスデータ

0|1

使用例

```
:PROTocol:PING:MACResolve 1  
:PROTocol:PING:MACResolve?  
> 1
```

:PROTocol:PING:PAYLoad**機能**

Ping 試験のパケットペイロードパターンを設定, または問い合わせます。

文法

```
:PROTocol:PING:PAYLoad ALL0|ALL1|ALTERNATE  
:PROTocol:PING:PAYLoad?
```

ALL0: All 0
ALL1: All 1
ALTERNATE: 0/1 bit

レスポンスデータ

ALL0|ALL1|ALTERNATE

使用例

```
:PROTocol:PING:PAYLoad ALL0  
:PROTocol:PING:PAYLoad?  
> ALL0
```

:PROTOCOL:PING:PSIZE:TYPE

機能

Ping 試験のパケットサイズの設定, または問い合わせます。

文法

:PROTOCOL:PING:PSIZE:TYPE FIX|INCREMENT

:PROTOCOL:PING:PSIZE:TYPE?

FIX:

Fixed

INCREMENT:

Increment

レスポンスデータ

FIX|INCREMENT

使用例

:PROTOCOL:PING:PSIZE:TYPE FIX

:PROTOCOL:PING:PSIZE:TYPE?

> FIX

:PROTOCOL:PING:PSIZE:VALUE**機能**

Ping 試験のパケットサイズを設定, または問い合わせます。

文法

```
:PROTOCOL:PING:PSIZE:VALUE <start>[,<step>]
:PROTOCOL:PING:PSIZE:VALUE?
```

<start>,<step>=<NR1>

<start>: パケットサイズの下限值

<step>: パケットサイズの変化量

パケットサイズインクリメントがオフの場合, <step>の値は無視されます。

レスポンスデータ

<NR1>,<NR1>

使用例

```
:PROTOCOL:PING:PSIZE:VALUE 64,8
:PROTOCOL:PING:PSIZE:VALUE?
> 64,8
```

:PROTOCOL:PING:REPLY:ENABLE**機能**

ストリーム 1~16 に対して, Ping 応答の送信の有無を設定します。
ストリーム 1~16 の Ping 応答の送信の有無を問い合わせます。

文法

```
:PROTOCOL:PING:REPLY:ENABLE <NR1>,<NR1>,...,<NR1>
:PROTOCOL:PING:REPLY:ENABLE?
```

<NR1>=0|1

0: Ping 応答を送信しない。

1: Ping 応答を送信する。

レスポンスデータ

<NR1>,<NR1>,...,<NR1>

使用例

ストリーム 1~4 には Ping 応答を送信し, ストリーム 5~16 は Ping 応答を送信しない設定にします。

```
:PROTOCOL:PING:REPLY:ENABLE
1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0
:PROTOCOL:PING:REPLY:ENABLE?
> 1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0
```

:PROTocol:PING:RESult**機能**

Ping 試験の結果を問い合わせます。

文法

:PROTocol:PING:RESult? <CHAR>

<CHAR>=HISTORY1|HISTORY2|...|HISTORY32|TOTAL

HISTORY1 History の表の 1 番目の結果

HISTORY2 History の表の 2 番目の結果

:

HISTORY32 History の表の 32 番目の結果

TOTAL: Total の表の結果

レスポンスデータ

HISTORY1～HISTORY32 を指定した場合

<from>=<HEX>: IP アドレス

<size>=<NR1>: パケットサイズ

<ttl>=<NR1>: TTL

<time>=<NR1>: ラウンドトリップタイム[ms]

測定結果が存在しない History の番号を指定した場合は、パラメータエラーになります。

TOTAL を指定した場合

<tx_arpreq>=<NR1>: Tx ARP Request

<rx_arpreq>=<NR1>: Rx ARP Reply 受信回数

<tx_pingreq>=<NR1>: Tx Ping Reqeust 送信回数

<rx_pingrep>=<NR1>: Rx Ping Reply

<time_min>=<NR1>: Time Min.

<time_max>=<NR1>: Time Max.

<time_ave>=<NR1>: Time Ave.

<timeout>=<NR1>: Timeout

使用例

:PROTocol:PING:RESult? HISTORY2

> #HC0A80101,128,128,10

:PROTocol:PING:SRC:IPV4**機能**

Ping 試験の送信元 IP アドレス (IPv4) を設定、または問い合わせます。

文法

:PROTocol:PING:SRC:IPV4 <HEX>

:PROTocol:PING:SRC:IPV4?

:PROTocol:PING:SRC:IPV6

```
>#HC0A80104
```

```
:PROTOcol:PING:SRC:IPV6?
```

```
>#H20010000000000000000000000000000000012
```

:PROTocol:PING:SRC:MAC

```
:PROTOcol:PING:SRC:MAC?
```

```
>#H0A0000123478
```

:PROTOCOL:PING:START

機能

Ping 試験を開始します。

文法

:PROTOCOL:PING:START

:PROTOCOL:PING:START:EVENT

機能

Ping 試験の開始イベントを問い合わせます。
開始イベントを読み取ると、値が 0 になります。

文法

:PROTOCOL:PING:START:EVENT?

レスポンスデータ

0|1

0: Ping 試験を未実行

Ping 試験開始後にイベントを読み取り済み

1: Ping 試験を開始した。

使用例

:PROTOCOL:PING:START

:PROTOCOL:PING:START:EVENT?

> 1

:PROTOCOL:PING:START:EVENT?

> 0

:PROTOCOL:PING:STATUS

機能

Ping 試験の状態を問い合わせます。

文法

:PROTOCOL:PING:STATUS?

レスポンスデータ

0|1|2

0: Ping 試験終了

1: Ping 試験中

2: 実行エラー

使用例

:PROTOCOL:PING:START

:PROTOCOL:PING:STATUS?

> 1

:PROTOCOL:PING:STOP

機能

Ping 試験を停止します。

文法

:PROTOCOL:PING:STOP

使用例

:PROTOCOL:PING:STOP

:PROTOCOL:PING:STATUS?

> 0

:PROTocol:PING:TIMEout

機能

Ping 試験のタイムアウト時間を設定, または問い合わせます。

文法

```
:PROTocol:PING:TIMEout <NR1>
:PROTocol:PING:TIMEout?
```

<NR1>: Ping タイムアウト時間 1～10 秒

レスポンスデータ

<NR1>

使用例

```
:PROTocol:PING:TIMEout 10
:PROTocol:PING:TIMEout?
> 10
```

:PROTocol:PING:VLAN

機能

Ping 試験の送信フレームの VLAN を設定, または問い合わせます。

文法

```
:PROTocol:PING:VLAN
<stack>[, <tpid1>, <pcp1>, <vid1>[, <tpid2>, <pcp2>, <vid2>]]
:PROTocol:PING:VLAN?
```

<stack>=0|1|2

<tpid1>, <tpid2>=<HEX>

<pcp1>, <vid1>, <pcp1>, <vid1>=<NR1>

<stack>に 0 を指定した場合は, <tpid1>以降を省略できます。

<stack>に 1 を指定した場合は, <tpid2>以降を省略できます。

レスポンスデータ

<stack>[, <tpid1>, <pcp1>, <vid1>[, <tpid2>, <pcp2>, <vid2>]]

使用例

```
:PROTocol:PING:VLAN 2, #H88A8, 0, 0, #H8100, 0, 0
:PROTocol:PING:VLAN?
> 2, #H88A8, 0, 0, #H8100, 0, 0
```

:ROUte:BERT**機能**

Port 設定の Frame BERT 設定を On または Off にします。
Port 設定の Frame BERT 設定を問い合わせます。

文法

```
:ROUte:BERT 0|1|OFF|ON
:ROUte:BERT?
```

```
0|OFF:      Frame BERT 設定 Off
1|ON:       Frame BERT 設定 On
```

レスポンスデータ

```
0|1
```

:ROUte:ETHer:NEGotiation:AUTO**機能**

OTU3, OTU4 アプリケーションでマッピングが GbE の場合, オートネゴシエーションを設定/問い合わせます。

文法

```
:ROUte:ETHer:NEGotiation:AUTO 0|1|OFF|ON
:ROUte:ETHer:NEGotiation:AUTO?
```

```
0|OFF:      オートネゴシエーション Off
1|ON:       オートネゴシエーション On
```

レスポンスデータ

```
0|1
```

使用例

```
:ROUte:ETHer:NEGotiation:AUTO ON
:ROUte:ETHer:NEGotiation:AUTO?
> 1
```

:ROUTe:FCONtrol

機能

Port 設定の Flow Control の On/Off を設定します。
Port 設定の Flow Control 設定を問い合わせます。

文法

```
:ROUTe:FCONtrol 0|1|OFF|ON  
:ROUTe:FCONtrol?
```

0|OFF: Flow Control 設定 Off
1|ON: Flow Control 設定 On

レスポンスデータ

0|1

:ROUTe:LFS:REPLy

機能

40GbE, 100GbE アプリケーションの場合, Port 設定の LFS Reply の On/Off を設定/問い合わせます。

OTU3 アプリケーションでマッピングが 40GbE, 10GbE の場合, および OTU4 アプリケーションでマッピングが 100GbE, 10GbE の場合には使用できません。

文法

```
:ROUTe:LFS:REPLy 0|1|OFF|ON  
:ROUTe:LFS:REPLy?
```

0|OFF: LFS Reply 設定 Off
1|ON: LFS Reply 設定 On

レスポンスデータ

0|1

:ROUTe:MODE

機能

Port のモードを設定/問い合わせます。

文法

```
:ROUTe:MODE LOOPBACK|NORMAL|THROUGH  
:ROUTe:MODE?
```

以下のどれかを設定します。

NORMAL ノーマルモード

LOOPBACK:ループバックモード

THROUGH:スルーモード

OTN3, OTN4 アプリケーションの場合に設定できます。

レスポンスデータ

LOOPBACK|NORMAL|THROUGH

使用例

ループバックに設定します。

```
:ROUTe:MODE LOOPBACK
:ROUTe:MODE?
> LOOPBACK
```

:ROUTe:MODE:THRough:OVERwrite:RANGe**機能**

スルーモードが OH Overwrite の場合、上書きするデータ範囲を設定/問い合わせます。

OTU3, OTU4 アプリケーションの場合に設定できます。

文法

```
:ROUTe:MODE:THRough:OVERwrite:RANGe
<type>[,<row>,<column>]
:ROUTe:MODE:THRough:OVERwrite:RANGe?
```

<type>=ALL|BYTE|ODU|OPU|OTU|OTU_ODU

<row>=<NR1>: <type>が BYTE の場合、データの行番号 1~4

<column>=<NR1>: <type>が BYTE の場合、データのカラム番号 1~16

レスポンスデータ

<type>[,<row>,<column>]

使用例

行 4, 列 13 の 1 バイトだけ上書きに設定します。

```
:ROUTe:MODE:THRough:OVERwrite:RANGe BYTE,4,13
:ROUTe:MODE:THRough:OVERwrite:RANGe?
> BYTE,4,13
```

:ROUTe:MODE:THRough:TYPE

機能

スルーモードの種類を設定/問い合わせます。

OTU3, OTU4 アプリケーションの場合に設定できます。

文法

:ROUTe:MODE:THRough:TYPE ANALYZED|OVERWRITE|TRANSPARENT

:ROUTe:MODE:THRough:TYPE?

以下のどれかを設定します。

ANALYZED

OVERWRITE

TRANSPARENT

レスポンスデータ

ANALYZED|OVERWRITE|TRANSPARENT

使用例

スルーモードの種類を OH Overwrite に設定します。

:ROUTe:MODE:THRough:TYPE OVERWRITE

:ROUTe:MODE:THRough:TYPE?

> OVERWRITE

:ROUTe:MPLStp:CWORD

機能

受信信号の MPLS-TP の Control Word の有り/無しを設定します。

40GbE, 100GbE アプリケーションの場合に設定できます。

文法

:ROUTe:MPLStp:CWORD 0|1

:ROUTe:MPLStp:CWORD?

0: Off (Control Word 無し)

1: On (Control Word 有り)

レスポンスデータ

0|1

:ROUte:VLAN:NUM**機能**

40GbE, 100GbE アプリケーションの場合, Port 設定の VLAN フィルタ数を設定, または問い合わせます。

文法

```
:ROUte:VLAN:NUM <NR1>
```

```
:ROUte:VLAN:NUM?
```

```
<NR1>=0~16
```

レスポンスデータ

```
<NR1>
```

使用例

フィルタ数を 4 に設定します。

```
:ROUte:VLAN:NUM 4
```

```
:ROUte:VLAN:NUM?
```

```
> 4
```

:ROUte:VLAN:VALue**機能**

40GbE, 100GbE アプリケーションの場合, Port 設定の VLAN のスタックと TPID を設定, または問い合わせます。

文法

```
:ROUte:VLAN:VALue <no>,<stack>,<tpid1>,<tpid2>
```

```
:ROUte:VLAN:VALue? <no>
```

```
<no>=<NR1>:          フィルタ番号 1~16
```

```
<stack>=1|2:   スタック
```

```
<tpid1>=<HEX>:      VLAN#1 TPID 0x0000~0xFFFF
```

```
<tpid2>=<HEX>:      VLAN#2 TPID 0x0000~0xFFFF
```

レスポンスデータ

```
<no>,<stack>,<tpid1>,<tpid2>
```

使用例

フィルタ番号 1 を VLAN スタック 2, TPID1 0x88A2, TPID2 0x8102 に設定します。

```
:ROUte:VLAN:VALue 1,2,#H88A2,#H8102
```

```
:ROUte:VLAN:VALue? 1
```

```
> 2,#H88A2,#H8102
```

:SENSe:MAPPing**機能**

:SOURce:MAPPingと同じ動作をします。

:SENSe:ODTU:MAIN:DETECT**機能**

受信側の Main TS (Tributary Slot)検出方法を設定/問い合わせます。
Main TS 検出方法を設定すると、TP/TS 設定の Combination は Offになります。

文法

:SENSe:ODTU:MAIN:DETECT <CHAR>

:SENSe:ODTU:MAIN:DETECT?

<CHAR>には、次のどれかを指定します。

<CHAR>	説明
MANUAL	TS を:SENSe:ODTU:MAIN:TSで指定します。
TP<NR1>	TP 番号を自動で検出します。
TS<NR1>	指定した TS 番号に格納されている TP 番号を自動で検出します。

<NR1>: ODTU4.1 の場合 1~80, ODTU4.8 の場合 1~10

レスポンスデータ

<CHAR>

使用例

ODTU4.1 の場合

:SENSe:ODTU:MAIN:DETECT TP80

:SENSe:ODTU:MAIN:DETECT?

> TP80

:SENSe:ODTU:MAIN:DETECT MANUAL

:SENSe:ODTU:MAIN:DETECT?

> MANUAL

:SENSe:ODTU:MAIN:TS**機能**

受信側の Main TS (Tributary Slot)を設定/問い合わせします。

ODTU4.1 では 1 つ, ODTU4.8 では 8 つ TS を設定します。

TS を設定すると, TP/TS 設定の Combination は Off になります。

文法

:SENSe:ODTU:MAIN:TS

<NR1>|{<NR1>,<NR1>,<NR1>,<NR1>,<NR1>,<NR1>,<NR1>,<NR1>}

:SENSe:ODTU:MAIN:TS?

<NR1>: ODTU4.1 の場合, 1~80

ODTU4.8 の場合, 1~10

レスポンスデータ

<NR1>|{<NR1>,<NR1>,<NR1>,<NR1>,<NR1>,<NR1>,<NR1>,<NR1>}

使用例

ODTU4.1 の場合

:SENSe:ODTU:MAIN:TS 80

:SENSe:ODTU:MAIN:TS?

> 80

ODTU4.8 の場合

:SENSe:ODTU:MAIN:TS 1,2,3,4,5,6,7,8

:SENSe:ODTU:MAIN:TS?

> 1,2,3,4,5,6,7,8

:SENSe:OTN:FEC

機能

Port 設定の GFEC Decode を設定, または問い合わせます。

文法

:SENSe:OTN:FEC 0|1|OFF|ON

:SENSe:OTN:FEC?

0|OFF: GFEC Decode 設定 Off

1|ON: GFEC Decode 設定 On

レスポンスデータ

0|1

使用例

GFEC Decode を On に設定します。

:SENSe:OTN:FEC ON

:SENSe:OTN:FEC?

> 1

:SENSe:PLM[:L]:PATtern

機能

PLM を検出するためのペイロードタイプを設定/問い合わせます。

ローオーダーのペイロードタイプを設定/問い合わせる場合は, :L を記載します。

文法

:SENSe:PLM[:L]:PATtern <HEX>

:SENSe:PLM[:L]:PATtern?

<HEX>: ペイロードタイプを表す 1 バイトの値

レスポンスデータ

<HEX>

使用例

ハイオーダーのペイロードタイプに 0x21 を設定します。

:SENSe:PLM:PATtern #H21

ローオーダーのペイロードタイプに 0x07 を設定します。

:SENSe:PLM:L:PATtern #H07

:SENSe:TIM:{SM|PM[:L]|TCM{1|2|3|4|5|6}}

機能

TIM の検出方法を設定/問い合わせます。

ローオーダの PM-TIM の検出方法を設定/問い合わせする場合は、:L を記載します。

文法

:SENSe:TIM:{SM|PM[:L]|TCM{1|2|3|4|5|6}} <CHAR>

:SENSe:TIM:{SM|PM[:L]|TCM{1|2|3|4|5|6}}?

<CHAR>

TIM の検出方法を以下から選択します。

OFF: TIM を検出しない。

DAPI: DAPI のみ TIM を検出する。

SAPI: SAPI のみ TIM を検出する。

SAPI_DAPI: SAPI と DAPI の TIM を検出する。

レスポンスデータ

DAPI|DAPI_SAPI|OFF|SAPI

使用例

SM-TIM の検出方法を Off にします。

```
:SENSe:TIM:SM OFF
```

PM-TIM の検出方法を DAPI にします。

```
:SENSe:TIM:PM DAPI
```

TCM2-TIM の検出方法を SAPI と DAPI にします。

```
:SENSe:TIM:TCM2 SAPI_DAPI
```

ハイオーダの PM-TIM の検出方法を問い合わせます。

```
:SENSe:TIM:PM?
```

```
> DAPI_SAPI
```

ローオーダの PM-TIM の検出方法を問い合わせます。

```
:SENSe:TIM:PM:L?
```

```
> OFF
```

:SENSe:TIM:{SM|PM[:L]|TCM{1|2|3|4|5|6}}:PATtern:{DAPI|SAPI}

機能

TIMを検出するためのパターンを設定/問い合わせします。

ローオーダーのPM-TIMの検出パターンを設定/問い合わせする場合は、:Lを記載します。

文法

:SENSe:TIM:{SM|PM[:L]|TCM{1|2|3|4|5|6}}:PATtern:{DAPI|SAPI} <HEX>

:SENSe:TIM:{SM|PM[:L]|TCM{1|2|3|4|5|6}}:PATtern:{DAPI|SAPI}?

<HEX>:パターンデータを表す 16 バイト以下の値

レスポンスデータ

<HEX>

使用例

SM-TTI の DAPI に"JPNMD1260A"を設定します。

:SENSe:TIM:SM:PATtern:DAPI
#H004A504E4D443132363041202020202020

TCM6-TTI の SAPI に"USAMD1260A"を設定します。

:SENSe:TIM:TCM6:PATtern:SAPI
#H005553404D443132363041202020202020

:SENSe:TPATtern:INVert**機能**

PRBS 試験パターン(受信)の設定を反転します。

No Frame アプリケーションの場合を除いて, このコマンドの設定値は

:SOURce:TPATtern:INVert と連動します。

どちらかのコマンドを送れば送信側, 受信側の両方が設定されます

文法

```
:SENSe:TPATtern:INVert 0|1|OFF|ON
```

```
:SENSe:TPATtern:INVert?
```

0|OFF: パターン反転 Off

1|ON: パターン反転 On

レスポンスデータ

0|1

使用例

試験パターンを送受信ともに反転します。

```
:SENSe:TPATtern:INVert ON
```

```
:SOURce:TPATtern:INVert ON
```

```
:SENSe:TPATtern:INVert?
```

```
> 1
```

```
:SOURce:TPATtern:INVert?
```

```
> 1
```

:SENSe:TPATtern:TYPE**機能**

受信側(Rx)のテストパターンの種類を設定します。

受信側(Rx)のテストパターンの種類を問い合わせます。

このコマンドの設定値は **:SOURce:TPATtern:TYPE** と連動します。どちらかのコマンドを送れば送信側, 受信側の両方が設定されます

文法

:SENSe:TPATtern:TYPE <CHAR>

:SENSe:TPATtern:TYPE?

<CHAR>

試験パターンのタイプを以下から選択します。

PRBS7

PRBS9

PRBS11

PRBS15

PRBS23

PRBS31

ALL0

ALL1

WORD16

SQUARE : Square Wave

Mapping によって, 設定できる試験パターンが変わります。

Mapping	試験パターン
40GbE, 100GbE, OTU4 Ethernet Client	PRBS31, ALL0, ALL1, WORD16
OTU3-PRBS, OTU4-PRBS	PRBS11, PRBS15, PRBS23, PRBS31, WORD16
OTU4-ODTU4.8-ODU2e-PRBS, OTU4-ODTU4.1-ODU0-PRBS	PRBS31, Word16
100GbE No Frame, 40GbE No Frame, OTU4 No Frame, OTU3 No Frame,	PRBS7, PRBS9, PRBS11, PRBS15, PRBS23, PRBS31, SQUARE

レスポンスデータ

ALL0 | ALL1 | PRBS7 | PRBS9 | PRBS11 | PRBS15 | PRBS23 | PRBS31 |
SQUARE | WORD16

使用例

:SENSe:TPATtern:TYPE WORD16

:SENSe:TPATtern:TYPE?

>WORD16

:SENSe:TPATtern:WORD**機能**

受信側(Rx)のテストパターンに Word16 を選択した場合のパターンを設定します。
 受信側(Rx)の Word16 パターンを問い合わせます。
 このコマンドの設定値は :SOURce:TPATtern:WORD と連動します。どちらかの
 コマンドを送れば送信側, 受信側の両方が設定されます

文法

```
:SENSe:TPATtern:WORD <HEX>
:SENSe:TPATtern:WORD?
```

<HEX>: Data Field の値 (2 byte)

レスポンスデータ

<HEX>

使用例

```
:SENSe:TPATtern:TYPE WORD16
:SENSe:TPATtern:WORD #HFF02
:SENSe:TPATtern:WORD?
>#HFF02
```

:SENSe:TRANsceiver:EQUalizer:CONTRol**機能**

トランシーバの Equalizer (Control)を設定します。
 トランシーバの Equalizer (Control)設定を問い合わせます。

文法

```
:SENSe:TRANsceiver:EQUalizer:CONTRol <lane>,<value >
:SENSe:TRANsceiver:EQUalizer:CONTRol? <lane>
```

<lane>,<value>=<NR1>

<lane>=<NR1>: レーン番号

<value>=<NR1>: Equalizer (Control)設定値

レスポンスデータ

<value>

Pre-Emphasis と VOD の設定は次のコマンドを参照してください。

```
:SOURce:TRANsceiver:EMPHasis:FIRSt
:SOURce:TRANsceiver:EMPHasis:PRE
:SOURce:TRANsceiver:EMPHasis:SECond
:SOURce:TRANsceiver:VOD
```

:SENSe:TRANsceiver:EQUalizer:DCGain

機能

トランシーバの Equalizer (DC Gain)を設定します。
トランシーバの Equalizer (DC Gain)設定を問い合わせます。

文法

```
:SENSe:TRANsceiver:EQUalizer:DCGain <lane>,<value>  
:SENSe:TRANsceiver:EQUalizer:DCGain? <lane>
```

<lane>,<value>=<NR1>

<lane>=<NR1>: レーン番号

<value>=<NR1>: Equalizer (DC Gain)設定値

レスポンスデータ

<value>

:SOURce:CFP:OPTical:OFF

機能

CFP の光出力をオフにします。

文法

```
:SOURce:CFP:OPTical:OFF
```

コマンドを送信してから、CFP の光出力がオフになるまで時間がかかります。

*OPCコマンドで処理の終了を確認してください。

:SOURce:CFP:OPTical:ON

機能

CFP の光出力をオンにします。

文法

```
:SOURce:CFP:OPTical:ON
```

コマンドを送信してから、CFP の光出力がオンになるまで時間がかかります。

*OPCコマンドで処理の終了を確認してください。

:SOURce:CFP:OPTical:STATus**機能**

CFP の光出力の状態を問い合わせます。

文法

:SOURce:CFP:OPTical:STATus?

レスポンスデータ

0: 光出力オフ

1: 光出力オン

使用例

:SOURce:CFP:OPTical:STATus?

> 1

:SOURce:CLOCK**機能**

Clock Source (送信クロックのクロック源) を設定します。

Clock Source (送信クロックのクロック源)設定を問い合わせます。

文法

:SOURce:CLOCK <CHAR>

:SOURce:CLOCK?

<CHAR>には、以下のどれかを選択します。

<CHAR>	送信クロックのクロック源
INTERNAL	内部発振器のクロック
INPUT_10MHZ	「10 MHz Input」端子に入力したクロック
INPUT_REFCLK	「Tx Ref Clock Input」端子に入力したクロック
RECEIVE	受信クロックに同期したクロック
INPUT_SYNC	「Sync Input」端子に入力したクロック

レスポンスデータ

<char>

使用例

:SOURce:CLOCK INTERNAL

:SOURce:CLOCK?

> INTERNAL

:SOURce:CLOCK:FREQuency:OFFSet**機能**

送信クロック周波数のオフセットを設定します。

送信クロック周波数のオフセット設定を問い合わせます。

文法

:SOURce:CLOCK:FREQuency:OFFSet <NR1>

:SOURce:CLOCK:FREQuency:OFFSet?

<NR1>: 基準クロック周波数からのオフセット (−120 ~ +120 ppm)

レスポンスデータ

< NR1>

使用例

送信クロック周波数のオフセットを 1 ppm に設定します。

:SOURce:CLOCK:FREQuency:OFFSet 1

:SOURce:CLOCK:FREQuency:OFFSet?

> 1

:SOURce:CLOCK:OUTPut:DIVide**機能**

「Tx Ref Clock Output」端子へ出力するクロックの分周比を設定します。
「Tx Ref Clock Output」端子へ出力するクロックの分周比を問い合わせます。

文法

```
:SOURce:CLOCK:OUTPut:DIVide DIV16|DIV64
:SOURce:CLOCK:OUTPut:DIVide?
```

次のどちらかを設定します。

DIV16: 1/16

DIV64: 1/64

レスポンスデータ

DIV16|DIV64

使用例

```
:SOURce:CLOCK:OUTPut:DIVide DIV64
:SOURce:CLOCK:OUTPut:DIVide?
> DIV64
```

:SOURce:CLOCK:OUTPut:M10**機能**

「10 MHz Output」端子へ出力する 10 MHz 信号を設定します。
「10 MHz Output」端子へ出力する 10 MHz 信号の設定を問い合わせます。

文法

```
:SOURce:CLOCK:OUTPut:M10 INTERNAL|LOCKED
:SOURce:CLOCK:OUTPut:M10?
```

次のどちらかを設定します。

INTERNAL: 内部発振器の 10 MHz

LOCKED: 受信クロックに同期した 10 MHz

レスポンスデータ

INTERNAL|LOCKED

使用例

```
:SOURce:CLOCK:OUTPut:M10 INTERNAL
:SOURce:CLOCK:OUTPut:M10?
> INTERNAL
```

:SOURce:CLOCK:PAYLoad:OFFSet[:L]**機能**

OTU3, OTU4 アプリケーションの場合、ペイロードオフセットを設定/問い合わせます。

ローオーダーのペイロードオフセットを設定/問い合わせる場合は、:L を記載します。

文法

:SOURce:CLOCK:PAYLoad:OFFSet[:L] <NR1>

:SOURce:CLOCK:PAYLoad:OFFSet[:L]?

<NR1>: ペイロードオフセット

ローオーダーが存在しない場合 (−120 ~ +120 ppm)

ハイオーダー/ローオーダーが存在する場合

ハイオーダー:(−40 ~ +40 ppm) ローオーダー:(−120 ~ +120 ppm)

レスポンスデータ

< NR1>

使用例

ペイロードオフセットを 10ppm に設定します。

ODTU4.1, ODTU4.8 の場合はハイオーダーのペイロードオフセットを設定します。

:SOURce:CLOCK:PAYLoad:OFFSet 10

ローオーダーのペイロードオフセットを−50ppm に設定します。

:SOURce:CLOCK:PAYLoad:OFFSet:L −50

:SOURce:EALarm:BIT**機能**

エラー・アラームを付加するビットを設定/問い合わせます。

文法

:SOURce:EALarm:BIT <BINARY>

:SOURce:EALarm:BIT?

<BINARY>

2 進数で、エラー・アラームを付加するビットには 1, 付加しないビットには 0 を設定します。最初の桁がビット 0 に対応します。

たとえば, CRC8 Error では 8 bit, CRC5 Error では 5 bit を設定します。設定できるビット数より少ないビット数を設定した場合, 残りのビットは 0 を設定したものとみなされます。また, 設定できるビット数より多いビット数を設定した場合は, 余分なビットは無視されます。

ただし, Invalid JC1&JC2 の場合は, JC2 の Bit 0~7, JC1 の Bit 0~7 の順にビットを設定します。

レスポンスデータ

<BINARY>

桁数は、設定できるビット数によって変わります。

使用例

CRC8 Error にビット 0, 3 にエラー・アラームを付加する場合

:SOURce:EALarm:BIT #B10010000

または

:SOURce:EALarm:BIT #B1001

Invalid JC1&JC2 のときに、JC2 の Bit0,1,2 と JC1 の Bit 0 にエラーを付加する場合

:SOURce:EALarm:BIT #B11100000010000000

:SOURce:EALarm:FAS:EXCLude

機能

OTN フレームの FAS をエラー挿入範囲から除外するかを設定/問い合わせます。

文法

:SOURce:EALarm:FAS:EXCLude 0|1|OFF|ON

:SOURce:EALarm:FAS:EXCLude?

0|OFF: FAS Exclude Off

1|ON: FAS Exclude On

レスポンスデータ

0|1

使用例

:SOURce:EALarm:FAS:EXCLude ON

:SOURce:EALarm:FAS:EXCLude?

> 1

:SOURce:EALarm:LANE**機能**

エラー・アラームを付加するレーンを設定します。

エラー・アラームを付加するレーンを問い合わせます。

文法

:SOURce:EALarm:LANE <BINARY>

:SOURce:EALarm:LANE?

<BINARY>

2 進数で、エラー・アラームを付加するレーンのビットには 1, 付加しないレーンのビットには 0 を設定します(先頭から順に レーン 0, レーン 1, ... に対応)。

たとえば, 100GbE/OTU4 では 20 bit (レーン 0~19), 40GbE/OTU3 では 4 bit (レーン 0~3)を設定します。**Lane** 数より少ないビット数を設定した場合, 残りのビットは 0 を設定したものとみなされます。また, レーン数より多いビット数を設定した場合は, 余分なビットは無視されます。

レスポンスデータ

<BINARY>

使用例

100GbE でレーン 0, 3, 9 にエラー・アラームを付加する場合

:SOURce:EALarm:LANE #B100100000010000000000

または

:SOURce:EALarm:LANE #B10010000001

:SOURce:EALarm:START[:AUNit]**機能**

エラー・アラーム付加を開始します。

すでにエラー・アラーム付加が動作している場合は, 再スタートとなります。

:AUNit を追加すると, 接続しているすべてのユニットのエラー・アラーム付加を同時に開始します。

文法

:SOURce:EALarm:START[:AUNit]

使用例

エラー・アラーム付加の開始を待つ場合は以下のようにします。

1. **:SOURce:EALarm:START:EVENT?** を送信します。
開始イベントがクリアされます。
2. **:SOURce:EALarm:START** を送信します。
エラー・アラーム付加が開始します。
3. **:SOURce:EALarm:START:EVENT?** を送信します。
レスポンスデータが 1 (開始) になるまで送信を繰り返します。

:SOURce:EALarm:START:EVENT[:AUNit]

機能

エラー・アラーム付加の開始イベントを問い合わせます。

文法

:SOURce:EALarm:START:EVENT[:AUNit]?

レスポンスデータ

0|1

0: エラー・アラーム挿入の開始前

本コマンドで読み取った後にレスポンスデータが 0 になります。

1: エラー・アラーム挿入が開始済み

ヘッダーに:AUNit を付けた場合は、接続しているすべてのユニットのエラー・アラーム挿入が開始すると 1 となります。

レスポンスデータ 1 を読み取ると、イベントの値は 0 にクリアされます。

使用例

:SOURce:EALarm:START[:AUNit]の使用例を参照してください。

:SOURce:EALarm:STATus[:AUNit]

機能

エラー・アラーム付加の動作状態を問い合わせます。

文法

:SOURce:EALarm:STATus[:AUNit]?

レスポンスデータ

0|1

0: エラー・アラーム付加動作停止中

1: エラー・アラーム付加動作中

ヘッダーに:AUNit を付けた場合は接続したユニットのどれか 1 つ以上が開始すると 1, すべてのユニットが停止すると 0 になります。

使用例

:SOURce:EALarm:STOP[:AUNit]の使用例を参照してください。

:SOURce:EALarm:STOP[:AUNit]

機能

エラー・アラーム付加を停止します。

:AUNit を追加すると、接続しているすべてのユニットのエラー・アラーム付加を停止します。

文法

:SOURce:EALarm:STOP[:AUNit]

使用例

エラー・アラーム付加の停止を待つ場合は次のようにします。

1. :SOURce:EALarm:STOP を送信します。
エラー・アラーム付加が停止します。
2. :SOURce:EALarm:STATus? を送信します。
レスポンスデータが 0 (停止) になるまで送信を繰り返します。

:SOURce:EALarm:SUBRow

機能

エラー・アラームを付加する SubRow を設定/問い合わせます。

文法

:SOURce:EALarm:SUBRow <NR1>

:SOURce:EALarm:SUBRow?

<NR1>: SubRow の番号

レスポンスデータ

<NR1>

使用例

:SOURce:EALarm:SUBRow 1

:SOURce:EALarm:SUBRow?

> 1

:SOURce:EALarm:TIMing:BURSt**機能**

エラー・アラーム付加のタイミングが **Burst** の場合のカウンタを設定します。

エラー・アラーム付加のタイミングが **Burst** の場合のカウンタ設定を問い合わせます。

文法

```
:SOURce:EALarm:TIMing:BURSt <NR1>
```

```
:SOURce:EALarm:TIMing:BURSt?
```

```
<NR1>
```

エラー・アラーム付加数を設定します。

レスポンスデータ

```
<NR1>
```

使用例

```
:SOURce:EALarm:TYPE INV_SH00
```

```
:SOURce:EALarm:TIMing:TYPE BURST
```

```
:SOURce:EALarm:TIMing:BURSt 1
```

```
:SOURce:EALarm:TIMing:BURSt?
```

```
> 1
```

:SOURce:EALarm:TIMing:ERRor**機能**

エラー・アラーム付加のタイミングが **Alternate** の場合の、エラーフレーム数を設定します。

エラー・アラーム付加のタイミングが **Alternate** の場合の、エラーフレーム数の設定を問い合わせます。

文法

```
:SOURce:EALarm:TIMing:ERRor <NR1>
```

```
:SOURce:EALarm:TIMing:ERRor?
```

```
<NR1>:          エラー・アラームを付加する(Error)フレーム数
```

レスポンスデータ

```
<NR1>
```

使用例

:SOURce:EALarm:TIMing:NORMalの使用例を参照してください。

:SOURce:EALarm:TIMing:NORMal

機能

エラー・アラーム付加のタイミングが **Alternate** の場合の、ノーマルフレーム数を設定します。

エラー・アラーム付加のタイミングが **Alternate** の場合の、ノーマルフレーム数の設定を問い合わせます。

文法

:SOURce:EALarm:TIMing:NORMal <NR1>

:SOURce:EALarm:TIMing:NORMal?

<NR1>: エラー・アラームなし(Normal)のフレーム数

レスポンスデータ

<NR1>

使用例

```
:SOURce:EALarm:TYPE FAS_MLD
:SOURce:EALarm:TIMing:TYPE ALTERNATE
:SOURce:EALarm:TIMing:ERROR 1
:SOURce:EALarm:TIMing:NORMal 2
:SOURce:EALarm:TIMing:ERROR?
> 1
:SOURce:EALarm:TIMing:NORMal?
> 2
```

:SOURce:EALarm:TIMing:RATE

機能

:SOURce:EALarm:TIMing:TYPE で Rate を選択したときの, エラー・アラーム
付加レートを設定します。
エラー・アラーム付加レートを問い合わせます。

文法

:SOURce:EALarm:TIMing:RATE <CHAR>|<NR3>
:SOURce:EALarm:TIMing:RATE?

<NR3>: 付加レートを数値で設定します。
3.6×10⁻⁶を設定する場合は 3.6E-6 とします。

<CHAR>
以下の文字列で付加レートを設定します。

<CHAR>	付加レート
R1E_2	1.0E-2
R1E_3	1.0E-3
R1E_4	1.0E-4
R1E_5	1.0E-5
R1E_6	1.0E-6
R1E_7	1.0E-7
R1E_8	1.0E-8
R1E_9	1.0E-9

レスポンスデータ

<NR3>

使用例

:SOURce:EALarm:TYPE BIT_ERROR
:SOURce:EALarm:TIMing:TYPE RATE
:SOURce:EALarm:TIMing:RATE R1E_6
:SOURce:EALarm:TIMing:RATE?
> 1.0E-6

:SOURce:EALarm:TIMing:TYPE**機能**

エラー・アラーム付加のタイミングを設定します。

エラー・アラーム付加のタイミング設定を問い合わせます。

文法

:SOURce:EALarm:TIMing:TYPE <CHAR>

:SOURce:EALarm:TIMing:TYPE?

<CHAR>

タイミングを以下から選択します。

<CHAR>	タイミング
SINGLE	Single
BURST	Burst
RATE	Rate
ALTERNATE	Alternate
ALL	All

Ethernet の Bit Error では RATE を指定します。

BURST を選択した場合は、:SOURce:EALarm:TIMing:BURSt でエラー・アラームの付加量を設定します。

RATE を選択した場合は、:SOURce:EALarm:TIMing:RATEでエラー・アラームの付加レートを設定します。

ALTERNATE を選択した場合は、:SOURce:EALarm:TIMing:ERRorと:SOURce:EALarm:TIMing:NORMalでエラー・アラームの付加量を設定します。

選択可能な値は、アプリケーションとエラー・アラーム付加の種類の選択によって異なります。

エラー・アラーム付加の種類を変更した場合、選択可能な値のいずれかに丸められることがあります。詳細については『MD1260A 40/100G イーサネット アナライザ 取扱説明書』を参照してください。

レスポンスデータ

<CHAR>

使用例

エラー・アラーム付加のタイミングを Single に設定します。

:SOURce:EALarm:TIMing:TYPE SINGLE

:SOURce:EALarm:TIMing:TYPE?

> SINGLE

:SOURce:EALarm:TYPE

機能

エラー・アラーム付加の種類を設定します。

エラー・アラーム付加の種類の設定を問い合わせます。

文法

:SOURce:EALarm:TYPE <CHAR>

:SOURce:EALarm:TYPE?

<CHAR>: エラー・アラームの種類を以下から選択します。

種別を変えるとタイミング設定は初期化されます。

40GbE, 100GbE:

エラー・アラームの種類	<CHAR>
Invalid Sync Header (00)	INV_SH00
Invalid Sync Header (11)	INV_SH11
Invalid Alignment Marker	INV_MARKER
BIP Error	BIP
Invalid Block Type (0x00)	INV_BT00
Invalid Block Type (0x2d)	INV_BT2D
Invalid Block Type (0x33)	INV_BT33
Invalid Block Type (0x66)	INV_BT66
Hi-BER	HIBER
LF	LF
RF	RF
Bit Errors	BIT_ERROR

OTN3, OTN4:

レイヤ	エラーの種類	<CHAR>
LLD	FAS	FAS_LLD
OTU3/ OTU4	FAS	FAS
	SM-BIP8	SM_BIP8
	SM-BEI	SM_BEI
	Uncorrectable Error	BIT_ALL_U
	Correctable Error	BIT_ALL_C
	Bit All	BIT_ALL

OTN3, OTN4(続き):

レイヤ	エラーの種類	<CHAR>
ODU3/ ODU4	PM-BIP8	PM_BIP8
	PM-BEI	PM_BEI
	TCMi-BIP8	TCM{1 2 3 4 5 6}_BIP8
	TCMi -BEI	TCM{1 2 3 4 5 6}_BEI
ODU2e/ ODU0	FAS	FAS_L
	PM-BIP8	PM_BIP8_L
	PM-BEI	PM_BEI_L
GMP	CRC8 Error	GMP_CRC8
	CRC5 Error	GMP_CRC5
	Invalid JC1	GMP_JC1
	Invalid JC2	GMP_JC2
	Invalid JC1&JC2	GMP_JC12
GMP(Lo)	CRC8 Error	GMP_CRC8_L
	Invalid JC1	GMP_JC1_L
	Invalid JC2	GMP_JC2_L
	Invalid JC1&JC2	GMP_JC12_L
GFP-T	Superblock CRC Error	GFP_SCRC
	cHEC Error	GFP_CHEC
	tHEC Error	GFP_THEC
Ethernet	Invalid Sync Header (00)	INV_SH00
	Invalid Sync Header (11)	INV_SH11
	Invalid Block Type (0x00)	INV_BT00
	Invalid Block Type (0x2d)	INV_BT2D
	Invalid Block Type (0x33)	INV_BT33
	Invalid Block Type (0x66)	INV_BT66
	Invalid Alignment Marker	INV_MARKER
	BIP Error	BIP
	LF	LF
	RF	RF
	66B Error	ERR_66B
	10B Error	ERR_10B
Test Pattern	Bit Errors	BIT_ERROR

OTN3, OTN4(続き):

レイヤ	アラームの種類	<CHAR>
LLD	OOF/LOF	OOF_LLD
	OOR/LOR	OOR
OTU3/ OTU4	OOF/LOF	OOF
	OOM/LOM	OOM
	SM-TIM	SM_TIM
	SM-BIAE	SM_BIAE
	SM-BDI	SM_BDI
	SM-IAE	SM_IAE
ODU3/ ODU4	ODU-AIS	AIS_ODU
	ODU-OCI	OCI
	ODU-LCK	LCK
	PM-TIM	PM_TIM
	PM-BDI	PM_BDI
	TCMi-TIM	TCM{1 2 3 4 5 6}_TIM
	TCMi-BIAE	TCM{1 2 3 4 5 6}_BIAE
	TCMi IAE	TCM{1 2 3 4 5 6}_IAE
	TCMi BDI	TCM{1 2 3 4 5 6}_BDI
	TCMi-LTC	TCM{1 2 3 4 5 6}_LTC
OPU3/ OPU4	Client-AIS	AIS_C
	CSF	CSF
ODU2e/ ODU0	OOF	OOF_L
	OOM	OOM_L
	ODU-AIS	AIS_ODU_L
	ODU-OCI	OCI_L
	ODU-LCK	LCK_L
	PM-TIM	PM_TIM_L
	PM-BDI	PM_BDI_L
Ethernet	High BER	HIBER

40GbE No Frame:, 100GbE No Frame, OTN3 No Frame:, OTN4 No Frame::

エラー・アラームの種類	<CHAR>
Bit Errors	BIT_ERROR

付加タイミングを選択できるエラー・アラームの場合は
:SOURce:EALarm:TIMing:TYPEで設定します。
また、レーンに対するエラー・アラームを設定した場合は、
:SOURce:EALarm:LANEでエラー・アラームの挿入レーンを設定します。

各種エラー・アラームの詳細については『MD1260A 40/100G イーサネット アナライザ 取扱説明書』を参照してください。

レスポンスデータ

<CHAR>

使用例

```
:SOURce:EALarm:TYPE?  
> BIT_ERROR
```

:SOURce:GFP:CSF:REPLacement

機能

GFP-T のエラーを置き換えるデータを設定/問い合わせます。

文法

```
:SOURce:GFP:CSF:REPLacement BLOCK|CSF  
:SOURce:GFP:CSF:REPLacement?
```

次のどちらかを選択します。

BLOCK: Ethernet Block

CSF: GFP-T CSF

レスポンスデータ

BLOCK|CSF

使用例

```
:SOURce:GFP:CSF:REPLacement BLOCK  
:SOURce:GFP:CSF:REPLacement?  
> BLOCK
```

:SOURce:GFP:{PTI|UPI}**機能**

GFP-T のペイロードを設定/問い合わせます。

文法

:SOURce:GFP:{PTI|UPI} <BINARY>

:SOURce:GFP:{PTI|UPI}?

<BINARY>: PTI の場合, 3 ビット
UPI の場合, 8 ビット

レスポンスデータ

<BINARY>

使用例

```
:SOURce:GFP:PTI #B000
:SOURce:GFP:PTI?
> #B001
:SOURce:GFP:UPI #B000000110
:SOURce:GFP:UPI?
> #B000000110
```

:SOURce:MAPPing**機能**

アプリケーションが 100GbE No Frame, または OTU4 No Frame の場合に, 物理レーンの数を設定します。

起動しているアプリケーションの種類を問い合わせます。

アプリケーションは:SYSTem:CONFIgで設定します。

文法

:SOURce:MAPPing <CHAR>

:SOURce:MAPPing?

<CHAR>

以下のどれかを設定します。

アプリケーション(レーン数)	<CHAR>
100GbE No Frame (10Lane)	E100G_N10
100GbE No Frame (20Lane)	E100G_N20
OTU4 No Frame (10Lane)	OTU4_N10
OTU4 No Frame (20Lane)	OTU4_N20

レスポンスデータ

<CHAR>

アプリケーション/マッピング	<CHAR>
40GbE	E40G
100GbE	E100G
ODU3-PRBS	OTU3
ODU4-PRBS	OTU4
ODU4-100GbE	OTU4_E
ODTU4.8-ODU2e-PRBS	OTU4_ODU2E
ODTU4.8-ODU2e-10GbE	OTU4_ODU2E_E
ODTU4.1-ODU0-PRBS	OTU4_ODU0
ODTU4.1-ODU0-GbE	OTU4_ODU0_E
40GbE No Frame	E40G_N
100GbE No Frame (10Lane)	E100G_N10
100GbE No Frame (20Lane)	E100G_N20
OTU3 No Frame	OTU3_N
OTU4 No Frame (10Lane)	OTU4_N10
OTU4 No Frame (20Lane)	OTU4_N20

使用例

起動しているアプリケーションが 100GbE のとき

:SOURCE:MAPPING?

> E100G

起動しているアプリケーションが 100GbE No Frame で, 10 Lane の測定をするとき

:SOURCE:MAPPING E100G_N10

:SOURCE:MAPPING?

> 100GE_N10

:SOURce:MAPPing:LANE

機能

Tx Lane に対するレーンマーカの値を, Tx Lane 0 から順番にレーンの数だけ設定します。

本リモートコマンドを送信すると, 操作画面上の Allow to Overlap ボタンは On になります。

Tx Lane に対するレーンマーカの値を問い合わせます。

文法

:SOURce:MAPPing:LANE <NR1>,<NR1>,<NR1>,...

:SOURce:MAPPing:LANE?

<NR1>

Tx Lane に対するレーンマーカの値

レスポンスデータ

<NR1>,<NR1>,<NR1>,...

使用例

100GbE/OTU4 の場合

:SOURce:MAPPing:LANE

0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19

40GbE/OTU3 の場合

:SOURce:MAPPing:LANE 0,1,2,3

:SOURce:ODTU:DUMMy:PATtern

機能

Dummy チャンネルに送信するデータを設定/問い合わせます。

文法

```
:SOURce:ODTU:DUMMy:PATtern COPY_MAIN|PRBS11
:SOURce:ODTU:DUMMy:PATtern?
```

次のどちらかを選択します。

COPY_MAIN: Main チャンネルと同じデータ

PRBS11: 擬似ランダムビットシーケンス

レスポンスデータ

COPY_MAIN|PRBS11

使用例

```
:SOURce:ODTU:DUMMy:PATtern COPY_MAIN
:SOURce:ODTU:DUMMy:PATtern?
> COPY_MAIN
```

:SOURce:ODTU:MAIN:TP

機能

Main TP (Tributary Port)を設定/問い合わせます。

文法

```
:SOURce:ODTU:MAIN:TP <NR1>
:SOURce:ODTU:MAIN:TP?
```

<NR1>の設定範囲は次のとおりです。

マッピング	<NR1>
ODTU4.1-ODU0-GbE/PRBS	1~80
ODTU4.8-ODU2e-10GbE/PRBS	1~10

レスポンスデータ

<NR1>

使用例

```
:SOURce:ODTU:MAIN:TP 16
:SOURce:ODTU:MAIN:TP?
> 16
```

:SOURce:ODTU:MAIN:TS**機能**

Main TP を格納する TS (Tributary Slot)を設定/問い合わせます。
ODTU4.1 では 1 つ, ODTU4.8 では 8 つ TS を設定します。

文法

```
:SOURce:ODTU:MAIN:TS
<NR1>|<NR1>,<NR1>,<NR1>,<NR1>,<NR1>,<NR1>,<NR1>,<NR1>
:SOURce:ODTU:MAIN:TS?
```

<NR1>: 1~80

レスポンスデータ

```
<NR1>|<NR1>,<NR1>,<NR1>,<NR1>,<NR1>,<NR1>,<NR1>,<NR1>
```

使用例

ODTU4.1 の場合

```
:SOURce:ODTU:MAIN:TS 2
:SOURce:ODTU:MAIN:TS?
> 2
```

ODTU4.8 の場合

```
:SOURce:ODTU:MAIN:TS 33,34,35,36,37,38,39,40
:SOURce:ODTU:MAIN:TS?
> 33,34,35,36,37,38,39,40
```

:SOURce:OTN:FEC**機能**

OTN フレームの GFEC Encode を設定/問い合わせます。

文法

```
:SOURce:OTN:FEC 0|1|OFF|ON
:SOURce:OTN:FEC?
```

0|OFF: GFEC Encode Off
1|ON: GFEC Encode On

レスポンスデータ

0|1

使用例

```
:SOURce:OTN:FEC OFF
:SOURce:OTN:FEC?
> 0
```

:SOURCE:OTN:OH[:L]

機能

OTN フレームのオーバーヘッドの値を設定/問い合わせます。

:L を省略すると、ハイオーダ(OTU4)の OH を設定/問い合わせます。

:L を付けると、ローオーダ(ODU2e, または ODU0)の OH を設定/問い合わせます。

文法

:SOURCE:OTN:OH[:L] <row>,<column>,<data>

:SOURCE:OTN:OH[:L]? <row>,<column>

<row>=<NR1>: 行番号 1~4

<column>=<NR1>: 列番号 1~16

<data>=<HEX>: 値 #H00~#HFF

レスポンスデータ

<data>=<HEX>

使用例

:SOURCE:OTN:OH:L 4,9,#H10

:SOURCE:OTN:OH:L? 4,9

> #H10

:SOURCE:SKEW:BIT

機能

スキュー発生量を設定します。

スキュー発生量を問い合わせます。

文法

:SOURCE:SKEW:BIT <NR1>

:SOURCE:SKEW:BIT?

<NR1>: bit 単位の Skew の設定値

レスポンスデータ

<NR1>

使用例

:SOURCE:SKEW:NSの使用例を参照してください。

:SOURce:SKEW:LANE**機能**

Skew を挿入するレーンを設定します。
Skew を挿入するレーンの設定を問い合わせます。

文法

:SOURce:SKEW:LANE <BINARY>
:SOURce:SKEW:LANE?

<BINARY>

2 進数で, Skew を入れるレーンのビットに 1, 入れないレーンのビットに 0 を設定します(先頭から順に レーン 0, レーン 1, ... に対応します)。

たとえば, 100GbE/OTU4 では 20 bit(レーン 0~19), 40GbE/OTU3 では 4 bit(レーン 0~3)設定します。Lane 数より少ないビット数を設定した場合, 残りのビットは 0 を設定したものとみなされます。また, レーン数より多いビット数を設定した場合, 余分なビットは無視されます。

レスポンスデータ

<BINARY>

使用例

100GbE/OTU4 で Tx Lane 0, 3, 9 に Skew を入れる場合

```
:SOURce:SKEW:LANE #B1001000000100000000000
```

または

```
:SOURce:SKEW:LANE #B10010000001
```

```
:SOURce:SKEW:LANE?
```

```
> #B1001000000100000000000
```

例: 100GbE/OTU4 で Physical Lane 9 に Skew を入れる場合

```
:SOURce:SKEW:LANE #B00000000001
```

注:

:SOURce:SKEW:TYPE の設定を変更すると本コマンドの設定は初期化されます。

:SOURce:SKEW:NS

機能

スキューの時間を問い合わせます。

文法

:SOURce:SKEW:NS?

レスポンスデータ

<NR2>: Skew の設定値を ns 単位に変換した値

使用例

100GbE で 66 bit (12.8 ns) のスキューを発生させるとき

:SOURce:SKEW:BIT 66

:SOURce:SKEW:NS?

>12.800

:SOURce:SKEW:TYPE

機能

Skew を挿入するレーンを設定します。

Skew を挿入するレーンを問い合わせします。

文法

:SOURce:SKEW:TYPE PHY_LANE|TX_LANE

:SOURce:SKEW:TYPE?

次のどちらかを選択します。

TX_LANE: Tx Lane

PHY_LANE: Physical Lane

:SOURce:SKEW:LANE でスキューを発生させるレーンを設定し、:SOURce:SKEW:BITでスキューの発生量をビット数で設定します。設定したスキューの発生量の時間換算値を知りたい場合は、:SOURce:SKEW:NS?で問い合わせます。

レスポンスデータ

PHY_LANE|TX_LANE

使用例

Tx Lane にスキューを挿入する場合

:SOURce:SKEW:TYPE TX_LANE

:SOURce:SKEW:TYPE?

> TX_LANE

:SOURce:STReam:BURSt:CONTRol:VALue**機能**

バースト中のギャップサイズを設定します。
バースト中のギャップサイズ設定を問い合わせます。

文法

```
:SOURce:STReam:BURSt:CONTRol:VALue <NR1>  
:SOURce:STReam:BURSt:CONTRol:VALue?
```

<NR1>: ギャップサイズ (frames)

レスポンスデータ

<NR1>

:SOURce:STReam:BURSt:ENABLE**機能**

バーストを送信するかどうかを設定します。
バースト送信の設定を問い合わせます。

文法

```
:SOURce:STReam:BURSt:ENABLE 0|1|OFF|ON  
:SOURce:STReam:BURSt:ENABLE?
```

0|OFF: バーストを送信しない

1|ON: バーストを送信する

レスポンスデータ

0|1

:SOURce:STReam:BURSt:SIZE**機能**

バーストサイズ(1 バーストで送信するフレーム数)を設定します。
バーストサイズを問い合わせます。

文法

```
:SOURce:STReam:BURSt:SIZE <NR1>  
:SOURce:STReam:BURSt:SIZE?
```

<NR1>: フレーム数

レスポンスデータ

<NR1>

:SOURce:STReam:CONTRol:RANGe**機能**

イーサネットフレームの送信間隔 (ギャップサイズ)の範囲を設定します。
イーサネットフレームの送信間隔の範囲を問い合わせます。

文法

```
:SOURce:STReam:CONTRol:RANGe <value1>,<value2>  
:SOURce:STReam:CONTRol:RANGe?
```

<value1>, <value2> = <NR1>

ギャップサイズ (byte) の最小値と最大値を設定します。
2 つの値のうち、小さい方が最小値、大きい方が最大値となります。

レスポンスデータ

<value1>, <value2> = <NR1>

<value1>: ギャップサイズの最小値

<value2>: ギャップサイズの最大値

使用例

ギャップサイズを 12 バイトから 96 バイトの範囲で送信します。

```
:SOURce:STReam:CONTRol:TYPE RANDOM  
:SOURce:STReam:CONTRol:RANGe 12,96
```

:SOURce:STReam:CONTRol:TYPE**機能**

イーサネットフレームの送信間隔を一定にするかランダムにするか設定します。
イーサネットフレームの送信間隔の設定を問い合わせます。

文法

```
:SOURce:STReam:CONTRol:TYPE FIXED|RANDOM  
:SOURce:STReam:CONTRol:TYPE?
```

設定の種類を以下から選択します。

FIXED Fixed

RANDOM Random

FIXED を選択したときは、:SOURce:STReam:CONTRol:VALueでギャップサイズを設定します。

RANDOM を選択したときは、:SOURce:STReam:CONTRol:RANGeでギャップサイズの範囲を設定します。

%や bit/s などの送信レートの設定もこのコマンドで設定できます。

操作画面の Test Pattern 設定画面の Stream Control 設定の単位を切り替えると、送信したいレートに対するギャップサイズを確認できます。

レスポンスデータ

FIXED | RANDOM

使用例

ギャップサイズを 12 バイト (100%) で送信します。

:SOURCE:STREAM:CONTROL:TYPE FIXED

:SOURCE:STREAM:CONTROL:VALUE 12

:SOURCE:STREAM:CONTROL:VALUE

機能

イーサネットフレームの送信間隔 (ギャップサイズ) を設定/問い合わせます。

文法

:SOURCE:STREAM:CONTROL:VALUE <NR2>

:SOURCE:STREAM:CONTROL:VALUE?

<NR2>: ギャップサイズ (byte)

レスポンスデータ

<NR2>

使用例

ギャップサイズを 12 バイト (100%) で送信します。

:SOURCE:STREAM:CONTROL:TYPE FIXED

:SOURCE:STREAM:CONTROL:VALUE 12

:SOURCE:STREAM:COUNT

機能

ストリーム中のフレーム数またはバースト数を設定/問い合わせます。

文法

:SOURCE:STREAM:COUNT <NR1>

:SOURCE:STREAM:COUNT?

<NR1>: フレーム数またはバースト数

レスポンスデータ

<NR1>

:SOURce:STReam:DURation:FRAMes

機能

フレーム数, またはストリームの繰り返し数を設定/問い合わせます。
:SOURce:STReam:DURation:REPeat:COUNTと同じ処理をします。

文法

:SOURce:STReam:DURation:FRAMes <NR1>
:SOURce:STReam:DURation:FRAMes?

<NR1>: 1 ストリームの場合はフレーム数,
 複数ストリームの場合は, 複数ストリームの繰り返し数

レスポンスデータ

<NR1>

使用例

:SOURce:STReam:DURation:TYPEの使用例を参照してください。

:SOURce:STReam:DURation:REPeat:COUNT

機能

フレーム数, またはストリームの繰り返し数を設定/問い合わせます。
:SOURce:STReam:DURation:FRAMesと同じ処理をします。

文法

:SOURce:STReam:DURation:REPeat:COUNT <NR1>
:SOURce:STReam:DURation:REPeat:COUNT?

<NR1>: 1 ストリームの場合はフレーム数
 複数ストリームの場合は, 複数ストリームの繰り返し数

レスポンスデータ

<NR1>

使用例

:SOURce:STReam:DURation:TYPEの使用例を参照してください。

:SOURce:STReam:DURation:TYPE**機能**

ストリーム送信の繰り返し方法の設定、または問い合わせをします。

文法

```
:SOURce:STReam:DURation:TYPE CONT|FRAME|REPEAT
:SOURce:STReam:DURation:TYPE?
```

設定の種類を以下から選択します。

リモートコマンドではストリーム送信の繰り返し方法に **Time** を設定できません。

```
CONT:      Continuous
FRAME:      Frame
REPEAT:     Repeat (Frames)
```

FRAME と REPEAT は、同じ動作をします。

1 ストリームの場合はフレーム数を示します。複数ストリームの場合は、複数ストリームの繰り返し数を示します。

フレーム数、または繰り返し数は、次のどちらかのコマンドで設定します。

```
:SOURce:STReam:DURation:REPeat:COUNT
:SOURce:STReam:DURation:FRAMES
```

レスポンスデータ

```
CONT|FRAME|REPEAT
```

FRAME、または REPEAT を設定したとき、クエリのレスポンスはアプリケーションによって次のとおりになります。

```
100GbE, 40GbE: REPEAT
OTU4, OTU3: FRAME
```

画面の **Duration** に **Time** が設定されているときのレスポンスデータは、REPEAT になります。このとき、:SOURce:STReam:DURation:REPeat:COUNT、または:SOURce:STReam:DURation:FRAMESで繰り返し数の換算値を問い合わせできます。

使用例

ストリームを 1000 回繰り返し送信するときは以下のようにします。

```
:SOURce:STReam:DURation:TYPE REPEAT
:SOURce:STReam:DURation:REPeat:COUNT 1000
```

:SOURce:STReam:ENABle

機能

各ストリームの送信を設定します。
各ストリームの送信設定を問い合わせます。

文法

```
:SOURce:STReam:ENABle <NR1>[,<NR1>,...]  
:SOURce:STReam:ENABle?
```

<NR1>[,<NR1>,...]

有効にするストリームの番号を 1 つ以上設定します。

レスポンスデータ

<NR1>[,<NR1>,...]

使用例

ストリーム 5,12,15 を送信するときは以下のようにします。

```
:SOURce:STReam:ENABle 5,12,15  
:SOURce:STReam:ENABle?  
> 5,12,15
```

:SOURce:STReam:ERRor:TYPE

機能

イーサネットフレームに付加するエラーの種類を設定します。
イーサネットフレームに付加するエラーの種類を問い合わせます。
:SOURce:STReam:ID で指定したストリームにエラーが付加されます。

文法

```
:SOURce:STReam:ERRor:TYPE FCS|OFF  
:SOURce:STReam:ERRor:TYPE?
```

エラーの種類を以下から選択します。

```
OFF  None  
FCS  FCS Error
```

レスポンスデータ

FCS|OFF

使用例

送信するすべてのイーサネットフレームに FCS エラーを付加します。

```
:SOURce:STReam:ERRor:TYPE FCS  
:SOURce:STReam:ERRor:TYPE?  
>FCS
```


:SOURce:STReam:FSIZe**機能**

送信するイーサネットフレームのフレームサイズを問い合わせます。

文法

:SOURce:STReam:FSIZe?

レスポンスデータ

<NR1>[,<NR1>]

<NR1>: イーサネットフレームのフレームサイズ(byte 単位)
フレームサイズの Type が RANDOM に設定されている場合は、実際に送信されるフレームサイズの最小値と最大値が返ります。

:SOURce:STReam:FSIZe:RANGe**機能**

送信するイーサネットフレームのフレームサイズの可変範囲を設定します。
送信するイーサネットフレームのフレームサイズの可変範囲を問い合わせます。

文法

:SOURce:STReam:FSIZe:RANGe <value1>,<value2>
:SOURce:STReam:FSIZe:RANGe?

<value1>,<value2> = <NR1>

フレームサイズの Type が RANDOM に設定されている場合に、フレームサイズの最小値と最大値を byte 単位で設定します。2 つの値のうち、小さい方が最小値、大きい方が最大値となります。

レスポンスデータ

<value1>,<value2> = <NR1>

<value1>: フレームサイズの最小値

<value2>: フレームサイズの最大値

使用例

フレームサイズを 64 バイトから 1500 バイトの範囲でランダム変化させます。

:SOURce:STReam:FSIZe:TYPE RANDOM

:SOURce:STReam:FSIZe:RANGe 64,1500

:SOURCE:STReam:FSIZe:TYPE

機能

送信するイーサネットフレームのフレームサイズ長を一定にするか、可変にするかを設定します。

送信するイーサネットフレームのフレームサイズ長の設定が一定か、可変かを問い合わせます。

文法

:SOURCE:STReam:FSIZe:TYPE FIXED|RANDOM

:SOURCE:STReam:FSIZe:TYPE?

設定の種類を以下から選択します。

FIXED

RANDOM

レスポンスデータ

FIXED|RANDOM

FIXED を選択したときは、:SOURCE:STReam:FSIZe:VALueでフレームサイズを設定します。

RANDOM を選択したときは、:SOURCE:STReam:FSIZe:RANGeで送信するフレームサイズの範囲を設定します。

使用例

:SOURCE:STReam:FSIZe:RANGeの使用例を参照してください。

:SOURCE:STReam:FSIZe:VALue

機能

フレームサイズのフレームサイズが Fixed に設定されている場合に、送信するイーサネットフレームのフレームサイズを設定します。

イーサネットフレームのフレームサイズを問い合わせます。

パターン設定長*1がFrame Size 設定*2を超えたときは、パターン設定長のフレームサイズで送信されます。このときに実際に送信されるフレームサイズの（最小）値を返します。

*1: パターン設定長: Header Pattern 長 + (On のとき)Test Frame 最小長 (18) + FCS 長(4)

*2: Frame Size 設定が Random のときは設定の最小値で比較されます。

文法

```
:SOURCE:STReam:FSize:VALue <NR1>
```

```
:SOURCE:STReam:FSize:VALue?
```

<NR1>: FIXED を選択したときのフレームサイズ(byte 単位)

実際に送信されるフレームサイズは:SOURCE:STReam:FSizeで確認できます。

レスポンスデータ

```
<NR1>
```

使用例

512 バイトのフレームを送信します。

```
:SOURCE:STReam:FSize:TYPE FIXED
```

```
:SOURCE:STReam:FSize:VALue 512
```

:SOURCE:STReam:HEADer**機能**

ストリームのヘッダーフォーマットと値を設定し、以下のコマンドの設定状態を初期化します。

```
:SOURCE:STReam:HEADer:VARiable{3|4|5}
```

ストリームのヘッダーフォーマットを問い合わせます。

文法

```
:SOURCE:STReam:HEADer <format>,<header>
```

```
:SOURCE:STReam:HEADer?
```

<format>=<STRING>: ヘッダーフォーマット

<header>=<HEX>: ヘッダーの値

<format>には次の表の順番で、文字列をコンマで区切って記載します。

グループ番号の順番が小さい順でない場合、または同じ順番の文字列が複数存在する場合は、エラー (-220) になります。

ヘッダーパターン(最大 75 バイト)を 16 進数で設定します。

75 バイトには、MAC アドレス 12 バイト(送信元アドレスとあて先アドレス)と 63 バイトの EtherType を含むパターンが含まれます。

表3.3.2-4 ヘッダフォーマットを指定する文字列

順番	<format>	ヘッダ
1	NONE	Ethernet のみ *1
	CUSTOM	Custom Header *1
2	MPLS_TP{1 2 3 4 5}	MPLS-TP Control Word 有り*2
	MPLS_TP{1 2 3 4 5}_NOCW	MPLS-TP Control Word 無し*2
3	B_TAG	PBB (B-TAG+I-TAG)
	I_TAG	PBB (I-TAG)
4	VLAN{1 2}	VLAN *3
5	MPLS{1 2 3}	MPLS *2
6	IPv4	IPv4 *4,*5
	IPv6	IPv6 *4,*6
	ARP	ARP *5,*6
7	ICMPV4_ECHO	ICMPv4 Echo
	ICMPV6_ECHO	ICMPv6 Echo
	ICMPV6_NS	ICMPv6 NS
	ICMPV6_NA	ICMPv6 NA

*1: NONEとCUSTOMIZEを指定した場合は、順番2以降の文字列を設定できません。順番2以降の文字列を設定するとエラー（-220）になります。

*2: 数字はラベル数を設定します。

*3: 数字はタグ数を設定します。

*4: ARPと同時に設定できません。

*5: IPv6と同時に設定できません。

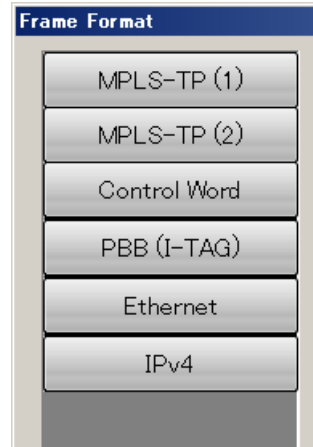
*6: IPv4と同時に設定できません。

レスポンスデータ

<STRING>,<HEX>

使用例

次の図に示すヘッダフォーマットを設定します。



```
:SOURce:STReam:HEADer
"MPLS_TP2,I_TAG,IPV4",&H00FF001000F000FF011000E088470001
008000010180000000000000050000003000000A0000008088E700000000
009230001F000092300010000800
:SOURce:STReam:HEADer?
>
"MPLS_TP2,I_TAG,IPV4",&H00FF001000F000FF011000E088470001
008000010180000000000000050000003000000A0000008088E700000000
009230001F00009230001000080045000002000040004011B99AC0A8
0000C0A80000
```

:SOURce:STReam:HEADer:ETHer:DA

機能

送信するイーサネットフレームの Destination MAC Address フィールドの値を設定/問い合わせます。

文法

```
:SOURce:STReam:HEADer:ETHer:DA <HEX>
:SOURce:STReam:HEADer:ETHer:DA?
```

<HEX>: Destination MAC Address (6 オクテットの 16 進数)

レスポンスデータ

<HEX>

使用例

:SOURce:STReam:HEADer:ETHer:SAの使用例を参照してください。

:SOURce:STReam:HEADer:ETHer:SA

機能

送信するイーサネットフレームの Source MAC Address フィールドの値を設定/問い合わせます。

文法

```
:SOURce:STReam:HEADer:ETHer:SA <HEX>
:SOURce:STReam:HEADer:ETHer:SA?
```

<HEX>: Source MAC Address (6 オクテットの 16 進数)

レスポンスデータ

<HEX>

使用例

Destination MAC Address フィールドを 00:00:00:00:00:01, Source MAC Address フィールドを 00:00:00:00:00:02 に設定します。

```
:SOURce:STReam:HEADer:ETHer:DA #H00000000000001
:SOURce:STReam:HEADer:ETHer:SA #H00000000000002
:SOURce:STReam:HEADer:ETHer:DA?
> #H00000000000001
:SOURce:STReam:HEADer:ETHer:SA?
> #H00000000000002
```

:SOURce:STReam:HEADer:ETHer:TYPE

機能

送信するイーサネットフレームの Type フィールドの値を設定します。

文法

```
:SOURce:STReam:HEADer:ETHer:TYPE <HEX>
:SOURce:STReam:HEADer:ETHer:TYPE?
```

<HEX>: Type フィールドの値 (2 オクテットの 16 進数)

使用例

Ethernet Type フィールドの値を 0x0800 (IPv4) に設定します。

```
:SOURce:STReam:HEADer:ETHer:TYPE #H0800
:SOURce:STReam:HEADer:ETHer:TYPE?
>#H0800
```

:SOURce:STReam:HEADer:PATtern**機能**

ストリームのヘッダーパターンを設定します。
ストリームのヘッダーパターンを問い合わせます。

文法

:SOURce:STReam:HEADer:PATtern <HEX>
:SOURce:STReam:HEADer:PATtern?

<HEX>:

ROUTe:BERT コマンドの設定が On の場合
ヘッダーパターン(MAC アドレス 12 バイト(送信元アドレスとあて先アドレス)と Ethernet Type フィールドの 2 バイト)を 16 進数で設定します。

ROUTe:BERT コマンドの設定が Off の場合
ヘッダーパターン(最大 257 バイト)を 16 進数で設定します。
2575 バイトには, MAC アドレス 12 バイト(送信元アドレスとあて先アドレス)と 2 バイトの EtherType フィールドが含まれます。

注:

ROUTe:BERT コマンドの設定が Off の場合に本コマンドでパターンを設定すると, :SOURce:STReam:HEADer で設定したフォーマットが CUSTOM に変更されます。

ROUTe:BERT コマンドの設定が Off の場合は,
:SOURce:STReam:HEADer でパターンを設定してください。

レスポンスデータ

<HEX>

使用例

Ethernet Type フィールドの値を 0x0800 (IPv4) に設定します。

```
:SOURce:STReam:HEADer:PATtern
#H89ABCDABCDEF12345634567888A861008100E0100800450303CE00
0040004017A199C0A80A14C0A80A18
:SOURce:STReam:HEADer:PATtern?
>#H89ABCDABCDEF12345634567888A861008100E0100800450303CE0
00040004017A199C0A80A14C0A80A18
```

:SOURce:STReam:HEADer:VARiable1:RANGe

機能

あて先 MAC アドレスの可変範囲を設定します。Stream Control/Header ダイアログの Modifiers タブの、フィールド 1 の可変範囲も同時に設定します。
あて先 MAC アドレスの可変範囲を問い合わせます。

文法

:SOURce:STReam:HEADer:VARiable1:RANGe
<offset>,<length>,<count>
:SOURce:STReam:HEADer:VARiable1:RANGe?

<offset>,<length>,<count>=<NR1>
<offset>: Ethernet フレーム先頭からのバイト数(0~5)
<length>: 変化させるバイト数(1~6)
<count>: 発生個数
<offset>+<length><7 となるようにします。
<count>に設定できる値は、<length>によって変わります。

<length>	<count>
1	1~256
2	1~65536
3	1~16777216
4	1~4294967296
5	1099511627776
6	281474976710656

<length>が 5, または 6 の場合に、表の数値以外の値を<count>に設定しても無視されます。

レスポンスデータ

<offset>,<length>,<count>

:SOURCE:STReam:HEADer:VARiable1:TYPE**機能**

あて先 MAC アドレスの可変方法(Off/Decrement/Increment/Random)を設定します。Stream Control/Header ダイアログの Modifiers タブの、フィールド 1 の変化方法も同時に設定します。

あて先 MAC アドレスの可変方法を問い合わせます。

文法

```
:SOURCE:STReam:HEADer:VARiable1:TYPE  
DECREMENT | INCREMENT | OFF | RANDOM  
:SOURCE:STReam:HEADer:VARiable1:TYPE?
```

タイプを次から選択します。

OFF OFF は Stream Control/Header 画面上の Fixed を示します。

DECREMENT

INCREMENT

RANDOM

レスポンスデータ

DECREMENT | INCREMENT | OFF | RANDOM

:SOURce:STReam:HEADer:VARiable2:RANGe

機能

送信元 MAC アドレスの可変範囲を設定します。Stream Control/Header ダイアログの Modifiers タブの、フィールド 2 の可変範囲も同時に設定します。
送信元 MAC アドレスの可変範囲を問い合わせます。

文法

:SOURce:STReam:HEADer:VARiable2:RANGe
<offset>,<length>,<count>
:SOURce:STReam:HEADer:VARiable2:RANGe?

<offset>,<length>,<count>=<NR1>
<offset>: Ethernet フレーム先頭からのバイト数(6～11)
<length>: 変化させるバイト数(1～6)
<count>: 発生回数(1～n)
<offset>+<length><13 となるようにします。
<count>に設定できる値は、<length>によって変わります。

<length>	<count>
1	1～256
2	1～65536
3	1～16777216
4	1～4294967296
5	1099511627776
6	281474976710656

<length>が 5, または 6 の場合に、表の数値以外の値を<count>に設定しても無視されます。
offset の設定範囲が画面とリモートで異なります。
Stream Control/Header 画面では offset の値は Source MAC アドレスの先頭からですが、リモートコマンドでは offset の値は Ethernet フレーム先頭からを示します。

レスポンスデータ

<offset>,<length>,<count>

:SOURCE:STReam:HEADer:VARiable2:TYPE**機能**

送信元 MAC アドレスの可変方法(Off/Decrement/Increment/Random)を設定します。Stream Control/Header ダイアログの Modifiers タブの、フィールド 2 の変化方法も同時に設定します。

送信元 MAC アドレスの可変方法を問い合わせます。

文法

```
:SOURCE:STReam:HEADer:VARiable1:TYPE
DECREMENT | INCREMENT | OFF | RANDOM
:SOURCE:STReam:HEADer:VARiable1:TYPE?
```

タイプを次から選択します。

OFF OFF は Stream Control/Header 画面上の Fixed を示します。

DECREMENT

INCREMENT

RANDOM

レスポンスデータ

```
DECREMENT | INCREMENT | OFF | RANDOM
```

:SOURCE:STReam:HEADer:VARiable{3|4|5}**機能**

Stream Control/Header ダイアログの Modifiers タブのフィールド 3, 4, 5 のパラメータを設定します。

Stream Control/Header ダイアログの Modifiers タブのフィールド 3, 4, 5 のパラメータを問い合わせます。

文法

```
:SOURCE:STReam:HEADer:VARiable{3|4|5}
<type>[, <field>, <offset>, <length>, <min>, <max>, <step>]
:SOURCE:STReam:HEADer:VARiable{3|4|5}?
```

<type>=<CHAR>

<type>には次の文字列を指定します。

OFF OFF は Stream Control/Header 画面上の Fixed を示します。

DECREMENT

INCREMENT

RANDOM

<type>に OFF を設定した場合、第 2 パラメータ以降を省略できます。また、第 2 パラメータ以降を設定しても無視されます。

<field>==CHAR>

< field >には次の表の文字列を指定します。

表3.3.2-5 フィールドを指定する文字列

< field >	フィールド名
TOP	Top of Frame *1
MPLS_TP_EXP{1 2 3 4 5}	MPLS-TP Tags Label
MPLS_TP_LABEL{1 2 3 4 5}	MPLS-TP Tags Exp
MPLS_TP_TTL{1 2 3 4 5}	MPLS-TP Tags TTL
CW_NIB	MPLS Control Word first nibble
CW_FLAG	MPLS Control Word flags
CW_FRG	MPLS Control Word FRG (Fragmentation)
CW_LENGTH	MPLS Control Word Length
CW_SEQ	MPLS Control Word Sequence Number
PBB_DA	PBB Destination MAC Address *2
PBB_SA	PBB Source MAC Address *2
PBB_{B I}_PCP	PBB Tags PCP
PBB_B_VID	PBB B-Tag VID
PBB_I_SID	PBB I-Tag SID
ETH_DA	Ethernet Destination MAC Address *3
ETH_SA	Ethernet Source MAC Address *3
ETH_TYPE	Ethernet Type
VLAN_{1 2}_PCP	VLAN Tags PCP
VLAN_{1 2}_VID	VLAN Tags VID
MPLS_EXP{1 2 3}	MPLS Tags Label
MPLS_LABEL{1 2 3}	MPLS Tags Exp
MPLS_TTL{1 2 3}	MPLS Tags TTL
IPV4_DA	IPv4 Destination Address
IPV4_SA	IPv4 Source Address
IPV4_TOS	IPv4 TOS
IPV4_TTL	IPv4 TTL
IPV4_PROT	IPv4 Protocol
IPV6_DA	IPv6 Destination Address
IPV6_SA	IPv6 Source Address
IPV6_TCCLASS	IPv6 Traffic Class
IPV6_FLABEL	IPv6 Flow Label
IPV6_HOP	IPv6 Hop Limit

表3.3.2-5 フィールドを指定する文字列 (続き)

< field >	フィールド名
ARP_SMAC	ARP Source MAC Address
ARP_SIP	ARP Source IP Address
ARP_TMAC	ARP Target MAC Address
ARP_TIP	ARP Target IP Address
ARP_OPE	ARP Operation
ICMPV4_CODE	ICMPv4 Code
ICMPV4_EC_ID	ICMPv4 Identifier
ICMPV4_EC_SEQ	ICMPv4 Sequence No.
ICMPV6_CODE	ICMPv6 Code
ICMPV6_EC_ID	ICMPv6 Identifier
ICMPV6_EC_SEQ	ICMPv6 Sequence No.
ICMPV6_NSNA_TADDR	ICMPv6 Target Address
ICMPV6_NSNA_SADDR	ICMPv6 Source Link Layer Address

*1: :SOURce:STReam:HEADerでCUSTOM以外に設定した場合に本コマンドでTOPを指定すると、エラー (-220) になります。

*2: ストリームのフォーマットに MPLS_TP と PBB が設定されている場合に指定できます。

*3: ストリームのフォーマットに MPLS_TP, または PBB が設定されている場合に指定できます。

Filed 3～5 の間で<field>に同じ文字列を設定できません。Filed 3～5 の間で<field>に同じ文字列を設定すると、エラー (-220) になります。

:SOURce:STReam:HEADerでストリームのフレームを変更すると、<type>の設定はOFFになります。

:SOURce:STReam:HEADerで設定したフレームフォーマットに存在しないフィールドを<format>に指定すると、エラー (-220) になります。

:SOURce:STReam:HEADerでストリームのフォーマットを Custom Header に設定すると、本コマンドの<format>の設定はTOPに変更されます。

:SOURce:STReam:HEADerでストリームのフォーマットを Custom Header 以外に設定すると、本コマンドの<format>の設定はETH_TYPEに変更されます。

<offset>,<length>,<min>,<max>,<step>=<NR1>

<offset>: 対象範囲の先頭からのビット数
設定できる値は、対象範囲のビット数によって変わります。

<length>: 変化させるビット数(1～32)
設定できる値は、対象範囲のビット数、<offset>の値によって

変わります。

<min>: フィールドの最小値
 <max>: フィールドの最大値
 <step>: フィールドの値の変化量

レスポンスデータ

<CHAR>, <CHAR>, <NR1>, <NR1>, <NR1>, <NR1>, <NR1>

<type>, <field>, <offset>, <length>, <min>, <max>, <step>の順番です。

使用例

Modifiers のフィールド 3 を, 次の画面のように設定します。

```
:SOURCE:STReam:HEADer:VARiable3
RANDOM,MPLS_TTL1,0,8,0,255,1
:SOURCE:STReam:HEADer:VARiable3?
> RANDOM,MPLS_TTL1,0,8,0,255,1
```

:SOURce:STReam:ID

機能

編集するストリームの ID を設定, または問い合わせます。

ストリームを編集する場合に事前に ID を設定します。

複数ユニット接続時, :SOURce:STReam:ID はユニットごとに設定する必要があります。

文法

:SOURce:STReam:ID <NR1>

:SOURce:STReam:ID?

<NR1>: ストリーム ID (1~16)

レスポンスデータ

<NR1>

使用例

ユニット番号 1,2,3 に対してストリーム No.2 を設定します。

(ユニット No.1 のストリーム設定)

:UENTry:ID 1

:SOURce:STReam:ID 2

(ユニット No.2 のストリーム設定)

:UENTry:ID 2

:SOURce:STReam:ID 2

(ユニット No.3 のストリーム設定)

:UENTry:ID 3

:SOURce:STReam:ID 2

:SOURce:STReam:RESolve:IP:TARGet

機能

ARP で問い合わせる IP アドレスの対象を設定, 問い合わせます。

文法

:SOURce:STReam:RESolve:IP:TARGet <CHAR>

:SOURce:STReam:RESolve:IP:TARGet?

<CHAR>=GATEWAY|IPADDRESS

GATEWAY: Gateway IP Address

IPADDRESS: Destination IP Address

レスポンスデータ

GATEWAY|IPADDRESS

使用例

:SOURce:STReam:RESolve:IP:TARGet IPADDRESS

:SOURce:STReam:RESolve:IP:TARGet?

> IPADDRESS

:SOURce:STReam:RESolve:IPV4:ROUTer

機能

ARP で問い合わせる Gateway の IPv4 アドレスを設定, 問い合わせます。

文法

:SOURce:STReam:RESolve:IPV4:ROUTer <HEX>

:SOURce:STReam:RESolve:IPV4:ROUTer?

<HEX>:Gateway の IPv4 アドレス (4 バイトの 16 進数)

使用例

Gateway の IP アドレスを 192.168.1.10 (0xC0A8010A) に設定します。

:SOURce:STReam:RESolve:IPV4:ROUTer #HC0A8010A

:SOURce:STReam:RESolve:IPV4:ROUTer?

>#HC0A8010A

:SOURce:STReam:RESolve:IPV6:ROUTer**機能**

NS で問い合わせる Gateway の IPv6 アドレスを設定, 問い合わせます。

文法

```
:SOURce:STReam:RESolve:IPV6:ROUTer <HEX>  
:SOURce:STReam:RESolve:IPV6:ROUTer?
```

<HEX>:Gateway の IPv6 アドレス (16 バイトの 16 進数)

使用例

Gateway の IP アドレスを 2001::1 に設定します。

```
:SOURce:STReam:RESolve:IPV6:ROUTer  
#H20010000000000000000000000000001  
:SOURce:STReam:RESolve:IPV6:ROUTer?  
>#H20010000000000000000000000000001
```

:SOURce:STReam:RESolve:MAC:RETRy

機能

MAC アドレス解決のリトライ回数を設定, または問い合わせます。

文法

:SOURce:STReam:RESolve:MAC:RETRy <NR1>

:SOURce:STReam:RESolve:MAC:RETRy?

<NR1>=1~100

レスポンスデータ

<NR1>

使用例

MAC アドレス解決のリトライ回数を 10 に設定します。

:SOURce:STReam:RESolve:MAC:RETRy 10

:SOURce:STReam:RESolve:MAC:RETRy?

> 10

:SOURce:STReam:RESolve:MAC:TIMEout

機能

MAC アドレス解決のタイムアウト時間を設定, または問い合わせます。

文法

:SOURce:STReam:RESolve:MAC:TIMEout <NR1>

:SOURce:STReam:RESolve:MAC:TIMEout?

<NR1>: タイムアウト時間 3~10 秒

レスポンスデータ

<NR1>

使用例

:SOURce:STReam:RESolve:MAC:TIMEout 10

:SOURce:STReam:RESolve:MAC:TIMEout?

> 10

:SOURce:STReam:RESolve:PING:TRY**機能**

Ping 実行回数を設定, または問い合わせます。

文法

```
:SOURce:STReam:RESolve:PING:TRY <NR1>
:SOURce:STReam:RESolve:PING:TRY?
```

<NR1>: Ping 実行回数 1~100

レスポンスデータ

<NR1>

使用例

```
:SOURce:STReam:RESolve:PING:TRY 20
:SOURce:STReam:RESolve:PING:TRY?
> 20
```

:SOURce:STReam:RESolve:PING:PAYLoad**機能**

Ping 実行時の Payload Type を設定, または問い合わせます。

文法

```
:SOURce:STReam:RESolve:PING:PAYLoad ALL0|ALL1|DEFAULT
:SOURce:STReam:RESolve:PING:PAYLoad?
```

ALL0: All 0
ALL1: All 1
ALTERNATE: 0/1 パターン

レスポンスデータ

ALL0|ALL1|ALTERNATE

使用例

```
:SOURce:STReam:RESolve:PING:PAYLoad ALL0
:SOURce:STReam:RESolve:PING:PAYLoad?
> ALL0
```

:SOURce:STReam:RESolve:PING:TIMEout

機能

Ping タイムアウト時間を設定, または問い合わせます。

文法

```
:SOURce:STReam:RESolve:PING:TIMEout <NR1>
:SOURce:STReam:RESolve:PING:TIMEout?
```

<NR1>: Ping タイムアウト時間 3～10 秒

レスポンスデータ

<NR1>

使用例

```
:SOURce:STReam:RESolve:PING:TIMEout 10
:SOURce:STReam:RESolve:PING:TIMEout?
> 10
```

:SOURce:STReam:RESolve:RESult

機能

MAC アドレス解決, または Ping 実行の結果を問い合わせます。

文法

```
:SOURce:STReam:RESolve:RESult? <CHAR>
```

<CHAR>=PING|RESOLVE

PING: Ping

RESOLVE: Resolve

レスポンスデータ

<NR1>,<NR1>,<NR1>,...<NR1>

<NR1>: MAC アドレス解決, または Ping 実行の結果

0: 成功

1: 対象外, または未実施

2: 失敗

MAC アドレス解決中, または Ping 実行中のレスポンスは 1 になります。

レスポンスデータは 16 個あり, Stream 1 から Stream 16 の結果です。

使用例

```
:SOURce:STReam:RESolve:RESult? RESOLVE
> 0,0,0,0,0,0,1,2,2,2,2,2,2,2,2,2
```

:SOURCE:STReam:RESolve:START**機能**

ストリームを指定して MAC アドレス解決, または Ping 実行を開始します。

文法

:SOURCE:STReam:RESolve:START <CHAR>

<CHAR>=ALL|IPV4ALL|IPV6ALL|S1|S2|S3|S4|S5|S6|S7|S8|S9|S10|S11|S12|S13|S14|S15|S16

ALL: すべてのストリーム

IPV4ALL: フレーム構成に IPv4 を含むすべてのストリーム

IPV6ALL: フレーム構成に IPv6 を含むすべてのストリーム

S1: Stream 1

S2: Stream 2

: :

S16: Stream 16

:SOURCE:STReam:RESolve:START:EVENT**機能**

MAC アドレス解決, または Ping 実行の開始イベントを問い合わせます。
開始イベントを読み取ると, 値が 0 になります。

文法

:SOURCE:STReam:RESolve:START:EVENT?

レスポンスデータ

0|1

0: MAC アドレス解決, Ping 実行が未実行

MAC アドレス解決, または Ping 実行が開始後にイベントを読み取り済み

1: MAC アドレス解決, または Ping 実行を開始した。

使用例

```
:SOURCE:STReam:RESolve:START ALL
```

```
:SOURCE:STReam:RESolve:START:EVENT?
```

```
> 1
```

```
:SOURCE:STReam:RESolve:START:EVENT?
```

```
> 0
```

:SOURce:STReam:RESolve:STATus

機能

MAC アドレス解決, または Ping 実行の状態を問い合わせます。

文法

:SOURce:STReam:RESolve:STATus?

レスポンスデータ

0|1|2

0: MAC アドレス解決, Ping 実行が終了

1: MAC アドレス解決, または Ping 実行が処理中

2: MAC アドレス解決, または Ping 実行が終了したが, 一部または全部のスト
リームで結果が失敗

使用例

:SOURce:STReam:RESolve:START ALL

:SOURce:STReam:RESolve:STATus?

> 1

:SOURce:STReam:RESolve:STOP

機能

MAC アドレス解決, または Ping 実行を中止します。

文法

:SOURce:STReam:RESolve:STOP

使用例

:SOURce:STReam:RESolve:STOP

:SOURce:STReam:RESolve:STATus?

> 2

:SOURce:STReam:RESolve:TYPE

機能

解決法の種類を設定, または問い合わせます。

文法

```
:SOURce:STReam:RESolve:TYPE <CHAR>
```

```
:SOURce:STReam:RESolve:TYPE?
```

```
<CHAR>=RES|RES_PING|PING
```

```
PING:                Ping Only
```

```
RES:                 Resolve Only
```

```
RES_PING:            Resolve & Ping
```

レスポンスデータ

```
PING|RES|RES_PING
```

使用例

```
:SOURce:STReam:RESolve:TYPE RES
```

```
:SOURce:STReam:RESolve:TYPE?
```

```
> RES
```

:SOURCE:STReam:STARt[:AUNit]

機能

ストリーム送信を開始します。

すでにストリームを送信している場合は再スタートとなります。

:AUNit を追加すると、接続しているすべてのユニットのストリーム送信を同時に開始します。

文法

:SOURCE:STReam:STARt[:AUNit]

ストリームに関するリモートコマンド (**:SOURCE:STReam:***) が送信された場合は、Stream 画面の Control Unit は Gap Size (byte) に変わります。

使用例

ストリーム送信の開始を待つ場合は以下のようにします。

1. **:SOURCE:STReam:STARt:EVENT?** を送信して、開始イベントをクリアします。
2. **:SOURCE:STReam:STARt** を送信して Stream 送信を開始します。
3. **:SOURCE:STReam:STARt:EVENT?** を、レスポンスデータが 1 (開始) になるまで繰り返し送信します。

:SOURCE:STReam:STARt:EVENT[:AUNit]

機能

ストリーム送信の開始イベントを問い合わせます。

ストリーム送信が開始すると 1 となります。

文法

:SOURCE:STReam:STARt:EVENT[:AUNit]?

レスポンスデータ

0|1

0: ストリーム送信が開始前

1: ストリーム送信が開始済み 本コマンドで読み取った後は 0 にクリアされます。

ヘッダーに:AUNit を付けた場合は、接続しているすべてのユニットのストリーム送信が開始すると 1 となります。

使用例

:SOURCE:STReam:STARt[:AUNit]の使用例を参照してください。

:SOURCE:STReam:STATUs[:AUNit]**機能**

ストリーム送信の動作状態を問い合わせます。

ヘッダーに:AUNit を付けた場合は接続したどれかのユニットの開始で 1, すべてのユニットの停止で 0 となります。

文法

:SOURCE:STReam:STATUs[:AUNit]?

レスポンスデータ

0|1

0: ストリーム送信停止中

1: ストリーム送信中

ヘッダーに:AUNitを付けた場合は接続したユニットのどれか1つ以上がストリーム送信を開始すると 1, すべてのユニットがストリーム送信を停止すると 0 になります。

使用例

:SOURCE:STReam:STOP[:AUNit]の使用例を参照してください。

:SOURCE:STReam:STOP[:AUNit]**機能**

ストリーム送信を停止します。

:AUNit を追加すると, 接続しているすべてのユニットのストリーム送信を停止します。

文法

:SOURCE:STReam:STOP[:AUNit]

使用例

ストリーム送信の停止を待つ場合は次のようにします。

1. :SOURCE:STReam:STOP を送信してストリーム送信を停止します。
2. :SOURCE:STReam:STATUs? をレスポンスデータが 0 (停止) になるまで繰り返し送信します。

:SOURce:STReam:TFRame:ENABle**機能**

:SOURce:STReam:ID で指定したストリームのテストフレームの有効/無効を設定します。

:SOURce:STReam:ID で指定したストリームのテストフレームの有効/無効を問い合わせます。

文法

:SOURce:STReam:TFRame:ENABle 0|1|OFF|ON

:SOURce:STReam:TFRame:ENABle?

0|OFF: テストフレーム無効

1|ON: テストフレーム有効

レスポンスデータ

0|1

使用例

テストフレームを有効に設定します。

:SOURce:STReam:TFRame:ENABle ON

:SOURce:STReam:TFRame:ENABle?

> 1

:SOURce:STReam:TFRame:FID**機能**

編集中のストリームに対するテストフレームのフローIDを設定します。

テストフレームのフローID 設定を問い合わせます。

テストフレームを設定するストリームは、:SOURce:STReam:IDで設定します。

文法

:SOURce:STReam:TFRame:FID <NR1>

:SOURce:STReam:TFRame:FID?

<NR1>: Flow ID(0～15)

レスポンスデータ

<NR1>

使用例

ストリーム No.5 のフローID を 12 に設定します。

:SOURce:STReam:ID 5

:SOURce:STReam:TFRame:FID 12

:SOURce:STReam:TFRame:FID?

> 12

:SOURce:STReam:TYPE**機能**

ストリームの送信方法を設定します。
ストリームの送信方法を問い合わせます。

文法

```
:SOURce:STReam:TYPE RANDOM|SEQUENTIAL
:SOURce:STReam:TYPE?
```

SEQUENTIAL: 送信対象のストリームを番号の若い順に繰り返し送信します。
RANDOM: 送信対象のストリームをランダムに送信します。

レスポンスデータ

```
RANDOM|SEQUENTIAL
```

:SOURce:TPATtern:INVert**機能**

PRBS 試験パターン (送信) の反転設定をします。
PRBS 試験パターン (送信) の反転設定を問い合わせます。
受信側は:SENSe:TPATtern:INVertで反転設定をします。

文法

```
:SOURce:TPATtern:INVert 0|1|OFF|ON
:SOURce:TPATtern:INVert?
```

0|OFF: パターン反転 Off
1|ON: パターン反転 On

レスポンスデータ

```
0|1
```

使用例

:SENSe:TPATtern:INVert の使用例を参照してください。

:SOURce:TPATtern:TYPE

機能

送信側(Tx)のテストパターンの種類を設定します。

送信側(Tx)のテストパターンの種類を問い合わせます。

文法

:SOURce:TPATtern:TYPE <CHAR>

:SOURce:TPATtern:TYPE?

<CHAR>

試験パターンのタイプを以下から選択します。

PRBS7
PRBS9
PRBS11
PRBS15
PRBS23
PRBS31
ALL0
ALL1
WORD16
SQUARE : Square Wave

Mapping によって、設定できる試験パターンが変わります。

Mapping	試験パターン
40GbE, 100GbE, OTU4 Ethernet Client	PRBS31, ALL0, ALL1, WORD16
OTU3-PRBS, OTU4-PRBS	PRBS11, PRBS15, PRBS23, PRBS31, WORD16
OTU4-ODTU4.8-ODU2e-PRBS, OTU4-ODTU4.1-ODU0-PRBS	PRBS31, Word16
100GbE No Frame, 40GbE No Frame, OTU4 No Frame, OTU3 No Frame,	PRBS7, PRBS9, PRBS11, PRBS15, PRBS23, PRBS31, SQUARE

レスポンスデータ

ALL0 | ALL1 | PRBS7 | PRBS9 | PRBS11 | PRBS15 | PRBS23 | PRBS31 |
SQUARE | WORD16

使用例

:SOURce:TPATtern:TYPE PRBS31

:SOURce:TPATtern:TYPE?

>PRBS31

:SOURce:TPATtern:WORD

機能
送信側(Tx)のテストパターンに Word16 を選択した場合のパターンを設定します。
送信側(Tx)の Word16 パターンを問い合わせます。

文法
:SOURce:TPATtern:WORD <HEX>
:SOURce:TPATtern:WORD?

<HEX>: Data Field の値 (2 byte)

レスポンスデータ
<HEX>

使用例
:SOURce:TPATtern:TYPE WORD16
:SOURce:TPATtern:WORD #H55AA
:SOURce:TPATtern:WORD?
>#H55AA

:SOURce:TRANsceiver:EMPHasis:FIRSt

機能
トランシーバの Pre-Emphasis (First)を設定します。
トランシーバの Pre-Emphasis (First)設定を問い合わせます。

文法
:SOURce:TRANsceiver:EMPHasis:FIRSt <lane>,<value>
:SOURce:TRANsceiver:EMPHasis:FIRSt? <lane>

<lane>,<value>=<NR1>
<lane>: レーン番号

アプリケーション	<lane>の範囲
40GbE,OTU3	0～3
100GbE,OTU4	0～19
40GbE No Frame, OTU3 No Frame	0～3
100GbE No Frame, OTU4 No Frame	0～19

<value>: 設定値(0～31)

レスポンスデータ
<NR1>

:SOURce:TRANsceiver:EMPHasis:PRE**機能**

トランシーバの Pre-Emphasis (Pre)を設定します。
トランシーバの Pre-Emphasis (Pre)設定を問い合わせます。

文法

```
:SOURce:TRANsceiver:EMPHasis:PRE <lane>,<value>  
:SOURce:TRANsceiver:EMPHasis:PRE? <lane>
```

<lane>,<value>=<NR1>

<lane>: レーン番号
 レーン番号の設定範囲は、:SOURce:TRANsceiver
 :EMPHasis:FIRSt の説明を参照してください。

<value>: 設定値(-15～+15)

レスポンスデータ

<NR1>

:SOURce:TRANsceiver:EMPHasis:SECond**機能**

トランシーバの Pre-Emphasis (Second)を設定します。
トランシーバの Pre-Emphasis (Second)設定を問い合わせます。

文法

```
:SOURce:TRANsceiver:EMPHasis:SECond <lane>,<value>  
:SOURce:TRANsceiver:EMPHasis:SECond? <lane>
```

<lane>,<value>=<NR1>

<lane>: レーン番号
 レーン番号の設定範囲は、:SOURce:TRANsceiver
 :EMPHasis:FIRSt の説明を参照してください。

<value>: 設定値(-15～+15)

レスポンスデータ

<NR1>

:SOURce:TRANsceiver:VOD**機能**

トランシーバの VOD を設定します。
トランシーバの VOD 設定を問い合わせます。

文法

```
:SOURce:TRANsceiver:VOD <lane>,<value>
:SOURce:TRANsceiver:VOD? <lane>
<lane>,<value>=<NR1>
<lane>:          レーン番号
                  レーン番号の設定範囲は、:SOURce:TRANsceiver
                  :EMPHasis:FIRSt の説明を参照してください。
<value>:         設定値(0～6)
```

レスポンスデータ

```
<NR1>
```

:SYSTem:CONFig**機能**

本器のアプリケーションを切り替えます。
 このコマンドを実行するとアプリケーションが再起動します。
 再起動が完了したかどうかは、:SYSTem:STATus? コマンドで確認できます。
 再起動中は :SYSTem:ERRor? と :SYSTem:STATus? 以外のコマンド以外は使用できません。

文法

```
:SYSTem:CONFig <CHAR>[,<CHAR>[,<CHAR>...[,<CHAR>]...]]
```

<CHAR>

<CHAR> には Unit ID 1,2,...,16 のアプリケーション/マッピングを指定します。
 接続しないユニット ID は NONE を指定します。ただし、後に続く NONE については省略できます。

<CHAR>には、次のどれかを指定します。

アプリケーション/マッピング	<CHAR>
なし	NONE
40GbE	E40G
100GbE	E100G
ODU3-PRBS	OTU3
ODU4-PRBS	OTU4
ODU4-100GbE	OTU4_E
ODTU4.8-ODU2e-PRBS	OTU4_ODU2E
ODTU4.8-ODU2e-10GbE	OTU4_ODU2E_E
ODTU4.1-ODU0-PRBS	OTU4_ODU0
ODTU4.1-ODU0-GbE	OTU4_ODU0_E
40GbE No Frame	E40G_N
100GbE No Frame	E100G_N
OTU3 No Frame	OTU3_N
OTU4 No Frame	OTU4_N

使用例

Unit ID 1 に 40GbE, Unit ID 3 に 100GbE を指定する場合

```
:SYSTem:CONFig E40G,NONE,E100G
```

Unit ID4 に ODU4-100GbE を指定する場合

```
:SYSTem:CONFig NONE,NONE,NONE,OTU4_E
```

起 動しているアプリケーション・マッピングの問 い 合 わ せ に
は, :SOURce:MAPPingを使用します。

```
:UENTry:ID 1
:SOURce:MAPPing?
> OTU4
:UENTry:ID 2
:SOURce:MAPPing?
> OTU4
```

Unit ID1 から Unit ID3 までのアプリケーション/マッピングを切り替えます。

```
:SYSTem:CONFig OTU4_E,OTU4_ODU2E_E
```

```
:SYSTem:ERRor?
> 0,"No error"
```

:SYSTem:STATus?で再起動が完了したことを確認します。

```
:SYSTem:STATus?
> 0
:
:
:SYSTem:STATus?
> 1
```

再起動が完了すると, ほかのリモートコマンドで操作できます。

```
:UENTry:ID 1
:SOURce:MAPPing?
> OTU4_E
:UENTry:ID 2
:SOURce:MAPPing?
> OTU4_ODU2E_E
```

100GbE No Frame, OTU4 No Frame アプリケーションのレーン数設定に
は, :SOURce:MAPPingを使用します。

:SYSTem:DATE

機能

日付を問い合わせます。
時刻は :SYSTem:TIME? で問い合わせます。

文法

:SYSTem:DATE?

レスポンスデータ

<year>, <month>, <day> = <NR1>
年・月・日の値をコンマ区切りで返します。

使用例

```
:SYSTem:DATE?;:SYSTem:TIME?  
> 2010,10,7;12,40,3  
(2010 年 10 月 7 日 12 時 40 分 3 秒)
```

:SYSTem:ERRor

機能

エラー・イベントキューに格納されているエラー情報を問い合わせます。
1 回につき、イベントキューから 1 個の情報を読み取ります。

文法

:SYSTem:ERRor?

レスポンスデータ

<error_event_number>,"<error_event_description>"

<error_event_number> = <NR1>

エラーコードの範囲は、-32768 ～ +32767 です。

0 のときは、エラーが無い、またはイベントが発生していないことを示します。

<error_event_description> = <STRING>

<error_event_number>について 1 対 1 に対応する、エラーメッセージ です (最大 255 文字)。

エラー・イベントキューには最大 128 個のエラーイベントが保持されます。
エラー・イベントキューがいっぱいになっている場合には、それ以上キューにためられないため、発生したエラー情報は内部で破棄されます (エラー・イベントキューにたまっているエラー情報は破棄せず、最新のエラー情報が破棄されます)。
このとき、128 個目のキューには「-350,"Queue overflow」が格納されます。

使用例

そのアプリケーションで対応していない機能のコマンドを送ったときは、「-113,"Undefined header"」となります。

例: 100GbE アプリケーションでは対応していないコマンドを送った場合

```
:CALCulate:MONitor:OTU:COLumn 1
:SYSTem:ERRor?
> -113,"Undefined header"
```

そのアプリケーションでは設定できないパラメータを設定した場合は、「-220,"Parameter error"」となります。

例: 100GbE アプリケーションでは対応していないパラメータを設定した場合

```
:SOURce:EALarm:TYPE FAS_MLD
:SYSTem:ERRor?
>-220,"Parameter error"
```

そのアプリケーションの状態では実行できないパラメータを設定した場合は、「-310,"System error"」となります。

例: :MMEMory:RECall でロードできない・存在しないファイルを設定したとき

```
:MMEMory:RECall "nonexistent_file"
:SYSTem:ERRor?
> -310,"System error"
```

このように、コマンドを送った直後に:SYSTem:ERRor?を送信すると、エラーを確認できます。「2.6 メッセージの実行を確認する」や「付録B エラーコード」も参照してください。

また、エラーはキューにたまるため、キューが空になったことをつど確認していないと、以下のように直前のコマンドのエラーを取得できない場合があります。

```
:SOURce:EALarm:TYPE FAS_MLD
:MMEMory:RECall "nonexistent_file"
:SYSTem:ERRor?
> -220,"Parameter error" ← :SOURce:EALarm:TYPE のエラー
:SYSTem:ERRor?
> -310,"System error" ← :MMEMory:RECall のエラー
:SYSTem:ERRor?
> 0,"No error" ← キューが空になると「No error」となる
:SYSTem:ERRor?
> 0,"No error"
```

:SYSTem:PRINT:COPY**機能**

スクリーンコピーを実行します。

文法

:SYSTem:PRINT:COPY

現在の表示画面を PNG 形式画像ファイルに出力します。

スクリーンコピーの画像ファイルは以下の場所に作成されます。

C:\Documents and Settings\Administrator

\My Documents\Anritsu\MD1260A\UserData\Screen Copy

:SYSTem:STATus**機能**

アプリケーションが起動しているかどうかを問い合わせます。

:SYSTem:CONFigで、アプリケーションを設定した後に使用します。

文法

:SYSTem:STATus?

レスポンスデータ

0|1

0: 停止, または起動中

1: 起動済み

:SYSTem:TERMination**機能**

レスポンスデータのターミネータの種類を設定します。

レスポンスデータのターミネータの種類を問い合わせます。

文法

:SYSTem:TERMination 0|1

:SYSTem:TERMination?

ターミネータの種類を表します。

0: LF

1: CR+LF

レスポンスデータ

0|1

本器起動時の状態は 0 (LF) です。本設定はバックアップされないため、起動時は必ず 0 (LF) となります。また、*RSTコマンドにより本設定値が変わることはありません。

GPIB オプションの場合は、EOI (GPIB バスラインの信号) も同時に発行されます。

使用例

「2.4 接続を確認する」のイーサネットの場合を参照してください。

:SYSTem:TIME

機能

時刻を問い合わせます。日付は:SYSTem:DATE?で問い合わせます。

文法

:SYSTem:TIME?

レスポンスデータ

<hour>,<min>,<sec> = <NR1>

時・分・秒の値をコンマ区切りで返します。

使用例

:SYSTem:DATEを参照してください。

:SYSTem:VERSion

機能

本器が参照している SCPI のバージョンを問い合わせます。

文法

:SYSTem:VERSion?

レスポンスデータ

1999.0

SCPI のバージョン 1999.0 が返ります。(year: 1999, revision number : 0)

:UENTry:ID**機能**

操作対象のユニット ID を設定, または問い合わせます。

文法

:UENTry:ID <NR1>

:UENTry:ID?

<NR1>: Unit 1 から Unit 16 までのユニット ID(1~16)

レスポンスデータ

<NR1>

範囲外, または接続されていない Unit ID を設定したときに -220 エラーとなります。

使用例

操作対象のユニットを ID 16 のユニットに設定します。

:UENTry:ID 16

:UENTry:ID?

> 16

:UENTry:LIST**機能**

操作可能なユニット ID を問い合わせます。

文法

:UENTry:LIST?

レスポンスデータ

<num of unit>, <unit id> [, <unit id>, <unit id>,....]= <NR1>

<num of unit>: 操作可能なユニットの数

<unit id>: ユニット ID の ID 値

操作可能なユニットが複数ある場合, ユニット ID の順番は次のとおりです。

マスタユニットの ID

スレーブユニットの ID (値の小さい順)

使用例

:UENTry:LIST?

> 3,1,2,16

ここでは、リモート制御プログラムの作成を支援する本器の機能について説明します。

4.1	操作記録機能の概要	4-2
4.2	操作記録機能の開始と停止	4-3

4.1 操作記録機能の概要

操作記録機能は、本器のリモート制御プログラムの作成を支援する機能です。操作記録機能を実行中に本器の画面でパラメータなどを設定すると、画面操作と同じ設定をするリモート制御コマンドがテキストファイルに保存されます。操作記録機能では、設定内容や測定結果を問い合わせるクエリコマンドもテキストファイルに保存できます。

```
*** Operation Log Start
:UENTry:ID 1
*** Test Pattern Setting
:SOURce:STReam:HEADer:ETHer:SA
#H0123456789AB
:SOURce:STReam:HEADer:ETHer:TYPE #H1234
:SOURce:STReam:ERRor:TYPE FCS
:SOURce:STReam:FSIZe:TYPE FIXED
:SOURce:STReam:FSIZe:VALue 60
:SOURce:STReam:FSIZe:RANGe 60,16376

*** Operation Log End
```

図4.1-1 操作記録機能で作成されたファイル

測定結果のファイルは次のフォルダに保存されます。

C:\Documents and Settings\Administrator\My Documents
\Anritsu\MD1260A\UserData\Operation Log

ファイル名は次のとおりになります。

‘OperationLog’+‘_’+ 年月日 + ‘T’+時分秒 + ミリ秒

2010年10月7日 16時40分55.324秒に保存したファイルの名前は、次のようになります。

OperationLog_20101007T164055324.txt

制限事項

操作記録機能には、次の制限があります。

- ・ OK ボタンがない設定ダイアログの操作は記録されません。
例:MDIO 画面の [Write], [Read]
- ・ 対応するリモート制御コマンドが存在しないパネル操作は記録されません。
「3.2.7 リモートできないパネル操作」を参照してください。

4.2 操作記録機能の開始と停止

操作記録機能の開始

アプリケーションを起動した後、次の手順で操作記録機能を開始します。

1. [System Menu]をタッチします。
2. [Operation REC]をタッチします。Operation Record Panel が開きます。
この時点から、画面操作に対応するリモートコマンドの記録が始まります。
ただし、Operation Record Panel に対する操作、対応するリモートコマンドが存在しない操作は、ファイルに記録されません。
3. System Menu, 設定エリア, 操作エリア, またはトップメニューのボタンをタッチして本器の操作をすると、操作に対応するリモートコマンドがファイルに保存されます。
4. Operation Record Panel が隠れた場合は、[System Menu] の [Operation REC]をタッチします。Operation Record Panel が表示されます。
5. 画面操作に対するコメントを記録する場合は、Operation Record Panel の Comment のテキストボックスをタッチして、文字を入力します。[Insert]をタッチすると、ファイルに保存されます。

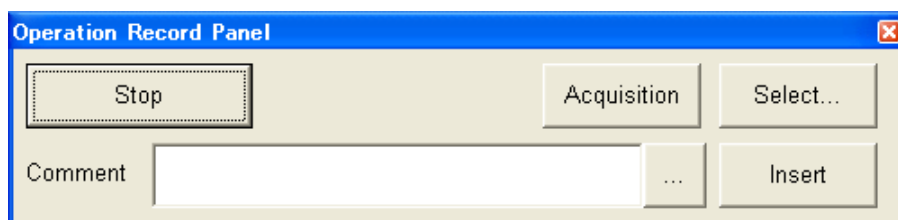


図4.2-1 Operation Record Panel

測定結果を取得するクエリコマンドをファイルに保存する場合は、次の操作をします。

1. Operation Record Panel の[Select]をタッチします。
Select 画面が開きます。
2. 読み取る測定項目をタッチします。
3. [OK]をタッチします。
4. Operation Record Panel の[Acquisition]をタッチします。
測定項目を読み取るリモートコマンドがファイルに保存されます。

操作記録機能の終了

Operation Record Panel の[Stop]をタッチします。

または、Operation Record Panel の閉じるボタンをタッチします。

ここでは、画面上に表示される測定項目とリモートコマンドとの対応について説明します。

パラメータの表記方法は、「表 3.1-1 コマンドの記述方法」を参照してください。

「A.3 40/100GbE アプリケーション」, 「A.4 OTU3/OTU4 アプリケーション」, および「A.5 No Frame アプリケーション」の表の「ID」は、:CALCulate:DATAと:MMEMory:LOG:ITEM コマンドのパラメータで指定できる文字列を示します。

A.1 レスポンスデータの形式

以下の表の「レスポンスデータ形式」は、:CALCulate:DATA?クエリに対するレスポンスデータの形式を示します。ここに記載されているレスポンスの形式は以下のいずれかとなります。

表A.1-1 レスポンスデータ形式

形式	説明
<status>	0 1 2 操作画面上に表示されるステータス LED の色に対応した値を返します。 0 正常 (緑) 1 異常 (赤) 2 ヒストリ (オレンジ)
<NR1>	整数値を表します。 例 123, -500
<NR2>	固定小数点形式の数値を表します。 例 12.345, -500.0
<NR3>	浮動小数点形式の数値を表します。 例 0.00E-11, 3.05E-11

いずれの測定項目も、測定値が無効の場合は画面表示と同様の文字列「-----」を返します。また、範囲外の値を示す場合は「>200.0」「<-200.0」のように「>(最大値)」「<(最小値)」という形式の文字列を返します。

測定項目が Lane 単位で複数ある場合は、Lane 数分の測定データを「,」で区切って返します。

注:

:CALCulate:DATA でカウンタ値を取得する前に、
:CALCulate:DATA:TYPE で取得するカウンタデータの種類 を Current, または Accumulated に設定してください。

A.2 共通項目

表A.2-1 測定項目一覧

測定項目	コマンド
Stream 状態表示 (LED)	:SOURce:STReam:STATus?
Error/Alarm Insertion 状態表示 (LED)	:SOURce:EALarm:STATus?
Counter 状態表示 (LED)	:CALCulate:COUNter:STATus?
Error/Alarm 状態表示 (LED)	:CALCulate:EALARM?
Log 状態表示 (LED)	:MMEMory:LOG:STATus?

レスポンスデータの書式は、コマンドの説明を参照してください。

A.3 40/100GbE アプリケーション

表A.3-1 測定項目とID 一覧

タブ	測定項目	ID	レスポンスデータ形式
(共通)	Link Status	LINK	<status>
	Elapsed Time (s)	ELAPSED	<NR1>
	Transmit Duration (ns)	TX_TIME	<NR2>
Test Frame	Tx Test Frame	TX_TFRAME	<NR1>
	Rx Test Frame	RX_TFRAME	
	Tx Rate (bit/s)	TX_TBPS	<NR1>
	Rx Rate (bit/s)	RX_TBPS	
	Sequence Error	RX_SEQ_ERR	<NR1>
	Current Latency (μs) *1	LAT_CURR	<NR2>
	Maximum Latency (μs) *2	LAT_MAX	<NR2>
	Minimum Latency (μs) *2	LAT_MIN	<NR2>
	Tx Test Frame (Other)	TX_TFRAME_O	<NR1>
	Rx Test Frame (Other)	RX_TFRAME_O	
	Tx Rate (bit/s) (Other)	TX_TBPS_O	<NR1>
	Rx Rate (bit/s) (Other)	RX_TBPS_O	
Distribution	Tx Frame Size Distribution	TX_FSIZE	<NR1>
	Rx Frame Size Distribution	RX_FSIZE	
	Tx Gap Size Distribution	TX_GSIZE	<NR1>
	Rx Gap Size Distribution	RX_GSIZE	

*1: LAT_CURR の値を取得する前に、:CALCulate:DATA:TYPE CURRENT を送信してください。

*2: LAT_MAX, LAT_MIN の値を取得する前に、
:CALCulate:DATA:TYPE ACCUM を送信してください

表A.3-1 測定項目とID 一覧 (続き)

タブ	測定項目	ID	レスポンスデータ形式
All Lanes	Tx Clock Status	CSLOSS	<status>
	Rx Clock Status	CUNLOCK	<status>
	Tx Frequency (Hz)	TX_FREQ	<NR1>
	Rx Frequency (Hz)	RX_FREQ	
	Tx Difference (ppm)	TX_FREQ_D	<NR1>
	Rx Difference (ppm)	RX_FREQ_D	
	Tx LF	TX_LF	<NR1>
	Rx LF	RX_LF	
	Tx RF	TX_RF	<NR1>
	Rx RF	RX_RF	
	Error Signals	RX_ERR_SIG	<NR1>
	Tx Errored Bytes	TEBYTE	<NR1>
	Rx Errored Bytes	REBYTE	
	Tx Good Bytes	TGBYTE	<NR1>
	Rx Good Bytes	RGBYTE	
	Tx FCS Errors	TFCS	<NR1>
	Rx FCS Errors	RFCS	
	Tx Fragments	TFRAGMENT	<NR1>
	Rx Fragments	RFRAGMENT	
	Tx Oversize & FCS Errors	TOAF	<NR1>
	Rx Oversize & FCS Errors	ROAF	
	Tx Undersize	TUNDERSIZE	<NR1>
	Rx Undersize	RUNDERSIZE	
	Tx Oversize	TOVERSIZE	<NR1>
	Rx Oversize	ROVERSIZE	
	Tx Good Frames	TGFRAME	<NR1>
	Rx Good Frames	RGFRAME	
	Tx Rate (bit/s)	TBPS	<NR2>
	Rx Rate (bit/s)	RBPS	
	Tx Rate (%)	TRATE	<NR2>
	Rx Rate (%)	RRATE	
	Pause Frame	RX_PAUSE	<NR1>
	Trigger Condition	RX_TRIGGER	<NR1>

表A.3-1 測定項目とID 一覧 (続き)

タブ	測定項目	ID	レスポンスデータ形式
All Lanes (続き)	Tx Broadcast	TBCAST_FRAME	<NR1>
	Rx Broadcast	RBCAST_FRAME	
	Tx Broadcast Bytes	TBCAST_BYTE	<NR1>
	Rx Broadcast Bytes	RBCAST_BYTE	
	Tx Multicast Frames	TMCAST_FRAME	<NR1>
	Rx Multicast Frames	RMCAST_FRAME	
	Tx Multicast Bytes	TMCAST_BYTE	<NR1>
	Rx Multicast Bytes	RMCAST_BYTE	
	Tx MPLS-TP	TMPLSTP	<NR1>
	Rx MPLS-TP	RMPLSTP	
	Tx PBB	TPBB	<NR1>
	Rx PBB	RPBB	
	Tx ARP Request	TARP_REQ	<NR1>
	Rx ARP Request	RARP_REQ	
	Tx ARP Reply	TARP_REP	<NR1>
	Rx ARP Reply	RARP_REP	
	Tx PINGv4 Request	TPINGV4_REQ	<NR1>
	Rx PINGv4 Request	RPINGV4_REQ	
	Tx PINGv4 Reply	TPINGV4_REP	<NR1>
	Rx PINGv4 Reply	RPINGV4_REP	
	Tx NDP (NS)	TNDP_NS	<NR1>
	Rx NDP (NS)	RNDP_NS	
	Tx NDP (NA)	TNDP_NA	<NR1>
	Rx NDP (NA)	RNDP_NA	
	Tx PINGv6 Request	TPINGV6_REQ	<NR1>
	Rx PINGv6 Request	RPINGV6_REQ	
	Tx PINGv6 Reply	TPINGV6_REP	<NR1>
	Rx PINGv6 Reply	RPINGV6_REP	
	Bit Errors (bit)	BER_CNT	<NR1>
	Bit Errors (rate)	BER_RATE	<NR3>
	Pattern Sync Loss	PSLOSS	<NR2>

表A.3-1 測定項目とID 一覧 (続き)

タブ	測定項目	ID	レスポンスデータ形式
Individual	Alignment Status	ALIGNMENT	<status>
	High-BER	HIBER	<status>
	Invalid Block (Count) (Rate)	IBLOCK	<NR1>
		IBLOCK_RATE	<NR3>
	Sync Header Lock	SHLOCK	<status>
	Alignment Marker Lock	AMLOCK	<status>
	Skew Stability	SSTAB	<status>
	Marker Map	MMAP	<NR1>
	Relative Skew (ns)	RSKEW	<NR2>
	Invalid Sync Header (Count) (Rate)	ISH	<NR1>
		ISH_RATE	<NR3>
	Invalid Alignment Marker (Count) (Rate)	IAM	<NR1>
		IAM_RATE	<NR3>
	BIP Error (bit) (Count) (Rate)	BIP_CNT	<NR1>
		BIP_RATE	<NR3>
Opt	LOS	LOS	<status>
	Programmable Alarm 1	PALARM1	<status>
	Programmable Alarm 2	PALARM2	
	Programmable Alarm 3	PALARM3	
	Global Alarm	GALARM	<status>
	Optical Power (dBm)	OPOWER	<NR1>

A.4 OTU3/OTU4 アプリケーション

マッピングにより、測定項目が異なります。

フレーム数と時間の測定結果がある場合、秒単位の測定結果を読み取るには、パラメータに_Sを追加します。

例: PM-TIM が発生したフレーム数を読み取る場合 PM_TIM

PM-TIM の発生時間(秒単位)で読み取る場合 PM_TIM_S

エラー数と発生率の測定結果がある場合、発生率の測定結果を読み取るには、パラメータに_Rを追加します。

例: PM-BIP8 の発生数を読み取る場合 PM_BIP8

PM-BIP8 の発生率を読み取る場合 PM_BIP8_R

ハイオーダー(OTU4)とローオーダー(ODU2e, ODU0)で同じ測定項目がある場合に、ローオーダーの測定結果を読み取る場合は、パラメータに_Lを追加します。

例: OTU4 の PM-BIP8 発生数を読み取る場合 PM_BIP8

ODU2e, または ODU0 の PM-BIP8 発生率を読み取る場合 PM_BIP8_L

表A.4-1 測定項目とID

タブ	測定項目	ID	レスポンスデータ形式
(共通)	Link Status	LINK	<status>
	Elapsed Time (second)	ELAPSED	<NR1>
Summary	Clock Source Loss	CSLOSS	<status>
	CDR Unlock	CUNLOCK	<status>
	Rx Frequency (Hz)	RX_FREQ	<NR1>
	Rx Frequency Difference (ppm)	RX_FREQ_D	<NR1>
	LOS	LOS	<status>
Statistics			
LLD	ILA/OLA	ILA	<status>
	Skew Stability	SSTAB	<status>
	Marker Map	MMAP	<NR1>
	Relative Skew (ns)	RSKEW	<NR2>
	LOF Lane (ms)	LOF_LLD	<NR2>
	OOF (frame)	OOF_LLD	<NR1>
	LOR (ms)	LOR	<NR2>
	OOR (frame)	OOR	<NR1>
	FAS-LLD	FAS_LLD	<NR1>
OTU3, OTU4	LOF	LOF	<NR2>
	OOF (second)	OOF_S	<NR2>
	(frame)	OOF	<NR1>
	LOM	LOM	<NR2>
	OOM (second)	OOM_S	<NR2>
	(frame)	OOM	<NR1>
	SM-TIM	SM_TIM	<NR1>
	SM-BIAE (second)	SM_BIAE_S	<NR2>
	(frame)	SM_BIAE	<NR1>
	SM-BDI (second)	SM_BDI_S	<NR2>
	(frame)	SM_BDI	<NR1>
	SM-IAE (second)	SM_IAE_S	<NR2>
	(frame)	SM_IAE	<NR1>
	FAS	FAS	<NR1>

表A.4-1 測定項目とID (続き)

タブ	測定項目	ID	レスポンスデータ形式
Statistics (続き)	OTU3, OTU4 (続き)	SM-BIP8 (count)	SM_BIP8
		(rate)	SM_BIP8_R
		SM-BEI (count)	SM_BEI
		(rate)	SM_BEI_R
		FEC-Uncorr EBs (count)	FUEB
		(rate)	FUEB_R
		FEC-Corr Errors (count)	FCB
	ODU3, ODU4	(rate)	FCB_R
		FEC-Corr 0s to 1s	FCB0S
		FEC-Corr 1s to 0s	FCB1S
		MSIM	MSIM
		ODU-AIS (second)	AIS_ODU_S
		(frame)	AIS_ODU
		ODU-OCI (second)	OCI_S
		(frame)	OCI
		ODU-LCK (second)	LCK_S
		(frame)	LCK
		PM-TIM (frame)	PM_TIM
		PM-BDI (second)	PM_BDI_S
		(frame)	PM_BDI
		Client-AIS (second)	AIS_C_S
		(frame)	AIS_C
		PLM	PLM
		CSF	CSF
		PM-BIP8 (count)	PM_BIP8
		(rate)	PM_BIP8_R
		PM-BEI (count)	PM_BEI
		(rate)	PM_BIP8_R
	TCM	TCMi-TIM	TCM{1 2 3 4 5 6}_TIM_S
		TCMi-BIAE (second)	TCM{1 2 3 4 5 6}_BIAE_S
		(frame)	TCM{1 2 3 4 5 6}_BIAE
		TCMi-BDI (second)	TCM{1 2 3 4 5 6}_BDI_S
		(frame)	TCM{1 2 3 4 5 6}_BDI

付録

付録A

表A.4-1 測定項目とID (続き)

タブ	測定項目	ID	レスポンスデータ形式	
Statistics (続き)	TCM(続き)	TCMi-IAE (second)	TCM{1 2 3 4 5 6}_IAE_S	<NR2>
		(frame)	TCM{1 2 3 4 5 6}_IAE	<NR1>
		TCMi-LTC (second)	TCM{1 2 3 4 5 6}_LTC_S	<NR2>
		(frame)	TCM{1 2 3 4 5 6}_LTC	<NR1>
		TCMi-BIP8 (count)	TCM{1 2 3 4 5 6}_BIP8	<NR1>
		(rate)	TCM{1 2 3 4 5 6}_BIP8_R	<NR3>
		TCMi-BEI (count)	TCM{1 2 3 4 5 6}_BEI	<NR1>
		(rate)	TCM{1 2 3 4 5 6}_BEI_R	<NR3>
	ODU2e, ODU0	LOFLOM	LOFLOM	<NR2>
		OOF (second)	OOF_L_S	<NR2>
		(frame)	OOF_L	<NR1>
		OOM (second)	OOM_L_S	<NR2>
		(frame)	OOM_L	<NR1>
		ODU-AIS (second)	AIS_ODU_L_S	<NR2>
		(frame)	AIS_ODU_L	<NR1>
		ODU-OCI (second)	OCI_L_S	<NR2>
		(frame)	OCI_L	<NR1>
		ODU-LCK (second)	LCK_L_S	<NR2>
		(frame)	LCK_L	<NR1>
		PM-TIM (frame)	PM_TIM_L	<NR1>
		PM-BDI (second)	PM_BDI_L_S	<NR2>
		(frame)	PM_BDI_L	<NR1>
		PLM	PLM_L	<NR1>
		CSF	CSF_L	<NR1>
		FAS	FAS_L	<NR1>
		PM-BIP8 (count)	PM_BIP8_L	<NR1>
		(rate)	PM_BIP8_L_R	<NR3>
		PM-BEI (count)	PM_BEI_L	<NR1>
		(rate)	PM_BEI_L_R	<NR3>

表A.4-1 測定項目とID (続き)

タブ	測定項目	ID	レスポンスデータ形式
Statistics (続き)	GMP	Tx Inc 1	<NR1>
		Rx Inc 1	
		Tx Inc 2	<NR1>
		Rx Inc 2	
		Tx Dec 1	<NR1>
		Rx Dec 1	
		Tx Dec 2	<NR1>
		Rx Dec 2	
		Tx Inc >2	<NR1>
		Rx Inc >2	
		Tx Dec >2	<NR1>
		Rx Dec >2	
		Rx Inc Over	<NR1>
		Rx Dec Over	
		Offset (ppm)	<NR2>
		CRC8 Error	<NR1>
		CRC5 Error	<NR1>
	GMP (ODU2e, ODU0)	Tx Inc 1	<NR1>
		Rx Inc 1	
		Tx Inc 2	<NR1>
		Rx Inc 2	
		Tx Dec 1	<NR1>
		Rx Dec 1	
		Tx Dec 2	<NR1>
		Rx Dec 2	
		Tx Inc >2	<NR1>
		Rx Inc >2	
		Tx Dec >2	<NR1>
		Rx Dec >2	
		Rx Inc Over	<NR1>
		Rx Dec Over	
		Offset (ppm)	<NR2>
		CRC8 Error	<NR1>

表A.4-1 測定項目とID (続き)

タブ	測定項目	ID	レスポンスデータ形式
Statistics (続き)			
Statistics (続き)	GFP-T	Invalid GFP-T Frame	GFP_INV_FRM
		CSF Signal (count)	GFP_CSF_SIG
		CSF Sync (count)	GFP_CSF_SYNC
		SSF (second)	GFP_SSF
		Superblock CRC	GFP_SCRC
		Correctable cHEC	GFP_CHEC_C
		Uncorrectable cHEC	GFP_CHEC_U
		Correctable tHEC	GFP_THEC_C
		Uncorrectable tHEC	GFP_THEC_U
	PCS Lane	Marker Lock	AMLOCK
		Marker Map	MMAP
		BIP Error	BIP_CNT
		Invalid Alignment Marker	IAM
	Ethernet	Alignment Status	ALIGNMENT
		High-BER	HIBER
		Sync Header Lock	SHLOCK
		Invalid Sync Header	ISH
		Invalid Alignment Marker	IAM
		Invalid Block	IBLOCK
		66B Error	ERR_66B
		10B Error	ERR_10B
		Tx LF	TX_LF
		Rx LF	RX_LF
		Tx RF	TX_RF
		Rx RF	RX_RF
		Error Signals	RX_ERR_SIG
		Tx Errored Bytes	TEBYTE
		Rx Errored Bytes	REBYTE

表A.4-1 測定項目とID (続き)

タブ	測定項目	ID	レスポンスデータ形式
Statistics (続き)			
Ethernet (続き)	Tx Good Bytes	TGBYTE	<NR1>
	Rx Good Bytes	RGBYTE	
	Tx FCS Errors	TFCS	<NR1>
	Rx FCS Errors	RFCS	
	Tx Fragments	TFRAGMENT	<NR1>
	Rx Fragments	RFRAGMENT	
	Tx Oversize & FCS Errors	TOAF	<NR1>
	Rx Oversize & FCS Errors	ROAF	
	Tx Undersize	TUNDERSIZE	<NR1>
	Rx Undersize	RUNDERSIZE	
	Tx Oversize	TOVERSIZE	<NR1>
	Rx Oversize	ROVERSIZE	
	Tx Good Frames	TGFRAME	<NR1>
	Rx Good Frames	RGFRAME	
	Tx Rate (bit/s)	TBPS	<NR1>
	Rx Rate (bit/s)	RBPS	
	Tx Rate (%)	TRATE	<NR2>
	Rx Rate (%)	RRATE	
Test Pattern	Pattern Sync Loss (second)	PSLOSS	<NR2>
	Bit Errors (bit)	BER_CNT	<NR1>
	Bit Errors (rate)	BER_RATE	<NR3>
Delay ^{*1}	PM Current (μs)	PM_DELAY	<NR2>
	TCMi Current (μs)	TCM{1 2 3 4 5 6}_DELAY	<NR2>
APS ^{*2}	Current (ms)	APS_TIME	<NR2>

*1: Max, Min, Average, History, Tx Delay Frame, および Rx Delay Frame の値をリモート制御で読み取りできません。

*2: Max, Min, Average, および History の値をリモート制御で読み取りできません。

* Opt タブの ID は, 「A.3 40/100GbE アプリケーション」を参照してください。

A.5 No Frame アプリケーション

表A.5-1 測定項目とID

タブ	測定項目	ID	レスポンスデータ形式
Statistics	Elapsed Time (s)	ELAPSED	<NR1>
	Clock Source Loss	CSLOSS	<status>
	Pattern Sync Loss (s)	PSLOSS	<NR2>
	Bit Errors (Count)	BER_CNT	<NR1>
	Bit Errors (rate)	BER_RATE	<NR3>
	Frequency (Hz)	RX_FREQ	<NR1>
	Frequency (ppm)	RX_FREQ_D	<NR1>
	CDR Unlock	CUNLOCK	<status>
	クロック数 (bit) *	CLK_CNT	<NR1>

*: MD1260A の画面には表示されません。

クロック数 (bit) は、ビット誤り率の計算で使用する受信ビット数です。

$$\text{ビット誤り率 (BER)} = \frac{\text{エラービット数}}{\text{受信ビット数}}$$

Opt タブの ID は、「A.3 40/100GbE アプリケーション」を参照してください。

エラーを問い合わせる:SYSTEM:ERRor? に対するレスポンスのコードとメッセージを説明します。

- ・ コマンドエラー
- ・ 実行エラー
- ・ 装置固有エラー

これらのエラーが発生すると、標準イベントステータスレジスタのビットが 1 になります。標準イベントステータスイネーブルレジスタとサービスリクエストイネーブルレジスタを設定することにより、エラーが発生したときにサービスリクエストが発生することができます。

エラーが発生したときに、1 になる標準イベントステータスレジスタのビットは次のとおりです。

表B-1 エラー番号と標準イベントレジスタの関係

エラーコード	メッセージ	エラー名称	標準イベントレジスタのビット
-113	Undefined header	コマンドエラー	5
-220	Parameter error	実行エラー	4
-310	System error	装置固有エラー	3

コマンドエラー

以下のエラーが発生したときに、標準イベントステータスレジスタの bit 5 が設定されます。

- ・ 「2.5 メッセージの書式」の文法に従わないメッセージを送信したとき
例: ヘッダーを書き間違えた場合
2 バイト文字が含まれる場合
- ・ 「3.3 メッセージの説明」に記載している共通コマンド、および機器固有コマンドの説明に従わないメッセージを送信したとき
- ・ そのアプリケーションでは対応していないコマンドを送ったとき
- ・ クエリに?をつけなかったとき

実行エラー

以下のエラーが発生したときに、標準イベントステータスレジスタの bit 4 が設定されます。

- ・ ヘッダーに続くパラメータの値が、設定可能な範囲外の場合
例: 送信クロック周波数オフセットの設定範囲は-120～+120 であるのに、200 を設定した。
- ・ メッセージが本器の状態により実行できない場合
例: CFP が装着されていないときに、CFP の MDIO レジスタを設定するメッセージを送信した。

- ・ そのアプリケーションでは指定できないパラメータを設定したとき

装置固有エラー

以下のエラーが発生したときに、標準イベントステータスレジスタの bit 3 が設定されます。

- ・ :MMEMory:RECall のパラメータに読み取れないファイル名、または存在しないファイル名を設定したとき
- ・ :MMEMory:RECall のパラメータファイル名に拡張子を含めたとき

ソフトウェアのバージョンアップに伴って変更されたコマンドを説明します。

C.1 Ver.2.0 の変更内容

C.1.1 廃止されたコマンド

次の表に廃止されたコマンドと、その代替コマンドを示します。

表C.1.1-1 廃止されたコマンドと代替コマンド

廃止されたコマンド	代替コマンド
:CALCulate:COUNter:ETHer:BERRors	:ROUTe:BERT
:SOURce:STReam:DURation:FRAMes	:SOURce:STReam:DURation:REPeat:COUNt
:SOURce:STReam:HEADer:ETHer:SA	:SOURce:STReam:HEADer:PATtern
:SOURce:STReam:HEADer:ETHer:DA	:SOURce:STReam:HEADer:PATtern
:SOURce:STReam:HEADer:ETHer:TYPE	:SOURce:STReam:HEADer:PATtern

:CALCulate:COUNter:ETHer:BERRors

40GbE/100GbE アプリケーションにおいて、カウンタの Bit Errors Count の実行を設定/問い合わせする場合は、:ROUTe:BERT を使用します。

:SOURce:STReam:DURation:FRAMes

40GbE/100GbE アプリケーションにおいて、送信フレーム数を設定/問い合わせする場合は、:SOURce:STReam:DURation:REPeat:COUNt を使用します。

:SOURce:STReam:HEADer:ETHer:DA

:SOURce:STReam:HEADer:ETHer:SA

:SOURce:STReam:HEADer:ETHer:TYPE

送信するイーサネットフレームのあて先 MAC アドレス、送信元 MAC アドレス、および Type フィールドの値を設定/問い合わせする場合は、

:SOURce:STReam:HEADer:PATtern を使用します。

C.1.2 パラメータが変更されたコマンド

以下のコマンドのパラメータが変更されました。

:CALCulate:DATA

<item>に設定するパラメータの追加および削除

追加されたパラメータ: TX_LF, RX_LF, TX_RF, RX_RF, RX_ERR_SIG,
TX_TIME, RX_PAUSE, RX_TRIGGER, TX_FSIZE, RX_FSIZE,
TX_GSIZE, RX_GSIZE, TX_TFRAME, RX_TFRAME,
RX_SEQ_ERR, LAT_CURR, LAT_MAX, LAT_MIN,
TX_FREQ, TX_FREQ_D

削除されたパラメータ: TXRX

 付録A 測定項目一覧

:SOURce:STReam:CONTRol:VALue

パラメータの書式: <NR1>から<NR2>に変更

:SOURce:STReam:DURation:TYPE

変更前::SOURce:STReam:DURation:TYPE FRAME

変更後::SOURce:STReam:DURation:TYPE REPEAT

:SOURce:EALarm:LANE

10Lane の測定で:SOURce:EALarm:LANE? を送信した場合のレスポンスデータが 20 桁から 10 桁のデータに変更

変更前:20 桁のデータ 例 #B100000000000000000000

変更後:10 桁のデータ 例 #B1000000000

C.1.3 追加されたコマンド

以下のコマンドが追加されました。

Capture に関するコマンド

:CALCulate:CAPTure:DATA:RXC
:CALCulate:CAPTure:DATA:RXD
:CALCulate:CAPTure:DATA:SIZE
:CALCulate:CAPTure:DATA:TRIGger:POSition
:CALCulate:CAPTure:START
:CALCulate:CAPTure:START:EVENT
:CALCulate:CAPTure:STATus
:CALCulate:CAPTure:STOP
:CALCulate:CAPTure:TRIGgered
:CALCulate:TRIGger:CONDition
:MMEMory:STORe:CAPTure
:MMEMory:STORe:CAPTure:ITEM

Lane Mapping に関するコマンド

:SOURce:MAPPing:LANE

Port に関するコマンド

:ROUTe:FCONtrol
:ROUTe:LFS:REPLy

Relative Skew に関するコマンド

:SOURce:SKEW:TYPE

Stream に関するコマンド

:SOURce:STReam:BURSt:CONtrol:VALue
:SOURce:STReam:BURSt:ENABLE
:SOURce:STReam:BURSt:SIZE
:SOURce:STReam:COUNt
:SOURce:STReam:DURation:REPeat:COUNt
:SOURce:STReam:ENABLE
:SOURce:STReam:ERRor:TYPE
:SOURce:STReam:FSIZe
:SOURce:STReam:HEADer:PATtern
:SOURce:STReam:HEADer:VARiable1:RANGe
:SOURce:STReam:HEADer:VARiable1:TYPE
:SOURce:STReam:HEADer:VARiable2:RANGe
:SOURce:STReam:HEADer:VARiable2:TYPE
:SOURce:STReam:ID
:SOURce:STReam:TFRame:ENABLE
:SOURce:STReam:TFRame:FID
:SOURce:STReam:TYPE

Test Frame に関するコマンド

:CALCulate:COUNter:GAP

:CALCulate:COUNter:SERRor

サマリステータスに関するコマンド

:CALCulate:LINK[:AUNit]

マルチポート機能に関するコマンド

:CALCulate:COUNter:START:AUNit

:CALCulate:COUNter:START:EVENT:AUNit

:CALCulate:COUNter:STATus:AUNit

:CALCulate:COUNter:STOP:AUNit

:CALCulate:EALarm:AUNit

:CALCulate:LINK:AUNit

:MMEMory:INITialize

:MMEMory:RECall:AUNit

:MMEMory:STORe:AUNit

:SOURce:EALarm:START:AUNit

:SOURce:EALarm:START:EVENT:AUNit

:SOURce:EALarm:STATus:AUNit

:SOURce:EALarm:STOP:AUNit

:SOURce:STReam:START:AUNit

:SOURce:STReam:START:EVENT:AUNit

:SOURce:STReam:STATus:AUNit

:SOURce:STReam:STOP:AUNit

:UENTry:ID

:UENTry:LIST

C.2 Ver.3.0 の変更内容

C.2.1 廃止されたコマンド

以下のコマンドが廃止されました。

:CALCulate:LINK[:AUNit]
ステータスエリアの Link ランプの状態を問い合わせる場合は、
:CALCulate:DATA? LINK を使用します。

C.2.2 変更されたコマンド

以下のコマンドが変更されました。

表C.2.1-1 変更されたコマンド

変更前	変更後
:CALCulate:CAPTure:TRIGgered	:CALCulate:CAPTure:TRIGger

C.2.3 パラメータが変更されたコマンド

以下のコマンドのパラメータが変更されました。

パラメータの表記方法は、「表 3.1-1 コマンドの記述方法」を参照してください。

:CALCulate:MONitor:OTU:DATA
カラム位置、データ長を指定するパラメータを追加

:CALCulate:DATA
<item>に設定するパラメータの追加および削除
「付録 A 測定項目一覧」を参照してください。

変更されたパラメータ:

MLOFOTL_SEC → LOF_LLD
MOOF_FRM → OOF_LLD
MFAS_CNT → FAS_LLD
MOOR_FRM → OOR
OLOF_SEC → LOF
OOOF_FRM → OOF
OOOF_SEC → OOF_S
OFAS_CNT → FAS
OLOM_SEC → LOM
OOOM_FRM → OOM
OOOM_SEC → OOM_S
SMBIP8_CNT → SM_BIP8
SMBIP8_RATE → SM_BIP8_R
PMBIP8_CNT → PM_BIP8
PMBIP8_RATE → PM_BIP8_R

互換性維持のため変更前のパラメータでも制御できますが、使用しないでください。

追加されたパラメータ: AIS_C[_S], AIS_ODU[_L][_S], CSF, ERR_10B, ERR_66B, FAS_L, FREQ_D_P[_L], GFP_CHEC_C, GFP_CHEC_U, GFP_CSF, GFP_CSF_SIG, GFP_CSF_SYNC, GFP_INV_FRM, GFP_SCRC, GFP_SSF, GFP_THEC_C, GFP_THEC_U, GMP_CRC5, GMP_CRC8[_L], GMP_DEC_O[_L], GMP_DEC{1|2|3}[_L], GMP_DEC{1|2|3}_TX[_L], GMP_INC_O[_L], GMP_INC{1|2|3}[_L], GMP_INC{1|2|3}_TX[_L], LCK[_L][_S], LOFLOM, MSIM, OCI[_L][_S], OOF_L, OOM_L, PLM[_L], PM_BDI[_L][_S], PM_BEI[_L][_R], PM_BIP8_L, PM_TIM[_L][_S], SM_BDI[_S], SM_BEI[_R], SM_BIAE[_S], SM_IAE[_S], SM_TIM, TCM{1|2|3|4|5|6}_BDI[_S], TCM{1|2|3|4|5|6}_BEI[_R], TCM{1|2|3|4|5|6}_BIAE[_S], TCM{1|2|3|4|5|6}_BIP8[_R], TCM{1|2|3|4|5|6}_IAE[_S], TCM{1|2|3|4|5|6}_LTC[_S], TCM{1|2|3|4|5|6}_TIM

:MMEMory:CATalog

変更されたパラメータ:

100GE → E100G
100GE_N → E100G_N
40GE → E40G
40GE_N → E40G_N

追加されたパラメータ: OTU4_E, OTU4_ODU0, OTU4_ODU0_E, OTU4_ODU2E, OTU4_ODU2E_E

削除されたパラメータ: ALL

:ROUTE:MODE

追加されたパラメータ: THROUGH

:SOURce:MAPPing

変更されたパラメータ:

100GE → E100G
100GE_10N → E100G_10N
100GE_20N → E100G_20N
40GE → E40G
40GE_N, ALL → E40G_N

追加されたパラメータ: NONE, OTU4_E, OTU4_ODU0, OTU4_ODU0_E, OTU4_ODU2E, OTU4_ODU2E_E

:SOURce:EALarm:TIMing:RATE

レスポンスデータの書式が<CHAR>から<NR3>に変更

:SOURce:EALarm:TIMing:TYPE

変更されたパラメータ: CRATE, RRATE → RATE

:SOURce:EALarm:TYPE

変更されたパラメータ:

FAS_MLD → FAS_LLD

FAS_OTU → FAS

LOFOTL → OOF_LLD

LOF_OTU → OOF

LOM → OOM

OOR → LOR

追加されたパラメータ: AIS_C, AIS_ODU[_L], BIT_ALL_C,
BIT_ALL_U, CSF, ERR_10B, ERR_66B, GFP_CHEC, GFP_SCRC,
GFP_THEC, GMP_CRC5, GMP_CRC8, GMP_JC1, GMP_JC12,
GMP_JC2, LCK[_L], OCI[_L], PM_BDI[_L], PM_BEI[_L],
PM_BIP8_L, PM_TIM[_L], SM_BDI, SM_BEI, SM_BIAE, SM_IAE,
SM_TIM, TCM{1|2|3|4|5|6}_BDI, TCM{1|2|3|4|5|6}_BDI,
TCM{1|2|3|4|5|6}_BIAE, TCM{1|2|3|4|5|6}_BIP8,
TCM{1|2|3|4|5|6}_IAE, TCM{1|2|3|4|5|6}_LTC,
TCM{1|2|3|4|5|6}_TIM

付
録

付
録
C

C.2.4 追加されたコマンド

以下のコマンドが追加されました。

APS に関するコマンド

:CALCulate:APS:ERRor:FREE

:CALCulate:APS:STARt[:AUNit]

:CALCulate:APS:STARt:EVENT[:AUNit]

:CALCulate:APS:STATus[:AUNit]

:CALCulate:APS:STOP[:AUNit]

:CALCulate:APS:TRIGger

Capture に関するコマンド

:CALCulate:CAPTure:DATA

:CALCulate:CAPTure:TRIGger:Manual

:CALCulate:CAPTure:TRIGger:POSition

:CALCulate:CAPTure:TRIGger:TYPE

:CALCulate:CAPTure:TYPE

Clock に関するコマンド

:SOURce:CLOCK:PAYlaod:OFFSet[:L]

Counter に関するコマンド

:SENSe:PLM[:L]:Pattern

:SENSe:TIM:{SM|PM[:L]}TCM{1|2|3|4|5|6}}

:SENSe:TIM:{SM|PM[:L]}TCM{1|2|3|4|5|6}}:PATtern:{DAPI|SAPI}

Data Monitor に関するコマンド
:CALCulate:MONitor:OTU:TYPE

Delay に関するコマンド
:CALCulate:DElay:STARt[:AUNit]
:CALCulate:DElay:STARt:EVENT[:AUNit]
:CALCulate:DElay:STATus[:AUNit]
:CALCulate:DElay:STOP[:AUNit]

Error/Alarm に関するコマンド
:SOURce:EALarm:BIT
:SOURce:EALarm:FAS:EXCLude
:SOURce:EALarm:SUBRow

GFP-T に関するコマンド
:SOURce:GFP:CSF:REPLacement
:SOURce:GFP:{PTI|UPI}

OH Preset に関するコマンド
:SOURce:OTN:OH[:L]
Port に関するコマンド
:ROUTe:ETHer:NEGtiation:AUTO
:ROUTe:MODE:THRough:OVERwite:RANGe
:ROUTe:MODE:THRough:TYPE
:SOURce:OTN:FEC

Selector に関するコマンド
:SYSTem:CONFig
:SYSTem:STATus

Stream に関するコマンド
:SOURce:STReam:DURation:FRAMES
:SOURce:STReam:HEADer:ETHer:DA
:SOURce:STReam:HEADer:ETHer:SA
:SOURce:STReam:HEADer:ETHer:TYPE

TP/TS に関するコマンド
:SENSe:ODTU:MAIN:DETECT
:SENSe:ODTU:MAIN:TS
:SOURce:ODTU:DUMMy:PATtern
:SOURce:ODTU:MAIN:TP
:SOURce:ODTU:MAIN:TS

C.3 Ver.3.1 の変更内容

C.3.1 追加されたコマンド

以下のコマンドが追加されました。

Opt に関するコマンド

:SOURce:CFP:OPTical:OFF
:SOURce:CFP:OPTical:ON
:SOURce:CFP:OPTical:STATus

Port に関するコマンド

:ROUte:MPLStp:CWORD

Stream に関するコマンド

:MMEMory:STReam:RECall
:SOURce:STReam:HEADer
:SOURce:STReam:HEADer:VARiable{3|4|5}

Test Frames に関するコマンド

:CALCulate:COUNter:FLOW:FIELD
:CALCulate:COUNter:FLOW:FIELD:ID
:CALCulate:COUNter:FLOW:FIELD:NFID
:CALCulate:COUNter:FLOW:TYPE

C.3.2 変更されたコマンド

以下のコマンドが変更されました。

:SOURce:STReam:HEADer:VARiable1:TYPE

変更前	変更後
Ethernet の Destination MAC Address を設定します。	Modifiers の Field1 の Destination MAC Address 可変方法を設定します。

:SOURce:STReam:HEADer:VARiable2:TYPE

変更前	変更後
Ethernet の Source MAC Address を設定します。	Modifiers の Field2 の Source MAC Address 可変方法を設定します。

C.3.3 パラメータが変更されたコマンド

以下のコマンドのパラメータが変更されました。

パラメータの表記方法は、「表 3.1-1 コマンドの記述方法」を参照してください。

:CALCulate:DATA

<item>に設定するパラメータの追加

「付録 A 測定項目一覧」を参照してください。

追加されたパラメータ: CLK_CNT, RBCAST_BYTE, RBCAST_FRAME,
RMCAST_BYTE, RMCAST_FRAME, RMPLSTP, RX_TBPS,
RX_TBPS_O, RX_TFRAME_O, TBCAST_BYTE, TBCAST_FRAME,
TMCAST_BYTE, TMCAST_FRAME, TMPLSTP, TX_TBPS,
TX_TBPS_O, TX_TFRAME_O

:SOURce:STReam:HEADer:VARiable1:TYPE

:SOURce:STReam:HEADer:VARiable2:TYPE

パラメータに DECREMENT を追加

C.4 Ver.3.2 の変更内容

C.4.1 追加されたコマンド

以下のコマンドが追加されました。

Port に関するコマンド

:ROUTe:VLAN:NUM
:ROUTe:VLAN:VALue
:SENSe:OTN:FEC

Stream に関するコマンド

:SOURce:STReam:RESolve:IP:TARGet
:SOURce:STReam:RESolve:IPV4:ROUTer
:SOURce:STReam:RESolve:IPV6:ROUTer
:SOURce:STReam:RESolve:MAC:RETRy
:SOURce:STReam:RESolve:MAC:TIMEout
:SOURce:STReam:RESolve:PING:TRY
:SOURce:STReam:RESolve:PING:PAYLoad
:SOURce:STReam:RESolve:PING:TIMEout
:SOURce:STReam:RESolve:RESult
:SOURce:STReam:RESolve:STARt
:SOURce:STReam:RESolve:STARt:EVENT
:SOURce:STReam:RESolve:STATus
:SOURce:STReam:RESolve:STOP
:SOURce:STReam:RESolve:TYPE

Protocol に関するコマンド

:PROTOcol:ARPNa:REPLY:ENABle
:PROTOcol:GARPns:DURation:TYPE
:PROTOcol:GARPns:ENABle
:PROTOcol:GARPns:INTerval
:PROTOcol:GARPns:STARt
:PROTOcol:GARPns:STATus
:PROTOcol:GARPns:STOP
:PROTOcol:GARPns:TYPE
:PROTOcol:PING:COUNT
:PROTOcol:PING:DST:IPV4
:PROTOcol:PING:DST:IPV6
:PROTOcol:PING:DST:MAC
:PROTOcol:PING:IPMode
:PROTOcol:PING:MACResolve
:PROTOcol:PING:PAYLoad
:PROTOcol:PING:PSIZe:TYPE
:PROTOcol:PING:PSIZe:VALUe
:PROTOcol:PING:REPLY:ENABle
:PROTOcol:PING:RESult
:PROTOcol:PING:SRC:IPV4

:PROToCol:PING:SRC:IPV6
:PROToCol:PING:STARt
:PROToCol:PING:STARt:ENABle
:PROToCol:PING:STATus
:PROToCol:PING:STOP
:PROToCol:PING:TIMEout
:PROToCol:PING:VLAN

C.4.2 パラメータが変更されたコマンド

以下のコマンドのパラメータが変更されました。

パラメータの表記方法は、「表 3.1-1 コマンドの記述方法」を参照してください。

:CALCulate:COUNter:FLOW:FIELD

<field>に設定するパラメータの追加

追加されたパラメータ: ARP_SMAC, ARP_SIP, ARP_TMAC, ARP_TIP,
ARP_OPE, ICMPV4_CODE, ICMPV4_EC_ID ICMPV4_EC_SEQ,
ICMPV6_CODE, ICMPV6_EC_ID ICMPV6_EC_SEQ,
ICMPV6_NSNA_TADDR ICMPV6_NSNA_SADDR

:CALCulate:DATA

<item>に設定するパラメータの追加

「付録 A 測定項目一覧」を参照してください。

:SOURce:STReam:HEADer

<field>に設定するパラメータの追加

追加されたパラメータ: ARP, ICMPV4_ECHO, ICMPV6_ECHO,
ICMPV6_NS, ICMPV6_NA

:SOURce:STReam:HEADer:VARiable{3|4|5}

<field>に設定するパラメータの追加

追加されたパラメータ: ARP_SMAC, ARP_SIP, ARP_TMAC, ARP_TIP,
ARP_OPE, ICMPv4_CODE, ICMPV4_EC_ID ICMPV4_EC_SEQ,
ICMPV6_CODE, ICMPV6_EC_ID ICMPV6_EC_SEQ,
ICMPV6_NSNA_TADDR ICMPV6_NSNA_SADDR

付録D コマンドのアプリケーション対応表

コマンドごとにアプリケーション対応表を示します。

✓: 対応

–: 非対応

表D-1 コマンドのアプリケーション対応表

アプリケーション コマンド	100GbE	40GbE	ODU4-PRBS	ODU4-100GbE	ODU4-ODTU4.8-ODU2e-PRBS	ODU4-ODTU4.8-ODU2e-10GbE	ODU4-ODTU4.1-ODU0-PRBS	ODU4-ODTU4.1-ODU0-GbE	ODU3- PRBS	100GbE No Frame	40GbE No Frame	OTU4 No Frame	OTU3 No Frame
:CALCulate:APS:ERRor:FREE	–	–	✓	✓	✓	✓	✓	✓	✓	–	–	–	–
:CALCulate:APS:STARt[:AUNit]	–	–	✓	✓	✓	✓	✓	✓	✓	–	–	–	–
:CALCulate:APS:STARt:EVENT[:AUNit]	–	–	✓	✓	✓	✓	✓	✓	✓	–	–	–	–
:CALCulate:APS:STATus[:AUNit]	–	–	✓	✓	✓	✓	✓	✓	✓	–	–	–	–
:CALCulate:APS:STOP[:AUNit]	–	–	✓	✓	✓	✓	✓	✓	✓	–	–	–	–
:CALCulate:APS:TRIGger	–	–	✓	✓	✓	✓	✓	✓	✓	–	–	–	–
:CALCulate:CAPTure:DATA	–	–	✓	✓	✓	✓	✓	✓	✓	–	–	–	–
:CALCulate:CAPTure:DATA:RXC	✓	✓	–	–	–	–	–	–	–	–	–	–	–
:CALCulate:CAPTure:DATA:RXD	✓	✓	–	–	–	–	–	–	–	–	–	–	–
:CALCulate:CAPTure:DATA:SIZE	✓	✓	✓	✓	✓	✓	✓	✓	✓	–	–	–	–
:CALCulate:CAPTure:DATA:TRIGger:PO Sition	✓	✓	✓	✓	✓	✓	✓	✓	✓	–	–	–	–
:CALCulate:CAPTure:STARt	✓	✓	✓	✓	✓	✓	✓	✓	✓	–	–	–	–
:CALCulate:CAPTure:STARt:EVENT	✓	✓	✓	✓	✓	✓	✓	✓	✓	–	–	–	–
:CALCulate:CAPTure:STATus	✓	✓	✓	✓	✓	✓	✓	✓	✓	–	–	–	–
:CALCulate:CAPTure:STOP	✓	✓	✓	✓	✓	✓	✓	✓	✓	–	–	–	–
:CALCulate:CAPTure:TRIGger:MANual	–	–	✓	✓	✓	✓	✓	✓	✓	–	–	–	–
:CALCulate:CAPTure:TRIGger:PO Sition	–	–	✓	✓	✓	✓	✓	✓	✓	–	–	–	–
:CALCulate:CAPTure:TRIGger:TYPE	–	–	✓	✓	✓	✓	✓	✓	✓	–	–	–	–
:CALCulate:CAPTure:TRIGger	✓	✓	✓	✓	✓	✓	✓	✓	✓	–	–	–	–
:CALCulate:CAPTure:TYPE	–	–	✓	✓	✓	✓	✓	✓	✓	–	–	–	–
:CALCulate:COUNter:ETHer:OVERsize	✓	✓	–	✓	–	✓	–	✓	–	–	–	–	–
:CALCulate:COUNter:FLOW:FIELD	✓	✓	–	–	–	–	–	–	–	–	–	–	–

表D-1 コマンドのアプリケーション対応表(続き)

アプリケーション コマンド	100GbE	40GbE	ODU4-PRBS	ODU4-100GbE	ODU4-ODTU4.8-ODU2e-PRBS	ODU4-ODTU4.8-ODU2e-10GbE	ODU4-ODTU4.1-ODU0-PRBS	ODU4-ODTU4.1-ODU0-GbE	ODU3- PRBS	100GbE No Frame	40GbE No Frame	OTU4 No Frame	OTU3 No Frame
:CALCulate:COUNter:FLOW:FIELD:ID	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:CALCulate:COUNter:FLOW:FIELD:Nfid	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:CALCulate:COUNter:FLOW:TYPE	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:CALCulate:COUNter:GAP	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:CALCulate:COUNter:SERRor	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:CALCulate:COUNter:STARt[:AUNit]	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:CALCulate:COUNter:STARt:EVENT[:AUNit]	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:CALCulate:COUNter:STATus[:AUNit]	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:CALCulate:COUNter:STOP[:AUNit]	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:CALCulate:DATA	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:CALCulate:DATA:TYPE	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:CALCulate:DELay:STARt[:AUNit]	—	—	✓	✓	✓	✓	✓	✓	✓	—	—	—	—
:CALCulate:DELay:STARt:EVENT[:AUNit]	—	—	✓	✓	✓	✓	✓	✓	✓	—	—	—	—
:CALCulate:DELay:STATus[:AUNit]	—	—	✓	✓	✓	✓	✓	✓	✓	—	—	—	—
:CALCulate:DELay:STOP[:AUNit]	—	—	✓	✓	✓	✓	✓	✓	✓	—	—	—	—
:CALCulate:EALarm[:AUNit]	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:CALCulate:MONitor:OTU:COLumn	—	—	✓	✓	✓	✓	✓	✓	✓	—	—	—	—
:CALCulate:MONitor:OTU:DATA	—	—	✓	✓	✓	✓	✓	✓	✓	—	—	—	—
:CALCulate:MONitor:OTU:TYPE	—	—	✓	✓	✓	✓	✓	✓	✓	—	—	—	—
:CALCulate:TRIGger:CONDition	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:MDIO:READ	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:MDIO:WRITe	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:MMEMory:CATalog	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:MMEMory:INITialize	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

表D-1 コマンドのアプリケーション対応表(続き)

アプリケーション コマンド	100GbE	40GbE	ODU4-PRBS	ODU4-100GbE	ODU4-ODTU4.8-ODU2e-PRBS	ODU4-ODTU4.8-ODU2e-10GbE	ODU4-ODTU4.1-ODU0-PRBS	ODU4-ODTU4.1-ODU0-GbE	ODU3- PRBS	100GbE No Frame	40GbE No Frame	OTU4 No Frame	OTU3 No Frame
:MMEMory:LOG:FNAME	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:MMEMory:LOG:ITEM	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:MMEMory:LOG:PREFix	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:MMEMory:LOG:STARt	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:MMEMory:LOG:STATus	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:MMEMory:LOG:STOP	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:MMEMory:LOG:TIMing	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:MMEMory:RECall[:AUNit]	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:MMEMory:STORe[:AUNit]	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:MMEMory:STORe:CAPTurE	✓	✓	✓	✓	✓	✓	✓	✓	✓	—	—	—	—
:MMEMory:STORe:CAPTurE:ITEM	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:MMEMory:STReam:RECall	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:PROToCol:ARPNa:REPLy:ENABle	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:PROToCol:GARPns:DURation:TYPE	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:PROToCol:GARPns:ENABle	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:PROToCol:GARPns:INTerval	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:PROToCol:GARPns:STARt	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:PROToCol:GARPns:STATe	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:PROToCol:GARPns:STOP	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:PROToCol:GARPns:TYPE	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:PROToCol:PING:COUNT	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:PROToCol:PING:DST:IPV4	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:PROToCol:PING:DST:IPV6	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:PROToCol:PING:DST:MAC	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:PROToCol:PING:IPMode	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:PROToCol:PING:PAYLoad	✓	✓	—	—	—	—	—	—	—	—	—	—	—

表D-1 コマンドのアプリケーション対応表(続き)

アプリケーション コマンド	100GbE	40GbE	ODU4-PRBS	ODU4-100GbE	ODU4-ODTU4.8-ODU2e-PRBS	ODU4-ODTU4.8-ODU2e-10GbE	ODU4-ODTU4.1-ODU0-PRBS	ODU4-ODTU4.1-ODU0-GbE	ODU3- PRBS	100GbE No Frame	40GbE No Frame	OTU4 No Frame	OTU3 No Frame
:PROTOCOL:PING:PSIZE:TYPE	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:PROTOCOL:PING:PSIZE:VALUE	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:PROTOCOL:PING:REPLY:ENABLE	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:PROTOCOL:PING:RESULT	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:PROTOCOL:PING:SRC:IPV4	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:PROTOCOL:PING:SRC:IPV6	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:PROTOCOL:PING:SRC:MAC	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:PROTOCOL:PING:START	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:PROTOCOL:PING:START:ENABLE	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:PROTOCOL:PING:STATUS	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:PROTOCOL:PING:STOP	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:PROTOCOL:PING:TIMEOUT	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:PROTOCOL:PING:VLAN	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:ROUTE:BERT	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:ROUTE:ETHER:NEGOTIATION:AUTO	—	—	—	—	—	—	—	✓	—	—	—	—	—
:ROUTE:FCONTROL	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:ROUTE:LFS:REPLY	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:ROUTE:MODE	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:ROUTE:MODE:THROUGH:OVERWRITE:RANGE	—	—	✓	✓	✓	✓	✓	✓	✓	—	—	—	—
:ROUTE:MODE:THROUGH:TYPE	—	—	✓	✓	✓	✓	✓	✓	✓	—	—	—	—
:ROUTE:MPLSTP:CWORD	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:ROUTE:VLAN:NUM	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:ROUTE:VLAN:VALUE	✓	✓	—	—	—	—	—	—	—	—	—	—	—

表D-1 コマンドのアプリケーション対応表(続き)

アプリケーション コマンド	100GbE	40GbE	ODU4-PRBS	ODU4-100GbE	ODU4-ODTU4.8-ODU2e-PRBS	ODU4-ODTU4.8-ODU2e-10GbE	ODU4-ODTU4.1-ODU0-PRBS	ODU4-ODTU4.1-ODU0-GbE	ODU3- PRBS	100GbE No Frame	40GbE No Frame	OTU4 No Frame	OTU3 No Frame
:SENSe:MAPPing	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:SENSe:ODTU:MAIN:DETECT	—	—	—	—	✓	✓	✓	✓	—	—	—	—	—
:SENSe:ODTU:MAIN:TS	—	—	—	—	✓	✓	✓	✓	—	—	—	—	—
:SENSe:OTN:FEC	—	—	✓	—	✓	✓	✓	✓	✓	—	—	—	—
:SENSe:PLM[:L]:PATtern	—	—	—	✓	✓	✓	✓	✓	✓	—	—	—	—
:SENSe:TIM:{SM PM[:L] TCM{1 2 3 4 5 6}}	—	—	—	✓	✓	✓	✓	✓	✓	—	—	—	—
:SENSe:TIM:{SM PM[:L] TCM{1 2 3 4 5 6}}:PATtern:{DAPI SAPI}	—	—	—	✓	✓	✓	✓	✓	✓	—	—	—	—
:SENSe:TPATtern:INVert	—	—	✓	—	✓	—	✓	—	✓	✓	✓	✓	✓
:SENSe:TPATtern:TYPE	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:SENSe:TPATtern:WORD	—	—	✓	✓	✓	✓	✓	✓	✓	—	—	—	—
:SENSe:TRANsceiver:EQUalizer:CONTRol	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:SENSe:TRANsceiver:EQUalizer:DCGain	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:SOURce:CFP:OPTical:OFF	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:SOURce:CFP:OPTical:ON	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:SOURce:CFP:OPTical:STATUS	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:SOURce:CLOCK	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:SOURce:CLOCK:FREQuency:OFFSet	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:SOURce:CLOCK:OUTPut:DIVide	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:SOURce:CLOCK:OUTPut:M10	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:SOURce:CLOCK:PAYLoad:OFFSet[:L]	—	—	—	✓	✓	✓	✓	✓	—	—	—	—	—

表D-1 コマンドのアプリケーション対応表(続き)

アプリケーション コマンド	100GbE	40GbE	ODU4-PRBS	ODU4-100GbE	ODU4-ODTU4.8-ODU2e-PRBS	ODU4-ODTU4.8-ODU2e-10GbE	ODU4-ODTU4.1-ODU0-PRBS	ODU4-ODTU4.1-ODU0-GbE	ODU3- PRBS	100GbE No Frame	40GbE No Frame	OTU4 No Frame	OTU3 No Frame
:SOURce:EALarm:BIT	—	—	—	✓	✓	✓	✓	✓	—	—	—	—	—
:SOURce:EALarm:FAS:EXCLude	—	—	✓	—	—	—	—	—	✓	—	—	—	—
:SOURce:EALarm:LANE	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:SOURce:EALarm:STARt[:AUNit]	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:SOURce:EALarm:STARt:EVENT[:AUNit]	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:SOURce:EALarm:STATus[:AUNit]	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:SOURce:EALarm:STOP[:AUNit]	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:SOURce:EALarm:SUBRow	—	—	✓	—	—	—	—	—	✓	—	—	—	—
:SOURce:EALarm:TIMing:BURSt	✓	✓	✓	✓	✓	✓	✓	✓	✓	—	—	—	—
:SOURce:EALarm:TIMing:ERRor	✓	—	✓	✓	✓	✓	✓	✓	—	—	—	—	—
:SOURce:EALarm:TIMing:NORMal	✓	—	✓	✓	✓	✓	✓	✓	—	—	—	—	—
:SOURce:EALarm:TIMing:RATE	✓	✓	✓	✓	✓	✓	✓	✓	✓	—	—	—	—
:SOURce:EALarm:TIMing:TYPE	✓	✓	✓	✓	✓	✓	✓	✓	✓	—	—	—	—
:SOURce:EALarm:TYPE	✓	✓	✓	✓	✓	✓	✓	✓	✓	—	—	—	—
:SOURce:GFP:CSF:REPLacement	—	—	—	—	—	—	—	✓	—	—	—	—	—
:SOURce:GFP:{PTI UPI}	—	—	—	—	—	—	—	✓	—	—	—	—	—
:SOURce:MAPPing	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:SOURce:MAPPing:LANE	✓	✓	✓	✓	✓	✓	✓	✓	✓	—	—	—	—
:SOURce:ODTU:DUMMy:PATtern	—	—	—	—	✓	✓	✓	✓	—	—	—	—	—
:SOURce:ODTU:MAIN:TP	—	—	—	—	✓	✓	✓	✓	—	—	—	—	—
:SOURce:ODTU:MAIN:TS	—	—	—	—	✓	✓	✓	✓	—	—	—	—	—
:SOURce:OTN:FEC	—	—	—	✓	✓	✓	✓	✓	✓	—	—	—	—
:SOURce:OTN:OH[:L]	—	—	—	✓	✓	✓	✓	✓	✓	—	—	—	—
:SOURce:SKEW:BIT	✓	✓	✓	✓	✓	✓	✓	✓	✓	—	—	—	—
:SOURce:SKEW:LANE	✓	✓	✓	✓	✓	✓	✓	✓	✓	—	—	—	—
:SOURce:SKEW:NS	✓	✓	✓	✓	✓	✓	✓	✓	✓	—	—	—	—
:SOURce:SKEW:TYPE	✓	—	✓	✓	✓	✓	✓	✓	—	—	—	—	—

表D-1 コマンドのアプリケーション対応表(続き)

アプリケーション コマンド	100GbE	40GbE	ODU4-PRBS	ODU4-100GbE	ODU4-ODTU4.8-ODU2e-PRBS	ODU4-ODTU4.8-ODU2e-10GbE	ODU4-ODTU4.1-ODU0-PRBS	ODU4-ODTU4.1-ODU0-GbE	ODU3- PRBS	100GbE No Frame	40GbE No Frame	OTU4 No Frame	OTU3 No Frame
:SOURce:STReam:BURSt:CONTRol:VALue	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:SOURce:STReam:BURSt:ENABle	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:SOURce:STReam:BURSt:SIZE	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:SOURce:STReam:CONTRol:RANGe	✓	✓	—	✓	—	✓	—	✓	—	—	—	—	—
:SOURce:STReam:CONTRol:TYPE	✓	✓	—	✓	—	✓	—	✓	—	—	—	—	—
:SOURce:STReam:CONTRol:VALue	✓	✓	—	✓	—	✓	—	✓	—	—	—	—	—
:SOURce:STReam:COUNT	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:SOURce:STReam:DURation:FRAMES	✓	✓	—	✓	—	✓	—	✓	—	—	—	—	—
:SOURce:STReam:DURation:REPeat:CO UNT	✓	✓	—	✓	—	✓	—	✓	—	—	—	—	—
:SOURce:STReam:DURation:TYPE	✓	✓	—	✓	—	✓	—	✓	—	—	—	—	—
:SOURce:STReam:ENABle	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:SOURce:STReam:ERRor:TYPE	✓	✓	—	✓	—	✓	—	✓	—	—	—	—	—
:SOURce:STReam:FSIZe	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:SOURce:STReam:FSIZe:RANG	✓	✓	—	✓	—	✓	—	✓	—	—	—	—	—
:SOURce:STReam:FSIZe:TYPE	✓	✓	—	✓	—	✓	—	✓	—	—	—	—	—
:SOURce:STReam:FSIZe:VALue	✓	✓	—	✓	—	✓	—	✓	—	—	—	—	—
:SOURce:STReam:HEADer	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:SOURce:STReam:HEADer:ETHer:DA	✓	✓	—	✓	—	✓	—	✓	—	—	—	—	—
:SOURce:STReam:HEADer:ETHer:SA	✓	✓	—	✓	—	✓	—	✓	—	—	—	—	—
:SOURce:STReam:HEADer:ETHer:TYPE	✓	✓	—	✓	—	✓	—	✓	—	—	—	—	—
:SOURce:STReam:HEADer:PATTern	✓	✓	—	—	—	—	—	—	—	—	—	—	—

表D-1 コマンドのアプリケーション対応表(続き)

アプリケーション コマンド	100GbE	40GbE	ODU4-PRBS	ODU4-100GbE	ODU4-ODTU4.8-ODU2e-PRBS	ODU4-ODTU4.8-ODU2e-10GbE	ODU4-ODTU4.1-ODU0-PRBS	ODU4-ODTU4.1-ODU0-GbE	ODU3- PRBS	100GbE No Frame	40GbE No Frame	OTU4 No Frame	OTU3 No Frame
:SOURce:STReam:HEADer:VARiable1:RANGe	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:SOURce:STReam:HEADer:VARiable1:TYPE	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:SOURce:STReam:HEADer:VARiable2:RANGe	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:SOURce:STReam:HEADer:VARiable2:TYPE	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:SOURce:STReam:HEADer:VARiable{3 4 5}	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:SOURce:STReam:ID	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:SOURce:STReam:RESolve:IP:TARGet	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:SOURce:STReam:RESolve:IPV4:ROUTer	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:SOURce:STReam:RESolve:IPV6:ROUTer	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:SOURce:STReam:RESolve:MAC:RETRy	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:SOURce:STReam:RESolve:MAC:TIMEout	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:SOURce:STReam:RESolve:PING:TRY	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:SOURce:STReam:RESolve:PING:PAYLoad	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:SOURce:STReam:RESolve:PING:TIMEout	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:SOURce:STReam:RESolve:RESult	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:SOURce:STReam:RESolve:STARt	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:SOURce:STReam:RESolve:STARt:EVENT	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:SOURce:STReam:RESolve:STATus	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:SOURce:STReam:RESolve:STOP	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:SOURce:STReam:RESolve:TYPE	✓	✓	—	—	—	—	—	—	—	—	—	—	—

表D-1 コマンドのアプリケーション対応表(続き)

アプリケーション コマンド	100GbE	40GbE	ODU4-PRBS	ODU4-100GbE	ODU4-ODTU4.8-ODU2e-PRBS	ODU4-ODTU4.8-ODU2e-10GbE	ODU4-ODTU4.1-ODU0-PRBS	ODU4-ODTU4.1-ODU0-GbE	ODU3- PRBS	100GbE No Frame	40GbE No Frame	OTU4 No Frame	OTU3 No Frame
:SOURce:STReam:STARt[:AUNit]	✓	✓	—	✓	—	✓	—	✓	—	—	—	—	—
:SOURce:STReam:STARt:EVENT[:AUNit]	✓	✓	—	✓	—	✓	—	✓	—	—	—	—	—
:SOURce:STReam:STATus[:AUNit]	✓	✓	—	✓	—	✓	—	✓	—	—	—	—	—
:SOURce:STReam:STOP[:AUNit]	✓	✓	—	✓	—	✓	—	✓	—	—	—	—	—
:SOURce:STReam:TFRame:ENABLE	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:SOURce:STReam:TFRame:FID	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:SOURce:STReam:TYPE	✓	✓	—	—	—	—	—	—	—	—	—	—	—
:SOURce:TPATtern:INVert	—	—	✓	—	✓	—	✓	—	✓	✓	✓	✓	✓
:SOURce:TPATtern:TYPE	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:SOURce:TPATtern:WORD	✓	✓	✓	✓	✓	✓	✓	✓	✓	—	—	—	—
:SOURce:TRANsceiver:EMPHasis:FIRSt	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:SOURce:TRANsceiver:EMPHasis:PRE	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:SOURce:TRANsceiver:EMPHasis:SECOnd	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:SOURce:TRANsceiver:VOD	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:SYSTem:CONFig	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:SYSTem:DATE	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:SYSTem:ERRor	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:SYSTem:PRINt:COPIY	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:SYSTem:STATus	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:SYSTem:TERMination	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:SYSTem:TIME	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:SYSTem:VERSion	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:UENTry:ID	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
:UENTry:LIST	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

付録

付録D

ここでは、Tera Term のマクロ機能を使用したサンプルプログラムを説明します。

E.1 サンプルプログラムの実行方法

1. Windows のメモ帳などのテキストエディタを起動します。
2. 本書のサンプルプログラムをコピーします。
3. コピーしたサンプルプログラムをテキストエディタに貼り付けします。
4. Tera Term マクロファイルの形式(拡張子 `tcl`)でファイルを保存します。
5. Tera Term を起動します。
6. 「2.7.1 イーサネットの場合(Windows 7/Vista)」を参照して、本器と通信できることを確認します。
7. Tera Term のメニューから [コントロール] - [マクロ] をクリックします。
8. ファイル選択画面が開きます。
4.で保存したファイルを選択します。

このほかのマクロの実行方法については、Tera Term のヘルプを参照してください。

E.2 例 1:ストリーム設定とカウンタの読み取り

このサンプルプログラムは、100GbE, または 40GbE アプリケーションのストリームを設定し、カウンタの値を読み取ります。

処理の流れ

1. ユニット ID が 1 の MD1260A を制御対象とします。
2. `:SOURce:MAPPing?`で、アプリケーションを問い合わせます。
アプリケーションが 100GbE, または 40GbE でない場合は、メッセージボックスを表示して処理を中止します。
3. Port を次のとおり設定します。
Mode: Normal
Bert: Off
LFS Reply: On
Flow Control: On
Rx MPLS-TP Control Word: Off
4. ステータスエリアの Link ランプを問い合わせます。
5. ストリーム 1 から 16 まで次のとおり設定します。
フレームサイズ: Random, 128~1518 (byte)
ギャップサイズ: Random, 2000~20000 (byte)
6. ストリーム 1 から 16 までを有効にして、ストリームを送信します。
7. カウンタを開始します。
8. カウンタが開始したか、1 秒おきに問い合わせます。
9. カウンタが開始されたら、経過時間を 1 秒おきに問い合わせます。
10. 経過時間が 10 秒以上になったら、次のデータを問い合わせます。
Opt タブ: LOS
Statistics タブ: Clock Source Loss, Rx Frequency (Hz), Rx Frequency (ppm), Bit Errors (rate), Bit Errors (Count)

```
; sample program for MD1260A ver 3.1
; Anritsu Corporation April, 2012
;
; set local echo to on
setecho 1
flushrecv
; specify module number of MD1260A
sendln ':UENT:ID 1'

; time out 3 second
timeout=3

; query application
sendln ':SOURce:MAPPING?'
waitln 'E100G' 'E40G'

result_mapp=result
if result_mapp=0 goto _timeout
if result_mapp=3 then
    messagebox 'Application is not 40GbE nor 100GbE' 'MD1260A Application'
end
endif

call check_error_code

; set port as normal mode
sendln ':ROUTe:MODE NORMAL'
call check_error_code
; set frame BERT off
sendln ':ROUTe:BERT 0'
call check_error_code
; set LFS reply on
sendln ':ROUTe:LFS:REPLY 1'
call check_error_code
; set flow control on
sendln ':ROUTe:FCONTROL 1'
call check_error_code
; set Rx MPLS-TP Control Word to off
sendln ':ROUTe:MPLStp:CWORD 0'
call check_error_code
pause 2

call check_link

; stream configuration
for id 1 16
    int2str id_str id
```

```

    specify_id=':SOURCE:STReam:ID '
    strconcat specify_id id_str
    sendln specify_id
    call check_error_code
    sendln ':SOURCE:STReam:CONTRol:TYPE RANDOM'
    call check_error_code
    sendln ':SOURCE:STReam:CONTRol:RANGe 2000,200000'
    call check_error_code
    sendln ':SOURCE:STReam:FSIZe:TYPE RANDOM'
    call check_error_code
    sendln ':SOURCE:STReam:FSIZe:RANGe 128,1518'
    call check_error_code
next

sendln ':SOURCE:STReam:ENABLe 1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16'
call check_error_code
sendln ':SOURCE:STReam:STARt'
call check_error_code

; check counter status to clear buffer
sendln ':CALCulate:COUNter:STARt:EVENT?'
pause 1 ; wait 1 second
waitln '0' '1'
call check_error_code

; start counter
sendln ':CALCulate:COUNter:STARt'
call check_error_code
pause 3

; query counter status
for i 1 10
    sendln ':CALCulate:COUNter:STARt:EVENT?'
    pause 1; wait 1 second
    waitln '0' '1'
    counter_start=result
    if result=0 goto _timeout
    if result=2 break
    call check_error_code
next

if counter_start=1 then
    messagebox 'Counter is not ready' 'Warning !'
endif

; waiting until elapsed time reaches 10 second
for i 1 20

```

```

    sendln ':CALCulate:DATA? ELAPSED'
    pause 1; wait 1 second
    recvln
    recvln
    ;call check_response ; debug
    if result=0 goto _timeout
    if result=1 then
        str2int e_time inputstr
        if e_time>9 break
    endif
    call check_error_code
next

; data acquisition
sendln ':CALCulate:DATA? LOS,CSLOSS,RX_FREQ,RX_FREQ_D,BER_CNT,BER_RATE'
call check_error_code
call check_alarm

messagebox 'Macro end successfully' 'Finish'

End

;      ----- subroutines -----

:_timeout
    messagebox 'No response from MD1260A.' 'Time out!'
    call check_error_code
    End

:check_link
    ; check status lamp of link
    sendln ':CALCulate:DATA? LINK'
    waitln '0' '1' '2'
    link_state=result
    ; in case of timeout
    if result=0 goto _timeout
    ; in case of link up
    ;if result=1  messagebox 'Link lamp of status area is lit' 'Link Up'
    ; in case of link down
    if result=2 messagebox 'Link lamp of status area is not lit' 'Link Down'

    call check_error_code

return

:check_alarm

```

```

; in case of link, check status lamp of error
sendln ':CALCulate:EALarm?'
waitln '0' '1' '2'

; in case of timeout
if result=0 then
    goto _timeout
; in case of no error
elseif result=1 then
    messagebox 'Error lamp of status area is not lit' 'No error occurring'
; in case of error occurring
elseif result=2 then
    messagebox 'Error lamp of status area is red' 'Error occurring'
end
elseif result=3 then
    messagebox 'Error lamp of status area is orange' 'No error occurring
currently'
endif

call check_error_code

return

:check_error_code
; query error
sendln ':SYSTem:ERRor?'
waitln 'No error'

; in case of timeout
if result=0 goto _timeout
; in case of error occurring
if result=2 then
    e_message='Error code = '
    strconcat e_message inputstr
    messagebox e_message 'Command Error occurred'
end
endif

; in case of no error

return

:check_response

;for debug
messagebox inputstr 'debug1'
int2str result_str result

```

```
messagebox result_str 'debug2'  
  
return
```

E.3 例 2:エラー挿入とカウンタの読み取り

このサンプルプログラムは、OTU4またはOTU3アプリケーションのエラーを設定し、カウンタの値を読み取ります。

処理の流れ

1. ユニット ID が 1 の MD1260A を制御対象とします。
2. :SOURce:MAPPing?で、アプリケーションを問い合わせます。
アプリケーションが OTU4, または OTU3 でない場合は、メッセージボックスを表示して処理を中止します。
3. エラーを次のとおり設定します。
Type: ErrorODU4-PM-BIP8
Timing: Rate, 10E-8
4. エラー挿入を開始します。
5. エラー挿入が開始したか, 1 秒おきに問い合わせます。
6. カウンタを開始します。
7. カウンタが開始したか, 1 秒おきに問い合わせます。
8. カウンタが開始されたら, 経過時間を 1 秒おきに問い合わせます。
9. 経過時間が 20 秒以上になったら, 次のデータを問い合わせます。
Summary タブ: LOS, Clock Source Loss, Rx Frequency (Hz),
Rx Frequency (ppm)
Statistics タブ: PM-BIP8 (count)


```
; sample program for MD1260A ver 3.0
; Anritsu Corporation June,2011
;; set local echo to on
setecho 1
flushrecv
; specify module number of MD1260A
sendln ':UENT:ID 1'

; time out 3 second
timeout=3

; query application
sendln ':SOURce:MAPPING?'
waitln 'OTU3' 'OTU4'

result_mapp=result
if result_mapp=0 goto _timeout
if result_mapp=3 then
    messagebox 'Application is not OTU3 nor OTU4.' 'MD1260A Application'
end
endif

call check_error_code

; set port parameter
sendln ':ROUTE:MODE NORMAL'
call check_error_code
sendln ':SOURce:OTN:FEC OFF'
call check_error_code

pause 1

; stream configuration
sendln ':SOURce:EALarm:TYPE PM_BIP8'
call check_error_code
sendln ':SOURce:EALarm:TIMing:TYPE RATE'
call check_error_code
sendln ':SOURce:EALarm:TIMing:RATE R1E_8'
call check_error_code
sendln ':SOURce:EALarm:START:EVENT?'; clear event buffer
pause 1 ; wait 1 second
waitln '0' '1'
call check_error_code

; start error insertion
sendln ':SOURce:EALarm:START'
call check_error_code
```

```
pause 2

; query error insertion status
for i 1 10
    sendln ':SOURce:EALarm:START:EVENT?'
    pause 1; wait 1 second
    waitln '0' '1'
    error_ins_start=result
    if result=0 goto _timeout
    if result=2 break
    call check_error_code
next

if error_ins_start=1 then
    messagebox 'Error insertion does not start' 'Warning !'
endif

; check counter status to clear buffer
sendln ':CALCulate:COUNTER:START:EVENT?'

waitln '0' '1'
call check_error_code

; start counter
sendln ':CALCulate:COUNTER:START'
call check_error_code
pause 3

; query counter status
for i 1 10
    sendln ':CALCulate:COUNTER:START:EVENT?'
    pause 1; wait 1 second
    waitln '0' '1'
    counter_start=result
    if result=0 goto _timeout
    if result=2 break
    call check_error_code
next

if counter_start=1 then
    messagebox 'Counter is not ready' 'Warning !'
endif

; waiting until elapsed time reaches 15 second
for i 1 20
```

```

    sendln ':CALCulate:DATA? ELAPSED'
    pause 1; wait 1 second
    recvln
    recvln
    ;call check_response ; debug
    if result=0 goto _timeout
    if result=1 then
        str2int e_time inputstr
        if e_time>19 break
    endif
    call check_error_code
next

; data acquisition
sendln ':CALCulate:DATA? LOS,CSLOSS,RX_FREQ,RX_FREQ_D,PM_BIP8'
call check_error_code
call check_alarm

messagebox 'Macro end successfully' 'Finish'

End

;      ----- subroutines -----

:check_response

    ;for debug
    int2str result_str result
    messagebox result_str 'debug1'

    return

:_timeout
messagebox 'No response from MD1260A.' 'Time out!'
End

:check_alarm
    ; in case of link, check status lamp of error
    sendln ':CALCulate:EALarm?'
    waitln '0' '1' '2'

    ; in case of timeout
    if result=0 then
        goto _timeout
    ; in case of no error
    elseif result=1 then
        messagebox 'Error lamp of status area is not lit' 'No error occurring'

```

```
        end
    ; in case of error occurring
elseif result=2 then
    messagebox 'Error lamp of status area is red' 'Error occurring'
elseif result=3 then
    messagebox 'Error lamp of status area is orange' 'No error occurring
currently'
    end
endif

call check_error_code

return

:check_error_code
    ; query error
    sendln ':SYSTem:ERRor?'
    waitln 'No error'

    ; in case of timeout
    if result=0 goto _timeout
    ; in case of error occurring
    if result=2 then
        e_message='Error code = '
        strconcat e_message inputstr
        messagebox e_message 'Command Error occurred'
        end
    endif

    ; in case of no error

return
```

E.4 例 3:キャプチャとデータ保存

このサンプルプログラムは、100GbE、または 40GbE アプリケーションでキャプチャしたデータをファイルに保存します。

処理の流れ

1. ユニット ID が 1 の MD1260A を制御対象とします。
2. :SOURce:MAPPing?で、アプリケーションを問い合わせます。
アプリケーションが 100GbE、または 40GbE でない場合は、メッセージボックスを表示して処理を終了します。
3. ノーマルモードに設定した後、ステータスエリアの Link ランプを問い合わせます。
4. Link ランプが緑色であれば、ステータスエリアの Error/Alarm ランプを問い合わせます。
5. Error/Alarm ランプが赤色でなければ、Port の設定をします。
Mode: Normal
Bert: Off
LFS Reply: On
Flow Control: On
Rx MPLS-TP Control Word: Off
6. Lane Mapping を設定します。
7. ストリーム 1 から 16 まで次のとおり設定します。
フレームサイズ: Random, 128~1600 (byte)
ギャップサイズ: Random, 2000~20000 (byte)
8. ストリーム 1 から 16 までを有効にします。
9. カウンタの状態を読み取ります。
10. カウンタを開始します。
11. カウンタの状態を読み取ります。
12. キャプチャデータが取得されたら、日付と時刻をファイル名につけて保存します。

```
; sample program for MD1260A ver 3.1
; Anritsu Corporation April,2012
;
; set local echo to on
setecho 1
flushrecv
; specify module number of MD1260A
sendln ':UENT:ID 1'

; time out 3 second
timeout=3

; query application
sendln ':SOURce:MAPPing?'
waitln 'E100G' 'E40G'

result_mapp=result
if result_mapp=0 goto _timeout
if result_mapp=3 then
    messagebox 'Application is not 40GbE nor 100GbE' 'MD1260A Application'
end
endif

call check_error_code

; set port as normal mode
sendln ':ROUTe:MODE NORMAL'
call check_error_code
; set frame BERT off
sendln ':ROUTe:BERT 0'
call check_error_code
; set LFS reply on
sendln ':ROUTe:LFS:REPLY 1'
call check_error_code
; set flow control on
sendln ':ROUTe:FCONtrol 1'
call check_error_code
; set Rx MPLS-TP Control Word to off
sendln ':ROUTe:MPLStp:CWORD 0'
call check_error_code

pause 2

call check_link
call check_alarm

; set lane mapping
```

```

if result_mapp=2 sendln ':SOURCE:MAPPING:LANE 0,1,2,3'
if result_mapp=1 sendln ':SOURCE:MAPPING:LANE
0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19'
call check_error_code

; stream configuration
for id 1 16
int2str id_str id
spcify_id=':SOURCE:STReam:ID '
strconcat spcify_id id_str
sendln spcify_id
call check_error_code
sendln ':SOURCE:STReam:CONTRol:TYPE RANDOM'
call check_error_code
sendln ':SOURCE:STReam:CONTRol:RANGe 2000,200000'
call check_error_code
sendln ':SOURCE:STReam:FSIZe:TYPE RANDOM'
call check_error_code
sendln ':SOURCE:STReam:FSIZe:RANGe 128,1600'
call check_error_code
next
sendln ':SOURCE:STReam:ENABLe 1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16'
call check_error_code

; set trigger to oversize
sendln ':CALCulate:TRIGger:CONDition #B00000001'
pause 1 ; wait 1 second
call check_error_code

; clear event buffer
sendln ':CALCulate:CAPTure:STARt:EVENT?'
call check_error_code

; start capture
sendln ':CALCulate:CAPTure:STARt'
call check_error_code
flushrecv

for i 1 10
  pause 1
  sendln ':CALCulate:CAPTure:STARt:EVENT?'
  waitln '0' '1'
  capture_result=result
  if result=0 goto _timeout
  if result=2 break
  call check_error_code

```

```

next

if capture_result=1 then
    messagebox "Capture does not start!" 'Capture error'
end
endif

; stream start
sendln ':SOURce:STReam:STARt'
call check_error_code

; wait trigger of capture
for i 1 10
    pause 1
    sendln ':CALCulate:CAPTure:STATus?'
    waitln '0' '1'
    capture_result=result
    ;call check_response
    if result=0 goto _timeout
    if result=1 break
    call check_error_code
next

if capture_result=2 then
    messagebox "Capture trigger doesn't raise!" 'Capture error'
end
endif

; save capture data to file
getdate filename "macro-%Y%m%d-%H%M%S"
header=':MMEMory:STORe:CAPTure '
strconcat header filename

sendln ':MMEMory:STORe:CAPTure:ITEM LIBPCAP'
call check_error_code
sendln header
call check_error_code

messagebox 'Macro end successfully' 'Finish'

End

:_timeout
messagebox 'No response from MD1260A.' 'Time out!'
End

:check_link

```



```

; check status lamp of link
flushrecv
sendln ':CALCulate:DATA? LINK'
waitln '0' '1' '2'
call check_response
link_state=result
; in case of timeout
if result=0 goto _timeout
; in case of link up
; if result=1  messagebox 'Link lamp of status area is lit' 'Link Up'
; in case of link down
if result=2 messagebox 'Link lamp of status area is not lit' 'Link Down'

call check_error_code

return

:check_alarm
; in case of link, check status lamp of error
sendln ':CALCulate:EALarm?'
waitln '0' '1' '2'

; in case of timeout
if result=0 then
    goto _timeout
; in case of no error
elseif result=1 then
    ;messagebox 'Error lamp of status area is not lit' 'No error occurring'
; in case of error occurring
elseif result=2 then
    messagebox 'Error lamp of status area is red' 'Error occurring'
    end
elseif result=3 then
    messagebox 'Error lamp of status area is orange' 'No error occurring
currently'
endif

call check_error_code

return

:check_error_code
; query error
sendln ':SYSTem:ERRor?'
waitln 'No error'

; in case of timeout

```

```
if result=0 goto _timeout
; in case of error occurring
if result=2 then
    e_message='Error code = '
    strconcat e_message inputstr
    messagebox e_message 'Command Error occurred'
end
endif

; in case of no error

return

:check_response

;for debug
messagebox inputstr 'debug1'
int2str result_str result
messagebox result_str 'debug2'

return
```

■ アルファベット順

C

CFP MDIO Analysis 3-40

G

GPIO インタフェース 2-2

GPIO ケーブルの接続 2-4

GPIO を設定する 2-8

L

Lambda Measure 3-40

O

Operation Record Panel 4-3

R

RFC2544 3-40

S

Service Disruption 3-40

■ 50音順

い

イーサネットインタフェース 2-2

イーサネットを設定する 2-5

き

機器固有コマンド 2-18

共通コマンド 2-18, 3-42

く

クエリ 2-15

クロスケーブル 2-3

こ

コマンド 2-15

コマンドエラー 2-21, B-1

コマンドの記述方法 3-2

し

実行エラー 2-22, B-1

す

数値データ 2-16

ステータスバイトレジスタ 2-19

ステータスバイトレジスタのビット定義 2-20

ストレートケーブル 2-3

せ

接続を確認する 2-9

そ

操作記録機能 4-2

装置固有エラー B-2

測定項目一覧 A-2

て

データ 2-16

は

バイナリデータ 2-17

パネル操作が無いメッセージ 3-40, 3-41

ひ

標準イベントステータスレジスタ 2-21

標準イベントレジスタのビット定義 2-21

ふ

プログラムメッセージ 2-15

へ

ヘッダ 2-16

め

メッセージの構造 2-16

メッセージの種類 2-15

メッセージの書式 2-15

も

文字データ 2-16

よ

用語 1-4

れ

レスポンスデータ形式 A-1

レスポンスメッセージ 2-15